TRULY SUBQUADRATIC HIGH-PRECISION GAUSSIAN SVM TRAINING ON STRUCTURED INPUTS

Anonymous authors
Paper under double-blind review

ABSTRACT

We establish truly subquadratic randomized algorithms for high-precision Gaussian support vector machine training by exploiting coordinate structure. For every fixed C, we give randomized algorithms that train n inputs from a sufficiently separated fixed-size real alphabet in $d \le C \log n$ dimensions to additive dual objective error $\exp[-(\log n)^3]$ using $n^{2-\delta+o(1)}$ arithmetic operations for some constant $\delta > 0$, with near-linear preprocessing and space. This truly subquadratic guarantee allows arbitrary repetitions and covers hard and soft margins, with or without bias, provided identical inputs have consistent labels in the hard-margin case. To achieve this, we construct an exact kernel-product procedure whose single success event holds for all real weight vectors simultaneously, including adaptive signed queries. Tensor spectral bounds then turn these products into fast optimization. Building on this procedure, local Taylor features and a diagonal correction extend the guarantee to inputs with distinct discrete parts and $k \le c \log n / \log\log n$ additional arbitrary real coordinates, for any fixed $c < 0.16$, using subquadratic space.

1 INTRODUCTION

Learning nonlinear decision boundaries from vector-valued observations is a basic problem in supervised learning. Kernel methods make it possible to combine such nonlinear prediction with convex training objectives by expressing similarities through a kernel. Support vector machines (SVMs) exemplify this approach, with applications ranging from handwritten digit recognition (Cortes & Vapnik, 1995) to land-cover classification in hyperspectral images (Melgani & Bruzzone, 2004). The Gaussian kernel offers a distance-based notion of similarity whose bandwidth controls the scale of interactions, allowing nonlinear prediction without explicitly constructing its infinite-dimensional feature map.

This flexibility comes with a computational cost: the SVM dual has only n variables, but its quadratic term involves all pairs of training examples. Explicitly forming the kernel matrix therefore takes quadratic time, even when the input consists of only $O(n \log n)$ coordinates. This gap motivates a basic algorithmic question: which structures in the data allow the original SVM objective to be optimized without materializing the dense matrix? We investigate finite-alphabet representations and their extension by continuous measurements, where the coordinates offer structure beyond the size of the kernel matrix.

A well-established approach replaces the kernel by a finite feature representation. Nyström sampling and random Fourier features provide such representations (Williams & Seeger, 2001; Rahimi & Recht, 2007); low-rank SVM solvers then exploit them within the optimization procedure (Fine & Scheinberg, 2001). More recently, Gu et al. (2025) developed fast structured quadratic-programming algorithms and applied them to Gaussian SVMs. The cost of this approach depends on the feature dimension needed at the requested accuracy. In particular, their formal bound retains the dependence on precision, input scale, and feasible-set parameters (Gu et al., 2025, Theorem G.7). Thus, a fast solver for a fixed feature representation does not by itself give fast training at arbitrarily small error for the original kernel.

We study this issue at additive dual objective error

$$\varepsilon_n = \exp[-(\log n)^3]. \tag{1}$$

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: Training guarantees at error ε_n. All dimension constants are fixed. Rows 1–2 require the alphabet separation in their theorems; their σ may differ. Rows 3–4 are deterministic consequences of tensor and Taylor methods (Appendices J and K).

Inputs	Conditions beyond the error tolerance	Arithmetic time
Finite alphabet	$d \le C \log n$; arbitrary repetitions; hard-margin labels agree at identical inputs	$n^{2-7\sigma/8+o(1)}$
Discrete and continuous	Distinct discrete parts; $d \le C \log n$; $k \le c \log n / \log\log n$, $c < 0.08\tau$, $1 \le \tau < 2$	$n^{2-7\sigma/8+o(1)}$
Boolean subset, full grid used	Distinct inputs; $q = e^{-a^2} \le 1/(4d)$	$O((2^d d + n \log n) \log^3 n)$
Arbitrary real inputs	$d \le c \log n / \log\log n$, $c < 1/8$; squared diameter $\le \log^2 n$; cap 2 and bias	$n^{3/2+4c+o(1)}$

At this tolerance, a logarithmic dependence on $1/\varepsilon_n$ contributes only a polylogarithmic factor, whereas a fixed positive power of $1/\varepsilon_n$ is superpolynomial in n. Our question is whether *coordinate structure can support truly subquadratic training at this precision, without a uniformly accurate global low-rank kernel approximation*.

1.1 Contributions and Approach

We answer this question for separated finite alphabets and for their extension by a limited number of continuous coordinates. Throughout, truly subquadratic means $n^{2-\delta+o(1)}$ for a constant $\delta > 0$ when the dimension and alphabet constants are fixed. Table 1 summarizes the scope alongside two deterministic comparisons.

Discrete Inputs. For any fixed C, inputs from a sufficiently separated fixed-size real alphabet in $d \le C \log n$ dimensions admit training in $n^{2-7\sigma/8+o(1)} \log(n/\varepsilon)$ arithmetic operations for every $\varepsilon \in (0, 1)$, with near-linear preprocessing and space (Theorems 1 and 6). Here $\sigma > 0$ depends on C and the alphabet size. Repetitions are unrestricted, and the guarantee covers hard and soft margins, with or without bias. Hard margins require consistent labels at identical inputs; soft margins allow conflicting labels and any positive box cap.

Continuous Coordinates and Repetitions. When the discrete parts are distinct, each input may additionally have $k \le c \log n / \log\log n$ arbitrary real coordinates, with no diameter bound (Theorem 7). At error ε_n, the time remains $n^{2-7\sigma/8+o(1)}$ and space is $n^{\tau+o(1)}$ for $1 \le \tau < 2$ and $c < 0.08\tau$, with a possibly smaller σ. Preprocessing is near-linear. Proposition 9 further allows every repeated discrete group to contain $O(n^{0.41})$ samples for polynomially bounded soft-margin caps, even if its kernel is singular. The corresponding hard-margin extension needs a bound on each group's no-bias dual value.

The Mechanism. The first ingredient is an exact kernel-product procedure whose success event holds for *all real weight vectors simultaneously* (Theorem 2). We extend probabilistic-polynomial counting (Alman & Williams, 2015; Björklund & Kaski, 2021) by retaining individual output rows and recovering each block's weighted sum before summing across blocks. This permits signed weights chosen adaptively by the optimizer. The second ingredient is spectral: tensor structure bounds the kernel's condition number on any subset of distinct discrete locations, while exact aggregation handles coincident inputs. Restarted acceleration converts the product bound into a training bound. For continuous coordinates, local Taylor features and a diagonal correction yield one fixed convex surrogate with a controlled objective error.

1.2 Related Works

Kernel approximation and optimization. Finite representations include Nyström approximations (Williams & Seeger, 2001; Musco & Musco, 2017), random features (Rahimi & Recht, 2007), Taylor features (Cotter et al., 2011), and Gaussian quadrature (Dao et al., 2017). Their guarantees

concern different errors, including statistical, spectral, and entrywise criteria (Bach, 2017; Avron et al., 2017; Modell, 2024). Objective perturbation for low-rank SVM training was analyzed by Fine & Scheinberg (2001), and effects on learned hypotheses by Cortes et al. (2010).

Gaussian summation and precision. Fast Gaussian transforms exploit local or Fourier expansions (Greengard & Strain, 1991; Yang et al., 2003; Morariu et al., 2008; Greengard et al., 2024). Density-estimation methods also accelerate kernel algebra (Charikar & Siminelakis, 2017; Charikar et al., 2020; Backurs et al., 2021); under a total-kernel-mass assumption, Indyk et al. (2025) support arbitrary signed vectors with polynomial dependence on inverse accuracy. Our discrete procedure is exact; its continuous extension evaluates a fixed local approximation. The degree bounds of Aggarwal & Alman (2022) explain why precision must remain explicit when comparing polynomial-feature methods. Conditional hardness results (Backurs et al., 2017; Gu et al., 2025) are consistent with our guarantee: the dimension constant in Gu et al. (2025, Theorem G.12) may depend on the exponent excluded, and our exponent saving may decrease with that constant.

2 Preliminaries

2.1 The Optimization Problem and Accuracy

Let $(x_i, y_i)_{i=1}^n$ be the input, with $n \geq 2$ and $y_i \in \{-1, 1\}$. We use the Gaussian kernel with its bandwidth absorbed into the coordinates: a kernel $e^{-\gamma\|x-x'\|^2}$, $\gamma > 0$, has this form after scaling the inputs by $\sqrt{\gamma}$. Define

$$K_{ij} = e^{-\|x_i - x_j\|_2^2}, \quad Y = \operatorname{diag}(y), \quad Q = YKY, \quad B = \max_{i,j} \|x_i - x_j\|_2^2. \tag{2}$$

Thus B is the squared diameter in kernel-scaled coordinates. The Gaussian Gram matrix K is positive semidefinite, as is Q. Following the standard SVM dual formulation (Cortes & Vapnik, 1995), we seek a feasible vector $\widehat{\alpha}$ with

$$F(\widehat{\alpha}) \geq \mathrm{OPT} - \varepsilon, \qquad F(\alpha) = \mathbf{1}^\top \alpha - \tfrac{1}{2}\alpha^\top Q \alpha, \qquad \mathrm{OPT} = \max_{\alpha \in \mathcal{C}} F(\alpha). \tag{3}$$

The four formulations arise by choosing $U \in (0, \infty]$ and either of the following feasible sets:

$$\mathcal{C}_U^0 = [0, U]^n, \qquad \mathcal{C}_U^{\mathrm{b}} = \{\alpha \in [0, U]^n : y^\top \alpha = 0\}. \tag{4}$$

Here $U = \infty$ denotes hard margin, with $[0, U]^n = \mathbb{R}^n_{\geq 0}$; $U = C_{\mathrm{box}} < \infty$ denotes the usual hinge-loss soft margin. The superscript b indicates an unregularized bias in the primal, whose dual constraint is $y^\top \alpha = 0$; superscript 0 fixes that bias to zero. We write $\mathcal{C}$ for the selected set. If only one label occurs, the bias constraint forces $\alpha = 0$.

The tolerance ε in (3) is an *absolute* error in the unnormalized dual objective. Feasibility, including the bias equality, is exact in our arithmetic model. Hard-margin instances are assumed to satisfy the consistency conditions in the relevant theorem; opposite labels at an identical input make their dual unbounded.

2.2 Coordinate Structure and Algorithmic Tools

The discrete dimension is d. For a finite real alphabet $\mathcal{A}$, discrete inputs lie in $\mathcal{A}^d$; mixed inputs have the form $x_i = (u_i, z_i) \in \mathcal{A}^d \times \mathbb{R}^k$. Repeated *full inputs* and repeated *discrete parts* are different: the latter may have distinct continuous coordinates. In the Boolean case $x_i = au_i$, $u_i \in \{0, 1\}^d$, put $q = e^{-a^2}$ and $H(u, z) = \sum_{j=1}^d [u_j \neq z_j]$. Then $K_{ij} = q^{H(u_i, u_j)}$. Brackets denote indicators.

Probabilistic polynomials. A probabilistic polynomial for a Boolean function g is a distribution over real polynomials P such that $\Pr[P(z) = g(z)] \geq 1 - \eta$ for every fixed Boolean input z. This is exact agreement with a failure probability η. For thresholds $T_t(z) = [\sum_j z_j \geq t]$ on d bits, Alman & Williams (2015) give degree $O(\sqrt{d \log(1/\eta)})$. The pointwise guarantee alone does not cover adaptively selected queries; Section 3 constructs the stronger simultaneous event needed for training.

