Teaching machines to discover particles

Gilles Louppe
@glouppe

HEP seminar, Radboud University. November 2, 2017.

How do you look for new laws?

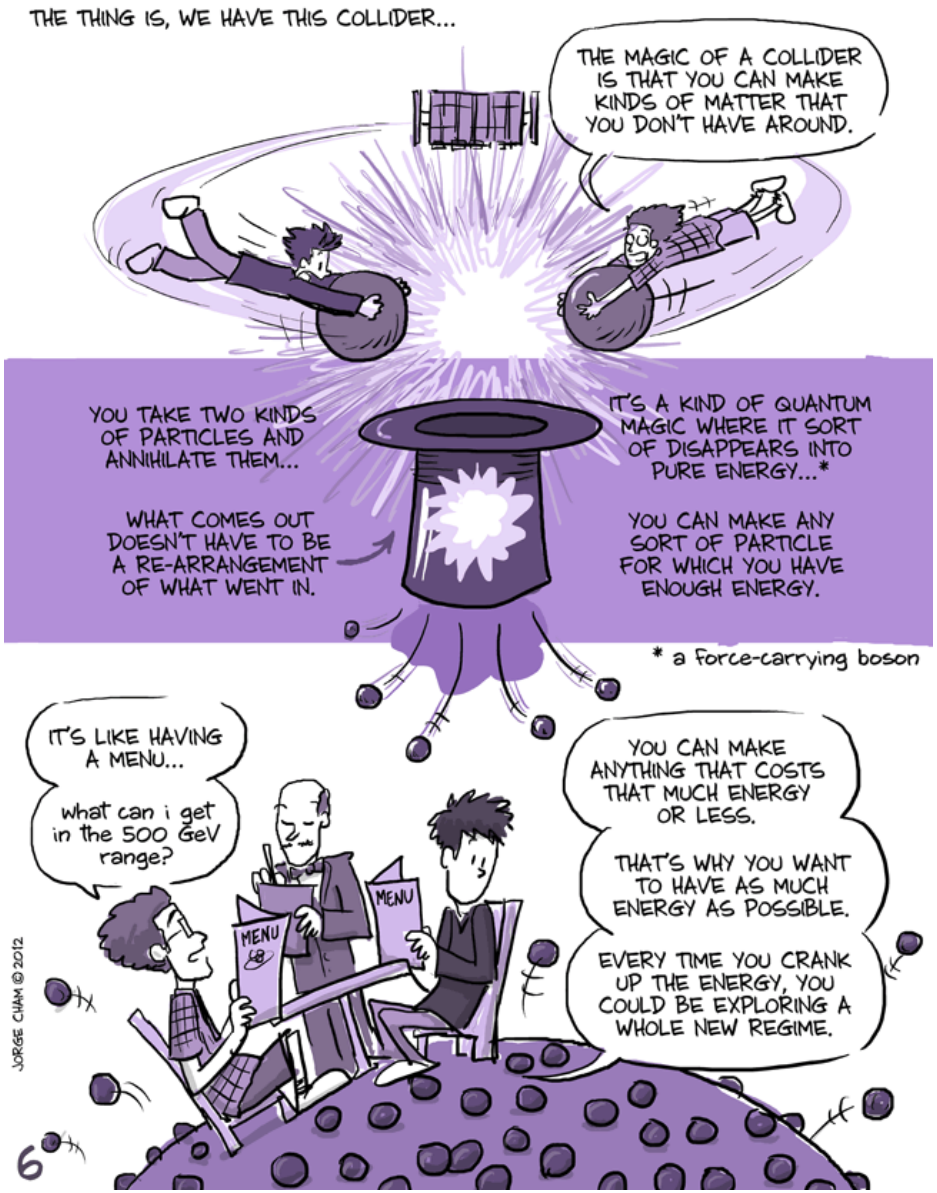
[watch here]

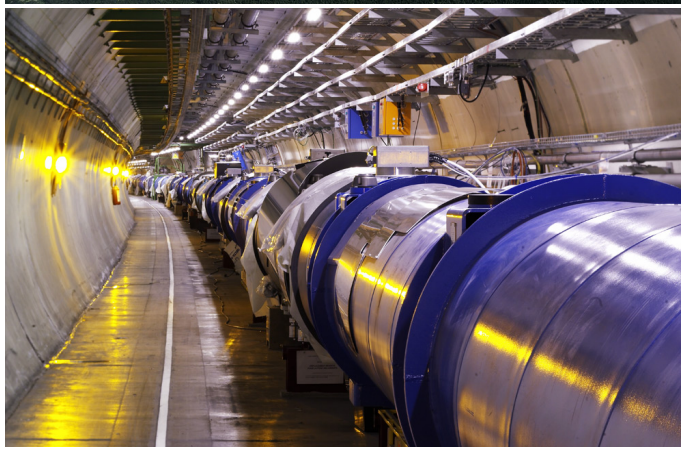
Can we automate the scientific method?

Particle Physics 101

Credits: Jorge Cham, PHD comics

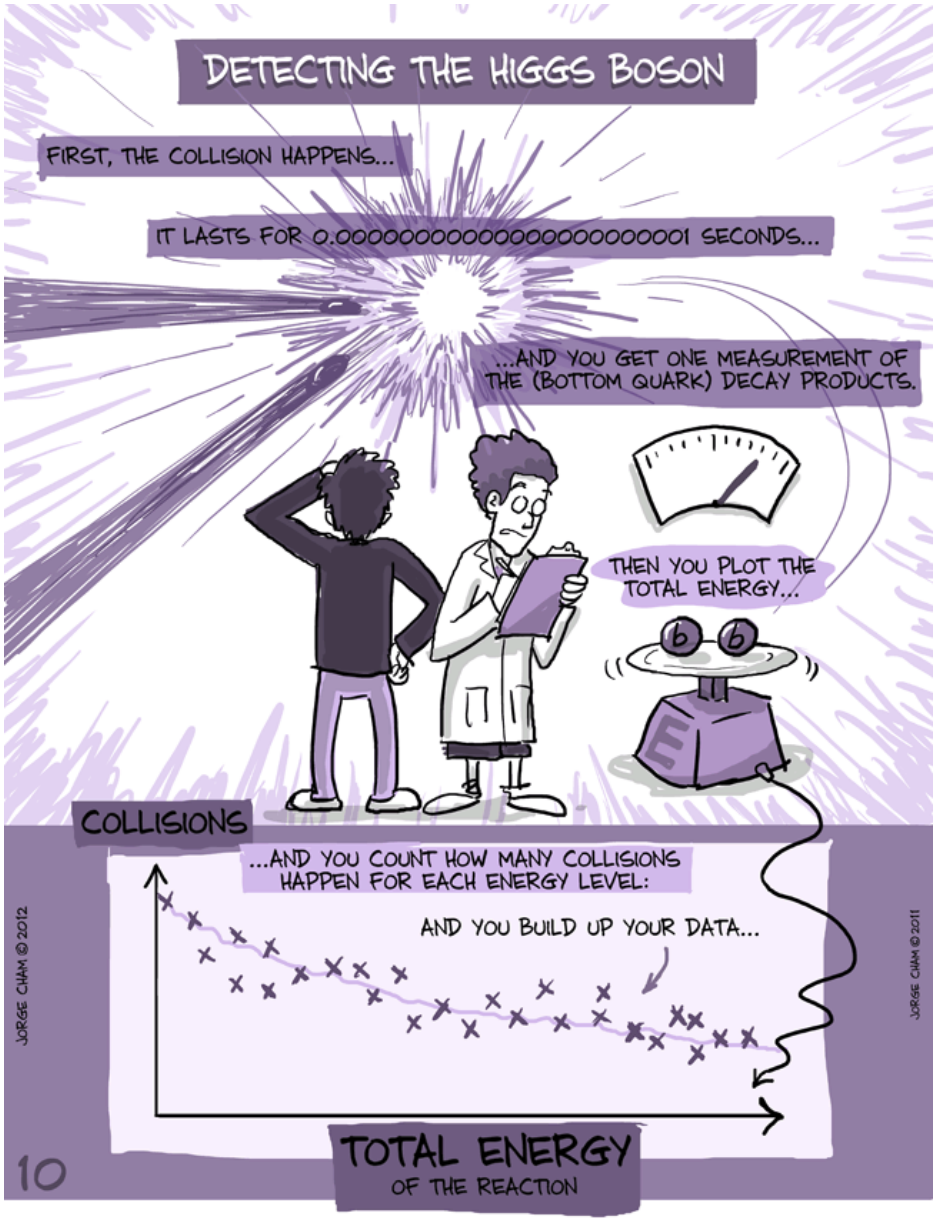
Testing for new physics

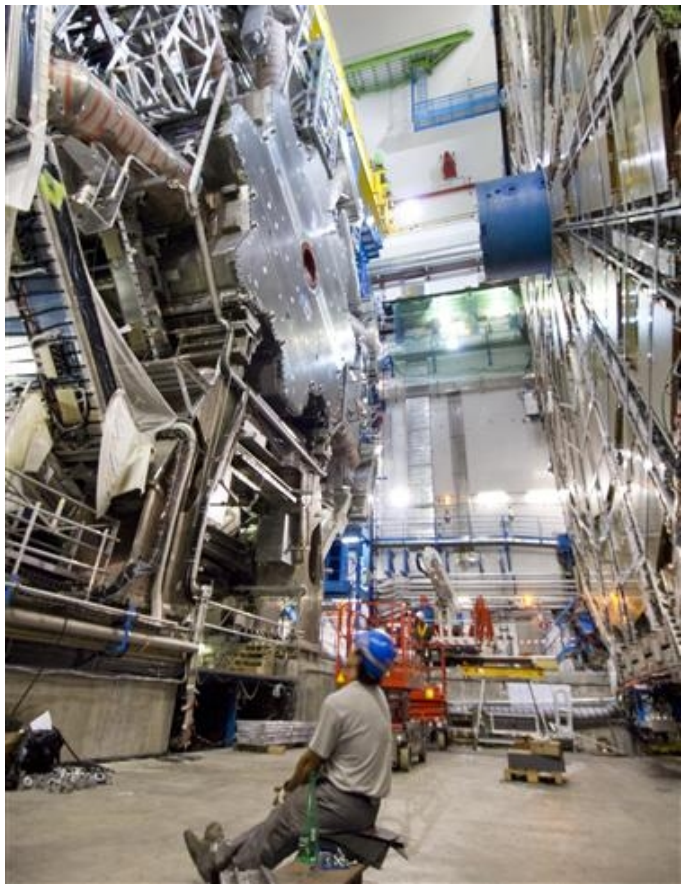




Testing for new physics

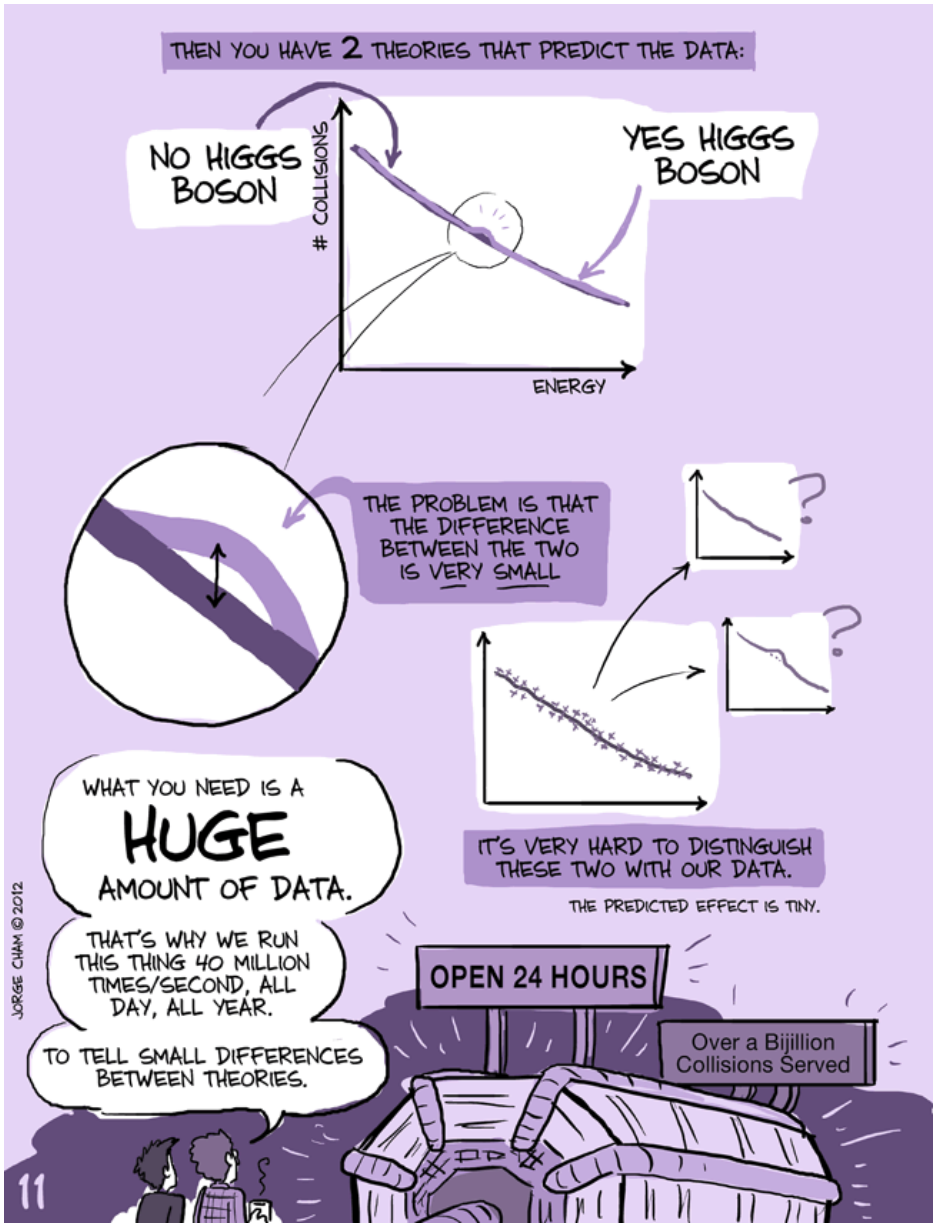
Credits: Jorge Cham, PHD comics

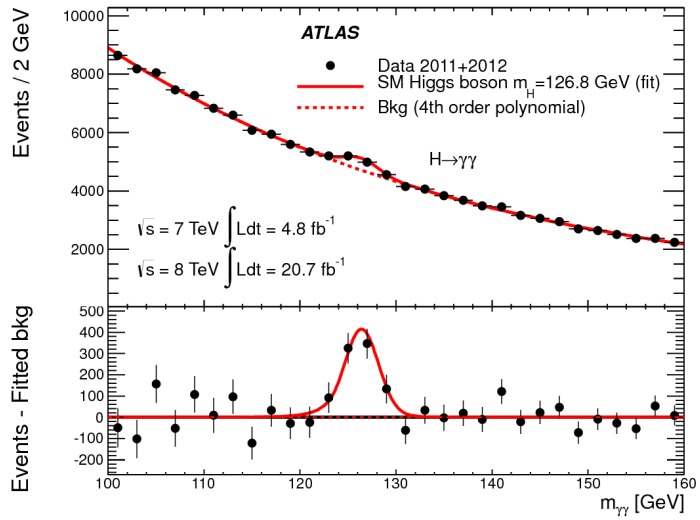




Credits: Jorge Cham, PHD comics

Testing for new physics

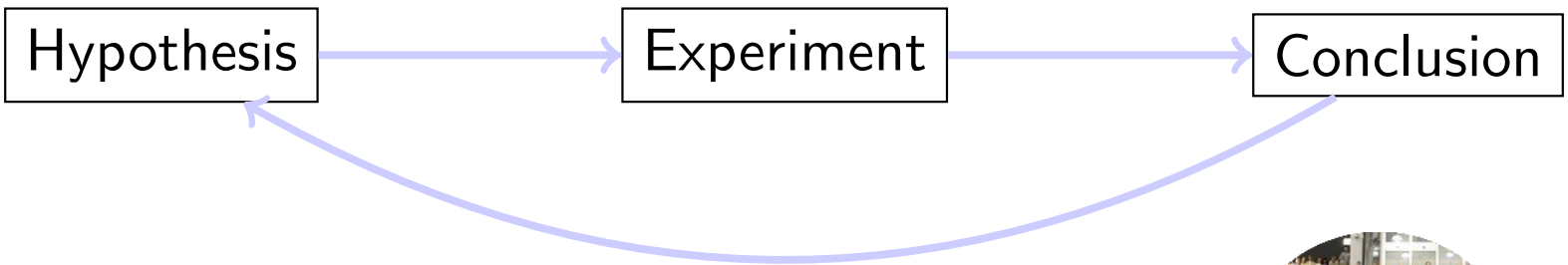


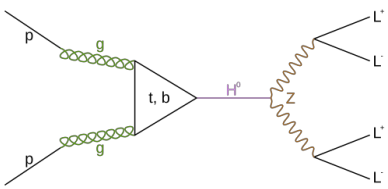


Hypothesis test based on the likelihood ratio

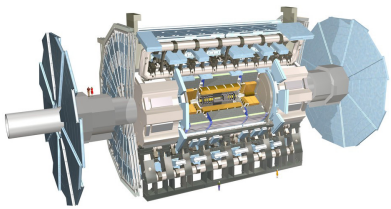
$$\frac{p(\mathbf{x}|\text{background})}{p(\mathbf{x}|\text{background} + \text{signal})}$$

The scientific method





The Higgs boson exists



LHC+ATLAS+CMS

Discovery!

