Improved Convergence for ℓ_∞ and ℓ_1 Regression via Iteratively Reweighted Least Squares

Alina Ene, Adrian Vladu [a]

Boston University

[a] alphabetic order

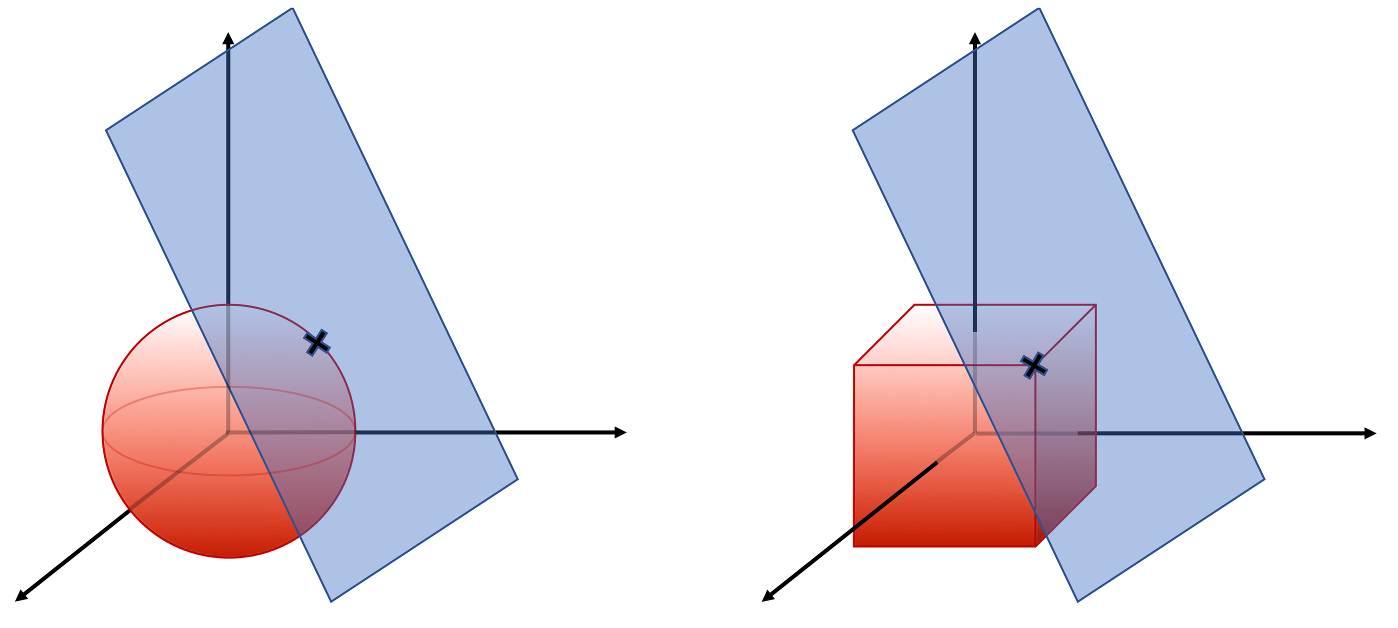
Regression Problems

Regression problems are standard in machine learning. In general, they can be phrased as minimizing a vector norm, subject to linear constraints:

$$\min_{\mathbf{Ax}=\mathbf{b}} \|\mathbf{x}\|_p$$

Depending on the choice of norm, obtaining fast algorithms can be

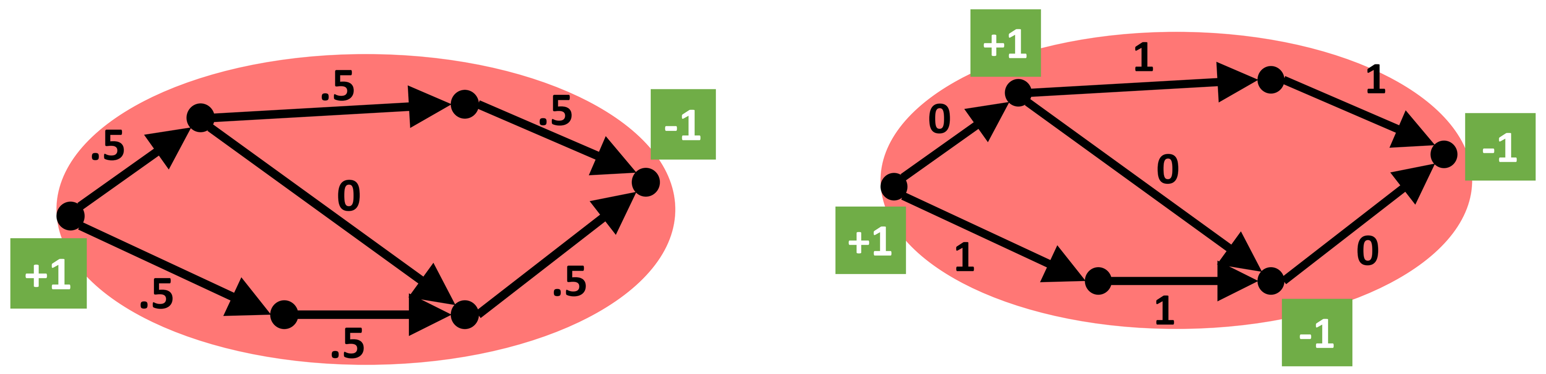
- very easy: for $p=2$, the solution is given by a single linear system solve $\mathbf{x} = \mathbf{A}^\top(\mathbf{AA}^\top)^+\mathbf{Ab}$.
- very hard: when $p \in \{1,\infty\}$, the problem is equivalent to linear programming.



$p=2$ $\qquad$ $p=\infty$

Natural benchmark: ℓ_∞/ℓ_1 regression on graphs

- $x \in \mathbb{R}^m$: flow on the graph's edges
- $\mathbf{b} \in \mathbb{R}^n$: demand that $\mathbf{x}$ is supposed to route
- $\mathbf{A} \in \mathbb{R}^{m\times n}$: matrix such that $\mathbf{Ax}$ outputs the demand routed by $\mathbf{x}$ (think of it as boundary operator)



ℓ_∞ regression $\iff$ maximum flow $\qquad$ ℓ_1 regression $\iff$ minimum cost flow

Currents approaches:

- first-order methods (gradient descent): in general, need at least $\Omega(m^{3/2}/\mathrm{poly}(\epsilon))$, running time strongly depends on matrix structure.
- second-order methods: interior point methods require $\widetilde{O}(m^{1/2}\log(1/\epsilon))$ linear system solves, $\widetilde{O}(\mathrm{rank}(\mathbf{A})^{1/2}\log(1/\epsilon))$ with a lot of work.
- hybrid method (first order iteration + linear system solve): $\widetilde{O}(m^{1/3}/\epsilon^{11/3})$ linear system solves [1], improved to $\widetilde{O}(m^{1/3}/\epsilon^{8/3})$ [2].

Practicality vs. theoretical guarantees?

Iteratively Reweighted Least Squares (IRLS) Methods

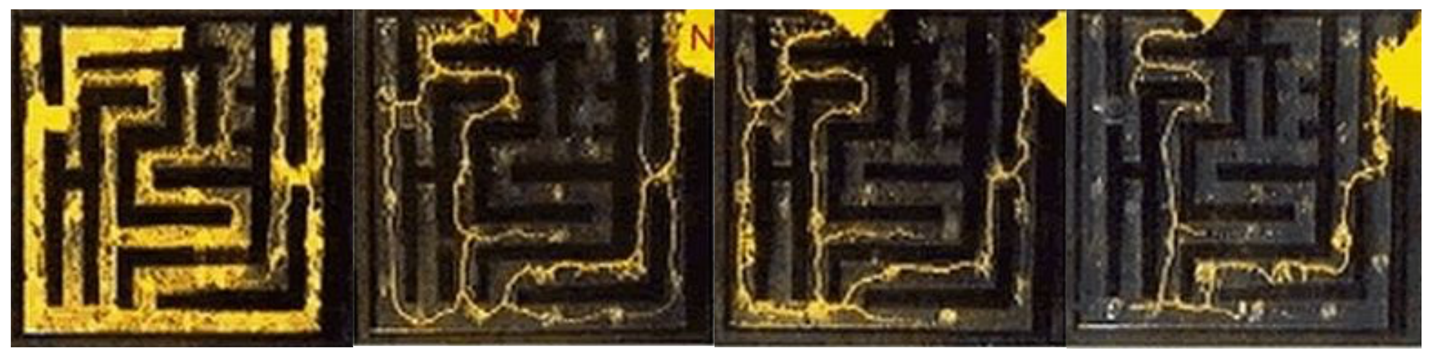
IRLS is a popular method used in practice, solves a sequence of weighted ℓ_2 minimization problems:

$$\left.\begin{array}{l}\mathbf{x} = \arg\min_{\mathbf{Ax}=\mathbf{b}} \sum r_i x_i^2 \\ \text{Update } \mathbf{r}\end{array}\right\} \text{repeat}$$

- Convergence to optimum is usually heuristic, can get stuck locally for certain starting points.
- No asymptotic convergence proofs, except for restricted classes of matrices.

Notable exception: "slime mold dynamic", inspired from the evolution of *Physarum polycephalum* in a maze.

- Can be thought of as ℓ_1 regression on graph.
- Dynamic given by $\mathbf{r} = 1/\mathbf{x}$ (or damped variations of it).
- [3] gives a version of the dynamic which converges in $\widetilde{O}\left(n^2\alpha^2/\epsilon^3\right)$ iterations, where $\alpha = \max_{\mathbf{A}' \text{ square submatrix of } \mathbf{A}} \det(\mathbf{A}')$.



Evolution of *Physarum polycephalum* in a maze.

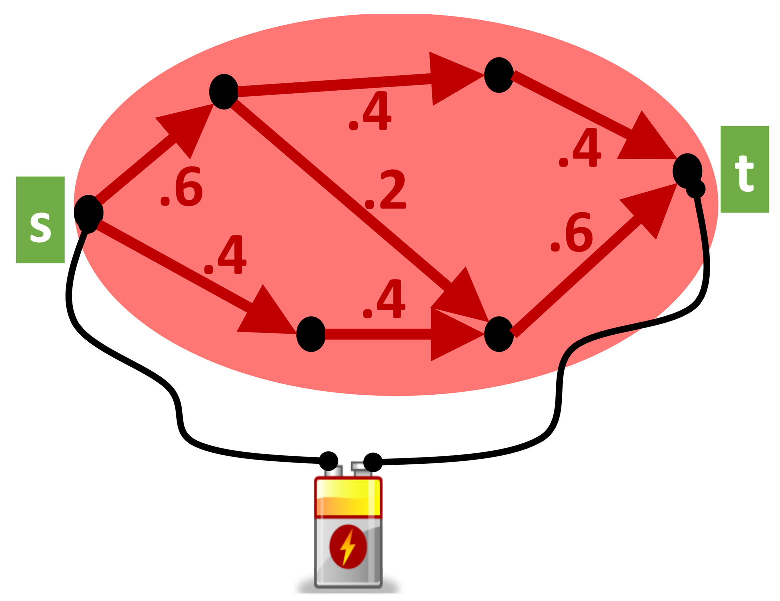
Our Method

For intuition, we consider regression on graphs. Everything works identically in the general case.

1. Weighted least squares step $\iff$ compute electrical flow.
2. Updating weights $\iff$ changing electrical resistances.

ℓ_∞ minimization.
Key idea: increase resistances for edges that are too congested.

- Start with a guess M on $\|\mathbf{x}^*\|_\infty$.
- Initialize with $\mathbf{r} = \mathbf{1}$.
- Compute electrical flow $\mathbf{x}$ for resistances $\mathbf{r}$.
- Update $r_i \leftarrow r_i \cdot \max\{1, (x_i/M)^2\}$.
- Repeat.



ℓ_1 minimization. Dual of ℓ_∞ minimization, can be recovered by applying the same method on the dual problem.

