Design with Formal Risk Guarantees via the Scenario Approach: A Sample Compression Framework

speaker: **Simone Garatti**

(Politecnico di Milano, Italy – email: simone.garatti@polimi.it)

Many thanks to all collaborators!

Marco C. Campi

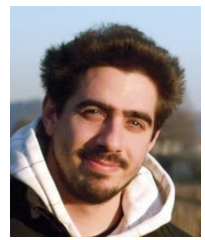
Many thanks to all collaborators!


Marco C. Campi


Algo Carè


Federico Ramponi


Maria Prandini


Alessandro Falsone


Lucrezia Manieri

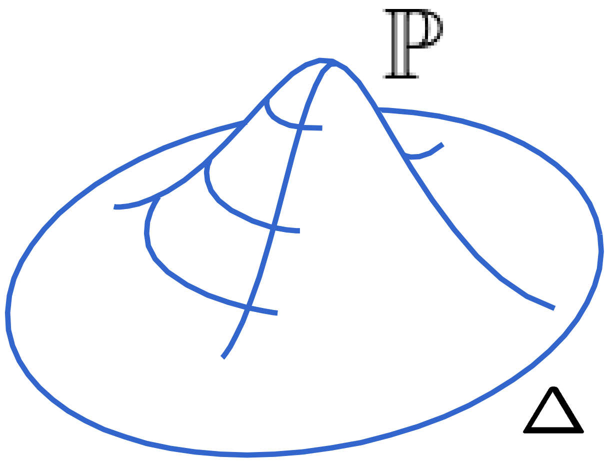

Dario Paccagnan


Kostas Margellos


Alex Gallo

Data-driven decision-making

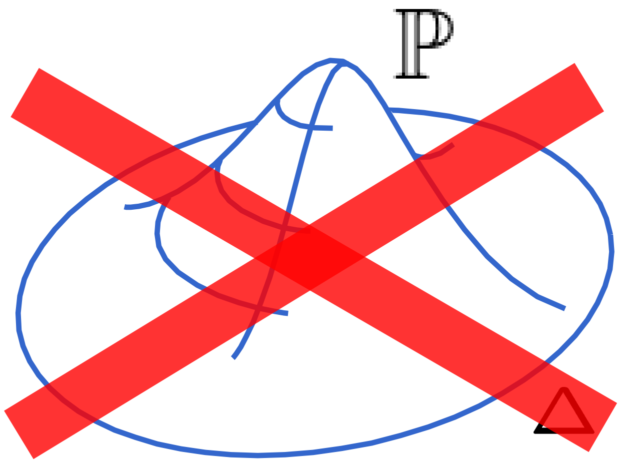
δ = uncertain element ⇒ exercise caution



decision $\mathcal{H}$

Data-driven decision-making

δ = uncertain element ⇨ exercise caution

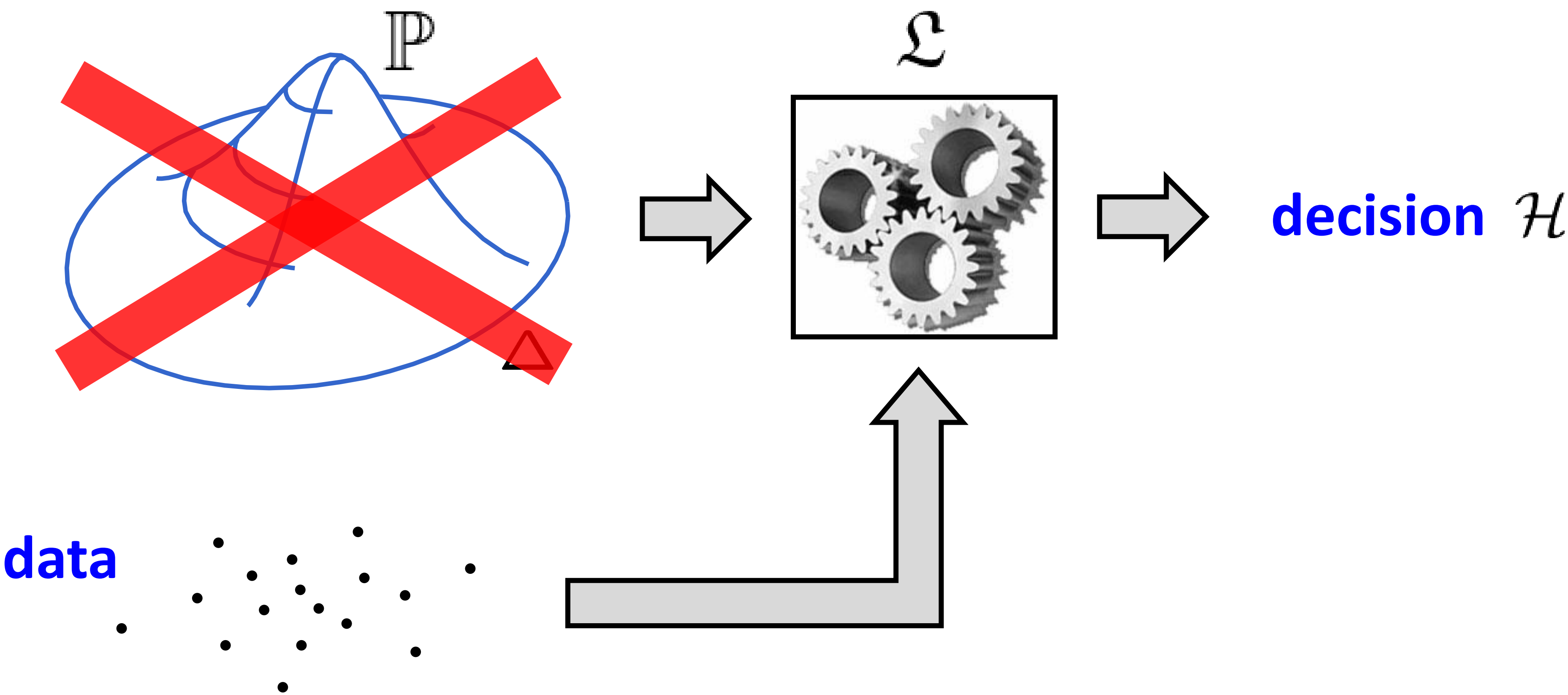


⇨

⇨ **decision** $\mathcal{H}$

Data-driven decision-making

δ = uncertain element ⇒ exercise caution



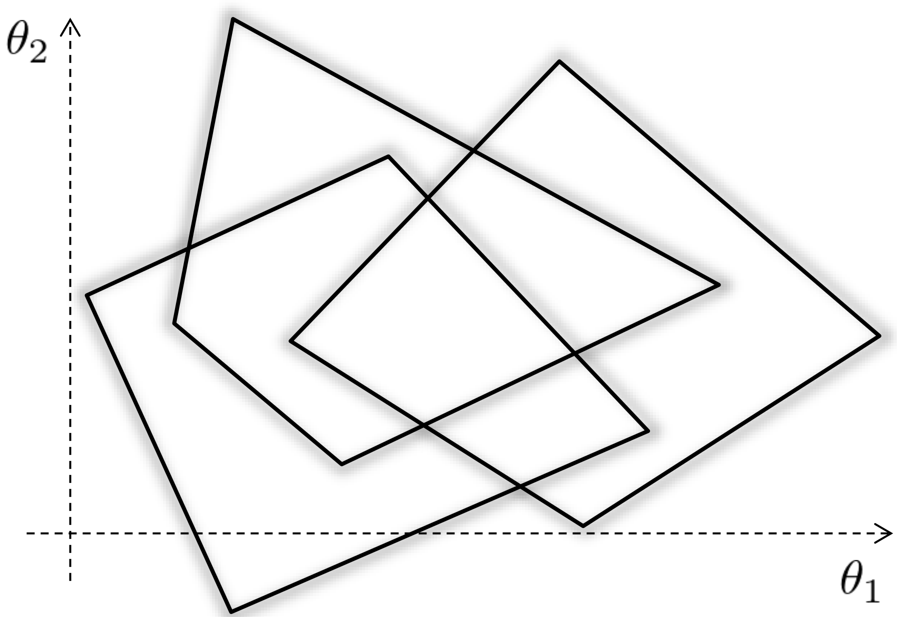
$\delta_1, \delta_2, \ldots, \delta_N$ (scenarios, i.i.d. sample)

Example: data-driven (scenario) robust optimization

data

$$\delta_i \rightarrow f(\theta, \delta_i) \leq 0$$

constraint



Example: data-driven (scenario) robust optimization

data

$\delta_i \to f(\theta, \delta_i) \leq 0$

constraint

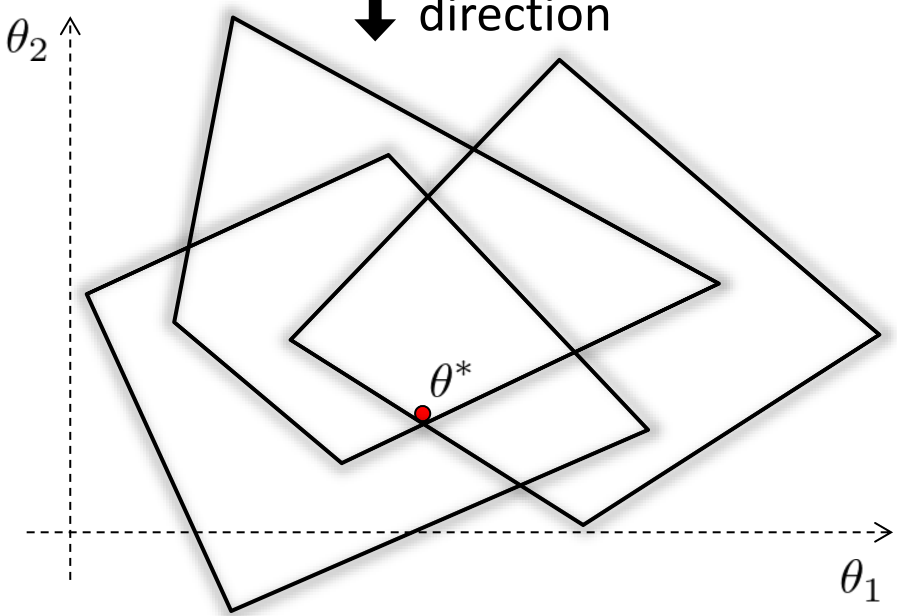
$\mathcal{H}$ = θ^* = solution to

$$\min_{\theta \in \Theta} \quad c(\theta)$$

$$\text{s.t.} \quad f(\theta, \delta_i) \leq 0$$

$$i = 1, \ldots, N$$

optimization direction



data-driven
robust H_2 control

Example: optimization with constraints relaxations

data

$$\delta_i \rightarrow f(\theta, \delta_i) \leq 0$$

constraint

$\mathcal{H}$ = θ^* = solution to

$$\min_{\theta \in \Theta, \xi_i \geq 0} \quad c(\theta) + \rho \sum_{i=1}^{N} \xi_i$$

