# Efficient Tropical SVMs via Mean-Payoff Games


**Xavier Allamigeon**
Inria and CMAP
École polytechnique
Palaiseau, France

**Samuel Boïté**
Inria and CMAP
École polytechnique
Palaiseau, France

**Stéphane Gaubert**
Inria and CMAP
École polytechnique
Palaiseau, France

**Théo Molfessis**
Inria and CMAP
École polytechnique
Palaiseau, France


SG :title is very good. I hesitated with Scalable Tropical SVMs via Mean-Payoff Games? but perhaps effficient is stronger.


## Abstract

In 2006, Gärtner and Jaggi introduced a tropical analogue of support vector machines, using a single tropical hyperplane in dimension $n$ to separate $n$ classes of points. Efficiently computing tropical separators has remained an open problem. We introduce an algorithm for Tropical Support Vector Machines that overcomes the combinatorial explosion of previous approaches. Our main result shows that the spectral radius of a specially constructed Shapley operator fully characterizes separability and margin: data are tropically separable if and only if the spectral radius is negative, with an optimal margin, both in the separable and inseparable case, is determined by this spectral raidus. This provides a reduction to mean-payoff games, a well studied class of problems in algorithmic game theory. This approach enables computing of an optimal separating hyperplane via scalable iterative algorithms – with a complexity linear in the size of the data set and pseudo-polynomial in the desired precision. Finally, we combine tropical classifiers with linear feature maps to construct piecewise-linear classifiers.


SG: added linearity (otherwise pseudo may look a bit weak)

## 1 Introduction

Classification is a fundamental task in machine learning, and Support Vector Machines (SVMs) have been a cornerstone method for decades. Traditional SVMs create decision boundaries using affine hyperplanes, which provide maximum-margin separation with strong generalization guarantees [26]. However, these linear boundaries become limiting when faced with complex, nonlinear data patterns, typically requiring kernel methods or feature engineering [23].

**Motivation: Beyond Linear Boundaries.** We explore tropical algebra, a framework where standard addition becomes the maximum operation, and multiplication becomes addition [16]. This leads to different geometric structures with attractive properties for machine learning: (1) instead of creating binary partitions, tropical hyperplanes divide space into multiple sectors, making them naturally suited for multi-class problems; (2) their piecewise-linear nature captures more complex patterns while maintaining interpretability; (3) the resulting decision boundaries coincide with those created by modern deep learning models with ReLU activations [29]. These properties provide richer, yet interpretable, decision boundaries that can capture nonlinear patterns in data while maintaining computational tractability [17].

SG: explore → leverage?

SG: reinforced “coincide”

Tropical geometry has emerged as a powerful tool for modeling piecewise-linear phenomena in machine learning. Together with polyhedral geometry, it has been used to bound the number of linearity regions ot functions realized by these networks [29, 20]. It has been successfully applied to linear regression [18, 1], principal component analysis [28], neural network analysis [17], and clustering [19]. Related work by Fotopoulos et al. [10] has explored neural network compression through tropical geometry, demonstrating the broader utility of these techniques.

SG: added this

SG added [18]

Preprint. Under review.

**Previous Work on Tropical SVMs.** Gärtner and Jaggi [11] introduced tropical SVMs using linear programming formulations. Their work demonstrated that tropical hyperplanes provide improved locality in decision boundaries, where only nearby support vectors influence classification, unlike classical SVMs where all support vectors contribute globally. Despite these theoretical benefits, their method required exploring all possible sector assignment combinations, leading to exponential worst-case complexity. This computational barrier has severely limited practical applications. Tang et al. [25] later developed specialized algorithms for binary classification cases where data points from the same category stay in the same sector, showing promising results in computational biology for analyzing evolutionary trees.

SG: I am not sure it is a quality (less stable than L1), perhaps just mention that this is an elegant geometric approach to multi-class problem.

**From Game Theory to Machine Learning.** Our approach uses concepts from game theory—specifically, Shapley operators from stochastic games [24] and their adaptation to tropical algebra [15]. The spectral properties of these operators have been extensively studied for analyzing fixed points and convergence in nonlinear systems [12]. Mean-payoff games, a class of two-player zero-sum games, are closely linked to the spectral radius of such operators [30].

SG Delete (we need a broader state of the art on mean-payoff games – and the theorem there only concerns approximating $\rho$ in $1/\epsilon$, here we need to approximate the eigenvector which is in $1/\epsilon^2$): Importantly, recent advances in solving these games efficiently [4] provide efficient algorithms that we can apply to tropical classification. – Indeed, we need instead to use the relative KM algorithm along the lines of [2], with a complexity in $1/\epsilon^2$, but this algorithm provides the eigenvector. [4] does not (only the eigenvalue is approximated, we can get sub and super eigenvectors however with a preoprocessing). The termination condition should be changed in the algo below to $\|T(x) - x\|_H \leq \epsilon$.

**Contributions.** We reformulate tropical classification using mean-payoff games, addressing the fundamental computational limitations of previous approaches:

1. We establish a direct connection between tropical separability and the spectral radius $\rho(T)$ of a Shapley operator constructed from class-specific tropical projections.
2. We prove that when data are separable, the optimal margin equals $-\rho(T)$, and in non-separable cases, $\rho(T)$ quantifies exactly how much the data points would need to be perturbed to achieve separability.
3. We develop an algorithm based on mean-payoff games and Krasnoselskii–Mann iterations that computes the optimal classifier in pseudo-polynomial time rather than exponential time.
4. We extend our framework to tropical polynomial classifiers, enabling more expressive decision boundaries while preserving theoretical margin guarantees.

This work makes tropical SVMs tractable for real-world applications, enabling natural multi-class classification, and opening new directions for piecewise-linear methods that balance expressivity, interpretability, and computational efficiency.

The remainder of the paper is organized as follows. Section 2 introduces the essential concepts from tropical geometry. Section 3 presents our spectral framework and main theoretical results, showing how tropical separability connects to spectral properties. Section 4 details our algorithm and implementation, explaining how we achieve pseudo-polynomial complexity. Section 5 extends the framework to tropical polynomials for more expressive decision boundaries. Section 6 explores connections with classical SVMs, and Section 7 discusses limitations and future directions. The code to reproduce our figures and experiments is available at `https://github.com/samuelbx/tropical-svm`.

## 2 Tropical Geometry Preliminaries

We now introduce the key concepts from tropical geometry that form the foundation of our approach. To make these abstract concepts more accessible, we include intuitive explanations alongside formal definitions.