Credits: Kyle Cranmer

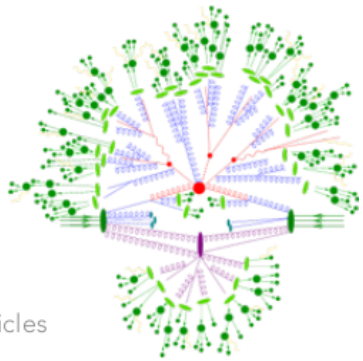
The Standard Model

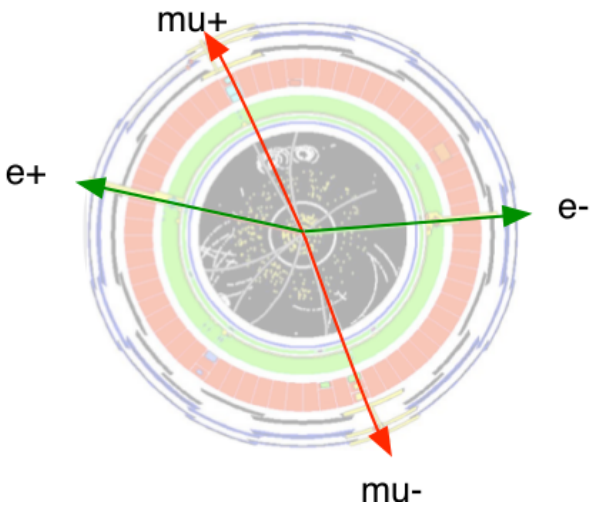
$$\mathcal{L}_{SM} = \underbrace{\frac{1}{4}\mathbf{W}_{\mu\nu}\cdot\mathbf{W}^{\mu\nu} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} - \frac{1}{4}G^a_{\mu\nu}G^{\mu\nu}_a}_{\text{kinetic energies and self-interactions of the gauge bosons}}$$
$$+ \underbrace{\bar{L}\gamma^\mu(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)L + \bar{R}\gamma^\mu(i\partial_\mu - \frac{1}{2}g'YB_\mu)R}_{\text{kinetic energies and electroweak interactions of fermions}}$$
$$+ \underbrace{\frac{1}{2}\left|(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)\,\phi\right|^2 - V(\phi)}_{W^\pm, Z, \gamma \text{and Higgs masses and couplings}}$$
$$+ \underbrace{g''(\bar{q}\gamma^\mu T_a q)\,G^a_\mu}_{\text{interactions between quarks and gluons}} + \underbrace{(G_1\bar{L}\phi R + G_2\bar{L}\phi_c R + h.c.)}_{\text{fermion masses and couplings to Higgs}}$$

1) We begin with Quantum Field Theory

2) Theory gives detailed prediction for high-energy collisions

hierarchical: 2 → O(10) → O(100) particles





3) The interaction of outgoing particles with the detector is simulated.

>100 million sensors

4) Finally, we run particle identification and feature extraction algorithms on the simulated data as if they were from real collisions.

~10-30 features describe interesting part

The **uniqueness** of particle physics lies in its highly precise and compact model.

This model should be leveraged by ML!

Likelihood-free inference

The players

Forward modeling
Generation
Simulation

Latent variables $\mathbf{z}$

$\boldsymbol{\theta} := (\boldsymbol{\mu}, \boldsymbol{\nu})$
Parameters

Prediction

$\mathbf{x} \sim p_r(\mathbf{x})$
Observations drawn from Nature

$\boldsymbol{\mu}$
Parameters of interest

$p(\mathbf{x}|\boldsymbol{\theta})$

$\mathbf{x} \sim p(\mathbf{x}|\boldsymbol{\theta})$
Simulated data (a lot!)

$\boldsymbol{\nu}$
Nuisance parameters

Inference

Inverse problem
Unfolding
Measurement

Parameter search

Likelihood-free assumptions

Operationally,

$$\mathbf{x} \sim p(\mathbf{x}|\boldsymbol{\theta}) \equiv \mathbf{z} \sim p(\mathbf{z}|\boldsymbol{\theta}), \mathbf{x} = g(\mathbf{z}; \boldsymbol{\theta})$$

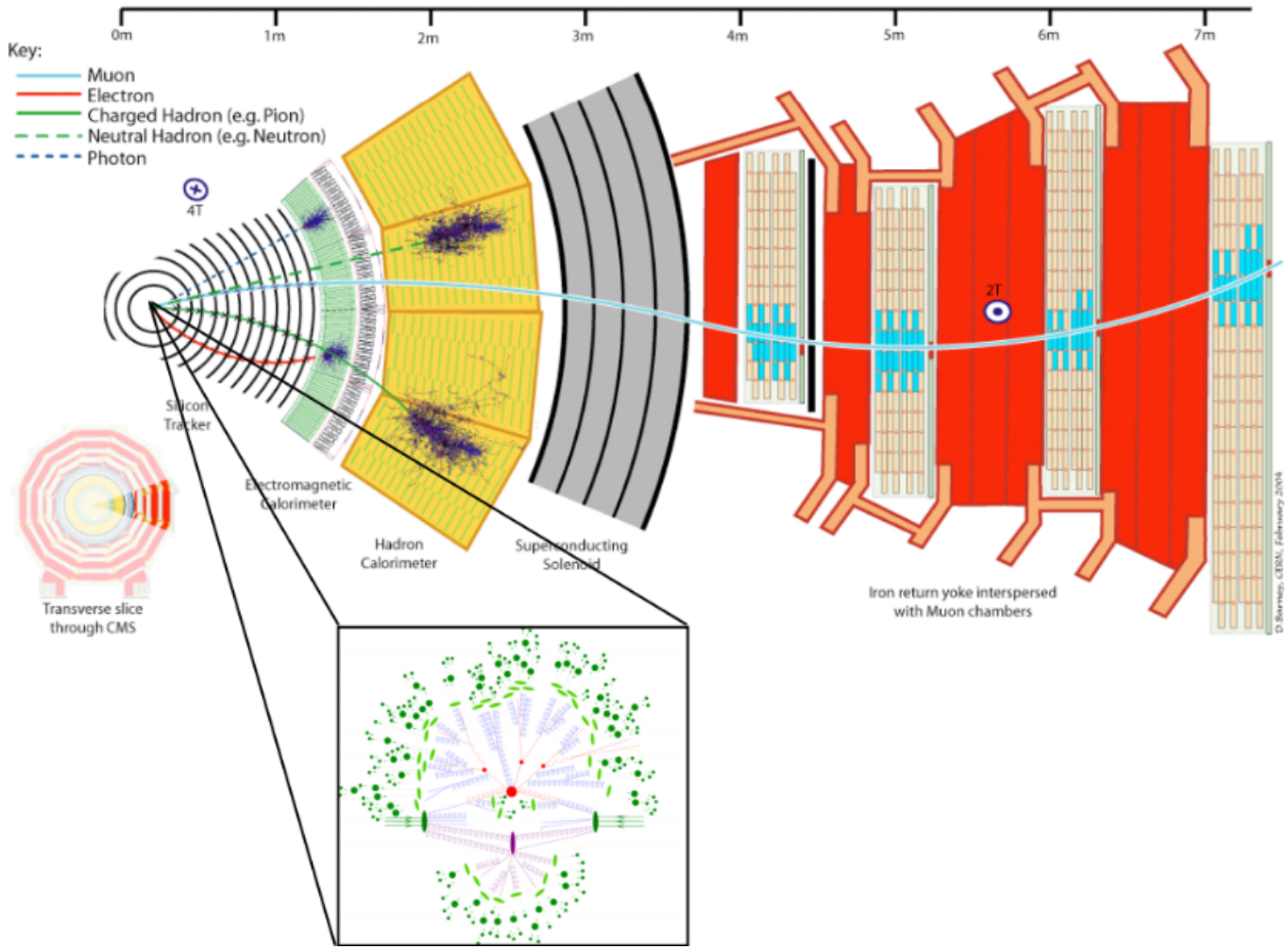
where

- $\mathbf{z}$ provides a source of randomness;
- g is a non-differentiable deterministic function (e.g. a computer program).

Accordingly, the density $p(\mathbf{x}|\boldsymbol{\theta})$ can be written as

$$p(\mathbf{x}|\boldsymbol{\theta}) = \int_{\{\mathbf{z}: g(\mathbf{z};\theta)=\mathbf{x}\}} p(\mathbf{z}|\boldsymbol{\theta}) d\mathbf{z}$$

Evaluating the integral is often intractable.



Determining and evaluating all possible execution paths and all $\mathbf{z}$ that lead to the observation $\mathbf{x}$ is not tractable.

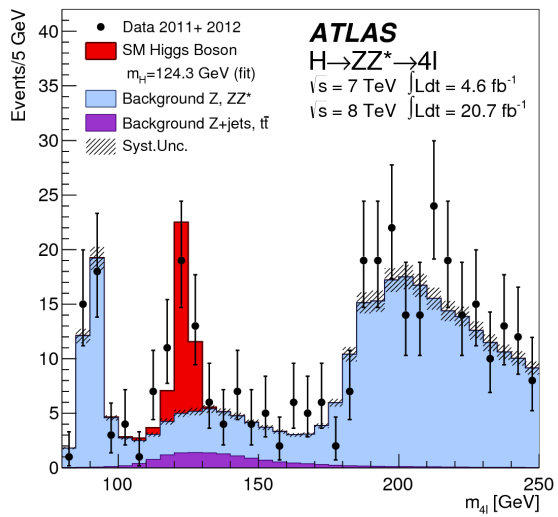
(And even less, normalizing that thing!)

Testing hypothesis (Inference)

Formally, physicists usually test a null $\boldsymbol{\theta} = \boldsymbol{\theta}_0$ by constructing the likelihood ratio test statistic

$$\Lambda(\mathcal{D}; \boldsymbol{\theta}_0) = \frac{p(\mathcal{D}|\boldsymbol{\theta}_0)}{\sup_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} p(\mathcal{D}|\boldsymbol{\theta})} = \frac{\prod_{\mathbf{x} \in \mathcal{D}} p(\mathbf{x}|\boldsymbol{\theta}_0)}{\sup_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \prod_{\mathbf{x} \in \mathcal{D}} p(\mathbf{x}|\boldsymbol{\theta})}$$

- Most measurements and searches for new particles are based on the distribution of a single variable $\mathbf{x} \in \mathbb{R}$.
- The likelihood $p(\mathbf{x}|\boldsymbol{\theta})$ is approximated using 1D histograms.
- Choosing a good variable $\mathbf{x}$ tailored for the goal of the experiment is the physicist's job.



Likelihood-free inference (Inference)

Given observations $\mathbf{x} \sim p_r(\mathbf{x})$, we seek:

$$\boldsymbol{\theta}^* = \arg\max_{\theta} p(\mathbf{x}|\boldsymbol{\theta})$$

- Histogramming $p(\mathbf{x}|\boldsymbol{\theta})$ does not scale to high dimensions.
- Can we automate or bypass the physicist's job of thinking about a good and compact representation for $\mathbf{x}$, without losing information?
- Hint: We do not need to know $p(\mathbf{x}|\boldsymbol{\theta})$ to find $\boldsymbol{\theta}^*$.

Credits: 1506.02169

Approximating likelihood ratios with classifiers (CARL)

The likelihood ratio $r(\mathbf{x})$ is invariant under the change of variable $\boldsymbol{u} = s(\mathbf{x})$, provided $s(\mathbf{x})$ is monotonic with $r(\mathbf{x})$:

$$r(\mathbf{x}) = \frac{p(\mathbf{x}|\boldsymbol{\theta}_0)}{p(\mathbf{x}|\boldsymbol{\theta}_1)} = \frac{p(s(\mathbf{x})|\boldsymbol{\theta}_0)}{p(s(\mathbf{x})|\boldsymbol{\theta}_1)}$$

A classifier s trained to distinguish $\mathbf{x} \sim p(\mathbf{x}|\theta_0)$ from $\mathbf{x} \sim p(\mathbf{x}|\theta_1)$ satisfies the condition above.

This gives an automatic procedure for learning a good and compact representation for $\mathbf{x}$!

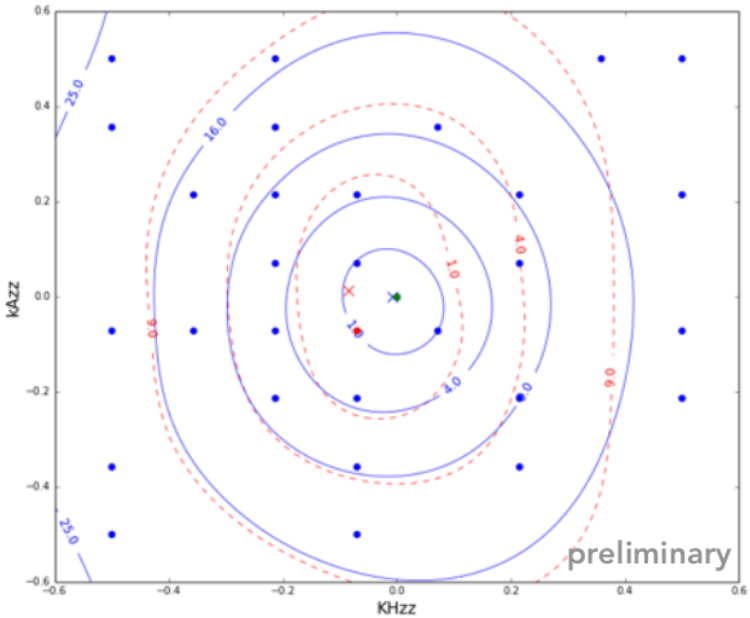
Therefore,

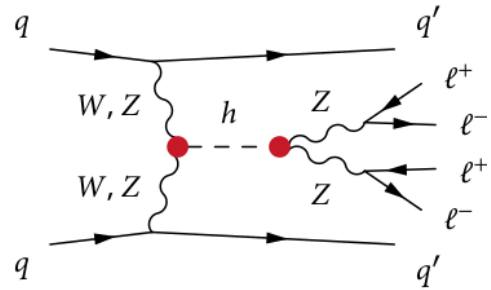
$$\begin{aligned}\boldsymbol{\theta}^* &= \arg\max_{\theta} p(\mathbf{x}|\boldsymbol{\theta}) \\ &= \arg\max_{\theta} \frac{p(\mathbf{x}|\boldsymbol{\theta})}{p(\mathbf{x}|\boldsymbol{\theta}_1)} \\ &= \arg\max_{\theta} \frac{p(s(\mathbf{x};\boldsymbol{\theta},\boldsymbol{\theta}_1)|\boldsymbol{\theta})}{p(s(\mathbf{x};\boldsymbol{\theta},\boldsymbol{\theta}_1)|\boldsymbol{\theta}_1)}\end{aligned}$$

where $\boldsymbol{\theta}_1$ is fixed and $s(\mathbf{x};\boldsymbol{\theta},\boldsymbol{\theta}_1)$ is a family of classifiers trained to distinguish between $\boldsymbol{\theta}$ and $\boldsymbol{\theta}_1$.

Application to the Higgs

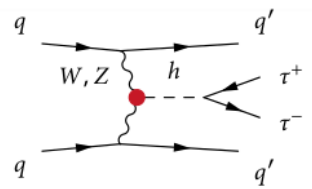
Preliminary work using fast detector simulation and CARL to approximate likelihoods using full kinematic information parameterized in coefficients of a Quantum Field Theory.





