Provably Accurate Shapley Value Estimation in High Dimensions via Sparse Leverage Sampling

Ricardo Parada[1], **Akbarkhuja Anvarkhujaev**[1], **Carlos Martin**[1], **William Chang**[1,†], **Long Tran-Thanh**[2], **Nam Phuong Tran**[3,†]
[1]Department of Applied Mathematics, University of California, Los Angeles, USA
[2]Department of Computer Science, University of Warwick, Coventry, United Kingdom
[3]Inria centre at the University of Lille, Lille, France

Abstract

Computing Shapley values exactly requires exponentially many evaluations of a cooperative value function, making approximation essential in high-dimensional applications. Recent leverage-score-based methods yield provably accurate estimators with sample complexity scaling linearly in the number of players, but this can still be too costly when only a small subset of players is truly influential. We study Shapley value estimation under a sparse influence model, where the Shapley vector is supported on an unknown set of size $s \ll n$. Using the regression characterization of Shapley values, we propose a sparse leverage sampling scheme and analyze it via sparse matrix concentration. Our main result shows that the resulting estimator achieves relative-error approximation with $\widetilde{O}(s^2)$ value function evaluations, up to logarithmic and accuracy factors, thereby replacing the ambient-dimension dependence of prior guarantees with dependence on the intrinsic support size. Empirically, our method matches or improves over dense Leverage SHAP on sparse synthetic and real-data benchmarks, while exhibiting better numerical stability at small sample sizes.

1 Introduction

As AI is increasingly deployed in high-stakes domains like healthcare, finance, and law, transparency in model behavior is crucial for building trust and identifying areas of improvement. To address this, Shapley values have emerged as a powerful game-theoretic approach in explainable AI (XAI) for interpreting opaque models [Shapley, 1951, Štrumbelj and Kononenko, 2014, Datta et al., 2016, Lundberg and Lee, 2017], quantifying the contribution of each input feature to a model's output.

Shapley values originate as a method for determining fair payouts in a cooperative game involving n players, by measuring how a player's addition to a set of others changes the value of the game. Formally, let the value function $v : 2^{[n]} \to \mathbb{R}$ be defined on sets $S \subseteq [n]$. The Shapley value for player i is

$$\phi_i = \frac{1}{n} \sum_{S \subseteq [n] \setminus \{i\}} \frac{v(S \cup \{i\}) - v(S)}{\binom{n-1}{|S|}}. \tag{1}$$

Shapley values are the unique values satisfying four desirable game-theoretic properties: Null Player, Symmetry, Additivity, and Efficiency [Shapley, 1951]; see [Molnar, 2020] for further background. In explainable AI, Shapley values are used to attribute predictions of a model f on input $\mathbf{x}$ relative to a baseline $\mathbf{y}$, by taking the players to be features and $v(S) = f(\mathbf{x}^S)$ where $x_i^S = x_i$ for $i \in S$

[†]Correspondence to phuong-nam.tran@inria.fr and chang314@g.ucla.edu.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

and $x_i^S = y_i$ otherwise [Lundberg and Lee, 2017].[1] They also find use in feature and data selection [Marcílio and Eler, 2020, Fryer et al., 2021, Jia et al., 2019, Ghorbani and Zou, 2019].

Although principled, computing Shapley values exactly requires exponentially many evaluations of v. The recent leverage-score refinement [Musco and Witter, 2025] of Kernel SHAP [Lundberg and Lee, 2017] brings the sample complexity down to $\widetilde{O}(n)$ value-function evaluations, but this linear dependence on n can still be prohibitive in high-dimensional settings. In many applications, however, only a small subset of players is truly influential while the rest are effectively dummies. This is particularly natural in explainable AI: high-dimensional inputs often contain many redundant or weakly relevant features, while a model's prediction may depend on only a few dominant signals. In tabular, biomedical, or text applications, a local decision is often driven by a small number of key covariates, biomarkers, or tokens, and sparse explanations therefore identify the features that strongly affect model behavior [Marcílio and Eler, 2020, Fryer et al., 2021]. Motivated by this, we ask whether this *sparse influence* structure can be exploited to reduce query complexity below the ambient dimension.

Our Contributions. Under a sparse influence model where the Shapley vector is supported on an unknown set of size $s \ll n$:

- We propose *Sparse-Leverage SHAP*, an estimator that combines the Kernel SHAP regression framework with a novel restricted-leverage sampling distribution tailored to sparse Shapley vectors. We prove a sparse matrix concentration result (Lemma 2) that holds uniformly over all supports of size at most s.
- We bound the restricted leverage mass of the Shapley design (Proposition 3) by $L_s = O(s \log n)$, the key structural fact driving sparse sample complexity. We further give a closed form for the restricted leverage scores (Proposition 1), so that sampling is tractable. Combining these yields relative-error approximation using $\widetilde{O}(s^2)$ evaluations (Theorem 6), replacing the $\widetilde{O}(n)$ dependence of [Musco and Witter, 2025] with the sparsity level.
- We further propose a computationally efficient variant based on Iterative Hard Thresholding (IHT), which replaces the exact ℓ_0 sparse regression step with a polynomial-time iterative solver while retaining provable statistical guarantees.
- Empirically, our estimator matches or exceeds dense Leverage SHAP on strongly-sparse problems and exhibits substantially improved numerical stability at small sample sizes.

Related Work. **Shapley value estimation.** The most direct approach is to subsample (1) [Castro et al., 2009, Strumbelj and Kononenko, 2010, Štrumbelj and Kononenko, 2014, Illés and Kerényi, 2019, Mitchell et al., 2022], but achieving sample reuse across players is hard since each term involves both $v(S)$ and $v(S \cup \{i\})$ [Wang and Jia, 2023]. Kernel SHAP solves a global regression problem and reuses subsets across all players; [Covert and Lee, 2021] analyze a modified unbiased variant. Amortized methods exploit shared structure across many input vectors [Schwab and Karlen, 2019, Jethani et al., 2021], an incomparable setting to ours.

Leverage scores are widely used in randomized matrix algorithms [Cohen et al., 2015, Alaoui and Mahoney, 2015, Drineas et al., 2008, Cohen et al., 2017, Musco and Musco, 2017, Rudi et al., 2018, Spielman and Srivastava, 2008, Agarwal et al., 2020] and active learning [Cohen and Migliorati, 2017, Avron et al., 2019, Chen and Price, 2019, Erdélyi et al., 2020, Chen and Derezinski, 2021, Gajjar et al., 2023, Musco et al., 2022, Cardenas et al., 2023, Shimizu et al., 2023]. The Shapley estimation problem of [Musco and Witter, 2025] is an active learning problem; the $O(n \log n + n/\varepsilon)$ leverage rate is near-optimal for ℓ_2 active linear regression [Batson et al., 2014, Chen and Price, 2019]. Our work shows that with sparsity, the dependence on n is replaced by dependence on s^2.

Sparse linear regression. Classical sparse estimators use ℓ_1 regularization [Tibshirani, 1996, Candès et al., 2006] or hard thresholding [Blumensath and Davies, 2009, Jain et al., 2014], with guarantees typically requiring structural assumptions on the design matrix, such as restricted isometry, restricted eigenvalue, compatibility, or related incoherence conditions [Bickel et al., 2009, van de Geer and Bühlmann, 2009, Wainwright, 2009]. Our setting differs: the design is fixed, the bottleneck is value-function queries, and sparsity is exploited primarily through the row-sampling distribution.

[1] The baseline $\mathbf{y}$ may be fixed or drawn from a data distribution [Lundberg et al., 2018, Janzing et al., 2020]; our methods are agnostic to this choice.

Moreover, in Section 4, we also couple our row-sampling scheme with iterative hard thresholding [Blumensath and Davies, 2009], replacing the exact sparse regression step with a computationally efficient polynomial-time procedure.

2 Preliminaries

The regression view of Shapley values. Computing all n Shapley values via Equation (1) requires $O(2^n)$ evaluations of v. While this can be reduced for structured v (e.g. linear models or trees [Lundberg et al., 2018, Campbell et al., 2022, Amoukou et al., 2022, Chen et al., 2018]), for general models one resorts to approximation [Strumbelj and Kononenko, 2010, Lundberg and Lee, 2017, Mitchell et al., 2022]. Kernel SHAP [Lundberg and Lee, 2017] is especially popular and is based on a connection between Shapley values and least-squares regression [Charnes et al., 1988]: letting $\mathbf{1}$ denote the all-ones vector in $\mathbb{R}^n$,

$$\boldsymbol{\phi} = \operatorname*{arg\,min}_{\mathbf{x}:\langle \mathbf{x},\mathbf{1}\rangle = v([n]) - v(\emptyset)} \|\mathbf{Z}\mathbf{x} - \mathbf{y}\|_2^2, \tag{2}$$

where the Kernel SHAP design $\mathbf{Z} \in \mathbb{R}^{(2^n-2)\times n}$ has rows indexed by sets $S \subseteq [n]$ with $0 < |S| < n$, weighted by the symmetric Kernel SHAP weight $w(s) = \big(\binom{n}{s} s(n-s)\big)^{-1}$:

$$[\mathbf{Z}]_S = \sqrt{w(|S|)}\, z_S^\top, \qquad [\mathbf{y}]_S = \sqrt{w(|S|)}\big(v(S) - v(\varnothing)\big),$$

and $z_S \in \{0,1\}^n$ is the indicator vector of S.

Solving (2) exactly requires evaluating v on all $2^n - 2$ subsets, so Kernel SHAP and its variants approximate it via subsampling: for a probability distribution $\mathbf{p}$ over rows and a sampling matrix $\mathbf{S}$ that selects m rows with rescaling $1/\sqrt{p_j}$, one solves

$$\widehat{\boldsymbol{\phi}} = \operatorname*{arg\,min}_{\langle \mathbf{x},\mathbf{1}\rangle = v([n]) - v(\emptyset)} \|\mathbf{S}\mathbf{Z}\mathbf{x} - \mathbf{S}\mathbf{y}\|_2^2, \tag{3}$$

using only m value-function evaluations. Kernel SHAP samples row S with probability proportional to $w(|S|)$; in practice, paired sampling and without-replacement variants substantially improve performance [Lundberg and Lee, 2017, Covert and Lee, 2021, Jethani et al., 2021].

The affine constraint can be removed by projecting onto $\mathbf{1}^\perp$: with $\Delta = v([n]) - v(\varnothing)$ and $P = I - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$, define $\mathbf{A} = \mathbf{Z}P$ and $\mathbf{b} = \mathbf{y} - \mathbf{Z}\mathbf{1} \cdot \frac{v([n]) - v(\varnothing)}{n}$. Equation (2) is then equivalent to the unconstrained least-squares problem [Musco and Witter, 2025]:

$$x_\star = \operatorname*{arg\,min}_{x \in \mathbb{R}^n} \|\mathbf{A}x - \mathbf{b}\|_2^2, \qquad \boldsymbol{\phi} = Px_\star + \frac{\Delta}{n}\mathbf{1}. \tag{4}$$

Sparse influence model. We assume that the value function depends only on an unknown active set $T_\star \subseteq [n]$ with $|T_\star| = s \ll n$. Equivalently, every player outside $T_\star$ is a dummy: for each $i \notin T_\star$,

$$v(S \cup \{i\}) - v(S) = 0 \qquad \text{for all } S \subseteq [n] \setminus \{i\}.$$

Thus adding or removing inactive players does not change the value of the game, and $v(S) = v(S \cap T_\star)$ for every $S \subseteq [n]$. By the dummy-player axiom of the Shapley value, $\phi_i = 0$ for all $i \notin T_\star$. Hence the Shapley vector is supported on at most s coordinates. In the regression viewpoint, this means that the relevant signal lies in the columns of the design indexed by $T_\star$. Our goal is to exploit this structure so that the number of value-function queries scales with the intrinsic sparsity s, rather than with the ambient dimension n.

This sparsity is preserved in the projected regression formulation. In particular $\mathbf{A}x - \mathbf{b} = \mathbf{Z}\left(Px + \frac{v([n]) - v(\varnothing)}{n}\mathbf{1}\right) - \mathbf{y}$. Since the efficient Shapley vector satisfies $\mathbf{1}^\top\phi = \Delta$, we have $P\phi + \Delta\mathbf{1}/n = \phi$. Hence, if $\boldsymbol{\phi}$ is s-sparse, the unconstrained problem $\min_x \|\mathbf{A}x - \mathbf{b}\|_2^2$ admits the s-sparse minimizer $x_\star = \boldsymbol{\phi}$.

Leverage scores. For a matrix $\mathbf{X} \in \mathbb{R}^{m\times n}$, $\mathbf{X}^+$ denotes the Moore–Penrose pseudoinverse. Let $\mathbf{A} = \mathbf{Z}P$ denote the projected Kernel SHAP design, where $P = I - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$. By the symmetry

calculation of Musco and Witter [2025, Lemma 3.3], $\mathbf{A}^\top \mathbf{A} = \frac{1}{n} P$. Moreover, Musco and Witter [2025, Lemma 3.2] showed that the unrestricted leverage score of a row $z \in \{0,1\}^n$, with $k = \|z\|_1$, has the closed form

$$\ell_z \triangleq [\mathbf{A}]_z (\mathbf{A}^\top \mathbf{A})^+ [\mathbf{A}]_z^\top = \binom{n}{k}^{-1}, \tag{5}$$

These are the leverage scores used by dense Leverage SHAP when no sparsity is available. When z is the indicator vector of a subset S, we use ℓ_z and ℓ_S interchangeably throughout the paper.

Under the sparse influence model, however, the relevant directions are not all of $\mathbb{R}^n$, but only sparse coordinate subspaces. We therefore use a restricted version of leverage. For a candidate support $T \subseteq [n]$ with $|T| = t$, let $\mathbf{A}_T \in \mathbb{R}^{(2^n-2)\times t}$ denote the restriction of $\mathbf{A}$ to the columns indexed by T. The leverage score of row z restricted to support T is

$$\ell_z(T) := [\mathbf{A}_T]_z (\mathbf{A}_T^\top \mathbf{A}_T)^+ [\mathbf{A}_T]_z^\top. \tag{6}$$

Since the true active support is unknown, we take the worst case over all supports of size at most s:

$$\ell_z^{(s)} := \max_{T \subseteq [n],\, |T| \le s} \ell_z(T), \qquad L_s := \sum_z \ell_z^{(s)}. \tag{7}$$

These quantities have distinct roles. The unrestricted score ℓ_z measures the importance of row z for approximating the full n-dimensional design. The restricted score $\ell_z(T)$ is the corresponding quantity when the active support T is known; in this case, $\sum_z \ell_z(T) = \operatorname{rank}(\mathbf{A}_T) \le |T|$, so the relevant leverage mass is at most s, independent of n. Since $T_\star$ is unknown, our sampling distribution uses the worst-case restricted scores $\ell_z^{(s)}$. For the Kernel SHAP design, Proposition 1 gives a closed form for $\ell_z^{(s)}$, and Proposition 3 shows that $L_s = O\big(s \log \frac{en}{s}\big)$. Thus restricted leverage sampling replaces the ambient-dimensional leverage budget by one controlled by the sparsity level.

3 Sample-Efficient Sparse-Leverage SHAP

For a general set function $v : 2^{[n]} \to \mathbb{R}$, exact computation of the Shapley values requires 2^n evaluations of v. The leverage-score refinement of [Musco and Witter, 2025] achieves a provably accurate approximation with $O\big(n \log \frac{n}{\delta} + \frac{n}{\varepsilon\delta}\big)$ value-function evaluations. While this is far better than exhaustive enumeration, it still scales linearly with n. In many settings, however, the underlying interaction structure is much simpler: if there is exactly one non-dummy player, the Shapley vector can be identified with $O(\log n)$ queries via binary search. These extremes suggest that the true difficulty may be governed by the sparsity level s rather than by n itself. Motivated by this, we combine restricted leverage-score sampling with sparse matrix concentration to derive guarantees that depend on the number of active players s rather than on n.

3.1 Sparse-Leverage SHAP Algorithm

In general, computing $\ell_z^{(s)}$ would require forming, for every candidate support $T \subseteq [n]$ with $|T| \le s$, the restricted design matrix $\mathbf{A}_T$ and its Gram matrix $\mathbf{A}_T^\top \mathbf{A}_T$, then evaluating $\ell_z(T) = [\mathbf{A}_T]_z (\mathbf{A}_T^\top \mathbf{A}_T)^+ [\mathbf{A}_T]_z^\top$. Since there are $\sum_{t=1}^s \binom{n}{t}$ possible supports, this brute-force computation is generally intractable. Fortunately, in the case of Shapley estimation, the situation is much simpler. As in the unrestricted case [Musco and Witter, 2025], we show that the *worst-case restricted* leverage scores $\ell_z^{(s)}$ defined in (7) admit a closed form. This closed form avoids the brute-force maximization over supports and makes the restricted-leverage sampling distribution computationally tractable.