Optimization and notation. Write $f = -F$, so $\nabla f(\alpha) = YK(Y\alpha) - \mathbf{1}$. Consequently, one gradient reduces to one kernel-vector product. For $0 < \mu I \preceq Q \preceq \Lambda I$, f is μ-strongly convex and Λ-smooth, with condition number Λ/μ (Bubeck, 2015). Here $M \preceq N$ means $N - M \succeq 0$, $\|M\|_2$ is the spectral norm, and vector norms are Euclidean unless indicated otherwise. We use $[n] = \{1, \dots, n\}$, $\mathbf{1}$ for the all-ones vector, $\circ$ for entrywise multiplication, and natural logarithms unless a base is specified.

Computation model. We count real arithmetic operations and comparisons; elementary scalar functions have unit cost. Discrete kernel parameters such as q are evaluated once and stored exactly. Gaussian prefactors and individual entries used in the continuous extension are also counted. Space counts stored real scalars and indices, including the input. Fixed dimension constants and alphabet sizes are independent of n, although symbol values may depend on n.

3 Exact Kernel Products and the Discrete Training Guarantee

We first treat a Boolean alphabet, where the kernel depends only on Hamming distance. This case contains both steps of the argument: computing adaptive kernel products and controlling the cost of training. Section 5 extends the same proof to fixed real alphabets.

Theorem 1. *For every fixed $C \geq 1$ there is a constant $\sigma = \sigma(C) \in (0, 1/10)$, with*

$$\sigma(C) = \Omega\left(\frac{1}{C \log^2(C+2)}\right),$$

such that the following holds. Let $u_1, \dots, u_n \in \{0,1\}^d$, $d \leq C \log n$, and $x_i = a u_i$, where $a > 0$ and

$$a^2 \geq \log \frac{12C}{\sigma}. \tag{5}$$

Labels are arbitrary, except that identical inputs must have equal labels for hard margins. For any $\varepsilon \in (0,1)$ and any of the four feasible sets in (4), a randomized algorithm returns (3) with probability at least $1 - n^{-10}$ in

$$n^{2-7\sigma/8+o(1)} \log(n/\varepsilon) \tag{6}$$

arithmetic operations and $n^{1+o(1)}$ space, after $n^{1+o(1)}$ preprocessing. At $\varepsilon = \varepsilon_n$, the time is $n^{2-\delta(C)+o(1)}$ with $\delta(C) = 7\sigma(C)/8 > 0$.

Appendix E.2 gives the exact reduction for repeated inputs. The alphabet makes products fast; separation controls optimization.

3.1 An Exact Product Procedure

We use the Boolean representation $K_{ij} = q^{H(u_i,u_j)}$ from Section 2.

Theorem 2. *Fix $C \geq 1$. Given $u_1, \dots, u_n \in \{0,1\}^d$, $d \leq C \log n$, and $q \in (0,1)$, there are randomized preprocessing costing $n^{1+o(1)}$ and a query procedure $\mathcal{O}$ costing $n^{2-\sigma(C)+o(1)}$, both using $n^{1+o(1)}$ space, such that*

$$\Pr\big[\mathcal{O}(v) = Kv \text{ for every } v \in \mathbb{R}^n\big] \geq 1 - n^{-10}. \tag{7}$$

The same event covers any sequence of adaptive queries.

Representing a kernel entry. For $z \in \{0,1\}^d$, let $|z| = \sum_j z_j$ and $T_k(z) = [|z| \geq k]$. The identity

$$q^{|z|} = 1 + \sum_{k=1}^{d} (q^k - q^{k-1}) T_k(z) \tag{8}$$

turns these Boolean outputs into a kernel value. For a block size s, set $\eta = (10ds)^{-1}$ and replace each T_k by P_k. A union bound over k gives a polynomial P_q with

$$\Pr[P_q(z) = q^{|z|}] \geq 1 - \frac{1}{10s}, \qquad \deg P_q = O(\sqrt{d \log(10ds)}). \tag{9}$$

The exponential values at the $d+1$ possible Hamming distances are coefficients of the construction. Substitute $z_j = X_j + Z_j - 2X_jZ_j$ and reduce powers of Boolean variables using $X_j^k = X_j$ and $Z_j^k = Z_j$ for $k \geq 1$. The resulting polynomial separates the two inputs:

$$P(X,Z) = \sum_{r=1}^{R} c_r X^{S_r} Z^{T_r}, \quad R \leq \sum_{j=0}^{\min(D,2d)} \binom{2d}{j}, \quad D = O(\sqrt{d\log(10ds)}). \tag{10}$$

Here $S_r, T_r \subseteq [d]$, $X^S = \prod_{j\in S} X_j$, and an empty product is one. Substitution at most doubles the degree. Here R counts monomials in a sampled polynomial.

3.2 One Success Event for Every Later Query

Partition $[n]$ into $m = \lceil n/s \rceil$ groups $G_1, \ldots, G_m$, each of size at most s. For a sampled polynomial, form

$$A_{ir} = u_i^{S_r}, \qquad W_{rg}(v) = c_r \sum_{j\in G_g} u_j^{T_r} v_j. \tag{11}$$

The matrix product $AW(v)$ gives all estimated block sums:

$$(AW(v))_{ig} = \sum_{j\in G_g} P(u_i, u_j) v_j. \tag{12}$$

The next argument makes this weighted block-counting step (Björklund & Kaski, 2021) correct independently of the weights.

Lemma 3. *Sample an odd number $L \geq 50\log(2n)$ of independent polynomials satisfying (9). With probability at least $1 - n^{-10}$,*

$$\sum_{g=1}^{m} \mathrm{median}_{\ell\in[L]} \big(A^{(\ell)} W^{(\ell)}(v)\big)_{ig} = \sum_{j=1}^{n} K_{ij} v_j \quad \textit{for all } i \in [n],\ v \in \mathbb{R}^n. \tag{13}$$

Proof. Let $E_{ig}^{(\ell)}$ be the event that sample ℓ equals the kernel on all pairs (u_i, u_j) with $j \in G_g$. By (9), $\Pr[E_{ig}^{(\ell)}] \geq 9/10$. Independence across samples and Hoeffding's inequality (Hoeffding, 1963) give

$$\Pr\left[\sum_{\ell=1}^{L} \mathbf{1}_{E_{ig}^{(\ell)}} \leq L/2\right] \leq e^{-2L(0.9-0.5)^2} = e^{-0.32L}.$$

There are at most n^2 pairs (i,g), so all of them have a strict majority of correct samples except with probability $n^2 e^{-0.32L} \leq n^{-10}$. On this event, each majority agrees entry by entry throughout its block. Its weighted sum is therefore correct for every v, regardless of signs or magnitudes. The median recovers that common value, and summing the blocks proves the claim. □

The event depends only on the input and sampled polynomials, so later queries need no further probability bound.

Recovery must precede summation. With three samples and three blocks, each sample may err on a different block and be correct on the other two. Every block then has a correct majority, yet all three total sums can be wrong. Taking the median only after summing would lose this guarantee.

3.3 Time and Space

Choose $s = \lfloor n^\sigma \rfloor$ with $\sigma = \Omega(1/(C\log^2(C+2)))$ small enough that $R \leq n^{0.1}$. Evaluate the product one $b \times b$ output submatrix at a time, $b = \lceil \sqrt{n} \rceil$. Since $sR \leq b$ and $R \leq b^{0.3}$, feature construction and rectangular multiplication (Le Gall, 2012) cost $b^{2+o(1)}$ per submatrix. Recover its block sums and discard it before processing the next submatrix. With $m = \lceil n/s \rceil$, this gives $n^{2-\sigma+o(1)}$ total query time and $n^{1+o(1)}$ space, proving Theorem 2. Appendix A.1 provides the full calculation; Appendix B verifies the polynomial construction cost.

4 From Kernel Products to SVM Training

Separation is needed for the iteration bound. We first analyze distinct locations, then reduce repetitions to that case.

4.1 Spectral bounds on an arbitrary subset

Lemma 4. *For distinct $u_i \in \{0,1\}^d$ and $K_{ij} = q^{H(u_i,u_j)}$,*

$$\mu I \preceq K \preceq \Lambda I, \qquad \mu = (1-q)^d, \quad \Lambda = (1+q)^d. \tag{14}$$

The same bounds hold for $Q = YKY$.

Proof. Let E embed the n observed coordinates into the full cube by zero-filling the others. Distinctness gives $E^\top E = I$, and

$$K = E^\top T^{\otimes d} E, \qquad T = \begin{pmatrix} 1 & q \\ q & 1 \end{pmatrix}.$$

The eigenvalues of T are $1-q$ and $1+q$, which proves (14). Since $Y^\top Y = I$, conjugation preserves the bounds. □

The full cube is used only to prove the spectral bound; the algorithm never constructs its kernel matrix. A bound from observed off-diagonal entries gives only $1-(n-1)q$, useless for constant q; the tensor identity instead gives a positive bound depending on d.

4.2 Reducing the Iteration Count

Set $f = -F$. An accelerated projected-gradient stage starts at $a_0 \in \mathcal{C}$, sets $b_1 = a_0$, and uses

$$a_j = \Pi_{\mathcal{C}}\big(b_j - \Lambda^{-1}\nabla f(b_j)\big), \qquad b_{j+1} = a_j + \frac{j-1}{j+2}(a_j - a_{j-1}). \tag{15}$$

Each gradient uses one kernel product with weights Yb_j. The extrapolated point b_j may lie outside $\mathcal{C}$, but the quadratic is smooth everywhere and Theorem 2 covers all these adaptive signed queries on the same success event.

Lemma 5. *Suppose $\mu I \preceq Q \preceq \Lambda I$, $\mu > 0$, and $\mathcal{C}$ is closed, convex, and contains zero. Start at zero and restart (15) every $h = \lceil\sqrt{8\Lambda/\mu}\rceil$ steps. After s stages,*

$$f(a_{sh}) - \min_{\mathcal{C}} f \le \frac{n}{2\mu}\, 2^{-s}. \tag{16}$$

Thus $O(\sqrt{\Lambda/\mu}\log(2+n/(\mu\varepsilon)))$ products and projections suffice for objective error ε. For a linear term $-c^\top a$, replace n by $\|c\|^2$.

Proof sketch. The accelerated estimate $f(a_h) - f^* \le 2\Lambda\|a_0 - a^*\|^2/(h+1)^2$ (Beck & Teboulle, 2009) and strong convexity halve the gap at each restart. The initial gap is at most $n/(2\mu)$ by completing the square. Appendix E gives the complete argument, including the proximal and inexact variants used below. □

This includes unbounded hard-margin feasible sets and reuses the same sampled polynomials at every step.

4.3 Exact Projection, Including Bias

Without bias, projection clips each coordinate to $[0, U]$ in $O(n)$ time. With bias, the solution is

$$[\Pi_{\mathcal{C}}(v)]_i = \mathrm{clip}(v_i - \lambda y_i, 0, U), \qquad g(\lambda) = \sum_i y_i\,\mathrm{clip}(v_i - \lambda y_i, 0, U) = 0. \tag{17}$$

Here clipping with $U = \infty$ means $\max(v_i - \lambda y_i, 0)$. The function g is continuous, nonincreasing, and piecewise linear. Sorting and scanning its at most $2n$ breakpoints finds an exact root in $O(n \log n)$ operations (Dai & Fletcher, 2006). If only one label occurs, the equality forces $\alpha = 0$. Appendix E.1 gives the derivation and handles flat intervals.

Why coincident inputs do not spoil the guarantee. The sample kernel can be singular when inputs coincide, so the spectral lemma is applied to distinct locations, not to the unreduced sample. For hard margins, identical inputs with the same label are replaced by one variable equal to their total dual mass. For soft margins, eliminating the positive and negative masses at each location gives a strongly convex quadratic plus an absolute-value term. Its proximal step, including the bias equality, is still an exact scalar breakpoint scan. Appendix E.2 proves the reduction, reconstruction, and an initial-gap bound independent of the cap and the multiplicities. The common acceleration proof in Appendix E includes these proximal steps.

