

On-line Variance Minimization

Manfred Warmuth Dima Kuzmin

University of California - Santa Cruz

19th Annual Conference on Learning Theory

Outline

- 1 Variance
- 2 Variance minimization on simplex
- 3 Variance minimization on unit sphere
- 4 On-line PCA
- 5 What's next?

Outline

- 1 Variance
- 2 Variance minimization on simplex
- 3 Variance minimization on unit sphere
- 4 On-line PCA
- 5 What's next?

Variance

- Symmetric positive definite matrix \mathbf{C} is covariance matrix of some random vector $\mathbf{p} \in \mathbb{R}^n$

$$\mathbf{C} = \mathbb{E} \left((\mathbf{p} - \mathbb{E}(\mathbf{p}))(\mathbf{p} - \mathbb{E}(\mathbf{p}))^\top \right)$$

- The variance along any vector \mathbf{w} is

$$\begin{aligned} \mathbb{V}(\mathbf{p}^\top \mathbf{w}) &= \mathbb{E} \left(\left(\mathbf{p}^\top \mathbf{w} - \mathbb{E}(\mathbf{p}^\top \mathbf{w}) \right)^2 \right) \\ &= \mathbb{E} \left(\left((\mathbf{p}^\top - \mathbb{E}(\mathbf{p}^\top)) \mathbf{w} \right)^2 \right) \\ &= \mathbf{w}^\top \mathbb{E} \left((\mathbf{p} - \mathbb{E}(\mathbf{p}))(\mathbf{p} - \mathbb{E}(\mathbf{p}))^\top \right) \mathbf{w} \end{aligned}$$

Variance minimization problem

Setup

On-line learning problem

- Pick a vector \mathbf{w}_t
- Receive a covariance matrix \mathbf{C}_t
- Loss is variance along vector \mathbf{w}_t : $L_t(\mathbf{w}_t) = \mathbf{w}_t^\top \mathbf{C}_t \mathbf{w}_t$

Goal

Achieve loss close to variance along best vector picked in hindsight

$$L_{\text{best}} = \inf_{\mathbf{u}} \mathbf{u}^\top \left(\sum_t \mathbf{C}_t \right) \mathbf{u}$$

Outline

- 1 Variance
- 2 Variance minimization on simplex**
- 3 Variance minimization on unit sphere
- 4 On-line PCA
- 5 What's next?

\mathbf{w}_t is a probability vector

Variance minimization in portfolio selection:

- \mathbf{p} is a random vector of relative stock price changes
- \mathbf{w} is stock investment proportions into n stocks
- $\mathbf{w} \cdot \mathbf{p}$ is our capital gain
- $\mathbf{w}^\top \mathbf{C} \mathbf{w}$ is variance of gain

Exponentiated Gradient Algorithm

- Maintains probability vector

$$w_{t+1,i} = \frac{w_{t,i} e^{-\eta(\mathbf{C}_t \mathbf{w}_t)_i}}{Z}$$

- Motivation

$$\mathbf{w}_{t+1} = \inf_{\mathbf{w} \in \text{simplex}} \sum_i w_i \ln \frac{w_i}{w_{t,i}} + \eta \mathbf{w}^\top \mathbf{C}_t \mathbf{w}$$

- Bound

$$\text{Var}_{\text{alg}} \leq \text{Var}_{\text{best}} + 2\sqrt{r \text{Var}_{\text{best}} \log n} + r \log n$$

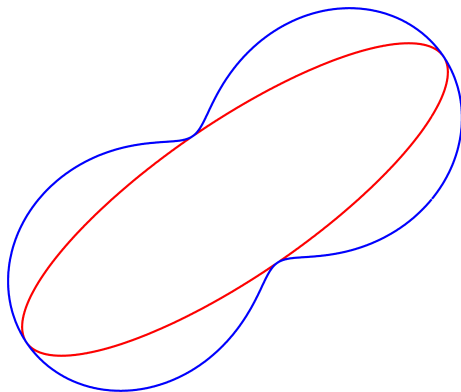
To do: Minimize tradeoff between gain and variance

- Pick a vector \mathbf{w}_t
- Receive a covariance matrix \mathbf{C}_t and gain vector \mathbf{p}_t
- Charge $\underbrace{-\mathbf{w}_t^\top \mathbf{p}_t}_{\text{gain}} + \delta \underbrace{\mathbf{w}_t^\top \mathbf{C}_t \mathbf{w}_t}_{\text{variance}}$

Outline

- 1 Variance
- 2 Variance minimization on simplex
- 3 Variance minimization on unit sphere**
- 4 On-line PCA
- 5 What's next?