Proposition 1 (Closed form of the restricted leverage score)**.** *For $1 \le s < n$ and $z \in \{0,1\}^n$ with $0 < \|z\|_1 = k < n$, the s-restricted leverage score is*

$$\ell_z^{(s)} \;=\; \tfrac{1}{\binom{n}{k}}\, \rho_s(k), \qquad \rho_s(k) := \min\Big\{1,\; \tfrac{s}{n-s} \max\big(\tfrac{k}{n-k}, \tfrac{n-k}{k}\big)\Big\}. \tag{8}$$

The shrinkage factor $\rho_s(k) \in (0,1]$ is determined by subset size and sparsity level: $\rho_s(k) = 1$ when $k < s$ or $k > n - s$, and is of order s/n when $k \approx n/2$.

The proof is in Appendix B. This closed form also clarifies how the restricted-leverage sampling distribution differs from unrestricted leverage sampling. Since the algorithm samples rows with

Algorithm 1 Sparse-Leverage SHAP

Require: number of players n, value oracle v, sparsity level s, sample budget m

Ensure: estimate $\widehat{\phi}$ of the Shapley value vector

1: Set $\Delta = v([n]) - v(\varnothing)$ and compute $p_z = \ell_z^{(2s)}/L_{2s}$ via Proposition 1.

2: Sample m subset pairs $(z_1, \bar{z}_1), \ldots, (z_m, \bar{z}_m)$ i.i.d. with $\mathbb{P}(z = z_j) = p_{z_j}$ (paired with their complements $\bar{z}_j$ as in Leverage SHAP) and evaluate v on each.

3: Form the diagonal weight matrix $W \in \mathbb{R}^{m\times m}$ with $W_{jj} = \frac{1}{m\, p_{z_j}}$, the sampled design $\mathbf{Z}' \in \mathbb{R}^{m\times n}$ with rows $\sqrt{w(\|z_j\|_1)}\, z_j^\top$, and the sampled response vector $\mathbf{b}' \in \mathbb{R}^m$.

4: Solve the weighted projected s-sparse regression:

$$\widehat{\phi}_\perp \;=\; \operatorname*{arg\,min}_{x\in\mathbb{R}^n,\ \|x\|_0\le s} \left\|W^{1/2}\left(\mathbf{Z}' P x - \mathbf{b}'\right)\right\|_2^2.$$

5: Return $\widehat{\phi} = P\widehat{\phi}_\perp + \frac{\Delta}{n}\mathbf{1}$.

probability proportional to $\ell_z^{(2s)}$, the induced distribution over subset sizes is reweighted by the factor $\rho_{2s}(k)$. Thus, compared with unrestricted leverage sampling, the restricted distribution places relatively more mass on extreme subset sizes, where $\rho_{2s}(k) = 1$, and relatively less mass on intermediate subset sizes. In particular, for balanced subsets $k \approx n/2$, we have $\rho_{2s}(k) \asymp s/n$ when $s \ll n$, so these subset sizes are substantially downweighted.

We now turn the restricted-leverage scores into a concrete sampling algorithm for sparse Shapley value estimation. Algorithm 1 is a sparse analogue of Leverage SHAP. It keeps the Kernel SHAP regression objective, but replaces dense leverage sampling with a restricted-leverage distribution designed for sparse Shapley vectors. Specifically, each subset row z is sampled with probability proportional to its worst-case $2s$-restricted leverage score, $p_z \propto \ell_z^{(2s)}$. This distribution gives higher sampling probability to rows that are informative for at least one s-dimensional coordinate subspace. After sampling subset pairs $(z, \bar{z})$, the algorithm evaluates the value function on the sampled subsets and forms the corresponding importance-weighted regression problem. It then solves this regression under an s-sparsity constraint, reflecting the assumption that only s players have nonzero Shapley values. Finally, the efficiency correction is added back so that the returned estimate satisfies the Shapley efficiency constraint.

The projection $P = I - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$ need not be formed explicitly: matrix-vector products $(z_S^\top P)x = \sum_{i\in S} x_i - \frac{|S|}{n}\sum_{i=1}^n x_i$ reduce to subset sums on the $2m$ sampled rows.

3.2 Sparse Spectral Approximation and Single-Run Accuracy

To analyze the Algorithm 1, we use a matrix-concentration bound that holds uniformly over all sparse supports. We state it directly for the independently sampled subset pairs used by the algorithm; the corresponding Bernoulli sampling bound is given in Appendix C.2.

Lemma 2 (Sparse spectral approximation)**.** *Let* $\mathbf{S}$ *sample* m *independent subset pairs with probabilities* $p_z = \ell_z^{(s)}/L_s$ *and rescaling as in Algorithm 1. For any* $\epsilon, \delta \in (0,1)$*, if*

$$m \ge C\frac{L_s}{\epsilon^2}\left(s\log\frac{en}{s} + \log\frac{2}{\delta}\right), \tag{9}$$

then, with probability at least $1-\delta$*, every* $T \subseteq [n]$ *with* $|T| \le s$ *satisfies*

$$(1-\epsilon)\|\mathbf{A}_T x\|_2^2 \le \|\mathbf{S}\mathbf{A}_T x\|_2^2 \le (1+\epsilon)\|\mathbf{A}_T x\|_2^2, \qquad \forall x \in \mathbb{R}^{|T|},$$

where $C > 0$ *is a universal constant.*

The proof is in Appendix C.4 (matrix Bernstein with a union bound over the $\binom{n}{\le s} \le (en/s)^s$ supports). In the dense Leverage SHAP analysis of [Musco and Witter, 2025], the relevant total leverage mass scales with n, giving $\widetilde{O}(n)$ samples. Here, the relevant complexity parameter L_s instead measures the largest energy each row contributes on any support of size at most s. The next proposition shows that $L_s = O(s\log n)$ for the sparse Shapley design, replacing the ambient-dimension dependence with the sparsity level.

Proposition 3 (Restricted leverage mass). *For any $s \in [n-1]$, $L_s = \sum_z \ell_z^{(s)} = O(s \log(en/s))$.*

The proof, given in Appendix B, evaluates the closed form of $\ell_z^{(s)}$ from Proposition 1 and bounds the resulting sum over subset sizes $k \in \{1, \ldots, n-1\}$. Combining these bounds gives the following guarantee for one run of Algorithm 1.

Theorem 4 (Single-run recovery guarantee). *Suppose $\boldsymbol{\phi}$ is supported on an unknown $T \subseteq [n]$ with $1 \le s < n$ and $|T| \le s$. For any $\epsilon \in (0,1)$, Algorithm 1, run with*

$$m = O\left(\frac{s^2}{\epsilon} \log \frac{en}{s} \log(2n)\right)$$

samples, returns an estimate satisfying the efficiency constraint and, with probability at least $3/4$,

$$\|\mathbf{Z}\widehat{\boldsymbol{\phi}} - \mathbf{y}\|_2^2 \le (1+\epsilon)\|\mathbf{Z}\boldsymbol{\phi} - \mathbf{y}\|_2^2.$$

The proof is in Appendix D.2. The $2s$-restricted scores enter because the difference between the internal s-sparse regression vector and $\boldsymbol{\phi}$ is $2s$-sparse. The returned efficient estimate need not itself be sparse. The fixed-residual bound behind Theorem 4 comes from a second-moment estimate and Markov's inequality, so driving the failure probability of a single run down to δ directly would multiply the sample size by $1/\delta$. We therefore fix the single-run success probability at $3/4$ and obtain confidence $1-\delta$ at only a $\log(1/\delta)$ factor through the amplification wrapper of Section 3.3.

3.3 Confidence Amplification and Main Guarantee

Theorem 4 gives an accurate estimate with constant probability $3/4$. To obtain confidence $1-\delta$, we repeat the procedure independently and select one of the returned estimates.

Indeed, for every efficient estimate $\widehat{\boldsymbol{\phi}}$, least-squares orthogonality and $\mathbf{A}^\top \mathbf{A} = P/n$ give

$$\|\mathbf{Z}\widehat{\boldsymbol{\phi}} - \mathbf{y}\|_2^2 = \|\mathbf{Z}\boldsymbol{\phi} - \mathbf{y}\|_2^2 + \frac{1}{n}\|\widehat{\boldsymbol{\phi}} - \boldsymbol{\phi}\|_2^2.$$

Thus accurate candidates are close to $\boldsymbol{\phi}$ in Euclidean norm. Algorithm 2 selects a candidate with the smallest radius containing a majority of the outputs. It uses the outputs of the single-run procedure (e.g., Algorithm 1)

Algorithm 2 Confidence Amplification for Sparse-Leverage SHAP

Require: number of players n, a single-run procedure $\mathcal{A}$ with its parameters fixed, failure probability $\delta \in (0,1)$

Ensure: selected estimate $\widehat{\boldsymbol{\phi}}$

1: Let R be the smallest odd integer with $R \ge 8\log(1/\delta)$ and set $q = (R+1)/2$.
2: **for** $j = 1, \ldots, R$ **do**
3: Run $\mathcal{A}$ with independent randomness to obtain $\widehat{\boldsymbol{\phi}}^{(j)}$.
4: **end for**
5: Compute $d(j,\ell) = \|\widehat{\boldsymbol{\phi}}^{(j)} - \widehat{\boldsymbol{\phi}}^{(\ell)}\|_2/\sqrt{n}$ for all $j, \ell \in [R]$.
6: For each j, let $\widehat{\rho}_j$ be the q-th smallest element of $\{d(j,\ell) : \ell \in [R]\}$, including $d(j,j) = 0$.
7: Choose $\widehat{j} \in \arg\min_{j \in [R]} \widehat{\rho}_j$ and return $\widehat{\boldsymbol{\phi}}^{(\widehat{j})}$.

Here $\mathcal{A}$ denotes Algorithm 1 with n, v, s, m fixed. The same wrapper (a.k.a. Algorithm 2) also applies to the other single-run procedure introduced in Section 4. The next lemma states the guarantee in a form that covers both estimators.

Lemma 5 (Outer-loop confidence amplification). *Suppose each independent run of $\mathcal{A}$ returns an efficient estimate and, for some fixed $E_0 \ge 0$,*

$$\mathbb{P}\Big(\|\mathbf{Z}\widehat{\boldsymbol{\phi}}^{(j)} - \mathbf{y}\|_2^2 \le \|\mathbf{Z}\boldsymbol{\phi} - \mathbf{y}\|_2^2 + E_0\Big) \ge \frac{3}{4}.$$

Then, with probability at least $1-\delta$, Algorithm 2 returns an estimate satisfying

$$\|\mathbf{Z}\widehat{\boldsymbol{\phi}} - \mathbf{y}\|_2^2 \le \|\mathbf{Z}\boldsymbol{\phi} - \mathbf{y}\|_2^2 + 9E_0.$$

The proof is in Appendix D.3. To see the selection mechanism, call a run good when its excess loss is at most E_0. With probability at least $1-\delta$, good runs form a strict majority. In the prediction norm, every good candidate is within $\sqrt{E_0}$ of $\boldsymbol{\phi}$, so any two good candidates are within $2\sqrt{E_0}$. The selected candidate therefore has a majority neighborhood of radius at most $2\sqrt{E_0}$. This neighborhood contains a good candidate, placing the selected estimate within $3\sqrt{E_0}$ of $\boldsymbol{\phi}$ and giving the factor nine in excess loss.

Theorem 6. *Run Algorithm 2 with Algorithm 1 as the single-run procedure, using the per-run budget in Theorem 4 with a sufficiently large universal constant. For any $\epsilon, \delta \in (0,1)$, with probability at least $1-\delta$, the selected estimate satisfies*

$$\|\mathbf{Z}\widehat{\boldsymbol{\phi}} - \mathbf{y}\|_2^2 \le (1+\epsilon)\|\mathbf{Z}\boldsymbol{\phi} - \mathbf{y}\|_2^2.$$

The total number of samples required is

$$O\left(\frac{s^2}{\epsilon}\log\frac{en}{s}\log(2n)\log\frac{2}{\delta}\right).$$

The proof of Theorem 6 is in Appendix D.4. If one run uses m subset pairs, the wrapper uses Rm pairs and at most $R(2m+2)$ value-function evaluations; candidate selection costs $O(R^2 n)$ arithmetic operations. The dense $\widetilde{O}(n)$ query complexity of [Musco and Witter, 2025] is thus replaced by $\widetilde{O}(s^2)$, with cost determined by the number of *active* players, and the bound remains meaningful even in the underdetermined regime $m \ll n$ where dense guarantees are vacuous. To our knowledge, this is the first leverage-score-based Shapley estimation guarantee whose query complexity scales polynomially with the number of non-dummy players s rather than n.

4 A Computationally Efficient Algorithm for Sparse-Leverage SHAP

Algorithm 1 reduces Shapley estimation to a sampled ℓ_0-constrained least-squares problem. Solving this step is NP-hard in general [Natarajan, 1995]. We therefore use Iterative Hard Thresholding (IHT) [Blumensath and Davies, 2009], keeping the sparse-leverage sampling framework and replacing the exact sparse regression solve by projected gradient steps followed by hard thresholding.

Algorithm 3 Sparse-Leverage IHT SHAP

Require: number of players n, value oracle v, sparsity s, samples m, iterations K, step size η
Ensure: estimate $\widehat{\boldsymbol{\phi}}$

1: Set $\Delta = v([n]) - v(\emptyset)$ and $P = I - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$.
2: Compute $p_z \propto \ell_z^{(3s)}$, sample m paired subsets, evaluate v, and form Z', b', and W as in Algorithm 1.
3: Define $B = W^{1/2}Z'P, c = W^{1/2}b', \boldsymbol{\phi}^0 = 0$.
4: **for** $t = 0, \dots, K-1$ **do**

$$\boldsymbol{\phi}^{t+1} = H_s\left(\boldsymbol{\phi}^t + \eta B^\top(c - B\boldsymbol{\phi}^t)\right),$$

where H_s keeps the s largest entries in magnitude.
5: **end for**
6: **return** $\widehat{\boldsymbol{\phi}} = P\boldsymbol{\phi}^K + \frac{\Delta}{n}\mathbf{1}$.

Algorithm 3 is a single-run computational version of Algorithm 1. Its efficient output can be passed to the same wrapper in Algorithm 2. The sampling step uses $3s$-restricted scores instead of $2s$-restricted scores, since the IHT analysis involves the union of three sparse supports. Algorithm 1 solves the ℓ_0-constrained regression exactly, whereas Algorithm 3 replaces this combinatorial step by K iterations of IHT. This makes the sparse regression step polynomial-time, at the cost of an additional optimization error that decreases geometrically with K.

Theorem 7. *Assume $s/n \le 1/12$. For any $\epsilon, \delta \in (0,1)$, run Algorithm 2 with Algorithm 3 as the single-run procedure, using*

$$m = O\left(\frac{s^2}{\epsilon}\log\frac{en}{s}\log(2n)\right), \qquad K = O\left(\log\frac{2}{\epsilon}\right).$$

Then, with probability at least $1-\delta$, the selected estimate satisfies

$$\|\mathbf{Z}\widehat{\boldsymbol{\phi}}-\mathbf{y}\|_2^2 \le (1+\epsilon)\|\mathbf{Z}\boldsymbol{\phi}-\mathbf{y}\|_2^2 + \frac{\epsilon}{n}\|\boldsymbol{\phi}\|_2^2.$$

Here, the same selection rule of Algorithm 2 applies since every single-run IHT output satisfies the efficiency constraint. The proof of Theorem 7 is in Appendix E. The additional term $n^{-1}\epsilon\|\boldsymbol{\phi}\|_2^2$ is the optimization error from terminating IHT after a finite number K of iterations. The exact ℓ_0 estimator in Algorithm 1 has no such term because it solves the sampled sparse regression problem exactly. In contrast, IHT is a polynomial-time iterative surrogate: after K iterations, its optimization error contracts geometrically, and taking $K = O(\log(2/\epsilon))$ makes the residual optimization error negligible, of order at most $O(n^{-1}\epsilon\|\boldsymbol{\phi}\|_2^2)$. Thus, the theorem separates the statistical sampling error, captured by the $(1+\epsilon)\|\mathbf{Z}\boldsymbol{\phi}-\mathbf{y}\|_2^2$ term, from the computational error introduced by the approximate sparse solver.