Proof sketch of Theorem 1. Separation gives $q \le \sigma/(12C)$, hence $\Lambda/\mu \le n^{\sigma/4}$ and $\mu \ge n^{-\sigma/6}$. Lemma 5 uses $O(n^{\sigma/8}\log(n/\varepsilon))$ iterations, each costing $n^{2-\sigma+o(1)}$ for multiplication and $O(n\log n)$ for projection. Their product is (6). The full proof, including the reduction for repetitions, is in Appendix A.2. □

5 Finite Real Alphabets and Kernel Scale

The Boolean proof uses two properties: only finitely many coordinate interactions must be counted, and the one-coordinate kernel has a controlled spectrum. Both extend to an arbitrary fixed-size real alphabet.

Theorem 6. *Fix $C \ge 1$ and an integer $L_0 \ge 2$. There is $\sigma = \sigma(C, L_0) \in (0, 1/10)$ such that the following holds. Let $\mathcal{A} = \{b_1, \dots, b_{L_0}\} \subset \mathbb{R}$, $\Delta = \min_{r\ne s} |b_r - b_s|$, and*

$$\rho := \max_r \sum_{s\ne r} e^{-(b_r-b_s)^2} \le \frac{\sigma}{12C}. \tag{18}$$

For any n inputs from $\mathcal{A}^d$, $d \le C\log n$, all four SVM formulations can be solved to additive error ε in $n^{2-7\sigma/8+o(1)}\log(n/\varepsilon)$ arithmetic operations with probability at least $1-n^{-10}$, using $n^{1+o(1)}$ space and $n^{1+o(1)}$ preprocessing. Hard margins require equal labels at identical inputs.

Proof sketch. Count coordinate interactions for each unordered symbol pair, apply threshold polynomials to these counts, and use one-hot encodings. For fixed L_0, a smaller σ preserves the product bound. The one-coordinate kernel has off-diagonal row sums at most ρ; its tensor power gives $(1-\rho)^d I \preceq K \preceq (1+\rho)^d I$ on distinct locations. Acceleration and exact aggregation finish the proof. Appendix C gives the full reduction and proof. □

All discrete coordinates share the alphabet. Appendix C also bounds ρ using only Δ, without a factor growing with L_0.

Which squared diameters are covered? For n distinct Boolean inputs, $B = \Theta_C(a^2\log n)$ (Appendix C). Thus a sufficiently large constant a^2, $a^2 = \Theta(\log\log n)$, or $a^2 = \Theta(\log n)$ gives logarithmic, $\log n\log\log n$, or log-squared diameter. Increasing a improves the condition bound without changing the polynomial degree.

6 Adding Continuous Coordinates

Write $x_i = (u_i, z_i)$ with $z_i \in \mathbb{R}^k$, and set

$$A_{ij} = e^{-\|u_i-u_j\|^2}, \qquad H_{ij} = e^{-\|z_i-z_j\|^2}, \qquad K = A \circ H.$$

Here $\circ$ denotes entrywise multiplication. The task is to approximate their kernel without losing the discrete spectral bounds or requiring a separate random guarantee at each optimization step.

Theorem 7. *Fix $C \ge 1$, an alphabet size $L_0 \ge 2$, $1 \le \tau < 2$, and $0 \le c < 0.08\tau$. There is $\sigma = \sigma(C, L_0, c, \tau) \in (0, 1/10)$ such that the following holds for sufficiently large n. Let the u_i be distinct members of $\mathcal{A}^d$, $|\mathcal{A}| = L_0$, with*

$$d \le C\log n, \qquad \rho \le \frac{\sigma}{12C}, \qquad k \le \frac{c\log n}{\log\log n}, \tag{19}$$

where ρ is defined in (18). For arbitrary $z_1, \dots, z_n \in \mathbb{R}^k$, all four SVM formulations admit error ε_n with probability at least $1 - n^{-10}$ in $n^{2-7\sigma/8+o(1)}$ arithmetic operations and $n^{\tau+o(1)}$ space, after $n^{1+o(1)}$ preprocessing.

Every $c < 0.16$ is covered by some $\tau < 2$; for $c < 0.08$, choose $\tau = 1$ to obtain near-linear space. The saving σ can depend on this space choice. Distinctness concerns the discrete parts. The proof combines local Gaussian expansions (Greengard & Strain, 1991; Yang et al., 2003), discrete polynomials, and a diagonal correction for objective error.

6.1 From Kernel Error to Objective Error

Kernel approximation can be analyzed directly at the objective (Fine & Scheinberg, 2001). Let $\overline{F}$ have the same feasible set and linear term as F, but quadratic matrix $\overline{Q}$. If $0 \preceq \overline{Q} - Q \preceq 2E$ and $\widehat{\alpha}$ is ξ-optimal for $\overline{F}$, then any maximizer α^* of F satisfies

$$\mathrm{OPT} - F(\widehat{\alpha}) \le F(\alpha^*) - \overline{F}(\alpha^*) + \xi \le \xi + (\alpha^*)^\top E \alpha^*. \tag{20}$$

The first inequality follows from $F(\widehat{\alpha}) \ge \overline{F}(\widehat{\alpha}) \ge \overline{F}(\alpha^*) - \xi$. The correction makes the approximate objective no larger than the original one. Only the original optimum needs a norm bound.

Lemma 8. *Suppose $0 < \mu \le \Lambda$, A has nonnegative entries, $\mu I \preceq A \preceq \Lambda I$, $H \succeq 0$, and $H_{ii} = 1$. Then $\mu I \preceq K = A \circ H \preceq \Lambda I$. For any symmetric $\widehat{H}$ with $|H_{ij} - \widehat{H}_{ij}| \le \eta$, set*

$$\overline{K} = A \circ \widehat{H} + \Lambda \eta I. \quad \textit{Then} \quad K \preceq \overline{K} \preceq K + 2\Lambda\eta I. \tag{21}$$

Solving its objective to error $\varepsilon/2$ gives error ε for F if

$$\eta = \frac{\varepsilon \mu^2}{2n\Lambda}. \tag{22}$$

Proof sketch. The Schur product theorem preserves the spectral bounds. Entrywise error gives $\|A \circ (\widehat{H} - H)\|_2 \le \Lambda\eta$. Optimality along the direction toward zero gives $\|\alpha^*\| \le \sqrt{n}/\mu$; substitute this in (20). The complete proof is in Appendix A.3. □

This includes hard margins. The corrected kernel $\overline{K}$ has condition number at most $(1 + 2\eta)\Lambda/\mu$.

6.2 Local Features Without a Diameter Bound

Use (22) with $\varepsilon = \varepsilon_n$, and write $L = \log(1/\eta) = O(\log^3 n)$ and $h = \sqrt{L/k}$ for $k \ge 1$. In each coordinate, sort the values and greedily cover them by intervals $[a, a + h)$, starting each interval at the smallest uncovered value. Their products partition the observed points into cells, each lying within distance $\sqrt{L}/2$ of its center. Call cells neighbors if their centers are at distance at most $2\sqrt{L}$. Points in nonneighboring cells are more than $\sqrt{L}$ apart, so setting those entries of $\widehat{H}$ to zero incurs error at most $e^{-L} = \eta$.

Each cell has only $e^{O(k)} = n^{o(1)}$ neighbors, which can be found in $n^{1+o(1)}$ total time. Appendix F proves it using interval indices and comparisons.

Set $b = \lceil n^{\tau/2} \rceil$ and $s = \lfloor n^{\sigma} \rfloor$. For neighboring cells, use exact entries if either contains at most bs points. For every other unordered cell pair with centers w, w', use a multi-index $\nu \in \mathbb{Z}^k_{\ge 0}$, with $|\nu| = \sum_j \nu_j$, $\nu! = \prod_j \nu_j!$, and $z^\nu = \prod_j z_j^{\nu_j}$. Define the translated Taylor features (Yang et al., 2003; Cotter et al., 2011)

$$\phi_\nu^{w,w'}(z) = e^{-\|z-w'\|^2 + \|w-w'\|^2/2} \sqrt{2^{|\nu|}/\nu!}\,(z - w)^\nu, \qquad |\nu| \le p := \lceil 3L \rceil. \tag{23}$$

For points z, z' in these cells, set $\widehat{H}(z, z') = \sum_{|\nu| \le p} \phi_\nu^{w,w'}(z) \phi_\nu^{w',w}(z')$. This expression is symmetric. Both displacements from their own centers have norm at most $\sqrt{L}/2$, so $|2(z - w)^\top (z' - w')| \le L/2$. The scalar Taylor remainder gives entrywise error at most η (Appendix F). The feature count is

$$r = \binom{k + p}{k} \le n^{2c+o(1)} \tag{24}$$

features per cell pair. For $k = 0$, use the discrete algorithm directly.

6.3 Combining Local Features With Exact Block Recovery

Local features and discrete monomials must jointly fit the inner dimension allowed by fast rectangular multiplication. Put $\gamma = 0.16\tau - 2c > 0$. Choose $\sigma < (1-\tau/2)/3$ small enough that the discrete polynomial $P(u_i, u_j) = \sum_{a=1}^{R} c_a f_a(u_i) g_a(u_j)$ has $R \le n^{\gamma/2}$. Partition each large database cell into groups of size at most s. For a cell pair with centers w, w', use row features $f_a(u_i)\phi_\nu^{w,w'}(z_i)$ and sums

$$W_{a\nu,g}(v) = c_a \sum_{j \in G_g} g_a(u_j)\phi_\nu^{w',w}(z_j)v_j. \tag{25}$$

Their product has inner dimension Rr. Correctness of the polynomial on a row and group gives its exact weighted sum for $A \circ \widehat{H}$. Lemma 3 then gives one event for all later queries.

The choices ensure $Rr \le b^{0.32}$ and $sRr \le b$. Rectangular multiplication (Vassilevska Williams et al., 2024), with direct evaluation for small cells, then costs $n^{2-\sigma+o(1)}$ per query in $n^{\tau+o(1)}$ space, after near-linear preprocessing. Appendix G gives the full feature, small-cell, and space calculations, including other precisions.

Completing Theorem 7. The discrete spectral bounds give $\mu \ge n^{-\sigma/6}$ and $\Lambda/\mu \le n^{\sigma/4}$. Lemma 5 solves the corrected objective to error $\varepsilon_n/2$ in $n^{\sigma/8+o(1)}$ queries. Lemma 8 transfers the error bound, giving total time $n^{2-7\sigma/8+o(1)}$. □

When only the discrete part repeats. Proposition 9 allows every group to contain $O(n^{0.41})$ samples, with unchanged complexity. For soft margins with $\log(2+U) = O(\log^3 n)$, no within-group spectral condition is needed. Hard margins instead require a supplied bound $e^{a_0(\log n)^3}$ on each group's no-bias dual value, for a fixed $a_0 > 0$. The extension uses within-group kernels as a block metric; Appendix H states the general group-size condition and proves the required inexact-projection bound.

7 Conclusion

We show that separated finite alphabets allow Gaussian SVMs to be trained at very high optimization accuracy in truly subquadratic time, and that the guarantee extends to a limited number of continuous coordinates. The main lesson is that fast kernel products must be paired with control of optimization: simultaneous recovery supports adaptive queries, while tensor spectral bounds limit the iteration count.

These same ingredients explain the present limits. At error ε_n, the Taylor degree $p = \Theta(\log^3 n)$ yields $\binom{k+p}{k} = n^{\Theta(\log\log n)}$ features when $k = \Theta(\log n)$, exceeding the budget for rectangular multiplication (Appendix G). Moreover, weaker alphabet separation worsens conditioning and can erase the product saving. Progress in these directions therefore calls for a more compact continuous representation or spectral bounds better adapted to the data.

Repeated discrete groups pose a further limitation: nearly coincident points with opposite labels can have arbitrarily large hard-margin dual values, so the uniform truncation and approximation tolerance in Appendix H need additional assumptions. Because large groups also make block projections expensive, extending this approach requires control of label geometry and faster block solvers. Finally, converting the asymptotic arithmetic bounds into a practical algorithm requires error-tolerant recovery and coefficient-growth control.

AI USE STATEMENT

Generative AI (OpenAI Codex and ChatGPT) was used for literature discovery and comparison, mathematical formulation and proof development, and manuscript drafting and formatting. The work is theoretical. The authors are responsible for the final content, including all AI-assisted claims and proofs.