$$\text{s.t.} \quad f(\theta, \delta_i) \leq \xi_i$$

$$i = 1, \ldots, N$$

Example: optimization with constraints relaxations

data

$\delta_i \rightarrow f(\theta, \delta_i) \leq 0$

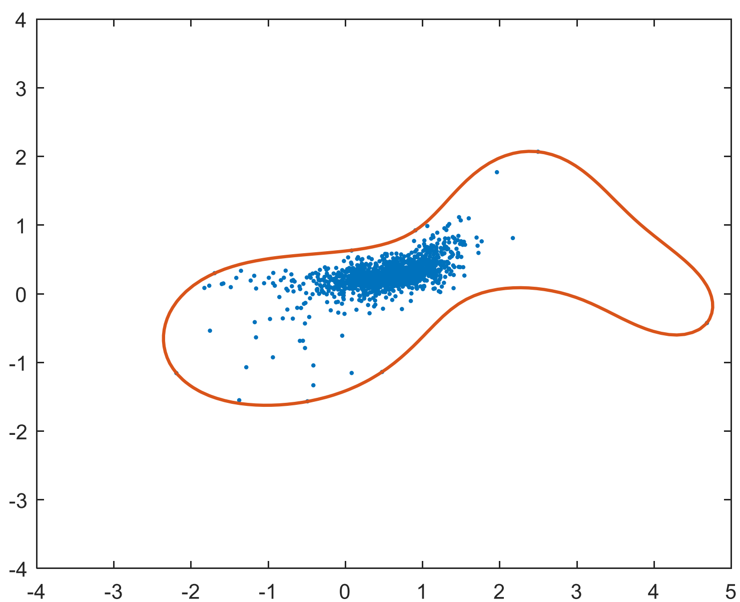
constraint

$\mathcal{H}$ = θ^* = solution to

$$\min_{\theta \in \Theta, \xi_i \geq 0} \quad c(\theta) + \rho \sum_{i=1}^{N} \xi_i$$

$$\text{s.t.} \quad f(\theta, \delta_i) \leq \xi_i$$

$$i = 1, \ldots, N$$



Support Vector methods
for reachability analysis

Example: classification

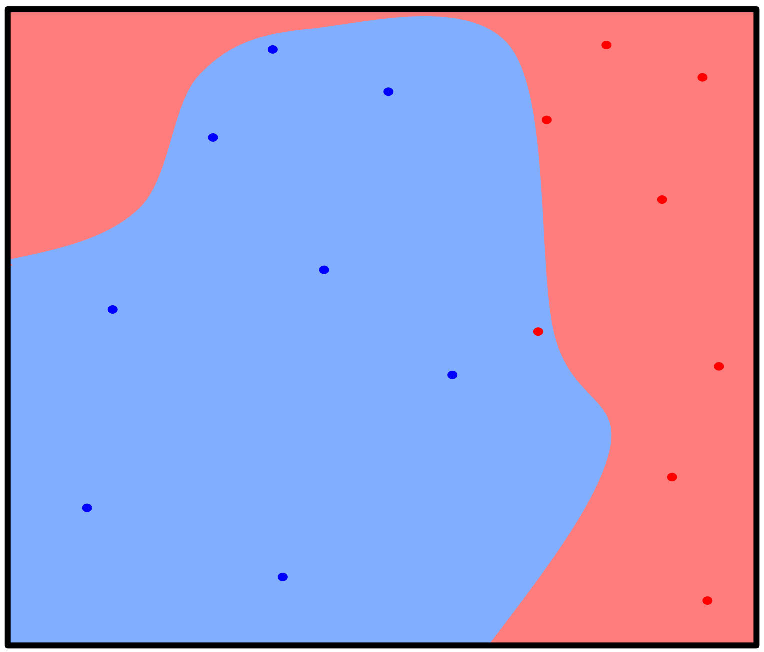
data

$\delta_i = (u_i, y_i)$

$u_i \in \mathbb{R}^d$

$y_i \in \{\text{red}, \text{blue}\}$

$\mathcal{H}$ = Neural Network classifier trained via an SGD-based training algorithm



A lesson from machine learning

Which is the data-driven decision scheme for the problem at hand?

Difficult to say a-priori without incurring in over-conservatism … a blend of approximate knowledge and heuristics, often in various attempts (hyperparameters tuning)

No limits in exploration, but some guidance is needed…

A lesson from machine learning

Which is the data-driven decision scheme for the problem at hand?

Difficult to say a-priori without incurring in over-conservatism … a blend of approximate knowledge and heuristics, often in various attempts (hyperparameters tuning)

No limits in exploration, but some guidance is needed…

⇒ **SCENARIO APPROACH**: a tool to provide accurate and rigorous certification of the actual performance of the explored decisions

→ final decision selection

→ dependable utilization of it

… when is it possible?

The risk of a decision

$\mathcal{H}$ is inappropriate for a new δ

↓

interaction between decision and environment

E.g.,

a new constraint is violated by $\mathcal{H}$

a new terminal state is outside $\mathcal{H}$

a new I/O pair is misclassified by $\mathcal{H}$

The risk of a decision

$$\mathrm{R}(\mathcal{H}) = \mathbb{P}\left\{ \mathcal{H} \text{ is inappropriate for a new } \delta \right\}$$

Risk = out-of-sample probability of inappropriateness

E.g., $\mathbb{P}\{\text{a new constraint is violated by } \mathcal{H}\}$

$\mathbb{P}\{\text{ a new terminal state is outside } \mathcal{H}\}$

$\mathbb{P}\{\text{a new I/O pair is misclassified by } \mathcal{H}\}$

The risk of a decision

$R(\mathcal{H}) = \mathbb{P}$? $\{\mathcal{H}$ is inappropriate for a new $\delta\}$

Risk = out-of-sample probability of inappropriateness

E.g., $\mathbb{P}$? $\{$a new constraint is violated by $\mathcal{H}\}$

$\mathbb{P}$? $\{$ a new terminal state is outside $\mathcal{H}\}$

$\mathbb{P}$? $\{$a new I/O pair is misclassified by $\mathcal{H}\}$

Issue: $\mathbb{P}$ is **not** available…

Main goal

⇨ **assess $\mathrm{R}(\mathcal{H})$ from data, the same used for design**

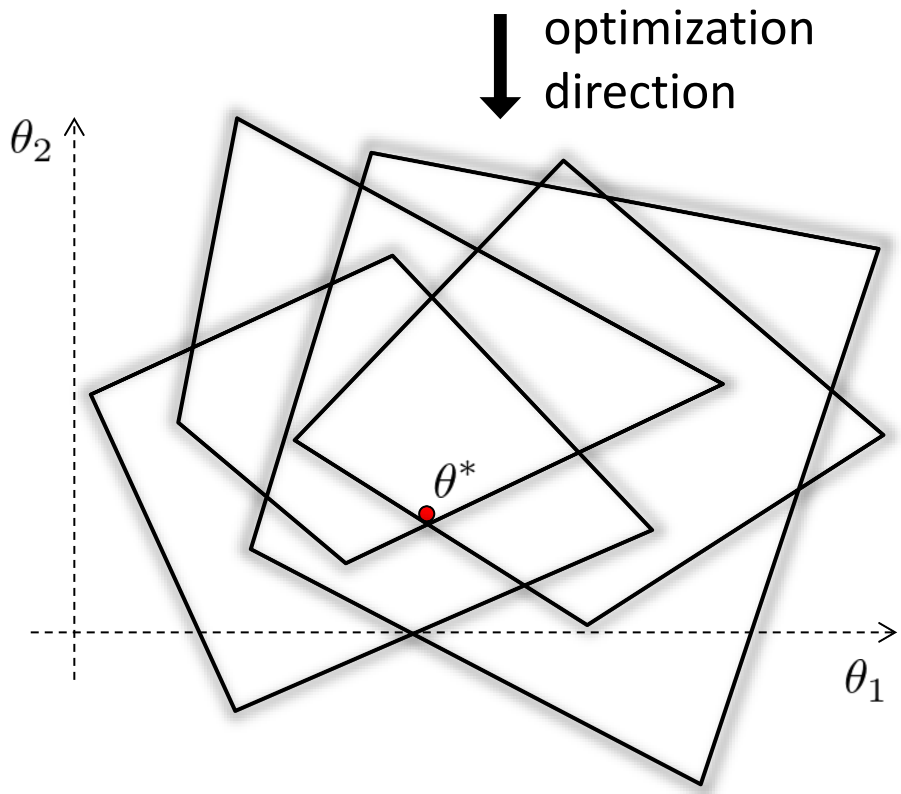
Main goal

➡ **assess $\mathrm{R}(\mathcal{H})$ from data, the same used for design**

Why not using new data for validation:

- using some data for testing rather than designing is a waste of information!
- scenarios (data) are often limited resources (collecting data can be time-consuming or burdensome, involving a monetary cost)
- in many contexts validation is not necessary… **data can play well a double role!**

