Superhedging and distributionally robust optimization with neural networks

Stephan Eckstein
joint work with Michael Kupper and Mathias Pohl

Robust Techniques in Quantitative Finance
University of Oxford
Robust optimization problems

Setup:
- Set of probability measures $Q$ on $\mathbb{R}^d$
- Continuous and bounded function $f : \mathbb{R}^d \to \mathbb{R}$.

Objective:
- Solve $$(P) := \sup_{\nu \in Q} \int f \, d\nu$$

Different choices of $Q$ lead to
- Constrained optimal transport (see e.g. Ekren and Soner (2018), includes martingale optimal transport)
- Distributionally robust optimization (see e.g. Esfahani and Kuhn (2015), Blanchet and Murthy (2016), Bartl et al. (2017), Obłój and Wiesel (2018))
- Mixtures of the above (Gao and Kleywegt (2017))
Robust optimization problems

Setup:
- Set of probability measures $\mathcal{Q}$ on $\mathbb{R}^d$
- Continuous and bounded function $f : \mathbb{R}^d \to \mathbb{R}$.

Objective:
- Solve

\[
(P) := \sup_{\nu \in \mathcal{Q}} \int f \, d\nu
\]

Different choices of $\mathcal{Q}$ lead to
- Constrained optimal transport (see e.g. Ekren and Soner (2018), includes martingale optimal transport)
- Distributionally robust optimization (see e.g. Esfahani and Kuhn (2015), Blanchet and Murthy (2016), Bartl et al. (2017), Obłój and Wiesel (2018))
- Mixtures of the above (Gao and Kleywegt (2017))
Robust optimization problems

Setup:
- Set of probability measures $Q$ on $\mathbb{R}^d$
- Continuous and bounded function $f : \mathbb{R}^d \to \mathbb{R}$.

Objective:
- Solve

\[
(P) := \sup_{\nu \in Q} \int f \, d\nu
\]

Different choices of $Q$ lead to
- Constrained optimal transport (see e.g. Ekren and Soner (2018), includes martingale optimal transport)
- Distributionally robust optimization (see e.g. Esfahani and Kuhn (2015), Blanchet and Murthy (2016), Bartl et al. (2017), Obłój and Wiesel (2018))
- Mixtures of the above (Gao and Kleywegt (2017))
Robust optimization problems

Setup:
- Set of probability measures $Q$ on $\mathbb{R}^d$
- Continuous and bounded function $f : \mathbb{R}^d \rightarrow \mathbb{R}$.

Objective:
- Solve
  \[
  (P) := \sup_{\nu \in Q} \int f \, d\nu
  \]

Different choices of $Q$ lead to:
- Constrained optimal transport (see e.g. Ekren and Soner (2018), includes martingale optimal transport)
- Distributionally robust optimization (see e.g. Esfahani and Kuhn (2015), Blanchet and Murthy (2016), Bartl et al. (2017), Obłój and Wiesel (2018))
- Mixtures of the above (Gao and Kleywegt (2017))
Example: Martingale optimal transport (MOT)
Stock $S_t$ for times $t = 1, 2$ has known marginals $\mu_1, \mu_2 \in \mathcal{P}(\mathbb{R})$.

$Q = \{ \nu \in \mathcal{P}(\mathbb{R}^2) : \nu_1 = \mu_1, \nu_2 = \mu_2, \text{ if } (S_1, S_2) \sim \nu, \text{ then } \mathbb{E}[S_2 | S_1] = S_1 \}$

The function $f$ can model an exotic option, e.g. $f(s_1, s_2) = (s_2 - Ks_1)^+$.