16 covariates
(using the CARL)

2 covariates
(density estimation)



Equivalent to 3x more data.
(idealized, no systematic uncertainty)

work with Juan Pavez, Gilles Louppe, Cyril Becot, and Lukas Heinrich; Johann Brehmer, Felix Kling, and Tilman Plehn

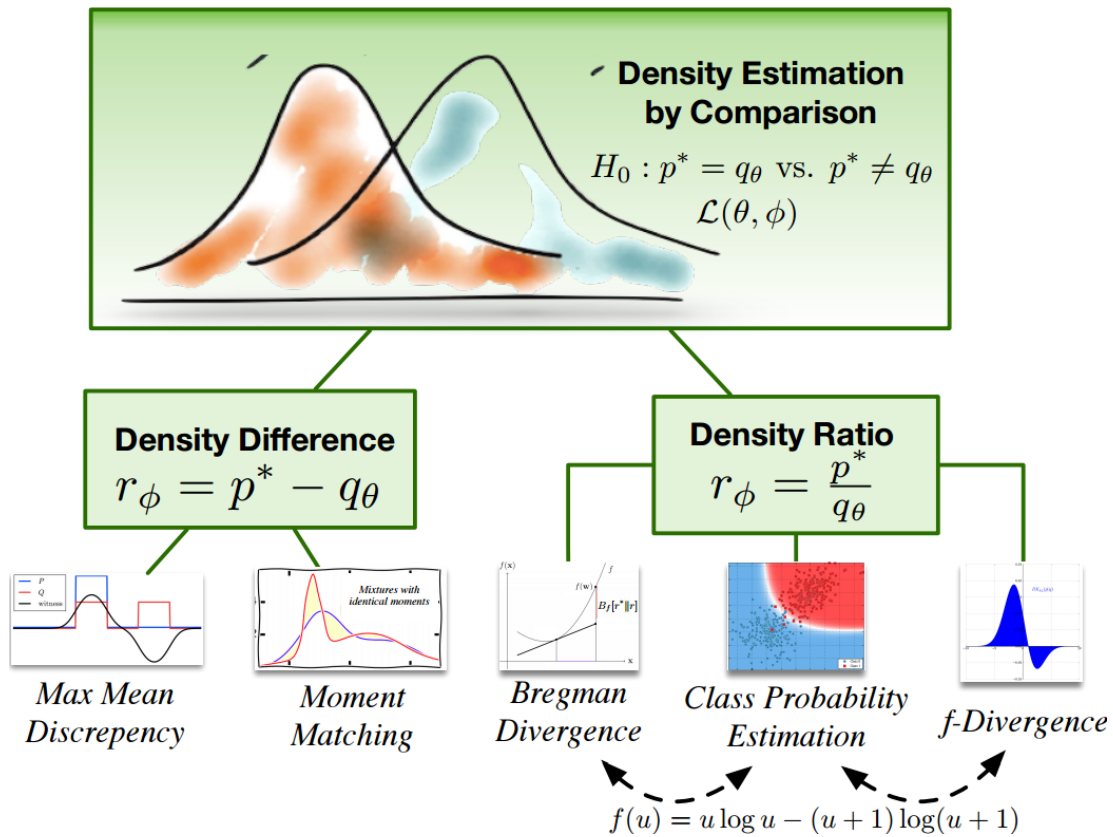
Credits: 1610.03483

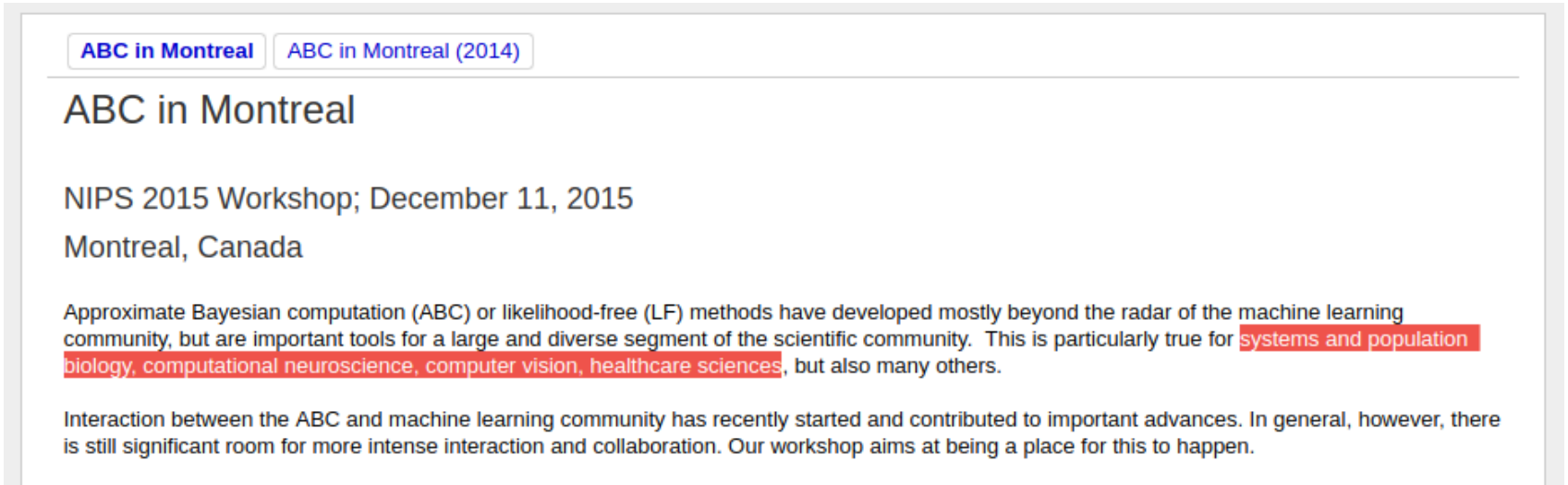
Learning in implicit generative models

Likelihood-free inference can be cast into the framework of "implicit generative models".

This framework ties together:

- Approximate Bayesian computation
- Density estimation-by-comparison algorithms (two sample testing, density ratio, density difference estimation)
- Generative adversarial networks
- Variational inference





Likelihood-free inference has become
a hot topic in machine learning!

Workshop Aims

Probabilistic models are an important tool in machine learning. They form the basis for models that generate realistic data, uncover hidden structure, and make predictions. Traditionally, probabilistic models in machine learning have focused on prescribed models. Prescribed models specify a joint density over observed and hidden variables that can be easily evaluated. The requirement of a tractable density simplifies their learning but limits their flexibility --- several real world phenomena are better described by simulators that do not admit a tractable density. Probabilistic models defined only via the simulations they produce are called implicit models.

Arguably starting with generative adversarial networks, research on implicit models in machine learning has exploded in recent years. This workshop's aim is to foster a discussion around the recent developments and future directions of implicit models.

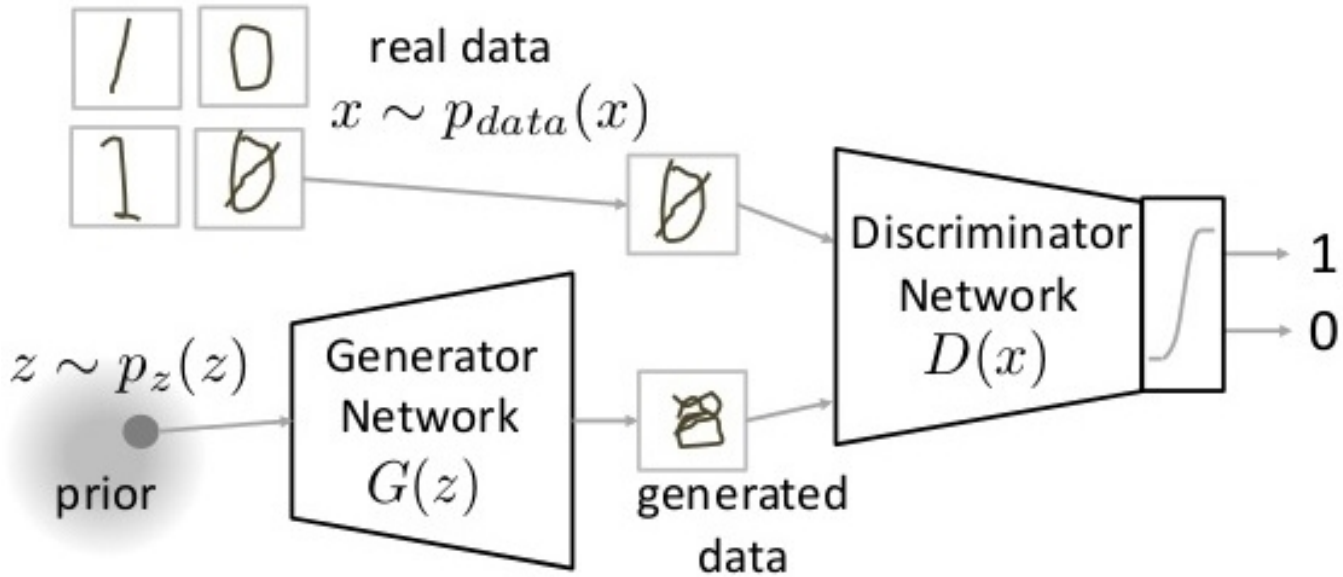
Implicit models have many applications. They are used in ecology where models simulate animal populations over time; they are used in phylogeny, where simulations produce hypothetical ancestry trees; they are used in physics to generate particle simulations for high energy processes. Recently, implicit models have been used to improve the state-of-the-art in image and content generation. Part of the workshop's focus is to discuss the commonalities among applications of implicit models.

Fast simulation (Prediction)

- Half the LHC computing power (300000 cores) is dedicated to producing simulated data.
- Huge savings (in time and $) if simulations can be made faster.
- Hand-made fast simulators are being developed by physicists, trading-off precision for speed.

Can we learn to generate data?
(i.e. can we build a fast proxy for $\mathbf{x} \sim p(\mathbf{x}|\boldsymbol{\theta})$*?)*

Generative adversarial networks



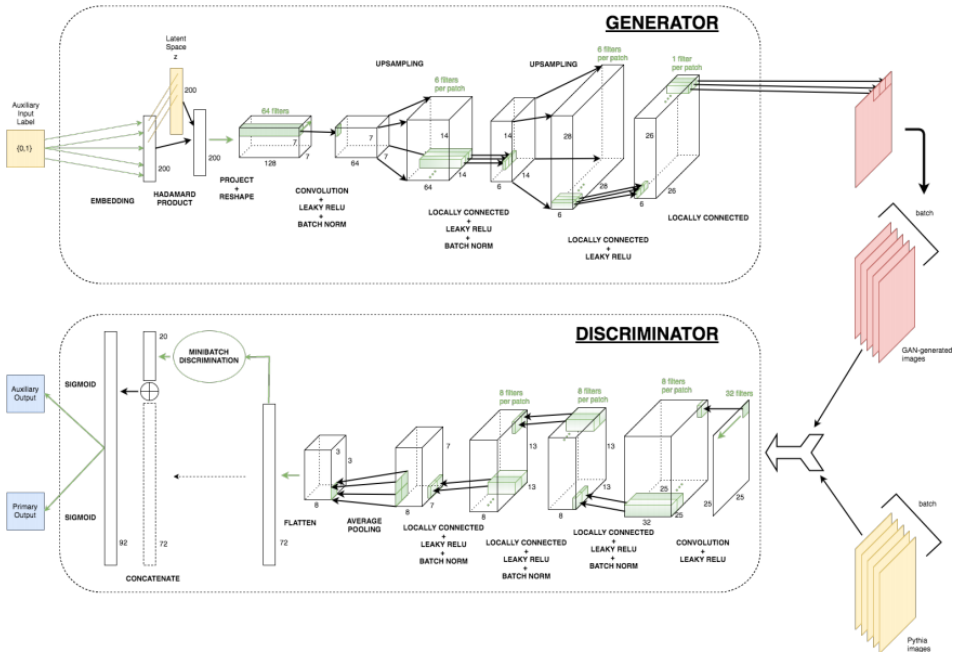
$$\mathcal{L}_d = \mathbb{E}_{\mathbf{z} \sim p(\mathbf{z})}[d(g(\mathbf{z}; \boldsymbol{\theta}); \boldsymbol{\phi})] - \mathbb{E}_{\mathbf{x} \sim p_r(\mathbf{x})}[d(\mathbf{x}; \boldsymbol{\phi})]$$

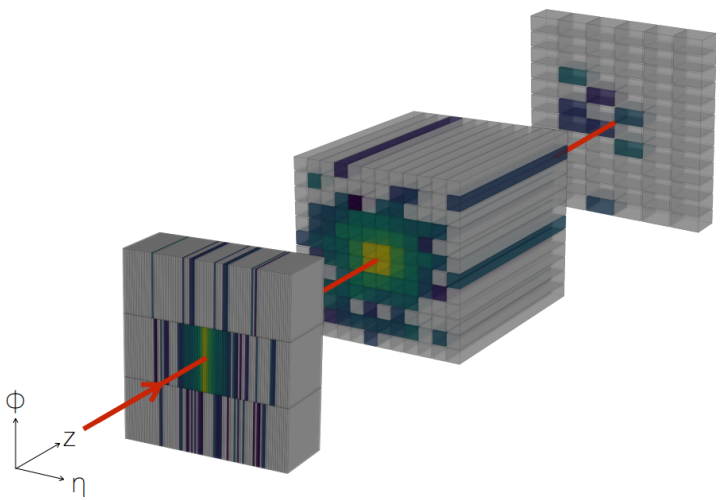
$$\mathcal{L}_g = -\mathcal{L}_d$$

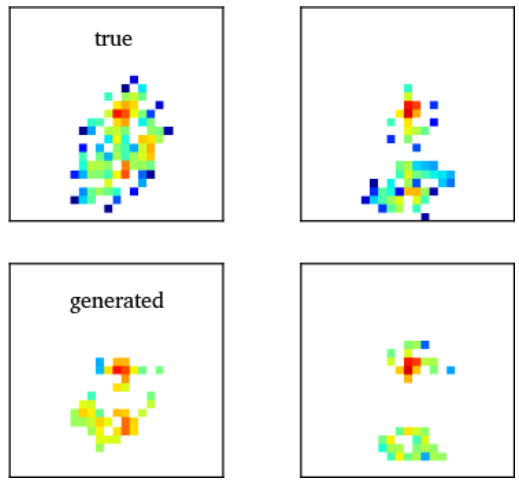
Which ones are real photographs?

Credits: 1701.05927, 1705.02355

Learning generative models (Prediction)







Challenges:

- How to ensure physical properties?
- Non-uniform geometry
- Mostly sparse
- GANs vs. VAE vs. Normalizing Flows?

What if the generator g in GANs isn't a neural net, but an actual physics simulator?

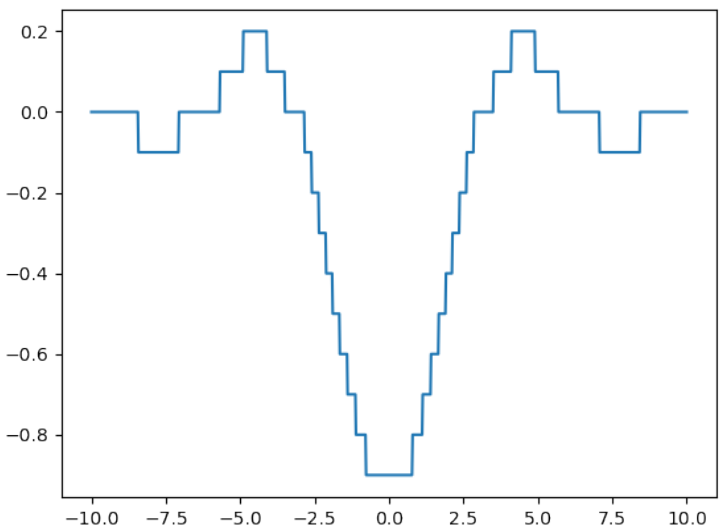
What if the generator g in GANs isn't a neural net, but an actual physics simulator?

Usually, we cannot compute $\nabla_{\boldsymbol{\theta}} d(g(\mathbf{z}; \boldsymbol{\theta}); \boldsymbol{\phi})$, because g is non-differentiable.