Variance of unit vectors



The ellipse is plot of vector $\mathbf{C}\mathbf{w}$, where \mathbf{w} is unit vector

The outer figure eight is direction \mathbf{w} times the variance $\mathbf{w}^T \mathbf{C}\mathbf{w}$

For an eigenvector, variance equals the eigenvalue and touches ellipse

Mixtures of Directions

- Our algorithm will pick a direction \mathbf{w}_i with probability ω_i
- **Expected variance**

$$\underbrace{\sum_i \omega_i \overbrace{\mathbf{w}_i^\top \mathbf{C} \mathbf{w}_i}^{\text{var.in.dir.}\mathbf{w}_i}}_{\text{expected variance}} = \sum_i \omega_i \text{tr}(\mathbf{C} \mathbf{w}_i \mathbf{w}_i^\top) = \text{tr}(\mathbf{C} \underbrace{\sum_i \omega_i \mathbf{w}_i \mathbf{w}_i^\top}_{\text{density matrix } \mathbf{W}})$$

Definition

$\mathbf{w}\mathbf{w}^\top$ for unit \mathbf{w} is called a **dyad**

- Symmetric positive definite matrix of rank one
- Trace one: $\text{tr}(\mathbf{w}\mathbf{w}^\top) = \mathbf{w}^\top \mathbf{w} = \|\mathbf{w}\|_2^2 = 1$
- Projection matrix onto direction \mathbf{w}

Density Matrices

- Convex combinations of dyads
- Symmetric positive definite matrices of trace one
- Eigenvalues form probability vector
- Many mixtures lead to the same matrix:

$$0.2 \text{ --- } + 0.3 \text{ / } + 0.5 \text{ | } = \text{ (ellipse) } = 0.29 \text{ \textbackslash } + 0.71 \text{ / }$$

- Can always be written as a convex combination of n dyads corresponding to eigenvectors

Diagonal case: $\sum_i \omega_i \mathbf{e}_i \mathbf{e}_i^T$

Variance Minimization with Density Matrices

Setup

- Pick density matrix $\mathbf{W}_t = \sum_i \omega_{t,i} \mathbf{w}_{t,i} \mathbf{w}_{t,i}^\top$
- Pick direction $\mathbf{w}_{t,i}$ with probability $\omega_{t,i}$
- Covariance matrix \mathbf{C}_t is obtained
- Loss is expected variance: $L_t(\mathbf{W}_t) = \sum_i \omega_{t,i} \mathbf{w}_{t,i}^\top \mathbf{C}_t \mathbf{w}_{t,i} = \text{tr}(\mathbf{W}_t \mathbf{C}_t)$

Goal

Do as well as best density matrix

- single dyad corresponding to smallest eigenvalue of $\sum_t \mathbf{C}_t$

Expert setting retained as diagonal case

$$\omega_t \cdot \mathbf{I}_t = \text{tr} \left(\begin{pmatrix} \omega_{t,1} & 0 & 0 & 0 \\ 0 & \omega_{t,2} & 0 & 0 \\ 0 & 0 & \omega_{t,3} & 0 \\ 0 & 0 & 0 & \omega_{t,4} \end{pmatrix} \begin{pmatrix} l_{t,1} & 0 & 0 & 0 \\ 0 & l_{t,2} & 0 & 0 \\ 0 & 0 & l_{t,3} & 0 \\ 0 & 0 & 0 & l_{t,4} \end{pmatrix} \right)$$

Deriving the Algorithm

$$\mathbf{W}_{t+1} = \underset{\mathbf{U} \text{ dens. mat.}}{\operatorname{arg\,inf}} \underbrace{\operatorname{tr}(\mathbf{U}(\log \mathbf{U} - \log \mathbf{W}_t))}_{\text{quantum relative entropy}} + \eta \underbrace{\operatorname{tr}(\mathbf{U}\mathbf{C}_t)}_{\text{expected variance}}$$

$$\mathbf{W}_{t+1} = \frac{\overbrace{\exp(\underbrace{\log \mathbf{W}_t}_{\text{symmetric}} - \eta \underbrace{\mathbf{C}_t}_{\text{symmetric}})}^{\text{symmetric positive definite}}}{\operatorname{tr}(\exp(\log \mathbf{W}_t - \eta \mathbf{C}_t))}$$

log, **exp** are matrix versions of logarithm and exponential

Bounds Generalize

$$\underbrace{\sum_{t=1}^T \text{tr}(\mathbf{W}_t \mathbf{C}_t)}_{\text{loss of algorithm}} \leq \frac{\underbrace{\eta \sum_{t=1}^T \text{tr}(\mathbf{U} \mathbf{C}_t)}_{\text{loss of comparator}} + \log n}{1 - e^{-\eta}}$$

loss of alg. \leq loss of best dens. $+ \sqrt{2 \text{loss of best dens.} \log n} + \log n$

Assumption: max. eigenvalue of $\mathbf{C}_t \leq 1$

Outline

- 1 Variance
- 2 Variance minimization on simplex
- 3 Variance minimization on unit sphere
- 4 On-line PCA**
- 5 What's next?

