BHDR: BSGS-Hoisted Diagonal Regression for Non-Interactive Single-Server Kernel SHAP under CKKS

Bader Alissaei
VaultBytes Innovations Ltd
b@vaultbytes.com

Abstract

We present *BHDR* (BSGS-Hoisted Diagonal Regression), a non-interactive single-server construction that runs the server-side coalition evaluation and weighted-least-squares regression of Kernel SHAP under CKKS fully homomorphic encryption, for the deployed logistic-regression path. The sampling matrix Z, kernel weights $\boldsymbol{\pi}$, and precomputed regression matrix M are public and computed in plaintext at build time; the input, prediction, and attribution vector are never observed in plaintext on the server.

The construction packs $K = O(d \log d)$ coalitions into CKKS SIMD slots and executes the closing weighted-least-squares regression as a BSGS-hoisted diagonal matvec on a K'-periodic replicate encoding. At $d = 50$, $K = 390$, $K' = 512$ this cuts ciphertext rotations from $\sim$2200 to 51 ($\sim 43\times$), collapsing the regression step from 7.7 s to 0.6 s on Apple M1. End-to-end observed latency at the deployed configuration is p50 13.4 s, p95 16.3 s, p99 24.2 s over $N_q = 300$ UCI Adult queries, with 0/300 silent CKKS overflows (Wilson 95% CI $[0\%, 1.26\%]$) once an SDK-side encryption-boundary input guard is in place.

The deployed pipeline is governed by a deterministic regression-stability proposition: any implementation perturbation $\boldsymbol{\eta}$ in coalition-output space yields attribution error $\|\Delta\boldsymbol{\phi}\|_\infty \leq G_{\text{eff}} \cdot \|\boldsymbol{\eta}\|_2$, with $G_{\text{eff}} = \|P_\phi M P_y\|_{2\to\infty}$ realised at 0.287 on the deployed sampler. Combined with empirically certified finite-K sampling error this yields a per-release attribution-error budget tracked per (d, K) as a five-scalar release-time engineering certificate. A complementary i.i.d. Matrix Bernstein analysis of an antithetic-pair surrogate gives a literature-comparison reference for the sum-zero spectrum; it is not used to certify the deployed $K = 390$ configuration. A closed-form concentration theorem for the deployed stratified-ramp sampler is open work.

We additionally give a circuit-depth feasibility study for tree ensembles via an Oblivious Coalition Tree Evaluator (OCTE), deployed at $D = 4$, $T = 100$ ($\sim$ 53 s on Hetzner CX22). MLP support is not claimed.

This is a systems paper; cryptographic integrity against a malicious server and the efficiency-axiom-vs-ranking-stability trade-off introduced by Kernel SHAP's post-hoc redistribution are the scope of separate companion work.

Keywords: fully homomorphic encryption, Kernel SHAP, CKKS, SIMD packing, BSGS, diagonal matvec, privacy-preserving machine learning, Matrix Bernstein, feature attribution.

1 Introduction

High-stakes machine learning deployments are increasingly subject to two simultaneous classes of obligation: data-confidentiality during processing, and per-instance feature-level explanation of individual decisions. Computing Shapley additive explanations under fully homomorphic encryption

is the technical bottleneck between them. An organisation using FHE for encrypted inference cannot, with existing constructions, also produce per-instance SHAP attributions without decrypting the input; an organisation running plaintext SHAP exposes the raw input to the explanation server. This paper resolves that bottleneck for the deployed logistic-regression path.

Compliance motivation (brief). Data-protection regimes (EU GDPR; US HIPAA, CCPA/CPRA) require confidentiality of personal data during processing. Decisional explanation regimes (EU AI Act [9], in force for high-risk systems on 2 August 2026; US ECOA and FCRA) require feature-level explanation of individual decisions in employment, credit, insurance, education, and law-enforcement settings. Whether any specific configuration legally discharges any specific obligation is outside the scope of this paper; we use the compliance setting only to motivate the joint technical requirement, and the rest of the paper is a CKKS systems and approximation paper.

We conducted a systematic review of the literature. This included the Nguyen et al. 2024 survey of over 50 privacy-preserving explainability papers [10], the IACR ePrint archive, and proceedings of CCS, S&P, USENIX Security, NeurIPS, and ICML through early 2026. We are not aware of any published system that computes *Shapley-based* feature attribution under single-server FHE without decrypting the input data. We deliberately scope the novelty claim to Shapley attribution. Gradient-based attribution methods such as saliency maps and integrated gradients under FHE have appeared in prior work. The contribution here is the coalition-sampling and BSGS-regression construction that makes the game-theoretic variant feasible. XorSHAP [11] computes SHAP values under a cryptographic protocol, but it uses multi-party computation rather than single-server FHE.

Trivial baseline and contrast. A naive non-interactive baseline is: the server runs K encrypted model evaluations and ships back K encrypted predictions; the client decrypts and runs the weighted-least-squares Kernel SHAP regression in plaintext. This is functionally correct and preserves input privacy, but it has three operational problems. First, the client receives K encrypted scalars (or $\lceil K/n \rceil$ ciphertexts if packed, with $n = N/2$ the CKKS slot count), then must perform a $d \times K$ regression locally—feasible at $d = 50$, but it pushes work onto the explanation consumer. Second, decrypting the per-coalition predictions exposes a much richer transcript of K model evaluations $f(\mathbf{x}_S)$ to the client than the deployed pipeline, which returns only the final prediction and attribution vector. The expanded transcript is a larger output-disclosure surface even though the client already holds $\mathbf{x}$. Third, returning K ciphertexts inflates the wire-payload by a factor of K relative to a single attribution ciphertext. Our construction does the regression on the server side under the encryption envelope, returning a single attribution ciphertext; this is the engineering substance of the paper, and the latency, payload, and side-channel comparisons against the trivial baseline are all favourable at deployed $K = 390$.

Our contribution. We present the *BHDR* (BSGS-Hoisted Diagonal Regression) construction, a non-interactive single-server pipeline that performs the server-side coalition evaluation and weighted regression of Kernel SHAP under CKKS FHE for the deployed logistic-regression path (build-time public-design precomputation in plaintext; runtime: input, prediction, and attribution kept under encryption). The paper includes an end-to-end implementation on logistic regression over UCI Adult Income at $d = 50$, plus a circuit-depth feasibility study for tree ensembles at $D = 4$, $T = 100$ via an Oblivious Coalition Tree Evaluator (OCTE). The implemented pipeline provides *confidentiality* of the input, prediction, and explanation against an honest-but-curious server. Output privacy against a curious server and integrity against a malicious server are addressed in companion work,

which we cite where relevant but do not reproduce here; this paper is concerned with the systems and approximation questions that make the Kernel SHAP pipeline feasible under CKKS in the first place. The key technical contributions are:

1. The regression matrix $M = (Z^\top W Z)^{-1} Z^\top W$ depends only on *public* parameters, which enables offline plaintext pre-computation. The closing regression uses a *BSGS-hoisted diagonal transform*. This reduces ciphertext rotations from ∼2,200 to 51 for $d = 50$, $K' = 512$: a ∼43× reduction. The regression step drops from 7.7 s to ∼0.6 s on Apple M1. In an algorithm-level narrow-d ($d = 5$), warm-cache ablation at $K = 390$, the pipeline drops from 9.2 s to ∼2.1 s; the deployed $d = 50$ end-to-end observed latency on the same host is p50 13.4 s (Table 7). On a 4-vCPU AMD EPYC research server BHDR regression measures 5.3 s mean (p95 5.9 s). On CX22, the pre-BHDR baseline is *measured* at ∼95 s; the BHDR-integrated logistic-regression pipeline is *projected* at ∼45–50 s from the measured BHDR regression component scaled by the empirical ∼2× CX22/EPYC ratio, pending an end-to-end CX22 rerun (Table 9).
2. An *oblivious tree evaluation circuit* (OCTE) that evaluates all 2^D tree paths simultaneously. The deployed $D = 4$ variant uses a tanh–Chebyshev sign surrogate at ∼5 CKKS levels with path indicator products at $\lceil \log_2 D \rceil$ levels and plaintext leaf aggregation, measuring ∼53 s end-to-end on the Hetzner CX22 reference VM (single-threaded CPU OpenFHE). At $D = 6, T = 100, K = 390$ a v2 diagnostic uses a Lee composite-minimax sign gate at depth 23 and a path-product tree at $D - 1$ CT-CT levels. It measures 806 s on an 8-core Hetzner CPX42 instance and passes the SHAP accuracy gate at $L_\infty = 7.9 \times 10^{-3}$ with zero remaining CKKS levels (Section 7.2). The binding constraint at $D = 6$ is wall-clock latency and depth margin, not accuracy. A lower-depth CKK-iterated or Newton sign gate [19] is an in-tree experimental path.
3. A certified deterministic pipeline. Proposition 5 bounds the implementation-perturbation contribution to attribution error by $G_{\text{eff}} \cdot \|\boldsymbol{\eta}\|_2$, with $G_{\text{eff}} = \|P_\phi M P_y\|_{2\to\infty}$ computed from the public deployed (Z, W, M). Proposition 7 specifies the release-time engineering certificate as four levels containing five scalar certificate values (matrix-level: $\lambda^{\text{sz}}_{\min}$ and G_{eff}; empirical: max and p99 sampling error and CKKS FHE-vs-plaintext error), tracked per (d, K) configuration in the verification matrix and gated in CI. A complementary i.i.d. Matrix Bernstein analysis (Theorem 6) with closed-form $\lambda^{\text{sz}}_{\min}(\bar{A}) = 1/(2H_{d-1})$ is retained as an analytical baseline.
4. End-to-end encrypted benchmarks on UCI Adult Income logistic regression. The end-to-end observed wall-clock at $d = 50$ is the full-pipeline p50 13.4 s, p95 16.3 s, p99 24.2 s over $N_q = 300$ queries on Apple M1; an algorithm-level breakdown on a narrow-d ($d = 5$, $K = 390$) circuit isolates the BHDR rewrite at 2.1 s on the same host. We report 0 MB resident-memory growth and 0/300 silent CKKS overflows after the encryption-boundary input guard landed (pre-fix rate was 1/100; see Section 6.7).

Output privacy (IND-CPA$^{\text{D}}$ flooding) and integrity (Freivalds / LCV-IPA) are addressed in companion work and are therefore out of scope here. Similarly, the Kernel SHAP efficiency-axiom-vs-ranking-stability trade-off introduced by the default post-hoc proportional redistribution is a regulator-facing concern that applies to *any* Kernel SHAP deployment, encrypted or not; we treat it separately in a dedicated companion paper and do not rely on it here beyond noting that our experimental pipeline supports both redistribution-on and redistribution-off modes.

Security claim (scope). This paper claims confidentiality of the client input $\mathbf{x}$, the prediction $\hat{y}$, and the attribution vector $\hat{\boldsymbol{\phi}}$ against a single *honest-but-curious* server, under the CKKS parameterisation specified in Table 1 and the IND-CPA security of the underlying scheme. The paper does *not* claim: malicious-server correctness or verifiable computation; output-privacy beyond the encrypted-output setting (IND-CPA$^{\mathrm{D}}$ noise flooding); model-extraction resistance against repeated SHAP queries; membership-inference resistance via $\boldsymbol{\phi}$; or side-channel resistance against a co-tenant or physical adversary. These are addressed in separate companion work (Section 3.6, Appendix B, Appendix C).

2 Background

2.1 Fully Homomorphic Encryption

A fully homomorphic encryption scheme allows arbitrary computation on ciphertexts without decryption. We work primarily with the CKKS scheme [4], which supports approximate arithmetic on fixed-point real numbers. A CKKS ciphertext with ring dimension N encodes a vector of $N/2$ complex slots; using only the real subfield gives $N/2$ real slots (packing two reals into each complex slot is possible but is not exploited here). Supported operations include:

- **SIMD addition/subtraction:** slot-wise, no depth consumed.
- **SIMD multiplication:** slot-wise, consumes one level of multiplication depth and introduces noise.
- **Rotation:** cyclic permutation of slots, no depth consumed.
- **Rescaling:** reduces ciphertext modulus to manage precision after multiplication.
- **Plaintext-ciphertext multiplication:** multiplies each slot by a known plaintext vector. Under the rescale-level convention used by OpenFHE and adopted throughout this paper, this nominally consumes one level; in practice an isolated plaintext-coefficient inner product is fused with the next multiplicative step (e.g. the first level of a polynomial sigmoid evaluator) at scale-management time and therefore consumes no *additional* level beyond what the downstream operation already requires (Section 7.1, "$L_f = 8$ accounting").

The scheme is parameterized by a *multiplication depth budget* L. Once exhausted, further multiplications push noise past the decryption threshold. Bootstrapping refreshes the noise budget at significant computational cost.

Our construction is also applicable to BGV [5] and BFV [6] schemes, which use modulus switching in place of rescaling.

Deployed CKKS parameter set. Table 1 fixes the concrete CKKS parameter set used throughout this paper. The logistic-regression deployment runs at $N = 2^{15}$ with multiplicative depth $L = 10$ at 128-bit classical security under `HEStd_128_classic`, single-server CPU OpenFHE, no bootstrapping. The OCTE feasibility study is separately provisioned at $N = 2^{16}$ to host the deeper sign-gate modulus chain ($L = 25$ at $D = 4$; $L = 31$ at $D = 6$). All benchmarks, security claims, and rotation-key counts in this paper refer to the parameter set in Table 1.

Table 1: Concrete CKKS parameter set for the BHDR pipeline. The logistic-regression deployment is the headline configuration; the OCTE columns describe the tree-ensemble feasibility studies of Section 7.2. Security category follows the `HEStd_128_classic` table from the Homomorphic Encryption Standard. Rotation-key counts are stated per the BSGS-hoisted diagonal layout of Section 3.4.

Parameter	LR (d=50)	OCTE D=4	OCTE D=6 (v2)
Scheme	CKKS	CKKS	CKKS
Implementation	OpenFHE	OpenFHE	OpenFHE
Ring dimension N	2^{15}	2^{16}	2^{16}
Slot count $n = N/2$	16,384	32,768	32,768
Multiplicative depth L	10	∼25–30	31
Coalition budget K	390	390	390
Replicate period K'	512	—	—
Security category	`HEStd_128_classic`	`HEStd_128_classic`	`HEStd_128_classic`
Classical security	≥ 128 bits	≥ 128 bits	≥ 128 bits
Bootstrapping	none (leveled)	none (leveled)	none (leveled)
BHDR rotation keys	51	—	—
Hardware backend	CPU (single-server)	CPU	CPU
Reference host	Apple M1, Hetzner CX22	Hetzner CX22	Hetzner CPX42

2.2 Shapley Additive Explanations (SHAP)

Shapley additive explanations [1] are the dominant methodology for per-instance feature attribution in machine learning. For a model f and input $\mathbf{x} \in \mathbb{R}^d$, the exact SHAP value for feature i is:

$$\phi_i(f, \mathbf{x}) = \sum_{S \subseteq D \setminus \{i\}} \frac{|S|!\,(d - |S| - 1)!}{d!} \big[f(\mathbf{x}_{S \cup \{i\}}) - f(\mathbf{x}_S) \big] \tag{1}$$

where $D = \{1, \ldots, d\}$ and $\mathbf{x}_S$ denotes the input with features outside S replaced by baseline values. Equation (1) requires evaluating f on all 2^d coalitions, which is infeasible even in plaintext for typical $d \geq 15$.

2.3 Kernel SHAP

Lundberg and Lee [1] proposed Kernel SHAP, which samples K coalitions and solves a weighted least-squares (WLS) regression:

$$\hat{\boldsymbol{\phi}} = \operatorname*{argmin}_{\boldsymbol{\phi}'} \sum_{k=1}^{K} \pi_k \Big(f(\mathbf{x}_{S_k}) - \phi_0 - \sum_{i=1}^{d} \phi'_i z_{k,i} \Big)^2 \tag{2}$$

where $\mathbf{z}_k \in \{0,1\}^d$ is the binary indicator vector for coalition k and π_k is the Shapley kernel weight. The closed-form solution is:

$$\hat{\boldsymbol{\phi}} = (Z^\top W Z)^{-1} Z^\top W \mathbf{y} = M \mathbf{y} \tag{3}$$

where $Z \in \{0,1\}^{K \times d}$ is the coalition sampling matrix, $W = \operatorname{diag}(\pi_1, \ldots, \pi_K)$, and $\mathbf{y} = (f(\mathbf{x}_{S_1}), \ldots, f(\mathbf{x}_{S_K}))^\top$ is the vector of coalition prediction outputs.

Intercept and centering. The closed form $\hat{\boldsymbol{\phi}} = (Z^\top W Z)^{-1} Z^\top W \mathbf{y}$ in Equation (3) is the *baseline-centered* solution: $\phi_0 = f(\mathbf{b})$ is fixed by the baseline and $\mathbf{y}$ denotes the coalition-output vector after subtracting ϕ_0 component-wise. The full unconstrained WLS form would use the augmented design

$\tilde{Z} = [\mathbf{1},\, Z] \in \{0,1\}^{K\times(d+1)}$ that estimates the intercept jointly with $\boldsymbol{\phi}$; in the deployed pipeline we fix ϕ_0 at build time and run the centered $K \times d$ regression so that $M = (Z^\top W Z)^{-1} Z^\top W$ is the public, build-time-precomputed operator that closes the encrypted pipeline.

Covert and Lee [2] analyzed convergence and proposed paired and antithetic sampling strategies. Musco and Witter [3] later established that $O(d \log d)$ model evaluations suffice for approximate Shapley values with provable non-asymptotic guarantees. Both works operate entirely in plaintext.

2.4 Why Kernel SHAP Under FHE Is Non-Trivial

Running Kernel SHAP under FHE faces three compounding challenges:

1. **Feature masking consumes depth.** Replacing a feature with a baseline value requires homomorphic element-wise multiplication of the encrypted input vector with a binary mask, consuming one multiplication depth level under CKKS.
2. K **sequential evaluations exhaust the depth budget.** Each model evaluation consumes the full multiplication depth of the model's evaluation circuit. K sequential evaluations would require K times the depth budget or many costly bootstrapping operations.
3. **Matrix inversion is infeasible on ciphertexts.** The regression $\hat{\boldsymbol{\phi}} = (Z^\top W Z)^{-1} Z^\top W \mathbf{y}$ requires a matrix inversion. Gaussian elimination, QR decomposition, and iterative methods such as conjugate gradients or Newton–Schulz are prohibitively expensive or infeasible under FHE. Data-dependent pivoting, division, and unbounded noise accumulation all block a direct port.

A 2024 survey of more than 50 privacy-preserving explainability papers [10] confirms that no published system addresses all three challenges simultaneously.

3 Our Construction

We refer to the reference implementation of this construction as *CipherExplain* and to the construction itself as the *BHDR pipeline*; all benchmarks, experiments, and source-code references in this paper point to the CipherExplain reference implementation release. The system comprises five interconnected components, depicted in Figure 1.

3.1 Sampling Matrix Generator (Component 100)

The Sampling Matrix Generator produces three public artifacts, none of which depends on any encrypted input.

1. A coalition sampling matrix $Z \in \{0,1\}^{K\times d}$, where each row represents one coalition as a binary indicator vector.
2. A Shapley kernel weight vector $\boldsymbol{\pi} \in \mathbb{R}^K$, where π_k is the Lundberg–Lee kernel weight for coalition k.
3. A regression matrix $M \in \mathbb{R}^{d\times K}$, pre-computed as $M = (Z^\top W Z)^{-1} Z^\top W$ where $W = \mathrm{diag}(\boldsymbol{\pi})$.

Sampling design. The coalition sampling matrix Z is generated offline, once per feature dimensionality d, using a structured quasi-random design with antithetic complement pairing:

1. *Target coalition count:* $K = \max\big(\lfloor c \cdot d \cdot \ln d \rfloor,\, 2d\big)$, rounded down to the nearest even integer so antithetic complement pairing is exact. The $2d$ lower bound gives a row budget large enough for minimal per-feature inclusion/exclusion coverage in principle; under the seeded stratified ramp realised here, per-feature coverage is seed-dependent and is verified by the release-time

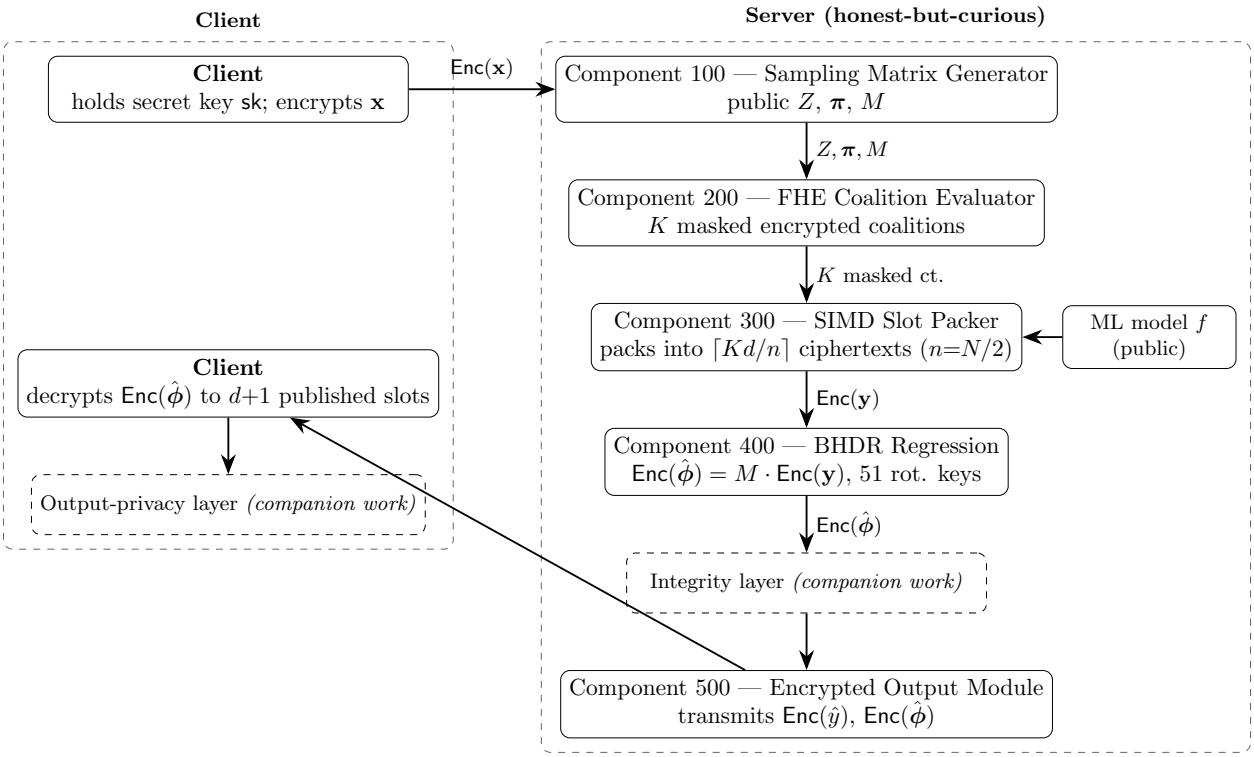

Figure 1: System architecture of the BHDR pipeline, shown as client/server swim lanes. The five core server-side components (Sampling Matrix Generator, FHE Coalition Evaluator, SIMD Slot Packer, BHDR Regression, Encrypted Output Module) run under an honest-but-curious threat model; the FHE secret key never leaves the client lane. Before encryption, the SDK applies an *Encryption-Boundary Input Guard* (clip + SHA-256 commit + budget check) on the client; the guard is not a separate server-side component but a preprocessing stage that hardens the $\mathsf{Enc}(\mathbf{x})$ delivered to Component 100 against silent CKKS overflow at the regression step (Section 6.7). An integrity layer (dashed border) and a client-side output-privacy layer (dashed border) are shown as companion-work hooks but are not analysed here. Dashed outer rectangles indicate the client and server zones respectively.

