The Free Matrix Lunch

Wouter M. Koolen
Royal Holloway University of London

Wojciech Kotłowski
CWI Centrum Wiskunde & Informatica

Manfred K. Warmuth
University of California Santa Cruz

Tuesday 24th April, 2012

The open problem (Warmuth, COLT 2007)

Recent interest in matrix generalizations of classical prediction tasks:

- Matrix Hedge (PCA)
- Matrix Winnow (learning subspaces)
- Matrix Exponentiated Gradient (regression)

The open problem (Warmuth, COLT 2007)

Recent interest in matrix generalizations of classical prediction tasks:

- Matrix Hedge (PCA)
- Matrix Winnow (learning subspaces)
- Matrix Exponentiated Gradient (regression)

In each case the matrix generalizations of classical algorithms have performance guarantees (worst-case regret bounds) *identical* to the classical tasks

The open problem (Warmuth, COLT 2007)

Recent interest in matrix generalizations of classical prediction tasks:

- Matrix Hedge (PCA)
- Matrix Winnow (learning subspaces)
- Matrix Exponentiated Gradient (regression)

In each case the matrix generalizations of classical algorithms have performance guarantees (worst-case regret bounds) *identical* to the classical tasks

Symmetric matrices have n^2 parameters and vectors n parameters. Thus matrices should be *harder* to learn!

The open problem (Warmuth, COLT 2007)

Recent interest in matrix generalizations of classical prediction tasks:

- Matrix Hedge (PCA)
- Matrix Winnow (learning subspaces)
- Matrix Exponentiated Gradient (regression)

In each case the matrix generalizations of classical algorithms have performance guarantees (worst-case regret bounds) *identical* to the classical tasks

Symmetric matrices have n^2 parameters and vectors n parameters. Thus matrices should be *harder* to learn!

What is going on? Are classical bounds loose, or is there a

Free Matrix Lunch?

This talk I

Fundamental task: predicting n-ary sequence with logarithmic loss

- Strong intuition from several interpretations

  probability forecasting - data compression - investment
- Algorithms derived from various principles

  Bayesian inference - universal coding - optimization - minimax
- Popular. Extremely well-studied. Simple. Often one-line proofs.

This talk I

Fundamental task: predicting *n*-ary sequence with logarithmic loss

- Strong intuition from several interpretations

  probability forecasting - data compression - investment
- Algorithms derived from various principles

  Bayesian inference - universal coding - optimization - minimax
- Popular. Extremely well-studied. Simple. Often one-line proofs.

We generalise the *problem* and lift the *algorithms* to the matrix domain.

This talk I

Fundamental task: predicting n-ary sequence with logarithmic loss

- Strong intuition from several interpretations

  probability forecasting - data compression - investment
- Algorithms derived from various principles

  Bayesian inference - universal coding - optimization - minimax
- Popular. Extremely well-studied. Simple. Often one-line proofs.

We generalise the *problem* and lift the *algorithms* to the matrix domain.

We prove and explain a

This talk II

We then consider the second fundamental Hedge or dot loss setting.

This talk II

We then consider the second fundamental Hedge or dot loss setting.

Here we show matrix prediction is *strictly harder*

Outline

Probability vector prediction

for trial $t = 1, 2, \ldots$ **do**

  Alg predicts with probability vector $\boldsymbol{\omega}_t$

  Nat returns basis vector $\boldsymbol{x}_t \in \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n\}$

  Alg incurs loss $-\log(\boldsymbol{\omega}_t^\top \boldsymbol{x}_t)$

end for

Probability vector prediction

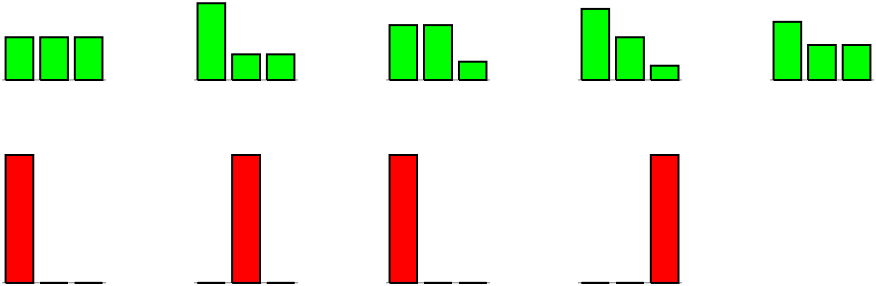
for trial $t = 1, 2, \ldots$ **do**

  Alg predicts with probability vector $\boldsymbol{\omega}_t$

  Nat returns basis vector $\boldsymbol{x}_t \in \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n\}$