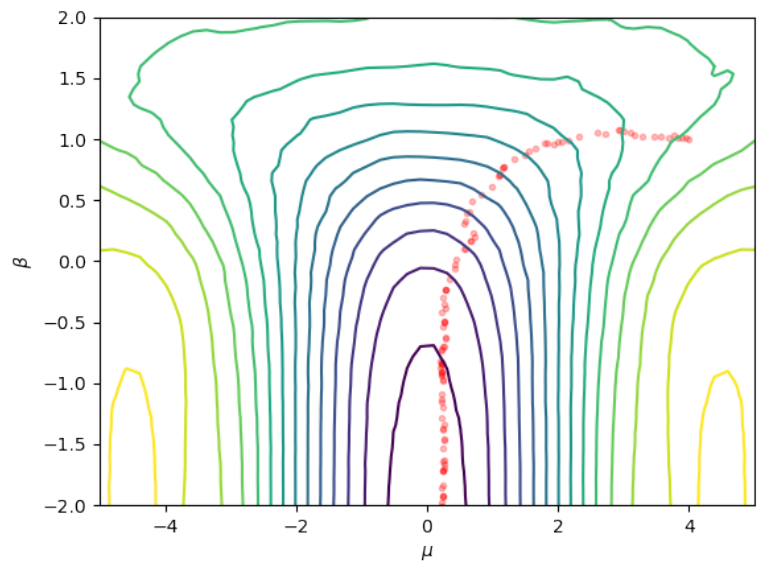
Variational Optimization

$$\min_{\boldsymbol{\theta}} f(\boldsymbol{\theta}) \leq \mathbb{E}_{\boldsymbol{\theta} \sim q(\boldsymbol{\theta}|\boldsymbol{\psi})}[f(\boldsymbol{\theta})] = U(\boldsymbol{\psi})$$

$$\nabla_{\boldsymbol{\psi}} U(\boldsymbol{\psi}) = \mathbb{E}_{\boldsymbol{\theta} \sim q(\boldsymbol{\theta}|\boldsymbol{\psi})}[f(\boldsymbol{\theta}) \nabla_{\boldsymbol{\psi}} \log q(\boldsymbol{\theta}|\boldsymbol{\psi})]$$



Piecewise constant $-\frac{\sin(\mathbf{x})}{\mathbf{x}}$



$q(\boldsymbol{\theta}|\boldsymbol{\psi} = (\mu, \beta)) = \mathcal{N}(\mu, e^{\beta})$

Adversarial Variational Optimization

- Replace the generative network with a non-differentiable forward simulator $g(\mathbf{z}; \boldsymbol{\theta})$.
- With VO, optimize upper bounds of the adversarial objectives:

$$U_d = \mathbb{E}_{\boldsymbol{\theta} \sim q(\boldsymbol{\theta}|\boldsymbol{\psi})}[\mathcal{L}_d] \tag{1}$$

$$U_g = \mathbb{E}_{\boldsymbol{\theta} \sim q(\boldsymbol{\theta}|\boldsymbol{\psi})}[\mathcal{L}_g] \tag{2}$$

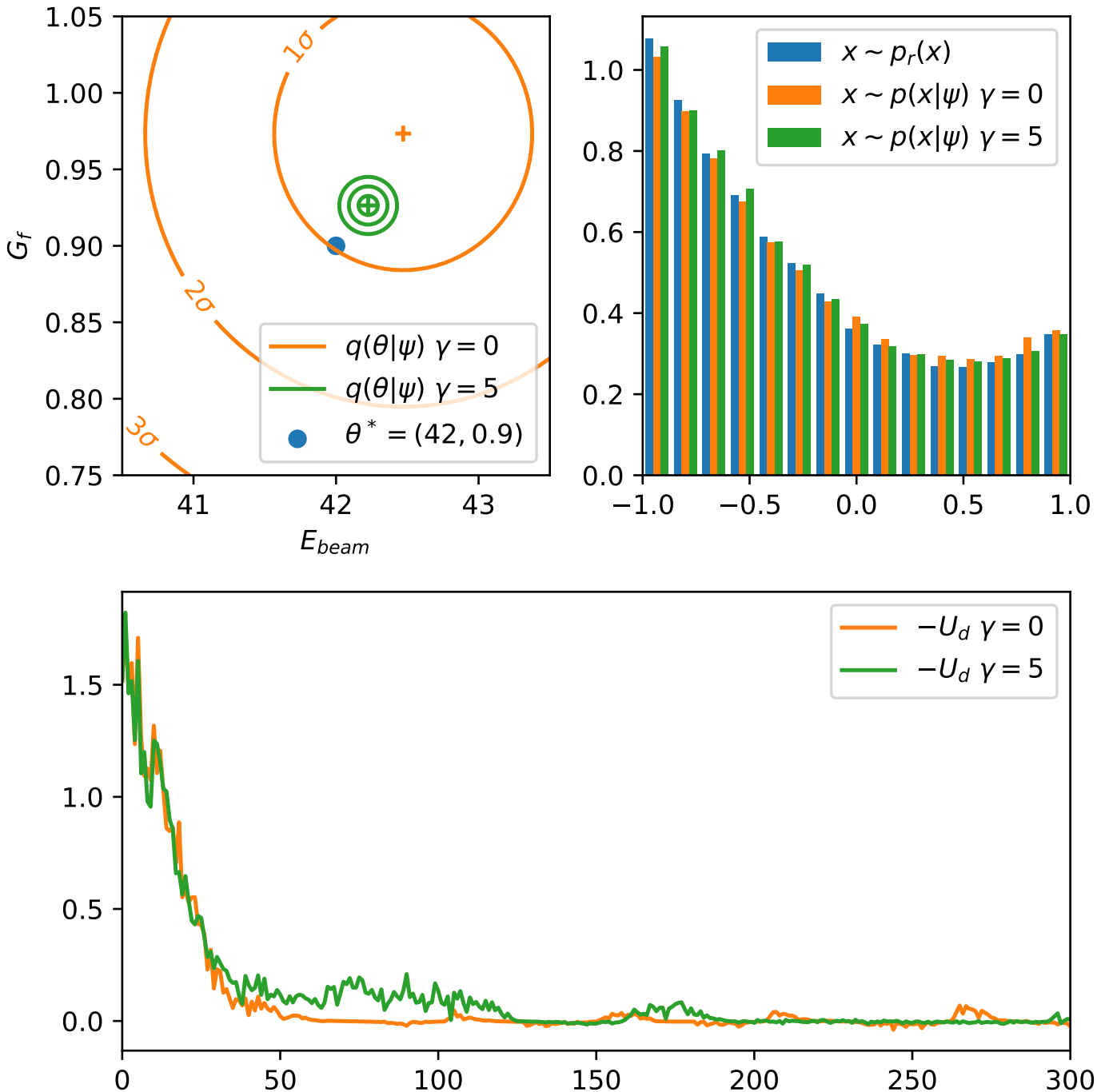
respectively over $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$.

Operationally, we get the marginal model:

$$\mathbf{x} \sim q(\mathbf{x}|\boldsymbol{\psi}) \equiv \boldsymbol{\theta} \sim q(\boldsymbol{\theta}|\boldsymbol{\psi}), \mathbf{z} \sim p(\mathbf{z}|\boldsymbol{\theta}), \mathbf{x} = g(\mathbf{z}; \boldsymbol{\theta})$$

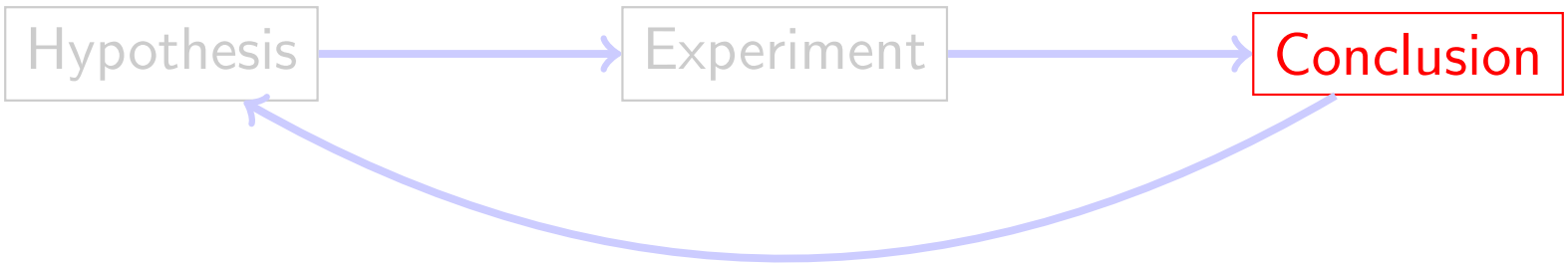
Toy example: $e^+e^- \to \mu^+\mu^-$

- Simplified simulator for electron–positron collisions resulting in muon–antimuon pairs.
- Observations: $\mathbf{x} = \cos(A) \in [-1, 1]$, where A is the polar angle of the outgoing muon wrt incoming electron.
- Parameters: E_{beam}, G_f.


1.05
1.00
0.95
0.90
0.85
0.80
0.75
G_f
41
42
43
E_{beam}
1σ
2σ
3σ
$q(\theta|\psi)\ \gamma = 0$
$q(\theta|\psi)\ \gamma = 5$
$\theta^* = (42, 0.9)$
1.0
0.8
0.6
0.4
0.2
0.0
−1.0
−0.5
0.0
0.5
1.0
$x \sim p_r(x)$
$x \sim p(x|\psi)\ \gamma = 0$
$x \sim p(x|\psi)\ \gamma = 5$
1.5
1.0
0.5
0.0
0
50
100
150
200
250
300
$-U_d\ \gamma = 0$
$-U_d\ \gamma = 5$

Powering the scientific method with AI

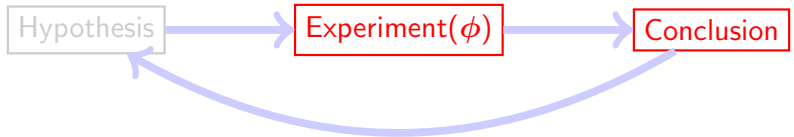
Automating the scientific process



Most efforts are focused on automating the analysis of experimental results to draw conclusions, assuming the hypothesis and experiment are fixed.

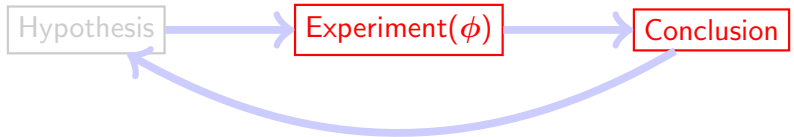
Can we also automate
the steps of hypothesis and experiments?

Optimal experimental desgin



Parameters $\boldsymbol{\theta}$ of the (standard) model are known with uncertainty $H[\boldsymbol{\theta}]$. *How to best reduce the uncertainty* $H[\boldsymbol{\theta}]$*?*

Optimal experimental desgin

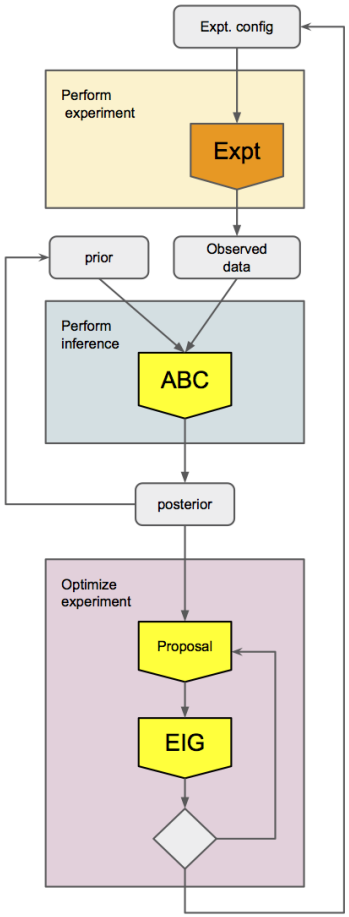


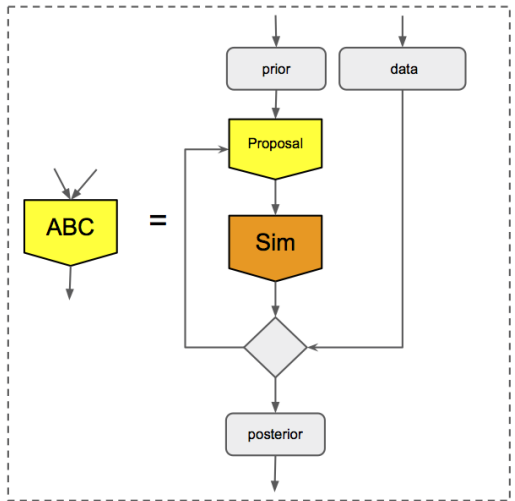
Parameters $\boldsymbol{\theta}$ of the (standard) model are known with uncertainty $H[\boldsymbol{\theta}]$. *How to best reduce the uncertainty* $H[\boldsymbol{\theta}]$*?*

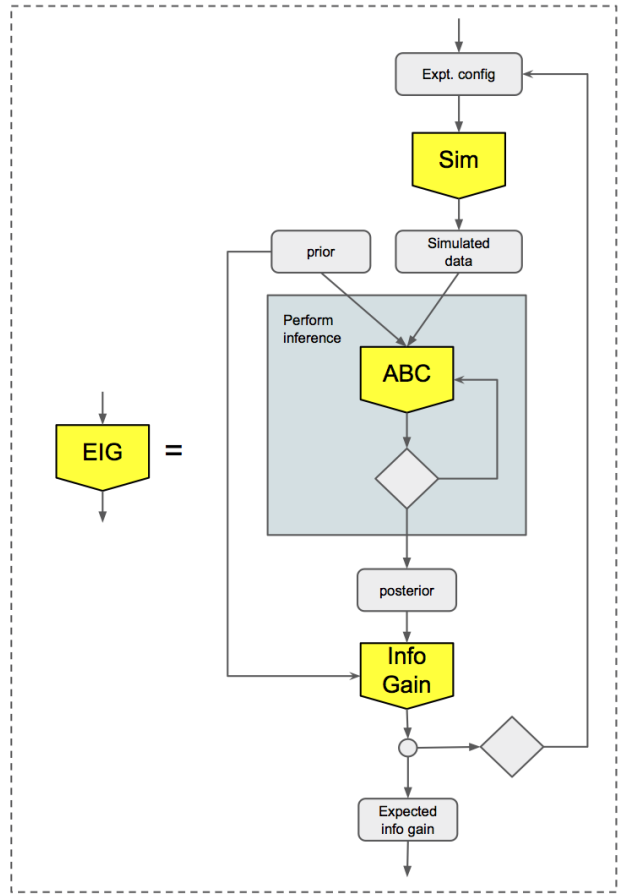
1. Assume an experiment with parameters $\boldsymbol{\phi}$ can be simulated.
2. Simulate the expected improvement $\Delta(\boldsymbol{\phi}) = H[\boldsymbol{\theta}] - \mathbb{E}_{\text{data}|\boldsymbol{\phi}}[H[\boldsymbol{\theta}|\text{data}]]$.
   - This embeds the full likelihood-free inference procedure.
3. Find $\boldsymbol{\phi}^* = \arg\max_{\boldsymbol{\phi}} \Delta(\boldsymbol{\phi})$
   - Computationally (super) heavy.

Credits: cranmer/active_sciencing

Active sciencing

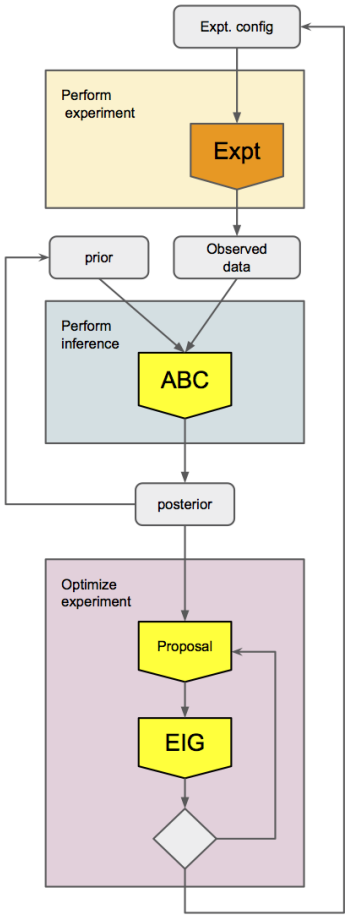


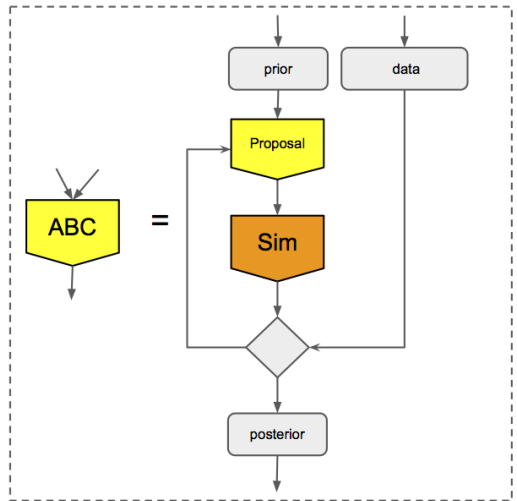


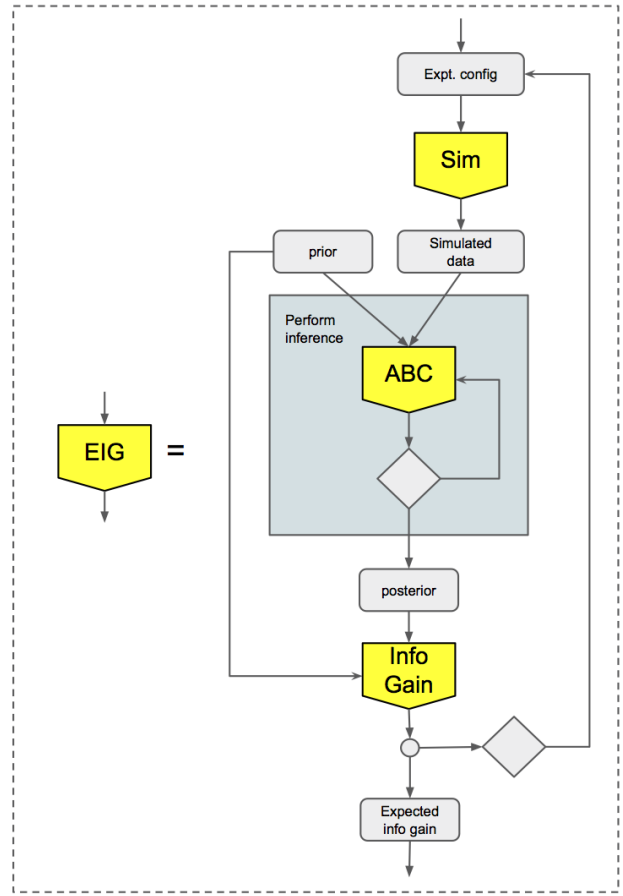


Credits: cranmer/active_sciencing

Active sciencing







Danilo J. Rezende @DeepSpiker · 3m

Replying to @KyleCranmer @glouppe @lukasheinrich_

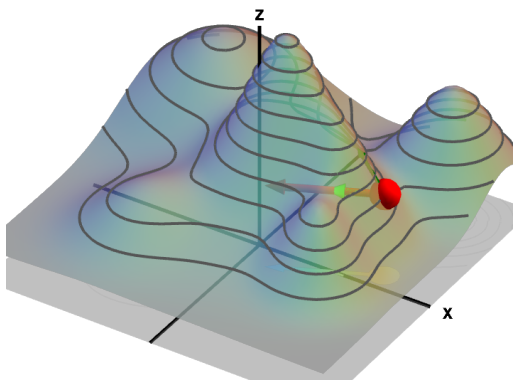
You have the full loop of the scientific method in a python notebook :)

Exploring the theory space

The Standard model admits several extensions.