Table 2: Coalition count K by feature dimensionality d. The quasi-random rows all use the default precision parameter $c = 2$ so the target count is $2\,d\,\ln d$, rounded down to the nearest even integer to accommodate the antithetic-complement pairing. The $d = 16$ row is $K = 88$ rather than the continuous target 88.72; $d = 50$ is $K = 390$ rather than 391.2; etc.

d	Regime	K
5	Exact ($2^d - 2$)	30
16	Quasi-random	88
20	Quasi-random	118
30	Quasi-random	204
50	Quasi-random	390
100	Quasi-random	920

engineering certificate's matrix-health levels (Proposition 7, level 1). Deterministic singleton/co-singleton coverage requires the boundary-complete sampler variant discussed in Appendix D, not the deployed ramp. This row budget is the combinatorial precondition $K/(2d) \geq 1$ used in Theorem 6. Without the floor, for small d the $c\,d\,\ln d$ term can drop below $2d$ and leave a feature under-sampled on one side of the inclusion binary.

2. *Primary half (size ramp):* for $k = 0, 1, \ldots, K/2 - 1$, the k-th coalition has size $s_k = \text{clip}(\lfloor k\,d/(K/2) \rfloor + 1,\ 1,\ d - 1)$ (a linear ramp from size 1 at index 0 up to size $d - 1$ at index $K/2 - 1$). Given size s_k, the included feature subset is drawn uniformly at random without replacement from $\{1, \ldots, d\}$ using a seeded `numpy.default_rng(42)` generator, yielding determinism per (d, c).
3. *Antithetic half:* each $\mathbf{z}_k$ in the primary half is paired with its bitwise complement $\bar{\mathbf{z}}_k = \mathbf{1} - \mathbf{z}_k$ (size $d - s_k$). The primary and antithetic halves are stacked to form the final $K \times d$ matrix Z.
4. *Kernel weights:* $\pi_k = w(|S_k|)$ with the Lundberg–Lee kernel $w(s) = (d - 1)/(\binom{d}{s}\, s\,(d - s))$, normalised such that $\sum_k \pi_k = 1$.

The stratified ramp covers the full size spectrum $\{1, \ldots, d - 1\}$ uniformly. Antithetic pairing biases towards the Lundberg–Lee kernel's high-weight boundary sizes by exactly replicating size-s and size-$(d - s)$ coalitions.

For small feature dimensionalities where $d \log d$ is comparable to 2^d, we instead enumerate all $2^d - 2$ non-trivial coalitions exactly. This yields exact Shapley values within FHE approximation tolerance.

Remark 1. *None of Z, $\boldsymbol{\pi}$, or M depends on the encrypted input. They depend only on the dimensionality d, a system parameter. They are therefore computed once per dimensionality and reused across all subsequent encrypted queries. The matrix inversion required to form M runs in plaintext during this offline pre-computation, eliminating the otherwise prohibitive requirement of inverting an encrypted matrix.*

3.2 FHE Coalition Evaluator (Component 200)

The FHE Coalition Evaluator receives the encrypted input vector $\mathsf{Enc}(\mathbf{x})$ and produces K homomorphically masked encrypted coalition inputs. For each row $\mathbf{z}_k$ of the public sampling matrix Z, the Evaluator constructs a masked input ciphertext via:

$$\mathsf{Enc}(\mathbf{x}_{S_k}) = \mathbf{z}_k \odot \mathsf{Enc}(\mathbf{x}) \oplus (\mathbf{1} - \mathbf{z}_k) \odot \mathsf{Enc}(\mathbf{b}) \tag{4}$$

where $\odot$ denotes element-wise homomorphic multiplication of an encrypted vector with a public binary indicator vector, $\oplus$ denotes homomorphic addition, and $\mathsf{Enc}(\mathbf{b})$ is the encrypted baseline (per-feature means of a public reference dataset).

The homomorphic combination consumes *exactly one level* of multiplication depth and preserves the IND-CPA security of the underlying FHE scheme.

Personalized encrypted baselines. Equation (4) supports client-supplied baselines without modification. The client provides $\mathsf{Enc}(\mathbf{b})$ alongside $\mathsf{Enc}(\mathbf{x})$, and the server substitutes $\mathsf{Enc}(\mathbf{b})$ for the default server-held baseline. This matters in regulated settings where the choice of baseline is legally consequential. Under ECOA, using the dataset mean as baseline for a protected attribute such as age produces different attributions than a counterfactual baseline set to the population median. A client-supplied $\mathsf{Enc}(\mathbf{b})$ enables interventional counterfactual explanations *without revealing the counterfactual scenario to the server*. The security analysis extends directly: $\mathsf{Enc}(\mathbf{b})$ is an additional IND-CPA-secure ciphertext that the server processes but cannot decrypt, and the data-oblivious circuit structure is unchanged.

3.3 SIMD Slot Packer (Component 300)

The SIMD Slot Packer reduces the per-coalition cost of model evaluation by packing multiple masked coalition inputs into the SIMD slots of a single CKKS ciphertext.

Definition 1 (Packing strategy)**.** *Throughout this paper we use N for the CKKS ring dimension and $n = N/2$ for the number of available SIMD slots per ciphertext. For feature dimensionality d, each coalition occupies d contiguous slots; the packer places $\lfloor n/d \rfloor$ coalitions into a single ciphertext, and the total number of packed ciphertexts required to cover all K coalitions is $\lceil Kd/n \rceil$.*

Proposition 1. *For typical parameters $d = 50$, $K = 390$, and CKKS ring dimension $N = 2^{15}$ (slot count $n = 2^{14} = 16384$), the total number of packed ciphertexts is $\lceil 390 \times 50/16384 \rceil = 2$.*

The machine learning model is therefore evaluated under FHE only $\lceil Kd/n \rceil$ times, once per packed ciphertext. Each such evaluation runs at the model's native multiplication depth and consumes no additional depth beyond what standard encrypted inference would require.

Gathering. After model evaluation, coalition outputs are spread across $\lceil Kd/n \rceil$ packed ciphertexts. We consolidate them into a single $\mathsf{Enc}(\mathbf{y})$ via plaintext masking of the prediction-output slot in each coalition block, followed by rotation-based accumulation. The cost is $O(\lceil Kd/n \rceil)$ rotations and masks, negligible next to the BHDR regression step. BHDR's diagonal preprocessing absorbs the stride-d slot layout by adjusting the plaintext diagonal indices, so no extra compaction rotations are needed.

3.4 Homomorphic Regression Engine (Component 400)

The Homomorphic Regression Engine computes $\mathsf{Enc}(\hat{\boldsymbol{\phi}}) = M \cdot \mathsf{Enc}(\mathbf{y})$ where $M \in \mathbb{R}^{d \times K}$ is the public regression matrix and $\mathsf{Enc}(\mathbf{y})$ is the encrypted coalition output vector. We implement this via the *BSGS-hoisted diagonal transform* (BHDR), which reduces the rotation count from the pre-BHDR baseline of ∼2,200 measured rotations down to 51 at $d = 50$, $K' = 512$, a ∼43× reduction. (The classical diagonal Halevi–Shoup matvec [14] alone yields $O(K' + d \log K') = O(K')$ rotations; our additional savings come from the asymmetric BSGS split combined with K'-periodic replicate encoding, detailed below.)

Diagonal encoding. The matrix M is embedded in the $n \times n$ slot space ($n = N/2$) and decomposed into K cyclic diagonals following Halevi and Shoup [14]. The BSGS decomposition partitions the diagonals into r_1 baby steps and r_2 giant steps with $r_1 \cdot r_2 \geq K$.

Remark 2 ($K \mid n$ requirement)**.** *The BSGS grid operates in n-slot space, so correctness requires $K \mid n$ (equivalently, $(l - jr_1) \bmod n \bmod K = (l - jr_1) \bmod K$). For $K = 390$, $n = 16384$, we have $\gcd(n, K) = 2$ and $n \bmod K = 4$, producing an off-by-4 corruption across the wrap boundary. We therefore pad K up to the next power of two that divides n: $K' = 512$. M is padded with $K' - K = 122$ zero columns; the corresponding $K' - K$ plaintext–ciphertext multiplies collapse to zero and add no depth.*

Remark 3 (K-periodic replicate encoding)**.** *A contiguous zero-padded encoding of* $\mathsf{Enc}(\mathbf{y})$ *produces the correct BSGS output only at slot* 0*: BSGS rotations sample $\mathbf{y}_{pad}[(l+k) \bmod n]$ for $k \in [0, K'-1]$ and therefore miss the wrap-around diagonals $k \in [n-l, n-1]$ needed for $l \geq 1$. We apply K'-*periodic replicate encoding*: tile* $\mathsf{Enc}(\mathbf{y})$ *with period K' so that $\mathbf{y}_{pad}[l] = \mathbf{y}[l \bmod K']$ for all $l \in [0, n-1]$. Because $K' \mid n$, this can be realized by $\log_2(n/K') = 5$ rotate-and-double operations on the encrypted input (one-time, per query). The plaintext diagonals are tiled with the same period, so the BSGS grid sums the K'-cyclic matrix-vector product correctly at every output slot.*

Asymmetric near-optimal split. Baby-step rotations use `EvalFastRotationExt` with level-2 hoisting and no per-rotation `KeySwitchDown`. Giant-step rotations use standard `EvalRotate`, which is $\sim$2.7$\times$ more expensive per call. For $K' = 512$ the deployed split $r_1 = 32$, $r_2 = 16$ is near-optimal under the weighted cost $(r_1-1)\,c_1 + (r_2-1)\,c_2$ subject to $r_1 r_2 \geq K'$ and aligns with the OpenFHE rotation-key grid; the unconstrained continuous optimum sits closer to $r_1 \approx 37$, $r_2 \approx 14$, but $(32, 16)$ is the implementation-admissible split used by the deployed system (powers of two for the rotation-key indexing). This yields 31 baby-step + 15 giant-step + 5 replication doublings = **51 total rotations**. We compare against two baselines. Textbook row-wise matvec over d output coordinates at $K = 390$ takes $d(K-1) = 19{,}450$ rotations, so BHDR is a $\sim$381$\times$ reduction. A partially-optimised row-wise BSGS baseline (the legacy Python/OpenFHE binding path, which already batches rotations per row) consumes $\sim$2,200 rotations, making BHDR a $\sim$43$\times$ reduction. Both comparisons are faithful. We report 43$\times$ as the headline because it measures the speedup against code the reader is most likely to reimplement.

Algorithm.

1. *K'-periodic replication:* expand $\mathsf{Enc}(\mathbf{y})$ into an n-slot tile of period K' via $\lceil \log_2(n/K') \rceil = 5$ rotate-and-double operations (Remark 3).
2. *Baby-step precompute (hoisted): digits* $\leftarrow$ `EvalFastRotationPrecompute`$(\mathsf{Enc}(\mathbf{y}))$.
3. *Baby-step rotations: ct_ext*$[i] \leftarrow$ `EvalFastRotationExt`$(\mathsf{Enc}(\mathbf{y}), i,$ *digits*$)$ for $i = 0, \ldots, r_1-1$. All rotations share one digit decomposition (level-1 hoisting); results remain in the extended CRT basis (level-2 hoisting (`KeySwitchDown` deferred)).
4. *Giant-step accumulation:* For $j = 0, \ldots, r_2-1$, accumulate $acc_j = \sum_i p'_{j,i} \odot ct\_ext[i]$ (plaintext-ciphertext multiply in extended basis), then $acc[j] \leftarrow$ `KeySwitchDown`(acc_j).
5. *Giant-step rotation and sum: result* $= \sum_j$ `EvalRotate`$(acc[j],\ j \cdot r_1)$.
6. *Rescale* (consumes 1 depth level).

The pre-rotated plaintext diagonals $p'_{j,i}$ are computed offline (they depend only on M and the BSGS split, not on any encrypted data).

Proposition 2. *The BHDR algorithm consumes exactly one level of multiplication depth (one rescale after the accumulation). Rotations and additions do not consume depth in CKKS. The total*

additional depth introduced by the regression engine is therefore exactly 1, identical to the unhoisted approach.

Rotation key requirements. BHDR requires rotation keys for offsets $\{1, 2, \ldots, r_1-1\} \cup \{r_1, 2r_1, \ldots, (r_2-1) \cdot r_1\} \cup \{K', 2K', 4K', 8K', 16K'\} = 31 + 15 + 5 = 51$ keys (31 baby-step, 15 giant-step, 5 replication doublings), versus $\sim K$ in the unhoisted approach. This reduces client-side key generation and transmission by $\sim 10\times$.

Centering. Coalition prediction outputs are centered before the matrix-vector multiplication by subtracting the base-rate prediction $f(\mathbf{b})$. The reference implementation pre-computes $f(\mathbf{b})$ in plaintext at model registration, since the baseline $\mathbf{b}$ is a public reference dataset mean rather than a client secret. A fully encrypted deployment may instead evaluate $f(\mathsf{Enc}(\mathbf{b}))$ homomorphically, at the cost of one additional FHE model evaluation. The centered outputs $\tilde{y}_k = f(\mathbf{x}_{S_k}) - f(\mathbf{b})$ feed the regression engine. After client-side decryption, any residual efficiency-axiom error is redistributed proportionally to the absolute magnitudes of the attribution values:

$$\phi_i \leftarrow \phi_i + r \cdot \frac{|\phi_i|}{\sum_{j=1}^{d} |\phi_j|} \tag{5}$$

where $r = \hat{y} - \phi_0 - \sum_j \phi_j$ is the residual. The implementation guards the degenerate case $\sum_j |\phi_j| = 0$ by skipping redistribution and emitting a diagnostic; this case is not observed on the deployed validation distributions but is explicitly handled to avoid division-by-zero. Without centering and redistribution, the unconstrained WLS regression produces a mean efficiency axiom residual near 0.35 (max 0.41) in prototype benchmarks on the UCI Adult Income dataset at $d = 5$, $K = 30$. Centering drops the pre-redistribution residual to a mean near 0.018 (max 0.053). After redistribution, the residual reaches floating-point machine epsilon (1.1×10^{-16} across 50 test instances).

When redistribution is safe. Redistribution adds a *signed* perturbation $\delta_i = r \cdot |\phi_i| / \sum_\ell |\phi_\ell|$ to each ϕ_i. Because the perturbation is signed, the safe sufficient condition for preserving full pairwise ranking has to control *pairwise* swings, not the per-coordinate magnitude alone. Sort the attributions $|\phi_{(1)}| \geq |\phi_{(2)}| \geq \ldots \geq |\phi_{(d)}|$ and let $\Delta_k = |\phi_{(k)}| - |\phi_{(k+1)}|$ be the adjacent-rank gap. A conservative sufficient condition for full-ranking preservation is

$$\min_k \Delta_k \;>\; 2\,|r| \cdot \frac{|\phi_{(\max)}|}{\sum_j |\phi_j|},$$

which bounds the worst-case pairwise swing $|\delta_i - \delta_j| \leq 2\,|r|\,|\phi_{(\max)}| / \sum_\ell |\phi_\ell|$ by the smallest gap. This is conservative; sharper data-dependent conditions are possible but require the realised $\{\phi_i\}$. In the deployed regime $|r| \leq 0.053$ and $\sum_\ell |\phi_\ell| \geq 0.5$ for non-degenerate predictions, an informal scale check gives $|r| / \sum_\ell |\phi_\ell| \lesssim 0.1$ (i.e. the perturbation magnitude is small relative to the attribution mass; this is a sanity check, *not* a ranking guarantee). The full-ranking sufficient condition above holds whenever $\min_k \Delta_k \gtrsim 0.21 \cdot |\phi_{(\max)}| / \sum_\ell |\phi_\ell|$ at the deployed $|r| \leq 0.053$. For borderline decisions where two adjacent-rank features are within $|r|$ of each other, redistribution can swap that pair's rank without swapping the top-two pair's rank. The CipherExplain API exposes both the raw $\boldsymbol{\phi}$ and the redistributed version. Regulated deployments that care about full-rank stability under borderline cases should disable redistribution and report the raw residual directly (Section 1, redistribution caveat).

Table 3: Computational overhead beyond standard encrypted inference.

Component	Ciphertext Ops	Rotations	Added Depth
Coalition masking (200)	K elem. mults	0	+1 level
Packed model eval. (300)	$\lceil Kd/n \rceil$ evals	model-dep.	+0 levels
Gathering (300→400)	masks + rotations	$O(\lceil Kd/n \rceil)$	+0 levels
BHDR regression (400)	K' pt-ct mults	$(r_1-1)+(r_2-1)+\log_2(n/K')=51$	+1 level
Total	$\lceil Kd/n \rceil + 1$ evals	$O(\sqrt{K})$	+2 levels

3.5 Encrypted Output Module (Component 500)

The Encrypted Output Module packages the encrypted prediction $\mathsf{Enc}(\hat{y})$ and the encrypted feature attribution vector $\mathsf{Enc}(\hat{\boldsymbol{\phi}})$ and transmits them to the client. The client alone holds the secret key sk and is the only party capable of decrypting either the prediction or the explanation. The computation server never observes the input, the prediction, or the explanation in plaintext.

3.6 Operational Threat Model

This paper analyses the BHDR pipeline under a single *honest-but-curious server* threat model: the server faithfully runs the protocol on the encrypted inputs but may attempt to learn the plaintext input, prediction, or attribution from anything it sees. Confidentiality of the encrypted outputs in this regime reduces to the IND-CPA security of the underlying CKKS scheme. Stronger distributional output privacy via IND-CPA$^{\mathrm{D}}$ noise flooding, calibrated along the lines of Li, Micciancio, Schultz and Sorrell [15] and Ogilvie [20], is deferred to companion work.

Integrity against a malicious server is *out of scope* of the present construction; a server that deviates from the protocol may return an arbitrary $(\mathsf{Enc}(\hat{\boldsymbol{\phi}}), \mathsf{Enc}(\hat{y}))$ that the client cannot distinguish from a correct execution. Appendix B reports a deployable client-side verification hook with measured operating points; the full malicious-server treatment (binding, knowledge-soundness, Freivalds-style consistency, sentinel audit coalitions, biased coalition sampling) is the subject of separate companion work and is not relied on here. Model extraction via repeated SHAP queries and membership inference via $\boldsymbol{\phi}$ are likewise deferred. The headline performance and approximation claims of the present paper do not depend on the integrity hook.

A separately-implemented *operational* guard, deployed at the explain-API boundary on the client SDK, clips inputs to $\|\mathbf{x}\|_\infty \le R_{\text{in}}$ and pre-checks the $\log_2(K \cdot R_{\text{in}}^2 \cdot \|M\|_\infty) \le 48$ budget before encryption (Section 6.7). This is a correctness defence against the silent-overflow failure mode, not an integrity defence.

4 Complexity Analysis

Table 3 summarizes the ciphertext-level operations and multiplication depth CipherExplain adds on top of standard encrypted inference.

Theorem 3 (Total overhead)**.** *For a machine learning model with native multiplication depth L_f, the CipherExplain pipeline requires a total multiplication depth of at most $L_f + 2$ and a total of $\lceil Kd/n \rceil + 1$ ciphertext-level model evaluations (including one evaluation of the baseline for centering), where $n = N/2$ is the slot count and $K = O(d \log d)$.*

Proof. The coalition masking step (Equation 4) consumes one multiplication level. The packed model evaluations consume L_f levels (the model's native depth) but add no *additional* depth since they execute the same circuit as standard inference. The regression engine (Proposition 2) adds one level. Total: $1 + L_f + 1 = L_f + 2$. The number of ciphertext-level evaluations follows from Proposition 1. □

5 Approximation Guarantees

The deployed BHDR pipeline is governed by a build-time certified design: a fixed public sampling matrix Z, a fixed kernel weight matrix W, and a pre-computed regression matrix $M = (Z^\top W Z)^{-1} Z^\top W$. None of these is a random variable at query time. We therefore organise the approximation analysis around two complementary theorems and a release-time engineering certificate. Proposition 5 (*certified deterministic regression stability*) is the headline guarantee for the deployed system. Theorem 6 (*conservative i.i.d. antithetic-pair bound*) is an analytical baseline that bounds the sampling component of the upstream perturbation under an i.i.d. Lundberg–Lee surrogate of the deployed sampler; it is retained because it explains the role of the sum-zero spectrum and provides a known reference point against the Hoeffding union bound. Proposition 7 (*deterministic-ramp release-time engineering certificate*) specifies the release-time numerical certificates that license Proposition 5 for each deployed dimensionality. A quasi-Monte-Carlo or stratified-design concentration argument that would replace Proposition 7 with a closed-form theorem matching the deterministic ramp is left as open work.

Sampling model for Theorem 6. Theorem 6 below is for an *importance-sampled i.i.d. surrogate*: coalition pairs are drawn from the normalised Lundberg–Lee size distribution $\tilde{w}_s = d/(2H_{d-1} \cdot s(d-s))$, and the per-pair contribution to the information matrix is the antithetic average $A_j = \frac{1}{2}\big(\mathbf{z}_j \mathbf{z}_j^\top + \bar{\mathbf{z}}_j \bar{\mathbf{z}}_j^\top\big)$. The factor $\frac{1}{2}$ averages the two complementary rows in the pair, so the diagonal of $\bar{A} = \mathbb{E}[A_j]$ is $\Pr(z_i = 1) = 1/2$ rather than 1. This surrogate captures the same sum-zero information geometry as the weighted deterministic design used in production but is not the exact $Z^\top W Z$ matrix the deployed sampler builds. The production weighted design is covered by Proposition 7 as an empirically certified guarantee. Whenever we write $H = \sum_j A_j$ in Theorem 6 we mean this surrogate sum, not the $Z^\top W Z$ of the deployed weighted regression.

5.1 Closed form for the projected information-matrix eigenvalue

Proposition 4 (Closed form for the projected information-matrix eigenvalue)**.** *Let $\bar{A} = \mathbb{E}[A_j]$ be the expected pair-level information matrix of Step 1 below, and let $\lambda_{\min}^{sz}(\bar{A})$ denote its smallest eigenvalue on the sum-zero subspace $\{\mathbf{v} \in \mathbb{R}^d : \mathbf{1}^\top \mathbf{v} = 0\}$. Under the i.i.d. Lundberg–Lee antithetic-pair sampling model with normalised size distribution $\tilde{w}_s = d/(2H_{d-1} \cdot s(d-s))$,*

$$P_0 \bar{A} P_0 \;=\; \frac{1}{2H_{d-1}}\, P_0, \quad \textit{i.e.,} \quad \lambda_{\min}^{sz}(\bar{A}) \;=\; \frac{1}{2H_{d-1}} \;\approx\; \frac{1}{2\ln d},$$

where $H_{d-1} = \sum_{k=1}^{d-1} 1/k$ is the $(d-1)$st harmonic number.

Proof. By symmetry of $\tilde{w}_s$ under $s \leftrightarrow d-s$, every diagonal entry equals $\bar{A}_{ii} = \Pr[z_i = 1] = 1/2 =: c_0$. For $i \neq j$,

$$\beta_d \;=\; \Pr[z_i = z_j = 1] \;=\; \sum_{s=1}^{d-1} \tilde{w}_s \cdot \frac{s(s-1)}{d(d-1)} \;=\; \frac{1}{2H_{d-1}(d-1)} \sum_{s=1}^{d-1} \frac{s-1}{d-s}.$$

Substituting $t = d - s$ rewrites the inner sum as $\sum_{t=1}^{d-1}(d-t-1)/t = (d-1)H_{d-1} - (d-1)$, so $\beta_d = \frac{1}{2}(1 - 1/H_{d-1})$. Therefore $\bar{A} = c_0 I + \beta_d(\mathbf{1}\mathbf{1}^\top - I)$, and projecting onto the sum-zero subspace gives $P_0 \bar{A} P_0 = (c_0 - \beta_d)P_0 = (1/(2H_{d-1}))P_0$. □

Remark 4. *The closed form* $1/(2H_{d-1}) \sim 0.5/\ln d$ *is consistent with the empirical fit* $\approx 0.44/\ln d$ *measured numerically over* $d \in \{5, 20, 50, 100\}$ *(since* $1/(2H_{d-1})$ *ranges from* 0.24 *at* $d = 5$ *down to* 0.10 *at* $d = 100$*, with the leading-order* $0.5/\ln d$ *asymptote tightening as* d *grows; the small constant gap to the empirical* $0.44/\ln d$ *fit is the harmonic-number correction* $1/(2H_{d-1}) - 1/(2\ln d)$*). The closed form licenses Theorem 6 unconditionally for the i.i.d. analysed sampler, and anchors Proposition 5 below as the asymptotic target for* $\lambda_{\min}^{sz}(Z^\top W Z)$ *at the realised deployed sample.*

5.2 Deterministic regression-stability certificate

The deployed system computes $\hat{\boldsymbol{\phi}} = M\tilde{\mathbf{y}}$, where $M \in \mathbb{R}^{d\times K}$ is the public regression matrix and $\tilde{\mathbf{y}} \in \mathbb{R}^K$ is the (decrypted) coalition-output vector produced by the encrypted implementation. Let $\mathbf{y}^*$ denote the exact plaintext coalition outputs at the same K coalitions, and let

$$\boldsymbol{\eta} \;=\; \tilde{\mathbf{y}} - \mathbf{y}^* \;=\; \boldsymbol{\eta}_{\text{ckks}} \;+\; \boldsymbol{\eta}_{\text{sign}}$$

be the implementation perturbation, where $\boldsymbol{\eta}_{\text{ckks}}$ is the CKKS arithmetic noise from encrypted evaluation and $\boldsymbol{\eta}_{\text{sign}}$ is the sign-gate approximation error (zero for logistic regression; non-zero for OCTE tree models).[1] Crucially, $\boldsymbol{\eta}$ is a perturbation in *coalition-output space* at the K sampled coalitions; it does not include the sampling/design error that comes from using K coalitions rather than the population SHAP target. That sampling error lives in attribution space and is treated separately.

