Improved Convergence for $\ell_\infty$ and $\ell_1$ Regression via Iteratively Reweighted Least Squares

Alina Ene, Adrian Vladu
IRLS Method

Basic primitive:

$$\min \sum r_i x_i^2$$

$$Ax = b$$
IRLS Method

Basic primitive:

\[ \min \sum r_i x_i^2 \]

\[ Ax = b \]

solution given by one linear system solve

\[ x = R^{-1} A^T (A^T R^{-1} A)^{-1} Ab \]

* \( R = \text{diag}(r) \)
IRLS Method

Basic primitive:

\[ \begin{align*}
\min & \sum r_i x_i^2 \\
A x &= b
\end{align*} \]

solution given by one linear system solve

\[ x = R^{-1} A^T (A^T R^{-1} A)^{-1} A b \]

* \( R = \text{diag}(r) \)

“Hard” problem:

\[ \begin{align*}
\min & \ |x|_p \\
A x &= b
\end{align*} \]

** \( p = \{1, \infty\} \)
**IRLS Method**

**Basic primitive:**

\[
\min \sum r_i x_i^2 \\
Ax = b
\]

solution given by one linear system solve

\[
x = R^{-1}A^T(A^TR^{-1}A)^{-1}Ab
\]

* \( R = \text{diag}(r) \)

**“Hard” problem:**

\[
\min |x|_p \\
Ax = b
\]

equivalent to linear programming

** \( p \in [1, \infty) \)**
IRLS Method

Basic primitive:

\[
\min \sum r_i x_i^2 \\
A x = b
\]

solution given by one linear system solve

\[
x = R^{-1} A^T (A^T R^{-1} A)^{-1} A b
\]

"Hard" problem:

\[
\min |x|_p \\
A x = b
\]

equivalent to linear programming

\[
p = \{1, \infty\}
\]

* \( R = \text{diag}(r) \)

** \( p = \{1, \infty\} \)
IRLS Method

Basic primitive:

\[ \min \sum r_i x_i^2 \]
\[ Ax = b \]

solution given by one linear system solve

\[ x = R^{-1} A^T (A^T R^{-1} A)^{-1} A b \]

“Hard” problem:

\[ \min |x|_p \]
\[ Ax = b \]

equivalent to linear programming

\[ * R = \text{diag}(r) \]

** \[ p = \{1, \infty\} \]
IRLS Method

Basic primitive:
\[
\min \sum r_i x_i^2 \\
A x = b
\]

solution given by one linear system solve
\[
x = R^{-1}A^T(A^T R^{-1}A)^{-1} Ab
\]

“Hard” problem:
\[
\min |x|_p \\
A x = b
\]
equivalent to linear programming

* \( R = \text{diag}(r) \)

** \( p = \{1, \infty\} \)
IRLS Method

Basic primitive:

\[ \min \sum r_i x_i^2 \]
\[ A x = b \]

Solution given by one linear system solve

\[ x = R^{-1} A^T (A^T R^{-1} A)^{-1} A b \]

“Hard” problem:

\[ \min |x|_p \]
\[ A x = b \]

Equivalent to linear programming

\[ * \quad R = \text{diag}(r) \]

** \quad p = \{1, \infty\}
Benchmark: Optimization on Graphs

\[ \min \ |x|_\infty \]

\[ Ax = b \]
Benchmark: Optimization on Graphs

minimize congestion of flow $x$

\[ \min |x|_\infty \]

\[ Ax = b \]
**Benchmark: Optimization on Graphs**

minimize congestion of flow $x$

$$\min |x|_\infty$$

$$Ax = b$$

boundary condition: $x$ routes demand from $s$ to $t$
Benchmark: Optimization on Graphs

minimize congestion of flow $x$

\[
\min |x|_\infty \\
Ax = b
\]

boundary condition: $x$ routes demand from $s$ to $t$

Maximum flow
Benchmark: Optimization on Graphs

\[
\text{min } |x|_1
\]

\[
Ax = b
\]
Benchmark: Optimization on Graphs

\[ \text{minimize cost of flow } x \]

\[ \min |x|_1 \]

\[ Ax = b \]
Benchmark: Optimization on Graphs

minimize cost of flow $x$

$\min |x|_1$

$Ax = b$

boundary condition: $x$ routes demand from $+1$ to $-1$
Minimum cost flow

Benchmark: Optimization on Graphs

minimize cost of flow $x$

$$\min |x|_1 \quad A x = b$$

boundary condition: $x$ routes demand from +1 to -1

Minimum cost flow
Benchmark: Optimization on Graphs

\[ \min |x|_\infty \]
\[ Ax = b \]

max flow

\[ \min |x|_1 \]
\[ Ax = b \]

min cost flow
Benchmark: Optimization on Graphs

Q: Are these problems really that hard?