Risk assessment via sample compression



Compression function

$$\kappa(\delta_1, \ldots, \delta_N) = \delta_{i_1}, \ldots, \delta_{i_k}$$

map extracting a subsample
from a sample of scenarios

Preference

$$\kappa(\delta_1, \ldots, \delta_N) \subseteq S \subseteq (\delta_1, \ldots, \delta_N)$$

$$\Downarrow$$

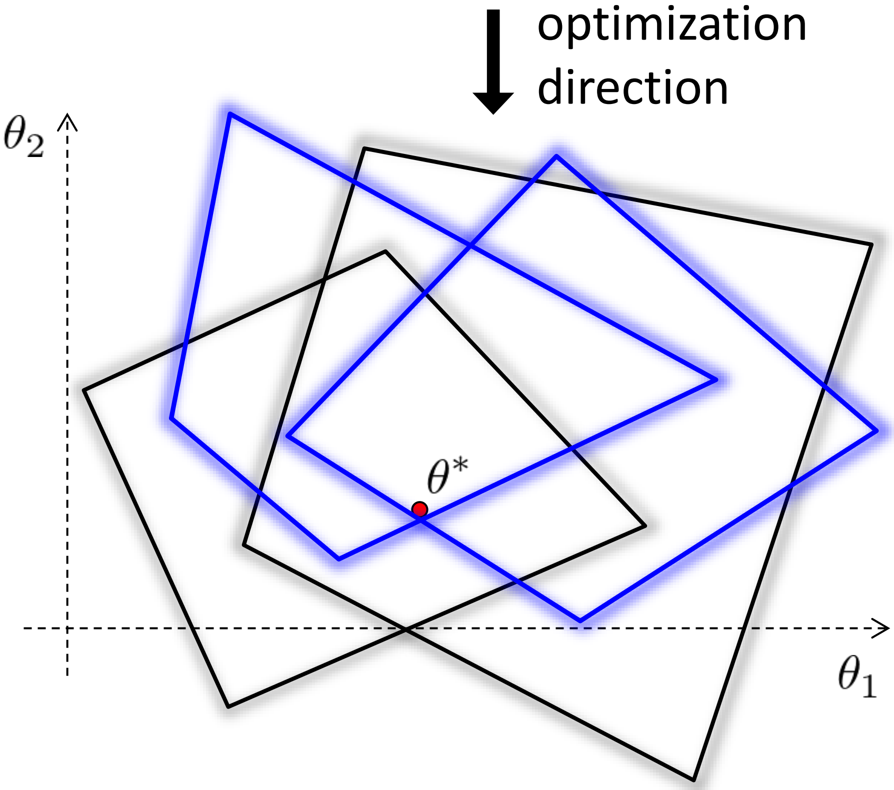
$$\kappa(S) = \kappa(\delta_1, \ldots, \delta_N)$$

Risk assessment via sample compression

Compression function

$$\kappa(\delta_1, \ldots, \delta_N) = \delta_{i_1}, \ldots, \delta_{i_k}$$

map extracting a subsample from a sample of scenarios



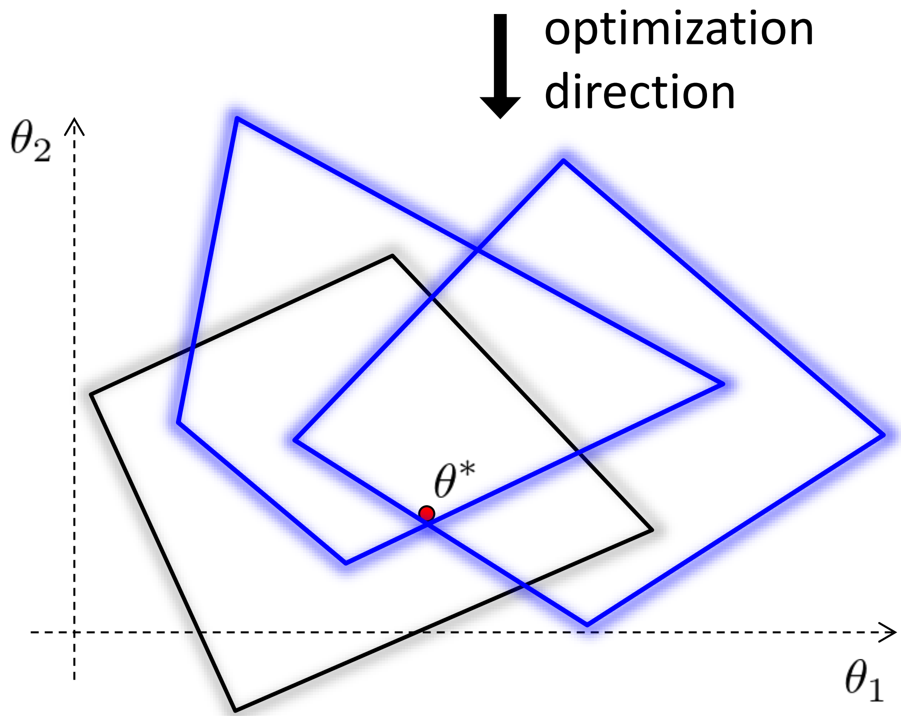
Preference

$$\kappa(\delta_1, \ldots, \delta_N) \subseteq S \subseteq (\delta_1, \ldots, \delta_N)$$

$$\Downarrow$$

$$\kappa(S) = \kappa(\delta_1, \ldots, \delta_N)$$

Risk assessment via sample compression



Compression function

$$\kappa(\delta_1, \ldots, \delta_N) = \delta_{i_1}, \ldots, \delta_{i_k}$$

map extracting a subsample from a sample of scenarios

Preference

$$\kappa(\delta_1, \ldots, \delta_N) \subseteq S \subseteq (\delta_1, \ldots, \delta_N)$$

$$\Downarrow$$

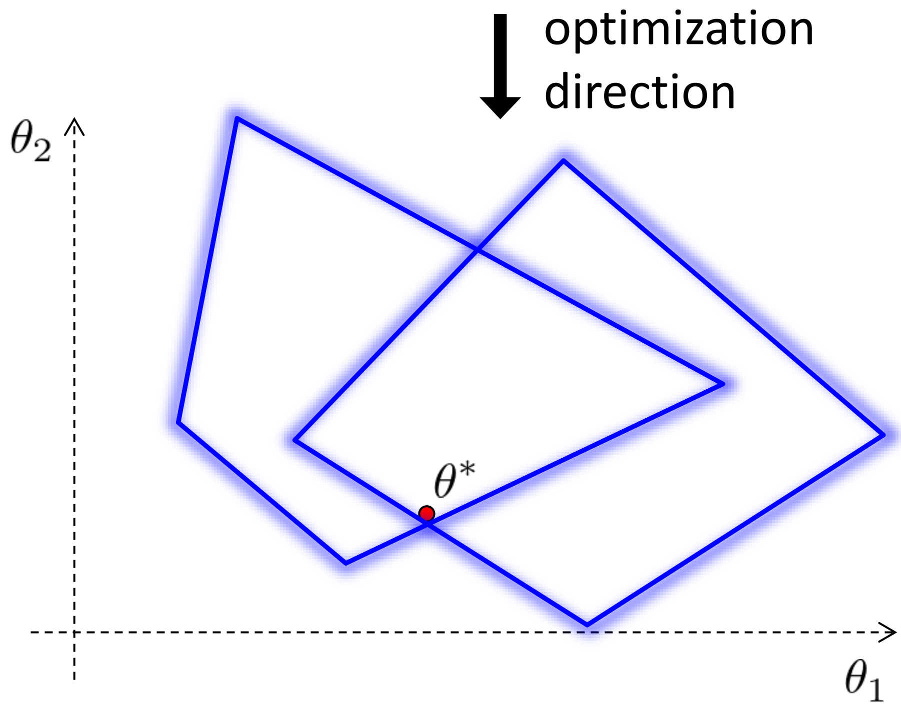
$$\kappa(S) = \kappa(\delta_1, \ldots, \delta_N)$$