**The Max-Plus Semiring.** The tropical (or max-plus) semiring $\mathbb{T} = \mathbb{R} \cup \{-\infty\}$ replaces traditional arithmetic operations with:

$$x \oplus y = \max(x, y) \quad \text{(tropical addition)} \tag{1}$$

$$x \odot y = x + y \quad \text{(tropical multiplication)} \tag{2}$$

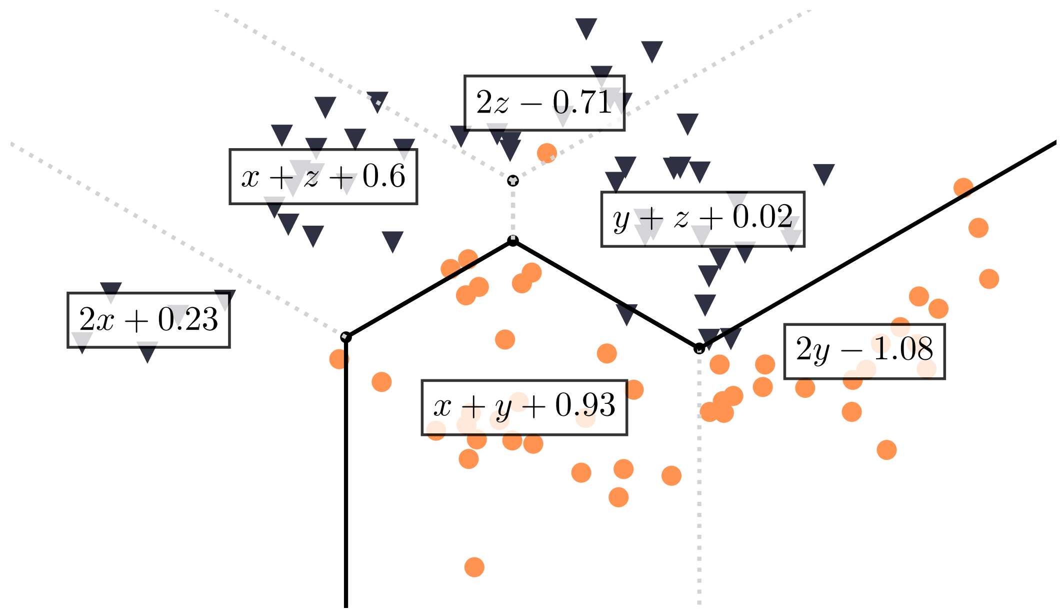



Figure 1: Visualization of a degree-2 tropical polynomial classifier. Each region corresponds to a sector where a specific affine combination of the features dominate, creating an interpretable piecewise-linear decision boundary. Inference remains computationally efficient, requiring only the evaluation of the dominant monomial at each test point.

These operations may seem strange at first, but they naturally model systems where we care about "bottlenecks" or "critical paths." For example, in project planning, if task A takes $x$ days and task B takes $y$ days, the project completion time depends on the maximum $(x \oplus y)$ of these durations if tasks are parallel, and their sum $(x \odot y)$ if sequential.

**Tropical Projective Space.** The tropical projective space identifies points that differ by adding the same constant to all coordinates. Formally, it's the quotient of $\mathbb{T}^d \setminus \{(-\infty, \ldots, -\infty)\}$ by the equivalence relation $x \sim y$ if $x = y + c \cdot \mathbf{1}$ for some constant $c$. In practice, we embed data from $\mathbb{R}^d$ into the projective space via:

$$x = (x_1, \ldots, x_d) \mapsto (x_1, \ldots, x_d, -(x_1 + \cdots + x_d))$$

This transformation makes our classifier invariant to shifts—adding the same constant to all features doesn't change the classification. It's similar to how projective geometry in computer vision makes analysis invariant to camera distance.

**Tropical Hyperplanes and Sectors.** A tropical hyperplane with apex $a \in \mathbb{T}^d$ is defined as:

$$\mathcal{H}_a = \left\{ x \in \mathbb{T}^d : \text{the maximum of } (x_i + a_i) \text{ over } 1 \leq i \leq d \text{ is attained at least twice} \right\} \tag{3}$$

This hyperplane divides the space into at most $d$ sectors. The $i$-th sector contains points where the maximum of $x + a$ occurs at the $i$-th coordinate:

$$S_i(a) = \{x \in \mathbb{T}^d : i \in \arg\max_j (x_j + a_j)\} \tag{4}$$

Unlike classical hyperplanes that create two half-spaces, tropical hyperplanes create multiple sectors—one for each dimension. They naturally support multi-class classification, where we can assign different sectors to different classes.

**Hilbert Seminorm and Tropical Distance.** The Hilbert seminorm in tropical geometry measures the "spread" of coordinates:

SG: I would delete "in tropical geometry"

$$\|x\|_H = \max_i x_i - \min_i x_i \tag{5}$$

This induces a projective distance $d_H(x,y) = \|x-y\|_H$ that remains invariant to adding the same constant to all coordinates [7]. We use this distance to define margins in tropical classification. For classification, it tells us how confidently a point belongs to its assigned sector rather than another sector.

**Tropical Convexity and Projections.** A set $C \subset \mathbb{T}^d$ is tropically convex if for all $x, y$ in $C$ and coefficients $\lambda, \mu$ in $\mathbb{T}$ with $\lambda \oplus \mu = 0$ (meaning $\max(\lambda, \mu) = 0$), the point $(\lambda \odot x) \oplus (\mu \odot y)$ is also in $C$ [7, 9]. The tropical convex hull of points $\{x_1, \dots, x_p\}$ is defined as:

$$\text{conv}_{\max}(X) = \left\{ \bigoplus_{i=1}^{p} \lambda_i \odot x_i : \lambda_i \in \mathbb{T}, \bigoplus_{i=1}^{p} \lambda_i = 0 \right\} \tag{6}$$

The tropical projection $P_X(y)$ of a point $y$ onto this convex hull is:

$$P_X(y) = \max\{z \in \text{conv}_{\max}(X) : z \leq y\} \tag{7}$$

SG. This is the projection on a cone, not on a convex set, perhaps we need to speak only of cones (modulo passing to the projective space), then this is ok. For a convex set, the max may be taken over a nonempty set

Tropical convexity generalizes the idea of conventional convexity to the max-plus setting. A tropical convex hull contains all tropical linear combinations of points. The projection finds the "closest" point in the convex hull that doesn't exceed our target point in any coordinate.

These tropical projections will play a central role in our classification framework. We will use them to build class-specific operators that characterize the separability of data.

## 3 Spectral Framework for Tropical SVMs

Having established the basics of tropical geometry, we now develop our spectral approach to tropical classification. The key insight is connecting the separability of data classes to the spectral properties of a specially constructed operator.

**Shapley Operators and Their Spectral Theory.** A Shapley operator $T : \mathbb{T}^d \to \mathbb{T}^d$ satisfies two fundamental properties [14]:

1. *Monotonicity:* If $x \leq y$ coordinatewise, then $T(x) \leq T(y)$ coordinatewise
2. *Additive homogeneity:* For any constant $\alpha \in \mathbb{R}$, $T(\alpha + x) = \alpha + T(x)$

The spectral radius of $T$ is defined as:

$$\rho(T) = \sup\{\lambda \in \mathbb{R} : \exists u \neq -\infty \text{ with } T(u) = \lambda + u\} \tag{8}$$

Equivalently, $\rho(T)$ is the smallest value $\lambda$ for which there exists a vector $u$ satisfying $T(u) \leq \lambda + u$ [21].