\[
\begin{align*}
\min \ |x|_\infty \\
Ax &= b \\
\text{max flow}
\end{align*}
\]

\[
\begin{align*}
\min \ |x|_1 \\
Ax &= b \\
\text{min cost flow}
\end{align*}
\]
Benchmark: Optimization on Graphs

\[ \min |x|_\infty \quad A x = b \]
max flow

Q: Are these problems really that hard?

\[ \min |x|_1 \quad A x = b \]
min cost flow

First order methods (gradient descent)

- running time strongly depends on matrix structure
- in general, takes time at least \( \Omega(m^{1.5}/\text{poly}(\varepsilon)) \)
Benchmark: Optimization on Graphs

\[ \min |x|_\infty \quad A x = b \]

\[ \min |x|_1 \quad A x = b \]

Q: *Are these problems really that hard?*

**First order methods (gradient descent)**

- running time strongly depends on matrix structure
- in general, takes time at least \( \Omega(m^{1.5}/\text{poly}(\varepsilon)) \)

**Second order methods (Newton method, IRLS)**

- interior point method: \( \tilde{O}(m^{1/2}) \) linear system solves
- can be made \( \tilde{O}(n^{1/2}) \) with a lot of work [Lee-Sidford ’14]

Benchmark: Optimization on Graphs

- **min** \( |x|_\infty \quad A x = b \)
- **max flow**

- **min** \( |x|_1 \quad A x = b \)
- **min cost flow**

**”Hybrid” method**

- [CKMST, STOC ’11] \( \tilde{O}(m^{1/3}/\varepsilon^{11/3}) \) linear system solves
- ~30 pages of description and proofs for complicated method
Benchmark: Optimization on Graphs

\[
\begin{align*}
\text{min } |x|_\infty \\
Ax = b
\end{align*}
\]

max flow

Q: Are these problems really that hard?

\[
\begin{align*}
\text{min } |x|_1 \\
Ax = b
\end{align*}
\]

min cost flow

First order methods (gradient descent)

→ running time strongly depends on matrix structure
→ in general, takes time at least \( \Omega(m^{1.5}/\text{poly}(\varepsilon)) \)

Second order methods (Newton method, IRLS)

→ interior point method: \( \tilde{O}(m^{1/2}) \) linear system solves
→ can be made \( \tilde{O}(n^{1/2}) \) with a lot of work [Lee-Sidford ’14]

“Hybrid” method

→ [Christiano-Kelner-Madry-Spielman-Teng ’11] \( \tilde{O}(m^{1/3}/\varepsilon^{11/3}) \) linear system solves
→ \( \sim30 \) pages of description and proofs for complicated method
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3}+1/\varepsilon^2)$ iterations
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\epsilon^{2/3}+1/\epsilon^2)$ iterations

* no matter what the structure of the underlying matrix is
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3}+1/\varepsilon^2)$ iterations

$$\min |x|_\infty \leq \text{OPT}$$

$$Ax = b$$
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3}+1/\varepsilon^2)$ iterations

\[ \text{Guess} \]
\[ \text{OPT value} \]
\[ \min |x|_\infty \]
\[ Ax = b \]
\[ \leq \text{OPT} (.5) \]
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3} \varepsilon^{-2/3} + 1/\varepsilon^2)$ iterations

\[
\min |x|_\infty \leq OPT \quad (\text{.5})
\]

\[
A x = b
\]

Guess OPT value

$|x|_\infty$
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3}+1/\varepsilon^2)$ iterations

$$\min |x|_\infty \quad Ax = b \quad \leq OPT \quad \text{value}$$

Guess

Initialize

$s \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad t$

$r = 1$
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3} + 1/\varepsilon^2)$ iterations

$s$ 1 .6 .4 .4 .4 1
1 .4 .2 1 .6 1
.4 1 1

Guess

OPT value

Initialize

Solve least squares problem

$\min |x|_\infty$

$Ax = b$

$r = 1$

$\min \sum r_i x_i^2$

$Ax = b$

$\leq \text{OPT} (0.5)$
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3} + 1/\varepsilon^2)$ iterations

Guess OPT value

Initialize

Solve least squares problem

Update $r$

\[
\min |x|_\infty \quad A x = b
\]

\[
\leq \text{OPT} (\cdot 5)
\]

\[
\min \sum r_i x_i^2 \quad A x = b
\]

\[
r_i \leftarrow r_i^* \max\{ (x_i/\text{OPT})^2, 1 \}
\]
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3} + 1/\varepsilon^2)$ iterations

- Guess OPT value
- Initialize
- Solve least squares problem
- Update $r$

$$\min |x|_\infty \quad Ax = b$$

$\leq$ OPT (.5)

$$\min \sum r_i x_i^2 \quad Ax = b$$

$$r_i \leftarrow r_i^* \max\{ (x_i/OPT)^2, 1 \}$$
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3} + 1/\varepsilon^2)$ iterations

1. Guess OPT value
2. Initialize
3. Solve least squares problem
4. Update $r$

$\min |x|_\infty$
$Ax = b$

$\leq$ OPT (.5)