Risk assessment via sample compression

Compression function

$$\kappa(\delta_1, \ldots, \delta_N) = \delta_{i_1}, \ldots, \delta_{i_k}$$

map extracting a subsample from a sample of scenarios



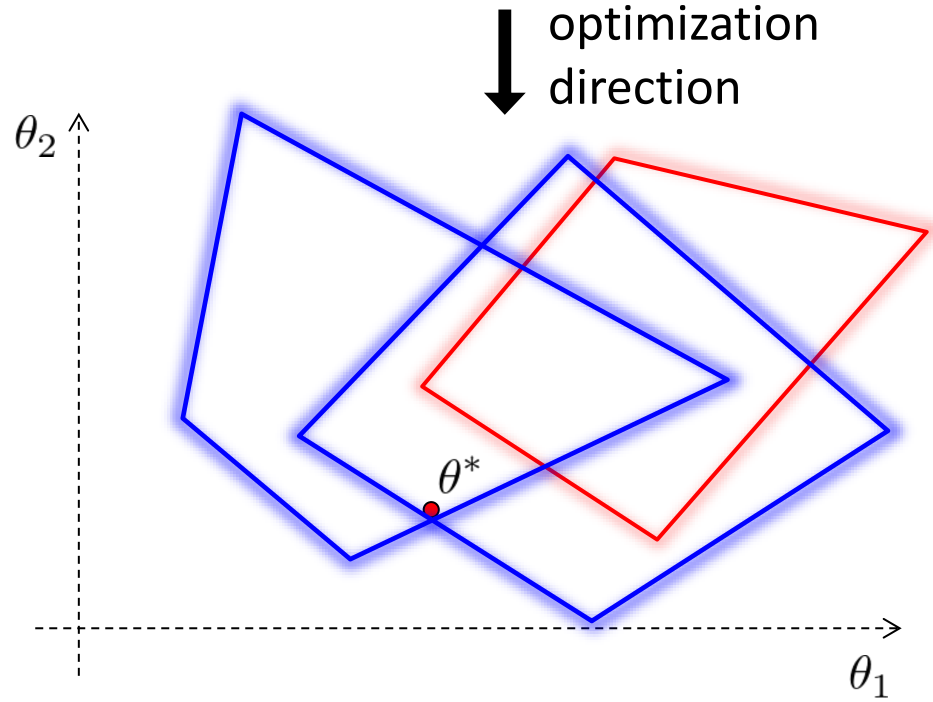
Coherence

a new scenario for which $\mathcal{H}$ is inappropriate is added

$\Downarrow$

the compression must change

Risk assessment via sample compression



Compression function

$$\kappa(\delta_1, \ldots, \delta_N) = \delta_{i_1}, \ldots, \delta_{i_k}$$

map extracting a subsample from a sample of scenarios

Coherence

a new scenario for which $\mathcal{H}$ is inappropriate is added

$\Downarrow$

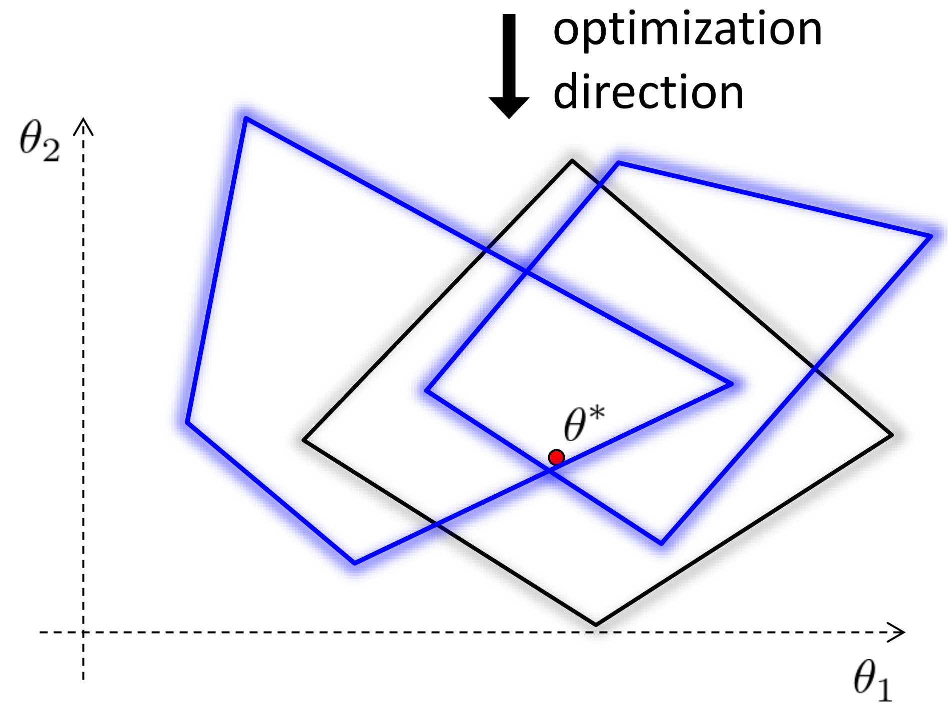
the compression must change

Risk assessment via sample compression

Compression function

$$\kappa(\delta_1, \ldots, \delta_N) = \delta_{i_1}, \ldots, \delta_{i_k}$$

map extracting a subsample from a sample of scenarios



Coherence

a new scenario for which $\mathcal{H}$ is inappropriate is added

$\Downarrow$

the compression must change

The main result in a nutshell

Risk: $\mathsf{R}(\mathcal{H}) = \mathsf{R}\big(\mathcal{H}(\delta_1, \ldots, \delta_N)\big)$

Complexity: $\pi = \big|\kappa(\delta_1, \ldots, \delta_N)\big|$

} random variables

↑ size of compressed set

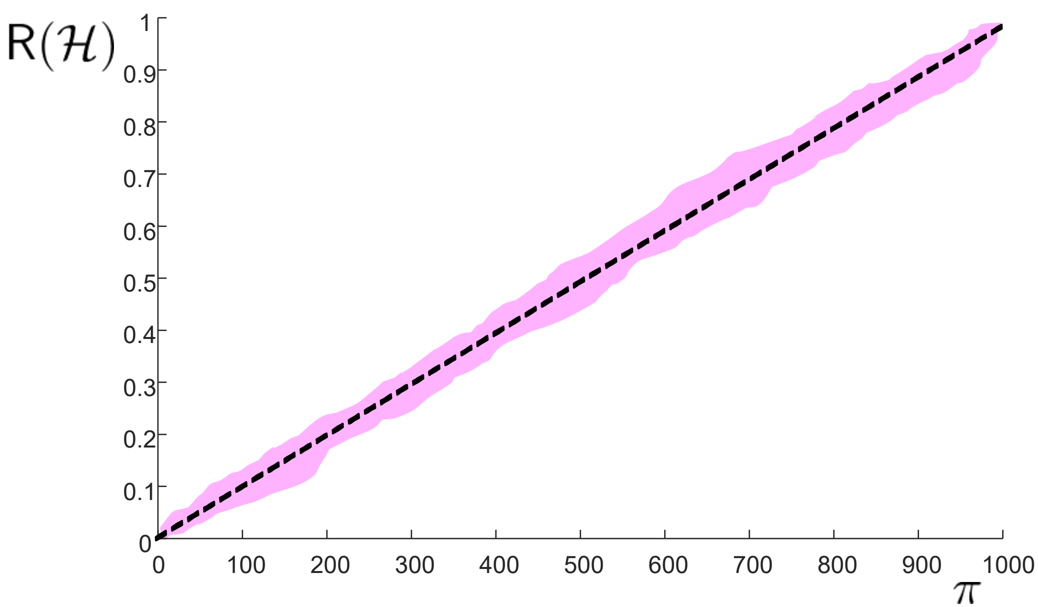
The main result in a nutshell

Risk: $\mathsf{R}(\mathcal{H}) = \mathsf{R}\big(\mathcal{H}(\delta_1, \ldots, \delta_N)\big)$

Complexity: $\pi = \big|\kappa(\delta_1, \ldots, \delta_N)\big|$

} random variables

Under preference and coherence, the joint distribution of risk and complexity is concentrated around/below $\mathsf{R}(\mathcal{H}) = \pi/N$



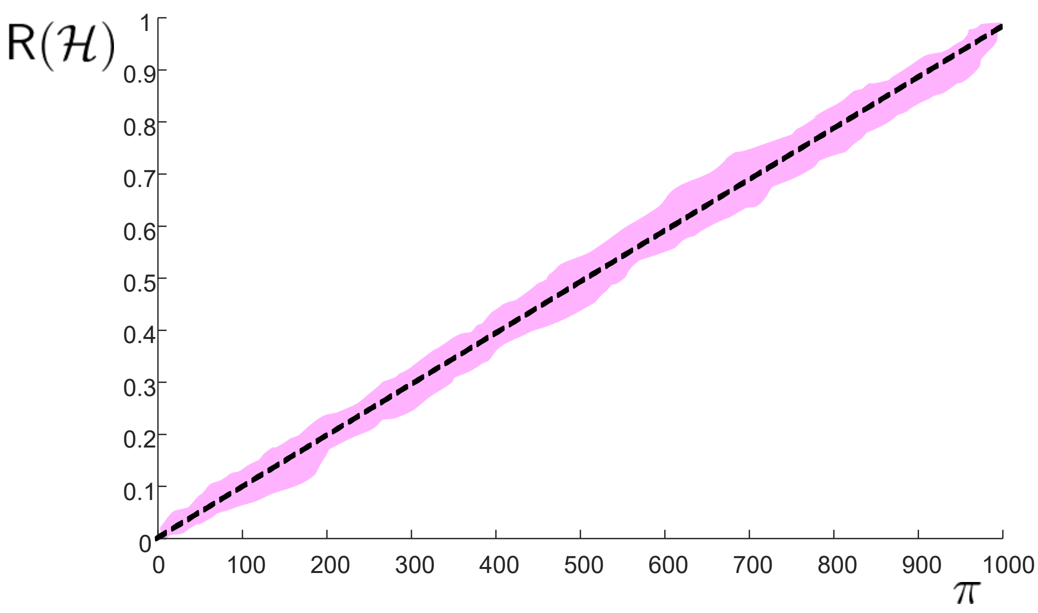
The main result in a nutshell

Risk: $\mathsf{R}(\mathcal{H}) = \mathsf{R}\big(\mathcal{H}(\delta_1, \ldots, \delta_N)\big)$

Complexity: $\pi = \big|\kappa(\delta_1, \ldots, \delta_N)\big|$

} random variables

Under preference and coherence, the joint distribution of risk and complexity is concentrated around/below $\mathsf{R}(\mathcal{H}) = \pi / N$



$\mathsf{R}(\mathcal{H})$ can be accurately estimated from π

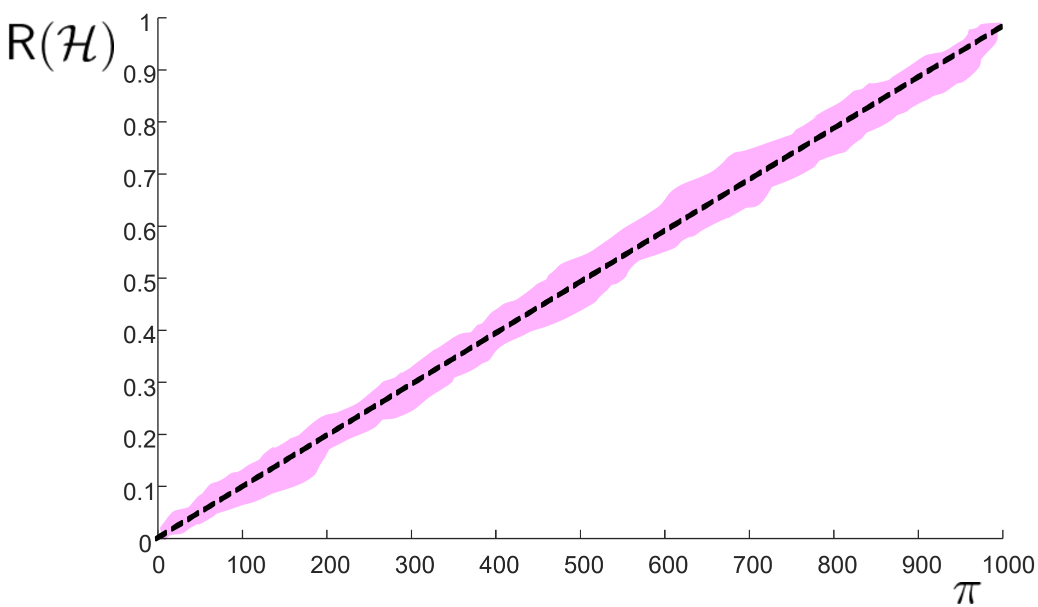
The main result in a nutshell

Risk: $\mathsf{R}(\mathcal{H}) = \mathsf{R}\big(\mathcal{H}(\delta_1, \dots, \delta_N)\big)$

Complexity: $\pi = \big|\kappa(\delta_1, \dots, \delta_N)\big|$

} random variables

Under preference and coherence, the joint distribution of risk and complexity is concentrated around/below $\mathsf{R}(\mathcal{H}) = \pi / N$



$\mathsf{R}(\mathcal{H})$ can be accurately estimated from π

observable!