Best m experts

- Pick set of m experts $\{i_1, \dots, i_m\}$ based on probability vector \mathbf{w}_t
- Receive loss vector \mathbf{l}_t
- Loss is total loss of the m experts $l_{i_1} + \dots + l_{i_m}$
and expected loss $m \mathbf{w}_t \cdot \mathbf{x}_t$
- Update \mathbf{w}_t

Minimizing loss \mathbf{l} on m experts
equivalent to maximizing gain $-\mathbf{l}$ on $n - m$ experts

New trick: cap weights

Super predator algorithm

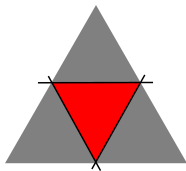


Preserves variety

Weights $\leq \frac{1}{m}$

$$\hat{w}_{t,i} = \frac{w_{t,i} e^{-\eta l_{t,i}}}{Z}$$

$$\mathbf{w}_{t+1} = \inf_{\mathbf{w} \text{ in capped simplex}} \Delta(\mathbf{w}, \hat{\mathbf{w}}_t)$$



expected loss of alg.

$$\leq \text{loss of best } m \text{ set} + \sqrt{2 \text{loss of best } m \text{ set } m \log n} + m \log n$$

Why capping?

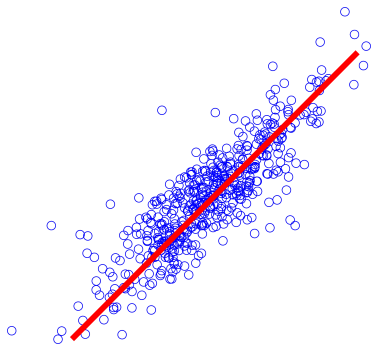
- m sets encoded as probability vectors $(0, \frac{1}{m}, 0, 0, \frac{1}{m}, 0, \frac{1}{m})$ called m -corners
- The convex hull of the m -corners = capped probability simplex
- We can effectively decompose any capped probability vector \mathbf{w} as convex combination of n m -corners

$$\mathbf{w} = \sum_j \alpha_j \mathbf{r}_j$$

Alternates to capping

- Dynamic programming: too expensive
- Follow the perturbed leader: cheap but inferior bounds

PCA



- On-line projection of data into low-dimensional subspace
- Best subspace in hindsight: k top eigenvectors of data covariance matrix

Rewrite quadratic loss into linear loss

Want rank k projection matrix \mathbf{P} that minimizes total square loss

$$\| \underbrace{\mathbf{P}}_k \mathbf{x} - \mathbf{x} \|_2^2 = \| \mathbf{P}\mathbf{x} - \mathbf{P}\mathbf{x} - (\mathbf{I} - \mathbf{P})\mathbf{x} \|_2^2 = \text{tr}(\underbrace{(\mathbf{I} - \mathbf{P})}_{n-k} \mathbf{x}\mathbf{x}^\top)$$

Want to choose $n - k$ dimensional subspace of minimum variance

Lift sets of expert alg. to matrices

- Pick $n - k$ dimensional subspace based on density matrix \mathbf{W}_t
- Choose complementary subspace \mathbf{P}_t
- Receive instance \mathbf{x}_t
- Incur loss $\|\mathbf{P}_t \mathbf{x}_t - \mathbf{x}_t\|_2^2$
and expected loss $(n - k) \text{tr}(\mathbf{W}_t \mathbf{x}_t \mathbf{x}_t^\top)$
- Update \mathbf{W}_t

Update and Winnow-like bound

expected loss of alg.

$$\leq \text{loss of best } k \text{ subspace} + \sqrt{2 \text{ loss of best } k \text{ subspace } k \log n} + k \log n$$

$$\widehat{\mathbf{W}}_t = \frac{\exp(\log \mathbf{W}_t - \eta \mathbf{x}_t \mathbf{x}_t^\top)}{\text{tr}(\exp(\log \mathbf{W}_t - \eta \mathbf{x}_t \mathbf{x}_t^\top))}$$

$$\mathbf{W}_{t+1} = \inf_{\substack{\mathbf{W} \text{ dens. matrix} \\ \text{w.eigenvals} \leq \frac{1}{n-k}}} \Delta(\mathbf{W}, \widehat{\mathbf{W}}_t)$$

Two families again

Regularize with $\|\mathbf{W} - \mathbf{W}_1\|_2^2$

[Crammer 06]

- $\mathbf{W} =$ lin. comb. of $\mathbf{x}_t \mathbf{x}_t^\top$
- Fast and kernelizable

Regularize with quantum relative entropy

- $\mathbf{W} = \frac{\exp(\text{lin. comb. of } \mathbf{x}_t \mathbf{x}_t^\top)}{Z}$
- Predict with random projection matrix
- Regret bounds instead of filtering loss

Key insight: Mixtures of experts generalize density matrices

Outline

- 1 Variance
- 2 Variance minimization on simplex
- 3 Variance minimization on unit sphere
- 4 On-line PCA
- 5 What's next?**

What's next?

- Shifting methodology from expert setting carries over
- Experiments
- Survey on “The Blessing and Curse of the Multiplicative Updates”
 - Adapt quickly
 - Loss of variety
 - Connections to Biology
- Work out probability calculus for density matrices

[Impromptu by Dima]