Remark 8 (Computational complexity)**.** *The restricted leverage scores depend only on the subset size, so the sampling distribution can be precomputed over $k = 1, \dots, n-1$ and sampled without enumerating all 2^n subsets. The query complexity is m sampled subsets (up to a factor two when complements are explicitly evaluated). Each IHT iteration requires one multiplication by B and one by $B^\top$, costing $O(mn)$ time in a dense implementation, plus $O(n)$ time for hard thresholding. Thus the arithmetic cost is $O(Kmn)$ and the storage is $O(mn)$, or less with sparse/bitset subset representations. In contrast, best subset selection, or sparse regression with an ℓ_0 constraint, is known to be NP-hard in general [Natarajan, 1995]. IHT replaces this NP-hard step with a polynomial-time iterative procedure, at the cost of a negligible optimization error after sufficiently many iterations.*

5 Experiments

We evaluate Sparse Leverage SHAP through three complementary experiments. Section 5.1 reproduces the real-data benchmark [Musco and Witter, 2025] on eight datasets. Section 5.2 focuses on datasets with empirically sparse Shapley vectors, testing whether accuracy tracks the effective sparsity of the Shapley vector. Section 5.3 isolates the mechanism in a controlled synthetic setting with a known active set and reports both estimation error and exact support-recovery rate. In all experiments, we use a single run of Sparse Leverage SHAP without the confidence-boosting wrapper. Detailed protocols and hyperparameters are deferred to Appendix F.

Methods compared. We compare four estimators: (i) *Kernel SHAP* [Lundberg and Lee, 2017] with replacement; (ii) *Optimised Kernel SHAP* (the `shap` library [Lundberg et al., 2020]); (iii) *Leverage SHAP* [Musco and Witter, 2025]; and (iv) *Sparse Leverage SHAP* (ours) using the IHT solver in section 4. All four enforce $\mathbf{1}^\top x = v([n]) - v(\emptyset)$ via projection. We report the normalised squared ℓ_2 error $\|\widehat{\boldsymbol{\phi}} - \boldsymbol{\phi}\|_2^2 / \|\boldsymbol{\phi}\|_2^2$, matching Musco and Witter [2025], with ground truth from Tree SHAP [Lundberg et al., 2020] on an XGBoost model [Chen and Guestrin, 2016] (real data) or via an active-player closed-form reduction (synthetic).

5.1 Benchmark accuracy on real data

We reproduce the protocol of Musco and Witter [2025] on the eight datasets in that paper, all obtained via `shap.datasets`: IRIS, California, Diabetes, Adult, Correlated, Independent, NHANES, and Communities, with feature counts ranging from $n = 4$ to $n = 101$. We run all four estimators at sample budgets $m \in \{0.2n, 0.5n, n, 2n, 5n, 10n, 20n, 40n, 80n, 160n\}$ and report medians with Q1–Q3 bands across 100 test points per dataset. A predicted-vs-true scatter plot at $m = 5n$ is shown in Appendix F (Figure 4); error-vs-m curves appear below.

Results. At $m = 10n$, Sparse Leverage SHAP achieves the lowest median error on Adult, Communities, and NHANES, while Leverage SHAP is best on California, Correlated, Diabetes, and Independent; the two are within $1.6\times$ on every dataset with $n \ge 8$. Both clearly dominate Kernel SHAP in the underdetermined regime $m < n$. On NHANES and Communities, dense Leverage SHAP exhibits numerical instability at small m, with median errors reaching 10^8 and 10^{16}, respectively, before the design becomes well-conditioned. Sparse Leverage SHAP is stable throughout, suggesting that the sparsity constraint also acts as a conditioning safeguard independent of its role in exploiting sparsity.

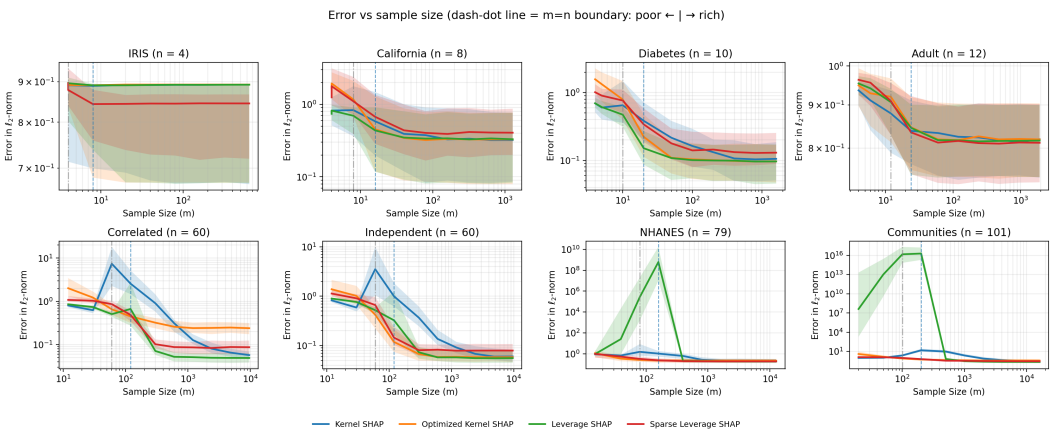


Figure 1: Normalised squared ℓ_2 error as a function of sample budget m on the eight real-data benchmarks. Lines show medians and shaded bands show Q1–Q3 over 100 test points per dataset. Dash-dot vertical lines mark $m = n$.

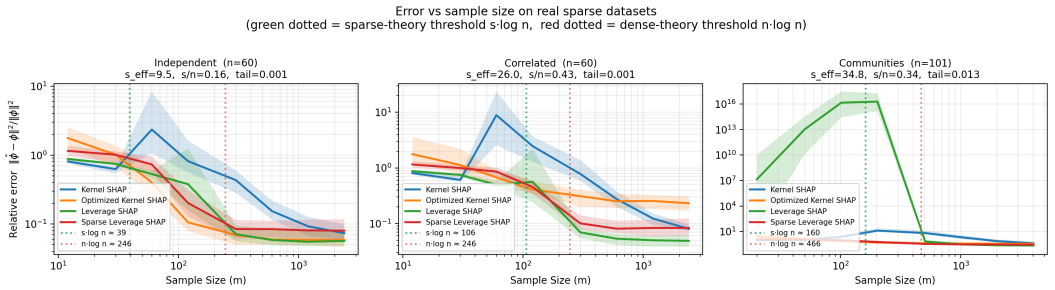


Figure 2: Error versus sample budget on three real datasets with approximately sparse Shapley vectors. Green dotted lines mark $s_{\text{eff}} \log n$ (sparse threshold); red dotted lines mark $n \log n$ (dense threshold). Q1–Q3 over 50 trials.

5.2 Real data with sparse Shapley vectors

Our guarantee replaces the linear dependence on n with a quadratic dependence on s, so any advantage should be largest when the Shapley vector is strongly sparse. To test this, we measure the empirical sparsity of Tree SHAP vectors and rerun the comparison on three of the highest-n benchmarks (Independent, Correlated, and Communities). We define the *effective sparsity* as $s_{\text{eff}}(\boldsymbol{\phi}) = |\{i : |\phi_i| > \tau \|\boldsymbol{\phi}\|_\infty\}|$ with $\tau = 0.05$. Independent is strongly sparse ($s_{\text{eff}}/n = 0.16$, tail mass $\approx 10^{-3}$), while Correlated and Communities are moderately sparse ($s_{\text{eff}}/n \in [0.34, 0.43]$); see Table A in Appendix F.

Findings. On Independent ($s/n = 0.16$), both leverage methods reach near-optimal error at $m \approx s_{\text{eff}} \log n \approx 39$, well below $n \log n \approx 246$, while Kernel SHAP requires roughly $4\times$ more samples. On Correlated, the gap between Sparse and Dense Leverage narrows and inverts at large m (final error 0.08 vs. 0.05), matching the prediction that the sparse advantage shrinks as s/n grows. On Communities, dense Leverage SHAP becomes numerically unstable for $m < 2n$ (median errors $> 10^{16}$ from rank-deficient weighted least squares on the $n = 101$ design), while Sparse Leverage SHAP remains stable and converges to 0.28, a 22% improvement over Optimised Kernel SHAP at $m = 40n$. Taken together, these results show parity with dense Leverage SHAP in accuracy on approximately sparse data and a clear robustness advantage in the small-m regime. They do not isolate the query-complexity gap suggested by our analysis. At these sizes $s_{\text{eff}}^2 \log n$ exceeds $n \log n$ (about 370 versus 246 for Independent), so the current bound does not yet predict an advantage, and the matching sample sizes of the two leverage methods are consistent with this.

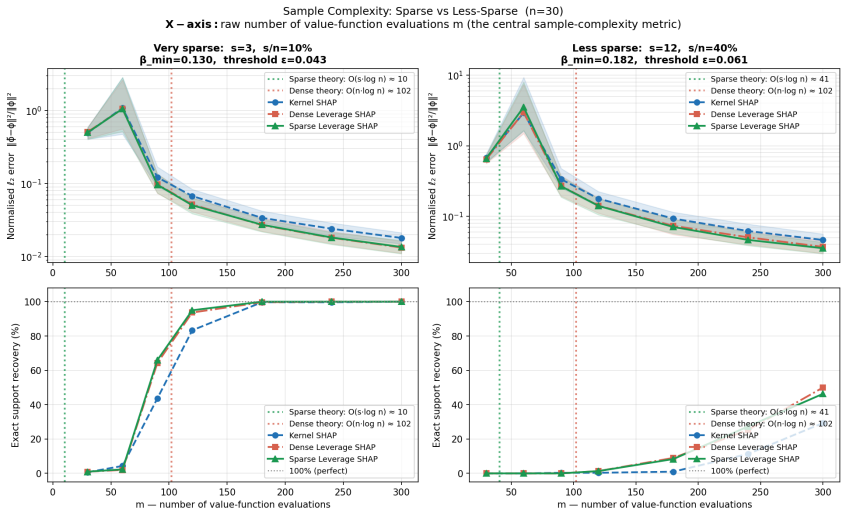


Figure 3: Synthetic sparse sigmoid game, $n = 30$. Top: ℓ_2 error (log scale). Bottom: exact support recovery rate. Left: $s = 3$. Right: $s = 12$. Green/red dotted lines mark $s \log n / n \log n$. Medians and Q1–Q3 over 300 trials.

5.3 Synthetic sparse game: sample complexity and support recovery

Real data is only approximately sparse, so we complement the above with a controlled synthetic experiment. We construct a sparse sigmoid game $v(S) = \sigma\big(\sum_{i \in S \cap R} a_i\big) - \sigma(0)$ on $n = 30$ players, where $R \subseteq [n]$ is drawn uniformly with $|R| = s$, and $a_i = \varepsilon_i u_i$ with $\varepsilon_i \in \{\pm 1\}$ uniform and $u_i \sim \mathrm{Unif}(1.5, 3.0)$. We test Setting A ($s = 3$, $s/n = 10\%$) and Setting B ($s = 12$, $s/n = 40\%$), with budgets $m \in \{n, \ldots, 10n\}$ and 300 trials per (m, method). Beyond ℓ_2 error, we report the exact support recovery rate, the fraction of trials with $\{i : |\widehat{\phi}_i| > \varepsilon\} = R$ at $\varepsilon = \beta_{\min}/3$, $\beta_{\min} = \min_{i \in R} |\phi_i|$.

Results. In Setting A, all three estimators behave similarly on ℓ_2 error, but the support-recovery curves separate clearly: both leverage methods cross the 50% recovery threshold at $m \approx 0.9\, n \log n$ and reach 100% by $m = 6n$, while Kernel SHAP lags by ≈ 30 evaluations. In Setting B, support recovery is much harder ($s \log n \approx 41$ is a larger fraction of $n \log n \approx 102$): no method exceeds 50% within budget, with the leverage methods leading Kernel SHAP by a small margin. The widening gap in Setting A supports the prediction that subset-size weighting in the sampling distribution—not just in the regression—drives sample-efficient support identification.

6 Conclusion

We studied Shapley value estimation under a sparse influence model. Building on [Musco and Witter, 2025], we proposed Sparse-Leverage SHAP, achieving relative-error approximation in $\widetilde{O}(s^2)$ value-function evaluations. The analysis combines a sparse matrix concentration result holding uniformly over all s-supports with a bound on the restricted leverage mass of the Shapley design and a confidence-amplification step. Empirically, Sparse-Leverage SHAP matches dense Leverage SHAP in accuracy on real data with approximately sparse Shapley vectors and on synthetic sparse games, and is markedly more numerically stable in the small-sample regime.

Limitations and Future Directions. Our guarantees assume exact sparsity; extending to compressible/weak-ℓ_1 vectors with best-s-sparse-approximation guarantees would better match practice. The estimator requires a known s, so data-driven selection is an important practical question. Interesting future directions include closing the gap between our $\widetilde{O}(s^2)$ upper bound and the known $\widetilde{\Omega}(s)$ lower bounds and extending our approach to hidden-symmetry models via group-structured sampling [Tran et al., 2024].

References

N. Agarwal, S. Kakade, R. Kidambi, Y.-T. Lee, P. Netrapalli, and A. Sidford. Leverage score sampling for faster accelerated regression and erm. In *Algorithmic Learning Theory*, pages 22–47. PMLR, 2020.

A. Alaoui and M. W. Mahoney. Fast randomized kernel ridge regression with statistical guarantees. *Advances in neural information processing systems*, 28, 2015.

S. I. Amoukou, T. Salaün, and N. Brunel. Accurate shapley values for explaining tree-based models. In *International conference on artificial intelligence and statistics*, pages 2448–2465. PMLR, 2022.

H. Avron, M. Kapralov, C. Musco, C. Musco, A. Velingker, and A. Zandieh. A universal sampling method for reconstructing signals with simple fourier transforms. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 1051–1063, 2019.

J. Batson, D. A. Spielman, and N. Srivastava. Twice-ramanujan sparsifiers. *SIAM Review*, 56(2): 315–334, 2014.

P. J. Bickel, Y. Ritov, and A. B. Tsybakov. Simultaneous analysis of lasso and dantzig selector. *The Annals of Statistics*, 2009.

T. Blumensath and M. E. Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 2009.

T. W. Campbell, H. Roder, R. W. Georgantas III, and J. Roder. Exact shapley values for local and model-true explanations of decision tree ensembles. *Machine Learning with Applications*, 9: 100345, 2022.

E. J. Candès, J. K. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 2006.

J. M. Cardenas, B. Adcock, and N. Dexter. Cs4ml: A general framework for active learning with arbitrary data based on christoffel functions. *Advances in Neural Information Processing Systems*, 36:19990–20037, 2023.

J. Castro, D. Gómez, and J. Tejada. Polynomial calculation of the shapley value based on sampling. *Computers & operations research*, 36(5):1726–1730, 2009.

A. Charnes, B. Golany, M. Keane, and J. Rousseau. Extremal principle solutions of games in characteristic function form: core, chebychev and shapley value generalizations. In *Econometrics of planning and efficiency*, pages 123–133. Springer, 1988.

J. Chen, L. Song, M. J. Wainwright, and M. I. Jordan. L-shapley and c-shapley: Efficient model interpretation for structured data. *arXiv preprint arXiv:1808.02610*, 2018.

T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

X. Chen and M. Derezinski. Query complexity of least absolute deviation regression via robust uniform convergence. In *Conference on Learning Theory*, pages 1144–1179. PMLR, 2021.

X. Chen and E. Price. Active regression via linear-sample sparsification. In *Conference on Learning Theory*, pages 663–695. PMLR, 2019.

A. Cohen and G. Migliorati. Optimal weighted least-squares methods. *The SMAI journal of computational mathematics*, 3:181–203, 2017.

M. B. Cohen, Y. T. Lee, C. Musco, C. Musco, R. Peng, and A. Sidford. Uniform sampling for matrix approximation. In *Proceedings of the 2015 conference on innovations in theoretical computer science*, pages 181–190, 2015.

M. B. Cohen, C. Musco, and C. Musco. Input sparsity time low-rank approximation via ridge leverage score sampling. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1758–1777. SIAM, 2017.

I. Covert and S.-I. Lee. Improving kernelshap: Practical shapley value estimation using linear regression. In *International conference on artificial intelligence and statistics*, pages 3457–3465, 2021.

A. Datta, S. Sen, and Y. Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *2016 IEEE symposium on security and privacy (SP)*, pages 598–617. IEEE, 2016.

P. Drineas, M. W. Mahoney, and S. Muthukrishnan. Relative-error cur matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, 30(2):844–881, 2008.

T. Erdélyi, C. Musco, and C. Musco. Fourier sparse leverage scores and approximate kernel learning. *Advances in Neural Information Processing Systems*, 33:109–122, 2020.