$Q$ are all potential martingale couplings
Numerical approach: Discretization

**Idea:** Reduce problem $(P)$ to a finite dimensional problem $(P_n)$ by going over to a discrete space.

**In the prior example for instance:**

- Approximate marginals

  $$
  \mu_1 \approx \mu_1^n = \sum_{i=1}^{n} \alpha_i \delta_{x_i}, \quad \mu_2 \approx \mu_2^n = \sum_{i=1}^{n} \beta_i \delta_{y_i}
  $$

- Replace $Q$ by

  $$
  Q_n = \{ \nu \in \mathcal{P}(\mathbb{R}^2) : \nu_1 = \mu_1^n, \nu_2 = \mu_2^n, \\
  \text{if } (S_1, S_2) \sim \nu, \text{ then } \mathbb{E}[S_2|S_1] = S_1 \}
  $$

- Solve $(P_n) = \inf_{\nu \in Q_n} \int f \, d\nu$ instead of $(P)$. 

Stephan Eckstein
Neural Network Hedging
September 5, 2018 4 / 21
Numerical approach: Discretization

In the prior example for instance:

Measures in $Q_n$ only put mass on intersections of grid
Numerical approach: Discretization

Discretization of \((P)\), as well as solving the discrete versions of \((P)\) are studied and applied successfully in a lot of works:

- OT: See e.g. Peyre and Cuturi (2018)
- Distributionally robust optimization: Esfahani and Kuhn (2015)

**Difficulty:** Discretization scales badly with dimension.

- In the prior example, \((P_n)\) has \(n^2\) parameters.
- In a MOT problem with \(T\) time steps, and \(K\) dimensional assets, \((P_n)\) has \(n^{T\cdot K}\) parameters!
Numerical approach: Discretization

Discretization of \((P)\), as well as solving the discrete versions of \((P)\) are studied and applied successfully in a lot of works:

- OT: See e.g. Peyre and Cuturi (2018)
- Distributionally robust optimization: Esfahani and Kuhn (2015)

**Difficulty:** Discretization scales badly with dimension.

- In the prior example, \((P_n)\) has \(n^2\) parameters.
- In a MOT problem with \(T\) time steps, and \(K\) dimensional assets, \((P_n)\) has \(n^{T\cdot K}\) parameters!
Alternative: Parametrization

**Idea:** Work with \( \{ \nu_\lambda : \lambda \in \Lambda \} \subset Q \) where \( \Lambda \) is a finite-dimensional parameter space that one can scale independently of dimension.

**Problem:** No *expressive* sets \( \{ \nu_\lambda : \lambda \in \Lambda \} \) available that are also *numerically feasible* to work with.
Using the dual formulation

We consider problems \((P) = \sup_{\nu \in Q} \int f d\nu\) which allow for a dual formulation of the form

\[
(D) = \inf_{h \in H: \ h \geq f} \varphi(h)
\]

where \(H \subset C(\mathbb{R}^d)\) and \(\varphi : \mathcal{H} \rightarrow \mathbb{R}\) is a linear functional.

**Example:** For the MOT problem from before

- \(\mathcal{H} = \{h(x, y) = h_1(x) + h_2(y) + h_3(x) \cdot (y - x): h_i \in C_b(\mathbb{R})\}\)
- \(\varphi(h) = \int_{\mathbb{R}} h_1 d\mu_1 + \int_{\mathbb{R}} h_2 d\mu_2\)

→ Parametrizing \(\mathcal{H}\) is simpler than parametrizing \(Q\) ! (See also Henry-Labordère (2013))
Using the dual formulation

We consider problems \((P) = \sup_{\nu \in Q} \int f d\nu\) which allow for a dual formulation of the form

\[
(D) = \inf_{\substack{h \in H: \\ h \geq f}} \varphi(h)
\]

where \(H \subset C(\mathbb{R}^d)\) and \(\varphi : H \rightarrow \mathbb{R}\) is a linear functional.

**Example:** For the MOT problem from before

- \(H = \{h(x, y) = h_1(x) + h_2(y) + h_3(x) \cdot (y - x) : h_i \in C_b(\mathbb{R})\}\)
- \(\varphi(h) = \int_{\mathbb{R}} h_1 d\mu_1 + \int_{\mathbb{R}} h_2 d\mu_2\)

→ Parametrizing \(H\) is simpler than parametrizing \(Q\)! (See also Henry-Labordère (2013))
Why use neural networks?