  Alg incurs loss $-\log(\boldsymbol{\omega}_t^\top \boldsymbol{x}_t)$

end for



Evaluation

Regret is loss of **Alg** minus the loss of the best fixed prediction:

$$\mathcal{R}_T \;:=\; \sum_{t=1}^{T} -\log\left(\boldsymbol{\omega}_t^\top \boldsymbol{x}_t\right) - \inf_{\boldsymbol{\omega}} \sum_{t=1}^{T} -\log\left(\boldsymbol{\omega}^\top \boldsymbol{x}_t\right).$$

Evaluation

Regret is loss of **Alg** minus the loss of the best fixed prediction:

$$\mathcal{R}_T \;:=\; \sum_{t=1}^{T} -\log\left(\boldsymbol{\omega}_t^\top \boldsymbol{x}_t\right) - \inf_{\boldsymbol{\omega}} \sum_{t=1}^{T} -\log\left(\boldsymbol{\omega}^\top \boldsymbol{x}_t\right).$$

In this problem we compete with the *empirical Shannon entropy*:

$$\inf_{\boldsymbol{\omega}} \sum_{t=1}^{T} -\log\left(\boldsymbol{\omega}^\top \boldsymbol{x}_t\right) \;=\; T\,H\left(\boldsymbol{\omega}^*\right) \qquad \text{where } \boldsymbol{\omega}^* = \frac{\sum_{t=1}^{T} \boldsymbol{x}_t}{T}$$

$\boldsymbol{\omega}^*$ is the *maximum likelihood estimator*

Goal: design online algorithms with low worst-case regret

Algorithms

For the Laplace predictor

$$\boldsymbol{\omega}_{t+1} := \frac{\sum_{q=1}^{t} \boldsymbol{x}_q + \mathbf{1}}{t+n} \qquad \mathcal{R}_T \leq (n-1)\log(T+1)$$

whereas for the Krychevsky-Trofimoff predictor

$$\boldsymbol{\omega}_{t+1} := \frac{\sum_{q=1}^{t} \boldsymbol{x}_q + \mathbf{1}/2}{t+n/2} \qquad \mathcal{R}_T \leq \frac{n-1}{2}\left(\log(T+1) + \log(\pi)\right)$$

Algorithms

For the Laplace predictor

$$\boldsymbol{\omega}_{t+1} := \frac{\sum_{q=1}^{t} \boldsymbol{x}_q + \mathbf{1}}{t+n} \qquad \mathcal{R}_T \le (n-1)\log(T+1)$$

whereas for the Krychevsky-Trofimoff predictor

$$\boldsymbol{\omega}_{t+1} := \frac{\sum_{q=1}^{t} \boldsymbol{x}_q + \mathbf{1}/2}{t+n/2} \qquad \mathcal{R}_T \le \frac{n-1}{2}\left(\log(T+1) + \log(\pi)\right)$$

Other algorithms include

Last Step Minimax $\quad \mathcal{R}_T \le \dfrac{n-1}{2}\log(T+1) + 1$

Shtarkov $\quad \mathcal{R}_T \le \dfrac{n-1}{2}\left(\log(T+1) - \log(n-2) + 1\right)$

