An Alternating Minimization Algorithm for Regression with Tropical Rational Functions

Alex Dunbar and Lars Ruthotto

Department of Mathematics, Emory University

Introduction

Tropical algebra has recently been connected extensively with neural networks. In particular, [1, 3] show that popular neural network architectures have tropical algebraic descriptions. We are working to answer the following questions:

1. Is it feasible to solve regression problems over tropical rational functions?
2. What are the implications for neural network training and initialization?

We present an algorithm for regression with tropical rational functions and examine its use as a heuristic for neural network initialization.

Tropical Rational Regression

Our method for tropical rational regression is based on the following known result about solutions to max-plus linear systems of equations (e.g. [2]):

Let $A \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$ and denote by $\boxplus$ and $\boxplus'$ max-plus and min-plus matrix-vector multiplication, respectively. Then,

$$\hat{\mathbf{x}} := (-A^\top) \boxplus' \mathbf{b} = \arg\min_{\mathbf{x}} \|A \boxplus \mathbf{x} - \mathbf{b}\|_p \quad \text{s.t.} \quad A \boxplus \mathbf{x} \leq \mathbf{b}$$

and

$$\hat{\mathbf{x}} + \frac{1}{2}\|A \boxplus \hat{\mathbf{x}}\|_\infty = \arg\min_{\mathbf{x}} \|A \boxplus \mathbf{x} - \mathbf{b}\|_\infty$$

Let $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N \subseteq \mathbb{R}^n \times \mathbb{R}$ be a dataset. A **tropical rational function** r is the difference of two tropical polynomials and has the form

$$r(\mathbf{x}) = p(\mathbf{x}) - q(\mathbf{x}) = \max_{\mathbf{w} \in W}(\mathbf{w}^\top \mathbf{x} + p_{\mathbf{w}}) - \max_{\mathbf{w} \in W}(\mathbf{w}^\top \mathbf{x} + q_{\mathbf{w}})$$

for some finite subset $W \subseteq \mathbb{Z}^n_{\geq 0}$. Set $X \in \mathbb{R}^{N \times |W|}$ to be the matrix with entries $X_{i,\mathbf{w}} = (\mathbf{w}^\top \mathbf{x}_i)$. The minimization problem for tropical regression is

$$\min_{r \in \mathbb{T}(\mathbf{x})} \left\| \begin{bmatrix} r(\mathbf{x_1}) & r(\mathbf{x}_2) & \cdots & r(\mathbf{x}_N) \end{bmatrix}^\top - \mathbf{y} \right\|_\infty = \min_{\mathbf{p},\mathbf{q}} \|X \boxplus \mathbf{p} - X \boxplus \mathbf{q} - \mathbf{y}\|_\infty.$$

For fixed $\mathbf{q}$, $\min_{\mathbf{p}} \|X \boxplus \mathbf{p} - (X \boxplus \mathbf{q} + \mathbf{y})\|_\infty$ is a tropical polynomial regression problem. So, we can compute

$$\mathbf{p}_*(\mathbf{q}) = -X^T \boxplus' (X \boxplus \mathbf{q} + \mathbf{y}) + \frac{1}{2}\left\| X \boxplus \left(-X^T \boxplus' (X \boxplus \mathbf{q} + \mathbf{y})\right) - (X \boxplus \mathbf{q} + \mathbf{y}) \right\|_\infty,$$

and similarly for $\mathbf{q}_*(\mathbf{p})$. Alternating between polynomial regression for the numerator provides a heuristic for ∞-norm regression over tropical rational functions using only max-plus and min-plus matrix-vector products.

Algorithm 1 Alternating fit for tropical rational functions

Input: Dataset $\mathcal{D} = (\mathbf{x}_i, y_i)_{i=1}^N \subseteq \mathbb{R}^n \times \mathbb{R}$,
Set of permissible exponents $W \subseteq \mathbb{Z}^n_{\geq 0}$,
Maximum number of iterations $k_{\max}$
Output: Vectors $\mathbf{p}$ and $\mathbf{q}$ of coefficients of tropical polynomials such that $p(\mathbf{x}_i) - q(\mathbf{x}_i) \approx y_i$

$X \in \mathbb{R}^{N \times |W|} \leftarrow X_{i,\mathbf{w}} = (\mathbf{w}^\top \mathbf{x}_i)$
$\mathbf{p}^0, \mathbf{q}^0 \leftarrow -\infty, \mathbf{q}^0_{\mathbf{0}} \leftarrow \text{mean}(\mathbf{y})$
for $k \leq k_{max}$ **do**
  $\mathbf{p}^k \leftarrow \arg\min_{\mathbf{p}} \|X \boxplus \mathbf{p} - X \boxplus \mathbf{q}^{k-1} - \mathbf{y}\|_\infty$
  $\mathbf{q}^k \leftarrow \arg\min_{\mathbf{q}} \|X \boxplus \mathbf{p}^k - X \boxplus \mathbf{q} - \mathbf{y}\|_\infty$
end for
$\mathbf{p} \leftarrow \mathbf{p}^{k_{\max}}; \mathbf{q} \leftarrow \mathbf{q}^{k_{\max}}$

