Near-Optimal Algorithms for Online Matrix Prediction

Elad Hazan (Technion)

Satyen Kale (Yahoo! Labs)

Shai Shalev-Shwartz (Hebrew University)

Three Prediction Problems:
I. Online Collaborative Filtering

Users: $\{1, 2, \ldots, m\}$
Movies: $\{1, 2, \ldots, n\}$

On round t:
User i_t arrives and is interested in movie j_t
Output predicted rating p_t in $[-1, 1]$
User responds with actual rating r_t in $[-1, 1]$
Loss $= (p_t - r_t)^2$

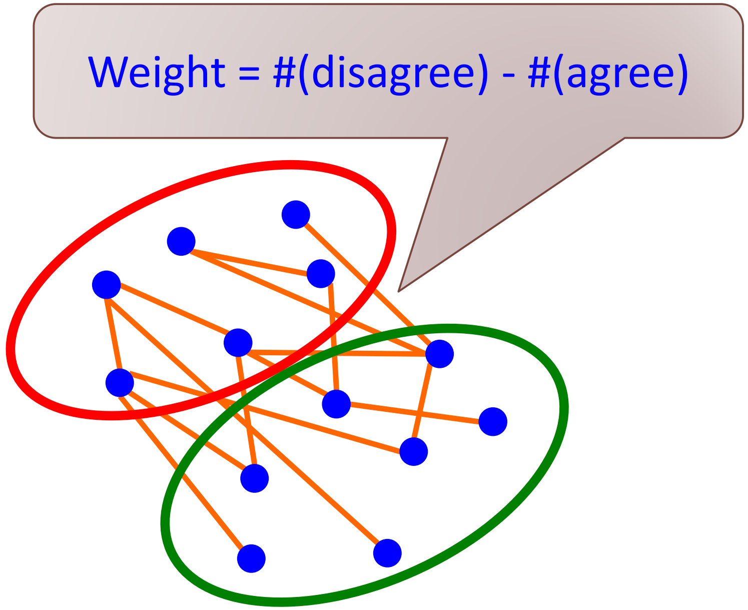
Sum of singular values

Comparison class: all $m \times n$ matrices with entries in $[-1, 1]$ of trace norm $\leq \tau$
For each such matrix W, predicted rating $= W(i_t, j_t)$

Regret = loss of alg – loss of best bounded trace-norm matrix

If no entries repeated, [Cesa-Bianchi, Shamir '11]: $O(n^{3/2})$ regret for $\tau = O(n)$

Three Prediction Problems:
II. Online Max Cut



2 political parties
Voters: {1, 2, …, n}

On round t:
Voters i_t, j_t arrive
Output prediction: votes agree or disagree
Loss = 1 if incorrect prediction, 0 o.w.

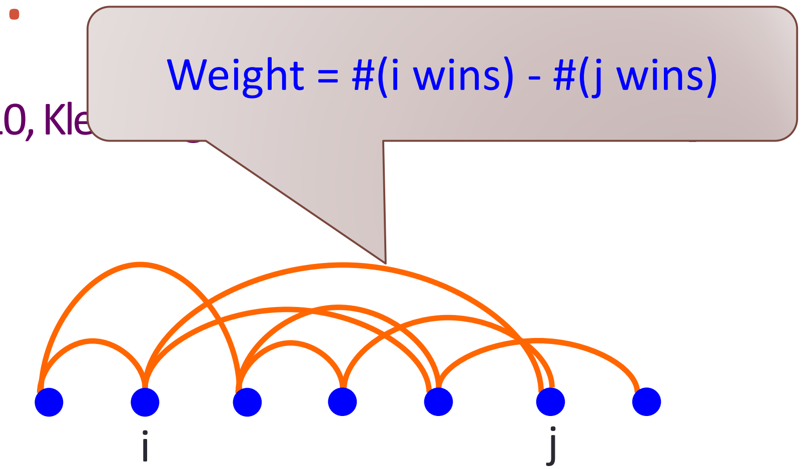
Best bipartition = Max Cut!

Comparison class: all possible bipartitions
Bipartition prediction = “agree” if i_t, j_t in same partition, “disagree” o.w.

Regret = loss of alg – loss of best bipartition

Inefficient alg using the 2^n bipartitions as experts: regret = $O\left(\sqrt{nT}\right)$

Three Prediction Problems:
III. Online Gambling [Abernethy '10, Kle



Best permutation = Min Feedback Arc Set!