D. Fryer, I. Strümke, and H. Nguyen. Shapley values for feature selection: The good, the bad, and the axioms. *IEEE Access*, 9:144352–144360, 2021.

A. Gajjar, C. Musco, and C. Hegde. Active learning for single neuron models with lipschitz nonlinearities. In *International Conference on Artificial Intelligence and Statistics*, pages 4101–4113. PMLR, 2023.

A. Ghorbani and J. Zou. Data shapley: Equitable valuation of data for machine learning. In *International conference on machine learning*, pages 2242–2251. PMLR, 2019.

F. Illés and P. Kerényi. Estimation of the shapley value by ergodic sampling. *arXiv preprint arXiv:1906.05224*, 2019.

P. Jain, A. Tewari, and P. Kar. On iterative hard thresholding methods for high-dimensional m-estimation. In *Advances in Neural Information Processing Systems*, 2014.

D. Janzing, L. Minorics, and P. Blöbaum. Feature relevance quantification in explainable ai: A causal problem. In *International Conference on artificial intelligence and statistics*, pages 2907–2916. PMLR, 2020.

N. Jethani, M. Sudarshan, I. C. Covert, S.-I. Lee, and R. Ranganath. Fastshap: Real-time shapley value estimation. In *International conference on learning representations*, 2021.

R. Jia, D. Dao, B. Wang, F. A. Hubis, N. Hynes, N. M. Gürel, B. Li, C. Zhang, D. Song, and C. J. Spanos. Towards efficient data valuation based on the shapley value. In *The 22nd international conference on artificial intelligence and statistics*, pages 1167–1176. PMLR, 2019.

S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

S. M. Lundberg, G. G. Erion, and S.-I. Lee. Consistent individualized feature attribution for tree ensembles. *arXiv preprint arXiv:1802.03888*, 2018.

S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee. From local explanations to global understanding with explainable ai for trees. *Nature machine intelligence*, 2(1):56–67, 2020.

W. E. Marcílio and D. M. Eler. From explanations to feature selection: assessing shap values as feature selection mechanism. In *2020 33rd SIBGRAPI conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 340–347, 2020.

R. Mitchell, J. Cooper, E. Frank, and G. Holmes. Sampling permutations for shapley value estimation. *Journal of Machine Learning Research*, 23(43):1–46, 2022.

C. Molnar. *Interpretable machine learning*. 2020.

C. Musco and C. Musco. Recursive sampling for the nystrom method. *Advances in neural information processing systems*, 30, 2017.

C. Musco and R. T. Witter. Provably accurate shapley value estimation via leverage score sampling. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=wg3rBImn3O.

C. Musco, C. Musco, D. P. Woodruff, and T. Yasuda. Active linear regression for ℓ_p norms and beyond. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 744–753. IEEE, 2022.

B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24 (2):227–234, 1995.

A. Rudi, D. Calandriello, L. Carratino, and L. Rosasco. On fast leverage score sampling and optimal learning. *Advances in Neural Information Processing Systems*, 31, 2018.

P. Schwab and W. Karlen. Cxplain: Causal explanations for model interpretation under uncertainty. *Advances in neural information processing systems*, 32, 2019.

L. S. Shapley. Notes on the n-person game—ii: The value of an n-person game. Technical report, 1951.

A. Shimizu, X. Cheng, C. Musco, and J. Weare. Improved active learning via dependent leverage score sampling. *arXiv preprint arXiv:2310.04966*, 2023.

D. A. Spielman and N. Srivastava. Graph sparsification by effective resistances. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 563–568, 2008.

E. Strumbelj and I. Kononenko. An efficient explanation of individual classifications using game theory. *The Journal of Machine Learning Research*, 11:1–18, 2010.

E. Štrumbelj and I. Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41(3):647–665, 2014.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B*, 1996.

P. N. Tran, D. Mandal, L. Tran-Thanh, et al. Symmetric linear bandits with hidden symmetry. *Advances in Neural Information Processing Systems*, 37:128699–128733, 2024.

J. A. Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12:389–434, 2012.

S. A. van de Geer and P. Bühlmann. On the conditions used to prove oracle results for the lasso. *Electronic Journal of Statistics*, 2009.

M. J. Wainwright. Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1-constrained quadratic programming. *IEEE Transactions on Information Theory*, 2009.

J. T. Wang and R. Jia. Data banzhaf: A robust data valuation framework for machine learning. In *International conference on artificial intelligence and statistics*, pages 6388–6421. PMLR, 2023.

Contents of Appendix

A Restricted leverage scores: total mass for fixed support **ii**

B Proof of Propositions 1 and 3 **ii**

C Sparse spectral approximation and approximate matrix multiplication **iv**

C.1 Geometry and restricted leverage scores iv

C.2 Spectral approximation under Bernoulli sampling iv

C.3 Fixed-residual approximate matrix multiplication vi

C.4 Paired sampling and proof of Lemma 2 ix

D Proofs of Theorems 4 and 6 **x**

D.1 Sparse subspace embedding xi

D.2 Single-run recovery guarantee xi

D.3 Outer-loop confidence amplification xii

D.4 High-confidence recovery guarantee xiii

E Proof of Theorem 7 **xiii**

F Additional experimental details and figures **xv**

F.1 Datasets, models, and estimators xv

F.2 Predicted-vs-true scatter (Section 5.1) xvi

F.3 Empirical sparsity of Tree SHAP vectors (Section 5.2) xvi

F.4 Synthetic sparse game (Section 5.3) xvi

A Restricted leverage scores: total mass for fixed support

For completeness we record the basic identity that motivates restricted-leverage sampling. Let $N_{\mathrm{row}} := 2^n - 2$.

Let $\mathbf{A} \in \mathbb{R}^{N_{\mathrm{row}} \times n}$ be the centered design matrix, and for a support $T \subset [n]$ write $\mathbf{A}_T \in \mathbb{R}^{N_{\mathrm{row}} \times |T|}$ for the column-restricted matrix. Recall that the *restricted leverage score* of row z relative to T is

$$\ell_z(T) \;:=\; [\mathbf{A}_T]_{z,:}\,(\mathbf{A}_T^\top \mathbf{A}_T)^+\,[\mathbf{A}_T]_{z,:}^\top. \tag{10}$$

These are the natural leverage scores for the sparse regression problem on support T.

Proposition 9 (Restricted leverage scores sum to at most the support size)**.** *For every fixed support $T \subset [n]$,*

$$\sum_z \ell_z(T) \;=\; \mathrm{rank}(\mathbf{A}_T) \;\le\; |T|. \tag{11}$$

In particular, if $\mathbf{A}_T$ has full column rank and $|T| = s$, then $\sum_z \ell_z(T) = s$.

Proof. Using the definition of $\ell_z(T)$ and summing over rows,

$$\begin{aligned}\sum_z \ell_z(T) &= \sum_z [\mathbf{A}_T]_{z,:}(\mathbf{A}_T^\top \mathbf{A}_T)^+ [\mathbf{A}_T]_{z,:}^\top = \mathrm{tr}\big(\mathbf{A}_T(\mathbf{A}_T^\top \mathbf{A}_T)^+ \mathbf{A}_T^\top\big) \\ &= \mathrm{tr}\big((\mathbf{A}_T^\top \mathbf{A}_T)^+ \mathbf{A}_T^\top \mathbf{A}_T\big)\,.\end{aligned}$$

The matrix $(\mathbf{A}_T^\top \mathbf{A}_T)^+ \mathbf{A}_T^\top \mathbf{A}_T$ is the orthogonal projector onto the row space of A_T, so its trace is exactly $\mathrm{rank}(A_T)$. □

Proposition 9 is the precise sense in which the leverage mass is linear in the support size rather than linear in n. Thus, if one samples proportionally to the *restricted* leverage scores $\ell_z(T)$ for a fixed support T, the expected sample complexity is of order s up to logarithmic factors, yielding an $O(s \log n)$-type guarantee.

By contrast, in the support-unknown setting considered in Theorem 6, the sample complexity is driven by

$$L_s = \sum_z \ell_z^{(s)}, \qquad \ell_z^{(s)} := \max_{|T| \le s} \ell_z(T),$$

so one must additionally control L_s for the particular Shapley design. Proposition 3 proves $L_s = O(s \log n)$; combining this with the uniform sparse bounds gives the $\widetilde{O}(s^2)$ query complexity.

B Proof of Propositions 1 and 3

In this section, we first calculate the closed form of $\ell_z^{(s)}$, proving Proposition 1. Based on this expression, we then derive an upper bound for L_s, proving Proposition 3. Note that, when z is the indicator vector of a subset S, we use ℓ_z and ℓ_S interchangeably.

Proof. The argument has four steps: (i) restrict to T and invert by Sherman–Morrison; (ii) express $\ell_S(T)$ as a convex quadratic in $r := |S \cap T|$; (iii) maximize at the endpoints; (iv) sum over k using a harmonic bound.

Step 1: restricted Gram matrix. Fix $T \subset [n]$ with $|T| = t \le s$. The matrix $\mathbf{A}_T^\top \mathbf{A}_T$ is the $T \times T$ principal submatrix of $\frac{1}{n}P$. In particular,

$$\mathbf{A}_T^\top \mathbf{A}_T \;=\; \tfrac{1}{n}\big(I_t - \tfrac{1}{n}\mathbf{1}_t\mathbf{1}_t^\top\big). \tag{12}$$

The $\frac{1}{n}$ inside the parentheses is inherited from P and is *not* the centering projection on $\mathbb{R}^t$. Since $t < n$, (12) is strictly positive definite (so the pseudoinverse equals the inverse), and Sherman–Morrison gives

$$\big(\mathbf{A}_T^\top \mathbf{A}_T\big)^{-1} \;=\; n\big(I_t + \tfrac{1}{n-t}\mathbf{1}_t\mathbf{1}_t^\top\big). \tag{13}$$

Step 2: closed-form leverage score. With $r := |S \cap T|$, restricting to coordinates in T gives

$$a_{S,T} \;=\; \sqrt{w(k)}\,\big(\mathbf{1}_{S\cap T} - \tfrac{k}{n}\mathbf{1}_T\big) \;\in\; \mathbb{R}^t. \tag{14}$$

Plugging (13) and (14) into the definition of $\ell_S(T)$, and using $\mathbf{1}_T^\top(\mathbf{1}_{S\cap T} - \frac{k}{n}\mathbf{1}_T) = r - \frac{tk}{n}$ together with the expansion $\|\mathbf{1}_{S\cap T} - \frac{k}{n}\mathbf{1}_T\|_2^2 = r - \frac{2kr}{n} + \frac{tk^2}{n^2}$, we obtain the convex-in-r form

$$\ell_S(T) \;=\; nw(k)\Big[r\big(1-\tfrac{2k}{n}\big) + \tfrac{tk^2}{n^2} + \tfrac{(r-tk/n)^2}{n-t}\Big]. \tag{15}$$

Step 3: maximize over r and t. The right-hand side of (15) is convex in r, so its maximum over feasible $r = |S\cap T| \in \{\max(0,k+t-n),\dots,\min(k,t)\}$ is attained at an endpoint. The unrestricted leverage score Musco and Witter [2025] is

$$\ell_S = a_S^\top\big(\mathbf{A}^\top\mathbf{A}\big)^+ a_S = w(k)\,k(n-k) = \tfrac{1}{\binom{n}{k}}. \tag{16}$$

Endpoint 1: T inside S. If $t \le k$, set $r = t$ in (15):

$$\ell_S(T) = w(k)\tfrac{t(n-k)^2}{n-t}, \qquad \frac{\ell_S(T)}{\ell_S} = \frac{t(n-k)}{k(n-t)}.$$

Maximizing over $t \le s$, $\ell_S(T) \le \frac{1}{\binom{n}{k}}\min\{1, \frac{s(n-k)}{k(n-s)}\}$.

Endpoint 2: T inside S^c. If $t \le n-k$, set $r = 0$ in (15):

$$\ell_S(T) = w(k)\tfrac{tk^2}{n-t}, \qquad \frac{\ell_S(T)}{\ell_S} = \frac{tk}{(n-k)(n-t)}.$$

Maximizing over $t \le s$, $\ell_S(T) \le \frac{1}{\binom{n}{k}}\min\{1, \frac{sk}{(n-k)(n-s)}\}$.

If $s \ge \min(k, n-k)$, choosing $T = S$ or $T = S^c$ attains the unrestricted score. Otherwise $s < \min(k,n-k)$, so for every $t \le s$ the feasible endpoints are $r = 0$ and $r = t$, and both displayed ratios increase with t. Thus the endpoint bounds are attained at $t = s$. Combining,

$$\ell_S^{(s)} \;=\; \tfrac{1}{\binom{n}{k}}\,\rho_s(k), \qquad \rho_s(k) := \min\Big\{1,\; \tfrac{s}{n-s}\max\big(\tfrac{k}{n-k}, \tfrac{n-k}{k}\big)\Big\}. \tag{17}$$

The shrinkage factor $\rho_s(k)$ in (17) is the one appearing in Proposition 1. This proves Proposition 1.

Step 4: sum over k. Since $\ell_S^{(s)}$ depends on S only through $k = |S|$, (17) gives

$$L_s \;=\; \sum_{k=1}^{n-1}\binom{n}{k}\cdot\tfrac{1}{\binom{n}{k}}\,\rho_s(k) \;=\; \sum_{k=1}^{n-1}\rho_s(k). \tag{18}$$

The function ρ_s is symmetric under $k \mapsto n-k$, so $L_s \le 2\sum_{k=1}^{\lfloor n/2\rfloor}\rho_s(k)$. For $k \le n/2$, $\max\big(\frac{k}{n-k}, \frac{n-k}{k}\big) = \frac{n-k}{k}$ and hence $\rho_s(k) = \min\{1, \frac{s(n-k)}{(n-s)k}\}$.

Case $s \le n/2$. Then $n - s \ge n/2$, $\rho_s(k) = 1$ for $k \le s$, and for $k > s$, $\rho_s(k) \le \frac{sn}{(n-s)k} \le \frac{2s}{k}$. So

$$L_s \;\le\; 2s + 4s\sum_{k=s+1}^{\lfloor n/2\rfloor}\tfrac{1}{k} \;\le\; Cs\log\tfrac{en}{s},$$

using $\sum_{k=s+1}^{\lfloor n/2\rfloor}\frac{1}{k} \le \log\frac{en}{s}$.

Case $s > n/2$. Trivially $\rho_s(k) \le 1$, so $L_s \le n-1 \le 2s \le 2s\log\frac{en}{s}$.

Combining the two cases yields the bound $L_s = O(s\log(en/s))$. For $s = n$, $L_n = n-1$ satisfies the same bound. □

C Sparse spectral approximation and approximate matrix multiplication

We first recall the restricted-leverage geometry, then establish spectral approximation and fixed-residual approximate matrix multiplication (AMM) bounds for Bernoulli sampling. We then prove the corresponding bounds for independently sampled subset pairs and complete the proof of Lemma 2. These bounds are used in the recovery proofs in Appendices D and E.

C.1 Geometry and restricted leverage scores

Support-specific orthonormal bases. For each $T \subseteq [n]$, let $d = \operatorname{rank}(\mathbf{A}_T)$ and choose an orthonormal basis $\mathbf{U}_T \in \mathbb{R}^{N_{\text{row}} \times d}$ for $\operatorname{col}(\mathbf{A}_T)$, where $N_{\text{row}} = 2^n - 2$. Write $u_{z,T}^\top$ for its z-th row. The Gram and row-norm definitions of leverage coincide for this support:

$$\ell_z(T) = [\mathbf{A}_T]_z (\mathbf{A}_T^\top \mathbf{A}_T)^+ [\mathbf{A}_T]_z^\top = \|u_{z,T}\|_2^2.$$

Indeed, substitution of a thin singular value decomposition of $\mathbf{A}_T$ into the Gram expression gives the squared row norm of its left singular vectors. This row norm is unchanged by an orthogonal change of basis, giving $\|u_{z,T}\|_2^2$. The basis $\mathbf{U}_T$ is chosen separately for each support; it is not obtained by restricting the columns of one global orthonormal basis. In particular, $\mathbf{A}$ has rank $n-1$, and $d \le |T|$ also covers $T = [n]$.

Worst-case s-restricted row energies. For $s \in \{1, \dots, n\}$, recall

$$\ell_z(T) := \|u_{z,T}\|_2^2, \qquad \ell_z^{(s)} := \max_{|T| \le s} \|u_{z,T}\|_2^2. \tag{19}$$

Since $\operatorname{col}(\mathbf{A}_T) \subseteq \operatorname{col}(\mathbf{A})$, we have $\ell_z(T) \le \ell_z$ and therefore $L_s \le \operatorname{rank}(\mathbf{A}) = n - 1$. At $s = n$, we use $\ell_z^{(n)} = \ell_z$ and $L_n = n - 1$.