Main result (cont'd)

Theorem (with M. Campi)

Assume preference and coherence

Choose $\beta \in (0,1)$ (confidence parameter)

Let $\epsilon_L(k), \epsilon^U(k)$ be the unique roots in (0,1) of polynomials

- $$\binom{N}{k}(1-\epsilon)^{N-k} - \frac{\beta}{2N}\sum_{m=k}^{N-1}\binom{m}{k}(1-\epsilon)^{m-k}$$
- $$\binom{N}{k}(1-\epsilon)^{N-k} - \frac{\beta}{2N}\sum_{m=N+1}^{2N}\binom{m}{k}(1-\epsilon)^{m-k}$$

Then, irrespective of $\mathbb{P}$ (distribution-free),

$$\mathbb{P}^N\left\{\delta_1,\ldots,\delta_N : \epsilon^L(\pi) \leq \mathsf{R}(\mathcal{H}) \leq \epsilon^U(\pi)\right\} \geq 1-\beta$$

Main result (cont'd)

true with confidence $1 - \beta$

claim: $\epsilon_L(\pi) \leq \mathsf{R}(\mathcal{H}) \leq \epsilon^U(\pi)$

Main result (cont'd)

true with confidence $1 - \beta$

claim: $\epsilon_L(\pi) \leq \mathsf{R}(\mathcal{H}) \leq \epsilon^U(\pi)$

close each other even with finite N,
gap goes to zero as $1/\sqrt{N}$

Main result (cont'd)

true with confidence $1-\beta$

claim: $\epsilon_L(\pi) \leq \mathbf{R}(\mathcal{H}) \leq \epsilon^U(\pi)$

Complexity is a universal observable to obtain very informative assessments of the actual risk !

with finite N,

$/\sqrt{N}$

$\mathrm{R}(\mathcal{H})$ ⇒ not accessible

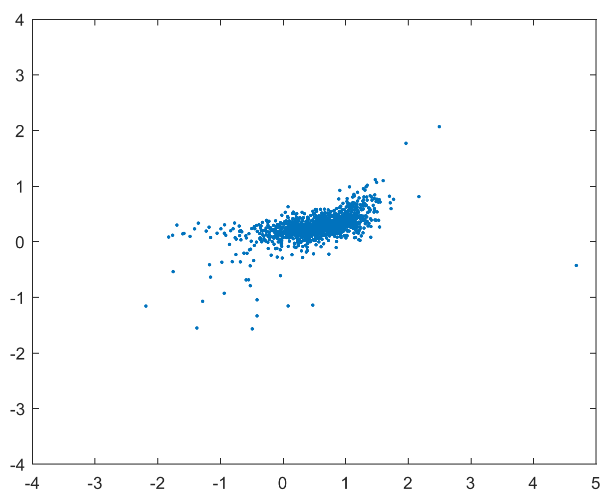
$[\epsilon_L(\pi), \epsilon^U(\pi)]$ ⇒ accessible

- accept/reject the solution
- make further decisions
- compare various "decisions"

Example: reachability analysis via SVDD

$$\begin{cases} x(t+1) = f_\delta\big(x(t), w_\delta(t)\big) \\ x(0) = \bar{x}_\delta \end{cases}$$

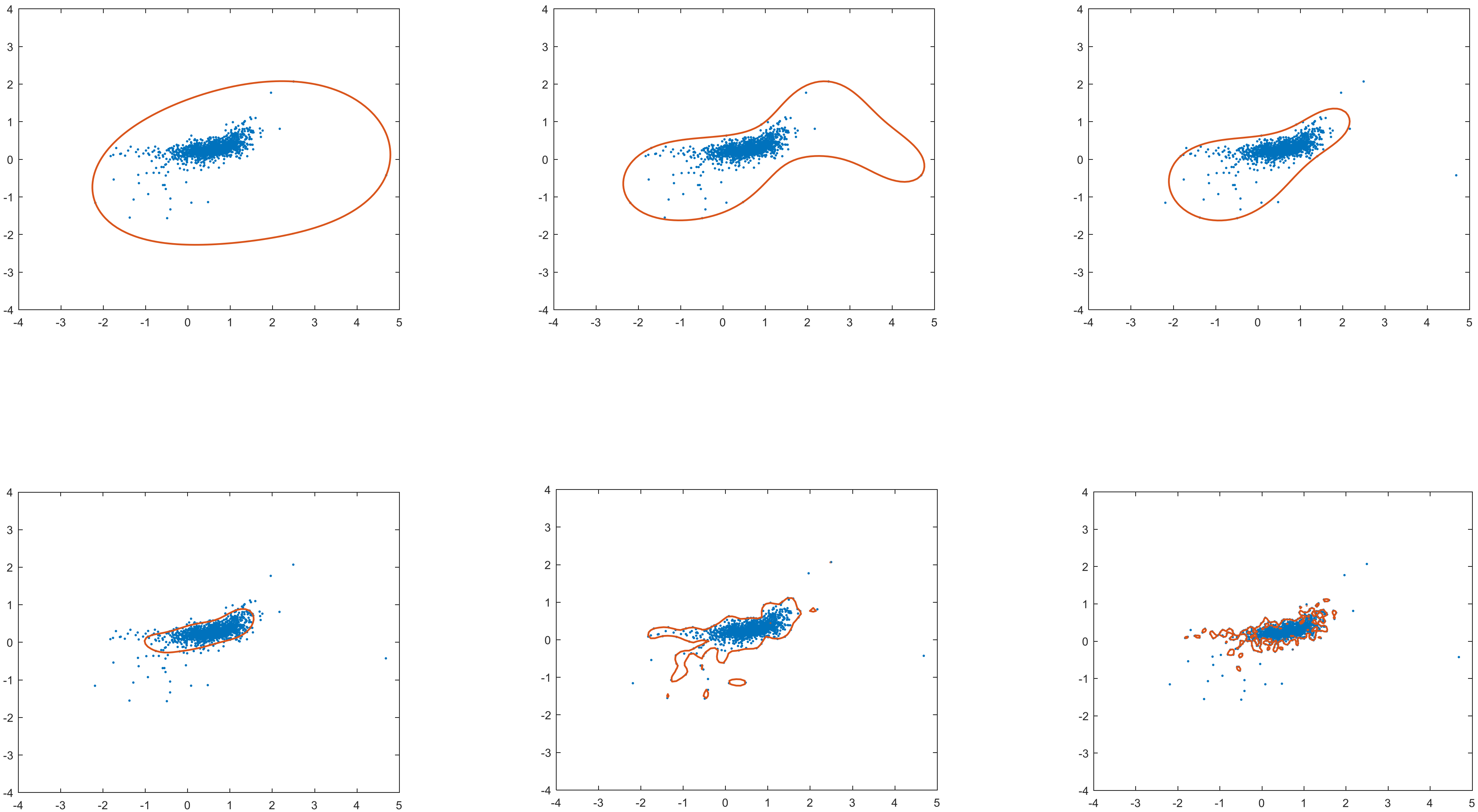
Goal *(Arcak, Devonport, Dietrich, Tu)* : construct a reachable set S such that the terminal state $x(\mathrm{T})$ lies in S with a prescribed probability



SVDD → S

hyperparameters

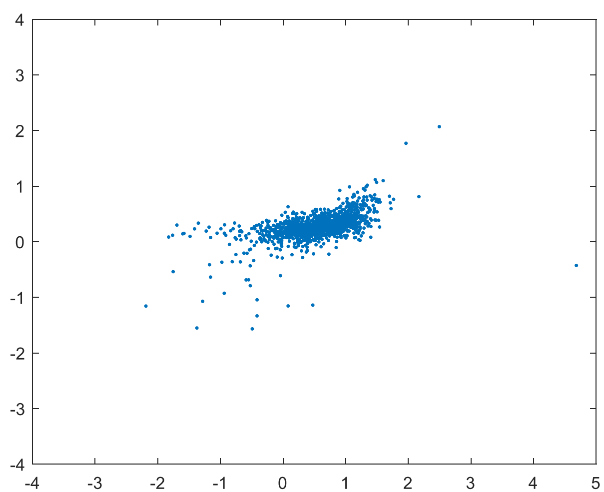
Example: reachability analysis via SVDD



Example: reachability analysis via SVDD

$$\begin{cases} x(t+1) = f_\delta\big(x(t), w_\delta(t)\big) \\ x(0) = \bar{x}_\delta \end{cases}$$

Goal *(Arcak, Devonport, Dietrich, Tu)* : construct a reachable set S such that the terminal state $x(T)$ lies in S with a prescribed probability



SVDD

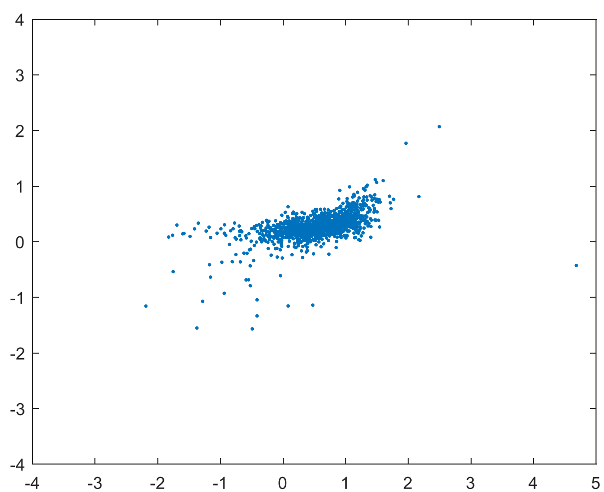
hyperparameters

S

Example: reachability analysis via SVDD

$$\begin{cases} x(t+1) = f_\delta\big(x(t), w_\delta(t)\big) \\ x(0) = \bar{x}_\delta \end{cases}$$

Goal *(Arcak, Devonport, Dietrich, Tu)* : construct a reachable set S such that the terminal state $\underbrace{x(T) \text{ lies in } S}_{\text{appropriateness}}$ with a prescribed probability