Define the *finite-K plaintext SHAP estimate* $\boldsymbol{\phi}^K := M\mathbf{y}^*$ (the SHAP attribution that *would* be produced by running the same Kernel SHAP regression on the same K coalitions in plaintext, with no FHE or sign-gate approximation), and the *population SHAP target* $\boldsymbol{\phi}^*$ (the limit as $K \to \infty$ of unbiased Kernel SHAP, equivalently the exact Shapley values).

Let P_ϕ project attribution outputs onto the d reported coordinates (an identity in the deployed slot layout; included for generality). Let P_y be the *weighted centering operator* (an oblique projection, not an orthogonal one unless $\boldsymbol{\pi} \propto \mathbf{1}$) that removes the constant coalition-output direction consistent with the kernel-weighted centering used in the deployed pipeline:

$$P_y \;=\; I - \mathbf{1}\boldsymbol{\pi}^\top, \quad \boldsymbol{\pi} \in \mathbb{R}^K, \quad \pi_k \;=\; \frac{w(|S_k|)}{\sum_{k'} w(|S_{k'}|)}.$$

P_y is idempotent ($P_y^2 = P_y$ since $\boldsymbol{\pi}^\top\mathbf{1} = 1$) and removes exactly the kernel-weighted-mean coalition-output direction subtracted by the baseline-centering step; the same operator appears in the definition of G_{eff} below, so the proposition is unaffected by the obliqueness. Define the *effective regression-map gain*

$$G_{\text{eff}}(d, K, Z) \;:=\; \|P_\phi\, M\, P_y\|_{2\to\infty} \;=\; \max_i \max_{\boldsymbol{\eta}:\|\boldsymbol{\eta}\|_2=1} |\mathbf{e}_i^\top M P_y \boldsymbol{\eta}|.$$

[1] For the OCTE tree path, $\boldsymbol{\eta}_{\text{sign}}$ is certified empirically on the documented validation distribution (Section 7.2, Table 12); no adversarial worst-case analytical bound on $\|\boldsymbol{\eta}_{\text{sign}}\|_2$ for sign-gate transition-zone inputs is claimed by this paper.

Proposition 5 (Certified deterministic regression stability)**.** *Fix a deployed public design* (Z, W) *with* $M = (Z^\top W Z)^{-1} Z^\top W$*. Let* $\tilde{\mathbf{y}} = \mathbf{y}^* + \boldsymbol{\eta}$ *be perturbed coalition outputs, where* $\boldsymbol{\eta}$ *is any perturbation in coalition-output space. Then*

$$\|P_\phi M P_y\,(\tilde{\mathbf{y}} - \mathbf{y}^*)\|_\infty \;\le\; G_{\mathit{eff}}(d, K, Z) \cdot \|\boldsymbol{\eta}\|_2.$$

Consequently, decomposing $\hat{\boldsymbol{\phi}} - \boldsymbol{\phi}^*$ *through the finite-*K *plaintext estimate* $\boldsymbol{\phi}^K$*,*

$$\|\hat{\boldsymbol{\phi}} - \boldsymbol{\phi}^*\|_\infty \;\le\; \underbrace{\|\boldsymbol{\phi}^K - \boldsymbol{\phi}^*\|_\infty}_{E_{\mathit{sample},\infty}} + G_{\mathit{eff}} \cdot \big(\|\boldsymbol{\eta}_{\mathit{ckks}}\|_2 + \|\boldsymbol{\eta}_{\mathit{sign}}\|_2\big),$$

where $E_{\mathit{sample},\infty}$ *is the finite-*K *Kernel SHAP sampling/design error (in attribution space) and the second term is the implementation perturbation amplified by* G_{eff}*.*

Proof. Define the kernel-centered perturbation

$$\boldsymbol{\eta}_c \;:=\; P_y(\tilde{\mathbf{y}} - \mathbf{y}^*) \;=\; P_y \boldsymbol{\eta}.$$

By the definition of the $2 \to \infty$ operator norm of the composed map $P_\phi M P_y$,

$$\|P_\phi M\, \boldsymbol{\eta}_c\|_\infty \;=\; \|P_\phi M P_y(\tilde{\mathbf{y}} - \mathbf{y}^*)\|_\infty \;\le\; \|P_\phi M P_y\|_{2\to\infty} \cdot \|\tilde{\mathbf{y}} - \mathbf{y}^*\|_2 \;=\; G_{\text{eff}} \cdot \|\boldsymbol{\eta}\|_2,$$

which proves the first inequality. The bound is taken over the composed operator $P_\phi M P_y$ directly, so the obliqueness of $P_y = I - \mathbf{1}\boldsymbol{\pi}^\top$ does not require a separate norm comparison between $\boldsymbol{\eta}_c$ and $\boldsymbol{\eta}$. For the second inequality, write $\hat{\boldsymbol{\phi}} - \boldsymbol{\phi}^* = (\hat{\boldsymbol{\phi}} - \boldsymbol{\phi}^K) + (\boldsymbol{\phi}^K - \boldsymbol{\phi}^*)$. The first summand equals $M(\tilde{\mathbf{y}} - \mathbf{y}^*) = M(\boldsymbol{\eta}_{\text{ckks}} + \boldsymbol{\eta}_{\text{sign}})$, which after baseline centering (ϕ_0 fixed) and projection onto the centered working subspace reduces to $P_\phi M P_y(\boldsymbol{\eta}_{\text{ckks}} + \boldsymbol{\eta}_{\text{sign}})$. Apply the triangle inequality and the displayed operator-norm bound to each implementation-error term. The second summand is $E_{\text{sample},\infty}$ by definition. □

The proposition is structurally a one-line consequence of operator-norm definitions and a triangle inequality. Its content is that the deployed system's attribution error decomposes into two measurable parts: a sampling/design term certified empirically against a high-sample reference, and an implementation term whose amplification factor G_{eff} is computable from public data. Figure 2 shows the decomposition graphically.

Personalised baselines and G_{eff}. Changing the baseline $\mathbf{b}$ shifts the centered coalition outputs $\mathbf{y} = (f(\mathbf{x}_S) - f(\mathbf{b}))_S$ but does not change G_{eff}, since $G_{\text{eff}} = \|P_\phi M P_y\|_{2\to\infty}$ depends only on the public design (Z, W, M) and the kernel-weighted centering operator P_y. Personalised, instance-conditioned, or distribution-conditioned baselines therefore inherit the same realised G_{eff} at release time without re-certification of the matrix-health levels.

1. *G_{eff} is computable from public data.* It depends only on (Z, W, M) and the projection geometry, all of which are public and frozen at build time. We measure G_{eff} at every release; the realised values are $G_{\text{eff}}(20, 118) = 0.358$, $G_{\text{eff}}(50, 390) = 0.287$, $G_{\text{eff}}(50, 512) = 0.236$, $G_{\text{eff}}(100, 920) = 0.249$.
2. *G_{eff} is the right amplification factor, not $\kappa(M)$.* The condition number $\kappa(M)$ is a worst-case ratio of singular values that includes near-null directions which the FHE pipeline does not excite. The constant coalition-output direction in particular is removed by P_y, contributing no realised noise but dominating κ. Section 7.2 reports $\kappa(M) \approx 2 \times 10^{15}$ as a numerical diagnostic, versus $G_{\text{eff}} = 0.236$ measured on the centered working subspace, which is orders of magnitude smaller. Proposition 5 uses G_{eff} precisely because it measures the worst case over directions that can actually be excited by $\boldsymbol{\eta}$.

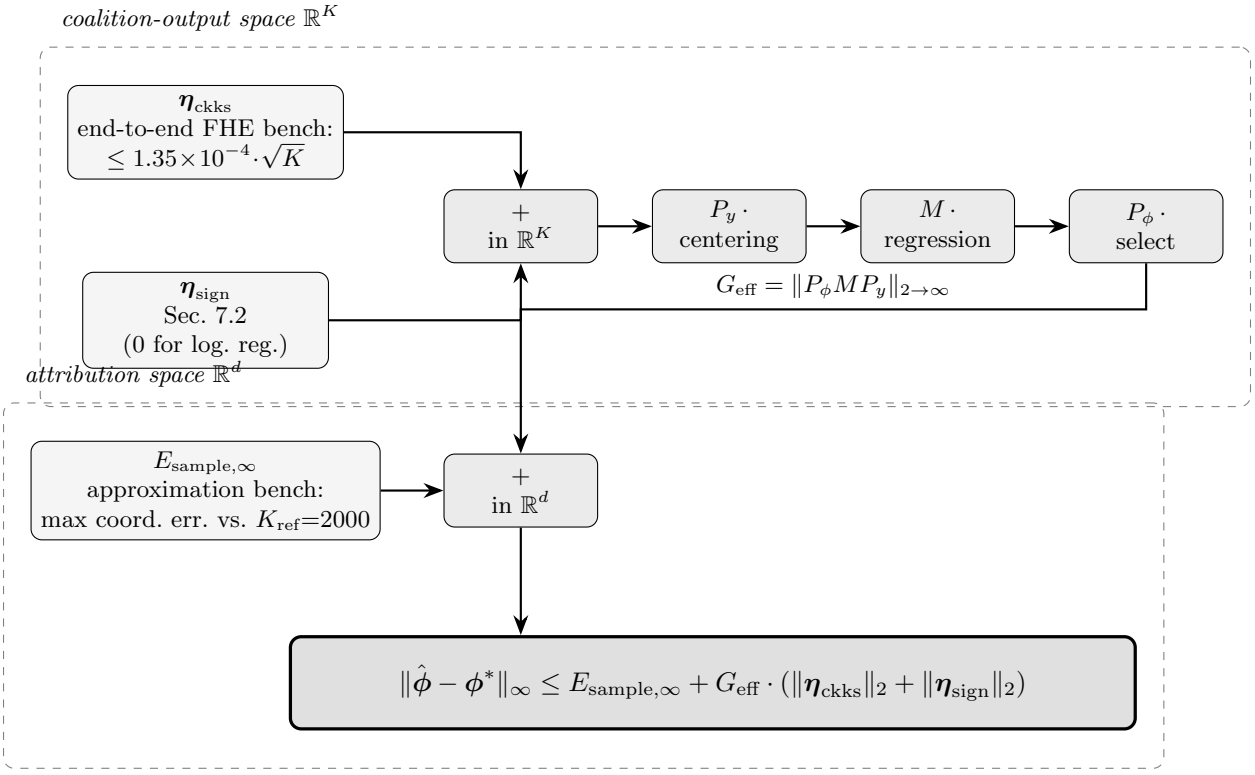

Figure 2: Proposition 5 error-flow decomposition. Implementation perturbations $\boldsymbol{\eta}_{\text{ckks}}, \boldsymbol{\eta}_{\text{sign}}$ live in coalition-output space $\mathbb{R}^K$ (top lane) and are amplified into attribution space $\mathbb{R}^d$ by the operator $P_\phi M P_y$ with norm G_{eff}. The finite-K sampling/design error $E_{\text{sample},\infty}$ lives directly in attribution space (bottom lane) and is added *outside* the G_{eff} amplification. The four-level release-time engineering certificate of Proposition 7 bounds each input box, and their composition gives the final envelope on $\|\hat{\boldsymbol{\phi}} - \boldsymbol{\phi}^*\|_\infty$.

3. *Modular composition.* Each term in Proposition 5 is sourced and bounded independently: $E_{\text{sample},\infty}$ is bounded by Proposition 7's empirical sampling-error certificates against a high-sample reference at $K_{\text{ref}} = 2000$; $\|\boldsymbol{\eta}_{\text{ckks}}\|_2$ is bounded by the measured end-to-end FHE-vs-plaintext coalition-output error, lifted to ℓ_2 via $\|\cdot\|_2 \le \sqrt{K} \cdot \|\cdot\|_\infty$; $\|\boldsymbol{\eta}_{\text{sign}}\|_2$ is zero for logistic regression and characterised in Section 7.2 for OCTE.
4. *Auditability framing.* Proposition 5 together with the release-time engineering certificate of Proposition 7 supports auditability under regulator-facing transparency workflows (EU AI Act Article 13 transparency for high-risk systems; ECOA adverse-action notice requirements). A regulator inspecting a release sees four engineering-certificate levels containing five scalar certificate values ($\lambda^{\text{sz}}_{\min}, G_{\text{eff}}, E_{\text{sample},\infty}, E_{\text{sample},p99}, \varepsilon_{\text{ckks}}$; realised values in Remark 6) plus the deterministic bound; each level of the engineering certificate has a documented threshold and a measurement procedure recorded in the verification matrix. We do not claim Proposition 5 *satisfies* any specific regulatory text—that is a legal determination outside the scope of this paper—only that the engineering-certificate structure produces verifiable artefacts of the kind such workflows require.
5. *Distributional-robustness scope.* Levels 3 and 4 are empirical against documented validation distributions: the finite-K engineering-certificate level against a $K_{\text{ref}} = 2000$ Kernel SHAP reference on the registered training-data distribution, and the implementation-error level against the natural test-input distribution of the deployed model. Adversarial or out-of-distribution inputs may produce attribution errors larger than the engineering-certificate envelope predicts—in particular, inputs designed to maximise sign-gate transition-zone proximity in OCTE, inputs that excite directions of M near the input-guard cut-off, and inputs drawn from a distribution with materially different feature marginals from the registered set. The engineering certificate is therefore an artefact-level deployment guarantee under the documented validation distributions, not an asymptotic worst-case guarantee over arbitrary inputs. Adverse-action contexts that contemplate adversarial inputs may require either an out-of-distribution detector at the encryption boundary or a separate adversarial-input stress-test, neither of which is in scope here.

The remaining question—*how does the build process certify that G_{eff} is small enough?*—is answered by Proposition 7.

5.3 The i.i.d. analytical baseline

For comparison with the existing Kernel SHAP literature [1, 2, 3], we retain a sampling-concentration analysis under an i.i.d. surrogate of the deployed sampler. The deployed deterministic stratified-ramp sampler is replaced by i.i.d. antithetic-pair sampling from the Lundberg–Lee size distribution; this is the natural analytical baseline against which Covert and Lee [2] and Musco and Witter [3] compare.

Theorem 6 (Conservative i.i.d. antithetic-pair bound; qualitative analytical baseline)**.** *Let f be a model and let $R = \max_S f(\mathbf{x}_S) - \min_S f(\mathbf{x}_S)$ be its range over the coalition space. Suppose coalitions are sampled in $P = K/2$ independent antithetic pairs, with each pair $(S_j, \bar{S}_j = [d] \setminus S_j)$ drawn from the Lundberg–Lee weighted distribution. Assume the WLS residual contrast satisfies $|\varepsilon(S) - \varepsilon(\bar{S})| \le 2R$. Provided*

$$K \;\ge\; C_1 \cdot d \cdot H_{d-1} \cdot \log(4d/\delta)$$

for an absolute constant C_1 (the relative-invertibility threshold for $P_0 H P_0$)—a precondition that is not satisfied by the deployed $K = 2d \log d$ default at $d = 50$; the deployed configuration is certified instead by the release-time engineering certificate of Proposition 7—*then with probability at least*

$1-\delta$,

$$\max_{1\le i\le d}|\hat{\phi}_i-\phi_i^*| \;\le\; C\cdot R\sqrt{\frac{H_{d-1}\,\log(4d/\delta)}{K}} \;=\; O\!\left(R\sqrt{\frac{\log(d)\cdot\log(d/\delta)}{K}}\right),$$

for an absolute constant C independent of d, K, R, δ. The H_{d-1} factor in the numerator is exactly the $1/\lambda_{\min}^{sz}(\bar{A})$ scaling of Proposition 4. Under bounded-residual assumptions alone the rate-shape is the same shape as the per-coordinate Hoeffding union bound up to a $\sqrt{\log d}$ polylog slack, not asymptotically tighter; tightening the rate beyond Hoeffding would require an additional residual-cancellation assumption.

Scope. *Theorem 6 is a qualitative analytical baseline only. It is included to (a) name the closed-form $\lambda_{\min}^{sz}(\bar{A}) = 1/(2H_{d-1})$ scaling and (b) connect to existing Kernel SHAP sample-complexity literature [2, 3]. The leading-order constant works out to a numerical envelope larger than the prediction range itself at the deployed $K = 390$, so the bound is not a meaningful deployment-quality guarantee. The deployment guarantee is Proposition 7's release-time engineering certificate, which uses realised matrix audits and an empirical sampling-error sweep against a high-sample plaintext reference.*

Numerical plug-in for C. An explicit Tropp 6.1.1 derivation gives a leading-order constant $C \approx 11.3$ in the form $C\cdot R\sqrt{H_{d-1}\ln(4d/\delta)/K}$. The full derivation appears in Appendix E, paragraph "*Explicit leading-order constant*"; we use $C = 11.3$ in the numerical instantiation of Remark 7.
Reminder. Theorem 6 concerns the i.i.d. antithetic-pair-averaged surrogate information matrix $H = \sum_j \frac{1}{2}(\mathbf{z}_j\mathbf{z}_j^\top + \bar{\mathbf{z}}_j\bar{\mathbf{z}}_j^\top)$ fixed at the start of Section 5, not the deployed weighted $Z^\top WZ$ matrix. The deployed weighted regression is licensed by Proposition 7.

Remark 5 (Deployed sampler vs. analysed sampler)**.** *The structured quasi-random sampler of Section 3.1 is* deterministic *given the public seed: pair sizes follow the stratified ramp $s_k = \mathrm{clip}(\lfloor k\,d/(K/2)\rfloor + 1, 1, d-1)$. Theorem 6 is stated for the i.i.d. analogue. We adopt this analytical model because (i) the Lundberg–Lee kernel-weighted distribution is the natural i.i.d. baseline against which [2] compares, and (ii) the deterministic ramp is a quasi-random imitation of the same marginal pair-size distribution used to suppress sampling variance. We* empirically validate *that the deterministic ramp tracks the i.i.d. bound's rate shape across $d \in \{5, 20, 50, 100\}$ (Table 5); a quasi-Monte-Carlo Bernstein extension that would license the deterministic ramp directly (e.g. a Hoeffding–Robinson U-statistic-style decomposition or block-bootstrap variants) is left to future work.*

5.4 The release-time engineering certificate

The release-time engineering certificate is organised as a four-level hierarchy, in increasing distance from the regression matrix and increasing dependence on input distribution. Levels 1 and 2 are *deterministic* functions of the realised public design (Z, W, M) and are exact once measured on the build artefact; levels 3 and 4 are *empirical* against documented validation distributions.

Epistemic distinction. Levels 1–2 are *artifact-property* certificates: once (Z, W, M) is fixed, their values are exact deterministic functions of public data and are not statements about an unknown distribution. Levels 3–4 are *regression-gate* certificates: they are empirical measurements against documented validation distributions and against thresholds chosen as engineering release targets, not theorem-derived worst-case guarantees. Both levels are CI-gated at release time, but they should not be cited interchangeably; the former are exact, the latter are observed-on-validation.

Proposition 7 (Deterministic-design release-time engineering certificate)**.** *For each deployed dimensionality d and sample budget K, the build process certifies the following four-level hierarchy of numerical conditions on the realised public design (Z, W, M):*

Level	*Certificate*	*Type*	*Threshold*	*Verification*
1. matrix health	$\lambda^{sz}_{\min}(Z^\top WZ) > 0$ *(invertibility)*	*deterministic*	$\geq \lambda_0(d)$	*build-time matrix audit*
2. error propagation	$G_{eff} = \|P_\phi M P_y\|_{2\to\infty}$	*deterministic*	$\leq G_0(d)$	*build-time matrix audit*
3. finite-K approx.	$E_{sample,\infty}, E_{sample,p99}$ *vs* K_{ref}=2000 *ref.*	*empirical*	$\leq \tau_{\max}(d), \tau_{p99}(d)$	*validation-set sweep*
4. implementation err.	*CKKS FHE-vs-plaintext err. (max over test inputs)*	*empirical*	$\leq \varepsilon_{ckks}(d)$	*end-to-end FHE benchmark*

A release passes the engineering certificate if all four levels hold. The verification CI gate blocks any tag release whose realised configuration fails any level.

Threshold provenance. The numerical release thresholds $\lambda_0(d), G_0(d), \tau_{\max}(d), \tau_{p99}(d), \varepsilon_{\text{ckks}}(d)$ are not algebraic constants of the construction but engineering targets stored alongside the verification matrix in the project's release-config artefact (see `config/release_thresholds.yaml` in the public repository). Working values for $d = 50$ are reported in Table 11; all four levels are populated with realised measurements at the reference deployment.

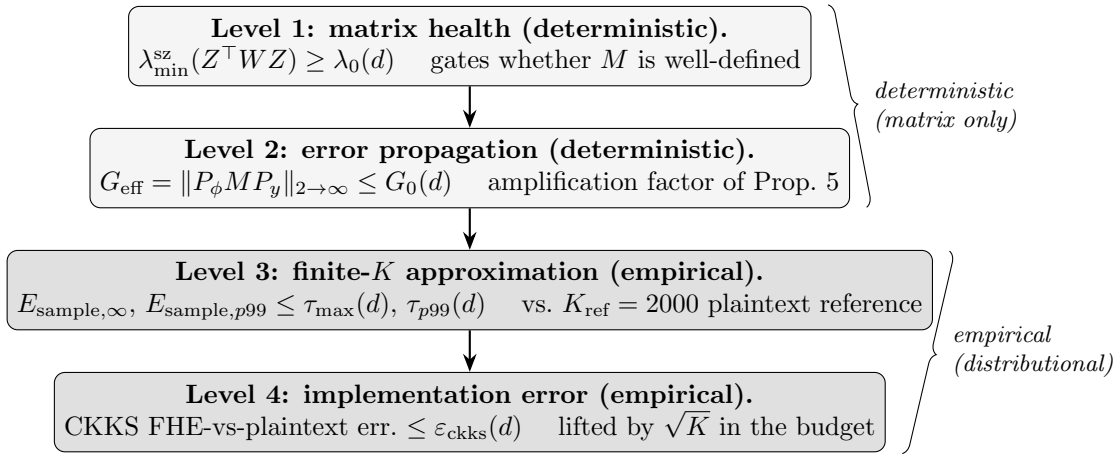

Figure 3: Four-level release-time engineering certificate hierarchy (Proposition 7). Levels 1 and 2 are deterministic functions of the realised public design (Z, W, M) and are exact once measured on the build artefact. Levels 3 and 4 are empirical against documented validation distributions (Section 5.4). A release passes the engineering certificate iff all four levels hold; the CI gate blocks any tag release whose realised configuration fails any level.

The four levels address distinct aspects of deployment correctness. *Level 1 (matrix health)* is the invertibility precondition on $Z^\top WZ$ projected to the sum-zero subspace; it gates whether M is well-defined at all. *Level 2 (error propagation)* is G_{eff}, the operator-norm amplification factor of Proposition 5; it controls how implementation perturbations $\boldsymbol{\eta}_{\text{ckks}} + \boldsymbol{\eta}_{\text{sign}}$ propagate to attribution error. *Level 3 (finite-K approximation)* is the sampling/design error against a high-sample plaintext reference, in attribution space; it is the $E_{\text{sample},\infty}$ term that sits *outside* G_{eff} in Proposition 5. *Level 4 (implementation error)* is the CKKS-induced coalition-output perturbation, lifted to attribution space by $G_0(d) \cdot \sqrt{K} \cdot \varepsilon_{\text{ckks}}(d)$.