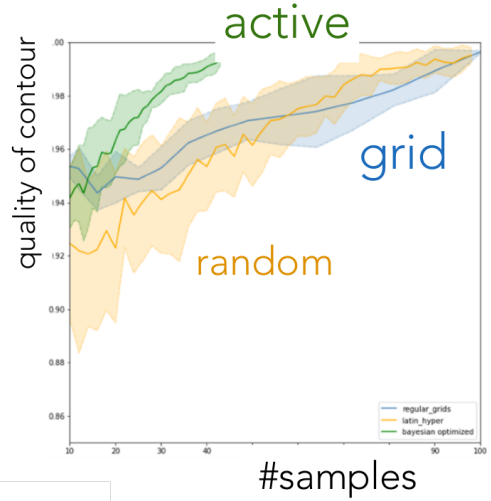
Can we explore the space of theories and find the envelope that agree with the data?

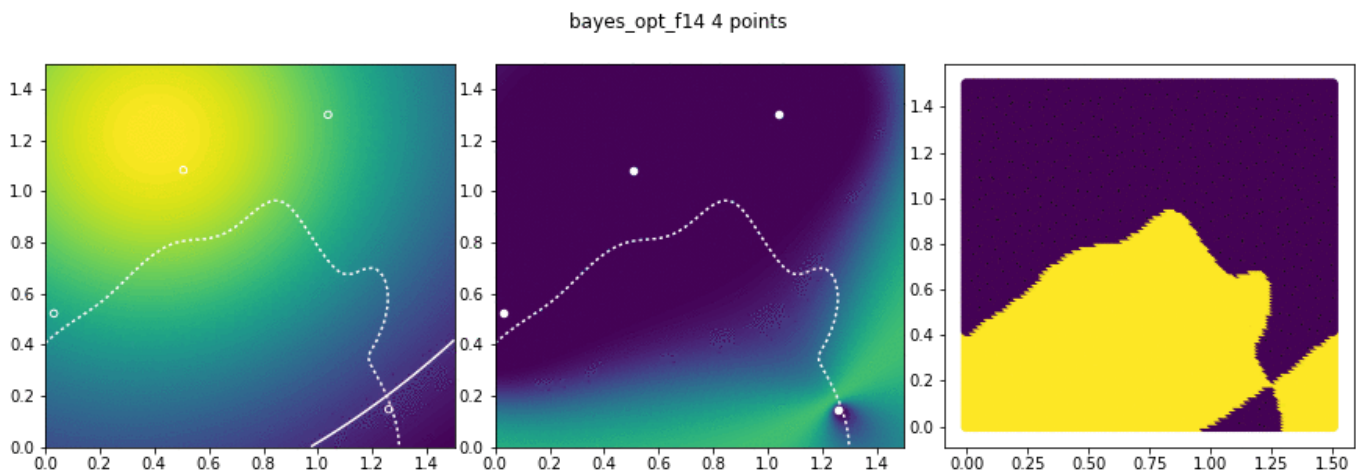
- Assume a generative model of theories, indexed by $\boldsymbol{\psi}$.
- Assume the experiment design $\boldsymbol{\phi}$ is fixed.
- Find $\{\boldsymbol{\psi} | \rho(p_r(\mathbf{x}|\boldsymbol{\phi}), p(\mathbf{x}|\boldsymbol{\psi}, \boldsymbol{\phi}, \boldsymbol{\theta}^*)) < \epsilon\}$.



Finding exclusion contours

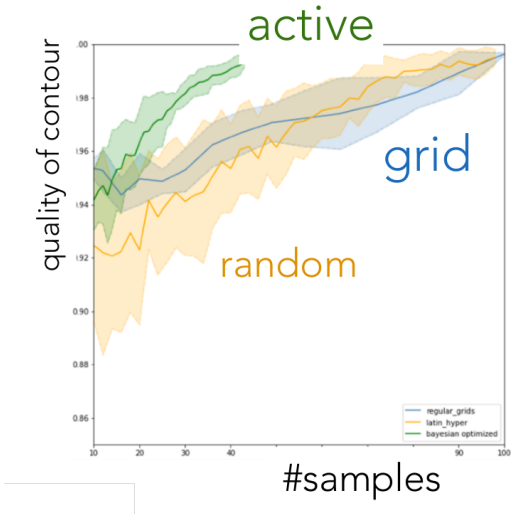
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

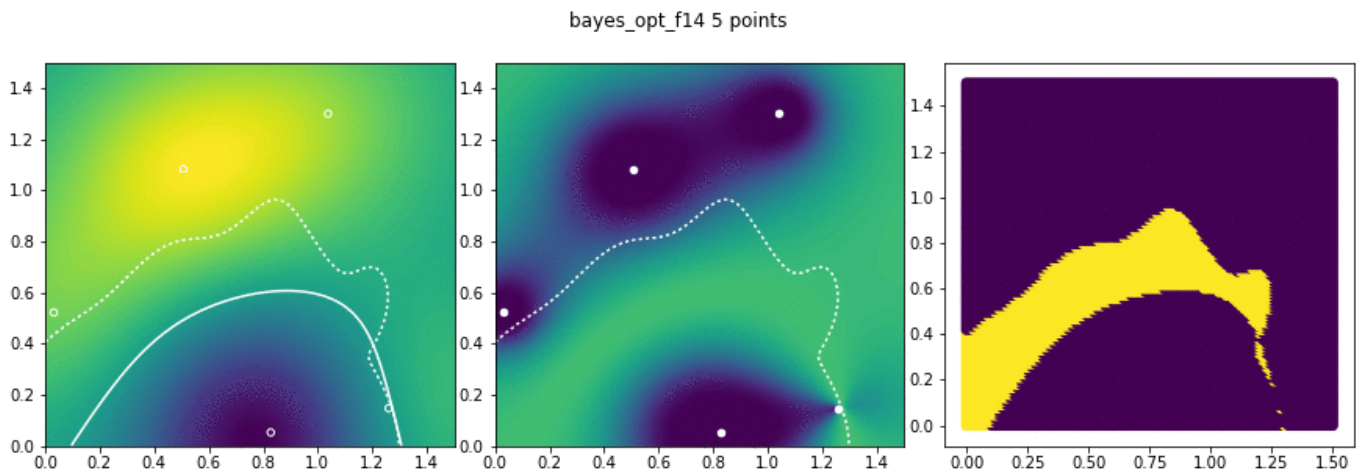




Finding exclusion contours

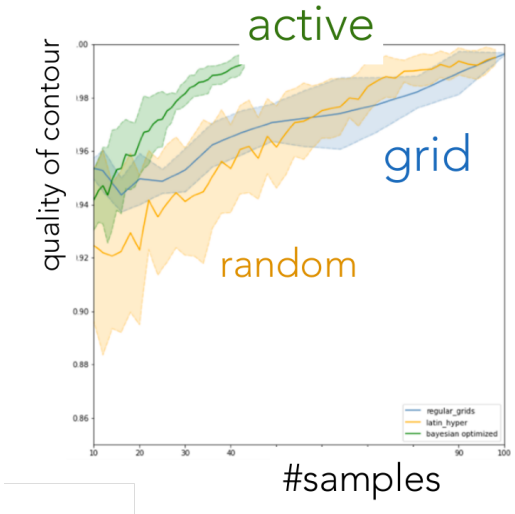
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

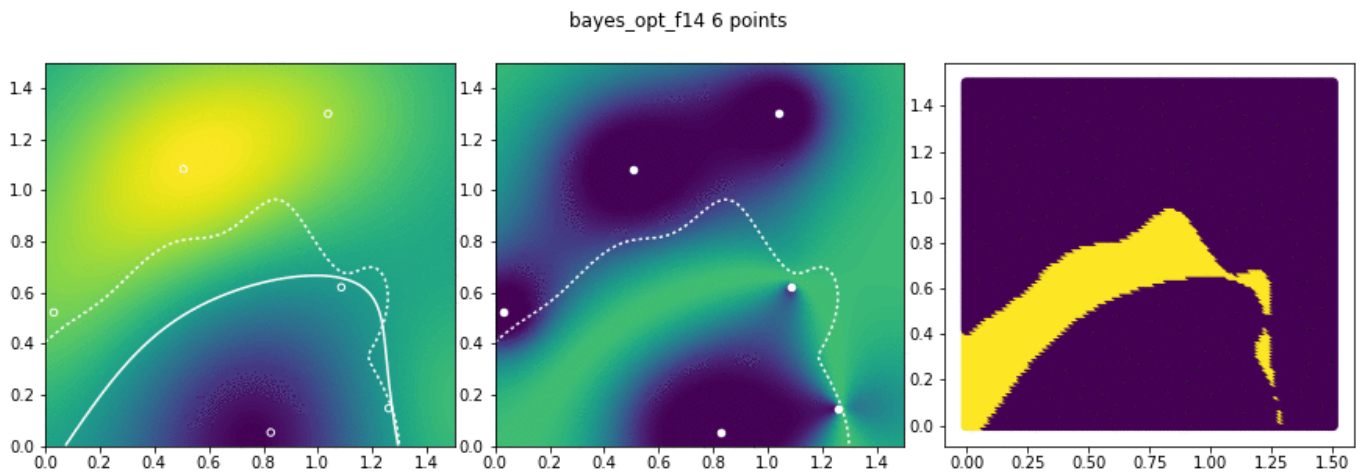




Finding exclusion contours

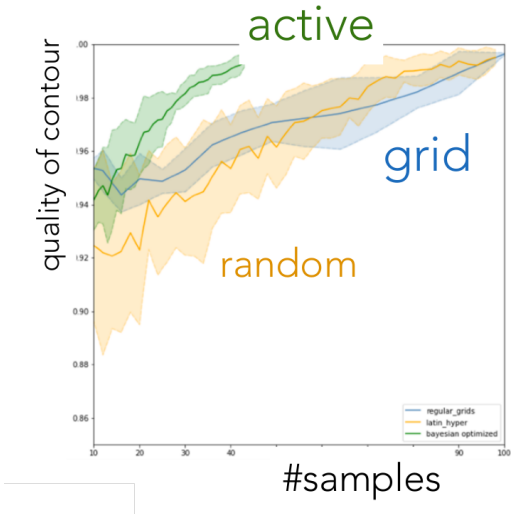
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

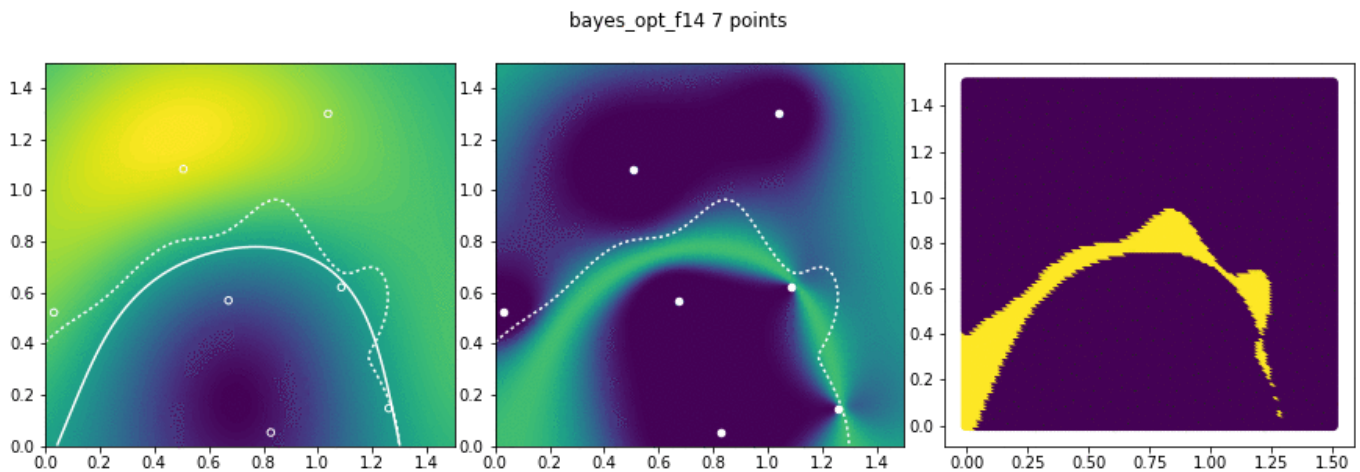




Finding exclusion contours

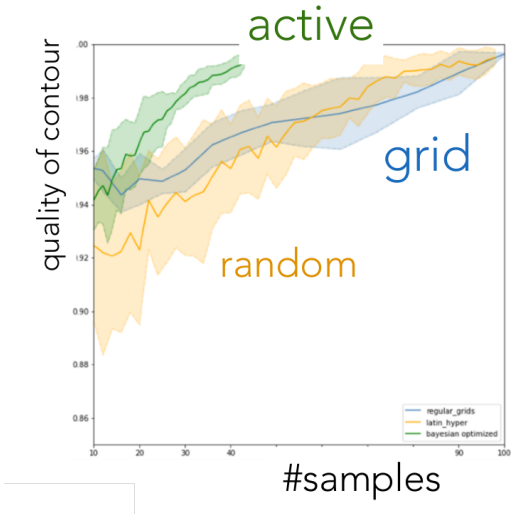
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

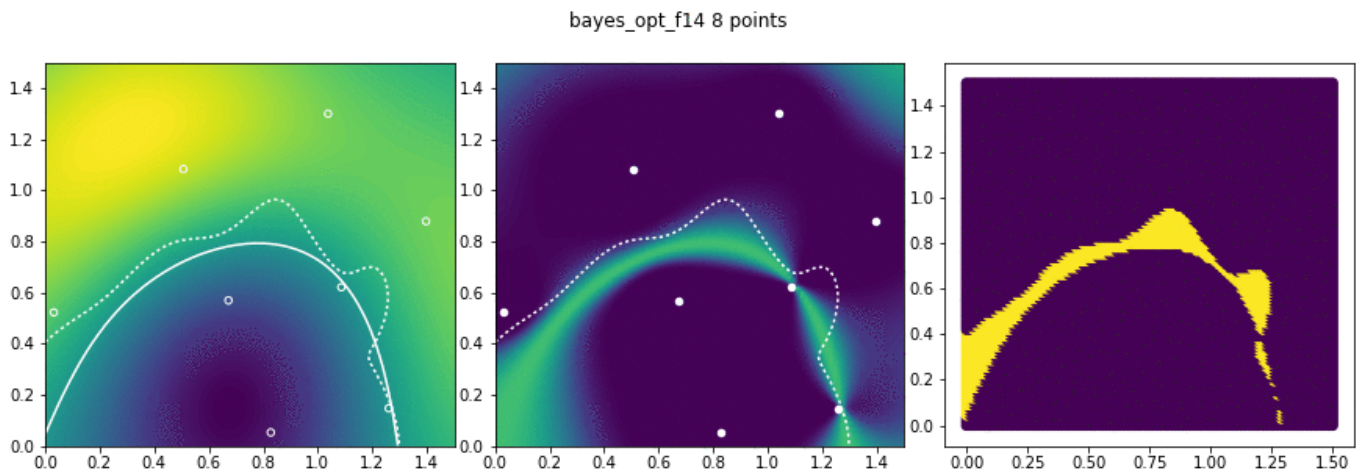




Finding exclusion contours

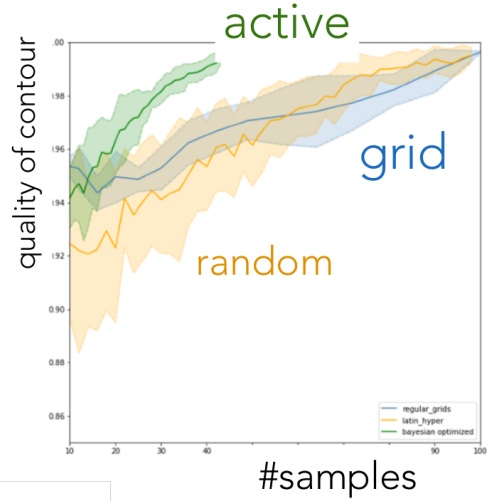
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