**Constructing the Classification Operator.** Consider a classification problem with $K$ classes, each represented by a set of points $X^1, \dots, X^K \subset \mathbb{T}^d$. We define an operator $T^k$ for each class $k$ by taking the tropical projection onto the convex hull of points in that class:

$$T^k(x) = P_{X^k}(x)$$

We then combine these operators into a single classification operator $T$ defined coordinatewise as:

$$T(x)_i = \max{}_2\{T^1(x)_i, \dots, T^K(x)_i\}$$

where $\max_2$ denotes the second-largest value among the outputs. For binary classification ($K = 2$), this simplifies to $T(x) = \min\{T^1(x), T^2(x)\}$.

**Main Theorem: Spectral Separability.** Our central result establishes the connection between tropical separability and the spectral radius:

SG: does not some part of the theorem (the last one?) requires to use the diagonal free variant?

**Theorem 1** (Spectral Separability Criterion)**.** *Let $X^1, \dots, X^K \subset \mathbb{T}^d$ be labeled point sets and let $T$ be the classification operator defined above. Then:*

*(1) Separability Criterion: The data are tropically separable by a hyperplane if and only if $\rho(T) < 0$.*

*(2) Margin Optimality: In the separable case, the maximum achievable margin equals $-\rho(T)$.*

*(3) Soft-Margin Interpretation: For binary classification with overlapping data, $\rho(T)$ is positive and quantifies the minimal perturbation needed to achieve separability.*

*Moreover, both $\rho(T)$ and an associated apex vector $a$ satisfying $T(a) = \rho(T) + a$ can be computed in pseudo-polynomial time using mean-payoff game algorithms (see Section 4).*

*Proof Sketch.* The proof follows two main directions. First, if there exists a tropical hyperplane $\mathcal{H}_a$ separating the classes with margin $\gamma > 0$, then using the monotonicity and homogeneity properties of the operator, we can show $T(a) \leq a - \gamma$. This implies $\rho(T) \leq -\gamma < 0$.

Conversely, if $\rho(T) < 0$, we can find an eigenvector $a$ satisfying $T(a) = \rho(T) + a$. Using this apex, we can construct a separating hyperplane by assigning sectors based on which class operator dominates at each coordinate. This hyperplane achieves a margin of exactly $-\rho(T)$.

For the non-separable case, we show that $\rho(T)$ quantifies the minimum perturbation required to make the data separable by constructing an explicit perturbation of magnitude $\rho(T)$.

A complete proof is provided in Appendix A.

**Benefits of the Spectral Approach.** Our framework significantly advances the tropical SVM foundation established by Gärtner and Jaggi [11]. Their work showed that tropical SVM could be formulated as finding a point of maximum margin within a tropical polytope defined by sector constraints, but required exhaustive exploration of sector assignments—leading to exponential complexity.

In contrast, our spectral characterization offers a complete theoretical understanding of when data are tropically separable, an exact formula for the optimal margin, an efficient algorithm to find the optimal classifier without combinatorial exploration and a natural interpretation for non-separable cases.

**Limitations.** The primary limitation of our spectral approach is its sensitivity to outliers. Since tropical convex hulls encompass all points in each class, a single misplaced point can significantly alter the classification boundary and potentially render previously separable data inseparable.

Additionally, because our method is formulated as a spectral criterion rather than an optimization problem, it's challenging to incorporate relaxation mechanisms that would handle misclassified points in a controlled way. Traditional SVMs use slack variables to allow soft margins and tolerate some misclassifications, but our current approach doesn't have a direct equivalent. This suggests an important direction for future research in tropical classification.

## 4 Algorithm and Implementation

Having established the theoretical foundation, we now present our algorithm for tropical SVM based on the spectral criterion. The key insight is that we can compute the spectral radius and optimal hyperplane efficiently without exploring all possible sector assignments.

**Efficient Computation of Tropical Projections.** The first step is to compute the tropical projections efficiently. The tropical projection $P_X(y)$ of a point $y$ onto the tropical convex hull of a set $X \subset \mathbb{T}^d$ can be computed using:

$$P_X(y)_i = \max_{1 \leq j \leq p} \left\{ X_{ij} + \min_{1 \leq k \leq d} (-X_{kj} + y_k) \right\}$$

where $p$ is the number of points in $X$ and $X_{ij}$ is the $i$-th coordinate of the $j$-th point.

This formulation can be viewed as a mean-payoff game [1] and computed in $\mathcal{O}(pd)$ time—linear in both the number of points and dimensions.

**Computing the Spectral Radius with Krasnoselskii-Mann Iterations.** To compute the spectral radius $\rho(T)$ and a corresponding eigenvector $a$, we apply a Krasnoselskii-Mann iteration scheme [21, 12], outlined in Algorithm 1.

SG: very good to quote [21, 12] but KM is not there, we should rather quote the original works of Baillon and Bruck [5] and perhaps followup by cominetti et al, and see [2, 3] for the relative KM which we use here.

**Algorithm 1** Krasnoselskii–Mann Iteration for Tropical SVM

**Input:** Shapley operator $T$, convergence threshold $\varepsilon > 0$
**Initialize:** $x^{(0)} \in \mathbb{R}^d$ (typically set to $\mathbf{1}_d$), $\lambda^{(0)} = 0$
**for** $k = 0, 1, 2, \ldots$ until convergence **do**
  $z^{(k+1)} \leftarrow \frac{1}{2}\big(x^{(k)} + T(x^{(k)})\big)$ {Average the current point with its image to stabilize}
  $x^{(k+1)} \leftarrow z^{(k+1)} - \max\big(z^{(k+1)}\big)\, \mathbf{1}_d$ {Project onto the tropical projective space}
  $\lambda^{(k+1)} \leftarrow 2\max\big(z^{(k+1)}\big) - \max\big(x^{(k)}\big)$ {Estimate the current spectral radius}
  **if** $|\lambda^{(k+1)} - \lambda^{(k)}| < \varepsilon$ {Check if the spectral radius has stabilized} **then**
    **break**
  **end if**
**end for**
**Return:** $\rho(T) \approx \lambda^{(\text{final})}$, $a \approx x^{(\text{final})} - \frac{1}{d}\sum_{i=1}^{d} x_i^{(\text{final})} \cdot \mathbf{1}_d$

SG: to be updated

Recent results by Allamigeon et al. [4] show that this iteration achieves an $\varepsilon$-approximation in $O(1/\varepsilon)$ iterations. By avoiding the combinatorial exploration of sector assignments, our method runs in pseudo-polynomial time—specifically $O\big(\frac{nd}{\varepsilon}\big)$ for $n$ data points in $d$ dimensions.