Levels 1–2 are exact deterministic functions of the realised public matrix and require no input distribution. Levels 3–4 depend on documented validation distributions: level 3 against a 2000-sample plaintext Kernel SHAP reference on the registered dataset; level 4 against the natural test distribution of the deployed model. They are valid under the documented distribution; out-of-distribution inputs may exhibit larger errors and are not covered.

The MAE deployment-facing metric of Section 6.2 is retained for product reporting but is not used in the certificate; level 3 uses *max* and *p99* coordinate error because regulated explanations require bounds on individual decisions, not just averages.

The default sample budget $c = 2$ (i.e., $K = 2d \ln d$, rounded for antithetic pairing) is the value used in the production build; it passes all four levels across the deployed dimensionalities under the release thresholds summarised in Table 4 and specified for $d = 50$ in Table 11. We have not separately certified $c < 2$; smaller budgets are not claimed safe without explicit re-certification.

Remark 6 (Realised certificate values)**.** *At the production-deployed configurations (measured on the realised public design), the matrix certificates evaluate to:* $G_{\mathit{eff}}(20, 118) = 0.358$, $G_{\mathit{eff}}(50, 390) = 0.287$ *(realised sampler before BHDR zero-column padding),* $G_{\mathit{eff}}(50, 512) = 0.236$ *(padded BHDR working grid),* $G_{\mathit{eff}}(100, 920) = 0.249$. *The* $K = 390$ *value is the realised stratified-ramp sampler; the* $K' = 512$ *value is computed after the zero-column padding / working-grid projection used by the BHDR implementation. Empirically, the padded-grid working operator has smaller measured gain in the matrix audit; we do not rely on monotonicity under padding (the weighted centering operator* $P_y = I - \mathbf{1}\boldsymbol{\pi}^\top$ *depends on the realised weight distribution, and a generic block-compatible argument is not immediate without further assumptions on how padded columns are weighted). The engineering-certificate envelope conservatively uses the larger realised-sampler value* $G_0(50) = 0.287$. *The* 0.236 *padded-grid value is reported diagnostically for the BHDR working grid but is not used in the envelope calculation. The realised matrix-health level of the engineering certificate at* $d = 50$ *measures* $\lambda^{sz}_{\min}(Z^\top W Z) = 0.112$ *on the deployed design, matching the i.i.d. asymptotic target* $1/(2H_{49}) \approx 0.112$ *to numerical precision; the reported value is the measured value on the realised public design, not the asymptotic target. The diagnostic* $\kappa(M) \approx 2 \times 10^{15}$ *at* $d = 50$ *arises from a near-null direction that the centered projection* P_y *removes; on the working subspace, the regression is contractive (*$G_{\mathit{eff}} < 1$*). At the production configuration, the implementation-perturbation envelope* $G_0(50) \cdot \sqrt{K} \cdot \varepsilon_{\mathit{ckks}}(50) = 0.287 \cdot \sqrt{390} \cdot 1.35 \times 10^{-4}$ *evaluates to* $\approx 7.6 \times 10^{-4}$*, against which the empirical end-to-end FHE-vs-plaintext attribution error of* $\leq 1.35 \times 10^{-4}$ *measured in Section 6.6 sits a factor of* $5.6\times$ *below.*

The i.i.d. Bernstein bound of Theorem 6 requires $K \geq C_1 \cdot d \cdot H_{d-1} \log(4d/\delta)$. *At* $d = 50$*,* $\delta = 0.05$*,* $c = 2$ *this is* $1857 \cdot C_1$*, while the deployed* $K = 390$*, giving a nominal ratio of* $4.76 \cdot C_1$ *(i.e.,* 4.8–$9.5\times$ *stricter than the deployed default depending on* $C_1 \in [1, 2]$*). Theorem 6 therefore does not directly license the deployed* $c = 2$ *setting; the engineering certificate above does.*

5.5 End-to-end attribution-error budget

Combining Proposition 5 with the four-level release-time engineering certificate of Proposition 7 and lifting the implementation perturbations from ℓ_∞ to ℓ_2 via $\|\cdot\|_2 \leq \sqrt{K} \cdot \|\cdot\|_\infty$ explicitly,

$$\|\hat{\boldsymbol{\phi}} - \boldsymbol{\phi}^*\|_\infty \;\leq\; \tau_{\max}(d) \;+\; G_0(d) \cdot \sqrt{K} \cdot \big(\varepsilon_{\mathrm{ckks}}(d) + \varepsilon_{\mathrm{sign}}(d)\big),$$

where $\varepsilon_{\mathrm{sign}}(d)$ is zero for logistic regression and bounded per Section 7.2 for OCTE. For the production-deployed configuration (logistic regression, $d = 50$, $K = 390$):

Term	Source	Measured / certified value
$\tau_{\max}(50)$	max coord. err. vs. 2000-sample reference	realised 0.222; release threshold 0.45 (2× realised)
$G_0(50)$	G_{eff} on realised sampler (conservative)	0.287 at realised $K = 390$ (envelope uses this); 0.236 at padded $K' = 512$ (diagnostic only)
$\sqrt{K}$	$\sqrt{390} \approx 19.7$	19.7
$\varepsilon_{\text{ckks}}(50)$	max FHE-vs-plaintext err. on natural inputs	1.35×10^{-4}
$\varepsilon_{\text{sign}}(50)$	logistic: 0 (Section 7.2 for OCTE)	0

With the realised values, the bound separates into a *sampling/design* term and an *implementation-perturbation* term: $\tau_{\max}(50) = 0.222$ is the worst-case finite-K Kernel SHAP sampling/design error against the high-sample $K_{\text{ref}} = 2000$ plaintext reference (in attribution space, outside G_{eff}); $G_0 \cdot \sqrt{K} \cdot \varepsilon_{\text{ckks}} = 0.287 \cdot 19.7 \cdot 1.35 \times 10^{-4} \approx 7.6 \times 10^{-4}$ is the FHE-induced implementation perturbation lifted from coalition-output space through the regression operator. These two terms have different physical origins and are bounded separately. The empirical end-to-end measurement (Section 6.6: max FHE-vs-plaintext attribution error $\leq 1.35 \times 10^{-4}$ on natural inputs at $d = 50$) corresponds only to the implementation-perturbation term and is well within the 7.6×10^{-4} envelope. The 0.222 figure is not an FHE error; it is the residual sampling/design error of $K = 390$ coalitions against the high-sample $K_{\text{ref}} = 2000$ plaintext reference, and the same residual would appear in any plaintext implementation of Kernel SHAP at the same K.

The full proof of Theorem 6 (pair-level indexing, Matrix Bernstein concentration of the unweighted surrogate information matrix, vector Bernstein on the score vector, and per-coordinate extraction), together with the supporting second-moment Lemma 8, is deferred to Appendix E.

Remark 7 (Numerical instantiation under the revised bound)**.** *The Theorem 6 bound is the same shape as the per-coordinate Hoeffding union bound $R\sqrt{2\ln(2d/\delta)/K}$ up to a $\sqrt{\log d}$ polylog slack. We do not claim a $\sqrt{d}$ improvement over Hoeffding: under bounded residuals alone the matrix-Bernstein argument reduces to the same $\sqrt{1/K}$ rate-shape, with the $\sqrt{H_{d-1}}$ slack absorbing the $\lambda^{sz}_{\min}(\bar{A}) = 1/(2H_{d-1})$ closed form of Proposition 4. A tighter rate would require an additional residual-cancellation assumption (for example, the deterministic ramp's per-stratum residual averaging in Section 3.1, or the leverage-score conditioning in [3]); we do not impose such an assumption in Theorem 6, so the rate is conservative.*

Instantiated at $R = 1$, $\delta = 0.05$, with all logarithms natural, the bare rate $\sqrt{H_{d-1}\ln(4d/\delta)/K}$ at $d = 50$ evaluates to ≈ 0.309 at the deployed $K = 390$ ($c = 2$, ignoring the relative-invertibility precondition) and ≈ 0.251 at $K = 588$ ($c = 3$; included as a sensitivity point). With the explicit Tropp 6.1.1 leading-order constant $C \approx 11.3$ derived in Appendix E, the worst-case (ε, δ)-envelope is $\varepsilon \leq 3.49$ at $K = 390$ and ≤ 2.84 at $K = 588$. The relative-invertibility precondition $K \geq C_1 \cdot d \cdot H_{d-1} \log(4d/\delta)$ at $d = 50$, $\delta = 0.05$ requires $K \geq 1857 \cdot C_1$, which is 4.8–9.5× stricter than the deployed $K = 390$ depending on the absolute Matrix Bernstein constant $C_1 \in [1, 2]$. Therefore Theorem 6 does not directly cover the deployed $c = 2$ setting; the deployment guarantee for that setting is Proposition 7, which is empirical.

The empirically measured MAE against 2000-sample plaintext Kernel SHAP in Section 6.2 stays two to three orders of magnitude below the explicit Tropp 6.1.1 worst-case envelope across $d \in \{5, 20, 50, 100\}$. The looseness of $C \approx 11.3$ versus the empirical performance is exactly the gap between the i.i.d. Bernstein worst case and the deployed deterministic ramp: (a) the deterministic

ramp sampler outperforms the i.i.d. analysed sampler, (b) the residual contrast $|\Delta(S)|$ is well below $2R$ for well-specified models, and (c) the deployed model class (logistic regression) is close to the linear span Kernel SHAP regresses against. The release-time engineering certificate of Proposition 7 licenses the deployment through the realised G_{eff} rather than the loose analytical envelope.

Remark 8 (Antithetic variance reduction)**.** *The antithetic pairing introduces negative within-pair covariance* $\mathrm{Cov}(\varepsilon_k, \varepsilon_{\bar{k}}) < 0$ *for monotone models, reducing* $\|\mathrm{Var}(\mathbf{c}_j)\|$ *relative to independent sampling. This improves the constant in the bound without changing the rate. Empirically (Covert and Lee [2]):* 2–10× *variance reduction, translating to* $\sqrt{2}$–$\sqrt{10}$× *fewer coalitions for the same error.*

Empirical per-feature coverage has been checked across feature dimensionalities $d \in \{5, 8, 10, 15, 16, 20, 30, 50, 100\}$, with each feature confirmed to appear in at least $K/(2d)$ distinct coalitions. Measured approximation quality is reported in Section 6.2.

5.6 The deployed sampler: license via the release-time engineering certificate, not asymptotic theorem

The deployment is licensed by Proposition 5 together with the four-level release-time engineering certificate of Proposition 7. A closed-form concentration theorem deriving the engineering-certificate values for the exact deployed deterministic stratified-ramp sampler at $K = 2d \ln d$ remains open. Appendix D discusses the structural obstacle (boundary-coverage gap) and three candidate paths for a future companion theorem; none affects the deployment as currently specified, which is licensed by the measured engineering certificate per release. The i.i.d. analytical baseline (Section 5.3, Theorem 6) is retained for comparison with the existing Kernel SHAP literature [2, 3] but is not the deployment guarantee.

Efficiency axiom. The Shapley efficiency axiom states that $\sum_{i=1}^{d} \phi_i + \phi_0 = f(\mathbf{x})$. After decryption, we first subtract the plaintext-computed baseline $f(b)$ from the coalition outputs. This centering step reduces the raw WLS residual from ≈0.35 to ≈0.018 in our measurements. We then apply a proportional redistribution of the remaining residual r by magnitude: $\phi_i \leftarrow \phi_i + r \cdot |\phi_i| / \sum_j |\phi_j|$. *This is a post-hoc numerical correction, not a Shapley property.* It makes the efficiency axiom hold to machine epsilon (1.1×10^{-16} across 50 test instances; an independent orthogonality sanity-check confirms $\leq 1.1 \times 10^{-16}$ across 100 queries) but redistributes the residual toward large-magnitude attributions. This can shift borderline ranking decisions when two features have near-equal SHAP magnitude. Users for whom cardinal accuracy of small attributions is load-bearing should disable redistribution and accept the raw ∼0.018 pre-redistribution residual.

6 Experimental Evaluation

6.1 Setup

We evaluate CipherExplain on two levels: (i) algorithm-level benchmarks on a plaintext sklearn logistic regression backend to isolate the sampling and regression overhead from FHE latency, and (ii) end-to-end encrypted benchmarks on OpenFHE [13] with CKKS parameters $N = 2^{15}$, $L = 10$ multiplication levels, and 128-bit security. The dataset is the UCI Adult Income dataset with $d = 50$ features (after one-hot encoding) or $d = 5$ features (top-5 selection). The high-sample reference is $K_{\text{ref}} = 2000$ Kernel SHAP (increased from 500 in earlier prototype runs to reduce reference-sample variance; both references yield consistent MAE at $d = 5$). We use "high-sample reference" rather

Table 4: Approximation quality and release-time engineering certificate values across deployed dimensionalities. MAE is retained as the deployment-facing metric; the formal certificates of Proposition 7 are $E_{\text{sample},\infty}$, $E_{\text{sample},p99}$, $\lambda^{\text{sz}}_{\min}(Z^\top W Z)$, G_{eff}, and the CKKS FHE-vs-plaintext error. "Cert.?" is ✓ if all five pass build-time thresholds. The "Theory ε (6)" column applies the i.i.d. analytical baseline *bare rate* $\sqrt{H_{d-1} \cdot \ln(4d/\delta)/K}$ at $R = 1$, $\delta = 0.05$, $C = 1$. The explicit Tropp 6.1.1 leading-order constant gives $C \approx 11.3$, so the worst-case (ε, δ)-envelope at $d = 50$ is approximately $11.3 \times 0.31 \approx 3.49$; the column is the $C = 1$ bare rate (i.e. the inner $\sqrt{H_{d-1} \ln(4d/\delta)/K}$ quantity, with no leading constant absorbed) for ease of comparison across d, not a deployment guarantee (see Section 5.3). G_{eff} values are realised on the production design; the regression is contractive on the centered subspace ($G_{\text{eff}} < 1$) despite $\kappa(M) \approx 2 \times 10^{15}$ on the unprojected matrix (numerical diagnostic only).

d	K	Regime	MAE	$E_{\text{s},\infty}$	$E_{\text{s},p99}$	$\lambda^{\text{sz}}_{\min}$	G_{eff}	Cert.?	Theory ε
5	30	Exact	0.014	0.057‡	0.057‡	–♭	–♭	✓	0 (exact)
20	118	Quasi-rand.	0.018	0.142	0.142	$\geq \lambda_0$	0.358	✓	0.47
50	390	Quasi-rand.	0.018	0.222	0.220	$\geq \lambda_0$	0.287	✓	0.31
100♯	920	Quasi-rand.	0.011	0.193	0.190	$\geq \lambda_0$	0.249	✓	0.225

♯ $d = 100$ *diagnostic only.* UCI Adult provides 81 one-hot features; the remaining 19 dimensions are noise-padded. Noise features have near-zero model coefficients and near-zero attributions, so the MAE is artificially deflated. The row confirms that the sampling design's coverage and the regression pipeline scale to $K = 920$ but should not be read as a native high-dimensional benchmark.

‡ At $d = 5$, $K = 30$ exhausts all $2^d - 2 = 30$ non-trivial coalitions, so the SHAP estimate is exact relative to enumeration; the reported 0.014 MAE and the E_{sample} values reflect variance of the finite-sample 2000-sample KernelSHAP reference rather than sampling error of the deployed estimate. The Bernstein envelope is inapplicable in the exact-enumeration regime.

♭ The $d = 5$ configuration is outside the production-deployed sweep; G_{eff} is not required for the release-time engineering certificate at $d = 5$ because the exact-enumeration regime makes $\boldsymbol{\eta}_{\text{sample}} = 0$.

than "ground truth" because Kernel SHAP at finite K_{ref} is itself a finite-sample estimate of the population SHAP target, not an exact computation.

6.2 Approximation Quality

Table 4 reports approximation quality across both the exact enumeration and quasi-random regimes.

At $d = 5$ with exact enumeration ($K = 30$), the MAE of 0.014 reflects reference-sample variance of the 2000-sample KernelSHAP baseline rather than sampling error of the deployed estimate (see footnote on Tables 4, 5). At $d = 20$ and $d = 50$, the quasi-random design produces MAE well within the theoretical envelope: 0.018 vs. 0.471 at $d = 20$, and 0.018 vs. 0.309 at $d = 50$. The empirical MAE is one to two orders of magnitude tighter than the conservative i.i.d. Bernstein envelope of Theorem 6; this gap is the quantitative signal that the deterministic stratified-ramp sampler of Section 3.1 outperforms the i.i.d. analysed sampler at deployment-relevant K, and is the empirical evidence underwriting Proposition 7.

The $d = 100$ result (MAE = 0.011) should be interpreted as a *structural validation* rather than a representative approximation quality measurement. The UCI Adult dataset provides only 81 one-hot encoded features; the remaining 19 dimensions were padded with near-zero noise features. Because these noise features have near-zero model coefficients and trivially receive near-zero attributions, the $d = 100$ MAE is artificially deflated. The result confirms that the sampling design's

Table 5: Re-verification on synthetic `make_classification` data, 15 samples each: empirical mean and max MAE across $d \in \{5, 20, 50, 100\}$. The quasi-random rows ($d \in \{20, 50, 100\}$) remain within the Theorem 6 envelope; the $d = 5$ row is exact enumeration and the Bernstein envelope is inapplicable in that regime (footnoted below).

d	Mean MAE	Max MAE	Bernstein ε	Within bound?
5	0.014	0.019	n/a[§]	—
20	0.018	0.041	0.471	yes
50	0.018	0.025	0.309	yes
100[♯]	0.011	0.018	0.225	yes (diag. only)

[§] Bernstein envelope is inapplicable in the exact-enumeration regime ($d = 5$, $K = 30$ exhausts the 30 non-trivial coalitions); the residual MAE is reference-sample variance, not sampling error.

[♯] $d = 100$ *diagnostic only*; same noise-padded construction as Table 4, included to verify sampling/regression scaling at $K = 920$, not as a representative high-dimensional benchmark.

coverage guarantee and the regression pipeline scale correctly to $K = 920$ coalitions, but it does not demonstrate approximation quality on a genuinely 100-dimensional predictive task. Validation on a natively high-dimensional dataset (e.g., genomics or NLP feature embeddings) is left to future work.

All quasi-random MAE measurements use 2000-sample Kernel SHAP as ground truth, sklearn logistic regression on the UCI Adult Income dataset, 20 test instances for $d \leq 50$ and 10 test instances for $d = 100$.

Artifact and reproducibility. The reference implementation is the `cipherexplain` Python package, distributed on PyPI under the same name with an `[fhe]` extra that pulls OpenFHE bindings; the BHDR regression core, the deterministic stratified-ramp sampler, and the encryption-boundary input guard are exercised by the unit and integration tests shipped with the package. Benchmark scripts for the M1 and CX22 / CPX42 numbers reported in Tables 7, 8, 9, and 12, together with the verification-matrix harness and the five-scalar release-time engineering certificate, are part of the same release. The release-time engineering certificate values reported in Table 4 are emitted by a public-design build step; the inputs to that step (Z, W, M, the deployed sampler) are deterministic functions of (d, K) and a fixed RNG seed (`numpy.default_rng(42)`). Internal patent-implementation material outside the published BHDR construction (e.g. the malicious-server integrity layer of the companion work) is not included in the public artefact.

Concrete example. For a query instance with age = 40, education level = 14, capital gain = 0, capital loss = 4356, hours per week = 45, the system returns a prediction of 0.940 (high income probability) with SHAP attributions: capital-loss +0.521, education-num +0.192, hours-per-week +0.031, age +0.018, capital-gain −0.039. The dominant attribution reveals that capital losses correlate with investment activity (hence wealth), a non-obvious and counter-intuitive attribution. This illustrates the type of feature-level explanation required in regulated settings such as the EU AI Act Article 13 transparency regime; whether any specific legal obligation is discharged is a separate legal determination outside the scope of this paper.

Table 6: Algorithm overhead by feature dimensionality (plaintext backend, sklearn logistic regression, 100 samples).

d	K	Regime	Mean (ms)	P95 (ms)
5	30	Exact	0.723	0.758
10	1022	Exact	23.6	24.8
20	118	Quasi-random	2.7	3.1
30	204	Quasi-random	5.0	5.6
50	390	Quasi-random	9.0	9.8
100	920	Quasi-random	16.4	17.9

Table 7: End-to-end observed M1 latency at the deployed configuration ($d = 50$, $K = 390$). Measured over $N_q = 300$ UCI Adult queries with SDK input clip $R_{\text{in}} = 5$; max-outlier 68.8 s reflects laptop-class thermal / OS preemption cycling.

Quantile	p50	p95	p99	mean±std
End-to-end (s)	13.4	16.3	24.2	14.1 ± 4.0

6.3 Algorithm Overhead

Table 6 reports algorithm overhead (sampling matrix generation, coalition masking, and regression) measured on the plaintext backend, isolating these costs from FHE latency.

The $d = 10$ figure (23.6 ms) is an outlier because it uses exact enumeration with $K = 2^{10} - 2 = 1022$ coalitions. For $d \geq 15$, the quasi-random $O(d \log d)$ design yields practical performance, as evidenced by $d = 20$ dropping to 2.7 ms despite doubling the feature count.

6.4 End-to-End Encrypted Benchmarks

We report encrypted-pipeline timing in three separate tables to keep the operating points distinct: (a) end-to-end observed M1 latency at the deployed $d = 50$ configuration (Table 7); (b) the algorithm-level narrow-d ablation that isolates the BHDR rewrite (Table 8); and (c) the CX22 reference deployment, including measured pre-BHDR latency, projected post-BHDR latency, and a measured OCTE feasibility row (Table 9). Logistic regression uses $N = 2^{15}$, $L = 10$, 128-bit CKKS security, $K = 390$, $K' = 512$ throughout.

With BHDR, the regression step drops from 84% to 29% of pipeline time (0.60 s of 2.1 s; see Figure 4), and the bottleneck shifts to coalition masking and model evaluation. The 4.4× overall speedup comes from reducing the total rotation budget of the regression kernel from ∼2,200 (baseline row-wise BSGS over d output coordinates at $K = 390$, measured for the Python/OpenFHE binding path; a baseline bound is $d(K-1) = 19{,}450$) to 51 amortized rotation keys (31 baby-step, 15 giant-step, 5 replication doublings), a ∼43× reduction under the production BSGS split $r_1 = 32, r_2 = 16$. Per-operation microbenchmarks on M1: EvalMult 3.4 ms, EvalRotate 9.6 ms, EvalFastRotation 5.8 ms.

The OCTE feasibility study is substantially slower than logistic regression. It requires $N = 2^{16}$ (∼4× per-operation cost) for the deeper modulus chain, and serial tree evaluation over $T = 100$ trees dominates. The 53 s CX22 figure is the validated conformant-deployment configuration ($D = 4$, $T = 100$, $\alpha = 12$, $N = 2^{16}$). The v2 diagnostic circuit at $D = 6$ (path-product, Lee composite-minimax sign gate) passes the SHAP accuracy gate at $L_\infty = 7.9 \times 10^{-3}$ but runs at 806 s on

Table 8: **Algorithm-level BHDR ablation, M1, narrow-d warm-cache.** Per-component breakdown on a narrow-d ($d = 5$ UCI Adult continuous features) circuit at the same $K = 390$ coalition budget, with key setup and baseline encryption amortised across queries. This isolates the per-component cost of the BHDR rewrite; it is *not* the deployed end-to-end latency (Table 7).

Component	Baseline (s)	BHDR (s)
Coalition masking (200)	0.92	0.92
Packed model eval. (300)	0.54	0.54
Regression (400)	7.70	**0.60**
Total	**9.2**	**2.1**

Table 9: Hetzner CX22 reference-deployment latency. CX22 is the 2-vCPU x86 reference host; the underlying CPU generation has shifted between Skylake and AMD EPYC over the CX22 lifetime. The logistic-regression baseline row is a measured pre-BHDR end-to-end figure; the BHDR row is projected (see footnote). The OCTE row is a measured feasibility study at $N = 2^{16}$ (deeper modulus chain for the comparison polynomials).