Outline

Density matrix prediction

for trial $t = 1, 2, \ldots$ **do**

Alg predicts with density matrix $\boldsymbol{W}_t$

Nat returns dyad $\boldsymbol{x}_t \boldsymbol{x}_t^\top$

Alg incurs loss $-\boldsymbol{x}_t^\top \mathbf{log}(\boldsymbol{W}_t) \boldsymbol{x}_t$

end for

Density matrix prediction

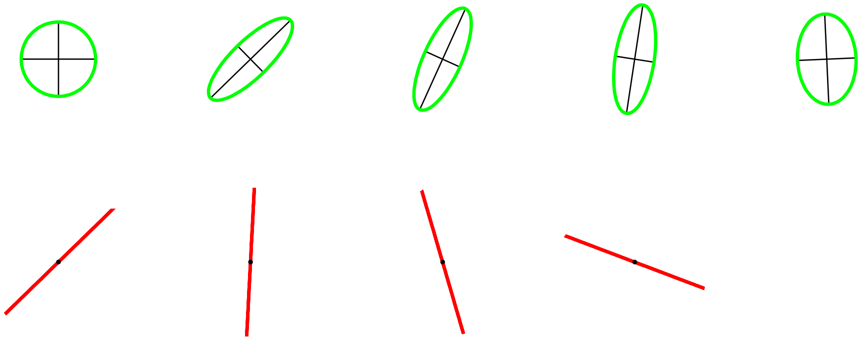
for trial $t = 1, 2, \ldots$ **do**

Alg predicts with density matrix $\boldsymbol{W}_t$

Nat returns dyad $\boldsymbol{x}_t \boldsymbol{x}_t^\top$

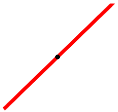
Alg incurs loss $-\boldsymbol{x}_t^\top \mathbf{log}(\boldsymbol{W}_t)\boldsymbol{x}_t$

end for



The outcomes: dyads

A dyad $\boldsymbol{x}\boldsymbol{x}^\top$ is a rank-one matrix, where $\boldsymbol{x}$ is a vector in $\mathbb{R}^n$ of unit length.



A dyad is a classical outcome in an arbitrary orthonormal basis:

$$\boldsymbol{x}\boldsymbol{x}^\top = \boldsymbol{U}^\top \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \boldsymbol{U}$$

There are continuously many dyads.

The predictions: density matrices

A density matrix $\boldsymbol{W}$ is a convex combination of dyads.

Positive-semidefinite matrix $\boldsymbol{W}$ of unit trace

The predictions: density matrices

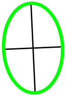
A density matrix $\boldsymbol{W}$ is a convex combination of dyads.

Positive-semidefinite matrix $\boldsymbol{W}$ of unit trace

A density matrix is a probability vector in an arbitrary orthonormal basis:

Decomposition:

$$\boldsymbol{W} = \sum_{i=1}^{n} \alpha_i \, \boldsymbol{a}_i \boldsymbol{a}_i^\top$$



eigenvalues $\boldsymbol{\alpha}$	probability vector
eigenvectors $\boldsymbol{a}_i$	orthonormal system

The predictions: density matrices

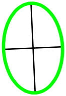
A density matrix $\boldsymbol{W}$ is a convex combination of dyads.

Positive-semidefinite matrix $\boldsymbol{W}$ of unit trace

A density matrix is a probability vector in an arbitrary orthonormal basis:

Decomposition:

$$\boldsymbol{W} = \sum_{i=1}^{n} \alpha_i \, \boldsymbol{a}_i \boldsymbol{a}_i^\top$$



eigenvalues $\boldsymbol{\alpha}$	probability vector
eigenvectors $\boldsymbol{a}_i$	orthonormal system

Note: different convex combinations of dyads may result in the same density matrix.