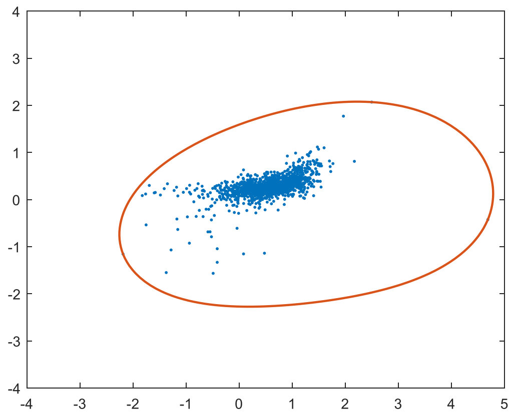


Compression = Support Vectors

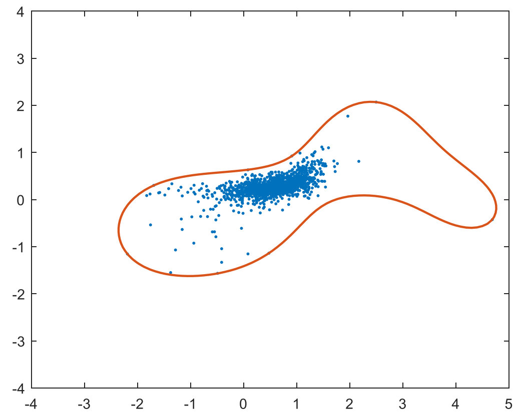
SVDD → S

hyperparameters

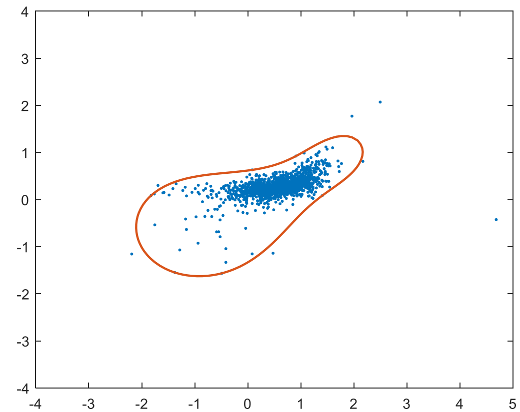
Example: reachability analysis via SVDD



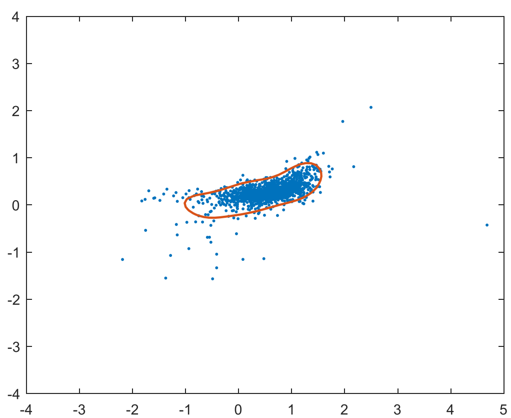
$0\% \leq \mathsf{R}(\mathcal{H}) \leq 1.9\%$



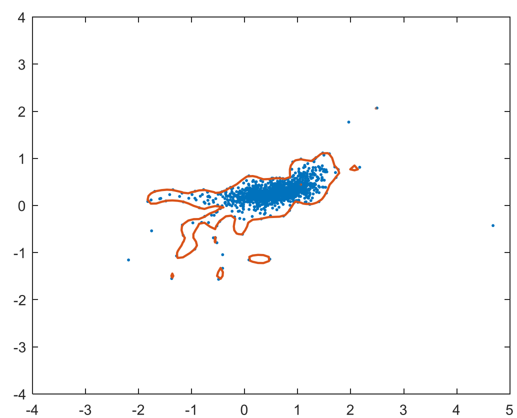
$0\% \leq \mathsf{R}(\mathcal{H}) \leq 1.9\%$



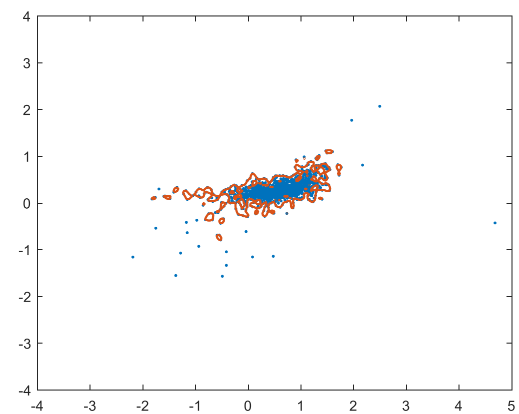
$0\% \leq \mathsf{R}(\mathcal{H}) \leq 2.8\%$



$4.3\% \leq \mathsf{R}(\mathcal{H}) \leq 11.2\%$

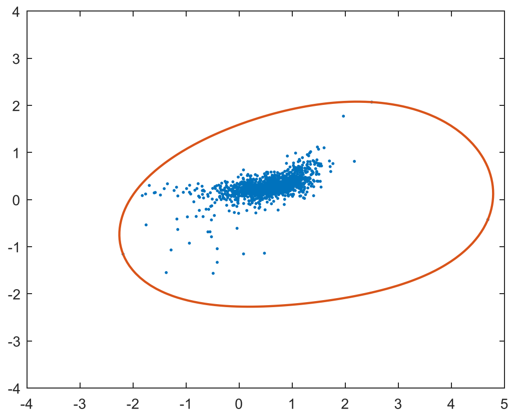


$2.6\% \leq \mathsf{R}(\mathcal{H}) \leq 8.5\%$

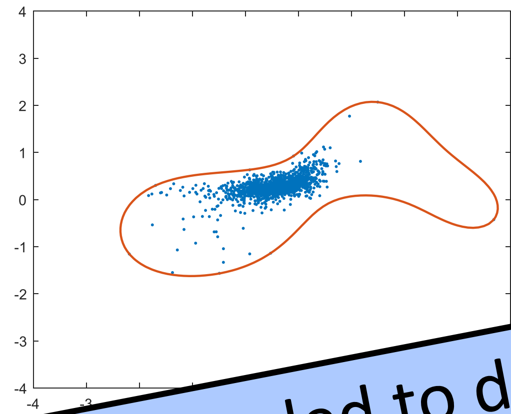


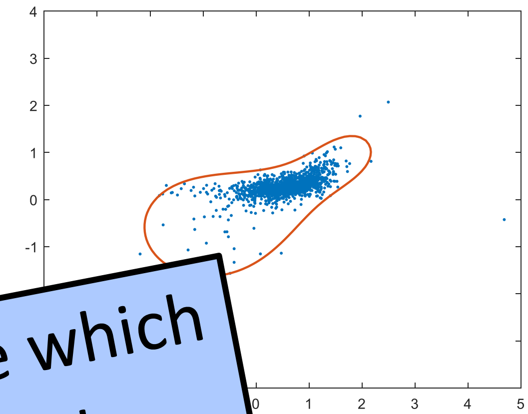
$12.4\% \leq \mathsf{R}(\mathcal{H}) \leq 22.7\%$

Example: reachability analysis via SVDD



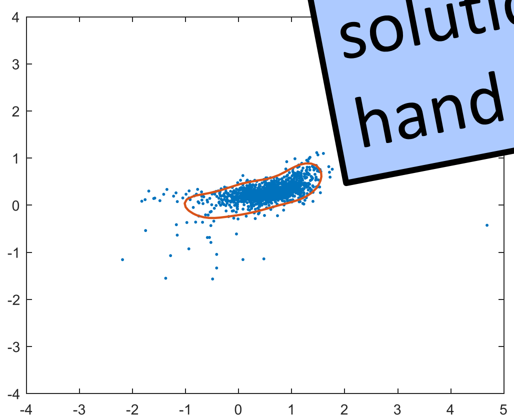
$0\% \leq \mathrm{R}(\mathcal{H}) \leq 1.9\%$



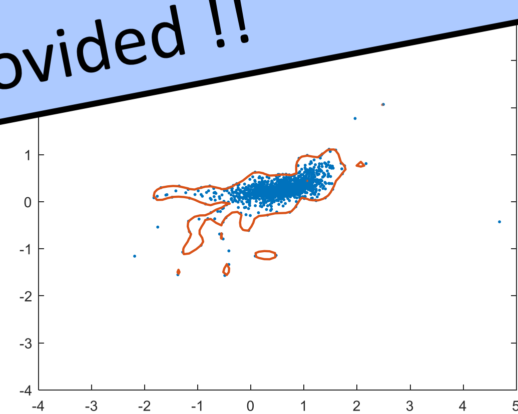


$(\mathcal{H}) \leq 2.8\%$

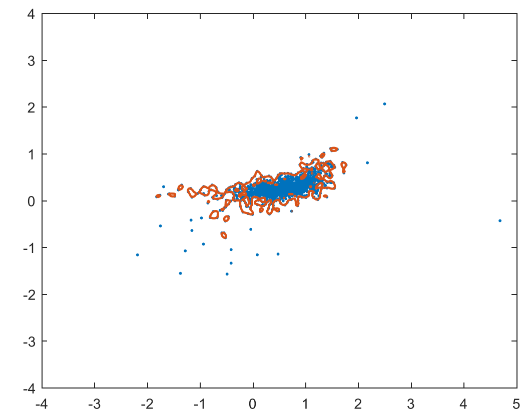
The information needed to decide which solution is best for the application at hand is provided !!