$\min \sum r_i x_i^2$
$Ax = b$

$r_i \leftarrow r_i^*$
$max\{(x_i/OPT)^2, 1\}$
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3}+1/\varepsilon^{2})$ iterations

Guess OPT value

Initialize

Solve least squares problem

Update $r$

min $|x|_{\infty}$
$Ax = b$

r = 1

min $\sum r_i x_i^2$
$Ax = b$

$r_i \leftarrow r_i^*$

max$\{(x_i/OPT)^2, 1\}$

$\leq$ OPT (.5)
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3}+1/\varepsilon^2)$ iterations

\[
\begin{align*}
\text{Guess OPT value} & \quad \text{min } |x|_\infty \\
\text{Initialize} & \quad Ax = b \\
\text{Solve least squares problem} & \quad r = 1 \\
\text{Update } r & \quad \text{min } \sum r_i x_i^2 \\
& \quad Ax = b \\
& \quad r_i \leftarrow r_i^* \\
& \quad \max\{(x_i/OPT)^2, 1\} \\
\leq \text{OPT (.5)}
\end{align*}
\]
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3}+1/\varepsilon^2)$ iterations

\[
\min |x|_\infty \\
Ax = b
\]

Guess OPT value

Initialize

Solve least squares problem

Update $r$

\[
\min \sum r_i x_i^2 \\
Ax = b
\]

$r_i \leftarrow r_i^* \max\{((x_i/OPT)^2, 1}\}

\leq OPT (.5)
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3}+1/\varepsilon^2)$ iterations

\begin{align*}
\text{Guess} & \quad \text{OPT value} \\
\text{Initialize} & \quad \min |x|_\infty \quad A\mathbf{x} = \mathbf{b} \\
\text{Solve least squares problem} & \quad r = 1 \\
\text{Update } r & \quad \min \sum r_i x_i^2 \quad A\mathbf{x} = \mathbf{b} \\
\text{} & \quad r_i \leftarrow r_i^* \\
\text{} & \quad \max\{(x_i/\text{OPT})^2, 1\} \\
\leq \text{OPT} (.5)
\end{align*}
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\varepsilon^{2/3}+1/\varepsilon^2)$ iterations

\[ \begin{align*}
\text{Guess} & \quad \text{OPT value} \\
\text{Initialize} & \quad \text{min } |x|_\infty \\
\text{Solve least squares problem} & \quad \text{Ax = b} \\
\text{Update r} & \quad r = 1 \\
\end{align*} \]

\[ \begin{align*}
\text{min } \sum r_i x_i^2 & \quad \text{Ax = b} \\
\text{r_i} & \quad \leftarrow r_i^* \\
\text{max}\{(x_i/OPT)^2, 1\} & \quad \leq \text{OPT (0.5)} \\
\end{align*} \]
This work

Natural IRLS method runs in $\tilde{O}(m^{1/3}/\epsilon^{2/3} + 1/\epsilon^2)$ iterations

Guess OPT value

$\min \ |x|_\infty$
$Ax = b$

$\leq OPT (.5)$

Initialize

$r = 1$

Solve least squares problem

$\min \sum r_i x_i^2$
$Ax = b$

Update $r$

$r_i \leftarrow r_i^*$

$max\{((x_i/OPT)^2, 1\}$
Nonstandard Optimization Primitive

$\rightarrow$ Objective function is $\max_{r \geq 0} \min_{Ax = b} \frac{\sum_{i} r_i x_i^2}{\sum_{i} r_i}$
Nonstandard Optimization Primitive

→ Objective function is $\max_{r \geq 0} \min_{Ax=b} \sum r_i x_i^2 / \sum r_i$

→ Similar analysis to packing/covering LP [Young ’01]
Nonstandard Optimization Primitive

→ Objective function is \( \max_{r \geq 0} \min_{Ax = b} \sum r_i x_i^2 / \sum r_i \)

→ Similar analysis to packing/covering LP [Young ’01]

→ \( \ell_1 \) version is a type of “slime mold dynamics” [Straszak-Vishnoi ’16, ‘17]
Nonstandard Optimization Primitive

→ Objective function is $\max_{r \geq 0} \min_{Ax=b} \sum_r x_i^2 / \sum_i r_i$

→ Similar analysis to packing/covering LP  [Young ’01]

→ $\ell_1$ version is a type of “slime mold dynamics” [Straszak-Vishnoi ’16, ‘17]
Nonstandard Optimization Primitive

→ Objective function is $\max_{r \geq 0} \min_{A\mathbf{x} = \mathbf{b}} \frac{\sum r_i x_i^2}{\sum r_i}$

→ Similar analysis to packing/covering LP [Young ’01]

→ $\ell_1$ version is a type of “slime mold dynamics” [Straszak-Vishnoi ’16, ‘17]

→ Any insights for new optimization methods?
Thank You!

More details at poster

#208