REPRODUCIBILITY STATEMENT

Sections 2–6 specify the computational model, assumptions, and algorithms; the appendix provides detailed proofs and parameter choices. This is a theoretical study without an implementation or empirical runtime evaluation.

ETHICS STATEMENT

The work analyzes optimization algorithms. It contains no human-subject study or personal dataset.

REFERENCES

Amol Aggarwal and Josh Alman. Optimal-degree polynomial approximations for exponentials and Gaussian kernel density estimation. In *Computational Complexity Conference*, volume 234 of *LIPIcs*, pp. 22:1–22:23, 2022. doi: 10.4230/LIPIcs.CCC.2022.22. URL https://arxiv.org/abs/2205.06249.

Josh Alman and Ryan Williams. Probabilistic polynomials and Hamming nearest neighbors. In *IEEE Symposium on Foundations of Computer Science*, pp. 136–150, 2015. doi: 10.1109/FOCS.2015.18. URL https://arxiv.org/abs/1507.05106.

Haim Avron, Michael Kapralov, Cameron Musco, Christopher Musco, Ameya Velingker, and Amir Zandieh. Random Fourier features for kernel ridge regression: Approximation bounds and statistical guarantees. In *International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 253–262, 2017. URL https://proceedings.mlr.press/v70/avron17a.html.

Francis Bach. On the equivalence between kernel quadrature rules and random feature expansions. *Journal of Machine Learning Research*, 18(21):1–38, 2017. URL https://www.jmlr.org/papers/v18/15-178.html.

Arturs Backurs, Piotr Indyk, and Ludwig Schmidt. On the fine-grained complexity of empirical risk minimization: Kernel methods and neural networks. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://arxiv.org/abs/1704.02958.

Arturs Backurs, Piotr Indyk, Cameron Musco, and Tal Wagner. Faster kernel matrix algebra via density estimation. In *International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 500–510, 2021. URL https://proceedings.mlr.press/v139/backurs21a.html.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009. doi: 10.1137/080716542. URL https://doi.org/10.1137/080716542.

Kristin P. Bennett and Erin J. Bredensteiner. Duality and geometry in SVM classifiers. In *International Conference on Machine Learning*, pp. 57–64, 2000. URL https://www.robots.ox.ac.uk/~cvrg/bennett00duality.pdf.

Andreas Björklund and Petteri Kaski. Counting short vector pairs by inner product and relations to the permanent. In *International Colloquium on Automata, Languages, and Programming*, volume 198 of *LIPIcs*, pp. 29:1–29:21, 2021. doi: 10.4230/LIPIcs.ICALP.2021.29. URL https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ICALP.2021.29.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. URL https://web.stanford.edu/~boyd/cvxbook/.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011. doi: 10.1561/2200000016. URL https://web.stanford.edu/~boyd/papers/admm_distr_stats.html.

Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, 8(3–4):231–357, 2015. URL https://arxiv.org/abs/1405.4980.

Moses Charikar and Paris Siminelakis. Hashing-based-estimators for kernel density in high dimensions. In *IEEE Symposium on Foundations of Computer Science*, 2017. doi: 10.1109/FOCS.2017.99. URL https://arxiv.org/abs/1808.10530.

Moses Charikar, Michael Kapralov, Navid Nouri, and Paris Siminelakis. Kernel density estimation through density constrained near neighbor search. In *IEEE Symposium on Foundations of Computer Science*, 2020. URL https://arxiv.org/abs/2011.06997.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995. doi: 10.1007/BF00994018. URL https://link.springer.com/article/10.1007/BF00994018.

Corinna Cortes, Mehryar Mohri, and Ameet Talwalkar. On the impact of kernel approximation on learning accuracy. In *International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pp. 113–120, 2010. URL https://proceedings.mlr.press/v9/cortes10a.html.

Andrew Cotter, Joseph Keshet, and Nathan Srebro. Explicit approximations of the Gaussian kernel. *arXiv preprint arXiv:1109.4603*, 2011. URL https://arxiv.org/abs/1109.4603.

Kurt Cutajar, Michael Osborne, John Cunningham, and Maurizio Filippone. Preconditioning kernel matrices. In *International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 2529–2538, 2016. URL https://proceedings.mlr.press/v48/cutajar16.html.

Yu-Hong Dai and Roger Fletcher. New algorithms for singly linearly constrained quadratic programs subject to lower and upper bounds. *Mathematical Programming*, 106(3):403–421, 2006. doi: 10.1007/s10107-005-0595-2. URL https://link.springer.com/article/10.1007/s10107-005-0595-2.

Tri Dao, Christopher De Sa, and Christopher Ré. Gaussian quadrature for kernel features. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/62f91ce9b820a491ee78c108636db089-Abstract.html.

Shai Fine and Katya Scheinberg. Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research*, 2:243–264, 2001. URL https://www.jmlr.org/papers/v2/fine01a.html.

Pontus Giselsson and Stephen Boyd. Preconditioning in fast dual gradient methods. In *IEEE Conference on Decision and Control*, pp. 5040–5045, 2014. URL https://stanford.edu/~boyd/papers/precond_fdgm.html.

Leslie Greengard and John Strain. The fast Gauss transform. *SIAM Journal on Scientific and Statistical Computing*, 12(1):79–94, 1991. doi: 10.1137/0912004. URL https://epubs.siam.org/doi/10.1137/0912004.

Leslie F. Greengard, Shidong Jiang, Manas Rachh, and Jun Wang. A new version of the adaptive fast Gauss transform for discrete and continuous sources. *SIAM Review*, 66(2):287–315, 2024. doi: 10.1137/23M1572453. URL https://epubs.siam.org/doi/10.1137/23M1572453.

Yuzhou Gu, Zhao Song, and Lichen Zhang. Faster algorithms for structured linear and kernel support vector machines. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/45a9d4290ae75ceb752b2a61be1d2baa-Abstract-Conference.html.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. doi: 10.1080/01621459.1963.10500830. URL https://doi.org/10.1080/01621459.1963.10500830.

Piotr Indyk, Michael Kapralov, Kshiteej Jitesh Sheth, and Tal Wagner. Improved algorithms for kernel matrix-vector multiplication under sparsity assumptions. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=wLnls9LS3x.

Kurt A. Kramer, Lawrence O. Hall, Dmitry B. Goldgof, Andrew Remsen, and Tong Luo. Fast support vector machines for continuous data. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(4):989–1001, 2009. doi: 10.1109/TSMCB.2008.2011645. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC4467789/.

François Le Gall. Faster algorithms for rectangular matrix multiplication. In *IEEE Symposium on Foundations of Computer Science*, pp. 514–523, 2012. doi: 10.1109/FOCS.2012.80. URL https://arxiv.org/abs/1204.1111.

Farid Melgani and Lorenzo Bruzzone. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on Geoscience and Remote Sensing*, 42(8):1778–1790, 2004. doi: 10.1109/TGRS.2004.831865. URL https://rslab.disi.unitn.it/papers/R30-TGARS-SVM-HYPER.pdf.

Alexander Modell. Entrywise error bounds for low-rank approximations of kernel matrices. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/e55653539c9a9aa096a1fc8ca77ff413-Abstract-Conference.html.

Vlad I. Morariu, Balaji V. Srinivasan, Vikas C. Raykar, Ramani Duraiswami, and Larry S. Davis. Automatic online tuning for fast Gaussian summation. In *Advances in Neural Information Processing Systems*, volume 21, 2008. URL https://proceedings.neurips.cc/paper/2008/hash/d96409bf894217686ba124d7356686c9-Abstract.html.

Cameron Musco and Christopher Musco. Recursive sampling for the Nyström method. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/a03fa30821986dff10fc66647c84c9c3-Abstract.html.

Yurii Nesterov and Arkadii Nemirovskii. *Interior-Point Polynomial Algorithms in Convex Programming*. Society for Industrial and Applied Mathematics, 1994. doi: 10.1137/1.9781611970791.

Neal Parikh and Stephen Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1 (3):127–239, 2014. doi: 10.1561/2400000003. URL https://web.stanford.edu/~boyd/papers/prox_algs.html.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, volume 20, 2007. URL https://proceedings.neurips.cc/paper_files/paper/2007/hash/013a006f03dbc5392effeb8f18fda755-Abstract.html.

Mark Schmidt, Nicolas Le Roux, and Francis Bach. Convergence rates of inexact proximal-gradient methods for convex optimization. In *Advances in Neural Information Processing Systems*, volume 24, pp. 1458–1466, 2011. URL https://proceedings.neurips.cc/paper/2011/hash/8f7d807e1f53eff5f9efbe5cb81090fb-Abstract.html.

Volker Strassen. Gaussian elimination is not optimal. *Numerische Mathematik*, 13(4):354–356, 1969. doi: 10.1007/BF02165411. URL https://link.springer.com/article/10.1007/BF02165411.

Virginia Vassilevska Williams, Yinzhan Xu, Zixuan Xu, and Renfei Zhou. New bounds for matrix multiplication: from alpha to omega. In *ACM-SIAM Symposium on Discrete Algorithms*, pp. 3792–3835, 2024. doi: 10.1137/1.9781611977912.134. URL https://arxiv.org/abs/2307.07970.

Christopher K. I. Williams and Matthias Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems*, volume 13, pp. 682–688, 2001. URL https://proceedings.neurips.cc/paper/2000/file/19de10adbaa1b2ee13f77f679fa1483a-Paper.pdf.

Changjiang Yang, Ramani Duraiswami, Nail A. Gumerov, and Larry Davis. Improved fast Gauss transform and efficient kernel density estimation. In *IEEE International Conference on Computer Vision*, 2003. URL https://users.umiacs.umd.edu/~ramanid/pubs/0464_yang.pdf.

A Additional proofs for the main text

A.1 Time and space for the exact kernel oracle

Take $s = \lfloor n^\sigma \rfloor$. The monomial count in (10) satisfies

$$\log R \le O\left(\sqrt{C\sigma}\log\frac{C+2}{\sigma}\right)\log n + o(\log n). \tag{26}$$

Thus $\sigma = c_*/(C\log^2(C+2))$, for a sufficiently small absolute $c_* > 0$, gives $R \le n^{0.1}$ and $\sigma < 1/10$. Appendix B proves this choice and the near-linear cost of sampling and expanding the polynomials.

Compute $AW(v)$ one $b \times b$ output submatrix at a time, with $b = \lceil \sqrt{n} \rceil$. Form the features, multiply for each sample, recover the block sums by their medians, add them to the row sums, and discard the submatrix. The conditions $sR \le b$ and $R \le b^{0.3}$ make both feature construction and rectangular multiplication cost $b^{2+o(1)}$ per submatrix (Le Gall, 2012). Hence

$$\left\lceil \frac{n}{b} \right\rceil \left\lceil \frac{m}{b} \right\rceil b^{2+o(1)} = n^{2-\sigma+o(1)}. \tag{27}$$

Only the input, groups, polynomials, and one set of sample submatrices are stored, using $n^{1+o(1)}$ space. This proves Theorem 2. Appendix B.1 gives the submatrix calculation for a general inner dimension, as needed later.