Teams: {1, 2, …, n}

In round t:
Teams i_t, j_t compete
Output: prediction which team will win
Loss = 1 if incorrect prediction, 0 o.w.

Comparison class: all possible permutations π
Permutation π prediction = i_t if $\pi(i_t) \leq \pi(j_t)$; j_t o.w.

Regret = loss of alg – loss of best permutation

Inefficient alg using the n! permutations as experts: regret = $O\left(\sqrt{n\log(n)T}\right)$

Trivial bound of $O\left(n\sqrt{T}\right)$ considered hard to improve (e.g. [Kanade, Steinke '12])

Results

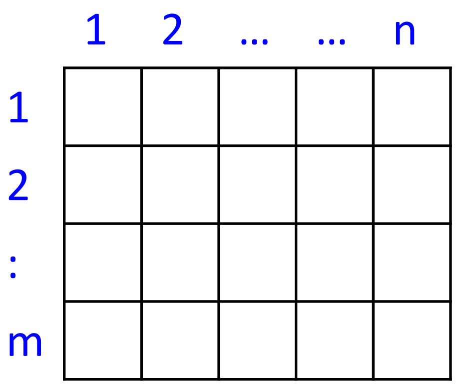
Stochastic; solves $50 open problem of [Srebro, Shamir '11]

	Upper Bound	Lower Bound
Online Collaborative Filtering	$O\left(\sqrt{\tau\sqrt{n}\log(n)T}\right)$	$\Omega\left(\sqrt{\tau\sqrt{n}T}\right)$
Online Max Cut	$O\left(\sqrt{n\log(n)T}\right)$	$\Omega\left(\sqrt{nT}\right)$
Online Gambling	$O\left(\sqrt{n\log^3(n)T}\right)$	$\Omega\left(\sqrt{n\log(n)T}\right)$

By [Kleinberg, Niculescu-Mizil, Sharma '10]

One meta-problem to rule them all: Online Matrix Prediction (OMP)

m x n matrices



In round t:

Receive pair i_t, j_t in $[m] \times [n]$

Output prediction p_t in $[-1, 1]$

Receive true value y_t in $[-1, 1]$

Suffer loss $L(p_t, y_t)$

Comparison class: set ***W*** of m x n matrices with entries in $[-1, 1]$

Prediction for matrix W: entry $W(i_t, j_t)$

Regret = loss of alg – loss of best comparison matrix

Online Collaborative Filtering as OMP

Users: $\{1, 2, \ldots, m\}$
Movies: $\{1, 2, \ldots, n\}$

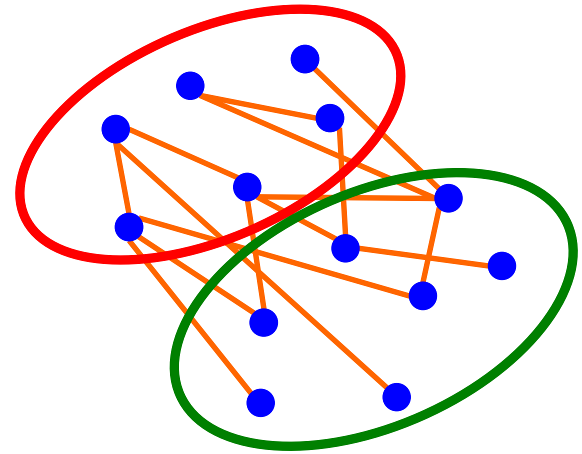
On round t:
User i_t arrives and is interested in movie j_t
Output predicted rating p_t in $[-1, 1]$
User responds with actual rating r_t
Loss $= (p_t - r_t)^2$

Comparison class: $\boldsymbol{W}$ = all $m \times n$ matrices with entries in $[-1, 1]$ of trace norm $\leq \tau$
For each such matrix W, predicted rating $= W(i_t, j_t)$

Online Max Cut as OMP

2 political parties

Voters: {1, 2, …, n}



On round t:

Voters i_t, j_t arrive

Output prediction: votes agree or disagree

Loss = 1 if incorrect prediction, 0 o.w.

Comparison class: all possible bipartitions

Bipartition prediction = "agree" if i_t, j_t in same partition, "disagree" o.w.