(Feed-forward) neural networks are parametrized functions that build on concatenation of several layers of simple functions.

\[ \mathbb{R}^k \ni x \mapsto A_l \circ \sigma \circ A_{l-1} \circ \ldots \circ \sigma \circ A_0(x) \]

where \( A_i \) are affine transformations and \( \sigma : \mathbb{R} \to \mathbb{R} \) is a non-linear activation function that is applied element-wise.

- Neural networks can approximate a lot of functions with a high, but numerically feasible amount of parameters.
- Successful in practice, even though theoretical understanding of numerical schemes is lacking.
Solution approach using neural networks

\[(D) = \inf_{h \in \mathcal{H}: h \geq f} \varphi(h)\]

**Step 1:** We replace \(\mathcal{H}\) by a set of neural network functions \(\mathcal{H}^m\). Thereby, trading strategies are then restricted to feed-forward neural networks.

Resulting **finite-dimensional** problem

\[(D_m) = \inf_{h \in \mathcal{H}^m: h \geq f} \varphi(h)\]

**Problem:** Constraint \(h \geq f\) prevents application of numerical schemes based on gradient-descent.
Solution approach using neural networks

\[(D^m) = \inf_{h \in \mathcal{H}^m: h \geq f} \varphi(h)\]

**Step 2:** Penalize the inequality constraint \( h \geq f \).

\[(D^m_\theta) = \inf_{h \in \mathcal{H}^m} \varphi(h) + \int \infty \max\{f - h, 0\} d\theta\]

If \( \theta \) gives positive mass of every open ball, then \((D^m_\theta) = (D^m)\).

**Implementation problem:** The mapping \( x \mapsto \infty \max\{0, x\} \) has no useful gradients.
Solution approach using neural networks

$$(D_{\theta}^m) = \inf_{h \in \mathcal{H}^m} \varphi(h) + \int_{\theta} \infty \max\{f - h, 0\} d\theta$$

**Step 3:** Approximate $x \mapsto \infty \max\{x, 0\}$ by a sequence of differentiable nondecreasing convex functions $(\beta_{\gamma})_{\gamma > 0}$, e.g. $\beta_{\gamma} = \gamma \max\{0, x\}^2$.

$$(D_{\theta, \gamma}^m) = \inf_{h \in \mathcal{H}^m} \varphi(h) + \int \beta_{\gamma}(f - h) d\theta$$
### Solution approach: Overview

<table>
<thead>
<tr>
<th>Label</th>
<th>Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(D)$</td>
<td>$\inf_{h \in \mathcal{H}: \ h \geq f} \varphi(h)$</td>
<td>initial problem</td>
</tr>
<tr>
<td>$(D^m)$</td>
<td>$\inf_{h \in \mathcal{H}^m: \ h \geq f} \varphi(h)$</td>
<td>finite dimensional version of $(D)$</td>
</tr>
<tr>
<td>$(D^m_\theta)$</td>
<td>$\inf_{h \in \mathcal{H}^m} \varphi(h) + \int \infty(f - h)^+ d\theta$</td>
<td>dominated version of $(D^m)$</td>
</tr>
<tr>
<td>$(D^m_{\theta, \gamma})$</td>
<td>$\inf_{h \in \mathcal{H}^m} \varphi(h) + \int \beta \gamma(f - h) d\theta$</td>
<td>penalized version of $(D^m_\theta)$</td>
</tr>
</tbody>
</table>

**Table:** Summary of problems occurring in the approach.
Solution approach: Theoretical results

Under mild assumptions on $\mathcal{H}$, $\mathcal{H}^m$ and $\theta$ it holds

$$\left(D^m\right) \rightarrow (D) \text{ for } m \rightarrow \infty$$ \hspace{1cm} (1)

$$\left(D^m_{\theta,\gamma}\right) \rightarrow (D^m) \text{ for } \gamma \rightarrow \infty$$ \hspace{1cm} (2)

Related results

- To (1): E.g. Bühler et al. (2018),
- To (2): E.g. Cominetti and San Martín (1994), Cuturi (2013), many others related to *Sinkhorn distance*, *Bregman projection* and *Schrödinger problem*.
Solution approach: Theoretical results