C.2 Spectral approximation under Bernoulli sampling

We first consider independent Bernoulli row sampling. In this subsection and Appendix C.3, m denotes the expected number of sampled rows. We prove concentration for a fixed support before taking a union bound over all supports of size at most s.

Bernoulli sampling setup

For each row $z \in [N_{\text{row}}]$, draw $\delta_z \sim \text{Bernoulli}(p_z)$ independently, and define

$$\mathbf{S} := \operatorname{diag}\left(\frac{\delta_1}{\sqrt{p_1}}, \dots, \frac{\delta_{N_{\text{row}}}}{\sqrt{p_{N_{\text{row}}}}} \right) \in \mathbb{R}^{N_{\text{row}} \times N_{\text{row}}}.$$

A row with $p_z = 0$ is assigned a zero diagonal entry. For every fixed support T,

$$\mathbf{S}^\top \mathbf{S} = \operatorname{diag}(\delta_z / p_z), \qquad \mathbf{U}_T^\top \mathbf{S}^\top \mathbf{S} \mathbf{U}_T = \sum_{z=1}^{N_{\text{row}}} \frac{\delta_z}{p_z} u_{z,T} u_{z,T}^\top.$$

Terms with $p_z = 0$ are zero by convention.

Sparse sampling probabilities. Fix $c > 0$ and set

$$p_z := \min\bigl(1, c\ell_z^{(s)}\bigr), \qquad z \in [N_{\text{row}}]. \tag{20}$$

Let the expected number of sampled rows be $m := \sum_z p_z$. If $\ell_z^{(s)} = 0$, then $u_{z,T} = 0$ for every $|T| \le s$, so omitting that row causes no error.

Fixed-support concentration

We begin with concentration on a fixed support.

Lemma 10 (Sparse spectral approximation, fixed support)**.** *Fix $\varepsilon, \delta \in (0,1)$ and construct $\mathbf{S}$ using (20). For any fixed $T \subseteq [n]$ with $|T| \le s$, let $d = \operatorname{rank}(\mathbf{A}_T)$ and let $\mathbf{U}_T$ be an orthonormal basis for $\operatorname{col}(\mathbf{A}_T)$. Then*

$$\mathbb{P}\big(\big\|I_d - \mathbf{U}_T^\top \mathbf{S}^\top \mathbf{S} \mathbf{U}_T\big\|_2 \le \varepsilon\big) \ge 1 - \delta$$

provided $c \ge C\varepsilon^{-2}\log(2s/\delta)$, for a universal constant $C > 0$. Taking $c = C\varepsilon^{-2}\log(2s/\delta)$ uses an expected number of rows at most $C(L_s/\varepsilon^2)\log(2s/\delta)$. The same probability conclusion also holds whenever $m \ge C(L_s/\varepsilon^2)\log(2s/\delta)$.

Proof. Fix $T \subset [n]$ with $|T| \le s$, and let $d := \operatorname{rank}(\mathbf{A}_T)$. If $d = 0$, the claim is immediate. Otherwise, use the orthonormal basis $\mathbf{U}_T$ defined above and define independent random $d \times d$ Hermitian matrices

$$X_z(T) := u_{z,T} u_{z,T}^\top - \frac{\delta_z}{p_z}\, u_{z,T} u_{z,T}^\top = \left(1 - \frac{\delta_z}{p_z}\right) u_{z,T} u_{z,T}^\top.$$

Rows with $p_z = 0$ contribute zero and are omitted from the following calculations. Since $\mathbb{E}[\delta_z] = p_z$, we have $\mathbb{E}[X_z(T)] = 0$. Moreover,

$$\sum_{z=1}^{N_{\mathrm{row}}} u_{z,T} u_{z,T}^\top = \mathbf{U}_T^\top \mathbf{U}_T = I_d,$$

so

$$\sum_{z=1}^{N_{\mathrm{row}}} X_z(T) = I_d - \mathbf{U}_T^\top S^\top S \mathbf{U}_T.$$

Thus it suffices to bound $\big\|\sum_z X_z(T)\big\|_2$.

Step 1: Uniform bound on $\|X_z(T)\|_2$. If $p_z = 1$, then $\delta_z = 1$ almost surely and $X_z(T) = 0$. Otherwise $p_z = c\,\ell_z^{(s)}$, and

$$\|X_z(T)\|_2 = \left|1 - \frac{\delta_z}{p_z}\right| \|u_{z,T}\|_2^2.$$

Since $\delta_z \in \{0,1\}$, we have $|1 - \delta_z/p_z| \le 1/p_z$. Therefore

$$\|X_z(T)\|_2 \le \frac{1}{p_z}\|u_{z,T}\|_2^2 \le \frac{1}{p_z}\ell_z^{(s)} = \frac{1}{c}.$$

Hence $L := \max_z \|X_z(T)\|_2 \le 1/c$.

Step 2: Variance proxy bound. Using $(uu^\top)^2 = \|u\|_2^2 uu^\top$,

$$X_z(T)^2 = \left(1 - \frac{\delta_z}{p_z}\right)^2 \|u_{z,T}\|_2^2\, u_{z,T} u_{z,T}^\top.$$

Also

$$\mathbb{E}\left[\left(1 - \frac{\delta_z}{p_z}\right)^2\right] = (1 - p_z) + p_z\left(1 - \frac{1}{p_z}\right)^2 = \frac{1}{p_z} - 1 \le \frac{1}{p_z}.$$

If $p_z = 1$, then $\mathbb{E}[X_z(T)^2] = 0$. For rows with $0 < p_z < 1$, $p_z = c\ell_z^{(s)}$, so

$$\mathbb{E}[X_z(T)^2] \preceq \frac{1}{p_z}\|u_{z,T}\|_2^2\, u_{z,T} u_{z,T}^\top \preceq \frac{\ell_z^{(s)}}{p_z}\, u_{z,T} u_{z,T}^\top \preceq \frac{1}{c}\, u_{z,T} u_{z,T}^\top.$$

Summing over z gives

$$\sum_{z=1}^{N_{\mathrm{row}}} \mathbb{E}[X_z(T)^2] \preceq \frac{1}{c}\sum_{z=1}^{N_{\mathrm{row}}} u_{z,T} u_{z,T}^\top = \frac{1}{c} I_d,$$

so the variance proxy satisfies $V := \big\|\sum_z \mathbb{E}[X_z(T)^2]\big\|_2 \le 1/c$.

Step 3: Matrix Bernstein. Applying matrix Bernstein from Tropp [2012, Theorem 6.1] to both $X_z(T)$ and $-X_z(T)$,

$$\Pr\left(\left\|\sum_{z=1}^{N_{\text{row}}} X_z(T)\right\|_2 \geq \varepsilon\right) \leq 2d \cdot \exp\left(-\frac{\varepsilon^2/2}{V + L\varepsilon/3}\right).$$

Using $V \leq 1/c$, $L \leq 1/c$, and $\varepsilon \in (0,1)$, there exists a universal $c_0 > 0$ such that

$$\Pr\left(\left\|I_d - \mathbf{U}_T^\top \mathbf{S}^\top \mathbf{S}\mathbf{U}_T\right\|_2 \geq \varepsilon\right) \leq 2d \cdot \exp\left(-c_0\, c\, \varepsilon^2\right).$$

Thus it suffices to choose $c \gtrsim \varepsilon^{-2} \log(2d/\delta)$.

Step 4: Relating c to the number of sampled rows. Recall that rows are sampled independently with probabilities $p_z = \min(1, c\,\ell_z^{(s)})$. Using $\min(1,x) \leq x$,

$$m = \sum_{z=1}^{N_{\text{row}}} \min(1, c\,\ell_z^{(s)}) \leq c \sum_{z=1}^{N_{\text{row}}} \ell_z^{(s)} = cL_s.$$

Hence $c \geq m/L_s$. Substituting this into the Bernstein bound gives

$$\Pr\left(\left\|I_d - \mathbf{U}_T^\top \mathbf{S}^\top \mathbf{S}\mathbf{U}_T\right\|_2 \geq \varepsilon\right) \leq 2d \cdot \exp\left(-\Omega\left(\frac{m\varepsilon^2}{L_s}\right)\right).$$

Setting the right-hand side to be at most δ and using $d \leq s$ yields

$$m \gtrsim \frac{L_s}{\varepsilon^2} \log \frac{2s}{\delta},$$

which is the stated bound. □

Union bound over supports

Bernoulli version of Lemma 2. Apply Lemma 10 to every $T \subseteq [n]$ with $|T| \leq s$, and take a union bound over the

$$\sum_{d=1}^{s} \binom{n}{d} \leq \left(\frac{en}{s}\right)^s$$

possible nonempty supports. The factor $2s$ in the fixed-support bound is absorbed by $s\log(en/s)$, so it suffices to choose

$$c \geq \frac{C}{\varepsilon^2}\left(s \log \frac{en}{s} + \log \frac{1}{\delta}\right).$$

Taking c equal to the right-hand side gives an expected number of sampled rows satisfying

$$m \leq C \frac{L_s}{\varepsilon^2}\left(s \log \frac{en}{s} + \log \frac{1}{\delta}\right).$$

Equivalently, for a sketch specified by its expected row count, the same concentration conclusion follows from the sufficient condition

$$m \geq C \frac{L_s}{\varepsilon^2}\left(s \log \frac{en}{s} + \log \frac{1}{\delta}\right),$$

because $m \leq cL_s$ implies $c \geq m/L_s$. This proves the Bernoulli version of the sparse spectral approximation. □

C.3 Fixed-residual approximate matrix multiplication

The sketched-regression argument used in Appendix D also needs an AMM bound for the population residual. This residual is fixed independently of the sketch. The AMM bound is uniform over sparse directions for that one fixed residual; its probability event need not hold for all residuals simultaneously. We first record the maximal second-moment inequality used in its proof.

Lemma 11 (Maximal second-moment bound)**.** *Let $X_1, ..., X_N$ be zero-mean independent random vectors in $\mathbb{R}^d$ satisfying $\mathbb{E}\|X_i\|_\infty^2 < \infty$. Then*

$$\mathbb{E}\left\|\sum_{i=1}^N X_i\right\|_\infty^2 \le 8(1+\log(2d))\sum_{i=1}^N \mathbb{E}\|X_i\|_\infty^2 \tag{21}$$

Proof. Let $X_1', \dots, X_N'$ be independent copies of $X_1, \dots, X_N$. Since $\mathbb{E}[X_i'] = 0$, Jensen's inequality gives

$$\mathbb{E}\left\|\sum_{i=1}^N X_i\right\|_\infty^2 \le \mathbb{E}\left\|\sum_{i=1}^N (X_i - X_i')\right\|_\infty^2.$$

Let $\sigma_1, \dots, \sigma_N$ be independent Rademacher variables. Each $X_i - X_i'$ is symmetric, and therefore

$$\sum_{i=1}^N (X_i - X_i') \overset{d}{=} \sum_{i=1}^N \sigma_i (X_i - X_i').$$

Using $\|a-b\|_\infty^2 \le 2\|a\|_\infty^2 + 2\|b\|_\infty^2$ and the fact that $(X_i')_{i=1}^N$ has the same distribution as $(X_i)_{i=1}^N$, we obtain

$$\mathbb{E}\left\|\sum_{i=1}^N X_i\right\|_\infty^2 \le 4\mathbb{E}\left\|\sum_{i=1}^N \sigma_i X_i\right\|_\infty^2.$$

Conditionally on $X_1, \dots, X_N$, set $v^2 = \max_{j\in[d]} \sum_i X_{ij}^2$. The scalar Rademacher moment generating function bound and a union bound give

$$\mathbb{P}_\sigma\left(\left\|\sum_i \sigma_i X_i\right\|_\infty > t\right) \le 2d\exp\left(-\frac{t^2}{2v^2}\right)$$

when $v > 0$. Integrating this tail above $t^2 = 2v^2\log(2d)$, and bounding the remaining probability by one, yields

$$\mathbb{E}_\sigma\left\|\sum_{i=1}^N \sigma_i X_i\right\|_\infty^2 \le 2\big(1+\log(2d)\big)\max_{j\in[d]}\sum_{i=1}^N X_{ij}^2.$$

The same inequality is trivial when $v = 0$. Moreover,

$$\max_{j\in[d]}\sum_{i=1}^N X_{ij}^2 \le \sum_{i=1}^N \max_{j\in[d]} X_{ij}^2 = \sum_{i=1}^N \|X_i\|_\infty^2.$$

Taking expectation over $X_1, \dots, X_N$ and combining the preceding inequalities gives

$$\mathbb{E}\left\|\sum_{i=1}^N X_i\right\|_\infty^2 \le 8\big(1+\log(2d)\big)\sum_{i=1}^N \mathbb{E}\|X_i\|_\infty^2,$$

which proves the claim. □

We now apply this inequality to the Bernoulli sketch of Appendix C.2.

Lemma 12 (Fixed-residual sparse approximate matrix multiplication)**.** *Let $\mathbf{A}$ be the projected Shapley design, let $\mathbf{S}$ be constructed as in (20), and fix a nonzero vector $r \in \mathbb{R}^{N_{\text{row}}}$ independently of $\mathbf{S}$ with $\mathbf{A}^\top r = 0$. There is a universal constant $C > 0$ such that*

$$\mathbb{E}_{\mathbf{S}}\left[\sup_{\substack{T\subseteq[n],\,|T|\le s\\ x\in\mathbb{R}^{|T|},\,\mathbf{A}_T x\neq 0}} \frac{\big|\langle \mathbf{S}\mathbf{A}_T x, \mathbf{S}r\rangle\big|^2}{\|\mathbf{A}_T x\|_2^2\|r\|_2^2}\right] \le C\,\frac{sL_s\log(2n)}{m}.$$

Consequently, for any $\alpha, \delta \in (0,1)$, if

$$m \ge C\,\frac{sL_s\log(2n)}{\alpha^2\delta},$$

then, with probability at least $1-\delta$, simultaneously for every $T \subseteq [n]$ with $|T| \le s$ and every $x \in \mathbb{R}^{|T|}$,

$$\big|\langle \mathbf{S}\mathbf{A}_T x, \mathbf{S}r\rangle\big| \le \alpha\|\mathbf{A}_T x\|_2\|r\|_2.$$

Proof. Set

$$g := \mathbf{A}^\top(\mathbf{S}^\top\mathbf{S} - I)r.$$

For $T \subseteq [n]$ and $x \in \mathbb{R}^{|T|}$, let $h \in \mathbb{R}^n$ be supported on T with $h_T = x$. Then $\mathbf{A}_T x = \mathbf{A}h$ and

$$\langle \mathbf{S}\mathbf{A}_T x, \mathbf{S}r\rangle = h^\top g.$$

Thus it suffices to control $\|g\|_\infty$ and then convert this coordinatewise control into a bound uniform over sparse h.