$4.3\% \leq \mathrm{R}(\mathcal{H}) \leq 11.2\%$



$2.6\% \leq \mathrm{R}(\mathcal{H}) \leq 8.5\%$

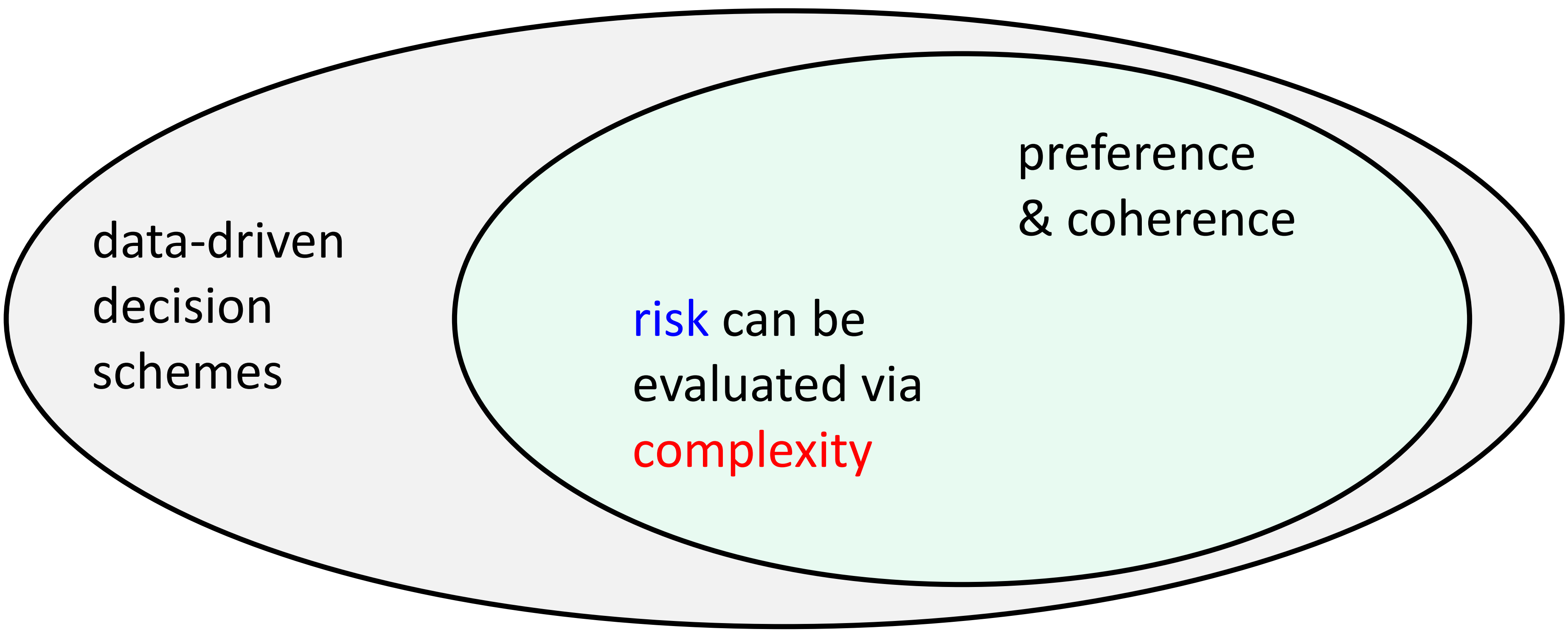


$12.4\% \leq \mathrm{R}(\mathcal{H}) \leq 22.7\%$

Scenario Approach: range of applicability

Many decision schemes (all scenario optimization schemes, many schemes in ML…) naturally satisfy compression properties… many yet to be discovered…

However, many others do not… notably: SGD



Scenario Approach: range of applicability

🙂 Many decision schemes (all scenario optimization schemes, many schemes in ML…) naturally satisfy compression properties… many yet to be discovered…

☹️ However, many others do not… notably: SGD

see also ***WeC02.4***

↑

Idea – the Pick-to-Learn (P2L) algorithm:

a meta-algorithm that builds on an existing data-driven decision scheme as a block-box to induce the compression properties

The Pick-to-Learn (P2L) algorithm

INPUT: scenarios $\delta_1, \delta_2, \ldots, \delta_N$, decision algorithm $\mathfrak{L}$,
initial decision $\mathcal{H}_0$

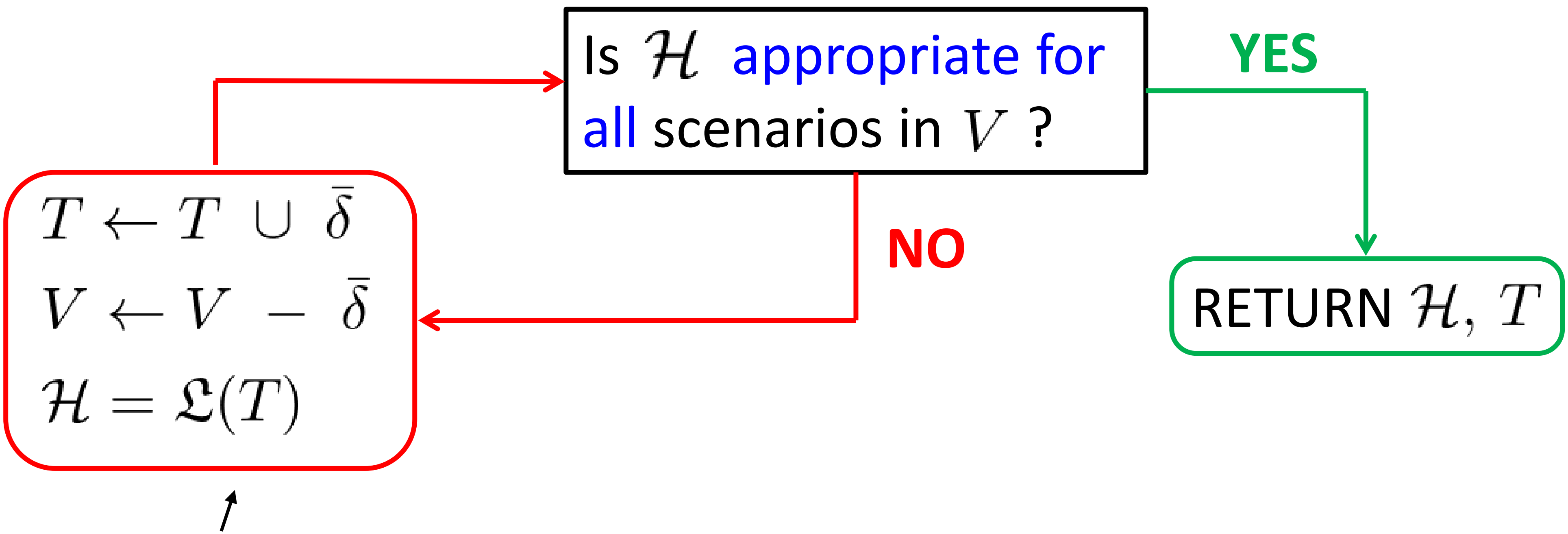
Possibly, **not** linkable to any meaningful compression

theory of the scenario approach **cannot** be directly used to evaluate the risk

The Pick-to-Learn (P2L) algorithm

INPUT: scenarios $\delta_1, \delta_2, \ldots, \delta_N$, decision algorithm $\mathfrak{L}$, initial decision $\mathcal{H}_0$

Initialization: $T = \emptyset,\ V = (\delta_1, \ldots, \delta_N),\ \mathcal{H} = \mathcal{H}_0$



$\bar{\delta}$ = element in V for which $\mathcal{H}$ is most inappropriate

P2L: main features

P2L: $\delta_1, \ldots, \delta_N \to \mathcal{H}$

⇨ new data-driven decision scheme $\mathfrak{L}'$

P2L: $\delta_1, \ldots, \delta_N \to T$

⇨ compression function κ' associated to $\mathfrak{L}'$

P2L: main features

P2L: $\delta_1, \ldots, \delta_N \to \mathcal{H}$

⇨ new data-driven decision scheme $\mathfrak{L}'$

P2L: $\delta_1, \ldots, \delta_N \to T$

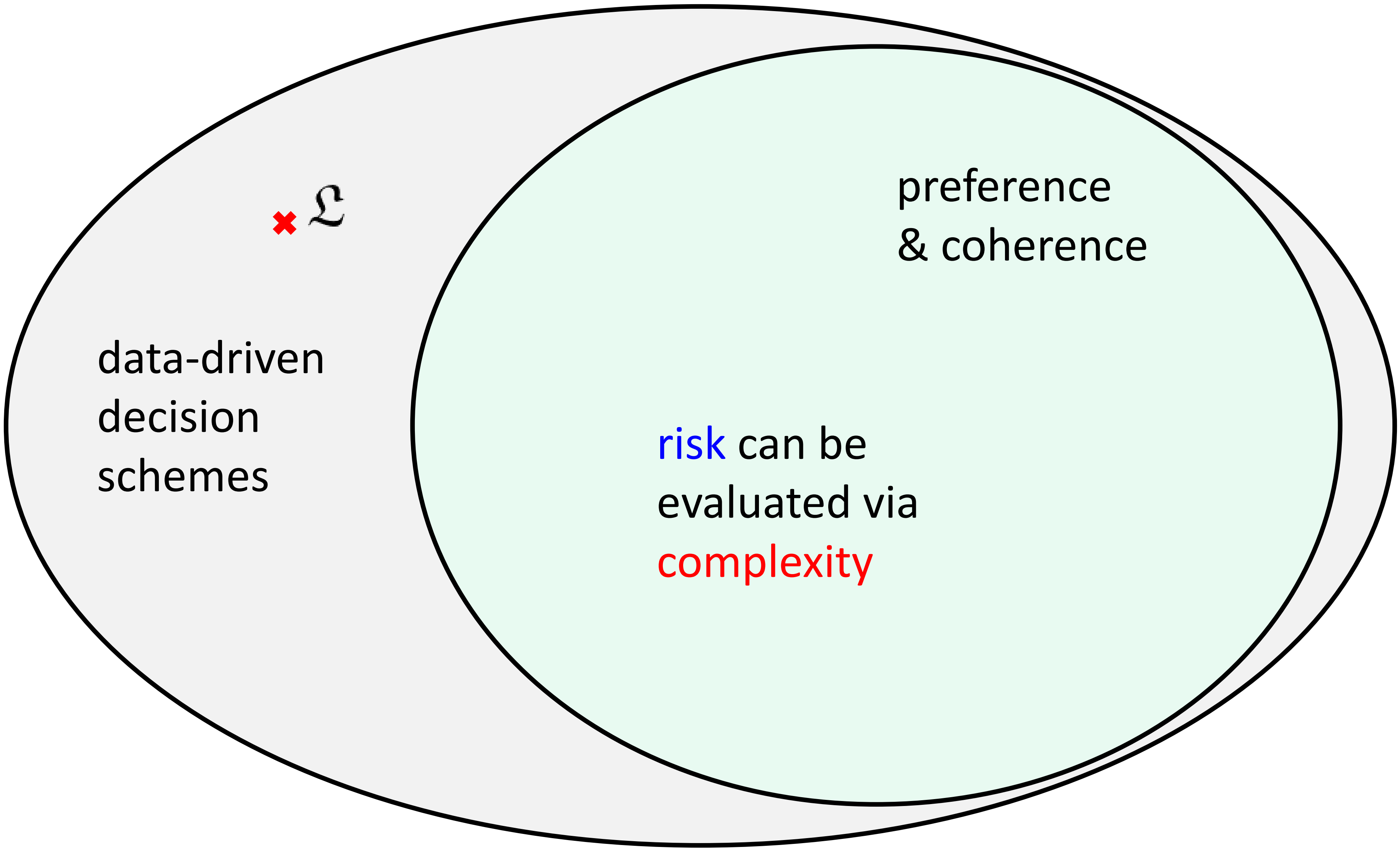
⇨ compression function κ' associated to $\mathfrak{L}'$