- M is a guess on $\|\mathbf{x}^*\|_1$.
- Update rule: $r_i \leftarrow r_i \cdot \min\left\{1, \frac{1}{r_i x_i} \cdot \frac{\sum r_i x_i^2}{M}\right\}^2$.

How to find M? Binary search, and check feasibility. (But can avoid this step with some care.)

Theorem

*For $p \in \{1,\infty\}$, there exists an algorithm ℓ_p-*Minimization*, which on input $(\mathbf{A},\mathbf{b},\epsilon,M)$,*

1. *returns a solution $\mathbf{x}$ such that $\mathbf{Ax}=\mathbf{b}$ and $\|\mathbf{x}\|_p \le (1+\epsilon)M$,*
2. *or certifies that $\min_{\mathbf{Ax}=\mathbf{b}} \|\mathbf{x}\|_p \ge (1-\epsilon)M$.*

Furthermore both algorithms finish in $O\left(\frac{m^{1/3}\log(1/\epsilon)}{\epsilon^{2/3}} + \frac{\log m}{\epsilon^2}\right)$ iterations, each of which can be implemented by solving a weighted least squares problem of the form $\min_{\mathbf{Ax}=\mathbf{b}} \sum_i r_i x_i^2$, where $\mathbf{r}$ is an arbitrary nonnegative vector.

Proof Technique (ℓ_∞-Minimization)

Square the objective, and write it as a saddle point problem:

$$\min_{\mathbf{Ax}=\mathbf{b}} \|\mathbf{x}\|_\infty^2 = \min_{\mathbf{Ax}=\mathbf{b}} \max_{\mathbf{r}\in\Delta} \sum r_i x_i^2 = \max_{\mathbf{r}\in\Delta} \min_{\mathbf{Ax}=\mathbf{b}} \sum r_i x_i^2 := \max_{\mathbf{r}\in\Delta} \underbrace{\mathcal{E}_\mathbf{r}}_{\text{electrical energy}}$$

Aim to increase $\mathcal{E}_\mathbf{r}/\|\mathbf{r}\|_1$, by increasing $\mathbf{r}$ such that

$$\frac{\mathcal{E}_{\mathbf{r}'} - \mathcal{E}_\mathbf{r}}{\|\mathbf{r}'-\mathbf{r}\|_1} \ge M^2\,. \qquad (1)$$

Key lemma: Lower bound increase in energy, after changing resistances.

$$\mathcal{E}_{\mathbf{r}'} \ge \mathcal{E}_\mathbf{r} + \sum_i r_i x_i^2 \left(1 - \frac{r_i}{r_i'}\right)\,. \qquad (2)$$

Plugging (2) into (1), we increase resistances such that each element gives the right amount of *bang for the buck*:

$$\frac{r_i x_i^2\left(1-\frac{r_i}{r_i'}\right)}{r_i' - r_i} = x_i^2 \cdot \frac{r_i}{r_i'} \ge M^2\,.$$

- If no resistance can be increased, we have a feasible solution.
- Otherwise, we prove that $\|\mathbf{r}\|_1$ increases very fast.

Takeaway notes

- Dominant term has $\epsilon^{-2/3}$ dependence, nonstandard in optimization.
- Similar to width-independent positive LP [4]. What else is there?
- **Conjecture:** can accelerate to $\widetilde{O}(m^{1/3}/\epsilon^{1/3} + 1/\epsilon)$; tight, since otherwise it yields faster max flow without looking at graph structure.

References

- P. Christiano, J. A. Kelner, A. Madry, D. A. Spielman, and S. Teng, "Electrical flows, laplacian systems, and faster approximation of maximum flow in undirected graphs", in *STOC 2011.*
- H. H. Chin, A. Madry, G. L. Miller, and R. Peng, "Runtime guarantees for regression problems", in *ITCS 2013.*
- D. Straszak and N. K. Vishnoi, "IRLS and slime mold: Equivalence and convergence", *CoRR abs/1601.02712,*
- N. E. Young, "Sequential and parallel algorithms for mixed packing and covering", in *FOCS 2001.*