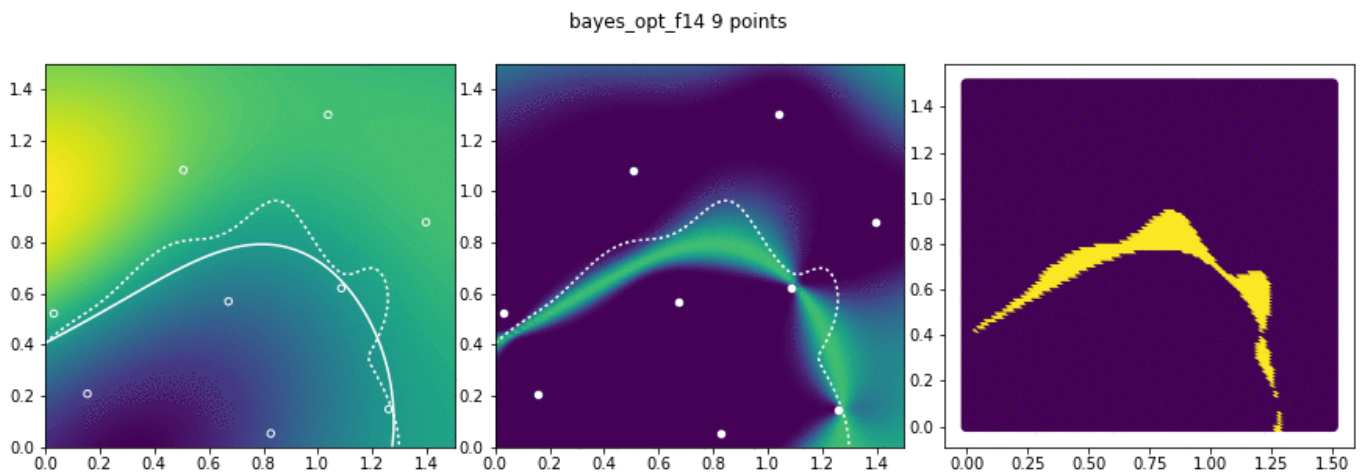




Finding exclusion contours

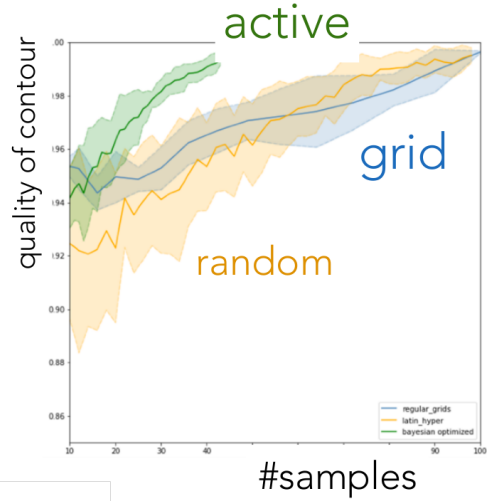
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

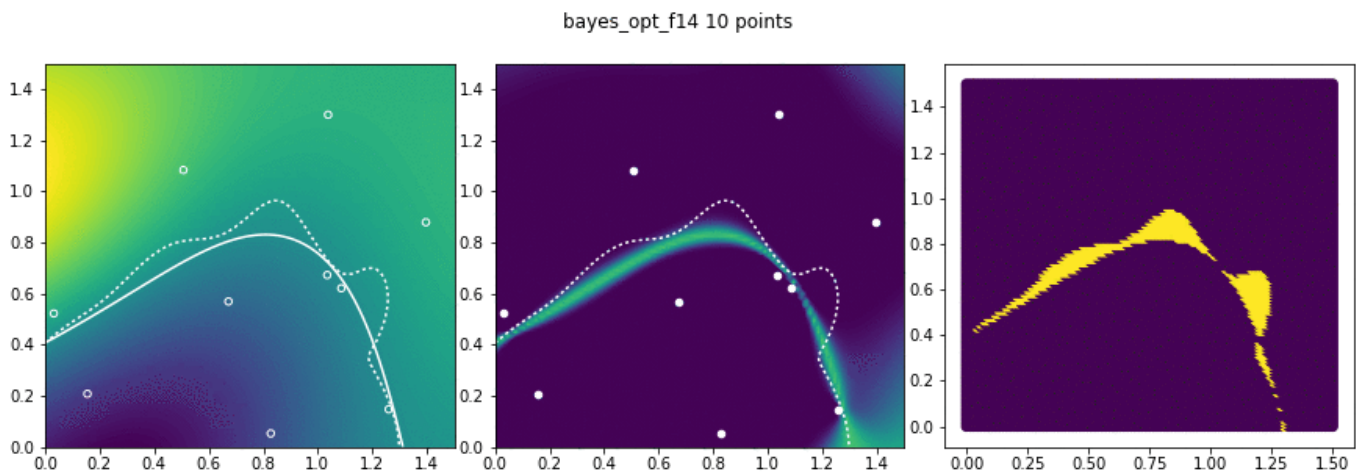




Finding exclusion contours

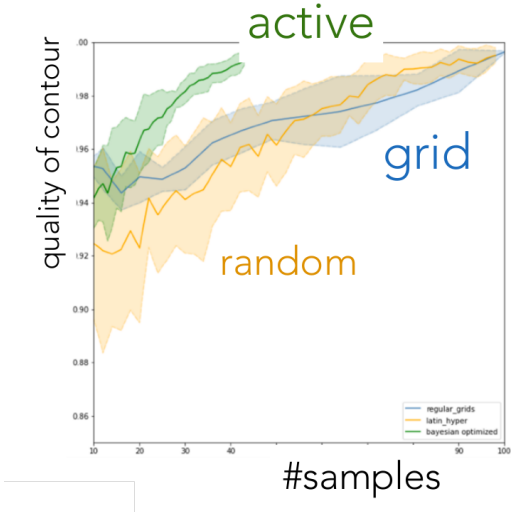
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

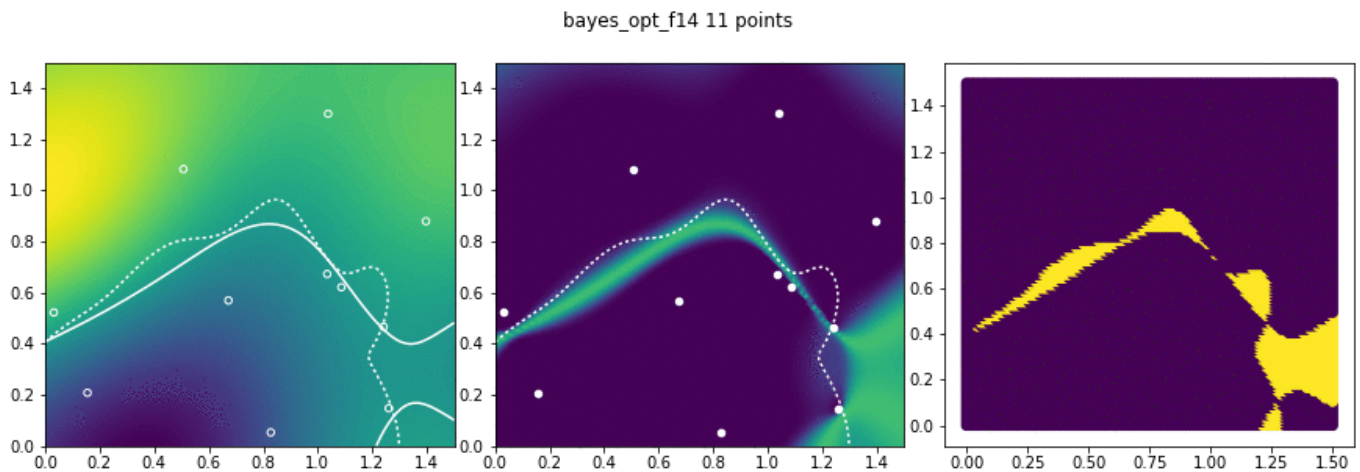




Finding exclusion contours

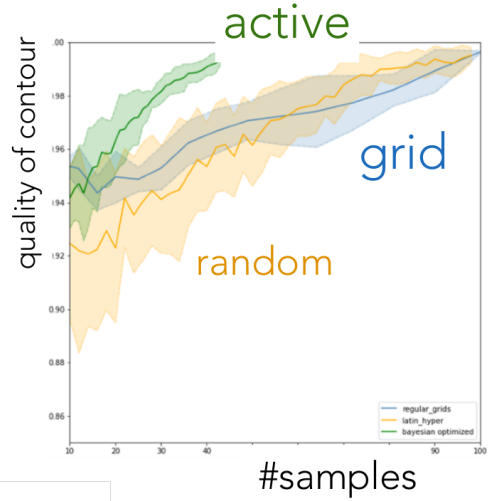
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

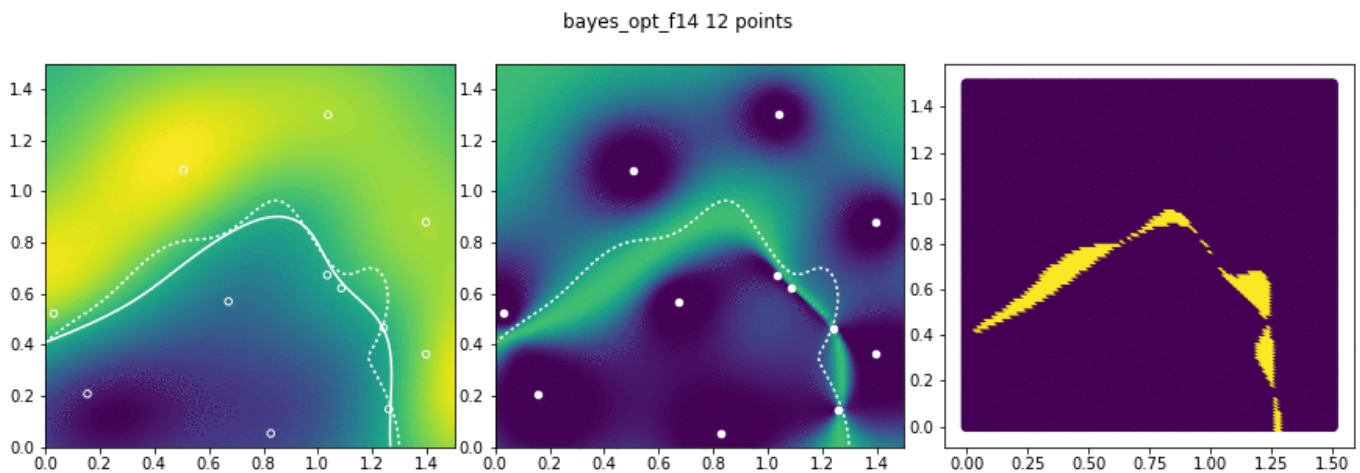




Finding exclusion contours

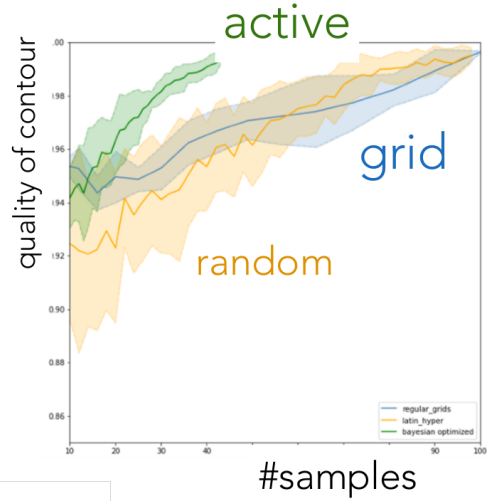
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

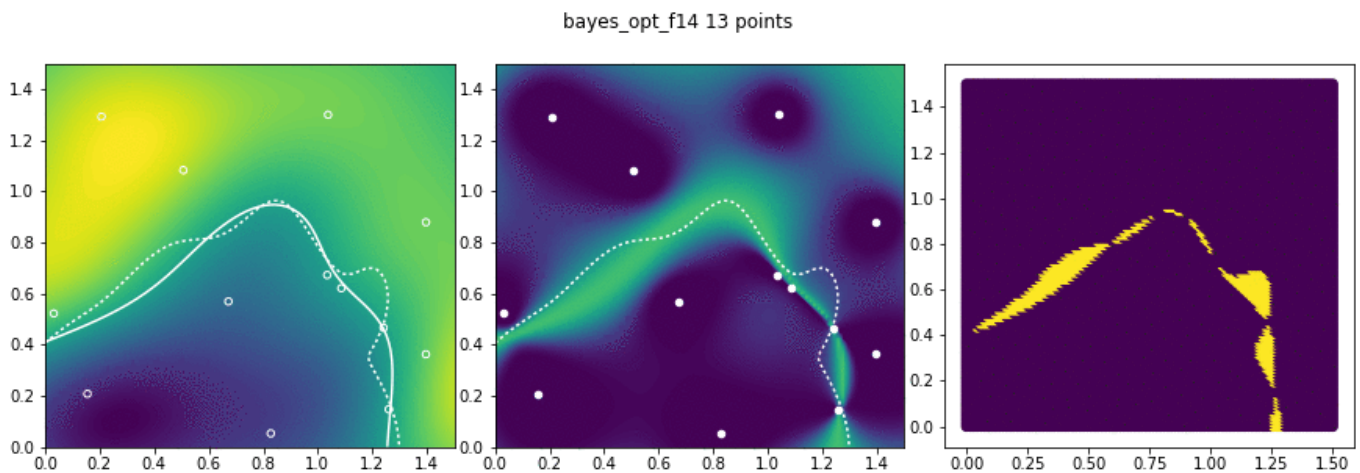




Finding exclusion contours

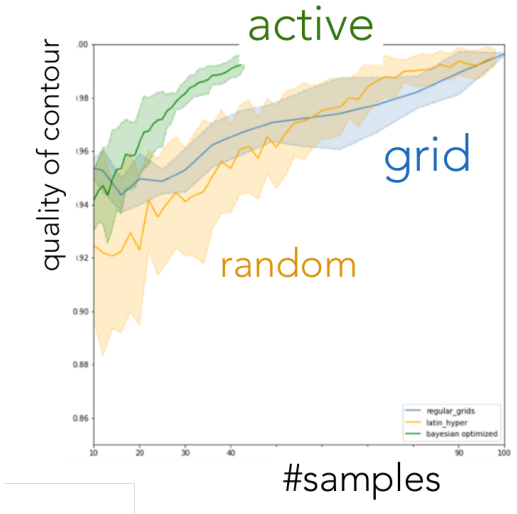
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

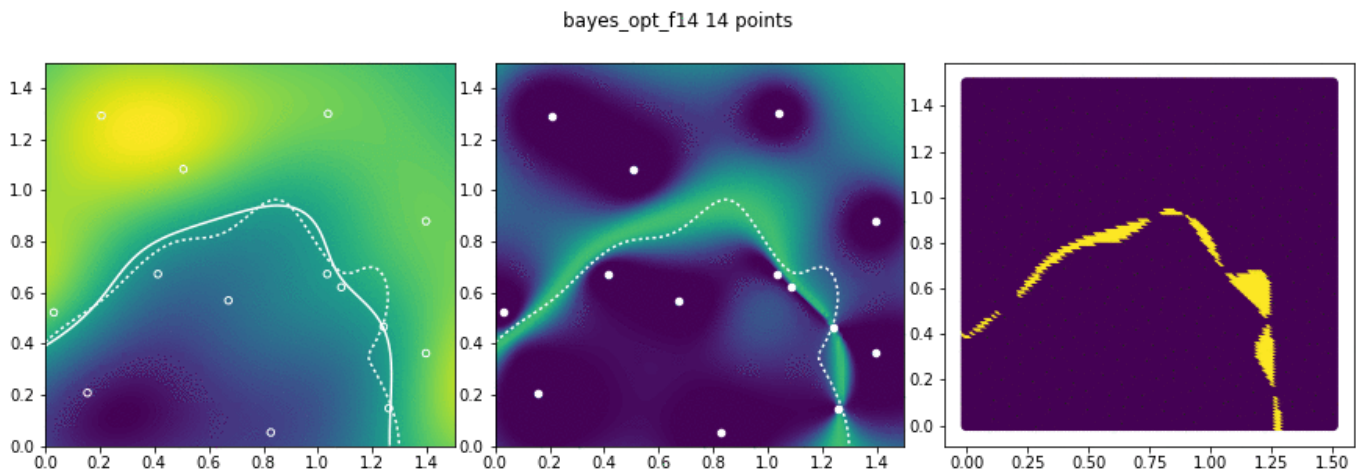




Finding exclusion contours

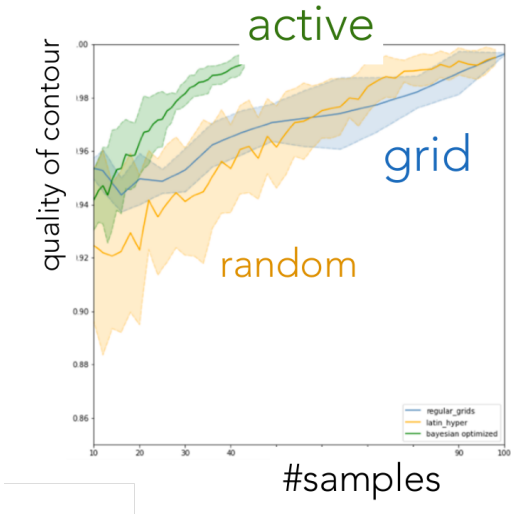
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