Theorem (with D. Paccagnan and M. Campi)

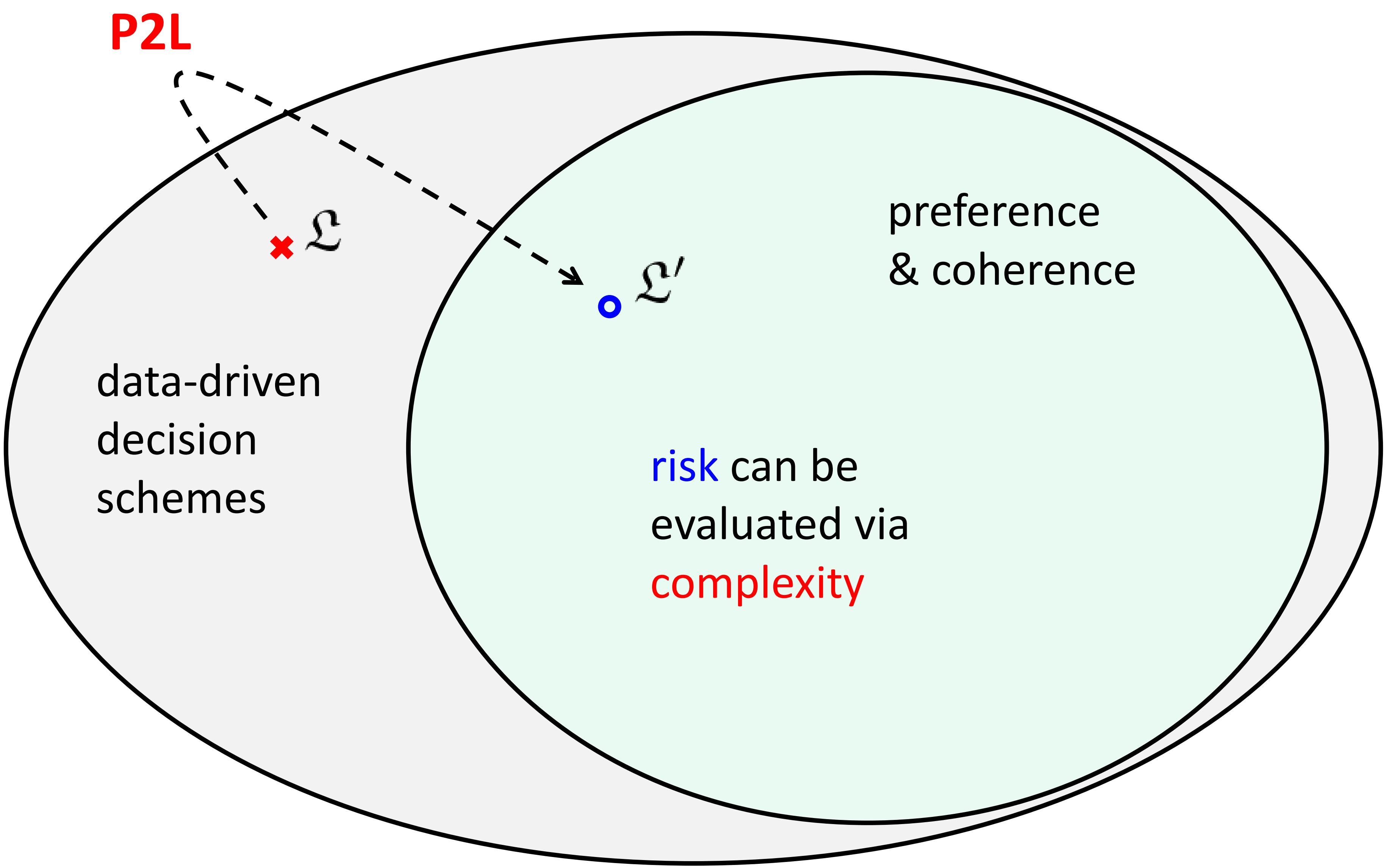
Preference and coherence hold true!

⇨ the risk of $\mathcal{H} = \mathfrak{L}'(\delta_1, \ldots, \delta_N)$ can be assessed via the size of T

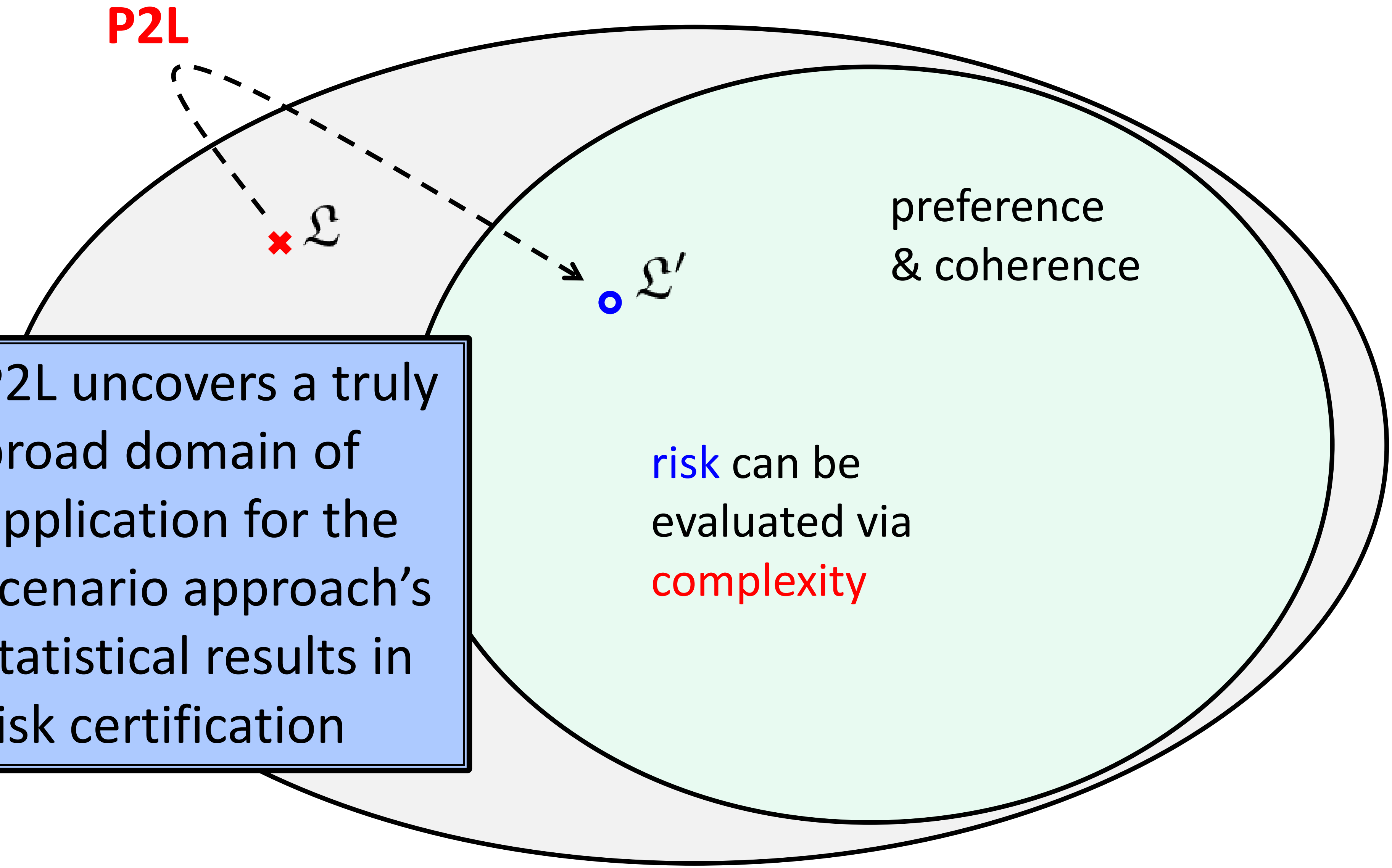
Scenario Approach: range of applicability (cont'd)



Scenario Approach: range of applicability (cont'd)



Scenario Approach: range of applicability (cont'd)



P2L uncovers a truly broad domain of application for the scenario approach's statistical results in risk certification

Thank you !

Relevant articles:

- *M.C. Campi, S. Garatti.* Compression, Generalization and Learning. *Journal of Machine Learning Research,* 24(339):1-74, 2023.

- *D. Paccagnan, M.C. Campi, S. Garatti,* The Pick-to-Learn Algorithm: Empowering Compression for Tight Generalization Bounds and Improved Post-training Performance. *In: Advances in Neural Information Processing Systems 36 (NeurIPS 2023),* 2023.

Research supported by:

- FAIR (Future Artificial Intelligence Research) project, funded by the NextGenerationEU program within the PNRR-PE-AI scheme (M4C2, Investment 1.3, Line on Artificial Intelligence)

- PRIN 2022 project 2022RRNAEX “The Scenario Approach for Control and Non-Convex Design” (CUP: D53D23001440006), funded by the NextGeneration EU program (Mission4, Component 2, Investment 1.1)

- PRIN PNRR project P2022NB77E “A data-driven cooperative framework for the management of distributed energy and water resources” (CUP: D53D23016100001), funded by the NextGeneration EU program (Mission 4, Component 2, Investment 1.1).