**The Complete Tropical SVM Algorithm.** Algorithm 2 presents our complete procedure for tropical SVM classification.

**Algorithm 2** Tropical SVM

**Input:** Labeled point sets $X^1, \ldots, X^K \subset \mathbb{T}^d$
**Construct Operators:**
  For each class $k$, compute the tropical projection operator $T^k(x) = P_{X^k}(x)$
  Combine via $T(x)_i = \max_2\{T^1(x)_i, \ldots, T^K(x)_i\}$ for each coordinate $i$
**Compute Spectral Properties:**
  Run Algorithm 1 to obtain $\rho(T)$ and apex $a$
**Assign Sectors to Classes:**
**if** $\rho(T) < 0$ {Separable case} **then**
  Assign coordinate $i$ to class $k$ if $T^k(a)_i > \rho(T) + a_i$ {Assignment based on dominant operator}
**else**
  Use a heuristic (e.g., majority vote) for non-separable data {Fall back to best-effort assignment}
**end if**
**Output:** Apex $a$, margin $-\rho(T)$ (if separable), and sector assignments for classification

**Empirical Performance Analysis.** We validated the computational complexity of our algorithm through empirical testing (detailed in Appendix B). Our implementation achieves tractable performance on standard benchmark datasets, with performance scaling linearly with both dataset size and feature dimension as predicted by our theoretical analysis.

For the first time, we demonstrate that tropical SVMs can be practically computed with pseudo-polynomial guarantees, removing the exponential barrier that limited previous approaches. While specialized classical SVM libraries remain faster on conventional tasks, our approach enables new applications where piecewise-linear decision boundaries offer advantages.

Future optimizations could include kernel methods, sparse representations for large datasets, and parallel implementations to further improve performance on high-dimensional problems.

# 5 Tropical Polynomials for Enhanced Expressivity

While tropical hyperplanes provide effective classification boundaries, we can achieve even greater expressivity by extending our framework to tropical polynomials. These polynomials create more flexible decision boundaries while maintaining theoretical guarantees.

**Tropical Polynomial Kernel.** A tropical polynomial in $\mathbb{T}^d$ takes the form:

$$f(x) = \max_{\alpha \in A}(c_\alpha + \langle \alpha, x \rangle) \tag{9}$$

where $A \subset \mathbb{Z}^d$ is a finite set of integer vectors (monomials), $c_\alpha \in \mathbb{R}$ are coefficients, and $\langle \alpha, x \rangle = \alpha_1 x_1 + \cdots + \alpha_d x_d$ is the tropical dot product.

To fit such polynomials, we use a feature map $\Phi_A : \mathbb{T}^d \to \mathbb{T}^{|A|}$ defined by:

$$[\Phi_A(x)]_\alpha = \langle \alpha, x \rangle \tag{10}$$

This feature map transforms the original data into a higher-dimensional space where each coordinate corresponds to a monomial term. A tropical hyperplane in this feature space corresponds to a tropical polynomial in the original space. This is conceptually similar to the kernel trick in classical SVMs, but with an explicit feature mapping based on tropical algebra.

**Strategic Monomial Selection.** The choice of monomial set $A$ critically affects both the expressivity and computational complexity of the resulting classifier. We explore two complementary approaches:

1. **Homogeneous Monomials:** We use all monomials of degree $s$, defined as $A_s = \{\alpha \in \mathbb{N}^d : \sum_i \alpha_i = s\}$. The number of such monomials is $\binom{s+d-1}{s}$, which grows polynomially with $s$ for fixed dimension $d$.
2. **Adaptive Selection:** We sample pairs of points from different classes and construct monomials corresponding to the slopes of separating lines between them. This approach focuses computational resources on the most discriminative monomials.

The degree parameter $s$ controls the trade-off between expressivity and overfitting. Higher degrees create more flexible boundaries but may overfit the training data. Cross-validation can guide this selection process.

Importantly, only a sparse subset of monomials typically becomes active in the final classifier, making the approach computationally efficient even with many candidate monomials.

**Classification with Tropical Polynomials.** To perform classification with tropical polynomials:

1. Map the original data to the feature space: $\Phi_A(X^k) = \{\Phi_A(x) : x \in X^k\}$ for each class $k$.
2. Apply the tropical SVM algorithm in this feature space to find apex $a \in \mathbb{T}^{|A|}$ and spectral radius $\rho(T)$.
3. The classifier in the original space is the tropical polynomial:

$$f_a(x) = \max_{\alpha \in A}(-a_\alpha + \langle \alpha, x \rangle) \tag{11}$$

4. Classify new points by identifying which sector of $f_a(x)$ they fall into.

**Margin Guarantees for Tropical Polynomials.** When extending to polynomial classifiers, we maintain theoretical guarantees on the margin. Specifically, when $\rho(T) < 0$ (indicating separability in the feature space), the data are separable in the original space with a margin of at least $-\rho(T)/\|\Phi_A\|_{\text{op},\infty}$, where $\|\Phi_A\|_{\text{op},\infty}$ is the operator norm of the feature map.

For homogeneous degree-$s$ monomials, this simplifies to $-\rho(T)/s$, meaning the margin scales inversely with the polynomial degree. This gives a principled way to balance expressivity with generalization through the choice of polynomial degree.

This approach maintains the core theoretical guarantees of the hyperplane formulation while substantially increasing model flexibility, as illustrated in Figures 2 and 3 on the following pages.

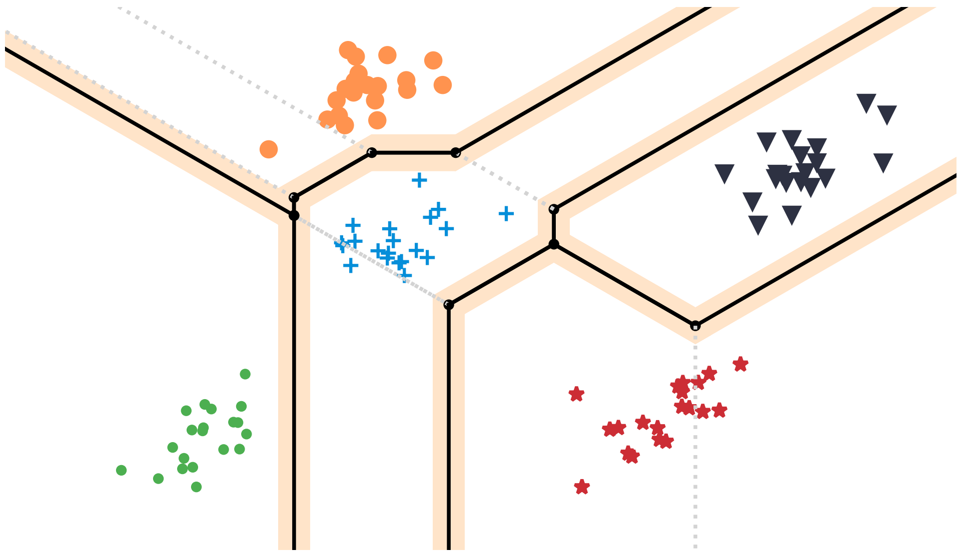


Figure 2: Multi-class classification using a cubic tropical polynomial classifier (degree-3 monomials). Each color represents a different class, and the boundaries show where the dominant monomial changes. Note how the polynomial naturally separates the five clusters with piecewise-linear boundaries. The light orange band around the decision boundaries indicates the margin, which maintains a guaranteed lower bound of $-\rho(T)/3$ due to our theoretical results.