The loss: matrix log loss

The logarithm of a density matrix $\boldsymbol{W} = \sum_i \alpha_i \boldsymbol{a}_i \boldsymbol{a}_i^\top$ is defined by

$$\mathbf{log}(\boldsymbol{W}) \;=\; \sum_i \log(\alpha_i)\, \boldsymbol{a}_i \boldsymbol{a}_i^\top.$$

Discrepancy between prediction $\boldsymbol{W}$ and dyad $\boldsymbol{x}\boldsymbol{x}^\top$: matrix log loss

$$-\boldsymbol{x}^\top \mathbf{log}(\boldsymbol{W})\boldsymbol{x}$$

The classical case

If **Alg** and **Nat** play in the same eigensystem, say

$$\boldsymbol{W} = \sum_i \omega_i \boldsymbol{e}_i \boldsymbol{e}_i^\top \qquad \text{and} \qquad \boldsymbol{x} = \boldsymbol{e}_j$$

then matrix log loss becomes classical log loss

$$-\boldsymbol{x}^\top \mathbf{log}(\boldsymbol{W})\boldsymbol{x} = -\boldsymbol{x}^\top \sum_i \log(\omega_i)\boldsymbol{e}_i \boldsymbol{e}_i^\top \boldsymbol{x} = -\log(\alpha_j) = -\log(\boldsymbol{\omega}^\top \boldsymbol{x})$$

The classical case

If **Alg** and **Nat** play in the same eigensystem, say

$$\boldsymbol{W} = \sum_i \omega_i \boldsymbol{e}_i \boldsymbol{e}_i^\top \qquad \text{and} \qquad \boldsymbol{x} = \boldsymbol{e}_j$$

then matrix log loss becomes classical log loss

$$-\boldsymbol{x}^\top \mathbf{log}(\boldsymbol{W})\boldsymbol{x} = -\boldsymbol{x}^\top \sum_i \log(\omega_i) \boldsymbol{e}_i \boldsymbol{e}_i^\top \boldsymbol{x} = -\log(\alpha_j) = -\log(\boldsymbol{\omega}^\top \boldsymbol{x})$$

But both players can deviate. Who is to gain?

Matrix log loss is proper

The Von Neumann or Quantum entropy

$$H(\boldsymbol{A}) \;=\; -\operatorname{tr}(\boldsymbol{A}\,\mathbf{log}\,\boldsymbol{A})$$

equals the Shannon entropy of eigenvalues $\boldsymbol{\alpha}$ of $\boldsymbol{A}$.

We now compete with the *empirical Von Neumann entropy*:

$$\inf_{\boldsymbol{W}} \sum_{t=1}^{T} -\boldsymbol{x}_t^\top \,\mathbf{log}(\boldsymbol{W})\boldsymbol{x}_t \;=\; T\,H(\boldsymbol{W}^*) \quad \text{where} \quad \boldsymbol{W}^* \;=\; \frac{\sum_{t=1}^{T} \boldsymbol{x}_t \boldsymbol{x}_t^\top}{T}$$

Matrix Algorithms

Matrix Laplace predicts with $\boldsymbol{W}_{t+1} =$

$$\operatorname*{argmin}_{\boldsymbol{W}} \left\{ \underbrace{-\operatorname{tr}(\mathbf{log}\, \boldsymbol{W})}_{n \text{ uniform outcomes}} + \sum_{q=1}^{t} -\boldsymbol{x}_q^\top \mathbf{log}(\boldsymbol{W}) \boldsymbol{x}_q \right\} = \frac{\sum_{q=1}^{t} \boldsymbol{x}_q \boldsymbol{x}_q^\top + \boldsymbol{I}}{t+n}$$

and Matrix KT predicts with $\boldsymbol{W}_{t+1} =$

$$\operatorname*{argmin}_{\boldsymbol{W}} \left\{ \underbrace{-\frac{1}{2}\operatorname{tr}(\mathbf{log}\, \boldsymbol{W})}_{\frac{n}{2} \text{ uniform outcomes}} + \sum_{q=1}^{t} -\boldsymbol{x}_q^\top \mathbf{log}(\boldsymbol{W}) \boldsymbol{x}_q \right\} = \frac{\sum_{q=1}^{t} \boldsymbol{x}_q \boldsymbol{x}_q^\top + \boldsymbol{I}/2}{t+n/2}$$

Two Free Matrix Lunches

Theorem

Classical and matrix worst-case regrets coincide for Laplace and for KT.

Two Free Matrix Lunches

Theorem

Classical and matrix worst-case regrets coincide for Laplace and for KT.

Proof for Laplace.