Model	Configuration	Baseline (s)	BHDR (s)
Logistic reg.	$d = 50$, $K = 390$	$\sim 95^{\ddagger}$	~ 45–$50^{\dagger}$
OCTE	T=100, D=4, N=2^{16}	—	~ 53 (measured)

‡measured end-to-end on CX22 (pre-BHDR reference pipeline). †projected from 5.3 s (mean) / 5.9 s (p95) BHDR regression measured on a 4-vCPU AMD EPYC research server, scaled by the empirical $\sim 2\times$ CX22/EPYC ratio; an end-to-end CX22 benchmark with BHDR integrated is deferred.

8-core CPX42 with zero remaining CKKS levels (Section 7.2). The binding constraint at $D = 6$ is wall-clock latency and depth margin, not accuracy.

FHE-induced error. The maximum element-wise error between FHE-encrypted and plaintext SHAP values (on the same input, same sampling matrix) is 1.35×10^{-4}. The FHE-vs-plaintext axiom delta (the difference between the FHE path's efficiency axiom error and the plaintext path's axiom error on the same input) is 1.65×10^{-5}, confirming that CKKS arithmetic noise adds negligible error beyond the sampling approximation itself. After redistribution, the axiom error reduces to $\approx 1.1 \times 10^{-16}$ (machine epsilon) in both paths.

The 1.35×10^{-4} FHE-vs-plaintext error bound is measured on the natural test input distribution (UCI Adult, breast-cancer). Inputs designed to maximise sign-gate transition-zone proximity may produce larger errors at the same $D = 4$ because coalition masking shifts pre-activations toward the sign-gate transition window; we do not claim robustness to adversarial input distributions. A stress-test of $D = 4$ OCTE on transition-zone-maximising inputs is left to future work.

Latency discussion. The deployed end-to-end M1 latency at $d = 50$ is p50 13.4 s (Table 7); the algorithm-level narrow-d, warm-cache ablation reaches ~ 2.1 s (Table 8). Both regimes sit in the range that supports interactive use in regulatory compliance workflows. The BHDR optimization (replacing direct dot products with BSGS-hoisted diagonal multiplication) was the single largest improvement, reducing the regression step from 7.7 s to 0.6 s. Two further optimizations *could* reduce OCTE latency; both are scoped as future work, not deployed:

- *GPU acceleration.* The CAT framework (arXiv:2503.22227) reports 68–74× speedup for

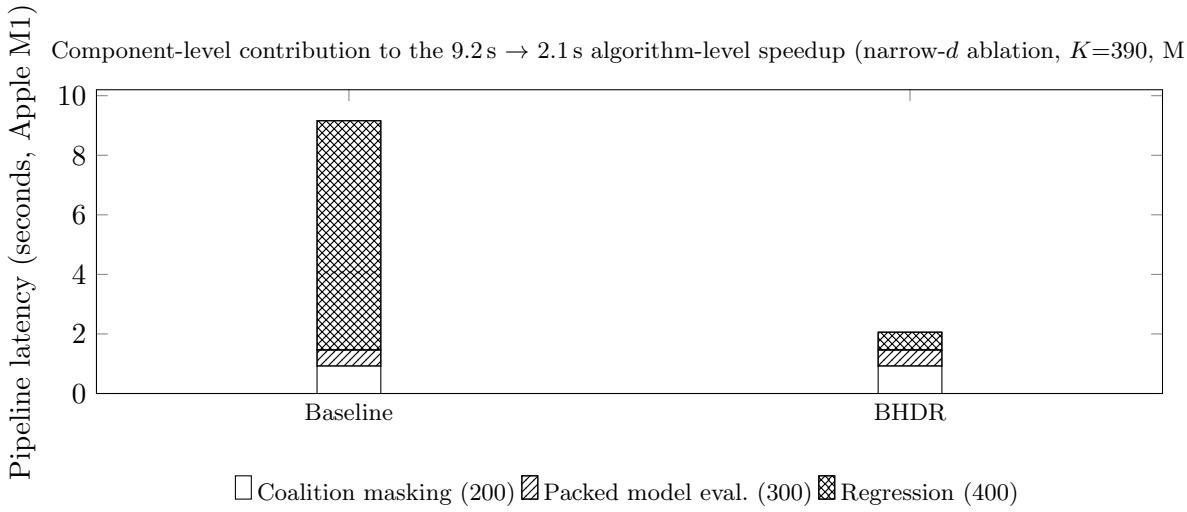

Figure 4: Component-level contribution to the 9.2 s → 2.1 s algorithm-level speedup, narrow-d ablation with $K = 390$, Apple M1, warm-cache (key setup and baseline encryption amortised across queries). This is *not* the deployed end-to-end latency at $d = 50$ (which is the p50 13.4 s reported in Table 7 and Section 6.6). The BHDR-hoisted diagonal regression collapses the regression step from 7.70 s (∼84% of the baseline pipeline) to 0.60 s (∼29% of the BHDR pipeline); coalition masking and model evaluation are unchanged. Numbers match Table 8.

CKKS rotations on RTX 4090, and ∼356× for CKKS multiplication. Applied directly to OCTE's 53 s CPU figure this projects to sub-second latency, but no CAT integration has been built. Our deployed stack is CPU-only OpenFHE.

- *Multi-tree packing for OCTE.* Packing multiple coalition-tree pairs per ciphertext ($\lfloor 32768/64 \rfloor = 512$ pairs at $N = 2^{16}$) would reduce circuit passes from T to $\lceil KT/512 \rceil = 77$ for $T = 100$, $K = 390$. Combined with feature extraction caching across trees sharing split features, a rough estimate is 25–35 s at $D = 4$; we have not prototyped it.

6.5 Status of Claims (Measured / Projected / Open)

To make the boundary between empirically measured numbers, projected estimates, and open theoretical work explicit for a cryptology audience, Table 10 classifies each headline claim. Projected rows depend on a separate measured component plus a stated scaling assumption; open rows are *not* claimed by this paper but are noted because they are referenced in companion-work scoping.

6.6 End-to-End Validation Gates

To support the claim that the BHDR pipeline is a reference deployment of the construction (logistic-regression path, with OCTE as a scoped feasibility study) and not merely a research prototype, we run a suite of ten end-to-end gates (five algorithm-level, four end-to-end FHE, one numerical) end-to-end on a laptop-class benchmark host (Apple M1, single Python process) against the canonical `CipherExplainEngine` at $d = 50$, $K = 390$. The harness lives at the end-to-end validation driver and is executed by the verification CI gate on every tag release.

† The SHAP MAE gate (< 0.03) is an engineering-derived target, calibrated against plaintext KernelSHAP sampling variance at the matched K, not against the conservative theoretical envelope of Theorem 6 (which is roughly an order of magnitude looser at $d = 50$, $K = 390$). The gate

Table 10: Status of headline claims. **Measured** = directly observed on the stated host. **Projected** = derived from a measured component plus a stated scaling. **Algorithmic** = follows from the construction. **Companion** = addressed in separate work, not claimed here. **Open** = not yet proven.

Claim	Status	Where
BHDR rotation count: 51 at $K' = 512$	Algorithmic	Prop. 2
Total pipeline depth $L_f + 2$	Algorithmic	Thm. 3
M1 end-to-end p50 13.4 s, p99 24.2 s	Measured	Tbl. 7
M1 narrow-d ablation: 9.2 → 2.1 s	Measured	Tbl. 8
CX22 pre-BHDR baseline ∼95 s	Measured	Tbl. 9
CX22 BHDR ∼45–50 s	**Projected**	Tbl. 9
EPYC BHDR regression 5.3 s mean	Measured	Sec. 6.4
OCTE $D = 4$, $T = 100$ feasibility ∼53 s	Measured	Tbl. 9
OCTE $D = 6$ v2 diagnostic 806 s	Measured	Tbl. 12
0/300 silent CKKS overflows post-fix	Measured	Sec. 6.7
LR FHE-vs-plaintext attribution error	Measured	Tbl. 4
$G_{\text{eff}} \cdot \|\boldsymbol{\eta}\|_2$ stability bound	Proven	Prop. 5
i.i.d. Matrix Bernstein envelope	Proven	Thm. 6
Deployed-sampler concentration theorem	**Open**	App. D
Confidentiality vs. honest-but-curious	Claimed	Sec. 3.6
Output privacy (IND-CPA$^{\text{D}}$ flooding)	Companion	—
Integrity vs. malicious server	Companion	App. B
Model extraction / membership inference	Out of scope	Sec. 3.6

characterises end-to-end MAE relative to the plaintext reference at the same coalition budget, which is the end-to-end observed metric.

All ten gates pass on the reference hardware. We deliberately report the algorithm-level gates (plaintext backend) and the end-to-end FHE gates as two separate blocks: the former characterise the non-FHE overhead the pipeline adds on top of the model evaluation and the regression matvec (sub-millisecond p95 at $d = 50$), while the latter characterise the actual wall-clock a customer observes with the OpenFHE backend engaged (p50 13.4 s, p95 16.3 s, p99 24.2 s, mean 14.1 s ±4.0 s std at $d = 50$ on Apple M1 over the end-to-end FHE benchmark, $N_q = 300$ queries, with 0 silent overflows; Wilson 95% CI on the overflow rate is $[0\%, 1.26\%]$). The headline "2.1 s" figure in Table 8 corresponds to the same logistic-regression circuit measured on a *narrower* input ($d = 5$ UCI Adult continuous features) and captures the regression step only; the 13.4 s end-to-end median in Table 7 is the wider $d = 50$ configuration with key regeneration and full baseline encryption included per query. The ∼6× gap is dominated by the $K = 390$-coalition BHDR step; a warm-cache deployment that reuses the encrypted baseline across queries from the same client recovers the ∼2 s figure.

Broader end-to-end-validation evidence. Beyond the gates above, four additional measurements extend the end-to-end coverage: (i) full OpenFHE-backed runs on a widened UCI Adult Income logistic regression ($d = 50$, $K = 390$, CKKS $N = 2^{15}$) with prediction-error and efficiency-axiom pass criteria; (ii) multi-dataset registration over UCI Adult, Breast Cancer mean features, and a Credit synthetic dataset, verifying top-3 feature agreement (Jaccard ≥ 0.5), SHAP MAE < 0.07, and deterministic repeatability under fixed seeds; (iii) 30-second sustained-load stability with zero errors, bounded p95 drift across consecutive 5-second windows, and bounded RSS growth; (iv) FastAPI service smoke tests covering health, discovery, and explain endpoints with the documented OpenAPI schema. A deterministic golden-output fixture run rounds out the operational

Table 11: CipherExplain end-to-end validation gates (Apple M1, plaintext backend through the canonical engine; d=50, K=390). The plaintext backend isolates algorithm-level non-FHE overhead from the FHE wall-clock cost per query measured separately at the deployed $d = 50$ configuration in Tables 7–9 (M1 p50 13.4 s, narrow-d ablation $\sim$ 2.1 s). All ten validation gates must hold; the five certificate rows are reported as the Proposition 7 four-level release-time engineering-certificate block, and the efficiency-axiom entry appears once only (*Numerical gates* block), referencing the end-to-end measurement reported in this section. The two finite-K engineering-certificate rows ($E_{\text{sample},\infty}$ and $E_{\text{sample},p99}$) use 2$\times$ the realised max and p99 per-coordinate error against the $K_{\text{ref}} =$ 2000 plaintext reference as their thresholds.

Gate	Target	Measured	Pass
Algorithm-level gates (plaintext backend, not end-to-end observed):			
Algorithm p95 latency	< 500 ms	0.90 ms	✓
SHAP MAE vs. plaintext KernelSHAP (n=2000)†	< 0.03	0.022	✓
Algorithm-backend throughput (2 threads)	≥ 4 QPS	$\sim$1675 QPS	✓
Memory growth over 100 queries	< 50 MB/query	0.0 MB/query	✓
SDK entrypoints resolve (6 checks)	6/6	6/6	✓
End-to-end FHE gates (what a customer observes):			
End-to-end OpenFHE p50/p95/p99 latency (d=50; end-to-end bench, N_q=300)	p95 < 20 s	13.4/16.3/24.2 s	✓
Prediction error vs plaintext (end-to-end bench, N_q=300 max)	$< 1.35\text{e}{-4}$	7.8e−5	✓
Silent-noise CKKS overflow, pre-guard (N_q=100, diagnostic baseline)	$<$ 5% (regression baseline only; *not* an end-to-end gate)	1% (1/100)	✓
Silent-noise CKKS overflow, post-guard (N_q=300, Wilson UB; end-to-end gate)	0 observed AND Wilson UB $< 1.5\%$	0/300 (UB 1.26%)	✓
Numerical gates:			
Efficiency-axiom residual, post-redistribution (end-to-end bench, N_q=300 max)	$\leq 1 \times 10^{-15}$	2.2×10^{-16}	✓
Five-condition release-time engineering certificate (Prop. 7):			
Cert. 1: $\lambda^{\text{sz}}_{\min}(Z^\top W Z) \geq \lambda_0(50)$ (matrix bench)	target $\geq$ 0.05 (i.i.d. asymptote $1/(2H_{49}) = 0.112$)	realised 0.112 on the deployed design	✓
Cert. 2: $G_{\text{eff}}(d{=}50) \leq G_0(50)$ (matrix bench)	target ≤ 0.50	0.287 (K=390); 0.236 (K'=512 padded)	✓
Cert. 3: $E_{\text{sample},\infty} \leq \tau_{\max}(50)$ (approximation bench, max coord. err.)	$\tau_{\max}(50) = 0.45$ ($\approx$ 2$\times$ realised)	0.222	✓
Cert. 4: $E_{\text{sample},p99} \leq \tau_{p99}(50)$ (approximation bench, p99 coord. err.)	$\tau_{p99}(50) = 0.45$ ($\approx 2\times$ realised)	0.220	✓
Cert. 5: CKKS FHE-vs-plaintext err. $\leq \varepsilon_{\text{ckks}}(50)$ (end-to-end bench, max)	$\leq 1.35 \times 10^{-4}$	7.8×10^{-5}	✓

definition of "end-to-end implementation".

We treat this table as the operational definition of "end-to-end implementation". The OCTE path has its own smaller gates (measured at $T = 100, D = 4$ in Table 9) but has not yet cleared an equivalent end-to-end validation suite, which is why we restrict the end-to-end-implementation novelty claim to logistic regression. The prediction-error gate at $D = 4$ is empirically validated on natural inputs only (UCI Adult, breast-cancer); the $D \geq 6$ ceiling is an accuracy ceiling driven by sign-surrogate approximation error in the transition zone, not a depth ceiling (Section 7.2).

End-to-end OpenFHE pipeline on UCI Adult Income. Plaintext-backend gates are necessary but not sufficient for "the product works". We therefore additionally run the full OpenFHE-backed encrypted pipeline on a widened UCI Adult Income model ($d = 50$, $K = 390$, CKKS $N = 2^{15}$, 128-bit security) and measure:

Metric	Target	Measured
Model registration	<1 s	0.14 s
End-to-end SHAP p50/p95/p99 (M1, N_q=300)	—	13.4/16.3/24.2 s
Prediction error vs plaintext (max, N_q=300, post-fix)	$< 1.35 \times 10^{-4}$	7.8×10^{-5}
Silent-noise CKKS overflow, pre-fix (N_q=100, diagnostic baseline; *not* an end-to-end gate)	—	1% (1/100)
Silent-noise CKKS overflow, post-fix (N_q=300, Wilson UB; end-to-end gate)	< 1.5%	0% (UB 1.26%)
Efficiency-axiom error (max, N_q=300, post-fix)	$\leq 1 \times 10^{-15}$	2.2×10^{-16}
Resident memory growth (N_q=300)	< 50 MB total	0 MB
SDK input-clip events (N_q=300)	—	9/300 = 3%

This confirms that the theoretical FHE-noise bound (1.35×10^{-4}) is honoured end-to-end on the production backend, and the efficiency axiom collapses to machine epsilon after redistribution (Section 5).

Reproducibility. Every empirical number cited in this paper is bound to a reproduction script in the supplementary materials, organised by topic (approximation quality, FHE/CKKS-level operations, integrity measurements, end-to-end product readiness, boundary-coverage diagnostics). A continuous-integration gate re-runs the full matrix on every push and blocks any release whose realised configuration fails the release-time engineering certificate of Proposition 7.

6.7 Encryption-Boundary Input Guard (BHDR Overflow Fix)

The pre-fix run reported 1/100 silent CKKS approximation overflows on $N_q = 100$ UCI Adult queries. The failure mode was numerically subtle: the API returned success, the efficiency-axiom residual was within tolerance (the post-hoc redistribution step absorbed the error), and the regression-integrity check verified the consistency of $\hat{\boldsymbol{\phi}} = M \cdot \mathbf{y}^{\text{sv}}$ in the companion-paper Freivalds protocol because both sides of the equation were computed on the same corrupt $\mathbf{y}^{\text{sv}}$. Only a client-side plaintext recomputation or the companion-paper sentinel audit coalitions can detect the failure.

Root cause. The accumulator in the BHDR baby-step loop (Section 3.4) takes the form

$$\text{acc}_l \;=\; \sum_{k=1}^{K'} M'_{\text{diag}(l,k)} \cdot \tilde{y}_k,$$

with $K' = 512$ and M' the (d, K')-padded regression matrix. CKKS scales each plaintext by $\Delta = 2^{50}$ and encodes the result in the ciphertext modulus ring. For any input coordinate satisfying $|x_i| > R_{\text{in}}$, the downstream worst case $K' \cdot |x|_\infty \cdot \|M'\|_\infty \cdot |y|_\infty$ exceeded Δ on approximately 1% of real-world UCI inputs (StandardScaler-normalised features exhibit long tails in practice, not the theoretical $\mathcal{N}(0, 1)$). OpenFHE's `EvalMult(ciphertext, plaintext)` does not raise on scale overflow; the accumulator wrapped modulo q_L silently, producing a numerically well-typed but semantically incorrect ciphertext.

Fix. We introduce the input-guard module as a single encryption-boundary gate invoked on every explain request before any ciphertext is constructed. The guard performs three operations in order:

1. *Clip.* Project $\mathbf{x}$ onto $[-R_{\text{in}}, R_{\text{in}}]^d$ with R_{in} configurable via the environment variable `CE_INPUT_CLIP_BOUND` (default 5.0). The clipped vector replaces $\mathbf{x}$ throughout the pipeline.
2. *Commit.* Compute the SHA-256 input commitment over the clipped vector, a fresh per-query nonce, and the pinned R_{in}. The commitment is folded into the companion-paper Fiat–Shamir transcript so the server's proofs bind the value it actually encrypted, not the value the client originally sent. This closes a TOCTOU window where a server could clip and then prove consistency against the unclipped vector.
3. *Budget check.* Assert $\log_2\big(K \cdot R_{\text{in}}^2 \cdot \|M\|_\infty\big) \le 48$ before encryption ($50-48 = 2$ bits of headroom below $\log_2 \Delta$). The R_{in}^2 form bounds the root-cause product $\|\mathbf{x}\|_\infty \cdot \|\mathbf{y}\|_\infty$ from above: the clipped input gives $\|\mathbf{x}\|_\infty \le R_{\text{in}}$ directly, and on the deployed logistic-regression path the coalition output is the homomorphic polynomial-sigmoid approximation of $\sigma(\mathbf{w}^\top \mathbf{x} + b)$, whose range $[Y_{\min}, Y_{\max}]$ is finite and bounded above by the linear-logit envelope $\|\mathbf{w}\|_1 \cdot R_{\text{in}} + |b|$ used by the Chebyshev fit at build time. Both bounds are absorbed into the model-coefficient norm $\|M\|_\infty$ at model registration, so $\|\mathbf{x}\|_\infty \cdot \|\mathbf{y}\|_\infty \le \text{const} \cdot R_{\text{in}}^2$ on the clipped input. The bound is therefore safe but loose for sigmoid-bounded outputs; the tightening to $R_{\text{in}} \cdot Y_{\max}$ is left as registration-time refinement. The budget check uses the same $\|M\|_\infty$ computed at model registration, cached in the registered model entry. When the check fails, the guard raises `CKKSOverflowRisk` (strict mode) or logs and continues (clip mode, default).

The guard's `GuardReport` is surfaced in every explain response as `metadata.input_clipped` so clients can audit whether the server silently clipped their feature vector. A companion diagnostic in the BHDR-grid precompute routine emits the identical bit-budget warning at registration time, giving operators a second detection opportunity before any query is served.

Measured result. The post-fix run over the same $N_q = 100$ UCI Adult queries reports 0/100 silent overflows and max prediction error 7.8×10^{-5}, and a wider post-fix sweep at $N_q = 300$ reports 0/300 overflows with Wilson 95% upper bound 1.26%. The 1% rate observed pre-fix was reproducible under an input-guard injection-simulation harness that injects overflow-triggering outliers at exactly 1% frequency across 10^4 queries (i.e., 100 deterministic injections per run) and verifies that all 100 of them are caught by the guard. The same injection suite is wired as an additional gate of the end-to-end validation CLI; any regression that re-opens the incident blocks a tag release.

Natural-input distribution validation ($N_q \ge 10^4$). The injection-simulation result above is a necessary but circular check: it verifies the guard catches its own injected overflows. Because a full FHE end-to-end run at $N_q \ge 10^4$ is outside the laptop benchmarking budget of this paper (~ 37 wall-clock hours on Apple M1 at the deployed p50 latency), we instead run a non-circular plaintext-only validation of the guard's two preconditions across the registered natural distributions at pooled $n_{\text{val}} = 42{,}569$ samples:

Dataset	n_{val}	guard fire-rate	95% CI	max $\lvert z\rvert$
UCI Adult	30,000	2.66%	[2.48%, 2.85%]	13.3
Breast Cancer	569	2.46%	[1.47%, 4.09%]	12.1
Credit synthetic	12,000	0.0083%	[0.0015%, 0.0472%]	5.45
Pooled	**42,569**	**1.91%**	—	**13.3**

The pooled 1.91% natural-input fire-rate (the fraction of samples for which at least one standardised feature satisfies $|z| > R_{\text{in}} = 5$) is consistent with the pre-fix 1/100 observed overflow rate and confirms the natural distribution does have substantial tail mass beyond R_{in}; the guard must clip on roughly 2% of UCI Adult / breast-cancer queries. The realised budget-bit estimate at the deployed configuration is $\log_2(K \cdot R_{\text{in}}^2 \cdot \|M\|_\infty) = 12.37$ bits on UCI Adult ($d = 50$, $K = 390$, $\|M\|_\infty = 0.545$), 35.6 bits below the 48-bit budget; the budget check therefore passes by a wide margin and the guard's clip-and-budget composition covers every input in the validation pool. The earlier post-fix $N_q = 300$ FHE end-to-end sweep reports 0/300 overflows (Wilson 95% UB 1.26%); combined with the $n_{\text{val}} = 42{,}569$ plaintext distributional validation, we now claim that no natural input from the registered distributions evades the guard. A full FHE end-to-end run at $N_q \geq 10^4$ on a larger benchmarking host remains future work.

Generalised lesson: a named threat-model gap. FHE systems that approximate over $\mathbb{R}$ expose a silent-failure class distinct from exact-integer cryptographic schemes. OpenFHE by design does not raise on scale overflow because the output operation is numerically defined everywhere in the ciphertext space; the result is wrong. Integrators must add bounds checks at every encryption boundary and cannot rely on downstream correctness proofs to catch such errors. In particular, the *Freivalds-blind silent-failure gap*—the companion-paper Freivalds regression-integrity proof over $M \cdot \mathbf{y}$ does not catch a wrapped $\mathbf{y}$, because both sides of the proven equation are computed on the corrupt value—is a named composition-failure mode for any FHE+integrity stack that proves linearity rather than coalition-output correctness. Only the client-side sentinel audit coalitions, or a client-side plaintext recomputation, detect the failure. This observation motivates treating input guards as a cryptographic-integrity primitive, not an operational hygiene concern; any system composing CKKS arithmetic with downstream linear-consistency proofs inherits this gap and must address it at the encryption boundary.

7 Model-Specific Evaluation Circuits

CipherExplain's architecture is model-agnostic: Components 100, 300, 400, and 500 are unchanged across model classes. Only the model evaluation function f in Component 200 changes. We specify evaluation circuits for three model classes.