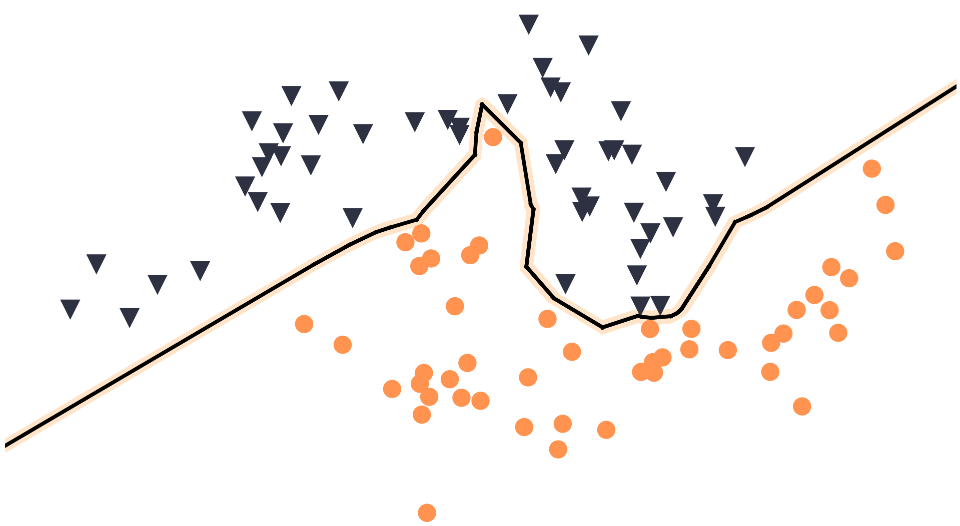


Figure 3: Visualization of a tropical polynomial classifier using the adaptive monomial selection strategy described in Section 5. Rather than using all possible monomials, this approach selects terms based on pairs of points from different classes, focusing computational resources on the most discriminative features. The resulting boundary adapts closely to the data's structure while maintaining margin guarantees.

# 6 Connection to Classical SVMs

Tropical hyperplanes can be seen as limits of classical hyperplanes under logarithmic scaling, a process known as Maslov dequantization [27]. This connection suggests a naive approach to tropical SVM: raise the data to a power $\beta > 0$, apply a classical SVM in the transformed space to maximize the margin, then map the result back via a logarithm. As $\beta$ tends to infinity, the resulting decision boundary converges to a tropical hyperplane.

However, this method suffers from severe numerical instability for large $\beta$ and fails to guarantee a good margin in the original space. The limiting hyperplane is typically suboptimal compared to the true tropical classifier obtained through our spectral approach, as illustrated in Figure 4.

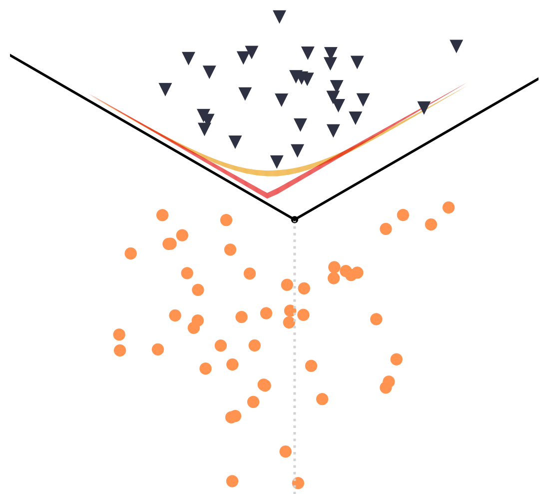


Figure 4: Comparison of tropical hyperplanes obtained through Maslov dequantization and our spectral approach. The limiting hyperplane (red) from dequantization is suboptimal compared to the spectral tropical classifier (black).

Interestingly, the connection could be exploited in the reverse direction: tropical SVMs may offer efficient, interpretable warm-starts for classical methods, especially in high-dimensional or large-scale settings.

# 7 Discussion and Future work

We addressed a key computational barrier limiting tropical SVM's practical application in machine learning. By reformulating tropical classification through spectral theory and mean-payoff games, we reduced complexity from exponential to pseudo-polynomial, making tropical SVMs feasible for real-world applications. Our framework provides natural multi-class capabilities, theoretical margin guarantees, and interpretable piecewise-linear decision boundaries.

Our work opens several promising avenues for further investigation. The relationship between tropical polynomials and ReLU networks deserves deeper exploration, potentially yielding insights into network expressivity and inspiring new tropical classifiers. Combining spectral tropical SVMs with classical kernel methods could create powerful hybrid models balancing interpretability with performance. Extending the analysis to derive proper generalization bounds would strengthen the theoretical foundation, particularly understanding how polynomial degree influences sample complexity. Developing tropical analogues to slack variables would address sensitivity to outliers and noisy data. Finally, implementing sparse representations, efficient data structures, and parallel algorithms could extend the approach to large datasets and high-dimensional problems.

**Acknowledgements**

TBD

## References


[1] Marianne Akian, Stéphane Gaubert, and Antoine Hochart. A game theory approach to the existence and uniqueness of nonlinear Perron-Frobenius eigenvectors. *Discrete and Continuous Dynamical Systems*, 40(1):207–231, 2020.

[2] Marianne Akian, Stéphane Gaubert, Yang Qi, and Omar Saadi. Tropical linear regression and mean payoff games: Or, how to measure the distance to equilibria. *SIAM J. Disc. Math.*, 37(2):632–674, May 2023.

[3] Marianne Akian, Stéphane Gaubert, Ulysse Naepels, and Basile Terver. Solving irreducible stochastic mean-payoff games and entropy games by relative krasnoselskii-mann iteration. In *48th International Symposium on Mathematical Foundations of Computer Science (MFCS 2023)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.

[4] Xavier Allamigeon, Stéphane Gaubert, Ricardo D. Katz, and Mateusz Skomra. Universal complexity bounds based on value iteration for stochastic mean payoff games and entropy games. *Information and Computation*, 302:105236, 2025.

[5] J. B. Baillon and R. E. Bruck. Optimal rates of asymptotic regularity for averaged nonexpansive mappings. In K. K. Tan, editor, *Proceedings of the Second International Conference on Fixed Point Theory and Applications*, pages 27–66. World Scientific Press, 1992.

[6] L. Breiman and C.J. Stone. Waveform Database Generator (Version 1). UCI Machine Learning Repository, 1984. DOI: https://doi.org/10.24432/C5CS3C.

[7] Guy Cohen, Stéphane Gaubert, and Jean-Pierre Quadrat. Duality and separation theorems in idempotent semimodules. *Linear Algebra and Its Applications*, 379:395–422, 2004. Special Issue on the Tenth ILAS Conference (Auburn, 2002).

[8] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[9] Mike Develin and Bernd Sturmfels. Tropical convexity. *Documenta Mathematica*, 9:1–27, 2004.