A.2 Complete proof of Boolean-alphabet training

Proof of Theorem 1. First assume the inputs are distinct, and write $\kappa = \Lambda/\mu$. Condition (5) gives $q \le \sigma/(12C) < 1/2$. Using $\log((1+q)/(1-q)) \le 3q$ and $\log(1-q) \ge -2q$,

$$\kappa \le n^{\sigma/4}, \qquad \mu \ge n^{-\sigma/6}. \tag{28}$$

Lemma 5 therefore requires $O(n^{\sigma/8}\log(n/\varepsilon))$ iterations. Each costs $n^{2-\sigma+o(1)}$ for multiplication and at most $O(n\log n)$ for projection. Their product is (6). Preprocessing has smaller cost, every returned point is feasible, and all iterations use the same success event. Appendix E.2 removes distinctness with unchanged bounds. □

A.3 Complete proof of the diagonal correction

Proof of Lemma 8. Schur multiplication by H preserves positive semidefiniteness and maps I to I, so it preserves the bounds $\mu I, \Lambda I$. For $E_0 = A \circ (\widehat{H} - H)$,

$$|v^\top E_0 v| \le \eta\, |v|^\top A |v| \le \Lambda\eta \|v\|^2,$$

which proves (21). At an optimum, the feasible direction toward zero gives $\mu\|\alpha^*\|^2 \le (\alpha^*)^\top Q\alpha^* \le \mathbf{1}^\top\alpha^* \le \sqrt{n}\|\alpha^*\|$. Apply (20) with $E = \Lambda\eta I$ and $\xi = \varepsilon/2$. □

B Polynomial size and construction time

This section supplies the parameter details for (26). Pad the inputs to $d_0 = \lceil C\log n \rceil$ coordinates. For $s = \lfloor n^\sigma \rfloor$, the degree after Boolean substitution is

$$D \le A\sqrt{d_0 \log(10 d_0 s)} = (A\sqrt{C\sigma} + o(1))\log n$$

for an absolute A. Choose σ small enough that $D < d_0$. The binomial bound in the main text gives, for another absolute constant A_1,

$$\frac{\log R}{\log n} \le A_1\sqrt{C\sigma}\log\frac{C+2}{\sigma} + o(1).$$

Set $\sigma = c_*/(C\log^2(C+2))$. The leading term becomes

$$\frac{A_1\sqrt{c_*}}{\log(C+2)}\log\frac{C(C+2)\log^2(C+2)}{c_*} = O\big(\sqrt{c_*}(1+\log(1/c_*))\big),$$

uniformly for $C \geq 1$. A sufficiently small absolute $c_* > 0$ makes it less than 0.08. Thus $R \leq n^{0.1}$ for large n; the finite initial range is absorbed in the constants.

The sampler of Alman & Williams (2015, Theorem 1.1 and Section 3) constructs each threshold polynomial in time quasilinear in its degree-bounded monomial count. Choose the degree constant to cover the intermediate products and Boolean substitutions, and write $R_0 = \sum_{j \leq D} \binom{2d_0}{j} \leq n^{0.1}$. Multilinear polynomials can be stored by their subsets of variables; naive multiplication and Boolean reduction cost $O(R_0^2 \operatorname{poly}(d_0))$. Combining the d_0 thresholds and substituting the disagreement indicators therefore costs at most $n^{0.2+o(1)}$ per sample. The $O(\log n)$ samples fit within near-linear preprocessing. Coefficients count as real scalars; their binary encoding lengths are not bounded here.

B.1 Submatrix evaluation and space

This gives the general space calculation used in Sections 3.3 and 6. For a $b \times b$ output submatrix with inner dimension D, form its row features and weighted sums using at most bs database points. This costs $O(bsD \operatorname{polylog} n)$ operations. If $sD \leq b$ and multiplication costs $b^{2+o(1)}$, feature construction fits the same bound. Repeat for the $O(\log n)$ sampled polynomials, recover each block sum by its median, add it to its row sum, and discard the submatrix. Reuse the same polynomials for all submatrices and queries. An $N \times M$ output with $N, M \geq b$ therefore costs $NMn^{o(1)}$ time and $b^{2+o(1)}$ working space, including the sample outputs. The multiplication itself uses at most its operation count in space. Final incomplete submatrices are padded with zeros.

For the Boolean case, $b = \lceil \sqrt{n} \rceil$, $D = R \leq n^{0.1}$, and $sR \leq b$. The bound of Le Gall (2012) gives the required $b^{2+o(1)}$ multiplication time because $R \leq b^{0.3}$. Both output dimensions n and $\lceil n/s \rceil$ are at least b. The input, grouping, polynomials, and one set of sample submatrices use $n^{1+o(1)}$ space in total.

C Full finite-alphabet reduction

Proof. For each of the $M = \binom{L_0}{2}$ unordered pairs $r < s$, define

$$h_{rs}(x, z) = \sum_{j=1}^{d} [\{x_j, z_j\} = \{b_r, b_s\}], \qquad q_{rs} = e^{-(b_r - b_s)^2}. \tag{29}$$

Equal symbols contribute zero squared distance, so

$$K(x, z) = \prod_{r<s} q_{rs}^{h_{rs}(x,z)}. \tag{30}$$

Apply (8) to each factor, giving every threshold error probability $(10Mds)^{-1}$. Use one-hot encodings $X_{j,r} = [x_j = b_r]$, $Z_{j,r} = [z_j = b_r]$; then the Boolean indicator in (29) is $X_{j,r}Z_{j,s} + X_{j,s}Z_{j,r}$. Multiply the M factor polynomials and substitute these indicators. The result is exact at each fixed input pair with probability at least $1 - 1/(10s)$ and has degree

$$D = O\Big(M\sqrt{d \log(Mds)}\Big) \quad \text{in } 2L_0 d \text{ Boolean variables.}$$

Because C, L_0 are fixed, one can choose $s = n^{\sigma}$ with $\sigma > 0$ small enough that the separated expansion has at most $n^{0.1}$ monomials. Lemma 3 and (27) now apply without change.

For the spectrum on distinct inputs, set $T_{rs} = e^{-(b_r - b_s)^2}$. Its diagonal is one and its off-diagonal row sums are at most ρ. Gershgorin's bound and the same embedding used in Lemma 4 give

$$(1-\rho)I \preceq T \preceq (1+\rho)I \quad \Longrightarrow \quad (1-\rho)^d I \preceq K \preceq (1+\rho)^d I. \tag{31}$$

Condition (18) gives (28) with q replaced by ρ. Lemma 5 and the projection formula finish the distinct-input case. Appendix E.2 handles repetitions. Appendix D gives the dependence of σ on L_0. □

A spacing bound independent of alphabet size. Put $q_\Delta = e^{-\Delta^2}$. After sorting the alphabet, the jth symbol on either side of any fixed symbol is at least $j\Delta$ away. Since $j^2 \ge 1 + 3(j-1)$, the row sum in (18) obeys

$$\rho \le \min\left\{(L_0-1)q_\Delta, 2\sum_{j\ge 1} q_\Delta^{j^2}\right\} \le \min\left\{(L_0-1)q_\Delta, \frac{2q_\Delta}{1-q_\Delta^3}\right\}.$$

Either bound supplies a sufficient spacing condition. The exact row sum can be smaller still, and computing it costs only $O(L_0^2)$ kernel evaluations. The polynomial exponent σ continues to depend on L_0 as described in Appendix D.

Squared diameter on Boolean subsets. For fixed C, any n distinct points in $\{0,1\}^d$ with $d \le C\log n$ have Hamming diameter $\Theta_C(\log n)$. Indeed, a set of diameter at most $\xi \log n$ lies in a Hamming ball of size at most $n^{\xi\log(eC/\xi)+o(1)}$, which is less than n for sufficiently small $\xi = \xi(C) > 0$. For inputs $x_i = au_i$, this proves

$$B = \Theta_C(a^2 \log n). \tag{32}$$

D Dependence on alphabet size

The full proof of Theorem 6 is in Appendix C. Here we check the size dependence in its polynomial step. After padding to $d_0 = \lceil C\log n\rceil$, let $M = \binom{L_0}{2}$ and $s = \lfloor n^\sigma \rfloor$. There are $2L_0 d_0$ variables and degree $D = O(M\sqrt{d_0 \log(10 M d_0 s)})$. Hence

$$\frac{\log R}{\log n} \le O\left(M\sqrt{C\sigma}\log\frac{CL_0+2}{\sigma}\right) + o(1).$$

The same calculation as in Appendix B permits

$$\sigma(C, L_0) = \frac{c_*}{CL_0^4 \log^2(CL_0+2)}$$

for a sufficiently small absolute $c_* > 0$, reducing it further if needed to make $\sigma < 1/10$. It gives $R \le n^{0.1}$. The number of factors and the one-hot expansion costs depend only on the fixed alphabet size. Their construction therefore fits the same $n^{0.2+o(1)}$ bound.

Symbol values may depend on n, since they enter only through the M scalars q_{rs}. The dataset may occupy any subset of $\mathcal{A}^d$. The algorithm never lists unobserved grid points.

E Accelerated projection and restarting

Here is a self-contained version of the standard accelerated-gradient argument (Beck & Teboulle, 2009; Schmidt et al., 2011), covering both absolute values and inexact projections. Let $\Phi = f + g$, where f is convex and Λ-smooth and g is proper, closed, and convex, possibly the indicator of $\mathcal{C}$. Allow an update x^+ satisfying, for all $z \in \operatorname{dom} g$,

$$\langle \nabla f(y) + \Lambda(x^+ - y), z - x^+\rangle + g(z) - g(x^+) \ge -\Delta, \qquad \Delta \ge 0.$$

Minimizing $g(x) + \langle \nabla f(y), x - y\rangle + \Lambda\|x-y\|^2/2$ gives $\Delta = 0$. Smoothness and convexity then give

$$\Phi(x^+) \le \Phi(z) + \frac{\Lambda}{2}\left(\|z-y\|^2 - \|z-x^+\|^2\right) + \Delta. \tag{33}$$

Indeed, combine the smooth upper bound at x^+ with $f(z) \ge f(y) + \langle \nabla f(y), z - y\rangle$ and the update condition.

Let $t_j = (j+1)/2$, $v_j = \Phi(a_j) - \Phi^*$, and $w_j = t_j a_j - (t_j - 1)a_{j-1} - a^*$. Use this update with the momentum in (15). Apply (33) to $z = (1 - 1/t_j)a_{j-1} + a^*/t_j$, and use convexity at z. For $j \ge 2$, the recurrence in (15) gives

$$\frac{2t_j^2}{\Lambda}v_j + \|w_j\|^2 \le \frac{2(t_j^2 - t_j)}{\Lambda}v_{j-1} + \|w_{j-1}\|^2 + \frac{2\Delta t_j^2}{\Lambda} \le \frac{2t_{j-1}^2}{\Lambda}v_{j-1} + \|w_{j-1}\|^2 + \frac{2\Delta t_j^2}{\Lambda}.$$

Here $t_j^2 - t_j \le t_{j-1}^2$ and $v_{j-1} \ge 0$. At $j = 1$, the right side is $\|a_0 - a^*\|^2 + 2\Delta/\Lambda$. Telescoping and $\sum_{j=1}^{h} t_j^2 \le h t_h^2$ give

$$v_h \le \frac{2\Lambda\|a_0 - a^*\|^2}{(h+1)^2} + h\Delta \le \tfrac{1}{2} v_0 + h\Delta, \tag{34}$$

where the last inequality assumes μ-strong convexity and $h \ge \sqrt{8\Lambda/\mu}$. Indeed, optimality gives $v_0 \ge \mu\|a_0 - a^*\|^2/2$. Setting $\Delta = 0$ proves the exact restart bound. For the quadratic $f(a) = a^\top Q a/2 - c^\top a$ with $Q \succeq \mu I$ and $0 \in \mathcal{C}$, completing the square gives $f(0) - f^* \le \|c\|^2/(2\mu)$. Taking $c = \mathbf{1}$ and applying the halving bound for s stages proves (16); the same calculation proves the stated general linear-term version.

E.1 Exact projection with one equality

For (17), clipping with $U = \infty$ means $\max(v_i - \lambda y_i, 0)$. The formula follows by minimizing $\|z - v\|^2/2 + \lambda y^\top z$ coordinate by coordinate. A root of g enforces the equality and gives the unique projection. The function g is continuous, nonincreasing, and piecewise linear. Its breakpoints are $y_i v_i$, together with $y_i(v_i - U)$ when $U < \infty$. Between consecutive breakpoints its slope is the negative of the number of coordinates strictly inside their bounds. Sorting the at most $2n$ breakpoints and scanning the affine pieces finds a root in $O(n \log n)$ operations, with no tolerance-dependent root search. If a whole interval has value zero, any point in it gives the same projection. If only one label occurs, the equality forces $\alpha = 0$ and training is immediate. This is the standard scalar-multiplier projection method (Dai & Fletcher, 2006).