The sampling probabilities provide the required coordinatewise control. For every $j \in [n]$, the singleton support $\{j\}$ is included in the maximization defining $\ell_z^{(s)}$. Since $\mathbf{A}^\top\mathbf{A} = P/n$,

$$\ell_z^{(s)} \geq \ell_z(\{j\}) = \frac{[\mathbf{A}]_{zj}^2}{(\mathbf{A}^\top\mathbf{A})_{jj}} = \frac{n^2}{n-1}[\mathbf{A}]_{zj}^2.$$

Consequently,

$$\|[\mathbf{A}]_z\|_\infty^2 \leq \frac{n-1}{n^2}\ell_z^{(s)}. \tag{22}$$

Using the Bernoulli variables from the sampling setup, write

$$g = \sum_{z=1}^{N_{\text{row}}} X_z, \qquad X_z := \left(\frac{\delta_z}{p_z} - 1\right) r_z [\mathbf{A}]_z^\top.$$

Rows with $p_z = 0$ are assigned $X_z = 0$; their design rows vanish. The vectors X_z are independent and mean zero. By Lemma 11, symmetrization followed by the conditional Rademacher maximal inequality gives

$$\mathbb{E}\|g\|_\infty^2 \leq 8\big(1 + \log(2n)\big) \sum_{z=1}^{N_{\text{row}}} \mathbb{E}\|X_z\|_\infty^2. \tag{23}$$

Rows with $p_z = 0$ contribute zero. If $p_z = 1$, then $X_z = 0$ almost surely. Otherwise $0 < p_z = c\ell_z^{(s)} < 1$, and

$$\mathbb{E}\left(\frac{\delta_z}{p_z} - 1\right)^2 = \frac{1}{p_z} - 1.$$

Together with (22), this yields

$$\mathbb{E}\|X_z\|_\infty^2 \leq \frac{r_z^2}{c\ell_z^{(s)}} \frac{n-1}{n^2}\ell_z^{(s)} = \frac{n-1}{cn^2} r_z^2.$$

Rows with $\ell_z^{(s)} = 0$ satisfy $[\mathbf{A}]_z = 0$ by (22) and make no contribution. Substituting the preceding bound into (23) gives

$$\mathbb{E}\|g\|_\infty^2 \leq C\,\frac{n-1}{cn^2}\log(2n)\|r\|_2^2.$$

Moreover, $m = \sum_z p_z \leq c\sum_z \ell_z^{(s)} = cL_s$, so $1/c \leq L_s/m$. Therefore

$$\mathbb{E}\|g\|_\infty^2 \leq C\,\frac{L_s\log(2n)}{nm}\|r\|_2^2. \tag{24}$$

We next convert (24) into a uniform sparse-direction bound. Suppose first that $s \leq n/2$. If $\|h\|_0 \leq s$, then

$$|h^\top g| \leq \|h\|_1\|g\|_\infty \leq \sqrt{s}\|h\|_2\|g\|_\infty.$$

The identity $\mathbf{A}^\top\mathbf{A} = P/n$ and the inequality $(\mathbf{1}^\top h)^2 \leq s\|h\|_2^2$ imply

$$\|\mathbf{A}h\|_2^2 = \frac{1}{n}\left(\|h\|_2^2 - \frac{(\mathbf{1}^\top h)^2}{n}\right) \geq \frac{n-s}{n^2}\|h\|_2^2.$$

Combining these inequalities with the bound preceding (24) gives

$$\mathbb{E}\sup_{\substack{\|h\|_0 \leq s \\ \mathbf{A}h \neq 0}} \frac{|h^\top g|^2}{\|\mathbf{A}h\|_2^2\|r\|_2^2} \leq C\,\frac{s(n-1)}{n-s}\,\frac{L_s\log(2n)}{m} \leq C\,\frac{sL_s\log(2n)}{m}.$$

For $s > n/2$, we use a bound valid for every $h \in \mathbb{R}^n$. Since $\mathbf{A}\mathbf{1} = 0$, we have $g \in \mathbf{1}^\perp$, and hence $h^\top g = (Ph)^\top g$. Also $\|\mathbf{A}h\|_2 = \|Ph\|_2/\sqrt{n}$. It follows that

$$|h^\top g| \le \|Ph\|_2 \|g\|_2 \le n \|\mathbf{A}h\|_2 \|g\|_\infty.$$

Using again the bound preceding (24),

$$\mathbb{E} \sup_{\substack{h \in \mathbb{R}^n \\ \mathbf{A}h \neq 0}} \frac{|h^\top g|^2}{\|\mathbf{A}h\|_2^2 \|r\|_2^2} \le C\,(n-1)\frac{L_s \log(2n)}{m} \le C\,\frac{sL_s \log(2n)}{m}.$$

The two cases prove the stated second-moment bound. Markov's inequality then gives

$$\mathbb{P}\left(\sup_{\substack{T \subseteq [n],\, |T| \le s \\ x \in \mathbb{R}^{|T|},\, \mathbf{A}_T x \neq 0}} \frac{\left|\langle \mathbf{S}\mathbf{A}_T x, \mathbf{S}r\rangle - \langle \mathbf{A}_T x, r\rangle\right|}{\|\mathbf{A}_T x\|_2 \|r\|_2} > \alpha \right) \le C\,\frac{sL_s \log(2n)}{m\alpha^2},$$

which is at most δ under the stated lower bound on m. Finally, if $\mathbf{A}^\top r = 0$, then $\langle \mathbf{A}_T x, r\rangle = x^\top \mathbf{A}_T^\top r = 0$. □

C.4 Paired sampling and proof of Lemma 2

Algorithms 1 and 3 use a fixed number of independently sampled subset pairs rather than independent Bernoulli row indicators. In this subsection, m denotes the number of independently sampled pairs, corresponding to $2m$ sampled rows. We give the spectral and fixed-residual AMM bounds for this sampling scheme, including the dependence within each pair.

Lemma 13 (Spectral approximation and fixed-residual AMM for paired sampling)**.** *Fix $s \in \{1, \dots, n\}$ and let $\pi_z = \ell_z^{(s)}/L_s$. Draw $z_1, \dots, z_m$ independently from π and retain both z_i and its complement $\bar{z}_i$. Let $\mathbf{S} \in \mathbb{R}^{2m \times N_{\mathrm{row}}}$ have rows*

$$\frac{e_{z_i}^\top}{\sqrt{2m\pi_{z_i}}}, \qquad \frac{e_{\bar{z}_i}^\top}{\sqrt{2m\pi_{z_i}}}, \qquad i \in [m],$$

where e_z is the coordinate vector of row z. There is a universal constant $C > 0$ such that the following hold. ***(i) Spectral approximation.*** *For every $\varepsilon, \delta \in (0,1)$, if*

$$m \ge C\frac{L_s}{\varepsilon^2}\left(s \log\frac{en}{s} + \log\frac{1}{\delta}\right),$$

then, with probability at least $1-\delta$, simultaneously for every $|T| \le s$,

$$\left\| I - \mathbf{U}_T^\top \mathbf{S}^\top \mathbf{S} \mathbf{U}_T \right\|_2 \le \varepsilon.$$

(ii) Fixed-residual AMM. *Moreover, for any fixed nonzero $r \in \mathbb{R}^{N_{\mathrm{row}}}$ independent of the samples with $\mathbf{A}^\top r = 0$,*

$$\mathbb{E}\left[\sup_{\substack{\|h\|_0 \le s \\ \mathbf{A}h \neq 0}} \frac{|\langle \mathbf{S}\mathbf{A}h, \mathbf{S}r\rangle|^2}{\|\mathbf{A}h\|_2^2 \|r\|_2^2} \right] \le C\frac{sL_s \log(2n)}{m}.$$

Consequently, the high-probability AMM conclusion of Lemma 12 holds under the same requirement $m \ge CsL_s \log(2n)/(\alpha^2\delta)$. Each draw contributes two rows and requires at most two value-function evaluations.

Proof. (i) Spectral approximation. Complementary subsets have the same Kernel SHAP weight and satisfy

$$[\mathbf{A}]_{\bar{z}} = -[\mathbf{A}]_z, \qquad \ell_{\bar{z}}^{(s)} = \ell_z^{(s)}, \qquad \pi_{\bar{z}} = \pi_z.$$

For a fixed support T with rank $d > 0$, write $\mathbf{U}_T = \mathbf{A}_T R_T$ for an appropriate matrix R_T. Thus $u_{\bar{z},T} = -u_{z,T}$, and the two contributions within a pair give

$$\mathbf{U}_T^\top \mathbf{S}^\top \mathbf{S} \mathbf{U}_T = \frac{1}{m}\sum_{i=1}^m \frac{u_{z_i,T} u_{z_i,T}^\top}{\pi_{z_i}}.$$

Set $C_i = u_{z_i,T}u_{z_i,T}^\top/(m\pi_{z_i})$ and $Y_i = C_i - I_d/m$. Since $\|u_{z,T}\|_2^2 \le \ell_z^{(s)}$, these matrices satisfy

$$\mathbb{E}C_i = I_d/m, \qquad 0 \preceq C_i \preceq (L_s/m)I_d.$$

Also, $L_s \ge \sum_z \ell_z(T) = d \ge 1$, so $\|Y_i\|_2 \le L_s/m$. For the variance,

$$\mathbb{E}C_i^2 = \frac{1}{m^2}\sum_z \frac{\|u_{z,T}\|_2^2}{\pi_z} u_{z,T}u_{z,T}^\top \preceq \frac{L_s}{m^2} I_d, \qquad \sum_{i=1}^m \mathbb{E}Y_i^2 \preceq \frac{L_s}{m} I_d.$$

The matrices Y_i are independent and mean zero. Applying matrix Bernstein to both signs, as above, gives

$$\mathbb{P}\left(\left\|\sum_{i=1}^m Y_i\right\|_2 > \varepsilon\right) \le 2d\exp\left(-c_0 m\varepsilon^2/L_s\right).$$

The rank-zero case is trivial. Taking a union bound over $|T| \le s$ proves the stated spectral bound.

(ii) Fixed-residual AMM. For AMM, define $g = \mathbf{A}^\top(\mathbf{S}^\top\mathbf{S} - I)r$. Orthogonality and the complementary-row identity imply

$$g = \sum_{i=1}^m X_i, \qquad X_i = \frac{[\mathbf{A}]_{z_i}^\top (r_{z_i} - r_{\bar{z}_i})}{2m\pi_{z_i}}.$$

Each X_i is mean zero: changing variables $z \mapsto \bar{z}$ gives

$$\mathbb{E}X_i = \frac{1}{2m}\sum_z [\mathbf{A}]_z^\top (r_z - r_{\bar{z}}) = \frac{1}{m}\mathbf{A}^\top r = 0.$$

The vectors X_i are independent across pairs. By (22),

$$\mathbb{E}\|X_i\|_\infty^2 = \frac{1}{4m^2}\sum_z \frac{\|[\mathbf{A}]_z\|_\infty^2}{\pi_z}(r_z - r_{\bar{z}})^2 \le \frac{(n-1)L_s}{4n^2m^2}\sum_z (r_z - r_{\bar{z}})^2 \le \frac{(n-1)L_s}{n^2m^2}\|r\|_2^2.$$

The last inequality uses $(a-b)^2 \le 2a^2 + 2b^2$ and the fact that complementation permutes the rows. Lemma 11 therefore yields

$$\mathbb{E}\|g\|_\infty^2 \le C\frac{(n-1)L_s\log(2n)}{n^2 m}\|r\|_2^2.$$

This is the coordinatewise second-moment bound used in Lemma 12. To make the conversion explicit, when $s \le n/2$ and $\|h\|_0 \le s$,

$$|h^\top g|^2 \le s\|h\|_2^2\|g\|_\infty^2, \qquad \|\mathbf{A}h\|_2^2 \ge \frac{n-s}{n^2}\|h\|_2^2.$$

Taking the supremum and expectation gives an upper bound $CsL_s\log(2n)/m$ after division by $\|r\|_2^2$. If $s > n/2$, then $g \in \mathbf{1}^\perp$ and

$$|h^\top g| \le \|Ph\|_2\|g\|_2 \le n\|\mathbf{A}h\|_2\|g\|_\infty.$$

The same expectation is consequently at most $C(n-1)L_s\log(2n)/m \le CsL_s\log(2n)/m$. Finally, $\langle \mathbf{SA}h, \mathbf{S}r\rangle = h^\top g$ because $\mathbf{A}^\top r = 0$. Markov's inequality proves the high-probability AMM statement. □

Proof of Lemma 2. The paired version in Lemma 2 follows from Lemma 13 by expressing $\mathbf{A}_T x$ in the orthonormal basis $\mathbf{U}_T$. □

D Proofs of Theorems 4 and 6

We first derive the sparse embedding consequence of Lemma 2 and prove the single-run recovery guarantee. We then prove the shared confidence-amplification lemma and apply it to obtain the high-confidence guarantee.

D.1 Sparse subspace embedding

The following corollary applies Lemma 2 to the supports arising in the regression argument. The internal regression vector $\widehat{\phi}_\perp$ and the ground truth ϕ are both s-sparse, so their difference is $2s$-sparse. The returned efficient estimate need not be sparse. When $2s \geq n$, we use the convention $\ell_z^{(2s)} = \ell_z^{(n)} = \ell_z$.

Corollary 14 (Sparse subspace embedding)**.** *Let $q = \min(2s, n)$, and let $\mathbf{S}$ be the sampling-and-rescaling matrix of Algorithm 1. If*

$$m \geq C \frac{L_q}{\varepsilon^2} \left(q \log \frac{en}{q} + \log \frac{2}{\delta} \right),$$

then, with probability at least $1 - \delta$, simultaneously for every $T \subseteq [n]$ with $|T| \leq q$,

$$(1 - \varepsilon)\|\mathbf{A}_T x\|_2^2 \leq \|\mathbf{S}\mathbf{A}_T x\|_2^2 \leq (1 + \varepsilon)\|\mathbf{A}_T x\|_2^2, \qquad \forall x \in \mathbb{R}^{|T|}.$$

Proof. Let $\mathbf{U}_T$ be an orthonormal basis for $\mathrm{col}(\mathbf{A}_T)$. For every $x \in \mathbb{R}^{|T|}$ there exists y with $\mathbf{A}_T x = \mathbf{U}_T y$, and

$$\|\mathbf{A}_T x\|_2^2 = \|y\|_2^2, \qquad \|\mathbf{S}\mathbf{A}_T x\|_2^2 = y^\top \mathbf{U}_T^\top \mathbf{S}^\top \mathbf{S} \mathbf{U}_T y.$$

Lemma 2, with its paired-sampling version proved in Lemma 13, bounds the eigenvalues of the middle matrix between $1 - \varepsilon$ and $1 + \varepsilon$ uniformly over T. This proves the claim. □

D.2 Single-run recovery guarantee

We now turn this embedding guarantee into an approximation theorem for the ℓ_0-constrained sampled regression. The single-run proof follows the sketched least-squares argument underlying Musco and Witter [2025], using a bound for the fixed population residual.

Proof of Theorem 4. Set $q = \min(2s, n)$ and

$$r^\star := \mathbf{b} - \mathbf{A}\phi, \qquad L_\star := \|r^\star\|_2^2 = \|\mathbf{Z}\phi - \mathbf{y}\|_2^2.$$

The Shapley regression characterization implies $\mathbf{A}^\top r^\star = 0$. This residual is fixed independently of the sampling matrix.

Consider one run of Algorithm 1. By Corollary 14 and the paired-sampling version of Lemma 12, it suffices to take

$$m \geq C L_q \left(q \log \frac{en}{q} + \frac{q \log(2n)}{\epsilon} \right) \tag{25}$$

so that, with probability at least $3/4$, the following two inequalities hold simultaneously:

$$\frac{1}{2}\|\mathbf{A}_T z\|_2^2 \leq \|\mathbf{S}\mathbf{A}_T z\|_2^2 \leq \frac{3}{2}\|\mathbf{A}_T z\|_2^2, \qquad |T| \leq q, \quad z \in \mathbb{R}^{|T|}, \tag{26}$$

$$\left|\langle \mathbf{S}\mathbf{A}_T z, \mathbf{S} r^\star \rangle\right| \leq \frac{\sqrt{\epsilon}}{4}\|\mathbf{A}_T z\|_2 \|r^\star\|_2, \qquad |T| \leq q, \quad z \in \mathbb{R}^{|T|}. \tag{27}$$

Each event has failure probability at most $1/8$. If $r^\star = 0$, the second inequality holds identically. We work on their intersection.

Step 1: optimality with ϕ as competitor. Let $h = \widehat{\phi}_\perp - \phi$ and $T = \mathrm{supp}(\widehat{\phi}_\perp) \cup \mathrm{supp}(\phi)$, so $|T| \leq q$. Since ϕ is feasible for the sampled s-sparse regression,

$$\|\mathbf{S}(\mathbf{A}\widehat{\phi}_\perp - \mathbf{b})\|_2^2 \leq \|\mathbf{S}(\mathbf{A}\phi - \mathbf{b})\|_2^2 = \|\mathbf{S} r^\star\|_2^2.$$

Using $\mathbf{A}\widehat{\phi}_\perp - \mathbf{b} = \mathbf{A}h - r^\star$ and expanding gives

$$\|\mathbf{S}\mathbf{A}h\|_2^2 \leq 2\langle \mathbf{S}\mathbf{A}h, \mathbf{S} r^\star \rangle. \tag{28}$$

Step 2: bound the inner product via AMM and orthogonality. Because $\mathbf{A}^\top r^\star = 0$, we have $\langle \mathbf{A}h, r^\star \rangle = 0$. Applying (27) on T to h_T yields

$$\|\mathbf{S}\mathbf{A}h\|_2^2 \le \frac{\sqrt{\epsilon}}{2}\|\mathbf{A}h\|_2\|r^\star\|_2. \tag{29}$$

Step 3: convert to an unsketched bound. The lower embedding inequality in (26) and (29) imply

$$\frac{1}{2}\|\mathbf{A}h\|_2^2 \le \frac{\sqrt{\epsilon}}{2}\|\mathbf{A}h\|_2\|r^\star\|_2.$$

Thus

$$\|\mathbf{A}h\|_2^2 \le \epsilon L_\star. \tag{30}$$

The conclusion also holds when $\mathbf{A}h = 0$.