[10] Konstantinos Fotopoulos, Petros Maragos, and Panagiotis Misiakos. Tropnnc: Structured neural network compression using tropical geometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):5027–5037, 2024.

[11] Bernd Gärtner and Martin Jaggi. Tropical support vector machines. Technical Report ACS-TR-362502-01, MPI, 2008.

[12] Stéphane Gaubert and Jeremy Gunawardena. The Perron-Frobenius theorem for homogeneous, monotone functions. *Transactions of the American Mathematical Society*, 356(12):4931–4950, 2004.

[13] Stéphane Gaubert and Sergei Sergeev. Cyclic projectors and separation theorems in idempotent convex geometry. *Journal of Mathematical Sciences*, 155(6):815–829, 2011.

[14] Vassili Kolokoltsov. On linear, additive, and homogeneous operators in idempotent analysis. In Victor Maslov and Semen Samborskiĭ, editors, *Idempotent Analysis*, volume 13, pages 87–102. American Mathematical Society, Providence, Rhode Island, 1992.

[15] Vassili N. Kolokoltsov and Victor P. Maslov. Idempotent analysis. In Vassili N. Kolokoltsov and Victor P. Maslov, editors, *Idempotent Analysis and Its Applications*, pages 1–44. Springer Netherlands, Dordrecht, 1997.

[16] Diane Maclagan and Bernd Sturmfels. *Introduction to Tropical Geometry*, volume 161 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, Rhode Island, 2015.

[17] Petros Maragos, Vasileios Charisopoulos, and Emmanouil Theodosis. Tropical geometry and machine learning. *Proceedings of the IEEE*, 109(5):728–755, 2021.

[18] Petros Maragos and Emmanouil Theodosis. Multivariate tropical regression and piecewise-linear surface fitting. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3822–3826, 2020.

[19] Anthea Monod, Bo Lin, Ruriko Yoshida, and Qiwen Kang. Tropical geometry of phylogenetic tree space: A statistical perspective. *Bulletin of Mathematical Biology*, 84(2):1–26, 2022.

[20] Guido Montúfar, Yue Ren, and Leon Zhang. Sharp bounds for the number of regions of maxout networks and vertices of minkowski sums. *SIAM Journal on Applied Algebra and Geometry*, 6(4):618–649, 2022.

[21] Roger D. Nussbaum. Convexity and log convexity for the spectral radius. *Linear Algebra and Its Applications*, 73:59–122, 1986.

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

[23] Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. The MIT Press, 2002.

[24] Lloyd S. Shapley. Stochastic games. *Proceedings of the National Academy of Sciences*, 39(10):1095–1100, 1953.

[25] Xiaoxian Tang, Houjie Wang, and Ruriko Yoshida. Tropical support vector machine and its applications to phylogenomics. *SIAM Journal on Applied Algebra and Geometry*, 4(4):620–640, 2020.

[26] Vladimir N. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999.

[27] Oleg Viro. Dequantization of real algebraic geometry on logarithmic paper. In Carles Casacuberta, Rosa Maria Miró-Roig, Joan Verdera, and Sebastià Xambó-Descamps, editors, *European Congress of Mathematics*, pages 135–146. Birkhäuser Basel, Basel, 2001.

[28] Ruriko Yoshida, Leon Zhang, and Xu Zhang. Tropical principal component analysis and its application to phylogenetics. *Bulletin of Mathematical Biology*, 81(2):568–597, 2019.

[29] Liwen Zhang, Gregory Naitzat, and Lek-Heng Lim. Tropical geometry of deep neural networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5824–5832. PMLR, 2018.

[30] Uri Zwick and Mike Paterson. The complexity of mean payoff games on graphs. *Theoretical Computer Science*, 158(1):343–359, 1996.

## A Proof of Theorem 1 (Spectral Separability)

We provide a complete proof of our main result on spectral separability.

**Lemma 2.** *If the hyperplane $\mathcal{H}_a$ separates point clouds $X^1, \ldots, X^K$ with a margin of at least $\gamma > 0$, then $T(a) \leq -\gamma + a$.*

*Proof.* Consider two different classes $k$ and $\ell$. Let $I^k$ denote the set of coordinates assigned to class $k$. For any point $x \in X^k$, the margin condition implies

$$d_H(x, S^\ell) = \max_i (x_i - a_i) - \max_{j \in I^\ell} (x_j - a_j) \geq \gamma. \tag{12}$$

This can be rewritten as: for all $i \in I^\ell$,

$$x_i - a_i \leq \max_j (x_j - a_j) - \gamma. \tag{13}$$

Taking the maximum over all $x \in X^k$, we get

$$\max_{x \in X^k} (x_i - a_i) \leq \max_{x \in X^k} (\max_j (x_j - a_j)) - \gamma. \tag{14}$$

By the definition of $T^k$, this implies

$$T^k(a)_i \leq -\gamma + a_i. \tag{15}$$

Since this holds for all $i \in I^\ell$ and all classes $k \neq \ell$, we have $T(a)_i \leq -\gamma + a_i$ for all $i$, which gives us $T(a) \leq -\gamma + a$. □

**Lemma 3.** *If $\rho(T) < 0$, then there exists a hyperplane $\mathcal{H}_a$ that separates the data with a margin of at least $-\rho(T)$.*

*Proof.* Since $\rho(T) < 0$, there exists $a \in \mathbb{R}^d$ such that $T(a) = \rho(T) + a$. We define the sectors as

$$I^k = \{i : T^k(a)_i > \rho(T) + a_i\}. \tag{16}$$

First, we show these sectors are disjoint. If $i \in I^k \cap I^\ell$ for $k \neq \ell$, then $T^k(a)_i > \rho(T) + a_i$ and $T^\ell(a)_i > \rho(T) + a_i$. But then the second-largest value among $\{T^1(a)_i, \ldots, T^K(a)_i\}$ would exceed $\rho(T) + a_i$, contradicting $T(a)_i = \rho(T) + a_i$.

Next, we show that each point belongs to its assigned sector. For $x \in X^k$ and $i \notin I^k$, we have

$$x_i \leq T^k(x)_i = (T^k(x) - T^k(a))_i + T^k(a)_i. \tag{17}$$

Since $T^k$ is non-expansive, $(T^k(x) - T^k(a))_i \leq \max_j (x_j - a_j)$. And since $i \notin I^k$, we have $T^k(a)_i \leq \rho(T) + a_i$. Combining these,

$$x_i - a_i \leq \max_j (x_j - a_j) + \rho(T). \tag{18}$$

With $\rho(T) < 0$, this implies $x_i - a_i < \max_j (x_j - a_j)$, meaning $i$ cannot be the $\arg\max$ of $x - a$. Therefore, the $\arg\max$ must lie in $I^k$, placing $x$ in its correct sector.