Let $\boldsymbol{W}_t^*$ denote the best density matrix for the first t outcomes. The regret of matrix Laplace can be bounded as follows:

$$\mathcal{R}_T = \sum_{t=1}^{T} \ell(\boldsymbol{W}_t, \boldsymbol{x}_t \boldsymbol{x}_t^\top) - \sum_{t=1}^{T} \ell(\boldsymbol{W}_T^*, \boldsymbol{x}_t \boldsymbol{x}_t^\top) \leq \sum_{t=1}^{T} \Big(\ell(\boldsymbol{W}_t, \boldsymbol{x}_t \boldsymbol{x}_t^\top) - \ell(\boldsymbol{W}_t^*, \boldsymbol{x}_t \boldsymbol{x}_t^\top) \Big). \quad (1)$$

Now consider the t^{th} term in the right-hand sum. With $\boldsymbol{S}_t = \sum_{q=1}^{t} \boldsymbol{x}_q \boldsymbol{x}_q^\top$

$$-\boldsymbol{x}_t^\top \left(\mathbf{log} \frac{\boldsymbol{S}_{t-1} + \boldsymbol{I}}{t-1+n} - \mathbf{log} \frac{\boldsymbol{S}_t}{t} \right) \boldsymbol{x}_t = \log \left(\frac{t-1+n}{t} \right) - \boldsymbol{x}_t^\top (\mathbf{log}(\boldsymbol{S}_{t-1} + \boldsymbol{I}) - \mathbf{log}\, \boldsymbol{S}_t) \boldsymbol{x}_t .$$

The matrix part is non-positive since $\boldsymbol{S}_{t-1} + \boldsymbol{I} \succeq \boldsymbol{S}_t$, and the logarithm is matrix monotone. It is zero for any sequence of identical dyads and (1) holds with equality since $\boldsymbol{W}_t^* = \boldsymbol{W}_T^*$ for all $t \leq T$. The same upper bound is also met by classical Laplace on any sequence of identical outcomes. □

But why...?

If **Alg** plays Laplace or KT, then **Nat** will never go out-eigensystem.

The classical case is the worst case.

Best eigenvectors are *observed*. No regret.

But why...?

If **Alg** plays Laplace or KT, then **Nat** will never go out-eigensystem.

The classical case is the worst case.

Best eigenvectors are *observed*. No regret.

If **Alg** plays in the eigensystem of past data, will **Nat** do too?

- For matrix log loss: only pathological counterexamples
- For other losses: real counterexamples

Shtarkov: The Queen of Lunches

In the classical case the minimax algorithm is due to Shtarkov.

Ultimate open problem: is the *classical minimax regret*

$$\min_{\boldsymbol{\omega}_1} \max_{\boldsymbol{x}_1} \cdots \min_{\boldsymbol{\omega}_T} \max_{\boldsymbol{x}_T} \sum_{t=1}^{T} - \log\left(\boldsymbol{\omega}_t^\top \boldsymbol{x}_t\right) - T\, H\left(\frac{\sum_{t=1}^{T} \boldsymbol{x}_t}{T}\right)$$

equal to the *matrix minimax regret*

$$\min_{\boldsymbol{W}_1} \max_{\boldsymbol{x}_1} \cdots \min_{\boldsymbol{W}_T} \max_{\boldsymbol{x}_T} \sum_{t=1}^{T} -\boldsymbol{x}_t^\top \,\mathbf{log}(\boldsymbol{W}_t)\boldsymbol{x}_t - T\, H\left(\frac{\sum_{t=1}^{T} \boldsymbol{x}_t \boldsymbol{x}_t^\top}{T}\right)$$

Only numerical evidence for this claim and intermediate conjectures.

Outline

The story for trace loss

The Free Matrix Lunch depends on the choice of loss.

The dot loss generalises to the trace loss:

$$\ell(\boldsymbol{\omega}, \boldsymbol{l}) = \boldsymbol{\omega}^\top \boldsymbol{l} \qquad\qquad \ell(\boldsymbol{W}, \boldsymbol{L}) = \mathrm{tr}(\boldsymbol{W}\boldsymbol{L})$$

for $\boldsymbol{l} \in \{0, 1\}^n$ and symmetric $\boldsymbol{L}$ with eigenvalues in $\{0, 1\}^n$.