Under mild assumptions on $\mathcal{H}$, $\mathcal{H}^m$ and $\theta$ it holds

$$(D^m) \to (D) \text{ for } m \to \infty \quad (1)$$

$$(D^m_{\theta, \gamma}) \to (D^m) \text{ for } \gamma \to \infty \quad (2)$$

Related results

- To (1): E.g. Bühler et al. (2018),
Numerics: MOT example

**MOT as in introduction**

\[
f(s_1, s_2) = (s_2 - s_1)^+\]

Marginals \(\mu_1, \mu_2\): Some mixtures of normals

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Neural network</th>
<th>Discretization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>NN with 4 layers &amp; hidden dimension 64,</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(\theta = \mu_1 \otimes \mu_2,)</td>
<td>(n = 600)</td>
</tr>
<tr>
<td></td>
<td>(\beta_{\gamma}(x) = 1000 \max{0, x}^2)</td>
<td>(Relaxed martingale constraint: (\varepsilon = 10^{-6}))</td>
</tr>
<tr>
<td>Optimizer</td>
<td>Approximate dual optimizer (\hat{h})</td>
<td>Approximate coupling (\hat{\nu})</td>
</tr>
<tr>
<td>Superhedging</td>
<td>0.2956</td>
<td>0.2990</td>
</tr>
<tr>
<td>price</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subhedging</td>
<td>0.0889</td>
<td>0.0844</td>
</tr>
<tr>
<td>price (inf over</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Q) instead)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Stephan Eckstein
Neural Network Hedging
September 5, 2018 15 / 21
MOT: Dual optimizer

Superhedging
MOT: Dual optimizer

Subhedging
Back to the primal

**Goal:** Use the neural network solution of the dual to obtain a (near) optimal measure of the primal!

The problem

\[(D_{\theta, \gamma}) = \inf_{h \in \mathcal{H}} \varphi(h) + \int \beta_\gamma(f - h) d\theta\]

has a primal formulation

\[(P_{\theta, \gamma}) = \sup_{\nu \in \mathcal{Q}} \int f d\nu - \int \beta^*_\gamma \left( \frac{d\nu}{d\theta} \right) d\theta\]

and if \((D_{\theta, \gamma})\) has an optimizer \(\hat{h}\), one can often show that \(\hat{\nu}\) given by

\[
\frac{d\hat{\nu}}{d\theta} = \beta'_\gamma(f - \hat{h})
\]

is an optimizer of \((P_{\theta, \gamma})\).
Back to the primal

**Goal:** Use the neural network solution of the dual to obtain a (near) optimal measure of the primal!

The problem

\[
(D_{\theta,\gamma}) = \inf_{h \in \mathcal{H}} \varphi(h) + \int \beta_{\gamma}(f - h) d\theta
\]

has a primal formulation

\[
(P_{\theta,\gamma}) = \sup_{\nu \in Q} \int fd\nu - \int \beta^*_{\gamma} \left( \frac{d\nu}{d\theta} \right) d\theta
\]

and if \((D_{\theta,\gamma})\) has an optimizer \(\hat{h}\), one can often show that \(\hat{\nu}\) given by

\[
\frac{d\hat{\nu}}{d\theta} = \beta'_{\gamma}(f - \hat{h})
\]

is an optimizer of \((P_{\theta,\gamma})\).
Back to the primal

**Goal:** Use the neural network solution of the dual to obtain a (near) optimal measure of the primal!

The problem

\[
(D_{\theta, \gamma}) = \inf_{h \in \mathcal{H}} \varphi(h) + \int \beta_{\gamma}(f - h) d\theta
\]

has a primal formulation

\[
(P_{\theta, \gamma}) = \sup_{\nu \in \mathcal{Q}} \int fd\nu - \int \beta^*_\gamma \left( \frac{d\nu}{d\theta} \right) d\theta
\]

and if \((D_{\theta, \gamma})\) has an optimizer \(\hat{h}\), one can often show that \(\hat{\nu}\) given by

\[
\frac{d\hat{\nu}}{d\theta} = \beta'_\gamma(f - \hat{h})
\]

is an optimizer of \((P_{\theta, \gamma})\).
MOT: Primal optimizer

Approximately optimal coupling: Superhedging
MOT: Primal optimizer

Approximately optimal coupling: Subhedging
MOT: Primal optimizer

Approximately optimal coupling: Subhedging
Bank is exposed to 6 types of risks $X_1, ..., X_6$ with known marginal exposures.