Results

We apply Algorithm 1 to two datasets. The **noisy sin** dataset consists of $N = 200$ pairs (x_i, y_i) consisting of uniformly spaced points $x_i \in [-1, 12]$ and $y_i = \sin(x_i) + \epsilon_i$, where $\epsilon_i \overset{iid}{\sim} \mathcal{N}(0, 0.05^2)$. The `peaks` dataset is generated in MATLAB and consists of 49^2 datapoints (x_i, y_i) equally spaced in the grid $[-3, 3]^2$ and values z_i corresponding to a nonlinear combination of Gaussians. For the noisy sin dataset, we fit a tropical rational function which is the difference of two degree 15 tropical polynomials. For the `peaks` dataset, we fit a tropical rational function which is the difference of two tropical polynomials with exponents $W = \{(w_1, w_2) | 0 \leq w_1, w_2 \leq 31\} \subseteq \mathbb{Z}^2$.

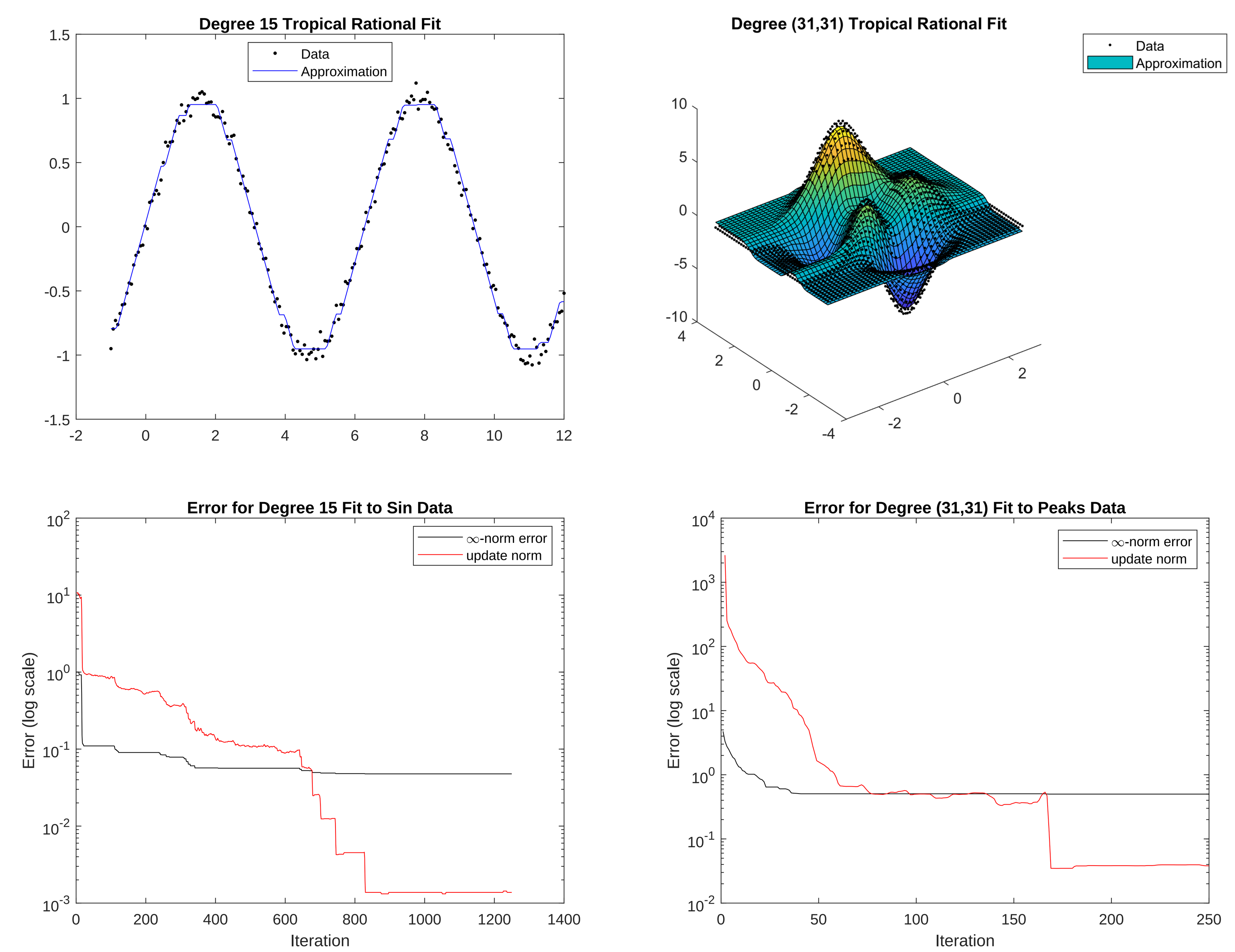

Fig. 1: Tropical rational approximations to noisy sin data and `peaks` data and the convergence behavior of Algorithm 1 during training.