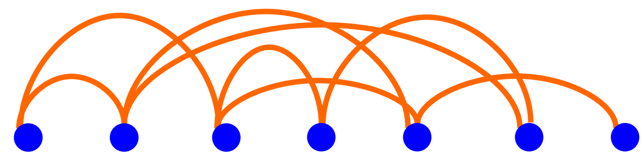
$\boldsymbol{W}$ = all 2^n "cut matrices" W_S corresponding to subsets S of $[n]$

$W_S(i, j) = 0$ if both i, j in S or $[n] \setminus S$,

$= 1$ o.w.

	S	S	[n] \ S	[n] \ S	[n] \ S
S	0	0	1	1	1
S	0	0	1	1	1
[n] \ S	1	1	0	0	0
[n] \ S	1	1	0	0	0
[n] \ S	1	1	0	0	0

Online Gambling as OMP



Teams: {1, 2, …, n}

In round t:
Teams i_t, j_t compete
Output: prediction which team will win
Loss = 1 if incorrect prediction, 0 o.w.

Comparison class: all possible permutations π
Permutation π prediction = i_t if $\pi(i_t) \le \pi(j_t)$; j_t o.w.

W = all n! "permutation matrices" W_π corresponding to permutations π

$W_\pi (i, j) = 1$ if $\pi(i) \le \pi(j)$
$= 0$ o.w.

	1	2	…	…	n
1	1	0	1	1	0
2	1	1	1	1	1
⋮	0	0	1	1	0
⋮	0	0	0	1	0
n	1	0	1	1	1

	π(1)	π(2)	…	…	π(n)
π(1)	1	1	1	1	1
π(2)	0	1	1	1	1
⋮	0	0	1	1	1
⋮	0	0	0	1	1
π(n	0	0	0	0	1

Decomposability

Symmetric square matrix of order $m + n$

W is **(β, τ)-decomposable** if

$$\begin{bmatrix} 0 & W \\ W^T & 0 \end{bmatrix} = P - N$$

where

P, N are positive semidefinite

Diagonal entries $P_{ii}, N_{ii} \leq \beta$

Sum of traces $Tr(P) + Tr(N) \leq \tau$

Class ***W*** is (β, τ)-decomposable if every W in ***W*** is.

Main Result for (®,τ)-decomposable OMP

An efficient algorithm for OMP with (β, τ)-decomposable ***W*** and Lipschitz losses with regret bound

$$\text{Regret} \;\leq\; O\left(\sqrt{\beta\tau \log(m+n)T}\right)$$

The Technology

Online Learning problem: in round t,

- Learner chooses density (i.e. psd, trace 1) matrix X_t
- Nature reveals loss matrix M_t with eigenvalues in $[-1, 1]$
- Learner suffers loss $\mathrm{Tr}(M_t X_t)$