E.2 Combining coincident inputs

Group identical full inputs into $m \le n$ distinct locations. Their kernel K_* satisfies the same bounds $\mu I \preceq K_* \preceq \Lambda I$ as above. Throughout, use the original sample size n in the dimension and complexity bounds. The kernel oracle can operate on m locations by padding its input list to length n with repetitions and giving these added entries zero query weights. Thus no assumption $d \le C \log m$ is needed.

Combining identical examples into weighted examples is standard (Kramer et al., 2009, Section 3.2). For soft margins, let p_r, q_r be the total dual masses of the positive and negative examples at location r, whose label counts are n_r^+, n_r^-. Their bounds are $P_r = n_r^+ U$, $N_r = n_r^- U$. Set $s_r = p_r - q_r$ and $c_r = P_r - N_r$. For fixed s_r,

$$\max_{\substack{p_r - q_r = s_r \\ 0 \le p_r \le P_r,\ 0 \le q_r \le N_r}} (p_r + q_r) = P_r + N_r - |s_r - c_r|, \qquad -N_r \le s_r \le P_r. \tag{35}$$

Indeed, take $p_r = \min\{P_r, N_r + s_r\}$ and $q_r = p_r - s_r$. The original problem is therefore exactly

$$\mathrm{OPT} = nU - \min_{s \in \mathcal{S}} \Phi(s), \qquad \Phi(s) = \tfrac{1}{2} s^\top K_* s + \sum_{r=1}^{m} |s_r - c_r|, \quad \mathcal{S} = \prod_r [-N_r, P_r], \tag{36}$$

intersecting $\mathcal{S}$ with $\mathbf{1}^\top s = 0$ when bias is present. Distribute p_r, q_r among their respective examples, with cap U, to recover a feasible original vector with exactly the same objective gap in $O(n)$ time.

Only the quadratic part needs differentiation. With $v = b - K_* b/\Lambda$, the proximal step is

$$s_r(\lambda) = \mathrm{clip}\big(c_r + S_{1/\Lambda}(v_r - \lambda - c_r), -N_r, P_r\big), \qquad S_t(z) = \mathrm{sign}(z)\max\{|z| - t, 0\}. \tag{37}$$

This follows by minimizing each quadratic plus absolute value (Parikh & Boyd, 2014). Without bias use $\lambda = 0$. With bias, choose $\sum_r s_r(\lambda) = 0$. The sum is continuous and nonincreasing, and each coordinate has at most four breakpoints:

$$v_r - P_r - \Lambda^{-1}, \quad v_r - c_r - \Lambda^{-1}, \quad v_r - c_r + \Lambda^{-1}, \quad v_r + N_r + \Lambda^{-1}.$$

Sorting and scanning their affine pieces finds an exact root in $O(m \log(2m))$ time, including coincident breakpoints and flat pieces. A root exists because the limiting sums are $\sum_r P_r \ge 0$ and $-\sum_r N_r \le 0$.

The objective is μ-strongly convex. Since $0 \in \mathcal{S}$,

$$\Phi(0) - \Phi(s) \le \|s\|_1 - \tfrac{\mu}{2}\|s\|^2 \le \frac{m}{2\mu}.$$

The preceding acceleration proof thus gives the same restart count as Lemma 5, independently of U and multiplicities. Grouping costs $O(nd \log n)$; the oracle, proximal steps, and lifting preserve every claimed time, space, preprocessing, and probability bound.

For hard margins, opposite labels at one location make the dual unbounded: assign mass t to one example of each label to get $F = 2t$, also satisfying the bias equality. Otherwise each group has one label. Replace its dual variables by their sum, a single nonnegative variable, and use the distinct-input algorithm. Assigning each returned mass to one original example lifts the solution exactly.

F Finding the neighboring cells

Let $h = \sqrt{L/k}$ and consider the greedy intervals in one coordinate. Their left endpoints are at least h apart. If two intervals have indices a, b, their centers therefore differ by at least $h|a-b|$. A product cell is indexed by a vector in $[n]^k$. Consequently, neighboring centers have index difference $t \in \mathbb{Z}^k$ with $\|t\|^2 \le 4k$. The number of possible differences is bounded independently of n:

$$\#\{t \in \mathbb{Z}^k : \|t\|^2 \le 4k\} \le e^{4k} \left(\sum_{j \in \mathbb{Z}} e^{-j^2} \right)^k = e^{O(k)}.$$

Enumerate these integer vectors recursively, stopping a branch when its squared norm exceeds $4k$. The same bound, summed over the k recursion depths, bounds the work by $ke^{O(k)}$. Store the occupied cell indices in lexicographic order. For each candidate neighbor, binary search this list and then check the actual center distance. Large gaps between interval endpoints only remove candidates. The total work, including sorting the original coordinates, is $nke^{O(k)} \operatorname{polylog} n = n^{1+o(1)}$. No floor operation on an arbitrary real coordinate is required.

Every point is within $h\sqrt{k}/2 = \sqrt{L}/2$ of its cell center. If centers are more than $2\sqrt{L}$ apart, all cross distances exceed $\sqrt{L}$, justifying the cutoff. Local Taylor truncation and distance cutoffs are standard in fast Gaussian summation (Yang et al., 2003; Morariu et al., 2008). Here the two centers give a direct scalar remainder bound. Put $t = 2(z-w)^\top (z'-w')$, so $|t| \le L/2$, and $m = p+1$. Expanding the dot-product powers shows that

$$\widehat{H}(z,z') = e^{-\|z-z'\|^2 - t} \sum_{j=0}^{p} \frac{t^j}{j!}.$$

Taylor's remainder and $m! \ge (m/e)^m$ imply

$$|H(z,z') - \widehat{H}(z,z')| \le e^{L/2} \frac{(L/2)^m}{m!} \le e^{L/2} (e/6)^{3L} \le e^{-L} = \eta, \tag{38}$$

since $m \ge 3L$ and $3\log 6 - 7/2 > 1$. Indeed, the exponential factor in the remainder is at most $e^{-\|z-z'\|^2 + \max\{0,-t\}} \le e^{L/2}$. Only entrywise accuracy and symmetry are needed for the diagonal correction.

G Parameters and total work for the continuous extension

Let $\gamma = 0.16\tau - 2c > 0$ and $M = \binom{L_0}{2}$. The discrete monomial calculation gives

$$\frac{\log R}{\log n} \le O\left(M\sqrt{C\sigma} \log \frac{CL_0 + 2}{\sigma} \right) + o(1).$$

For a sufficiently small absolute $c_* > 0$, take

$$\sigma = \min\left\{ \frac{c_* \gamma^2}{CL_0^4 \log^2((CL_0+2)/\gamma)}, \frac{1-\tau/2}{4}, \frac{1}{20} \right\}.$$

The leading monomial exponent is then below $\gamma/4$, so $R \le n^{\gamma/2}$ for large n. Sampling and expanding the $O(\log n)$ polynomials costs $n^{\gamma+o(1)}$; they are reused across all cell pairs. Since $\gamma \le 0.16\tau < 0.32$, this cost and the cell construction fit in $n^{1+o(1)}$ preprocessing.

Equation (22) gives $L = \log(2n\Lambda/(\varepsilon_n \mu^2)) = O(\log^3 n)$. Hence

$$\log r \le k \log \frac{e(k+p)}{k} \le (2c + o(1)) \log n, \qquad Rr \le n^{0.16\tau - \gamma/2 + o(1)}.$$

The multi-indices and their monomials can be generated in $\mathrm{poly}(k, p)$ time per feature, which is polylogarithmic in n. They are evaluated anew for each submatrix using that cell pair's two centers. For $b = \lceil n^{\tau/2} \rceil$, $Rr \le b^{0.32}$ and $sRr \le b$ because $\sigma + 0.16\tau < \tau/2$. Thus the feature construction, rectangular multiplication, and recovery cost $b^{2+o(1)}$ per submatrix. Their space is at most the same quantity.

Write n_a for the size of cell a. Entries involving a cell of size at most bs are computed directly during each query; no table of these entries is stored. Their number is at most

$$2 \sum_a n_a \sum_{\substack{b' \text{ neighboring } a \\ n_{b'} \le bs}} n_{b'} \le 2nbse^{O(k)} = n^{1+\tau/2+\sigma+o(1)} \le n^{2-\sigma+o(1)},$$

since $\tau/2 + 2\sigma < 1$. For two larger cells, both output dimensions exceed b, so padding final submatrices changes the work by only a constant factor. Summing their costs gives

$$n^{-\sigma+o(1)} \sum_{a,b'} n_a n_{b'} = n^{2-\sigma+o(1)}.$$

This upper bound includes both orientations and pairs consisting of one cell. Only one submatrix's sample outputs and feature matrices are kept at a time. Together with the input, cell lists, polynomials, and output vector, the space is $n^{\tau+o(1)}$.

At most n^2 row–group events need recovery, since the database groups for each output row are disjoint. Lemma 3 therefore gives failure probability at most n^{-10}. The partition, centers, and polynomials stay fixed; recomputation changes neither the surrogate matrix nor the event covering every later weight vector.

The same calculation also retains a previously chosen discrete exponent. If $R \le n^{\nu+o(1)}$, a submatrix width n^β, $1/2 \le \beta < 1$, suffices whenever

$$\nu + 2c < 0.32\beta, \qquad \sigma + \nu + 2c < \beta, \qquad \beta + 2\sigma < 1.$$

The space is then $n^{2\beta+o(1)}$, with unchanged query time. For example, put $\gamma_0 = 0.32 - 2c > 0$. With $R \le n^{\gamma_0/2}$ and $\sigma < \min\{1/10, \gamma_0/4\}$, take $\beta = 1 - \gamma_0 - \sigma$. All three inequalities hold. This gives space $n^{2-2\gamma_0-2\sigma+o(1)}$ without reducing σ or strengthening the corresponding alphabet separation.

Keeping the exact spectral ratio gives training time

$$n^{2-\sigma+o(1)} \left(\frac{1+\rho}{1-\rho}\right)^{d/2} \log \frac{n}{\mu\varepsilon}, \qquad \rho = \max_r \sum_{s \ne r} e^{-(b_r - b_s)^2}, \tag{39}$$

whenever the local feature count fits the stated matrix dimensions. At error ε_n and fixed $\rho < 1$, a positive saving thus only requires $(C/2)\log((1+\rho)/(1-\rho)) < \sigma$.

For fixed $a > 1$ and $\varepsilon = \exp[-(\log n)^a]$, replace γ by $0.16\tau - (a-1)c > 0$. Now $L = O(\log^a n)$ and $r \le n^{(a-1)c+o(1)}$; the remaining bounds are unchanged. For $k = o(\log n / \log\log n)$, including fixed dimension, $r = n^{o(1)}$. One can retain any admissible discrete choice with $R \le n^{0.1}$ and $\sigma < 1/10$, using $b = \lceil \sqrt{n} \rceil$: $Rr \le b^{0.32}$, $sRr \le b$, and the small-cell work is $n^{3/2+\sigma+o(1)} \le n^{2-\sigma+o(1)}$. Thus these additional continuous coordinates preserve both the discrete time exponent and near-linear space, regardless of their magnitudes.

H Repeated discrete parts

Let $I_1, \dots, I_t$ group samples with the same discrete part, and set $b_g = |I_g|$, $b_0 = \max_g b_g$, and

$$D = \mathrm{blockdiag}(H_{I_1 I_1}, \dots, H_{I_t I_t}).$$

For $H_g = H_{I_g I_g}$ and $Y_g = \mathrm{diag}(y_{I_g})$, let

$$\mathcal{V}_g = \sup_{a \geq 0} \{\mathbf{1}^\top a - \tfrac{1}{2} a^\top Y_g H_g Y_g a\}$$

be the group's no-bias hard-margin dual value. These values measure label separation within each group; the relation between margins and SVM dual geometry is classical (Bennett & Bredensteiner, 2000). The algorithm uses a stated upper bound on $\mathcal{V}_g$, without computing these optima. Block preconditioning (Giselsson & Boyd, 2014; Cutajar et al., 2016) controls the iteration count, and an interior-point method (Nesterov & Nemirovskii, 1994) solves each projection to the accuracy required by (34).