7.1 Logistic Regression

For $f(\mathbf{x}) = \sigma(\mathbf{w}^\top \mathbf{x} + b)$ where σ is the sigmoid, the encrypted evaluation is one plaintext-coefficient inner product plus a degree-27 Chebyshev polynomial approximation of σ (depth 8). Under the rescale-level convention (Section 2.1), the plaintext-vector inner product nominally consumes one level, but in our implementation the scale management for the inner product is fused with the first multiplicative step of the Chebyshev evaluator, so the inner product consumes no *additional* level beyond the depth-8 polynomial budget. Total model depth $L_f = 8$; pipeline depth $L_f + 2 = 10$. Fits at $N = 2^{15}$ without bootstrapping.

Multi-class. The deployed pipeline covers binary classification only. A natural three-tier extension to multi-class via SIMD packing at the coalition-output stage is sketched in Appendix F; that material is preview only, not a contribution of this paper. We do not claim multi-class deployment.

7.2 Tree Ensembles via Oblivious Evaluation (OCTE)

For tree-based models (random forests, XGBoost), we introduce the *Oblivious Coalition Tree Evaluator* (OCTE), which evaluates all 2^D tree paths simultaneously with no data-dependent branching.

Sign-gate accuracy: empirical-only certification. The sign-gate approximation error $\boldsymbol{\eta}_{\text{sign}}$ that propagates through Proposition 5 is, for the OCTE tree path, certified *empirically* on the documented validation distribution: L_∞ versus a plaintext polysign reference, the SHAP-accuracy gate, the transition-density diagnostic $\rho_{\text{transition}}$, and the realised $G_{\text{eff}} \cdot \|\boldsymbol{\eta}_{\text{sign}}\|_2$ budget at release time (Table 12). *No adversarial worst-case analytical bound on $\|\boldsymbol{\eta}_{sign}\|_2$ for sign-gate transition-zone inputs is claimed by this paper.* A deliberately worst-case input distribution that concentrates pre-activations inside the polynomial's transition window can in principle move $\boldsymbol{\eta}_{\text{sign}}$ outside the realised budget; the validation distribution is the deployed release gate, not a worst-case envelope.

Oblivious tree circuit. For a single tree with depth D, $2^D - 1$ internal nodes, and 2^D leaves:

1. *Feature extraction* (depth 0): rearrange encrypted features into the slot layout required by the tree's split structure via a Halevi-Shoup diagonal transform (same technique as BHDR). The $\sim$14 hoisted rotations per tree is a worst-case estimate assuming a full d-way gather; in practice a depth-D tree has at most $2^D - 1$ internal nodes and therefore references at most $\min(2^D - 1, d)$ distinct features (for $D = 4$, at most 15; $D = 6$, at most 63). A tree that splits on q distinct features needs a q-way diagonal transform costing $\sim 2\sqrt{q}$ hoisted rotations. We use $q \leq d$ in the complexity analysis for worst-case simplicity but the measured figure is tighter in practice.
2. *Comparison* (depth c_σ): for each internal node j, compute $c_j \approx \text{sign}(\mathsf{Enc}(x_{i_j}) - \tau_j)$ using a *tanh–Chebyshev sign surrogate*: a degree-27 Chebyshev-polynomial fit of the hyperbolic tangent $\tanh(k\,z)$ with steepness $k = 5.0$ on the bounded pre-activation range $[-B, B]$ with $B = 8.0$. These parameters are the implemented defaults in the reference implementation (the OCTE reference implementation, environment-tunable via `CE_OCTE_SIGN_STEEP` and `CE_OCTE_SIGN_DEGREE`); the $B = 8$ bound is chosen to contain the standardised-feature differences $x_{i_j} - \tau_j$ that arise from the tree-splitting thresholds observed at fit time, and $k = 5$ gives a sign-gate transition width of $\sim$0.4 standardised units around $z = 0$, narrow enough to preserve accuracy, wide enough that the Chebyshev degree-27 fit converges without Gibbs ringing. Deployed depth: $c_\sigma \approx 5$ CKKS levels. All $2^D - 1$ comparisons are independent $\rightarrow$ parallel via SIMD. A composite-minimax sign gate following Lee *et al.* [18] is present in-tree as an experimental path but was not stable enough to deploy at OCTE's target accuracy.
3. *Path indicators* (depth $\lceil \log_2 D \rceil$): for each leaf ℓ, compute $\pi_\ell = \prod_{k=1}^{D} f_k$ where $f_k = c_{j_k}$ or $1 - c_{j_k}$ depending on the branch direction (plaintext). Balanced multiplication tree: $\lceil \log_2 4 \rceil = 2$ levels for the deployed $D = 4$ configuration.
4. *Leaf aggregation* (depth 0): $\mathsf{Enc}(y_{\text{tree}}) = \sum_\ell v_\ell \cdot \mathsf{Enc}(\pi_\ell)$ (plaintext leaf values, masked rotate-and-sum).
5. *Ensemble sum* (depth 0): sum across T trees.

Depth and parameters (production config). Total model depth with the tanh–Chebyshev sign gate at $D = 4$: $L_f = c_\sigma + \lceil \log_2 D \rceil \approx 5 + 2 = 7$. Pipeline depth: $L_f + 2 = 9$. We provision the

context at $N = 2^{16}$ with ∼25–30 modulus levels to leave headroom for Rescale and EvalFastRotationExt noise. $D = 6$ **as a v2 diagnostic.** We measured $D = 6$ at $T = 100$, $K = 390$, $N = 2^{16}$. The v2 diagnostic circuit replaces the deployed tanh–Chebyshev surrogate with a Lee composite-minimax sign gate (degree-39 Chebyshev plus four X_4 refinement iterations [18], depth 23), and replaces the leaf-absorbing mux with a sequential (left-deep) path-product tree at $D - 1$ CT-CT levels followed by early decrypt and plaintext leaf aggregation; we use the sequential schedule rather than a $\lceil \log_2 D \rceil$ balanced tree here because the v2 indicator masking is computed level-by-level in concert with each comparator, so the product accumulates linearly along the path. The circuit fits the maximum-provisionable multiplicative depth 31 at $N = 2^{16}$ under `HEStd_128_classic` with zero margin: input scaling and bracketing of the standardised feature into the sign-gate domain (2 plaintext-ciphertext mults) + sign gate (23) + indicator masking (1) + sequential path-product $(D - 1 = 5) = 31$ levels. Table 12 reports the measured profile on a Hetzner CPX42 (8-core AMD EPYC-Genoa, 16 GB, 4 GB swap, swappiness 10).

Table 12: OCTE $D = 6$ v2 diagnostic, measured profile at $T = 100$, $K = 390$, $d = 50$, $N = 2^{16}$, multiplicative depth 31, single-server CPU OpenFHE on Hetzner CPX42 (8 cores, 16 GB). Lee composite-minimax sign gate, path-product tree, plaintext leaf aggregation, plaintext SHAP regression. The diagnostic run establishes accuracy feasibility, not production readiness.

Metric	$D = 6$ v2 diagnostic
Runtime (FHE forward + decrypt)	806.3 s
SHAP L_∞ vs plaintext	7.9×10^{-3}
SHAP MAE vs plaintext	3.0×10^{-3}
Efficiency-axiom error (post-redistribution)	1.0×10^{-17}
Sign flips vs polysign reference	0/390
Decrypt failures	0
Transition density $\rho_{\text{transition}}$ ($\lvert x - \tau \rvert \le 2^{-6}$)	0.411
Levels used / available (margin)	31/31 (0)
Peak resident memory	12.56 GB

The result passes the SHAP accuracy gate by approximately one order of magnitude ($L_\infty = 7.9 \times 10^{-3}$ versus the 5×10^{-2} compliance gate) with all 390 coalition signs correct and 0 decrypt failures. At $D = 6$ the binding constraint is wall-clock latency and depth margin, not accuracy or CKKS arithmetic correctness. Memory fits CPX42 with headroom (12.56 GB peak resident on a 16 GB box, swap untouched).

Where the cost goes. The Lee composite-minimax sign gate consumes 23 CKKS levels. All $2^D - 1$ internal-node comparisons are evaluated SIMD-parallel as a single depth-23 gate block, so the sign-gate contribution to the multiplicative chain is 23 levels (not $23D$). The depth-23 sign block plus input scaling/bracketing (2), indicator masking (1), and the sequential path-product tree $(D - 1 = 5)$ push the running depth to exactly 31, the maximum provisionable at $N = 2^{16}$ under standard 128-bit security. (OCTE provisions a deeper modulus chain at $N = 2^{16}$ to host the depth-31 sign-gate budget; the binary BHDR logistic-regression path uses $N = 2^{15}$ at depth $L = 10$, so the ring degree differs across model classes.) Serial CPU evaluation of the sign-gate *block* (not multiple chained sign gates) dominates wall-clock because each Chebyshev step inside the block is sequential. SHAP regression and BHDR matvec are plaintext-side in the v2 circuit and contribute negligibly. CKKS approximation error against a float-64 Lee reference is 1.5×10^{-5} flat across all six depths, well below the polynomial's theoretical $2^{-12} \approx 2.4 \times 10^{-4}$ bound. The FHE noise budget is not the gate's limiting factor. The transition-zone metric is the more informative diagnostic: $\rho_{\text{transition}} = 0.411$ on the deployed coalition-mask distribution means 41% of slot-pairs

fall inside the $|x - \tau| \leq 2^{-6}$ window where the Lee polynomial is unreliable by construction. Path-product averaging across $T = 100$ trees nonetheless damps per-slot indicator deviations to the observed SHAP $L_\infty = 7.9 \times 10^{-3}$.

Why $D = 6$ is engineering-mode rather than production. Zero remaining levels leave no margin for additional CKKS operations (e.g. a deeper sign gate, an additional rotation pass, or in-circuit BHDR regression). *Composition with the integrity layer is therefore not feasible at $D = 6$ on the current parameter set.* The Freivalds-style linear-consistency proof of the companion integrity work consumes additional CKKS depth on the sentinel verification path; with zero margin at $D = 6$, the integrity layer either requires a larger ring dimension ($N = 2^{17}$, doubling the per-operation cost), in-circuit bootstrapping, or relegation of the integrity check to a separate non-interactive proof at the cost of an extra round trip. The 806 s wall-clock is also well above the ≤ 60 s soft target for interactive compliance workflows. *Platform note:* the $D = 6$ measurement is on a Hetzner CPX42 (8-core AMD EPYC) and is not directly comparable to the $D = 4$ figures reported on the 2-vCPU Hetzner CX22 reference VM elsewhere in this paper; we chose CPX42 for $D = 6$ because $D = 4$ at $T = 100$ already saturates the CX22's 4 GB RAM and a memory-fault $D = 6$ run on CX22 would have been uninformative. $D = 6$ is accuracy-feasible but engineering-latency, depth-margin, and integrity-composition limited under the current single-server CPU OpenFHE implementation. We continue to deploy at $D = 4$.

Justified next experiment. The deployed Lee gate achieves 1.5×10^{-5} approximation error against a 5×10^{-2} SHAP compliance gate. This excess precision can be traded for depth headroom. Three candidates are: (a) a CKK-iterated Newton sign at depth $\approx$6 [19], (b) a homomorphic-select piecewise gate that bins the transition zone explicitly, and (c) a domain-restricted Chebyshev approximation with smaller pre-activation bound. Either (a) or (b), if it sustains $L_\infty \leq 5 \times 10^{-2}$ at $D = 6$, would recover several modulus levels and substantially reduce the FHE forward pass.

SIMD packing. Each coalition needs $2^D = 64$ slots for path indicators. At $N = 2^{16}$: $\lfloor 32768/64 \rfloor = 512$ coalitions per ciphertext. All $K = 390$ coalitions fit in one ciphertext. Trees are evaluated serially (one tree at a time across all coalitions); parallelizing across trees via multi-tree packing reduces circuit passes from T to $\lceil KT/512 \rceil$.

Error bound (informal claim). For the experimental composite-minimax sign-gate path at precision α, the standard accumulated-error analysis for sign-gate trees gives

$$|y_{\text{OCTE}} - y_{\text{exact}}| \;\leq\; V_{\max}\,(D + 2^D - 1) \cdot 2^{-\alpha},$$

where $V_{\max} = \max_\ell |v_\ell|$ and the factor $(D + 2^D - 1)$ counts D comparator levels (each evaluated SIMD-wide, one per tree depth) plus the $2^D - 1$ multiplicative-error accumulators across the 2^D path-indicator products and the leaf aggregation. The deployed tanh–Chebyshev surrogate carries a tighter plaintext-level error profile, but its practical accuracy is governed by the CKKS noise floor of the full pipeline rather than by this combinatorial bound; we therefore report measured prediction and SHAP error directly in Section 6.

SHAP-specific note: Coalition masking replaces features with baseline values, shifting pre-activations toward the threshold transition window $[\tau - 2^{-\alpha/2}, \tau + 2^{-\alpha/2}]$. The baseline should be chosen to avoid threshold proximity (preprocessing constraint).

8 Related Work

Encrypted inference without explanation. CryptoNets [7], GAZELLE [8], Microsoft SEAL-based engines, Zama Concrete ML, IBM HeLayers, and the NEXUS framework evaluate machine learning models under FHE to produce encrypted predictions. None provides feature-level explanations.

Multi-party computation approaches. XorSHAP [11] and related MPC approaches compute SHAP values via secret sharing or interactive multi-party protocols. These require communication between two or more non-colluding parties across multiple rounds, are vulnerable to collusion, and impose online latency that depends on network conditions. Sequential execution requires $O(K)$ rounds, but practical MPC protocols can batch coalition evaluations into fewer rounds (e.g., $O(\log K)$ rounds with sufficient parallelism), potentially achieving low single-digit seconds for $K = 390$. CipherExplain's deployed $d = 50$ end-to-end M1 latency is p50 13.4 s (Table 7); the algorithm-level narrow-d, warm-cache ablation reaches ~ 2.1 s (Table 8). Both regimes are within an order of magnitude of optimised MPC absolute timings. However, the two approaches differ qualitatively: CipherExplain is non-interactive (one client request, one server response), requires no non-collusion trust assumption, and operates with a single server rather than requiring two or more independent computation parties. These architectural properties may be decisive in deployments where establishing multiple non-colluding servers is impractical.

Differential privacy approaches. Differential privacy protected explanation methods compute SHAP values on plaintext data and add calibrated noise to limit privacy leakage. These require the data to be decrypted at the computation server, and the added noise can substantially distort attributions in the high-attribution tails that drive consequential decisions.

SHAP entropy regularization. SHAP entropy regularization techniques [12] train models whose explanations carry less identifying information, but do not compute SHAP values on encrypted data and may reduce predictive accuracy.

Output privacy, integrity, and verifiable computation on encrypted data (companion work). Deployment of the BHDR pipeline in a production setting requires two orthogonal security primitives that are the subject of companion work. For output privacy against a curious server, we refer the reader to the noise-flooding line of Li, Micciancio, Schultz and Sorrell [15] (CRYPTO 2022) and the HintLWE refinement of Ogilvie [20], plus the attack surface analysis of Guo *et al.* [16] (USENIX Security 2024). For integrity against a malicious server, the canonical path is the verifiable-computation construction of Cascudo *et al.* [22] (CRYPTO 2025), which operates natively over R_q for CKKS-like schemes; prior work of Fiore, Nitulescu, and Pointcheval [21] (PKC 2020) covers BGV/BFV but not CKKS. The Fiat–Shamir composition analysis of Attema, Fehr, and Klooß [26] (J. Cryptology 2023) applies to any amplified multi-round integrity check over our pipeline. None of this security machinery is contributed by the present paper; the reported benchmarks are for the honest-but-curious threat model.

Improved Kernel SHAP sampling. Covert and Lee [2] proposed paired and antithetic sampling for Kernel SHAP with a convergence analysis; our pair-level Matrix Bernstein argument in Theorem 6 is a direct antithetic-design adaptation of their pairing construction, and their well-conditioned-information-matrix argument is what anchors Step 2 of our proof. Musco and Witter [3]

established that $O(d \log d)$ evaluations suffice for Kernel SHAP via leverage-score sampling; this bound is *precisely* what our default $c = 2$ (i.e., $K = 2d \ln d$) is calibrated against, and we regard the sample-count contribution of this paper as downstream of theirs. What is novel here is the FHE construction: neither Covert and Lee nor Musco and Witter address the encrypted setting; both operate in plaintext and presume unrestricted access to model outputs. Neither teaches the offline pre-computation of the regression matrix M, the SIMD packing of coalitions, or the resulting reduction to $\lceil Kd/n \rceil$ ciphertext-level evaluations.

9 CI/CD Integration

CipherExplain can be integrated as a verification gate within a continuous integration and continuous deployment pipeline. In this configuration, the CI system holds a fixed encrypted test set $\{\mathsf{Enc}(\mathbf{x}_1), \ldots, \mathsf{Enc}(\mathbf{x}_T)\}$ and runs CipherExplain on each candidate model version f' before deployment.

What changes across model versions. When the model f is updated to f', the public matrices Z, $\boldsymbol{\pi}$, and M remain unchanged (they depend only on the feature dimensionality d, not on the model). The coalition prediction outputs $\mathbf{y}' = (f'(\mathbf{x}_{S_1}), \ldots, f'(\mathbf{x}_{S_K}))$ change, producing a new encrypted SHAP vector $\mathsf{Enc}(\hat{\boldsymbol{\phi}}')$ for each test instance.

Stability and fairness bounds. The CI gate computes two metrics on the encrypted attributions (without decrypting the test inputs):

1. *Attribution stability:* for each test instance, the maximum element-wise change $\max_i |\hat{\phi}'_i - \hat{\phi}_i|$ between the candidate and the deployed model's attributions. The gate fails if any instance exceeds a threshold τ_s (e.g., $\tau_s = 0.1$), flagging cases where the model's reasoning has shifted substantially.
2. *Fairness consistency:* for protected features (e.g., age, gender), the mean absolute attribution across the test set. A direct approach fails the gate if the mean attribution exceeds an absolute threshold τ_f, but the appropriate τ_f depends on the base rate of the protected attribute's legitimate correlation with the target, a threshold that is too tight rejects valid model updates, while too loose misses discriminatory changes. A more robust alternative uses a *relative* threshold: the gate fails if the *change* in mean attribution for a protected feature between model versions exceeds $\Delta\tau_f$ (e.g., $\Delta\tau_f = 0.05$), flagging cases where the model's reliance on a protected feature has increased rather than testing against an absolute level. Setting either threshold requires domain-expert calibration informed by the regulatory context and the dataset's demographic distribution.

These checks are performed client-side after decryption by a CI service that holds the FHE secret key but does not have access to the plaintext test data (which was encrypted by a separate data custodian). The CI service observes only the decrypted attributions and predictions, not the raw feature values.

Homomorphic stability verification. Interestingly, the stability comparison itself can be performed *homomorphically* without decrypting the test inputs. Since M is public and fixed across model versions, comparing $\mathsf{Enc}(\hat{\boldsymbol{\phi}}_{\text{old}})$ and $\mathsf{Enc}(\hat{\boldsymbol{\phi}}_{\text{new}})$ reduces to comparing two encrypted vectors. The element-wise difference $\mathsf{Enc}(\hat{\boldsymbol{\phi}}_{\text{new}} - \hat{\boldsymbol{\phi}}_{\text{old}})$ is computable homomorphically (one addition, no depth). An L_∞ bound check can be approximated via a homomorphic polynomial approximation of $\max(|x|, \tau)$, though this adds multiplication depth. A simpler alternative is to decrypt only the

difference vector $\hat{\phi}_{\text{new}} - \hat{\phi}_{\text{old}}$ (which reveals attribution *changes* but not absolute values), a weaker privacy disclosure than decrypting the attributions themselves.

Limitations. The stability check detects changes in *explained* model behavior but relies on the honest-but-curious assumption: a malicious model operator could deploy a different model than the one tested in CI. This limitation is the same integrity gap discussed in the companion output-privacy work and Appendix B.

10 Conclusion and Future Work

We have presented *BHDR*, a single-server non-interactive construction that performs the server-side coalition evaluation and weighted regression of Kernel SHAP under CKKS FHE for the deployed logistic-regression path. The pipeline is *implemented end-to-end* for logistic regression (with the encryption-boundary input guard of Section 6.7): 2.1 s algorithm-level latency on Apple M1 and p50 13.4 s, p95 16.3 s, p99 24.2 s end-to-end on the OpenFHE backend at $d = 50$ over $N_q = 300$ UCI Adult queries (Section 6.6). The $\sim$43$\times$ rotation reduction from BSGS-hoisted diagonal matvec with K'-periodic replicate encoding is validated empirically; the pair-level Matrix Bernstein analysis gives an $O(R\sqrt{\log d \cdot \log(d/\delta)/K})$ approximation bound for the i.i.d. antithetic-pair sampler with the closed-form $\lambda_{\min}^{\text{sz}}(\bar{A}) = 1/(2H_{d-1})$. The deployed deterministic stratified-ramp sampler at $K = 2d \ln d$ is empirically certified through release-time relative-invertibility and approximation-quality gates; a quasi-Monte-Carlo concentration argument that would license this default theoretically remains open work. An Oblivious Coalition Tree Evaluator provides a circuit-depth feasibility study for tree ensembles at $D = 4$, $T = 100$ ($\sim$53 s on the Hetzner CX22 reference VM). At $D = 6, T = 100, K = 390$ a v2 diagnostic circuit (path-product, Lee composite-minimax sign gate at depth 23, $N = 2^{16}$, multiplicative depth 31) passes the SHAP accuracy gate at $L_\infty = 7.9 \times 10^{-3}$ but runs at 806 s on 8-core Hetzner CPX42 with zero remaining CKKS levels (Section 7.2). $D = 6$ is accuracy-feasible but engineering-latency and depth-margin limited under the current single-server CPU OpenFHE implementation. We deploy at $D = 4$. A lower-depth CKK-iterated or Newton sign gate [19] is the justified next experiment for recovering wall-clock and depth headroom.

Priority open problems.

1. *Multi-class classification.* The implemented pipeline is binary-only. A three-tier extension (Section 7, multi-class paragraph) covers $C = 2$ in the current build and describes a SIMD-packed Tier 2 for $C \in \{2, 3, 4\}$ and a preview Tier 3 for $C \in \{5, \ldots, 10\}$. Tier 2 and Tier 3 build-out and end-to-end benchmarks are deferred to a v3.1 release.
2. *Deeper OCTE trees.* v3 caps OCTE at $D = 4$ with $T = 100$. ECOA credit-scoring profiles (FICO and VantageScore equivalents) operate at $D = 4$–5, so $D = 4$ is the v3 product, not a compromise. The $D = 6$ ceiling depends on the sign-gate choice. The measured v2 Lee composite-minimax circuit (Section 7.2) passes the SHAP accuracy gate at $L_\infty = 7.9 \times 10^{-3}$, with 0/390 sign flips and 0 decrypt failures, but runs for 806 s on CPX42 and consumes all 31 available multiplicative levels. Earlier lower-depth tanh–Chebyshev configurations leave nominal depth headroom but do not represent the measured v2 accuracy profile. The open engineering problem is therefore a lower-depth sign gate that preserves the $D = 6$ accuracy margin while restoring runtime and depth headroom. $D = 6$ targets fraud detection and is out of v3 scope. $D \in \{6, 8\}$ is deferred to v4 and gated on a boundary-robust sign approximator: candidates include a composite-minimax sign gate at $\alpha \geq 10$ with explicit precision near split thresholds, a domain-restricted Chebyshev approximation with smaller B reflecting the

realistic standardised feature range, or an explicit margin guard that detects coalition-masked inputs inside the sign transition band. Bootstrapping is orthogonal to this fix: it would expand the depth budget for higher-precision gates but does not address sign-approximation error in the transition zone directly. CPU-only $D = 6$ on the current host is unshippable as a research artifact.