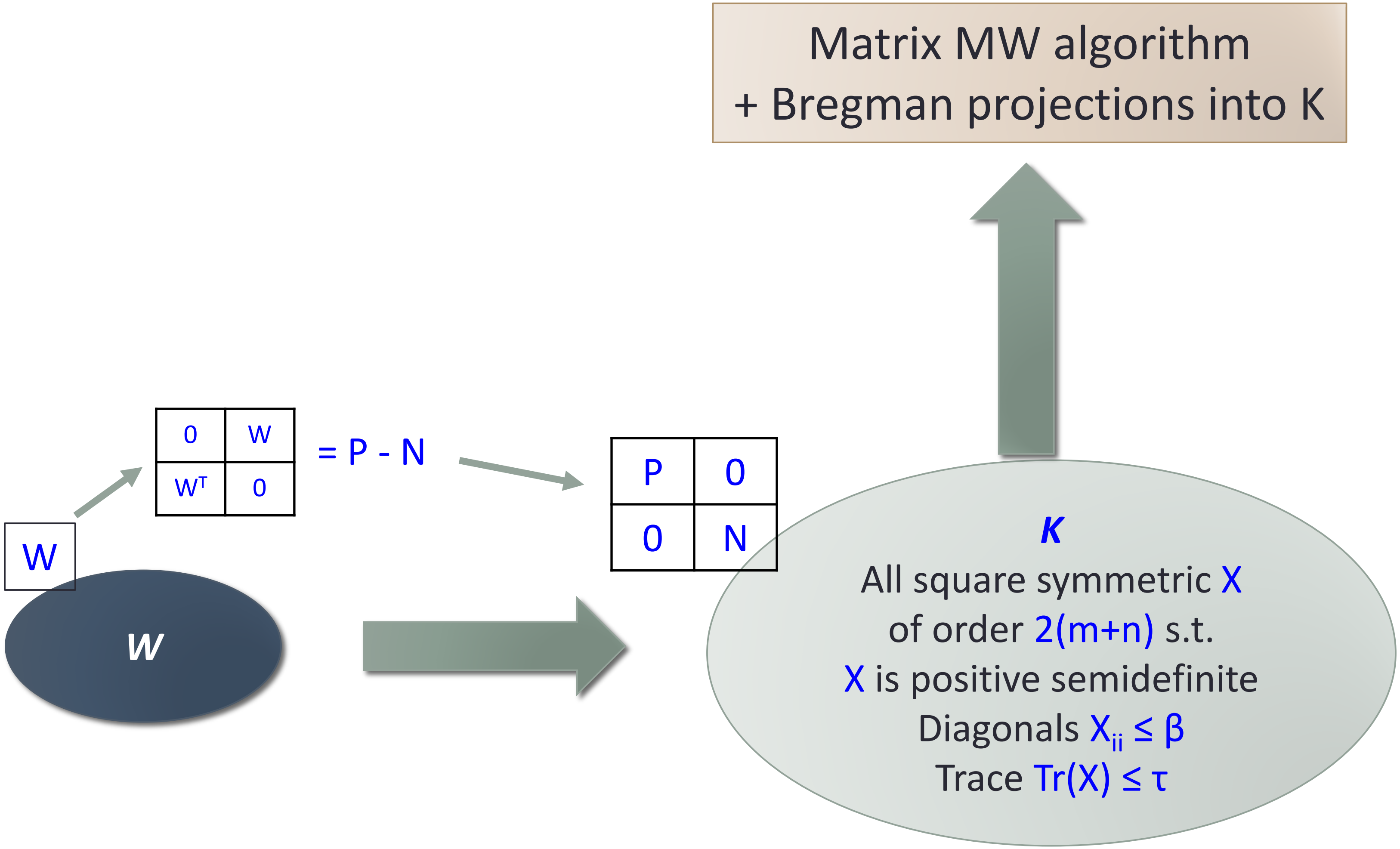
Goal: minimize regret = loss of learner – loss of best density matrix

Matrix Exponentiated Gradient [Tsuda, Rätsch, Warmuth ’06]/
Matrix Multiplicative Weights [Arora, K. ‘07] algorithm

$$X_t = \exp(-\eta \sum_{s<t} M_s)/Z_t$$

Theorem: $\mathrm{Regret} = O(\sqrt{\log(n) T})$

Overview of Algorithm for OMP



Decomposability Theorems

Online Collaborative Filtering

Trace norm $\leq \tau$ matrices are $(\sqrt{m+n}, 2\tau)$-decomposable.

$$\text{Regret} \;\leq\; O\left(\sqrt{\tau\sqrt{n}\log(n)T}\right)$$

Online Max Cut

Cut matrices W_S are $(½, 2n)$-decomposable.

$$\text{Regret} \;\leq\; O\left(\sqrt{n\log(n)T}\right)$$

Online Gambling

Permutation matrices W_π are $(O(\log n), O(n \log n))$-decomposable.

$$\text{Regret} \;\leq\; O\left(\sqrt{n\log^3(n)T}\right)$$

Decomposability Theorems

Online Collaborative Filtering

Trace norm $\leq \tau$ matrices are $(\sqrt{m+n}, 2\tau)$-decomposable.

$$\text{Regret} \;\leq\; O\left(\sqrt{\tau\sqrt{n}\log(n)T}\right)$$

Online Max Cut

Cut matrices W_S are $(½, 2n)$-decomposable.

$$\text{Regret} \;\leq\; O\left(\sqrt{n\log(n)T}\right)$$

Online Gambling

Permutation matrices W_π are $(O(\log n), O(n \log n))$-decomposable.

$$\text{Regret} \;\leq\; O\left(\sqrt{n\log^3(n)T}\right)$$

Decomposability for OCF

Thm: Any symmetric matrix M of order n with entries in [-1, 1] and trace norm τ is (√n, τ)-decomposable