An expert opinion $\bar{\mu} \in \mathcal{P}(\mathbb{R}^6)$ about the joint distribution of $(X_1, ..., X_6)$ is given.

Goal: Calculate bounds on

$$AVaR_\alpha \left( \sum_{i=1}^{6} X_i \right)$$

under constraints that

- $(X_1, ..., X_6)$ have marginals $\mu_1, ..., \mu_6$
- Joint distribution $\nu \sim (X_1, ..., X_6)$ is in a Wasserstein ball around $\bar{\mu}$ of a given radius $\rho$: $W_p(\bar{\mu}, \nu) \leq \rho$
Risk aggregation (DNB Case Study - Aas and Puccetti (2014))

- Bank is exposed to 6 types of risks $X_1, ..., X_6$ with known marginal exposures.
- An expert opinion $\bar{\mu} \in \mathcal{P}(\mathbb{R}^6)$ about the joint distribution of $(X_1, ..., X_6)$ is given.

**Goal:** Calculate bounds on

$$AVaR_{\alpha} \left( \sum_{i=1}^{6} X_i \right)$$

under constraints that

- $(X_1, ..., X_6)$ have marginals $\mu_1, ..., \mu_6$
- Joint distribution $\nu \sim (X_1, ..., X_6)$ is in a Wasserstein ball around $\bar{\mu}$ of a given radius $\rho$: $W_p(\bar{\mu}, \nu) \leq \rho$
Bank risk aggregation (DNB Case Study)

Using only marginal information:

Average Value at Risk for $\alpha = 0.95$

Reference model $\bar{\mu}$

24165 (Best Case) 30506 36410 (Worst Case)
Bank risk aggregation (DNB Case Study)

Worst case around reference model:

Average Value at Risk for $\alpha = 0.95$

Worst case over Wasserstein ball around $\tilde{\mu}$ of radius $\rho$

Reference model $\tilde{\mu}$

24165 (Best Case) -> 30506 -> 36410 (Worst Case)

($\rho = 1/6$) 33175

($\rho = 1$) 36142

Stephan Eckstein
Neural Network Hedging
September 5, 2018 21 / 21
Thank you
References

Aas, Kjersti and Puccetti, Giovanni
Bounds on total economic capital: the DNB case study

Alfonsi, Aurélien and Corbetta, Jacopo and Jourdain, Benjamin
Sampling of probability measures in the convex order and approximation of Martingale Optimal Transport problems

Bartl, Daniel and Drapeau, Samuel and Tangpi, Ludovic
Computational aspects of robust optimized certainty equivalents

Blanchet, Jose and Murthy, Karthyek RA
Quantifying distributional model risk via optimal transport

Deep hedging.
R. Cominetti and J. San Martín.
Asymptotic analysis of the exponential penalty trajectory in linear programming.

M. Cuturi.
Sinkhorn distances: Lightspeed computation of optimal transport.

Eckstein, Stephan and Kupper, Michael
Computation of optimal transport and related hedging problems via penalization and neural networks.

I. Ekren and H. M. Soner.
Constrained optimal transport.

Esfahani, Peyman Mohajerin and Kuhn, Daniel
Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations
R. Gao and A. Kleywegt
Data-Driven Robust Optimization with Known Marginal Distributions
*Working paper, 2017*

Henry-Labordère, Pierre
Automated option pricing: Numerical methods

Guo, Gaoyue and Obloj, Jan
Computational Methods for Martingale Optimal Transport problems

Obloj, Jan and Wiesel, Johannes
Statistical estimation of superhedging prices

Peyré, Gabriel and Cuturi, Marco
Computational optimal transport