Step 4: Pythagorean decomposition. Orthogonality gives

$$\|\mathbf{A}\widehat{\boldsymbol{\phi}}_\perp - \mathbf{b}\|_2^2 = \|\mathbf{A}h - r^\star\|_2^2 = L_\star + \|\mathbf{A}h\|_2^2 \le (1+\epsilon)L_\star.$$

Step 5: restore efficiency. The returned estimate $\widehat{\boldsymbol{\phi}} = P\widehat{\boldsymbol{\phi}}_\perp + \Delta\mathbf{1}/n$ satisfies $\mathbf{Z}\widehat{\boldsymbol{\phi}} - \mathbf{y} = \mathbf{A}\widehat{\boldsymbol{\phi}}_\perp - \mathbf{b}$. Consequently, one run has excess loss at most $\epsilon L_\star$ with probability at least $3/4$, as claimed.

Sample complexity. Proposition 3 and $L_n = n - 1$ imply $L_q = O(s\log(en/s))$. Since $q \le 2s$ and $\epsilon < 1$, (25) is satisfied by

$$m = O\left(\frac{s^2}{\epsilon}\log\frac{en}{s}\log(2n)\right).$$

One run uses at most $2m + 2$ value-function evaluations. □

D.3 Outer-loop confidence amplification

We next prove Lemma 5, which applies the shared selection rule to independent efficient estimates with a constant-confidence guarantee.

Proof. Let $L_\star := \|\mathbf{Z}\boldsymbol{\phi} - \mathbf{y}\|_2^2$, and call estimate j good when its loss is at most $L_\star + E_0$. All estimates satisfy the same efficiency constraint as $\boldsymbol{\phi}$. Since $\boldsymbol{\phi}$ minimizes (2), least-squares orthogonality therefore gives the Pythagorean identity

$$\|\mathbf{Z}\widehat{\boldsymbol{\phi}}^{(j)} - \mathbf{y}\|_2^2 = L_\star + \left\|\mathbf{Z}\left(\widehat{\boldsymbol{\phi}}^{(j)} - \boldsymbol{\phi}\right)\right\|_2^2. \tag{31}$$

Thus every good run lies in the prediction-norm ball of radius $\rho := \sqrt{E_0}$ around $\boldsymbol{\phi}$.

Let $\mathcal{G}$ be the set of good runs. Independence and Hoeffding's inequality imply

$$\mathbb{P}\left(|\mathcal{G}| \le \frac{R}{2}\right) \le \exp\left(-2R\left(\frac{3}{4} - \frac{1}{2}\right)^2\right) = \exp(-R/8) \le \delta.$$

We now give the selection rule, which does not require $\boldsymbol{\phi}$. Set $q = (R+1)/2$ and, for $j, \ell \in [R]$, define

$$d(j,\ell) := \left\|\mathbf{Z}\left(\widehat{\boldsymbol{\phi}}^{(j)} - \widehat{\boldsymbol{\phi}}^{(\ell)}\right)\right\|_2.$$

These distances are computable from the candidates alone. Indeed, with $\Delta_{j\ell} := \widehat{\boldsymbol{\phi}}^{(j)} - \widehat{\boldsymbol{\phi}}^{(\ell)}$, the efficiency constraint and $\mathbf{A}^\top\mathbf{A} = P/n$ give $d(j,\ell)^2 = n^{-1}\Delta_{j\ell}^\top P\Delta_{j\ell}$. Let $\widehat{\rho}_j$ be the q-th smallest element of $\{d(j,\ell) : \ell \in [R]\}$, counting $d(j,j) = 0$, and select $\widehat{j} \in \arg\min_{j\in[R]} \widehat{\rho}_j$.

It remains to verify that this rule extracts an accurate candidate from the hidden majority. Work on the event $|\mathcal{G}| > R/2$, so $|\mathcal{G}| \ge q$. If $j, \ell \in \mathcal{G}$, the triangle inequality gives $d(j,\ell) \le 2\rho$. Hence every $j \in \mathcal{G}$ has at least q candidates within distance 2ρ, and consequently $\widehat{\rho}_{\widehat{j}} \le 2\rho$.

The ball centered at $\widehat{j}$ with radius $\widehat{\rho}_{\widehat{j}}$ contains at least q candidates, while $\mathcal{G}$ also contains at least q candidates. Since $2q > R$, these two sets intersect. Choose ℓ in their intersection. This candidate is good, and therefore

$$\left\|\mathbf{Z}\left(\widehat{\boldsymbol{\phi}}^{(\widehat{j})} - \boldsymbol{\phi}\right)\right\|_2 \le d(\widehat{j}, \ell) + \left\|\mathbf{Z}\left(\widehat{\boldsymbol{\phi}}^{(\ell)} - \boldsymbol{\phi}\right)\right\|_2 \le 3\rho.$$

Applying (31) to the selected candidate and using $\rho^2 = E_0$ yields

$$\|\mathbf{Z}\widehat{\boldsymbol{\phi}}^{(\widehat{j})} - \mathbf{y}\|_2^2 \le L_\star + 9\rho^2 = L_\star + 9E_0,$$

as claimed. □

D.4 High-confidence recovery guarantee

We conclude by applying Lemma 5 to the single-run guarantee of Theorem 4.

Proof of Theorem 6. Apply Theorem 4 with accuracy $\epsilon/9$. Increasing the universal constant in the per-run sample budget by a factor of nine therefore gives

$$\mathbb{P}\Big(\|\mathbf{Z}\widehat{\boldsymbol{\phi}}^{(j)} - \mathbf{y}\|_2^2 \le \|\mathbf{Z}\boldsymbol{\phi} - \mathbf{y}\|_2^2 + \frac{\epsilon}{9}\|\mathbf{Z}\boldsymbol{\phi} - \mathbf{y}\|_2^2\Big) \ge \frac{3}{4}.$$

Lemma 5, with $E_0 = (\epsilon/9)\|\mathbf{Z}\boldsymbol{\phi} - \mathbf{y}\|_2^2$, then proves the relative-error guarantee for Algorithm 2. Since $R = O(\log(2/\delta))$ independent runs use at most $R(2m+2)$ evaluations, the stated total query complexity follows. □

E Proof of Theorem 7

We will use the following lemma to prove Theorem 7.

Lemma 15 (IHT contraction under a restricted embedding)**.** *Let $\boldsymbol{\phi} \in \mathbb{R}^n$ and $\boldsymbol{\phi}^0 \in \mathbb{R}^n$ be s-sparse and suppose $c = B\boldsymbol{\phi} + e$. Consider the IHT update*

$$\boldsymbol{\phi}^{t+1} = H_s\left(\boldsymbol{\phi}^t + \eta B^\top (c - B\boldsymbol{\phi}^t)\right),$$

where H_s keeps the s largest entries in magnitude and $\eta > 0$. Assume that for every $\Omega \subseteq [n]$ with $|\Omega| \le 3s$, all eigenvalues of $B_\Omega^\top B_\Omega$ lie in $[\mu, M]$. Let $\beta := 2\max\{|1-\eta\mu|, |1-\eta M|\}$, and suppose $\beta < 1$. Define

$$\zeta_{3s} := \max_{|\Omega| \le 3s} \|(B^\top e)_\Omega\|_2.$$

Then,

$$\|\boldsymbol{\phi}^K - \boldsymbol{\phi}\|_2 \le \beta^K \|\boldsymbol{\phi}^0 - \boldsymbol{\phi}\|_2 + \frac{2\eta}{1-\beta}\zeta_{3s}.$$

Proof. Let $h^t := \boldsymbol{\phi}^t - \boldsymbol{\phi}$. Since $c = B\boldsymbol{\phi} + e$, the pre-thresholded vector satisfies

$$\boldsymbol{\phi}^t + \eta B^\top (c - B\boldsymbol{\phi}^t) - \boldsymbol{\phi} = (I - \eta B^\top B)h^t + \eta B^\top e.$$

Let

$$T_\star := \mathrm{supp}(\boldsymbol{\phi}), \qquad T_t := \mathrm{supp}(\boldsymbol{\phi}^t), \qquad T_{t+1} := \mathrm{supp}(\boldsymbol{\phi}^{t+1}),$$

and set $\Omega_t := T_\star \cup T_t \cup T_{t+1}$. Since $\boldsymbol{\phi}$, $\boldsymbol{\phi}^t$, and $\boldsymbol{\phi}^{t+1}$ are all s-sparse, $|\Omega_t| \le 3s$. Because H_s is the best s-sparse approximation in Euclidean norm and $\boldsymbol{\phi}$ is s-sparse,

$$\|\boldsymbol{\phi}^{t+1} - (\boldsymbol{\phi}^t + \eta B^\top (c - B\boldsymbol{\phi}^t))\|_2 \le \|\boldsymbol{\phi} - (\boldsymbol{\phi}^t + \eta B^\top (c - B\boldsymbol{\phi}^t))\|_2.$$

The terms outside Ω_t agree on the two sides and cancel after squaring. The triangle inequality on Ω_t therefore gives

$$\|\boldsymbol{\phi}^{t+1} - \boldsymbol{\phi}\|_2 \le 2\left\|\left((I - \eta B^\top B)h^t + \eta B^\top e\right)_{\Omega_t}\right\|_2.$$

Since h^t is supported on $T_\star \cup T_t \subseteq \Omega_t$,

$$\left\|\left((I - \eta B^\top B)h^t\right)_{\Omega_t}\right\|_2 = \left\|(I - \eta B_{\Omega_t}^\top B_{\Omega_t})h^t_{\Omega_t}\right\|_2.$$

By the restricted eigenvalue assumption,

$$\|I - \eta B_{\Omega_t}^\top B_{\Omega_t}\|_2 \le \max\{|1-\eta\mu|,\ |1-\eta M|\} = \frac{\beta}{2}.$$

The factor two in the definition of β accounts for the hard-thresholding step. Thus

$$\|\boldsymbol{\phi}^{t+1} - \boldsymbol{\phi}\|_2 \le \beta\|\boldsymbol{\phi}^t - \boldsymbol{\phi}\|_2 + 2\eta\zeta_{3s}.$$

Iterating this recursion proves the claim. □

Proof of Theorem 7. We first prove a constant-confidence guarantee for one run, with tolerance $\varepsilon_0 := \varepsilon/9$. Let $\Delta := v([n]) - v(\varnothing)$ and $r_\star := \mathbf{b} - \mathbf{A}\boldsymbol{\phi}$. Since $\boldsymbol{\phi}$ is the population least-squares solution of the projected Shapley regression,

$$\mathbf{A}^\top r_\star = 0.$$

In particular, $r_\star$ is fixed independently of the random sketch.

Let $\mathbf{S}$ be the sampling-and-rescaling matrix induced by one run of Algorithm 3, using m_0 subset pairs. We use the IHT system $B = \mathbf{SA}$ and $c = \mathbf{Sb}$. Since $\mathbf{b} = \mathbf{A}\boldsymbol{\phi} + r_\star$, we have

$$c = B\boldsymbol{\phi} + e, \qquad e := \mathbf{S}r_\star.$$

Set $\tau := 1/6$ and $\alpha := \sqrt{\varepsilon_0/128}$. By the spectral and fixed-residual bounds for paired sampling in Lemma 13, applied with sparsity level $3s$, it suffices to take

$$m_0 = O\left(\frac{sL_{3s}\log(2n)}{\varepsilon_0}\right)$$

with a sufficiently large universal constant. The embedding fails with probability at most $1/8$, and the fixed-residual AMM inequality fails with probability at most $1/8$. Consequently, with probability at least $3/4$, the following two inequalities hold simultaneously for every $\Omega \subseteq [n]$ with $|\Omega| \le 3s$:

$$(1-\tau)\|\mathbf{A}_\Omega u\|_2^2 \le \|\mathbf{SA}_\Omega u\|_2^2 \le (1+\tau)\|\mathbf{A}_\Omega u\|_2^2 \qquad \forall u \in \mathbb{R}^{|\Omega|},$$

and

$$|\langle \mathbf{SA}_\Omega u, \mathbf{S}r_\star\rangle| \le \alpha\|\mathbf{A}_\Omega u\|_2\|r_\star\|_2 \qquad \forall u \in \mathbb{R}^{|\Omega|}.$$

When $r_\star = 0$, the second inequality holds identically. We condition on this event for the single-run analysis.

Fix any Ω with $|\Omega| \le 3s$. Since $\mathbf{A}^\top\mathbf{A} = P/n$, we have

$$\mathbf{A}_\Omega^\top\mathbf{A}_\Omega = \frac{1}{n}\left(I_{|\Omega|} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top\right).$$

Hence the eigenvalues of $\mathbf{A}_\Omega^\top\mathbf{A}_\Omega$ lie in

$$\left[\frac{1}{n}\left(1 - \frac{|\Omega|}{n}\right), \frac{1}{n}\right].$$

Using the sparse embedding event, the eigenvalues of $B_\Omega^\top B_\Omega = \mathbf{A}_\Omega^\top\mathbf{S}^\top\mathbf{SA}_\Omega$ therefore lie in $[\mu, M]$, where

$$\mu := \frac{1-\tau}{n}\left(1 - \frac{3s}{n}\right), \qquad M := \frac{1+\tau}{n}.$$

Choose $\eta = n$. Since $s/n \le 1/12$ and $\tau = 1/6$,

$$\beta := 2\max\{|1-\eta\mu|,\ |1-\eta M|\} = 2\max\left\{1 - (1-\tau)\left(1 - \frac{3s}{n}\right), \tau\right\} \le \frac{3}{4}.$$

By Lemma 15 and $\boldsymbol{\phi}^0 = 0$,

$$\|\boldsymbol{\phi}^K - \boldsymbol{\phi}\|_2 \le \beta^K\|\boldsymbol{\phi}\|_2 + \frac{2\eta}{1-\beta}\zeta_{3s}.$$

It remains to bound ζ_{3s}. For any Ω with $|\Omega| \le 3s$,

$$\|(B^\top e)_\Omega\|_2 = \sup_{\|u\|_2=1} |\langle B_\Omega u, e\rangle| = \sup_{\|u\|_2=1} |\langle \mathbf{S}\mathbf{A}_\Omega u, \mathbf{S}r_\star\rangle|.$$

The fixed-residual AMM inequality gives

$$|\langle \mathbf{S}\mathbf{A}_\Omega u, \mathbf{S}r_\star\rangle| \le \alpha \|\mathbf{A}_\Omega u\|_2 \|r_\star\|_2.$$

Moreover, $\mathbf{A}^\top\mathbf{A} = P/n$ implies $\|\mathbf{A}_\Omega u\|_2 \le \|u\|_2/\sqrt{n}$. Thus

$$\zeta_{3s} \le \frac{\alpha}{\sqrt{n}}\|r_\star\|_2, \qquad \|\boldsymbol{\phi}^K - \boldsymbol{\phi}\|_2 \le \beta^K \|\boldsymbol{\phi}\|_2 + 8\alpha\sqrt{n}\|r_\star\|_2.$$

Choose $K = O(\log(2/\varepsilon_0))$ sufficiently large that $\beta^{2K} \le \varepsilon_0/2$. Squaring the last bound and using $\alpha^2 = \varepsilon_0/128$ yields

$$\|\boldsymbol{\phi}^K - \boldsymbol{\phi}\|_2^2 \le 2\beta^{2K}\|\boldsymbol{\phi}\|_2^2 + 128\alpha^2 n\|r_\star\|_2^2 \le \varepsilon_0\|\boldsymbol{\phi}\|_2^2 + \varepsilon_0 n\|r_\star\|_2^2.$$

Now use the orthogonality $\mathbf{A}^\top r_\star = 0$. Since $\mathbf{A}\boldsymbol{\phi}^K - \mathbf{b} = \mathbf{A}(\boldsymbol{\phi}^K - \boldsymbol{\phi}) - r_\star$,

$$\|\mathbf{A}\boldsymbol{\phi}^K - \mathbf{b}\|_2^2 = \|\mathbf{A}(\boldsymbol{\phi}^K - \boldsymbol{\phi})\|_2^2 + \|r_\star\|_2^2.$$

Also $\|\mathbf{A}(\boldsymbol{\phi}^K - \boldsymbol{\phi})\|_2^2 \le n^{-1}\|\boldsymbol{\phi}^K - \boldsymbol{\phi}\|_2^2$. Therefore

$$\|\mathbf{A}\boldsymbol{\phi}^K - \mathbf{b}\|_2^2 \le (1+\varepsilon_0)\|\mathbf{A}\boldsymbol{\phi} - \mathbf{b}\|_2^2 + \frac{\varepsilon_0}{n}\|\boldsymbol{\phi}\|_2^2.$$

Finally, one run returns $\widehat{\boldsymbol{\phi}} = P\boldsymbol{\phi}^K + \Delta\mathbf{1}/n$, so

$$\mathbf{Z}\widehat{\boldsymbol{\phi}} - \mathbf{y} = \mathbf{Z}P\boldsymbol{\phi}^K + \mathbf{Z}\mathbf{1}\cdot\frac{\Delta}{n} - \mathbf{y} = \mathbf{A}\boldsymbol{\phi}^K - \mathbf{b}.$$

Since $\mathbf{1}^\top\boldsymbol{\phi} = \Delta$, we also have $P\boldsymbol{\phi} + \Delta\mathbf{1}/n = \boldsymbol{\phi}$ and hence $\mathbf{Z}\boldsymbol{\phi} - \mathbf{y} = \mathbf{A}\boldsymbol{\phi} - \mathbf{b}$. Writing $L_\star := \|\mathbf{Z}\boldsymbol{\phi} - \mathbf{y}\|_2^2$, the single-run guarantee is therefore

$$\|\mathbf{Z}\widehat{\boldsymbol{\phi}} - \mathbf{y}\|_2^2 \le L_\star + E_0, \qquad E_0 := \varepsilon_0\left(L_\star + \frac{\|\boldsymbol{\phi}\|_2^2}{n}\right),$$

with probability at least $3/4$.