The story for trace loss

The Free Matrix Lunch depends on the choice of loss.

The dot loss generalises to the trace loss:

$$\ell(\boldsymbol{\omega}, \boldsymbol{l}) = \boldsymbol{\omega}^\top \boldsymbol{l} \qquad\qquad \ell(\boldsymbol{W}, \boldsymbol{L}) = \operatorname{tr}(\boldsymbol{W}\boldsymbol{L})$$

for $\boldsymbol{l} \in \{0,1\}^n$ and symmetric $\boldsymbol{L}$ with eigenvalues in $\{0,1\}^n$.

Hedge and Matrix Hedge

$$\boldsymbol{\omega}_t = \boldsymbol{\omega}_{t-1} e^{-\eta \boldsymbol{l}_t} / Z_t \qquad\qquad \boldsymbol{W}_t = \mathbf{exp}(\mathbf{log}\, \boldsymbol{W}_{t-1} - \eta \boldsymbol{L}_t) / Z_t$$

The story for trace loss

The Free Matrix Lunch depends on the choice of loss.

The dot loss generalises to the trace loss:

$$\ell(\boldsymbol{\omega}, \boldsymbol{l}) = \boldsymbol{\omega}^\top \boldsymbol{l} \qquad\qquad \ell(\boldsymbol{W}, \boldsymbol{L}) = \mathrm{tr}(\boldsymbol{W}\boldsymbol{L})$$

for $\boldsymbol{l} \in \{0,1\}^n$ and symmetric $\boldsymbol{L}$ with eigenvalues in $\{0,1\}^n$.

Hedge and Matrix Hedge

$$\boldsymbol{\omega}_t \;=\; \boldsymbol{\omega}_{t-1} e^{-\eta \boldsymbol{l}_t} / Z_t \qquad\qquad \boldsymbol{W}_t \;=\; \mathbf{exp}(\mathbf{log}\, \boldsymbol{W}_{t-1} - \eta \boldsymbol{L}_t) / Z_t$$

with tuned *learning rate* η both have regrets bounded by

$$\sqrt{\frac{T \log n}{2}} \qquad \text{as well as} \qquad \sqrt{2L^* \log n} + \log n.$$

No free lunch for trace loss

In dimension $n = 2$ the minimax regrets for T trials are

$$\sqrt{\frac{T+1}{2\pi}} \qquad\qquad \sqrt{\frac{T}{4}}$$

Far from a free lunch.

No free lunch for trace loss

In dimension $n = 2$ the minimax regrets for T trials are

$$\sqrt{\frac{T+1}{2\pi}} \qquad\qquad \sqrt{\frac{T}{4}}$$

Far from a free lunch.

The matrix case is *different* and *easier* to analyse:

- **Alg** predicts *deterministically* in past eigensystem
- **Nat** plays Hadamard (worse than classical randomisation)

No free lunch for trace loss

In dimension $n = 2$ the minimax regrets for T trials are

$$\sqrt{\frac{T+1}{2\pi}} \qquad\qquad \sqrt{\frac{T}{4}}$$

Far from a free lunch.

The matrix case is *different* and *easier* to analyse:

- **Alg** predicts *deterministically* in past eigensystem
- **Nat** plays Hadamard (worse than classical randomisation)

We submitted the case $n > 2$ as an open problem to COLT 2012.

Outline

Summary

- Matrix log loss
  - Learning a matrix of n^2 parameters with regret for n
  - Eigenvectors are learned for free
  - Classical data is worst-case
- Trace loss
  - No free matrix lunch
  - **Nat** exploits matrix power

Many open problems

- Does the free matrix lunch hold for the matrix minimax algorithm? cf. Shtarkov
- Same questions for other losses
- What properties of the loss function and algorithm cause the free matrix lunch to occur? Proper scoring rules?
- Is there a general regret-bound preserving lift of classical algorithms to matrix log loss prediction?

Thank you!