Proposition 9. *The guarantees of Theorem 7, with the same σ, hold when its distinctness assumption is replaced by*

$$\sum_g b_g^{\omega_0 + 1/2} \leq n^{2-\sigma+o(1)}, \qquad b_0^2 \leq n^{\tau+o(1)}, \qquad \omega_0 = \log_2 7, \tag{40}$$

provided at least one of the following holds: (i) $U < \infty$ and $\log(2+U) = O(\log^3 n)$; (ii) $\mathcal{V}_g \leq T := e^{a_0 (\log n)^3}$ for every g, where $a_0 > 0$ is fixed. Case (ii) allows any $U \in (0, \infty]$.

In particular, every group may have size $O(n^\nu)$, where $\nu = (1-\sigma)/(\omega_0 - 1/2) > 0.41$. Indeed, $\sum_g b_g^{\omega_0+1/2} \leq n b_0^{\omega_0 - 1/2}$ and $2\nu < 1 \leq \tau$. The simpler condition $\sum_g b_g^2 \leq n^{1+o(1)}$ also implies (40), since $(\omega_0 + 1/2)/2 < 2 - \sigma$. Here we use $\sigma \leq 1/20$, as chosen in Appendix G.

Case (ii) includes constant labels, for which $\mathcal{V}_g \leq b_g/2$, and $H_g \succeq \zeta I$, for which $\mathcal{V}_g \leq b_g/(2\zeta)$. It also allows mixed labels with singular kernels. For example, take $z = (0, h, 2, 2+h)$, $y = (1, 1, -1, -1)$, and $0 \leq h \leq 1/4$. With $\gamma = e^{-1/16} - e^{-49/16} > 0$, the vector $c = (1, 0, -1, 0)^\top/\gamma$ satisfies $Y_g H_g c \geq \mathbf{1}$. Weak duality therefore gives

$$\mathcal{V}_g \leq \tfrac{1}{2} c^\top H_g c = \frac{1 - e^{-4}}{\gamma^2}.$$

This bound is independent of h, whereas $\lambda_{\min}(H_g) \leq 1 - e^{-h^2} \to 0$.

Proof. Let G be the kernel on distinct discrete vectors and S the incidence matrix, so $A = S^\top G S$. The discrete spectral bounds and the Schur product theorem give

$$\mu S^\top S \preceq A \preceq \Lambda S^\top S \quad \Longrightarrow \quad \mu D \preceq K \preceq \Lambda D, \qquad \|A\|_2 \leq b_0 \Lambda. \tag{41}$$

This holds even when D is singular.

First restrict the feasible set to a finite box $[0, V]^n$ without changing its optimum. In case (i), take $V = U$. In case (ii), write $F_g(a) = \mathbf{1}^\top a - a^\top Y_g H_g Y_g a/2$ and $B_* = \mu^{-1} \sum_g \mathcal{V}_g \leq tT/\mu$. Equation (41) gives, for all $\alpha \geq 0$,

$$F(\alpha) \leq \mu^{-1} \sum_g F_g(\mu \alpha_g) \leq B_*. \tag{42}$$

The identity $\mathbf{1}^\top \alpha = 4F(\alpha/2) - F(\alpha)$ bounds $\{\alpha \in \mathcal{C} : F(\alpha) \geq 0\}$ by $\mathbf{1}^\top \alpha \leq 4B_*$, so an optimum exists. At an optimum, the feasible direction toward zero gives $\alpha^{*\top} Q \alpha^* \leq \mathbf{1}^\top \alpha^*$, hence $\mathbf{1}^\top \alpha^* \leq 2F(\alpha^*) \leq 2B_*$. Thus $V = \min\{U, 2tT/\mu\}$ suffices. In both cases, $\log(2+V) = O(\log^3 n)$.

The same comparison also quantifies the group condition: for hard margins without bias, taking suprema in $\Lambda^{-1} \sum_g F_g(\Lambda \alpha_g) \leq F(\alpha) \leq \mu^{-1} \sum_g F_g(\mu \alpha_g)$ gives

$$\frac{\sum_g \mathcal{V}_g}{\Lambda} \leq \mathrm{OPT} \leq \frac{\sum_g \mathcal{V}_g}{\mu}.$$

Write $V_+ = \max\{1, V\}$ and choose

$$\lambda_0 = \frac{\varepsilon_n}{2nV_+^2}, \qquad \eta = \frac{\lambda_0}{2b_0\Lambda}, \qquad P = Y\left(D + \frac{\lambda_0}{\mu} I\right) Y. \tag{43}$$

For the symmetric entrywise η-approximation $\widehat{H}$ of Section 6, put $\overline{K} = A \circ \widehat{H} + 3\lambda_0 I/2$ and $Q_0 = Y\overline{K}Y$. Lemma 8's error estimate gives $\|A \circ (\widehat{H} - H)\|_2 \le b_0 \Lambda \eta = \lambda_0/2$, hence

$$K + \lambda_0 I \preceq \overline{K} \preceq K + 2\lambda_0 I, \qquad \mu P \preceq Q_0 \preceq 2\Lambda P. \tag{44}$$

The original optimum satisfies $\lambda_0 \|\alpha^*\|^2 \le \varepsilon_n/2$. By (20), it suffices to solve the corrected objective to error $\varepsilon_n/2$.

Use the metric P, $L_P = 2\Lambda$, and $h = \lceil \sqrt{8L_P/\mu} \rceil$. For the extrapolated point b_j, the projection problem is

$$\ell_j = Pb_j + \frac{\mathbf{1} - Q_0 b_j}{L_P}, \qquad J_j(a) = \tfrac{1}{2} a^\top P a - \ell_j^\top a, \qquad a \in \mathcal{C}_V, \tag{45}$$

where $\mathcal{C}_V$ is the capped feasible set. Return a feasible a_j with

$$\nabla J_j(a_j)^\top (a_j - z) \le \theta \quad (z \in \mathcal{C}_V), \qquad \theta = \frac{\varepsilon_n}{8 L_P h}. \tag{46}$$

In the coordinates $P^{1/2}a$, condition (46) is the update condition in Appendix E with $\Lambda = L_P$ and $\Delta = L_P\theta$; the algorithm does not compute $P^{1/2}$. Thus (34) gives $v_{\text{next}} \le v/2 + \varepsilon_n/8$ for the corrected objective gap. Initially $v \le nV$, since $Q_0 \succeq 0$. Therefore $O(\log(2 + nV/\varepsilon_n)) = O(\log^3 n)$ restarts give corrected error $\varepsilon_n/2$, using $n^{\sigma/8+o(1)}$ steps.

Section H.1 solves each projection in $\sum_g b_g^{\omega_0+1/2} \operatorname{polylog} n$ time and $O(n + b_0^2)$ working space, forming one kernel block at a time and reusing its storage. Multiplication by P similarly costs $O((1+k)\sum_g b_g^2)$ time and $O(n + b_0^2)$ space. These costs meet the asserted bounds by (40). Preprocessing only forms the group indices and the kernel oracle; the block construction is charged to the optimization steps. Since $\log(1/\eta) = O(\log^3 n)$, the local expansions and kernel products retain their previous bounds. All gradient queries use the same success event. □

H.1 Solving the block projection to the required accuracy

Rescale $a = Vx$ and write $j(x) = J(Vx) = V^2 x^\top P x/2 - V\ell^\top x = \sum_g j_g(x_g)$ on $[0,1]^n$. Set $\beta = b_0 + \lambda_0/\mu \ge \|P\|_2$ and $M = 1 + nV^2\beta + V\|\ell\|$, so $\|\nabla j(x)\| \le M$. Dual decomposition (Boyd et al., 2011, Section 2.2) leaves only one scalar coupling the groups. For any $|\lambda| \le M + \theta$, the independent block solves below return x_λ satisfying

$$\big(\nabla j(x_\lambda) + \lambda y\big)^\top (x_\lambda - z) \le \theta/2 \qquad (z \in [0,1]^n). \tag{47}$$

Without bias, take $\lambda = 0$ and return Vx_λ.

One scalar for the bias. Let n_+, n_- count the labels. If either count is zero, return zero. Otherwise, at $\lambda_- = -M - \theta$ and $\lambda_+ = M + \theta$, (47) gives $G_- := y^\top x_{\lambda_-} \ge n_+ - 1/2 > 0$ and $G_+ := y^\top x_{\lambda_+} \le -n_- + 1/2 < 0$. These bounds follow by comparing with the box vertices maximizing and minimizing $y^\top z$, using $|(\nabla j)_i| \le M$. Bisect this interval, retaining endpoints with opposite signs of $G_\lambda = y^\top x_\lambda$, until its width is at most $\theta/(2n)$. If $G_\lambda = 0$, return Vx_λ. Otherwise set

$$x = q x_{\lambda_-} + (1-q) x_{\lambda_+}, \qquad q = \frac{-G_+}{G_- - G_+}.$$

Then $y^\top x = 0$ exactly. For every feasible z, the quadratic identity below and (47) give

$$\begin{aligned} \nabla j(x)^\top (x - z) &\le q \nabla j(x_{\lambda_-})^\top (x_{\lambda_-} - z) + (1-q)\nabla j(x_{\lambda_+})^\top (x_{\lambda_+} - z) \\ &\le \theta/2 + q G_- (\lambda_+ - \lambda_-) \le \theta. \end{aligned}$$

The first right side minus its left side is $q(1-q)V^2\|x_{\lambda_-} - x_{\lambda_+}\|_P^2 \ge 0$. Thus Vx meets (46), using $O(\log(2 + nM/\theta))$ block-solve calls. This argument only uses the certificates, so the block minimizers need not be exact.

A Newton step certifies each block. Fix a block of size b, put $P_g = P_{I_g I_g}$, and consider $\psi(u) = j_g(u) + \lambda y_g^\top u$. Use $\varphi(u) = -\sum_i(\log u_i + \log(1-u_i))$. At a point with Newton decrement at most $1/4$ for $\varphi + s\psi$, let d be the full Newton step and put $C = \operatorname{diag}(u_i^{-2} + (1-u_i)^{-2})$. Then $u + d \in (0,1)^b$, and the following vectors are positive:

$$p_i = \frac{u_i^{-1} - d_i u_i^{-2}}{s}, \qquad r_i = \frac{(1-u_i)^{-1} + d_i(1-u_i)^{-2}}{s}.$$

Because ψ is quadratic, the Newton equation gives $\nabla\psi(u+d) = p - r$ exactly. Hence, for all $z \in [0,1]^b$,

$$\nabla\psi(u+d)^\top(u+d-z) \le (u+d)^\top p + (\mathbf{1}-u-d)^\top r = \frac{2b - d^\top C d}{s} \le \frac{2b}{s}. \tag{48}$$

Here $d^\top C d \le 1/16$ implies $|d_i| < \min\{u_i, 1-u_i\}$, proving the stated positivity. Choosing $s = 4n/\theta$ and summing over blocks proves (47).

Starting at $u = \mathbf{1}/2$ and $s_0 = [200\sqrt{n}(M+\theta)]^{-1}$, short-step path following reaches this s in $O(\sqrt{b}\log(2+nM/\theta))$ Newton steps (Nesterov & Nemirovskii, 1994; Boyd & Vandenberghe, 2004). Indeed, $\|\nabla\psi\| \le 2\sqrt{n}(M+\theta)$ and $\nabla^2\varphi \succeq 8I$ give a constant initial Newton neighborhood. The box barrier has parameter $2b$, and adding a convex quadratic preserves self-concordance.