Finally, for the margin, consider $x \in X^k$ and sector $S^\ell$ with $\ell \neq k$. The distance is.

$$d_H(x, S^\ell) = \max_j (x_j - a_j) - \max_{i \in I^\ell} (x_i - a_i). \tag{19}$$

Using the inequalities derived above, we get $d_H(x, S^\ell) \geq -\rho(T)$. □

For any Shapley operator $T$, we define its fixed-point set as $S(T) = \{x \in \mathbb{T}^d : x \leq T(x)\}$. An important property is that any tropically convex set $V$ can be written as $V = S(P_V)$.

In the binary case where data overlaps, we have $T = \min(T^1, T^2)$, hence $S(T)$ characterizes the intersection of convex hulls of both point clouds. As shown in [1], this implies that $\rho(T)$ equals the inner radius of this intersection, measuring the extent of overlap. In the non-separable case, the apex $a$ given by our Algorithm lies exactly at the center of the inner ball of $V1 + \cap V^2$.

In the multi-class case, the corresponding Shapley operator $T$ can be equivalently expressed as

$$T = \max_{1 \leq k < l \leq n} \min(T^k, T^l). \tag{20}$$

Intuitively, this can be thought of as representing the union of pairwise intersections between data classes, although the $\max$ operator does not strictly correspond to a union of fixed-point sets. This formulation provides an intuition on why our operator effectively characterizes separability across multiple classes.

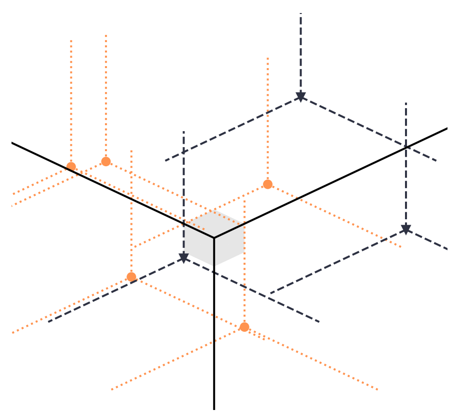


Figure 5: Non-separable case: $\rho(T) \geq 0$, indicating class overlap. The spectral radius quantifies the minimum perturbation required to achieve separability, and the inner radius of the convex hulls' intersection.

Before moving to the next proof, we define diagonal-free "projections" [13]:

$$P_X^{\mathrm{DF}}(y)_i = \max_{1 \leq j \leq p} \left\{ X_{ij} + \min_{k \neq i}(-X_{kj} + y_k) \right\}. \tag{21}$$

Both $P$ and $P^{\mathrm{DF}}$ have identical fixed-point sets and can serve as the class operators $T^k$ in our framework.

**Lemma 4.** *For binary classification with $\rho(T) \geq 0$, there exists a perturbation of the point sets $X^1$ and $X^2$, with each point moved by at most $\rho(T)$ in the tropical metric, such that the tropical convex hulls of the perturbed sets have empty intersection.*

*Proof.* Let $a$ be the eigenvector corresponding to $\rho(T)$, i.e., $T(a) = \rho(T) + a$. Let $\mathcal{H}_a$ be the tropical hyperplane with apex $a$. We construct a perturbation by projecting onto $\mathcal{H}_a$ any point whose distance to $\mathcal{H}_a$ is less than $\rho(T)$.

For a point $x \in \mathbb{T}^d$, its distance to $\mathcal{H}_a$ is

$$d_H(x, \mathcal{H}_a) = \max_i(x_i - a_i) - \max_2(x_i - a_i), \tag{22}$$

where $\max_2$ denotes the second-largest value.

For each point $x_j \in X^1 \cup X^2$, we define a perturbed point $\tilde{x}_j$ as follows:

- If $d_H(x_j, \mathcal{H}_a) \geq \rho(T)$, then $\tilde{x}_j = x_j$ (no perturbation)
- If $d_H(x_j, \mathcal{H}_a) < \rho(T)$, let $s = \arg\max_i(x_{ji} - a_i)$ be the sector of $x_j$. We set:

$$\tilde{x}_{ji} = \begin{cases} x_{ji} - d_H(x_j, \mathcal{H}_a) & \text{if } i = s \\ x_{ji} & \text{otherwise} \end{cases} \tag{23}$$

This projection ensures that $\tilde{x}_j$ lies exactly on $\mathcal{H}_a$.

Therefore, for any $x^c$ in the tropical convex hull of class $c \in \{1, 2\}$,

$$[x^c]_i \leq T^c(x^c)_i = \underbrace{(T^c(x^c) - T^c(a))_i}_{\leq \max_{k \neq i}(x^c_k - a_k)} + T^c(a)_i. \tag{24}$$

Fix a coordinate $i$. If $x \in X^1 \cup X^2$ is in sector $s \neq i$, and reaches it second argmax at coordinate $t$, then for $k \neq s$, by definition, $(\tilde{x}_j - a)_k \leq (x_j - a)_t \leq (\tilde{x}_j - a)_s$, hence $\tilde{x}_{jk} - \max_{k \neq i}(\tilde{x}_{jk} - a_k) \leq a_i$. Otherwise, $x_j$ is in the $i$-th sector and $\max_{k \neq i}(\tilde{x}_{jk} - a_k) = x_{jt} - a_t$, thus

$$\tilde{x}_{ji} - \max_{k \neq i}(\tilde{x}_j - a)_k = (\tilde{x}_j - a)_i - (\tilde{x}_j - a)_t + a_i \geq a_i,$$

with equality if $d_H(x_j, \mathcal{H}_a) \leq \rho(T)$.

Suppose by symmetry that $T(a)_i = T^1(a)_i = \rho(T) + a_i$. We also have $T^2(a) \geq \rho(T) + a_i$. Then, using the proof of Theorem 22 in [1], there exists witness points $x_{j_1} \in X^1$ and $x_{j_2} \in X^2$ in sector $i$ , with $x_{j_1}$ being at distance $\rho(T)$ from $\mathcal{H}_a$ and $x_{j_2}$ at distance greater than $\rho(T)$. Therefore, $\tilde{x}_{j_1 i} - \max_{k \neq i}(\tilde{x}_{j_1 k} - a_k) = a_i$ and $\tilde{x}_{j_2 i} - \max_{k \neq i}(\tilde{x}_{j_2 k} - a_k) \geq a_i$. Moreover, for any $j$ such that $x_j \in X^1$ is in sector $i$, Equation 24 gives $d_H(x_j, \mathcal{H}_a) \leq \rho(T)$.

Let $\tilde{T}^1$ and $\tilde{T}^2$ be the diagonal-free projections over transformed projections, and $\tilde{T} = \min(\tilde{T}^1, \tilde{T}^2)$. We have just shown that $\tilde{T}(a)_i = \tilde{T}^1(a)_i = a_i$, and finally $\tilde{T}(a) = a$. $\square$

We summarize the proof of the Main Theorem:

*Proof of Theorem 1.* For part 1 (Separability Criterion): If the data are separable with margin $\gamma > 0$, then by Lemma 2, $\rho(T) \leq -\gamma < 0$. Conversely, if $\rho(T) < 0$, then by Lemma 3, the data are separable.

For part 2 (Margin Optimality): Lemma 2 shows that no hyperplane can achieve a margin larger than $-\rho(T)$, while Lemma 3 provides a construction achieving exactly this margin.