Eigenvalue decomposition: $M = \sum_i \lambda_i v_i v_i^\top$

Define $P = \sum_{\lambda_i > 0} \lambda_i v_i v_i^\top$ and $N = \sum_{\lambda_i < 0} |\lambda_i| v_i v_i^\top$

Clearly Tr(P) + Tr(N) = trace-norm(M) = τ.

$$(P + N)^2 = \sum_i \lambda_i^2 v_i v_i^\top = M^2$$

Diagonals of $(P + N)^2 = M^2$ bounded by n.
So diagonals of (P + N) bounded by √n.
So diagonals of P, N bounded by √n.

Decomposability Theorems

<u>Online Collaborative Filtering</u>

Trace norm $\leq \tau$ matrices are $(\sqrt{m + n}, 2\tau)$-decomposable.

$$\text{Regret} \;\leq\; O\left(\sqrt{\tau\sqrt{n}\log(n)T}\right)$$

<u>Online Max Cut</u>

Cut matrices W_S are $(\frac{1}{2}, 2n)$-decomposable.

$$\text{Regret} \;\leq\; O\left(\sqrt{n\log(n)T}\right)$$

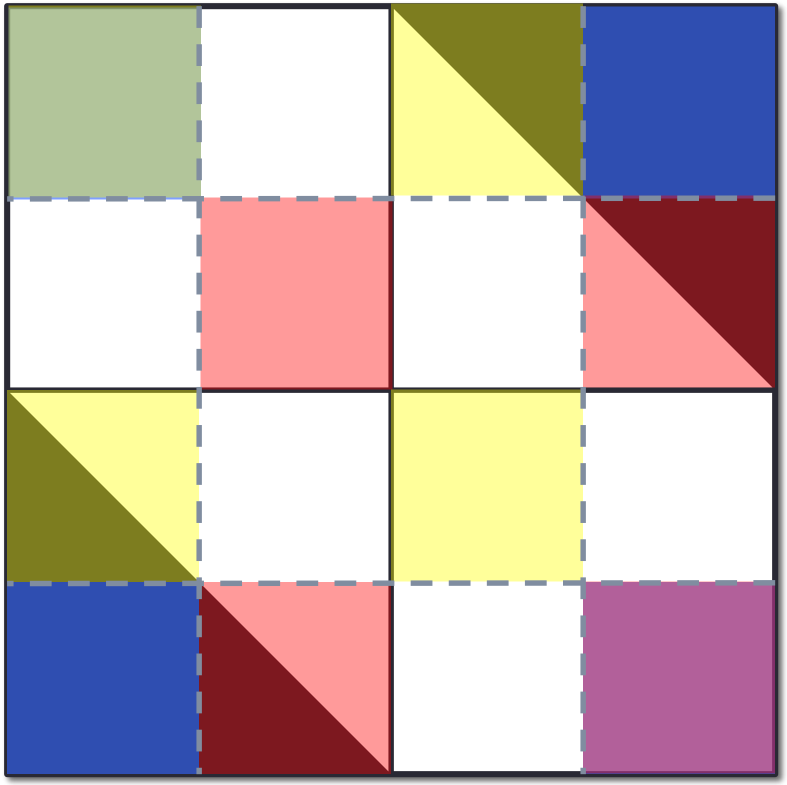
<u>Online Gambling</u>

Permutation matrices W_π are $(O(\log n), O(n \log n))$-decomposable.

$$\text{Regret} \;\leq\; O\left(\sqrt{n\log^3(n)T}\right)$$

Decomposability for Online Gambling

Thm: The all 1's upper triangular matrix of order n is (O(log n), O(n log n))-decomposable.



T(n) = One rank-1 matrix + two non-overlapping T(n/2)

B(n) = 1 + B(n/2)

B(n) = O(log n).

Concluding Remarks

- Gave near-optimal algorithms for various online matrix prediction problems
- Exploited *spectral structure* of comparison matrices to get near-tight convex relaxations
- Solved 2 COLT open problems from [Abernethy '10] and [Shamir, Srebro '11]

- **Open problem:** get rid of the logarithmic gap between upper and lower bounds
  - Decompositions in the paper are optimal up to constant factors, so a fundamentally different algorithm seems necessary

Thanks!