Each Newton matrix is $sV^2P_g + C \succ 0$. Recursive inversion of its leading block and positive definite Schur complement uses Strassen products (Strassen, 1969), giving $T(b) = 2T(b/2) + O(b^{\omega_0})$ and, with successive evaluation and storage reuse, $W(b) = W(b/2) + O(b^2)$. Identity padding handles arbitrary block sizes. For each λ, form and solve one block at a time, retaining only its output vector. Including kernel construction, the work is $\sum_g O(kb_g^2 + b_g^{\omega_0+1/2}\log(2+nM/\theta))$ and the space is $O(n + b_0^2)$. Finally, (45) gives $\|\ell_j\| \le 6\beta V\sqrt{n} + \sqrt{n}/L_P$, so $\log(2+nM/\theta) = O(\log^3 n)$. Including the scalar bisection proves the claimed projection cost.

I A UNIFORM LOW-RANK OBSTRUCTION

The following elementary example distinguishes fast products from uniformly accurate low-rank replacement of a quadratic objective.

Proposition 10. *Let $n = 2^d$, $x_u = \sqrt{B/d}\,u$ for $u \in \{0,1\}^d$, and $B \ge 1$. Set $\mu = (1 - e^{-B/d})^d$. Then*

$$\mu \ge (2d)^{-d} = \exp[-O(\log n \log\log n)].$$

Every matrix $\widetilde{K}$ of rank below n has $\|K - \widetilde{K}\|_2 \ge \mu$. Moreover, for balanced labels y, let $\mathcal{C} = \{\alpha \in [0,2]^n : y^\top\alpha = 0\}$. Any quadratic surrogate $\widetilde{F}$ whose symmetric Hessian has rank $r \le n-2$ satisfies

$$\sup_{\alpha\in\mathcal{C}} |F(\alpha) - \widetilde{F}(\alpha)| \ge \frac{\mu(n-r-1)}{4}, \tag{49}$$

regardless of its linear or constant term.

Proof. The tensor spectrum gives $K \succeq \mu I$, and $1 - e^{-1/d} \ge 1/(2d)$ proves the lower bound on μ. A unit vector in $\ker\widetilde{K}$ proves the spectral claim. For the objective claim, let E be the intersection of the surrogate Hessian's nullspace with $y^\perp$, so $\dim E \ge n-r-1$. An extreme point w of $E\cap[-1,1]^n$ has at least $\dim E$ coordinates equal to ± 1: otherwise a nonzero direction in E vanishing on those coordinates would permit a small displacement in both directions. Thus $\|w\|^2 \ge n-r-1$. The points $\mathbf{1}, \mathbf{1}+w, \mathbf{1}-w$ all lie in $\mathcal{C}$. Writing $D_F = F - \widetilde{F}$, the second difference is

$$D_F(\mathbf{1}+w) + D_F(\mathbf{1}-w) - 2D_F(\mathbf{1}) = -w^\top Q w.$$

Its magnitude is at least $\mu(n-r-1)$ and at most $4\sup_{\mathcal{C}}|D_F|$, proving (49). □

Thus, for all sufficiently large n, uniform error ε_n requires rank at least $n-1$, even when $B = 1$. This statement is about uniform quadratic approximation. It is not a lower bound on the time needed to find one nearly optimal point. In particular, the blockwise recovery in Section 3 and the tensor multiplication below remain compatible with this obstruction.

J Deterministic comparisons and an exact bias identity

J.1 Two deterministic bounds

When the full cube is small, direct tensor multiplication is preferable. Embed $v \in \mathbb{R}^n$ in a vector indexed by $\{0,1\}^d$, with zeros at unobserved vertices. Along each coordinate, replace every pair by

$$(v_0, v_1) \longmapsto (v_0 + qv_1,\ qv_0 + v_1). \tag{50}$$

After d passes, restrict back to the observed points. This computes Kv in $O(2^d d)$ operations by the same tensor identity as Lemma 4.

Proposition 11. *For distinct Boolean inputs and $q \le 1/(4d)$, all four SVM formulations can be trained to error ε deterministically in*

$$O\big((2^d d + n\log n)\log(n/\varepsilon)\big) \tag{51}$$

operations and $O(2^d + n)$ space.

Proof. Bernoulli's inequality gives $(1-q)^d \ge 1 - dq \ge 3/4$, while $(1+q)^d \le e^{dq} < 3/2$. Thus $\kappa < 2$. Combine (50), Lemma 5, and the projection cost. □

For a full cube, $n = 2^d$ and $a^2 = \log(4d)$ yield $B = \Theta(\log n \log\log n)$ and time $O(n\log^4 n)$ at ε_n. This is a useful special case, but when $2^d = n^{10}$ the explicit grid is too large. Theorem 1 avoids this dependence on the number of unobserved points.

For arbitrary real inputs at smaller dimension, familiar Taylor features also suffice. Appendix K proves that, for fixed $0 < c < 1/8$,

$$d \le \frac{c\log n}{\log\log n}, \quad B \le (\log n)^2, \quad \alpha \in [0,2]^n, \quad y^\top \alpha = 0 \implies \text{time } n^{3/2+4c+o(1)} \tag{52}$$

at error ε_n. The proof combines positive semidefinite Taylor features (Cotter et al., 2011) with a classical interior-point method (Nesterov & Nemirovskii, 1994). This is a parameter-specific consequence of established methods, included to compare input restrictions.

J.2 Transferring between the two bias formulations

The preceding algorithm treats bias directly by projection. A separate identity also relates the optimal values with and without bias. It uses the sign-reversed copies familiar from Backurs et al. (2017, Appendix C), but chooses their extra coordinate so that the cross-kernel block is an exact multiple of the original.

Proposition 12. *Given a no-bias hard-margin instance with finite optimum V, replace each (x_i, y_i) by $((x_i, 1), y_i)$ and $((x_i, -1), -y_i)$, and add the bias constraint. Write $\tau = e^{-4}$ and $\lambda = 1 - \tau$. The new instance has optimum $V' = 2V/\lambda$, twice as many points, one extra coordinate, and squared diameter $B + 4$. An ε'-optimal nonnegative pair (u, v) yields $z = \lambda(u+v)/2$ with original objective error at most $\lambda\varepsilon'/2$.*

Proof. The new matrix and equality are

$$Q' = \begin{pmatrix} Q & -\tau Q \\ -\tau Q & Q \end{pmatrix}, \qquad y^\top(u - v) = 0.$$

With $p = (u+v)/2$ and $r = (u-v)/2$, direct expansion gives

$$F'(u,v) = \frac{2}{\lambda} F(\lambda p) - (1+\tau) r^\top Q r. \tag{53}$$

Since $Q \succeq 0$, $F'(u,v) \le 2V/\lambda$. Taking $u = v = \alpha/\lambda$ for any original feasible α gives $F'(u,v) = 2F(\alpha)/\lambda$ and satisfies the bias constraint. Taking suprema proves equality of the optimal values. The same identity gives the error conversion. Cross-copy squared distances are $\|x_i - x_j\|^2 + 4$, proving the diameter claim. □

The identity also holds for soft margin if the copy instance uses box cap C_{box}/λ; both maps then preserve their boxes. It transfers value-approximation reductions with constant changes in size, scale, and tolerance. In particular, applying it to the no-bias hardness statement of Gu et al. (2025, Theorem G.12) preserves its $\Theta(\log^2 n)$ squared-diameter scale for the biased formulation. This consequence uses the displayed identity, rather than the larger scale of their published bias corollary. It does not assert a common positive exponent saving for all logarithmic dimension constants.

K TAYLOR FEATURES AT SMALLER DIMENSION

Proposition 13. *Fix $0 < c < 1/8$. For arbitrary real inputs with $d \le c \log n / \log\log n$ and $B \le (\log n)^2$, the problem with $0 \le \alpha_i \le 2$ and $y^\top \alpha = 0$ can be solved to additive error ε_n in $n^{3/2+4c+o(1)}$ deterministic arithmetic operations.*

Proof. Translate so that $x_1 = 0$, and hence $\|x_i\|^2 \le B$. The Gaussian Taylor features (Cotter et al., 2011) are

$$\phi_\nu(x) = e^{-\|x\|^2} \sqrt{\frac{2^{|\nu|}}{\nu!}}\, x^\nu, \qquad \nu \in \mathbb{Z}^d_{\ge 0}, \quad |\nu| \le p.$$

There are $k = \binom{d+p}{p}$ such features. If $\widetilde{K} = \Phi\Phi^\top$, the omitted features form a positive semidefinite residual $E = K - \widetilde{K}$ with

$$E_{ii} = \Pr[\mathrm{Pois}(2\|x_i\|^2) > p], \qquad |E_{ij}| \le \sqrt{E_{ii}E_{jj}}.$$

This reduces the entire entrywise error estimate to one scalar tail. Set $\eta = \varepsilon_n/(32n^2)$ and choose $m = p + 1 > 2eB$ with $m \log(m/(2eB)) \ge \log(1/\eta)$, treating $B = 0$ directly. The Poisson Chernoff bound gives $|E_{ij}| \le \eta$. Writing $\ell = \log n$, we can take $p = O(\ell^3/\log \ell)$, so

$$\log k \le d \log \frac{e(d+p)}{d} \le (2c + o(1)) \log n.$$

On the box, the objective perturbation is at most $\eta\|\alpha\|_1^2/2 \le \varepsilon_n/16$. Since $\widetilde{F} \ge F$, any feasible $\widehat{\alpha}$ satisfies

$$\mathrm{OPT} - F(\widehat{\alpha}) \le \widetilde{\mathrm{OPT}} - \widetilde{F}(\widehat{\alpha}) + \varepsilon_n/16.$$

Thus surrogate error $\varepsilon_n/2$ suffices.

The optimization cost follows from short-step path following (Nesterov & Nemirovskii, 1994) and the low-rank Newton solves used in SVM training by Fine & Scheinberg (2001). Let $Z = Y\Phi$ and use the box barrier

$$\varphi(\alpha) = -\sum_i \big(\log \alpha_i + \log(2 - \alpha_i)\big)$$

on the equality subspace. Its barrier parameter is $2n$. If both labels occur n_+, n_- times, its analytic center is

$$z_i = \begin{cases} 2n_-/n, & y_i = 1, \\ 2n_+/n, & y_i = -1. \end{cases}$$

Indeed, $y^\top z = 0$ and $\nabla\varphi(z)$ is a multiple of y. Each coordinate is at least $2/n$ from a box boundary. If only one label occurs, return zero.

Follow the objectives $\varphi + \tau\widetilde{f}$ from $\tau_0 = 1/(100n^2)$ to $\tau = \Theta(n/\varepsilon_n)$. Since $\widetilde{K}_{ii} \le 1$, its spectral norm is at most n, $\|\nabla\widetilde{f}\| = O(n^{3/2})$ on the box, and $\nabla^2\varphi \succeq 2I$. Thus z starts in a constant Newton neighborhood. Short-step path following takes $O(\sqrt{n}\log(n/\varepsilon_n))$ steps. Adding the convex quadratic preserves self-concordance because it has zero third derivative.

After rescaling, a Newton matrix is $D + ZZ^\top$ for positive diagonal D. Woodbury's identity reduces its solve to a $k \times k$ system: forming $Z^\top D^{-1} Z$ costs $O(nk^2)$ and factoring it costs $O(k^3)$. The equality requires a scalar Schur complement and a constant number of additional solves. Since $k < n$, the total is $n^{3/2}k^2 \operatorname{polylog} n = n^{3/2+4c+o(1)}$. □

Using instead the low-rank quadratic-programming solver of Gu et al. (2025) with this positive semidefinite feature factor gives the expected bound $n^{1+c(\omega+1)+o(1)}$, where ω is the square matrix-multiplication exponent. This is a substitution into an existing solver, not a new optimization method. An approximate bias equality can be repaired by removing excess mass from the appropriate label class. The ℓ_1 change equals the equality residual; on the box the gradient has ℓ_∞ norm at most $1 + 2n$. Asking the solver for equality error $\varepsilon_n/O(n)$ therefore allows exact repair within the requested objective error.