For part 3 (Soft-Margin Interpretation): Lemma 4 shows that in the binary case, $\rho(T)$ is positive and characterizes the overlap between the tropical convex hulls. $\square$

# B Empirical Evaluation

To validate our theoretical complexity analysis, we evaluated both standard tropical SVM and an enhanced tropical polynomial implementation (with $k = 4$ sampling points per class) against scikit-learn's LinearSVC on benchmark datasets [22, 6]. All experiments used 5-fold cross-validation with standardized features. We compare training times for a fixed convergence threshold. All experiments were conducted on a MacBook Air M2.

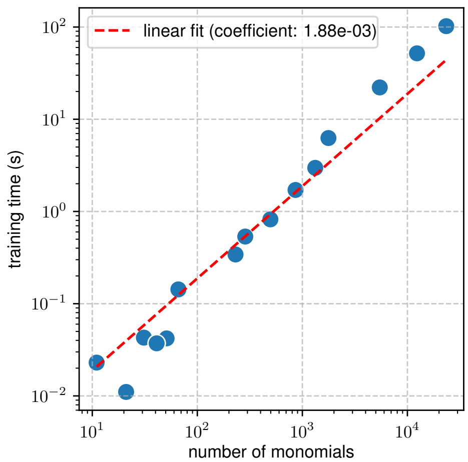
| **Dataset** | **#C** | **#S** | **Accuracy (%) / Training Time (s)** | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | **Tropical SVM** | | **Tropical Poly** | | **Linear SVC** | |
| | | | Acc | Time | Acc | Time | Acc | Time |
| Iris | 3 | 150 | 66.7 | 0.0004 | 93.3 | 0.038 | 92.7 | 0.0003 |
| Wine | 3 | 178 | 73.7 | 0.0008 | 92.7 | 0.036 | 97.8 | 0.0004 |
| Breast Cancer | 2 | 569 | 82.6 | 0.0025 | 91.7 | 0.174 | 96.7 | 0.0006 |
| Waveform | 3 | 5000 | 50.5 | 0.021 | 63.7 | 3.571 | 86.7 | 0.0147 |

Table 1: Performance comparison across benchmark datasets (5-fold cross-validation). Accuracy standard deviations range from 1.9% to 9.6%. #C: number of classes; #S: number of samples.

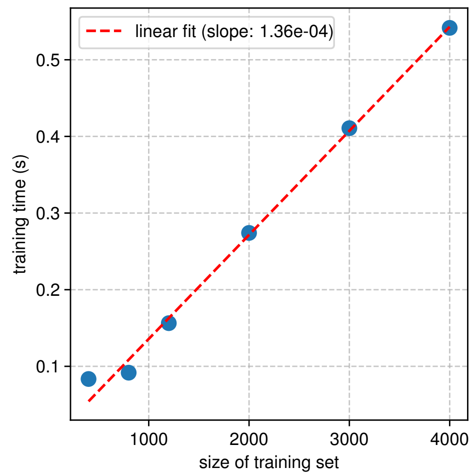
The standard tropical SVM implementation exhibits training times close to LinearSVC on smaller datasets, demonstrating practical computational feasibility. The tropical polynomial variant requires additional computation but substantially improves accuracy, approaching LinearSVC on datasets like Iris.

Notably, the spectral radius values closely align with our theoretical predictions: datasets yielding negative spectral radius values (Wine: -7.53, Breast Cancer: -1.76, Iris: -0.97) exhibited the highest tropical polynomial accuracy, while the non-separable Waveform dataset (spectral radius: 0.87) proved more challenging.

As expected, the number of monomials increases from $d + 1$ in standard tropical SVM to $O(d^k)$ in the polynomial variant, explaining the observed computation-accuracy tradeoff. While specialized classical SVM implementations remain faster, our results conclusively demonstrate that tropical SVMs can be practically computed with pseudo-polynomial guarantees.




(a) Training time vs. number of monomials




(b) Training time vs. size of training set

Figure 6: Performance scaling analysis of tropical SVM on MNIST data

Figure 6 reveals that the tropical SVM algorithm exhibits linear scaling behavior both with respect to the number of effective monomials and the training set size. We used a Python script that systematically varies PCA dimensions, polynomial degrees, and sample sizes on MNIST data [8].

## NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the contributions—including the spectral characterization, margin optimality, pseudo-polynomial algorithm, and extension to tropical polynomials—which are supported by both theoretical results and empirical evaluations.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: The paper explicitly discusses limitations such as sensitivity to outliers and the challenge of incorporating soft margins (see the "Limitations" paragraph in Section 3)

   Guidelines:

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

3. **Theory Assumptions and Proofs**

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

Answer: [Yes]

Justification: All major theoretical results—including the spectral separability theorem—are supported by clearly stated assumptions and complete proofs provided in the Appendix A.

Guidelines:

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

4. **Experimental Result Reproducibility**

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

Answer: [Yes]

Justification: The algorithm is fully described in Section 4. The experimental section provides details on data splits (5-fold cross-validation), training times, accuracy metrics, and references a publicly available GitHub repository for code and reproduction (see Table 1 and Appendix B).

Guidelines:

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

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

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

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

   Answer: [Yes]

   Justification: The paper includes a link to a GitHub repository (`https://github.com/samuelbx/tropical-svm`) with code and instructions for reproducing the experiments.

   Guidelines:

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

6. **Experimental Setting/Details**

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

   Answer: [Yes]

   Justification: The paper details the experimental setup, including the use of 5-fold cross-validation, data standardization, and provides training times and accuracy metrics in Table 1 and related figures in the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

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

   Answer: [Yes]

   Justification: The results are reported with standard deviations.

   Guidelines:

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

8. **Experiments Compute Resources**

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

   Answer: [Yes]

   Justification: All experiments were conducted on a MacBook Air M2. These specifications are typical for small-scale benchmark experiments, and the reported training times reflect the performance on this hardware.

   Guidelines:

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

9. **Code Of Ethics**

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

   Answer: [Yes]

   Justification: The work is theoretical and algorithmic with supporting experiments on standard benchmark datasets; there are no ethical issues or concerns related to the research.

   Guidelines:

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

10. **Broader Impacts**

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

Answer: [NA]

Justification: The research is primarily theoretical and methodological with no direct societal impact, and no discussion of broader impacts is necessary.

Guidelines:

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

11. **Safeguards**

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

Answer: [NA]

Justification: The work does not involve the release of data or models that present a high risk of misuse.

Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes] .

Justification: All referenced works and datasets are properly cited, and the GitHub repository is expected to include license information consistent with proper academic practice.

Guidelines:

- The answer NA means that the paper does not use existing assets.

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

13. **New Assets**

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

    Answer: [Yes]

    Justification: The paper introduces a novel algorithm and provides experimental code, with documentation available in the supplementary GitHub repository.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

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

    Answer: [NA]

    Justification: The work does not involve crowdsourcing or human subjects.

    Guidelines:

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

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

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

    Answer: [NA]

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

Guidelines:

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