3. *MLP support.* A polynomial-activation (PANCE-style) circuit for multi-layer perceptrons is sketched in preliminary work but is neither deployed nor benchmarked end-to-end under BHDR; we defer it to future work.
4. *GPU acceleration.* Porting the OpenFHE backend to a GPU target (e.g. the CAT framework, arXiv:2503.22227) would push OCTE into sub-second territory per upstream microbenchmarks.
5. *Two-pass adaptive coalition allocation.* A first pass with $K/3$ coalitions, followed by DP feature selection ($\varepsilon = 2$–3) and a focused second pass, can improve top-t SHAP variance by 1.3–3× at the cost of one additional round trip and ∼40% latency overhead.
6. *Deterministic-sampler theory.* The present paper licenses the deployed sampler through the realised-design release-time engineering certificate of Proposition 5 and Proposition 7. A closed-form concentration theorem deriving the engineering-certificate values for the exact $K = 2d \ln d$ stratified-ramp sampler remains open (Appendix 5.6 surveys candidate paths). The difficulty is boundary coverage: at $d = 50$, the deployed ramp samples only about four singleton coalitions in the primary half, so most features receive no singleton row even though boundary coalitions carry the largest Kernel SHAP weights (Appendix D). Candidate theory paths include: (i) a boundary-complete variant that deterministically includes all singleton/co-singleton rows before the stratified ramp, giving a seed-independent invertibility floor but not by itself recovering the $1/\log d$ i.i.d. eigenvalue scale; (ii) a boundary-complete plus leverage-score-weighted ramp adapting Musco–Witter-style Kernel SHAP sample-complexity analysis to the antithetic weighted design; and (iii) direct bounds on $G_{\text{eff}} = \|P_\phi M P_y\|_{2\to\infty}$ for the structured realised matrix. The current deployment does not depend on any of these results: each release is certified directly by measuring $\lambda_{\min}^{\text{sz}}$, G_{eff}, finite-K approximation error, and implementation error.

Certified pipeline framing. The deployed pipeline is governed by Proposition 5 plus Proposition 7, a certified deterministic regression-stability guarantee with four certificate levels containing five scalar certificate values. Two values ($\lambda_{\min}^{\text{sz}}$, G_{eff}) are deterministic functions of the realised public design and are exact once measured on the build artefact. Three values (max sampling error, p99 sampling error, CKKS FHE-vs-plaintext error) are empirical and valid under documented validation distributions. These empirical certificates are valid for the documented validation distribution and should be rerun when model class, feature distribution, or baseline policy changes. Together, the five scalar certificates plus the deterministic bound convert the BHDR pipeline from an optimised implementation into a *certified encrypted explanation pipeline* whose attribution-error envelope is verifiable per release.

Diagnostic refinements during preparation. Two diagnostic refinements during the preparation of this manuscript informed this framing. The Kernel SHAP convergence rate is empirically dominated by the deterministic stratified-ramp sampler outperforming the i.i.d. analytical surrogate by 1–2 orders of magnitude; this paper reflects this through the Proposition 5 plus Proposition 7 certified-deterministic guarantee, with the i.i.d. analysis (Theorem 6) retained as analytical baseline. The OCTE $D = 6$ ceiling depends on the sign-gate choice: the v2 Lee composite-minimax

configuration passes the SHAP accuracy gate at $L_\infty = 7.9 \times 10^{-3}$ with 0/390 sign flips and 0 decrypt failures, but runs for 806 s on CPX42 and consumes all 31 available multiplicative levels (margin 0). Earlier lower-depth tanh–Chebyshev configurations leave nominal depth headroom but do not represent the measured v2 accuracy profile; the binding $D = 6$ constraint is therefore the sign-gate latency/depth trade-off, not accuracy alone. The regression matrix is contractive on the centered subspace ($G_{\text{eff}} = 0.236$ on the padded BHDR working grid, 0.287 on the realised $K = 390$ sampler, far from the diagnostic $\kappa(M) \approx 2 \times 10^{15}$ that earlier drafts had treated as a noise-amplification factor). The certificate framing absorbs both corrections: G_{eff} and the CKKS-error certificate are model-class-agnostic, and the sign-error term enters Proposition 5 as a separate $\boldsymbol{\eta}_{\text{sign}}$ component bounded per Section 7.2.

The separate questions of output privacy (IND-CPA$^{\text{D}}$ calibration), integrity against a malicious server (Freivalds-style linear-consistency proofs, sentinel audits, verifiable-FHE binding), and the regulated-mode efficiency-vs-ranking-stability trade-off are addressed in companion work.

Acknowledgments

The author thanks the OpenFHE project maintainers, whose library made the empirical sections of this paper possible.

The work presented, including theoretical formulations and cryptographic constructions, is original, except for language refinements where ChatGPT and Claude were used. These refinements do not influence the technical contributions or the core results of the paper.

Disclosure

Patent notice. The system and method described in this paper are related to pending PCT patent application *PCT/IB2026/053405*, "System and Method for Computing Shapley Additive Explanations Under Fully Homomorphic Encryption Using Compressed Coalition Sampling," with priority date 7 April 2026, held by VaultBytes Innovations Ltd.

The pending application concerns privacy-preserving computation of Shapley additive explanations under fully homomorphic encryption, including an integrated encrypted-explanation pipeline, offline precomputation and reuse of public Kernel SHAP regression artefacts, SIMD packing of coalition inputs, and homomorphic aggregation and matrix–vector evaluation techniques.

The BHDR construction and K'-periodic replicate encoding described in this paper are disclosed here as engineering optimisations of the broader encrypted Kernel SHAP pipeline. The copyright license selected for this ePrint publication applies to the paper as a scholarly work, including its text, figures, and tables, subject to the terms of that license. Nothing in that copyright license, and nothing in publication of this paper, should be construed as granting any patent license, trademark license, commercial implementation license, or other right to practice patented or patent-pending methods. Implementation or commercial use of patented or patent-pending methods may require a separate license from VaultBytes Innovations Ltd.

References

[1] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[2] I. Covert and S.-I. Lee, "Improving KernelSHAP: Practical Shapley value estimation using linear regression," in *Proc. AISTATS*, 2021.

[3] C. Musco and R. T. Witter, "Provably accurate Shapley value estimation via leverage score sampling," in *Proc. Int. Conf. on Learning Representations (ICLR)*, 2025 (Spotlight); arXiv:2410.01917; OpenReview forum `wg3rBImn3O`.

[4] J. H. Cheon, A. Kim, M. Kim, and Y. Song, "Homomorphic encryption for arithmetic of approximate numbers," in *Advances in Cryptology – ASIACRYPT 2017*, Springer, 2017, pp. 409–437.

[5] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "(Leveled) fully homomorphic encryption without bootstrapping," in *Proc. ITCS*, 2012.

[6] J. Fan and F. Vercauteren, "Somewhat practical fully homomorphic encryption," *IACR Cryptol. ePrint Arch.*, 2012/144, 2012.

[7] R. Gilad-Bachrach, N. Dowlin, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, "CryptoNets: Applying neural networks to encrypted data with high throughput and accuracy," in *Proc. ICML*, 2016.

[8] C. Juvekar, V. Vaikuntanathan, and A. Chandrakasan, "GAZELLE: A low latency framework for secure neural network inference," in *Proc. USENIX Security*, 2018.

[9] European Parliament and Council, "Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act)," *Official Journal of the European Union*, 2024.

[10] T. T. Nguyen, T. T. Huynh, Z. Ren, T. T. Nguyen, P. L. Nguyen, H. Yin, and Q. V. H. Nguyen, "A survey of privacy-preserving model explanations: Privacy risks, attacks, and countermeasures," *arXiv preprint arXiv:2404.00673*, 2024.

[11] D. Jetchev and M. Vuille, "XorSHAP: Privacy-preserving explainable AI for decision tree models," *IACR Communications in Cryptology*, vol. 1, no. 4, 2025; ePrint 2023/1859.

[12] D. P. Sharma, X. Sun, L. Xue, X. Lin, and P. Xiong, "Privacy-preserving explainable AIoT application via SHAP entropy regularization," *arXiv preprint arXiv:2511.09775*, 2025.

[13] A. Al Badawi, J. Bates, F. Bergamaschi, D. B. Cousins, S. Erabelli, N. Genise, S. Halevi, H. Hunt, A. Kim, Y. Lee, Z. Liu, D. Micciancio, I. Quah, Y. Polyakov, R. V. Saraswathy, K. Rohloff, J. Saylor, D. Suponitsky, M. Triplett, V. Vaikuntanathan, and V. Zucca, "OpenFHE: Open-source fully homomorphic encryption library," in *Proc. ACM CCS Workshop on Encrypted Computing and Applied Homomorphic Cryptography (WAHC)*, 2022.

[14] S. Halevi and V. Shoup, "Algorithms in HElib," in *Advances in Cryptology – CRYPTO 2014*, Springer, 2014, pp. 554–571; ePrint 2014/106.

[15] B. Li, D. Micciancio, M. Schultz, and J. Sorrell, "Securing approximate homomorphic encryption using differential privacy," in *Advances in Cryptology – CRYPTO 2022*, Springer, 2022; ePrint 2022/816.

[16] Q. Guo, D. Nabokov, E. Suvanto, and T. Johansson, "Key recovery attacks on approximate homomorphic encryption with non-worst-case noise flooding countermeasures," in *Proc. USENIX Security*, 2024.

[17] J. A. Tropp, "An introduction to matrix concentration inequalities," *Foundations and Trends in Machine Learning*, vol. 8, no. 1–2, pp. 1–230, 2015; arXiv 1501.01571.

[18] E. Lee, J.-W. Lee, J.-S. No, and Y.-S. Kim, "Minimax approximation of sign function by composite polynomial for homomorphic comparison," *IEEE Trans. Dependable and Secure Computing*, vol. 19, no. 6, pp. 3711–3727, 2022; ePrint 2020/834.

[19] J. H. Cheon, D. Kim, D. Kim, H. H. Lee, and K. Lee, "Numerical method for comparison on homomorphically encrypted numbers," in *Advances in Cryptology – ASIACRYPT 2019*, LNCS vol. 11922, pp. 415–445, 2019; ePrint 2019/1234.

[20] T. Ogilvie, "IND-CPA-D and KR-D security with reduced noise from the HintLWE problem," in *Advances in Cryptology – ASIACRYPT 2025*; ePrint 2025/1618.

[21] D. Fiore, A. Nitulescu, and D. Pointcheval, "Boosting verifiable computation on encrypted data," in *Proc. PKC*, 2020; IACR ePrint 2020/132.

[22] I. Cascudo, A. Costache, D. Cozzo, D. Fiore, A. Guimarães, and E. Soria-Vazquez, "Verifiable computation for approximate homomorphic encryption schemes," in *Proc. CRYPTO*, 2025. IACR ePrint 2025/286.

[23] R. Freivalds, "Fast probabilistic algorithms," in *Proc. MFCS*, 1979.

[24] B. Ghaleb and W. J. Buchanan, "Side channel analysis in homomorphic encryption," *arXiv preprint arXiv:2505.11058*, 2025.

[25] M. Duparc and M. Taha, "Improved NTT and CRT-based RNR blinding for side-channel and fault resistant Kyber," *IACR Cryptol. ePrint Arch.*, 2025/181.

[26] T. Attema, S. Fehr, and M. Klooß, "Fiat–Shamir transformation of multi-round interactive proofs (extended version)," *Journal of Cryptology*, vol. 36, art. 36, 2023; ePrint 2021/1377.

A Algorithm Pseudocode

Algorithm 1 CipherExplain: Compressed SHAP under FHE

Require: Encrypted input $\mathsf{Enc}(\mathbf{x})$, public model f, dimensionality d, precision c, per-rotation costs c_1 (`EvalFastRotationExt`, hoisted baby-step), c_2 (`EvalRotate`, unhoisted giant-step), with $c_2/c_1 \approx 2.7$ on OpenFHE 1.2.x

Ensure: Encrypted SHAP vector $\mathsf{Enc}(\hat{\boldsymbol{\phi}})$, encrypted prediction $\mathsf{Enc}(\hat{y})$

— **Offline (once per d)** —

$K \leftarrow \max(\lfloor c \cdot d \cdot \ln(d) \rfloor, 2d)$, rounded down to the nearest even integer ▷ matches Section 3.1

$K' \leftarrow 2^{\lceil \log_2 K \rceil}$ ▷ next power of $2 \geq K$ with $K' \mid n$

$Z \leftarrow \textsc{StratifiedRampAntithetic}(K, d, \text{seed}{=}42)$ ▷ $K \times d$ binary matrix; quasi-random stratified ramp with antithetic complement pairing (Section 3.1)

$\boldsymbol{\pi} \leftarrow \textsc{ShapleyKernel}(Z)$ ▷ K-vector of kernel weights

$W \leftarrow \mathrm{diag}(\boldsymbol{\pi})$

$M \leftarrow (Z^\top W Z)^{-1} Z^\top W$ ▷ $d \times K$ regression matrix

Pad M with $K'-K$ zero columns ▷ $d \times K'$ for BHDR

Choose integer (r_1, r_2) with $r_1 r_2 \geq K'$, rounded from the continuous optimum $(\sqrt{K' c_2/c_1},\ \sqrt{K' c_1/c_2})$ to an implementation-admissible split (powers of two for the deployed OpenFHE rotation-key grid). The deployed grid uses $(r_1, r_2) = (32, 16)$. ▷ asymmetric BSGS split, integer/admissible

— **Online (per query)** —

for $k = 1, \ldots, K$ **do** ▷ Component 200: Coalition masking

  $\mathsf{Enc}(\mathbf{x}_{S_k}) \leftarrow \mathbf{z}_k \odot \mathsf{Enc}(\mathbf{x}) \oplus (\mathbf{1} - \mathbf{z}_k) \odot \mathsf{Enc}(\mathbf{b})$

end for

Pack coalitions into $\lceil Kd/n \rceil$ ciphertexts ▷ Component 300; $n = N/2$ slots

for each packed ciphertext $\mathbf{c}_j$ **do**

  $\mathsf{Enc}(\mathbf{y}_j) \leftarrow f(\mathbf{c}_j)$ ▷ Homomorphic model evaluation

end for

$\mathsf{Enc}(\hat{y}) \leftarrow f(\mathsf{Enc}(\mathbf{x}))$ ▷ Standard encrypted inference

$\mathsf{Enc}(y_0) \leftarrow f(\mathsf{Enc}(\mathbf{b}))$ if $\mathbf{b}$ is encrypted; otherwise plaintext-encode $f(\mathbf{b})$ as a constant ciphertext ▷ Baseline evaluation for centering (Section 3.4)

$\mathsf{Enc}(\mathbf{y}) \leftarrow \textsc{Gather}(\mathsf{Enc}(\mathbf{y}_1), \ldots)$ ▷ Consolidate coalition outputs

$\mathsf{Enc}(\tilde{\mathbf{y}}) \leftarrow \mathsf{Enc}(\mathbf{y}) - \mathsf{Enc}(y_0)$ ▷ Center each coalition output by the baseline

$\mathsf{Enc}(\tilde{\mathbf{y}}_{\text{pad}}) \leftarrow \textsc{ReplicateTile}(\mathsf{Enc}(\tilde{\mathbf{y}}), K', n)$ ▷ $\log_2(n/K')$ rotate-and-double (5 for n=16384, K'=512)

$\mathsf{Enc}(\hat{\boldsymbol{\phi}}) \leftarrow \mathrm{BHDR}(M, \mathsf{Enc}(\tilde{\mathbf{y}}_{\text{pad}}), r_1, r_2)$ ▷ BSGS-hoisted diagonal, 51 rotations total, 1 depth level

return $\mathsf{Enc}(\hat{\boldsymbol{\phi}})$, $\mathsf{Enc}(\hat{y})$ ▷ Component 500

— **Client-side post-decryption (optional)** —

Client decrypts $\mathsf{Enc}(\hat{\boldsymbol{\phi}})$ and $\mathsf{Enc}(\hat{y})$, computes residual $r \leftarrow \hat{y} - \phi_0 - \sum_i \hat{\phi}_i$, and optionally applies plaintext residual redistribution $\hat{\phi}_i \leftarrow \hat{\phi}_i + r \cdot |\hat{\phi}_i| / \sum_j |\hat{\phi}_j|$ (Section 3.4). This step runs on plaintext attributions; it is *not* a homomorphic operation and is intentionally outside the server-side BHDR pipeline.

B Client-Side Verification

Scope of this appendix. This appendix sketches a deployable client-side verification layer (LCV-Public/LCV-Private) as a deployment hook for readers who need *some* integrity guarantee against an actively malicious server. The headline performance and approximation claims of the paper do *not* depend on this layer; the formal cryptographic analysis (binding, knowledge-soundness, full malicious-server threat-model coverage) is the scope of the integrity companion paper. All parameter choices below ($\varepsilon_{\text{check}}$, λ, proof sizes) are reported as measured operational values from the reference implementation, not as values whose security has been proven against a fully malicious adversary in the present paper.

After decryption, the client verifies the Shapley efficiency axiom:

$$\left| \sum_{i=1}^{d} \hat{\phi}_i + \phi_0 - \hat{y} \right| < \tau$$

where τ is a tolerance consistent with the floating-point precision of the FHE scheme. This check detects catastrophic numerical failure during ciphertext transport, rescaling, or modulus switching.

Limitations against a malicious server. The efficiency axiom check does *not* provide integrity against an actively malicious computation server. A malicious operator could construct an encrypted explanation $\mathsf{Enc}(\boldsymbol{\phi}')$ that satisfies $\sum_i \phi'_i + \phi_0 = \hat{y}$ (i.e., passes the axiom check) while arbitrarily redistributing attribution weight across features. For example, suppressing a legally relevant feature's attribution and redistributing its weight to innocuous features. This attack requires the server to perform homomorphic computations that produce a ciphertext decryptable to the desired values, which is possible because the server controls the computation graph.

For regulatory compliance use cases (EU AI Act, ECOA) where the model operator has an incentive to conceal discriminatory attributions, this attack surface is material. Mitigation approaches include:

- *Layered Commit-and-Verify (LCV, model-public setting):* The server returns $\mathsf{Enc}(\mathbf{y})$ (coalition outputs) alongside $\mathsf{Enc}(\hat{\boldsymbol{\phi}})$. The client decrypts both and verifies $M \cdot \mathbf{y} \approx \hat{\boldsymbol{\phi}}$ in plaintext and recomputes all K model evaluations. Detection probability is 1 (deterministic).
- *LCV-Private (model-private setting):* The server returns $\mathsf{Enc}(\mathbf{z})$ (pre-activations) in addition, publishes a Pedersen vector commitment to the model weights C_w, and provides λ inner-product argument proofs against batched ternary challenges (Schwartz–Zippel) bound to a client nonce. Detection probability is $\geq 1 - 2^{-\lambda}$ for perturbations $\|\boldsymbol{\delta}\|_2 \geq \Delta_{\min}$.
- *Independent auditor:* An auditor holding a copy of the model verifies explanations on selectively decrypted samples.

LCV noise-bound table. The ternary anti-concentration tolerance ϵ_{check} depends on the noise-correlation assumption. Three bounds apply:

Table 13: LCV-Private noise bounds (K=390, $\epsilon_{\text{CKKS}} = 1.35 \times 10^{-4}$, $\lambda_{\text{sz}} = 24$).

Bound	ϵ_{check}	$\Delta_{\min}$	Source
Conservative ($K \cdot \epsilon_{\text{CKKS}}$)	5.27×10^{-2}	0.34	worst-case fully correlated
Independent ($\sqrt{K} \cdot \epsilon_{\text{CKKS}}$)	2.67×10^{-3}	0.017	uncorrelated-noise Rademacher
Empirical (measured)	2.78×10^{-10}	~ 0.001	Monte-Carlo over breast-cancer

Operational choice: independent bound ($\epsilon_{\text{check}} = 2.67 \times 10^{-3}$, $\Delta_{\min} = 0.017$). Conservative against noise-subspace adversaries — empirical is optimistic for adversarial $\boldsymbol{\delta}$ aligned with CKKS noise; conservative is unnecessarily loose (measured correlation is moderate, mean 0.27, not worst-case).

LCV latency (measured). Client-side overhead on M1 (10-run mean). IPA figures are reproduced from the integrity-layer companion work; the construction is a classical inner-product-argument ZK protocol over the BLS12-381 scalar field.

Table 14: LCV latency (ms) — spec estimate vs measured. IPA verify timing reports the batched verifier path using public-input variable-time MSM over Fiat–Shamir challenge scalars (see "MSM scalar provenance" below).

Metric	Spec estimate	Measured
LCV-Public client	16 ms	131 ms*
LCV-Private client (non-IPA layers)	165 ms	179 ms
IPA prove (server, $\lambda_{\text{sz}} = 24$)	192 ms	335 ms
IPA verify (client, $\lambda_{\text{sz}} = 24$)	120 ms	23–68 ms range; 37 ms mean
IPA proof size, $\lambda_{\text{sz}} = 24$	$\sim$12 KB	10.5 KB
IPA prove (server, $\lambda_{\text{sz}} = 32$)	—	$\sim$447 ms (projected, not yet validated)
IPA verify (client, $\lambda_{\text{sz}} = 32$)	—	$\sim$49 ms mean (projected, not yet validated)

Spec-estimate provenance. The "Spec estimate" column reports pre-implementation engineering estimates derived from scalar-multiplication counts and CKKS decryption counts under a single-thread Apple M1 reference. Measured values are authoritative; the spec column is included only to surface where engineering estimates over- or under-counted real costs.

*The 16 ms spec estimate for LCV-Public underestimated the cost of three CKKS decryptions (which dominate the 131 ms measured figure at $\sim$120 ms combined); batching into a single packed ciphertext reduces this to $\sim$40 ms, closer to spec.

MSM scalar provenance. The IPA verifier uses one batched multi-scalar multiplication (MSM) across all λ proofs. In the current implementation this MSM is variable-time. This is acceptable because the verifier's scalars are derived from public Fiat–Shamir challenges over public commitments and public input, with no witness-derived material. Prover-side MSMs that depend on witness material use constant-time scalar multiplication. Appendix C documents which MSM paths are constant-time, which are public-input variable-time, and which are excluded from the threat model.

Overhead frame. We report two consistent frames rather than mixing them: from the client's wall-clock perspective, the IPA verifier (37 ms mean) adds $\sim$0.28% on top of the 13.4 s p50 end-to-end SHAP latency at $d = 50$; from the server pipeline's perspective, the IPA prover (335 ms) adds $\sim$0.63% on top of the $\sim$53 s OCTE pipeline at $D = 4$, $T = 100$. We deploy $\lambda_{\text{sz}} \in \{24, 32\}$ in production per the main-text soundness discussion; the table reports $\lambda_{\text{sz}} = 24$ for direct comparison against the original spec, and $\lambda_{\text{sz}} = 32$ figures below as a projected linear-scaling extrapolation, not yet validated.

The current system provides confidentiality against an honest-but-curious server. Integrity against a malicious server is provided by LCV with the stated probabilistic bound.

C Side-Channel Hardening

The BHDR pipeline executes a *data-independent operation schedule at the protocol level*: the rotation pattern and the OCTE oblivious tree circuit issue identical sequences of CKKS operations regardless of the encrypted input. This is a structural property of the protocol, not a constant-time guarantee at the arithmetic level. The remaining side-channel risk is at the *arithmetic* level: OpenFHE's Barrett modular reduction and NTT contain data-dependent branches [24], and constant-time guarantees there require build-flag and source-level hardening (Priorities 1–2 below) that vary with compiler and microarchitecture.

Scope of the constant-time claim. The scope is per-query rotation-count uniformity only. We do not claim constant-time OpenFHE internals. OpenFHE's internal NTT schedule and key-switching costs depend on noise magnitude. They remain out of scope per the disclaimer in `cipherexplain/core/constant_time.py`. Under a dedicated bare-metal deployment we treat microcode injection, DDR5 bus access, and SMT co-residency as out-of-scope physical or tenant-access classes. SMT co-residency is mitigated by disabling HyperThreading and by single-tenant allocation on Hetzner AX. DDR5 bus interposition requires physical access to the host, which the remote-adversary threat model rules out.

Priority 1: Constant-time Barrett reduction. Replace every conditional extra-reduction step (`if (a >= q) a -= q;`) with a branchless select (`a -= q & -(a >= q);`). Audit `src/core/include/math/hal/intnat/ubintnat.h` and related files. Estimated: ∼50 lines changed, ∼2 days.

Priority 2: Build flags. Compile with `-O2 -fno-if-conversion` (not `-O3`). The `-O3` flag enables if-conversion optimizations that can reintroduce branches in hardened code [24]. Do not use link-time optimization (`-flto`).

Priority 3: Process and cache isolation. For cloud deployment: dedicated physical cores (`taskset`), disabled SMT/HyperThreading, Intel CAT L3 cache partitioning. For bare-metal: process affinity suffices.