ReLU Neural Networks

A **fully connected ReLU neural network** is a function $\nu : \mathbb{R}^n \to \mathbb{R}$ such that

$$\nu = \rho^{(L)} \circ \sigma \circ \rho^{(L-1)} \circ \sigma \circ \cdots \circ \sigma \circ \rho^{(1)}$$

where $\sigma(\mathbf{x}) = \max(\mathbf{x}, 0)$ and $\rho^{(j)}$ is affine linear. By [3], ReLU neural networks with integer weights are tropical rational functions. Because

$$\max(\mathbf{a}_1^\top \mathbf{x} + b_1, \mathbf{a}_2^\top \mathbf{x} + b_2) = \begin{bmatrix} 1 & 1 & -1 \end{bmatrix} \sigma \left(\begin{bmatrix} \mathbf{a}_1 - \mathbf{a}_2^\top \\ \mathbf{a}_2 \\ -\mathbf{a}_2 \end{bmatrix} \mathbf{x} + \begin{bmatrix} b_1 - b_2 \\ b_2 \\ -b_2 \end{bmatrix} \right)$$

is a fully connected network and $\max(\nu_1, \nu_2) = \sigma(\nu_1 - \nu_2) + \sigma(\nu_2) - \sigma(-\nu_2)$ for any networks ν_1, ν_2, we can express a tropical rational function as a ReLU network whose depth grows with $\log_2(\#\text{monomials})$ and whose hidden layers have a natural 3×6 block sparsity structure.

Neural Network Initialization

We use Algorithm 1 to provide a heuristic for weight initialization in ReLU networks, which we fit to the noisy sin and `peaks` datasets. For each dataset, we compare fully connected networks with weight initialization determined by tropical rational regression (Full Trop), fully connected networks with the same architecture and randomly intialized weights (Full Rand), networks with the block 3×6 sparsity pattern enforced during training and tropical initialization (Sparse Trop), and networks with the block 3×6 sparsity pattern and randomly initialized weights (Sparse Rand). The random initialization is the PyTorch default. Networks are trained in PyTorch with a mean square error loss function.

The tropical rational regression initialization heuristic **outperforms** the PyTorch default initialization for the noisy sin dataset:

	Full Trop	Sparse Trop	Full Rand	Sparse Rand
Loss at Initialization	0.003859	0.003837	0.596953	0.776548
Loss at 1000 Epochs	0.002364	0.002611	0.006050	0.002907

Tab. 1: Training loss across initialization strategies for neural networks fit to noisy sin data

However, the tropical initialization **does not outperform** the default initialization for the `peaks` dataset:

	Full Trop	Sparse Trop	Full Rand	Sparse Rand
Loss at Initialization	0.155525	0.154782	3.882785	3.482387
Loss at 100 Epochs	0.011248	0.011371	0.005648	0.006236

Tab. 2: Training loss across initialization strategies for neural networks fit to `peaks` data

Conclusions and Future Directions

We have proposed an algorithm for regression with tropical rational functions. This method experimentally appears to be effective and quick to compute. The use of this method as a heuristic for the initialization of neural networks remains open, with successful and unsuccessful examples. Future goals are to

- Develop a better theoretical understanding of the convergence behavior of Algorithm 1. In particular, develop an informed criterion to stop iterations.
- Modify Algorithm 1 to enforce sparsity during tropical rational regression. This could allow for a data-informed choice of monomial basis for the numerator and denominator polynomials.
- Explore different neural network architectures for initialization (e.g. skip connections, architectures informed by activation regions). In general, there are many neural networks which can represent a given function. Different architectures may benefit more from tropical initialization.

References

[1] Vasileios Charisopoulos and Petros Maragos. *A Tropical Approach to Neural Networks with Piecewise Linear Activations.* 2018. arXiv: `1805.08749 [stat.ML]`.

[2] Petros Maragos and Emmanouil Theodosis. *Tropical Geometry and Piecewise-Linear Approximation of Curves and Surfaces on Weighted Lattices.* 2019. arXiv: `1912.03891 [cs.LG]`.

[3] Liwen Zhang, Gregory Naitzat, and Lek-Heng Lim. "Tropical geometry of deep neural networks". In: *International Conference on Machine Learning.* PMLR. 2018, pp. 5824–5832.