All IHT outputs satisfy the efficiency constraint. Apply Algorithm 2 to $R = O(\log(2/\delta))$ independent runs. By Lemma 5, with probability at least $1-\delta$, the selected estimate satisfies

$$\|\mathbf{Z}\widehat{\boldsymbol{\phi}} - \mathbf{y}\|_2^2 \le L_\star + 9E_0 = (1+\varepsilon)L_\star + \frac{\varepsilon}{n}\|\boldsymbol{\phi}\|_2^2,$$

where the last equality uses $\varepsilon_0 = \varepsilon/9$. Proposition 3 gives $L_{3s} = O(s\log(en/s))$. Thus the total number of sampled pairs is

$$Rm_0 = O\left(\frac{s^2}{\varepsilon}\log\frac{en}{s}\,\log(2n)\,\log\frac{2}{\delta}\right).$$

The candidate selection uses no additional value-function queries. □

F Additional experimental details and figures

This appendix provides full protocols, hyperparameters, and supplementary figures for the experiments in Section 5.

F.1 Datasets, models, and estimators

Datasets and models (Section 5.1). The eight datasets are obtained via `shap.datasets`: IRIS, California, Diabetes, Adult, Correlated (`corrgroups60`), Independent (`independentlinear60`), NHANES, and Communities (`communitiesandcrime`); feature counts range from $n = 4$ to $n = 101$. We fit one XGBoost model per dataset: 100 trees of depth 4 for the smaller datasets, 50 trees of depth 3 for the synthetic generators (Correlated, Independent), and 80 trees of depth 4 for NHANES and Communities. Interventional Shapley values use a background set of 20 points drawn uniformly from X. We run all four estimators at sample budgets $m \in \{0.2n, 0.5n, n, 2n, 5n, 10n, 20n, 40n, 80n, 160n\}$ and report medians with Q1–Q3 bands across 100 test points per dataset.

Estimators. We compare four estimators: (i) *Kernel SHAP* [Lundberg and Lee, 2017] with replacement; (ii) *Optimised Kernel SHAP* (the `shap` library [Lundberg et al., 2020]); (iii) *Leverage SHAP* [Musco and Witter, 2025], which samples subset sizes uniformly with paired complements and solves a weighted projected OLS problem; and (iv) *Sparse Leverage SHAP* (ours), which samples subset sizes in proportion to the closed-form $\ell_z^{(s)}$ of Proposition 1 and solves the weighted projected s-sparse regression of Algorithm 1. All four enforce $\mathbf{1}^\top x = v([n]) - v(\emptyset)$ via projection.

Error metric. We report the normalised squared ℓ_2 error $\|\widehat{\boldsymbol{\phi}} - \boldsymbol{\phi}\|_2^2 / \|\boldsymbol{\phi}\|_2^2$, matching Musco and Witter [2025]. On real data, ground truth $\boldsymbol{\phi}$ is from Tree SHAP [Lundberg et al., 2020] on an XGBoost model [Chen and Guestrin, 2016]. On the synthetic game, ground truth is computed in closed form via an active-player reduction exploiting the Null Player axiom, using 2^s oracle calls per active player rather than 2^n.

F.2 Predicted-vs-true scatter (Section 5.1)

Figure 4 provides a complementary view of the benchmark experiment in Section 5.1: at $m = 5n$, points concentrated along the identity line indicate accurate estimates. Across the eight datasets, both leverage methods produce predictions that are visually closer to the diagonal than Kernel SHAP, with the Sparse and Dense Leverage variants nearly indistinguishable at this budget on the smaller datasets and Sparse Leverage SHAP showing reduced spread on the higher-n datasets (NHANES, Communities) where the dense design matrix is poorly conditioned.

F.3 Empirical sparsity of Tree SHAP vectors (Section 5.2)

Sparsity measures. The *effective sparsity* is $s_{\text{eff}}(\boldsymbol{\phi}) = |\{i : |\phi_i| > \tau \|\boldsymbol{\phi}\|_\infty\}|$ with $\tau = 0.05$, and the *tail mass* is $\|\boldsymbol{\phi} - \boldsymbol{\phi}_{s_{\text{eff}}}\|_2^2 / \|\boldsymbol{\phi}\|_2^2$, where $\boldsymbol{\phi}_s$ keeps the top-s entries of $\boldsymbol{\phi}$ in absolute value. Both are computed on Tree SHAP ground truth over 30 test points per dataset.

Dataset	n	s_{eff} (mean)	s_{eff}/n	Tail mass
Independent	60	9.5	0.16	0.001
Communities	101	34.8	0.34	0.013
Correlated	60	26.0	0.43	0.001

Table A: Empirical sparsity of Tree SHAP vectors, averaged over 30 test points.

Setup details. On the three highest-n benchmark datasets, Independent is strongly sparse ($s_{\text{eff}}/n = 0.16$, tail mass $\approx 10^{-3}$), while Correlated and Communities are moderately sparse ($s_{\text{eff}}/n \in [0.34, 0.43]$). We use the same protocol as in Section 5.1 with 50 trials per (m, method) pair.

F.4 Synthetic sparse game (Section 5.3)

Setup. We construct a sparse sigmoid game $v(S) = \sigma\big(\sum_{i \in S \cap R} a_i\big) - \sigma(0)$ on $n = 30$ players, where $R \subseteq [n]$ is drawn uniformly with $|R| = s$, and $a_i = \varepsilon_i u_i$ with $\varepsilon_i \in \{\pm 1\}$ uniform and $u_i \sim \text{Unif}(1.5, 3.0)$. We test Setting A ($s = 3$, $s/n = 10\%$) and Setting B ($s = 12$, $s/n = 40\%$), with budgets $m \in \{n, 2n, \ldots, 10n\}$ and 300 trials per (m, method) pair.

We use without-replacement paired sampling and omit `shap.KernelExplainer` since the game is a value oracle. Beyond ℓ_2 error, we report the *exact support recovery rate*, defined as the fraction of trials with $\{i : |\widehat{\phi}_i| > \varepsilon\} = R$ at threshold $\varepsilon = \beta_{\min}/3$, where $\beta_{\min} = \min_{i \in R} |\phi_i|$. The closed-form ground truth uses 2^s oracle calls per active player rather than 2^n, exploiting the Null Player axiom.

Choice of $n = 30$. We use $n = 30$ for clean numerical behavior: the importance-sampling weight ratio $1/[k(n-k)]$ is ≈ 8 at $n = 30$ but reaches $\approx 10^{14}$ at $n = 50$; the regime $m \ll 2^n$ holds comfortably.

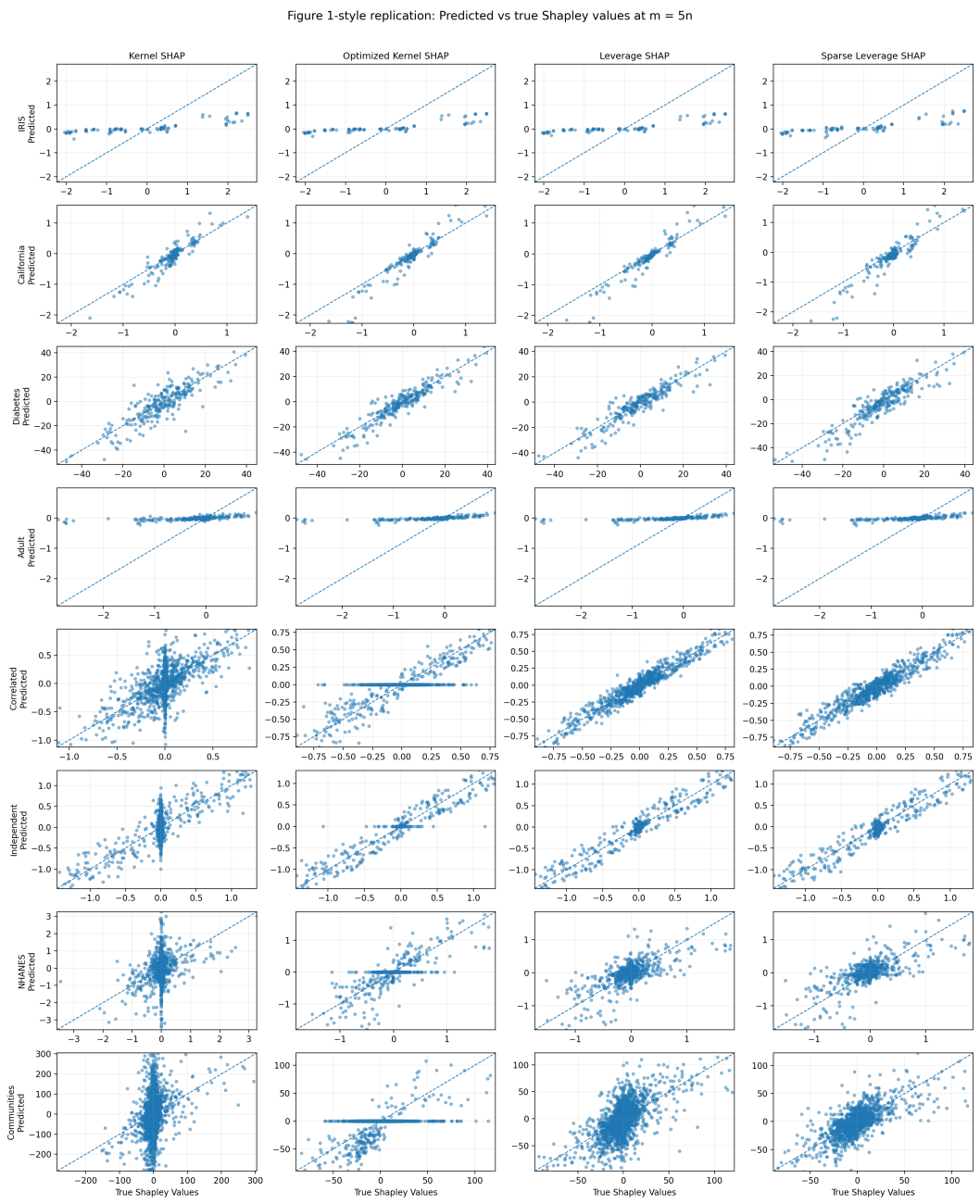


Figure 4: Predicted versus true Shapley values at $m = 5n$, on the eight real-data benchmarks of Musco and Witter [2025]. Columns are estimators; rows are datasets. Points concentrated along the identity line indicate accurate estimates.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

Justification: The abstract and the Introduction enumerate four contributions: (i) the Sparse-Leverage SHAP estimator with a sparse matrix concentration result (Lemma 2); (ii) a relative-error approximation guarantee using $\widetilde{O}(s^2/\varepsilon)$ value-function evaluations (Theorem 6); (iii) a bound on the restricted leverage mass of the Shapley design, $L_s = O(s \log n)$ (Proposition 3); and (iv) a computationally efficient IHT guarantee (Theorem 7). The guarantees include confidence amplification and are established in Sections 3 and 4 and their appendices. The single-run estimators are evaluated in the "Experiments" section.

Guidelines:

- The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: The "Limitations and Future Directions" paragraph of the Conclusion discusses four concrete limitations: the exact-sparsity assumption (versus the more realistic compressible/weak-ℓ_1 setting), the requirement of a known sparsity level s, the quadratic dependence on s in our upper bound, and the restriction to sparsity rather than the complementary hidden-symmetry setting. Additional practical limitations (e.g., weight-ratio ill-conditioning at large n) are noted in Appendix F.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The theoretical statements (including Lemma 2, Proposition 3, Proposition 1, Theorem 6, Proposition 9, Corollary 14, and Theorem 7) are numbered, cross-referenced, and state their assumptions explicitly. The sparse-influence model is formally defined in Section 2. Complete proofs, including the IHT guarantee, are provided in Appendices B, C, D, and E, with intuition provided in the main text where proofs are deferred.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Section 5 and Appendix F specify the datasets (eight from `shap.datasets` with feature counts $n = 4$ to $n = 101$), models (XGBoost with explicit tree counts and depths per dataset), the background-set size, sample budgets, number of trials (100, 50, or 300 depending on experiment), the ground-truth computation (Tree SHAP for real data; closed-form active-player reduction for synthetic), and the synthetic game specification. Algorithm 1 fully specifies the estimator.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: All datasets used are publicly available through the `shap.datasets` API, and Algorithm 1 together with the experimental protocol in Section 5 and Appendix F provides sufficient detail for re-implementation. Code is not released with the current submission to preserve anonymity, but will be made publicly available upon acceptance.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Appendix F specifies the XGBoost hyperparameters per dataset category (100 trees/depth 4 for smaller datasets; 50 trees/depth 3 for synthetic generators; 80 trees/depth 4 for NHANES and Communities), the background-set size, the sample-budget grid $m \in \{0.2n, 0.5n, n, 2n, 5n, 10n, 20n, 40n, 80n, 160n\}$, and the test-point counts (100, 50, 300 depending on experiment). The estimators are the weighted projected OLS of Leverage SHAP and the weighted projected s-sparse regression of Sparse Leverage SHAP (Algorithm 1).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All quantitative results (Figures 1, 2, 3) report medians with Q1–Q3 interquartile bands, computed over 100 test points (Section 5.1), 50 trials (Section 5.2), or 300 trials (Section 5.3). The variability captured is over random test points and random subset samples drawn by each estimator. Medians and quartiles are used rather than means/standard-deviations because the error distributions are heavy-tailed in the underdetermined regime $m < n$.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: The paper does not explicitly report compute resources. The experiments are lightweight by modern standards: they involve fitting small XGBoost models (≤ 100 trees) on tabular datasets with $n \leq 101$ features, and solving weighted least-squares / sparse regression problems with at most $160n \leq 16{,}160$ rows. All experiments run on a standard CPU laptop within a few hours; no GPU or cluster compute is required. We will add an explicit compute statement in the camera-ready version.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The research is a methodological contribution to Shapley value estimation. It uses only publicly available tabular benchmarks distributed with the SHAP library and introduces no new data collection, human-subjects research, or deployed systems. Anonymity has been preserved in the submission.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: The paper contributes an improved sampling algorithm for Shapley value estimation, a foundational tool in explainable AI. The contribution is methodological and does not introduce new models, data, or deployments. Any downstream impact depends entirely on the models whose predictions a user chooses to explain with Shapley values, and is not specific to the improved sampling scheme proposed here. If anything, more accurate and query-efficient Shapley estimators tend to have positive societal effects by making interpretability more accessible in high-stakes domains.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release any pre-trained models, generative systems, or scraped data. The only artifact is an estimation algorithm for Shapley values, which poses no such risk.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All baselines and datasets used are properly cited: Kernel SHAP [Lundberg and Lee, 2017], Tree SHAP [Lundberg et al., 2020], Leverage SHAP [Musco and Witter, 2025], and XGBoost [Chen and Guestrin, 2016]. The eight tabular datasets are obtained via the `shap.datasets` API distributed with the SHAP library [Lundberg and Lee, 2017, Lundberg et al., 2020]. These assets are used in accordance with the MIT License (SHAP, XGBoost), which permits research use with attribution.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper does not release any new datasets or pre-trained models. The only new contribution is the Sparse-Leverage SHAP algorithm, which is fully documented in Algorithm 1 and Section 3.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.

- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or any research with human subjects. All experiments are conducted on publicly available tabular benchmarks and on a synthetic cooperative game.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human subjects, so IRB approval is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLMs were used to assist with mathematical consistency checks, proof revision, and language editing. The experiments use classical numerical solvers and XGBoost.

    Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.