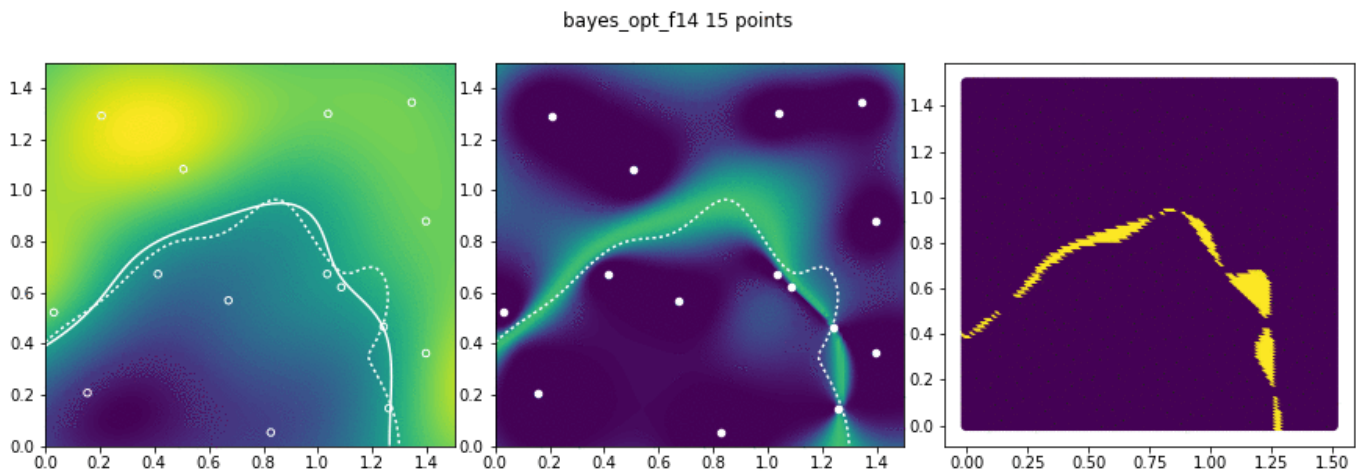




Finding exclusion contours

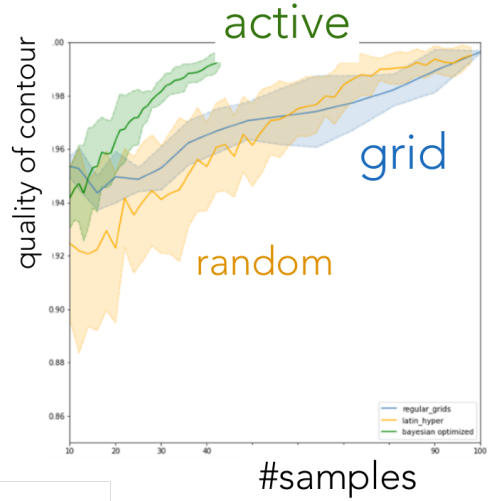
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

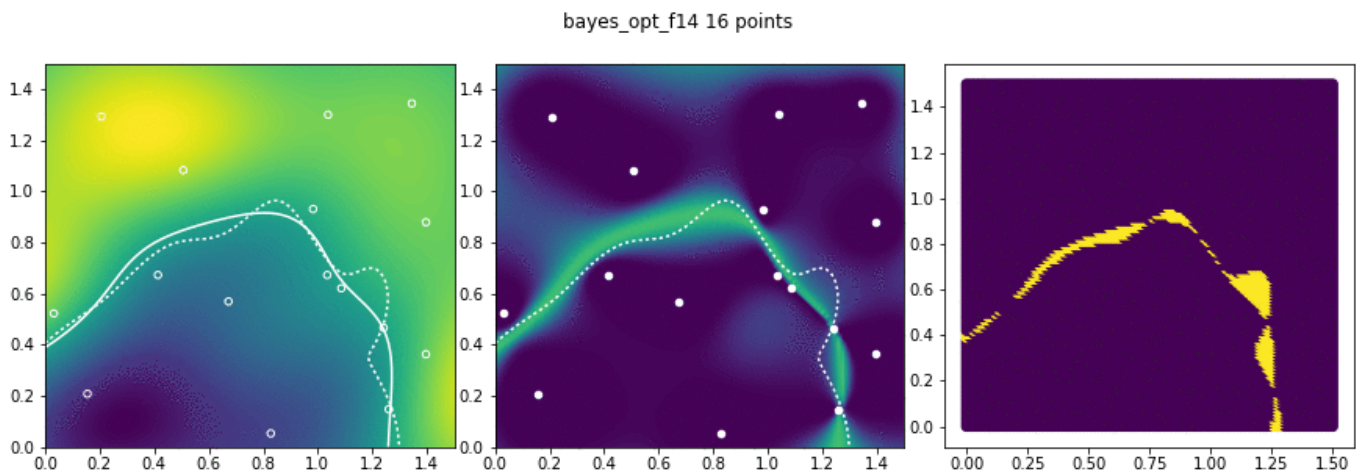




Finding exclusion contours

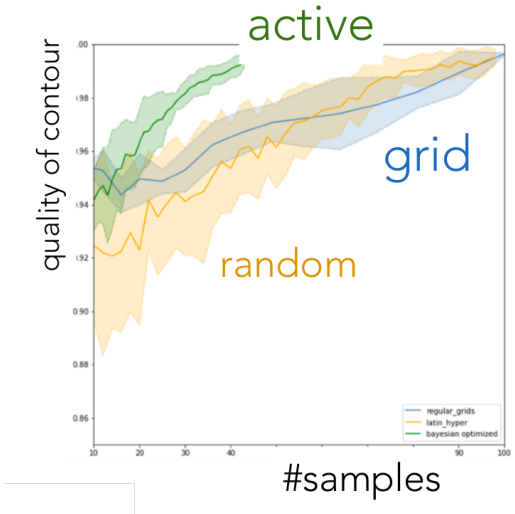
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

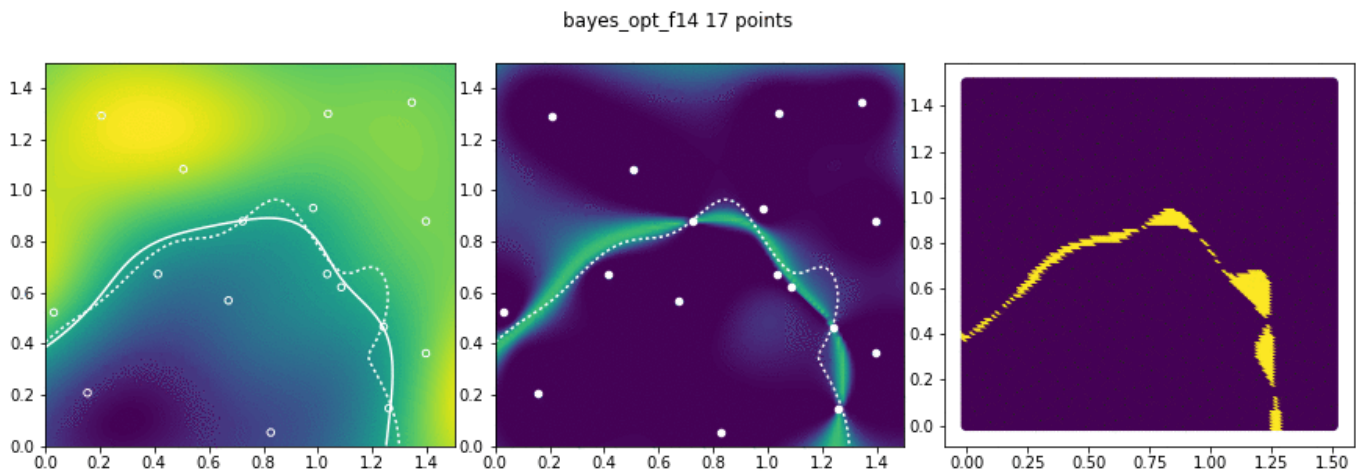




Finding exclusion contours

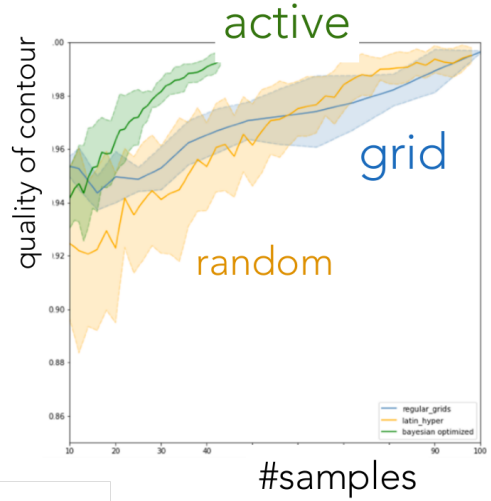
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

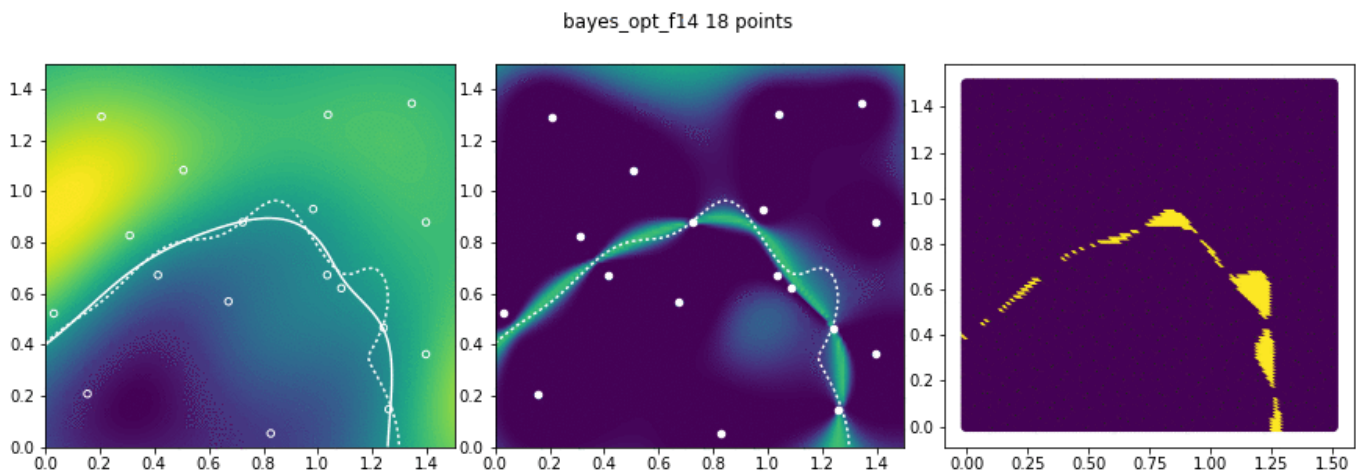




Finding exclusion contours

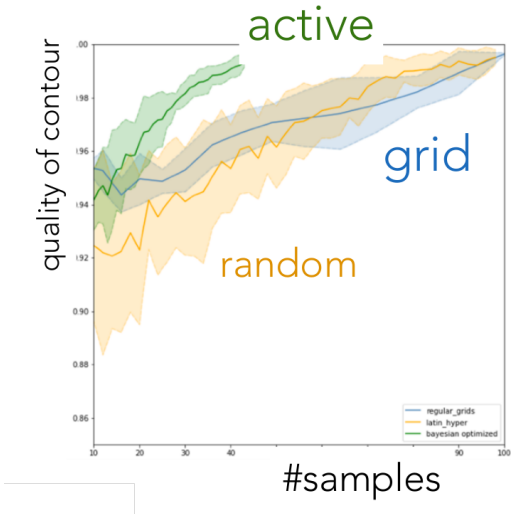
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

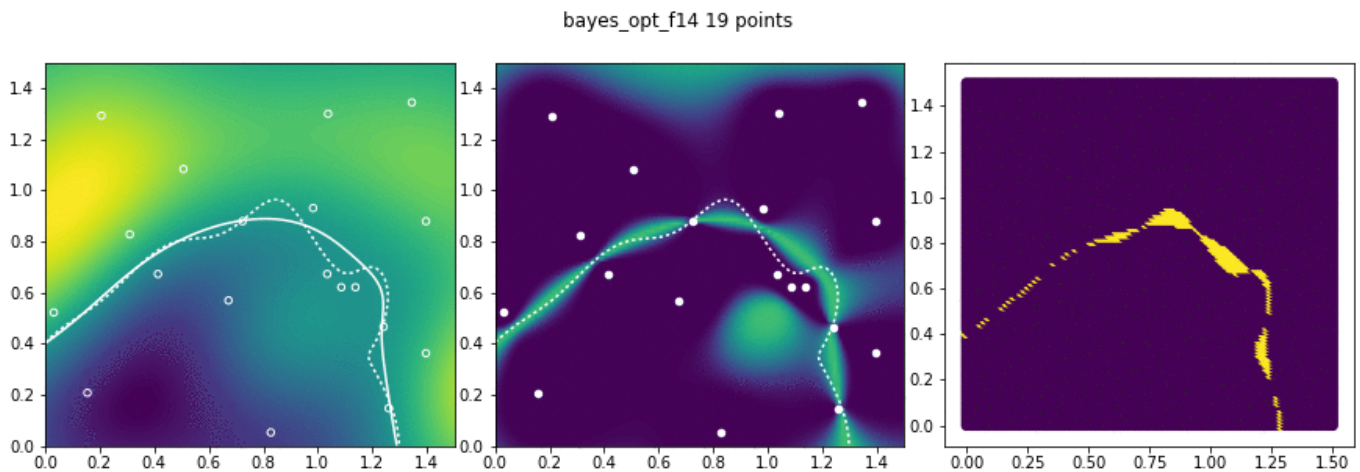




Finding exclusion contours

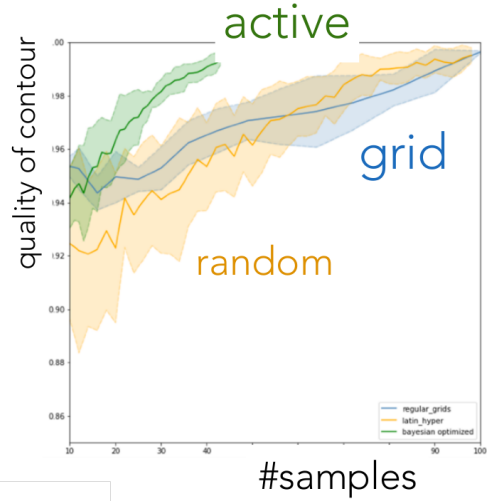
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

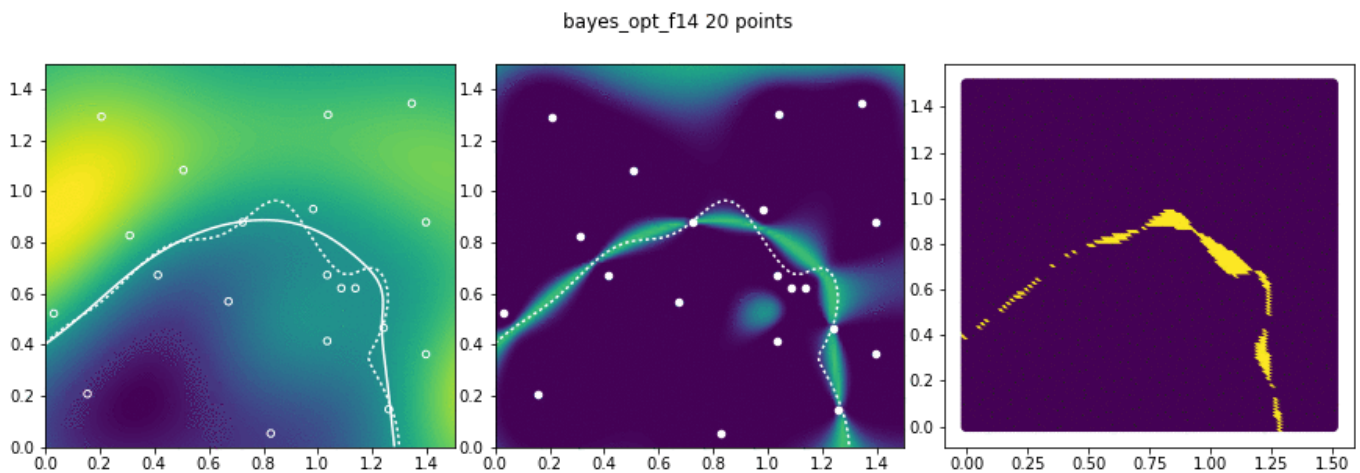




Finding exclusion contours

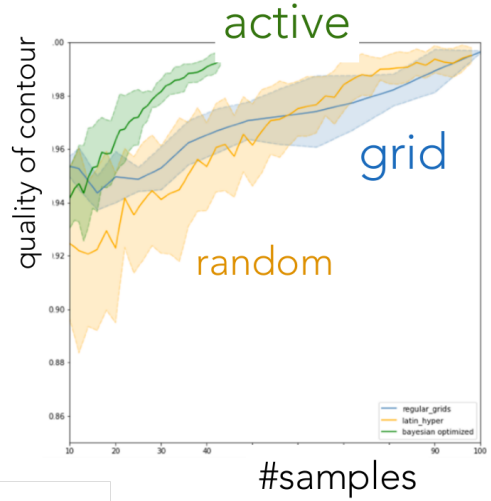
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