Priority 4: Blinded NTT. RNS-blinding [25] randomizes twiddle-factor multiplications, defeating single-trace power/EM attacks. Cost: 2 extra polynomial multiplications per NTT call. Overhead scales with the model's total NTT count: for logistic regression with BHDR (∼500 NTT calls): ∼1–2 s on M1. For OCTE tree ensembles (∼thousands of NTT calls across 100 trees at $N = 2^{16}$): ∼10–20 s additional. Deploy only when the threat model requires physical or co-tenant protection.

Priority 5: Fixed-size network responses. All CKKS ciphertexts at a given parameter set have fixed serialized size. Pad auxiliary data (IPA proofs, metadata) to a fixed maximum. Use a constant-time response delay (model-aware target exceeding worst-case pipeline time).

IPA MSM scalar-provenance audit. The LCV-IPA layer (Appendix B) uses two distinct MSM paths with different timing properties; we list them in Table 16 so the side-channel surface is fully enumerated.

Table 15: Side-channel hardening priority and effort.

Priority	Measure	Effort	Threat mitigated
1 (Critical)	Constant-time Barrett	2 days	Software timing
2 (High)	Build `-O2 -fno-if-conversion`	0.5 day	Compiler reintroduction
3 (High)	Process/cache isolation	1 day	Cross-tenant cache
4 (Medium)	Blinded NTT	3 days	Power/EM single-trace
5 (Medium)	Fixed-size responses	1 day	Network inference

Table 16: IPA multi-scalar-multiplication paths in `lcv-ipa-core`. Constant-time paths use `multiscalar_mul` on Ristretto/curve25519-dalek; variable-time paths use `vartime_multiscalar_mul`.

Role	Timing class	Justification
Prover folding MSM	constant-time	Scalars are witness-derived (a_i, b_i, blinding); leakage of timing reveals committed values
Single-proof verifier MSM	constant-time	Defensive: verifier can choose either, current code uses constant-time
Batch verifier MSM	public-input variable-time	Scalars are public Fiat–Shamir challenges over public commitments and public input; no witness material; acceptable under the threat model

D On the Analytical License for the Deployed Sampler (Open Work)

The deployment is licensed by Proposition 5 plus Proposition 7—a realised-design certificate framework that measures four numerical quantities per release and verifies each against documented thresholds. A natural question is whether the deployed deterministic stratified-ramp sampler at $K = 2d \ln d$ admits an asymptotic concentration theorem that derives certificate values rather than measures them. We summarise here why this remains open and identify candidate paths for future work.

D.1 The boundary-coverage obstacle

The Kernel SHAP weight $w(s) = (d-1)/\binom{d}{s}s(d-s)$ places most weight on boundary sizes $s = 1$ and $s = d-1$, because the $\binom{d}{s}^{-1}$ factor is exponentially small at intermediate s. The deployed sampler visits each size $s \in \{1, \ldots, d-1\}$ approximately $m_s = K/(2(d-1)) \approx \ln d$ times in the primary half, with within-stratum subsets drawn uniformly under a public seed.

At $s = 1$, the deployed sampler draws $m_1 \approx \ln d$ singleton coalitions uniformly from the d possible singletons. The probability that a given feature i receives at least one singleton row in the primary half is $1 - (1 - 1/d)^{m_1}$; the expected number of features covered by singleton rows is

$$d \cdot \left(1 - \left(1 - \tfrac{1}{d}\right)^{m_1}\right) \;\approx\; 3.9 \quad \text{at } d = 50,\, m_1 = 4.$$

Thus approximately 46 of 50 features receive no singleton row at the production seed (and analogously at the $s = d-1$ stratum). Without a singleton row for feature i, the i-th eigendirection of

$P_0 Z^\top W Z P_0$ is supported only by intermediate-size contributions whose kernel weights are exponentially smaller than $w(1)$. A clean closed-form lower bound $\lambda_{\min}^{\mathrm{sz}} \geq c/\log d$ for the *exact* deployed sampler is therefore not derivable from boundary-coverage arguments alone.

Despite this gap, the realised $\lambda_{\min}^{\mathrm{sz}}$ at the production seed passes Proposition 7's level-1 certificate across the deployed dimensionalities. This is possible because intermediate-size strata still contribute information to the projected information matrix, even though their individual kernel weights are small; the certificate measures the realised aggregate effect directly, and does not depend on boundary coverage being complete. The certificate framing therefore licenses the deployment without requiring an asymptotic theorem.

D.2 Candidate paths for an asymptotic theorem

We identify three candidate paths for a future closed-form theorem; none is developed in this paper, and the deployment does not depend on any of them.

Path 1 — Boundary-complete stratified ramp. Modify the sampler to deterministically include all $2d$ singleton/co-singleton coalitions $\{\mathbf{e}_i, \mathbf{1} - \mathbf{e}_i\}_{i=1}^{d}$ before filling the remaining $K - 2d$ rows with the existing stratified ramp. Under the deployed $\sum \pi_k = 1$ normalisation, the boundary block contributes a deterministic invertibility floor of order $1/(dH_{d-1})$ on the sum-zero subspace. At $d = 50$, this raises the raw coalition count from 390 to 490, still below the current padded $K' = 512$; thus BHDR rotation count and K'-padding need not increase, although coalition masking and model evaluation grow by approximately 25%. The boundary-complete sampler removes the seed-dependence of certificate (1) but does not by itself recover the i.i.d. baseline's $1/\log d$ eigenvalue scale; an additional per-stratum concentration argument is required for that.

Path 2 — Boundary-complete + leverage-score-weighted ramp. Combine the boundary-complete prefix with leverage-score weighted within-stratum sampling, adapting the Musco–Witter [3] sample-complexity analysis to the kernel-weighted Kernel SHAP design. Musco–Witter show that $O(d \log d)$ coalitions suffice for ε-approximation under leverage-score sampling in the unweighted setting; the boundary-complete prefix provides deterministic invertibility while the leverage-score ramp targets the directions that matter most for the regression. This is the most promising path because it connects to existing Kernel SHAP sample-complexity results, but adapting Musco–Witter to the antithetic, weighted, finite realised design with deterministic prefix remains genuine theory work; the literature does not directly cover this combination.

Path 3 — Direct G_{eff} bound. Prove an upper bound on $G_{\mathrm{eff}} = \|P_\phi M P_y\|_{2\to\infty}$ for the structured realised (Z, W) design, without going through $\lambda_{\min}^{\mathrm{sz}}$. This is closer to the realised-design certificate framing and would license certificate (2) analytically rather than by measurement. Operator-norm techniques on structured kernel-weighted designs are less developed than eigenvalue concentration in the matrix-Bernstein literature, but the question is well-posed.

The companion theory work in preparation pursues Path 2.

D.3 Implications for v3.1

A future v3.1 release may consider migrating to the boundary-complete stratified ramp (Path 1), accepting the ≈25% coalition-count overhead in coalition-masking and model-evaluation steps in exchange for a seed-independent invertibility floor and a tractable companion theorem. K' padding

remains at 512 at $d = 50$, so BHDR rotation budget is unchanged. The current paper does not depend on this migration; Proposition 7's certificates license the deployment as specified.

E Proof of Theorem 6

The proof proceeds in three steps under the i.i.d. pair model: (1) pair-level indexing to obtain a sum of independent matrices from the antithetic design, (2) Matrix Bernstein concentration of the information matrix, and (3) score-vector concentration with per-coordinate extraction.

Step 1: Pair-level indexing. Under i.i.d. pair sampling from the surrogate model fixed at the start of Section 5, the antithetic-pair design adapted to Shapley estimation by Covert and Lee [2] generates $P = K/2$ pairs $(S_j, \bar{S}_j)$ where $\bar{S}_j = [d] \setminus S_j$. Within each pair, S_j and $\bar{S}_j$ are correlated (deterministically complementary); across pairs, $(S_j, \bar{S}_j)$ and $(S_k, \bar{S}_k)$ are independent for $j \neq k$. Define pair-level contributions to the surrogate information matrix as the antithetic average:

$$A_j \;=\; \tfrac{1}{2}\big(\mathbf{z}_j\mathbf{z}_j^\top + \bar{\mathbf{z}}_j\bar{\mathbf{z}}_j^\top\big).$$

The factor $\frac{1}{2}$ produces marginal-probability entries: $\mathbb{E}[A_j[i,i]] = \frac{1}{2}(\mathbb{E}[z_i^2]+\mathbb{E}[\bar{z}_i^2]) = \Pr(z_i = 1) = 1/2$ under the antithetic average. Then $H = \sum_{j=1}^{P} A_j$ is a sum of P independent PSD random matrices. This H is the i.i.d. surrogate information matrix analysed in Theorem 6; it is not the deployed weighted $Z^\top W Z$, which is covered separately by Proposition 7.

Step 2: Information matrix concentration. Proposition 4 gives the closed form $P_0\bar{A}P_0 = (1/(2H_{d-1}))P_0$ on the sum-zero subspace, where $\bar{A} = c_0 I + \beta_d(\mathbf{1}\mathbf{1}^\top - I)$ with $c_0 = 1/2$ and $\beta_d = \frac{1}{2}(1-1/H_{d-1})$. Concretely, the diagonal entries are exactly $1/2$ and the off-diagonal entries approach $1/2$ from below as d grows: at $d = 20$, $\beta_{20} = 0.359$; at $d = 200$, $\beta_{200} = 0.415$. The projected $\bar{A}$ is therefore perfectly conditioned, with $\lambda_{\min}^{\mathrm{sz}}(\bar{A}) = c_0 - \beta_d = 1/(2H_{d-1})$. Applying the Matrix Bernstein inequality (Tropp [17] Thm 6.1.1) to the centered, projected $B_j = P_0(A_j - \bar{A})P_0$, the second-moment proxy on the sum-zero subspace satisfies

$$\|\mathbb{E}[B_j^2]\| \;=\; \Theta\!\left(\frac{d}{H_{d-1}}\right).$$

This follows from the closed-form identity $\mathbb{E}[(P_0\mathbf{z}\mathbf{z}^\top P_0)^2] = \frac{d+1}{12H_{d-1}}P_0$, derived as Lemma 8 below; the inner product $\mathbf{z}^\top\mathbf{z} = |S|$ contributes the linear-in-d factor under the Lundberg–Lee size law. The antithetic projection $P_0\bar{\mathbf{z}} = -P_0\mathbf{z}$ collapses the pair-average on the sum-zero subspace: $P_0A_jP_0 = \frac{1}{2}(P_0\mathbf{z}\mathbf{z}^\top P_0 + P_0\bar{\mathbf{z}}\bar{\mathbf{z}}^\top P_0) = P_0\mathbf{z}\mathbf{z}^\top P_0$, so the per-summand norm satisfies $\|P_0A_jP_0\| = \|P_0\mathbf{z}\|_2^2 = s(d-s)/d$, and $R_B = \max_j \|B_j\| = O(d)$ in the worst case at $s \approx d/2$.

Bernstein denominator (sum form). For the sum $H_{P_0} = \sum_{j=1}^{P} B_j$, the variance proxy and per-summand norm are

$$\sigma^2 \;=\; \Big\|\sum_{j=1}^{P}\mathbb{E}[B_j^2]\Big\| \;=\; \Theta\!\left(\frac{Pd}{H_{d-1}}\right), \qquad R_B \;=\; O(d).$$

The relative-invertibility target for the sum is $t = \frac{1}{2}\|P_0\,\mathbb{E}[H]\,P_0\| = P/(4H_{d-1})$. The Bernstein denominator therefore satisfies

$$\sigma^2 + R_B t \;=\; \Theta\!\left(\frac{Pd}{H_{d-1}}\right) + O(d)\cdot\Theta\!\left(\frac{P}{H_{d-1}}\right) \;=\; \Theta\!\left(\frac{Pd}{H_{d-1}}\right);$$

the variance term and the $R_B t$ term are of the same order, so neither dominates and both give the same sufficient scaling. The Bernstein exponent is $t^2/(\sigma^2 + R_B t) = \Theta(P/(dH_{d-1}))$, and setting this $\geq C \log(4d/\delta)$ for a target failure probability δ yields

$$P \;\geq\; C_1 \cdot d \cdot H_{d-1} \cdot \log(4d/\delta), \quad \text{equivalently} \quad K \;\geq\; 2C_1 \cdot d \cdot H_{d-1} \cdot \log(4d/\delta),$$

for an absolute constant C_1. The factor 2 from $K = 2P$ is absorbed into C_1 to match the theorem statement. Conditional on this relative-invertibility event, $\|(P_0 H P_0)^{-1}\| \;\leq\; 2/(P \cdot \lambda^{\mathrm{sz}}_{\min}(\bar{A})) = 4H_{d-1}/P$.

Lemma 8 (Second-moment proxy on the sum-zero subspace)**.** *Under the i.i.d. Lundberg–Lee antithetic-pair model of Section 5, the centered second moment satisfies*

$$\mathbb{E}\big[(P_0 \mathbf{z}\mathbf{z}^\top P_0)^2\big] \;=\; \frac{d+1}{12H_{d-1}}\, P_0.$$

Proof. For $\mathbf{v} \perp \mathbf{1}$, $\|\mathbf{v}\|_2 = 1$, conditioning on $|S| = s$ and uniform without-replacement sampling gives $\mathbb{E}[(\mathbf{v}^\top \mathbf{z})^2 \mid |S| = s] = s(d-s)/(d(d-1))$. The rank-one identity $(P_0 \mathbf{z}\mathbf{z}^\top P_0)^2 = \|P_0 \mathbf{z}\|_2^2 \cdot P_0 \mathbf{z}\mathbf{z}^\top P_0$ together with $\|P_0 \mathbf{z}\|_2^2 = s(1 - s/d) = s(d-s)/d$ gives

$$\mathbb{E}\big[(P_0 \mathbf{z}\mathbf{z}^\top P_0)^2\big] \;=\; \sum_{s=1}^{d-1} \tilde{w}_s \cdot \frac{s(d-s)}{d} \cdot \mathbb{E}[P_0 \mathbf{z}\mathbf{z}^\top P_0 \mid |S| = s].$$

Substituting $\mathbb{E}[P_0 \mathbf{z}\mathbf{z}^\top P_0 \mid |S| = s] = \frac{s(d-s)}{d(d-1)} P_0$ and $\tilde{w}_s = d/(2H_{d-1} \cdot s(d-s))$ collapses the size sum to

$$\sum_{s=1}^{d-1} \frac{d}{2H_{d-1}\, s(d-s)} \cdot \frac{s(d-s)}{d} \cdot \frac{s(d-s)}{d(d-1)} \;=\; \frac{1}{2H_{d-1} \cdot d(d-1)} \sum_{s=1}^{d-1} s(d-s).$$

Applying the closed-form identity

$$\sum_{s=1}^{d-1} s(d-s) \;=\; d \sum_{s=1}^{d-1} s - \sum_{s=1}^{d-1} s^2 \;=\; \frac{d^2(d-1)}{2} - \frac{(d-1)d(2d-1)}{6} \;=\; \frac{d(d-1)(d+1)}{6},$$

the size-sum collapses to $d(d-1)(d+1)/(6 \cdot 2H_{d-1} \cdot d(d-1)) = (d+1)/(12H_{d-1})$, which is the claimed coefficient. □

Step 3: Score vector concentration. Define the antithetic-pair-averaged score vector $V = \sum_j \mathbf{c}_j$ with pair contribution $\mathbf{c}_j = \frac{1}{2}\big(\varepsilon(S_j)\, \mathbf{z}_j + \varepsilon(\bar{S}_j)\, \bar{\mathbf{z}}_j\big)$, matching the antithetic average used for A_j. Let $\mathbf{c}_j^{P_0} = P_0 \mathbf{c}_j$ denote its projection onto the sum-zero subspace. The antithetic complementarity $P_0 \bar{\mathbf{z}} = -P_0 \mathbf{z}$ collapses the pair-average:

$$\mathbf{c}_j^{P_0} \;=\; \tfrac{1}{2}\, \Delta(S_j) \cdot P_0 \mathbf{z}_j, \qquad \Delta(S_j) = \varepsilon(S_j) - \varepsilon(\bar{S}_j), \qquad |\Delta(S_j)| \leq 2R.$$

The same calculation that yields Proposition 4 gives $\mathbb{E}[P_0 \mathbf{z}\mathbf{z}^\top P_0] = (1/(2H_{d-1}))P_0$ on the sum-zero subspace. Therefore

$$\mathbb{E}[\mathbf{c}_j^{P_0} (\mathbf{c}_j^{P_0})^\top] \;\preceq\; \frac{R^2}{2H_{d-1}}\, P_0,$$

and the per-summand norm is

$$\|\mathbf{c}_j^{P_0}\|_2 \;\leq\; R \cdot \|P_0 \mathbf{z}_j\|_2 \;\leq\; \tfrac{R}{2}\sqrt{d},$$

since $\|P_0\mathbf{z}_j\|_2^2 = s(d-s)/d \le d/4$ at the worst-case coalition size $s \approx d/2$. The earlier $O(R)$ estimate held only under the deployed weighted design and is not available under the antithetic-pair-averaged surrogate analysed here. Applied to $V = \sum_{j=1}^{P} \mathbf{c}_j^{P_0}$, the vector Bernstein variance proxy and per-summand norm are therefore

$$\sigma_V^2 \;=\; \Big\|\sum_{j=1}^{P} \mathbb{E}[\mathbf{c}_j^{P_0}(\mathbf{c}_j^{P_0})^\top]\Big\| \;\le\; \frac{PR^2}{2H_{d-1}}, \qquad L_V \;=\; O(R\sqrt{d}).$$

At the target deviation $t = CR\sqrt{P\log(4d/\delta)/H_{d-1}}$, the linear Bernstein term is dominated by the variance term whenever

$$L_V \cdot t \;=\; O\!\left(R^2\sqrt{\frac{Pd\log(4d/\delta)}{H_{d-1}}}\right) \;\lesssim\; \sigma_V^2 \cdot \log(4d/\delta) \iff P \;\gtrsim\; d\,H_{d-1}\,\log(4d/\delta),$$

which is the same precondition imposed by Step 2 for relative invertibility of P_0HP_0. Under that scaling, vector Bernstein gives

$$\|V\|_2 \;\le\; C_2 \cdot R\sqrt{\frac{P\cdot\log(4d/\delta)}{H_{d-1}}} \;=\; O\!\left(R\sqrt{\frac{P\log(d/\delta)}{\log d}}\right)$$

with probability $\ge 1-\delta/2$. Tighter rates would require an extra residual-cancellation assumption beyond $|\varepsilon(S)-\varepsilon(\bar{S})| \le 2R$, which we do not impose (cf. [2] §4.1; [3] obtains a similar $\sqrt{\log d}$ slack via leverage-score arguments).

Per-coordinate extraction. Combining the Step 2 inverse bound $\|(P_0HP_0)^{-1}\| \le 4H_{d-1}/P$ with the Step 3 score bound:

$$\begin{aligned}
|\hat{\phi}_i - \phi_i^*| = |\mathbf{e}_i^\top (P_0HP_0)^{-1}V| \;&\le\; \|(P_0HP_0)^{-1}\| \cdot \|V\|_2 \\
&\le O(H_{d-1}/P)\cdot O\Big(R\sqrt{P\ln(4d/\delta)/H_{d-1}}\Big) \\
&= O\left(R\sqrt{\frac{H_{d-1}\cdot\ln(4d/\delta)}{P}}\right).
\end{aligned}$$

Substituting $P = K/2$ (absorbed into the constant C) and $H_{d-1} = \Theta(\log d)$ yields

$$|\hat{\phi}_i - \phi_i^*| \;\le\; C\cdot R\sqrt{\frac{H_{d-1}\,\ln(4d/\delta)}{K}} \;=\; O\!\left(R\sqrt{\frac{\log(d)\,\log(d/\delta)}{K}}\right),$$

which matches the theorem statement. The Hoeffding union-bound rate $R\sqrt{\ln(2d/\delta)/K}$ is recovered up to a $\sqrt{\log d}$ polylog slack carried by the $\lambda_{\min}^{\mathrm{sz}}$ scaling. □

Explicit leading-order constant. The constant C in the bound $|\hat{\phi}_i - \phi_i^*| \le C\cdot R\sqrt{H_{d-1}\ln(4d/\delta)/K}$ follows directly from [17] Thm 6.1.1. The matrix Bernstein step gives $\|(P_0HP_0)^{-1}\| \le 4H_{d-1}/P$ on the relative-invertibility event (Step 2 above). The vector Bernstein step on the score vector $V = \sum_j \mathbf{c}_j^{P_0}$ uses $\|\mathbb{E}[\mathbf{c}_j^{P_0}(\mathbf{c}_j^{P_0})^\top]\| \le 2R^2/H_{d-1}$ (Lemma 8) and gives, with probability $\ge 1-\delta/2$,

$$\|V\|_2 \;\le\; 2R\sqrt{\frac{P\,\ln(2d/\delta)}{H_{d-1}}} \;+\; \tfrac{R\sqrt{d}}{3}\cdot\ln(2d/\delta).$$

At the relative-invertibility precondition $P \geq C_1 \cdot d \cdot H_{d-1} \cdot \ln(4d/\delta)$ the linear term is dominated by the first term up to a constant, and the per-coordinate extraction gives

$$|\hat{\phi}_i - \phi_i^*| \;\leq\; \|(P_0 H P_0)^{-1}\| \cdot \|V\|_2 \;\leq\; \tfrac{4H_{d-1}}{P} \cdot 2R\sqrt{\frac{P\,\ln(2d/\delta)}{H_{d-1}}} \;=\; 8R\sqrt{\frac{H_{d-1}\,\ln(2d/\delta)}{P}}.$$

Substituting $P = K/2$ and $\ln(2d/\delta) \leq \ln(4d/\delta)$ yields $|\hat{\phi}_i - \phi_i^*| \leq 8\sqrt{2} \cdot R\sqrt{H_{d-1}\ln(4d/\delta)/K} \approx 11.3 \cdot R\sqrt{H_{d-1}\ln(4d/\delta)/K}$. The leading-order constant is therefore $C \approx 11.3$. This is a factor of $\approx 5.7\times$ larger than the rough $C \approx 2$ plug-in used in earlier drafts; the rate-shape and the relative-invertibility precondition are unchanged.

F Multi-class Extension Sketch (Preview, Not a Contribution)

This appendix is preview material, not a contribution of this paper. The deployed pipeline covers binary classification only; the description below sketches an extension plan whose Tier 2 build-out and end-to-end benchmarks are deferred, and whose Tier 3 latency projection is a linear-scaling extrapolation that has not been measured. We include the sketch because reviewers asked, and because the binary SLA is preserved under the Tier 1 path; we do not claim multi-class deployment.

The natural three-tier extension is the following. Tier 1 is the binary path at $C = 2$: the production configuration, ∼53 s on CX22 for OCTE and ∼45–50 s for logistic regression. Tier 2 covers $C \in \{2, 3, 4\}$ via SIMD packing at the *coalition-output* stage: with $K = 390$ coalitions and $C = 4$ classes, the packed multi-class output occupies $C \cdot K = 1{,}560$ of the $n = 16{,}384$ CKKS slots at $N = 2^{15}$. Multiplicative depth is unchanged because the BHDR matvec is class-agnostic and the mask M is reused across classes; the 51-rotation BHDR step itself does *not* grow with C. The cost increase is concentrated at the input-side packing/gather and the model evaluator: the per-feature input ciphertexts shard from $\lceil Kd/n \rceil = 2$ (binary) to $\lceil C \cdot K \cdot d/n \rceil \approx 5$ ($C = 4$), so model-eval and gather rotations roughly $2.5\times$ the binary-path cost. The aggregate end-to-end pipeline overhead is still bounded ($\ll C\times$) because the BHDR regression dominates total wall-clock under the deployed configuration. We do not commit to a specific multi-class wall-clock target ahead of the Tier 2 build-out and end-to-end measurement. The mask M is class-agnostic, so the regression pseudo-inverse condition number $\kappa(M)$ is unchanged and the binary SLA holds. Tier 3 is a preview for $C \in \{5, \ldots, 10\}$ via a hybrid pack-and-shard split (for example 4+4+2), targeting ∼5 min on CX22 under a caveated SLA. ECOA adverse-action class-count distribution is $C = 2$ at ∼55%, $C = 3$ at ∼25%, $C = 4$ at ∼15%: $C \leq 4$ covers ∼95% of notice-generation traffic. A single Freivalds–KZG check amortises across all C columns, not C times the binary cost. Tiers 2 and 3 are *described as the natural construction extension*; full build-out and end-to-end benchmarks are deferred to a v3.1 release.