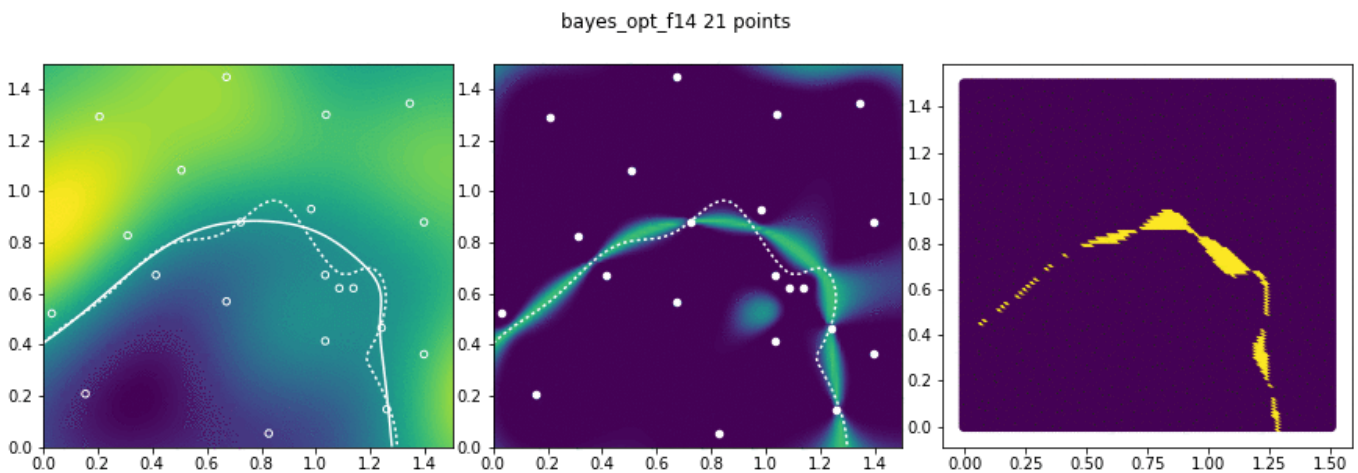




Finding exclusion contours

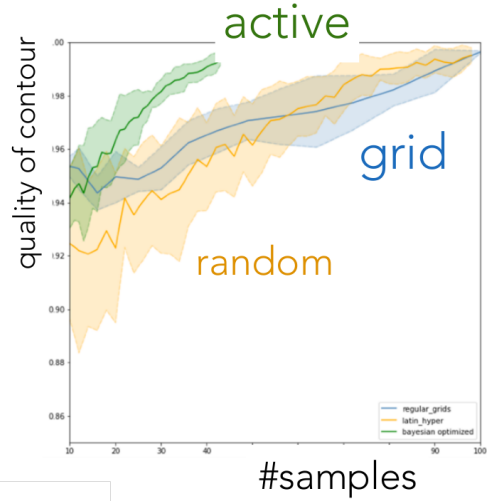
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

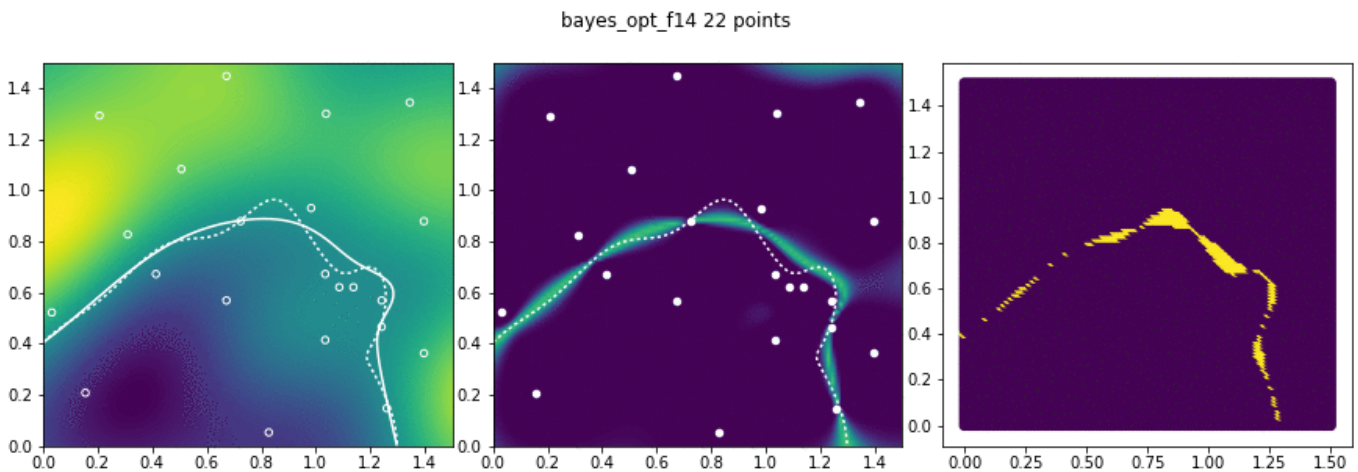




Finding exclusion contours

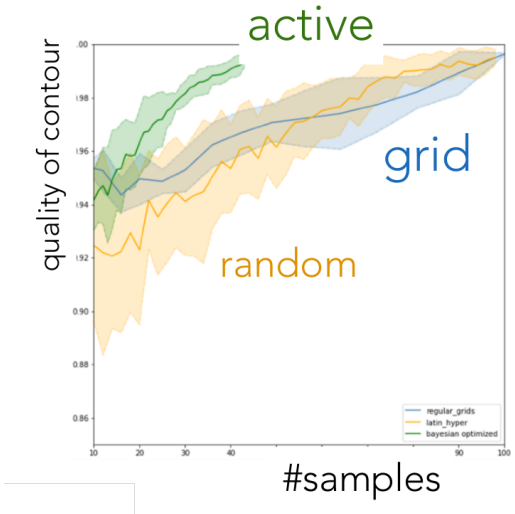
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

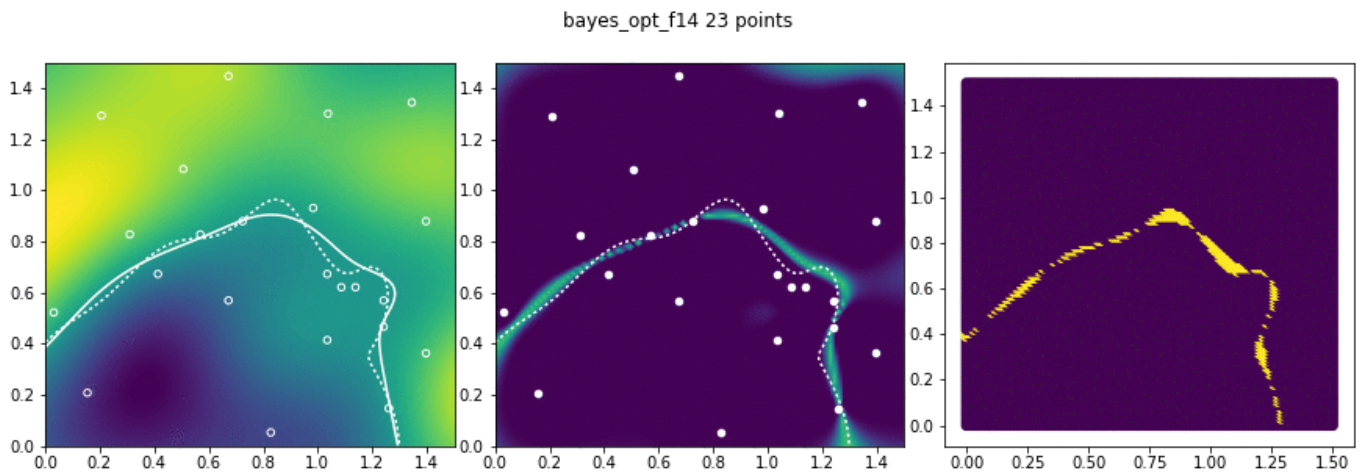




Finding exclusion contours

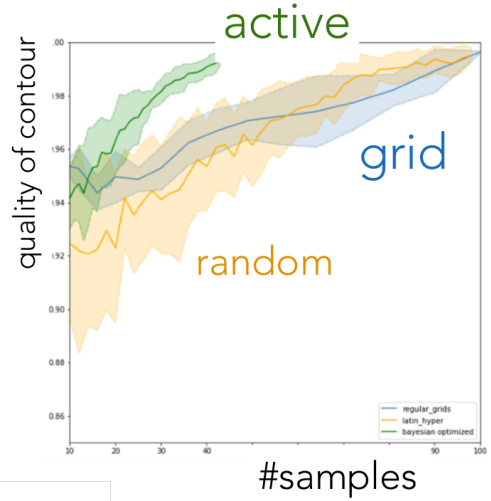
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

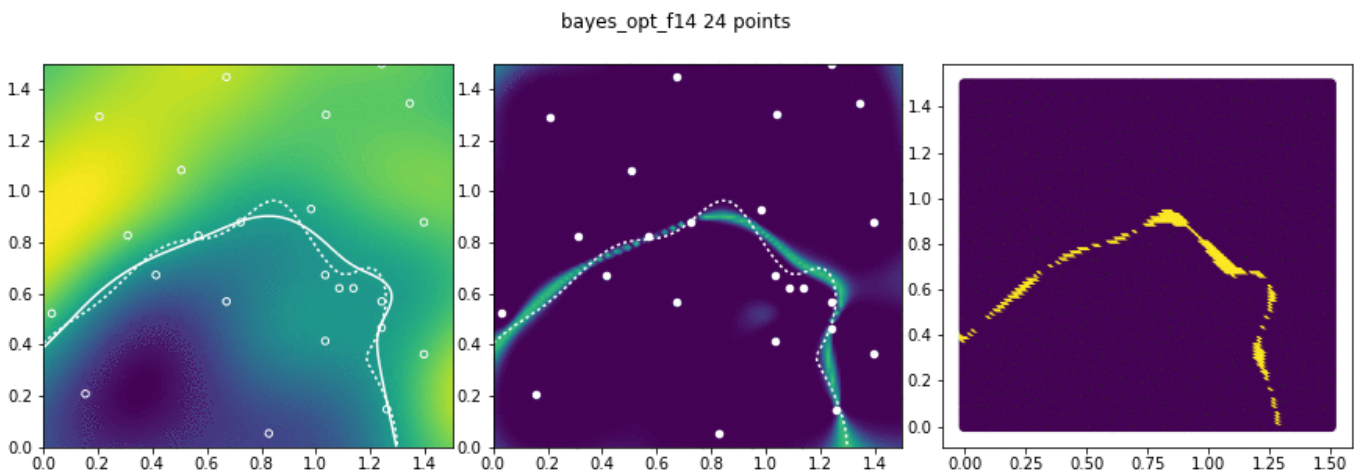




Finding exclusion contours

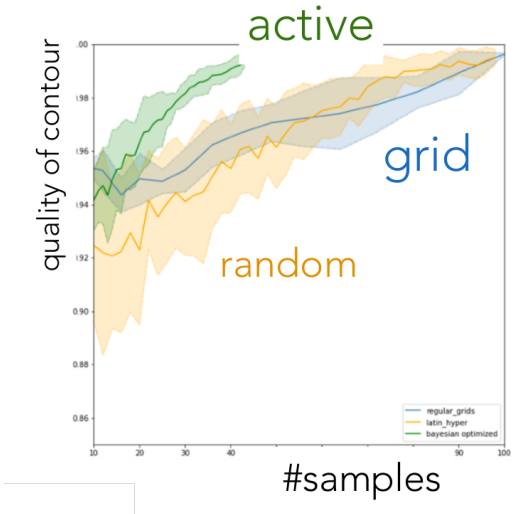
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

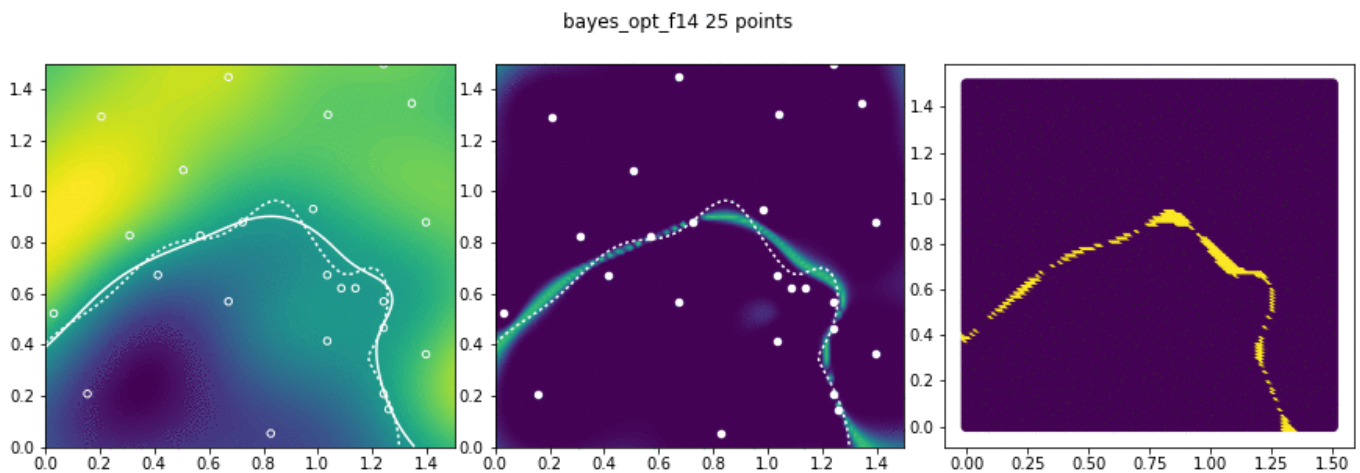




Finding exclusion contours

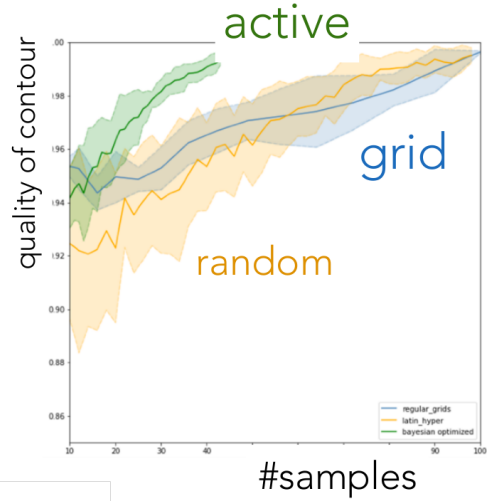
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

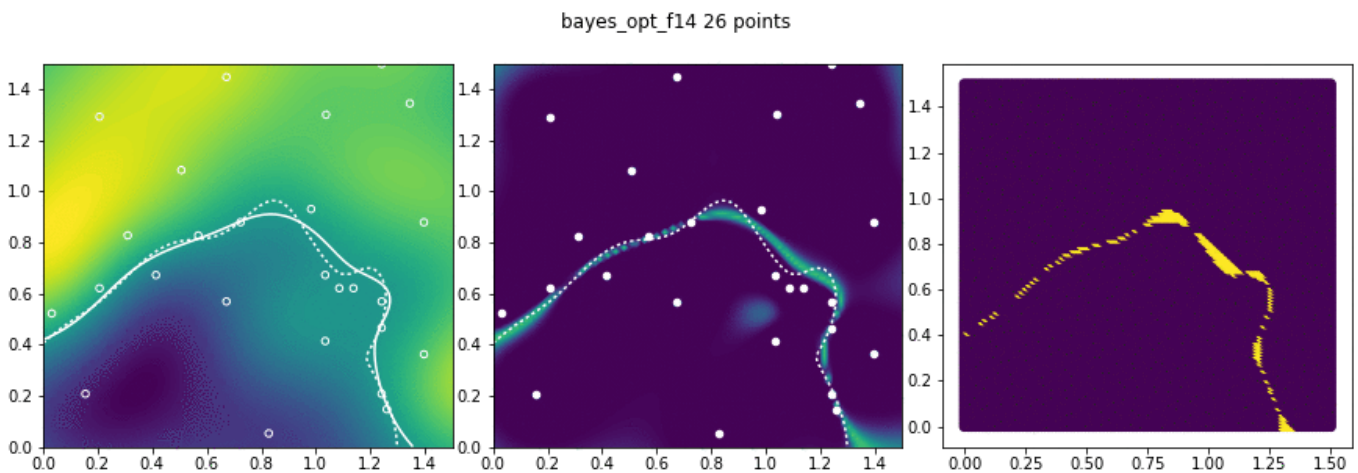




Finding exclusion contours

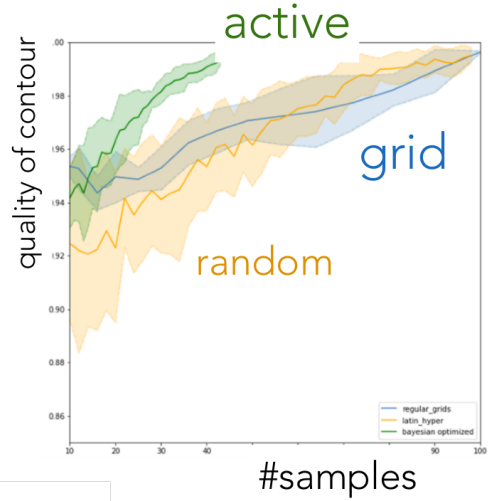
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

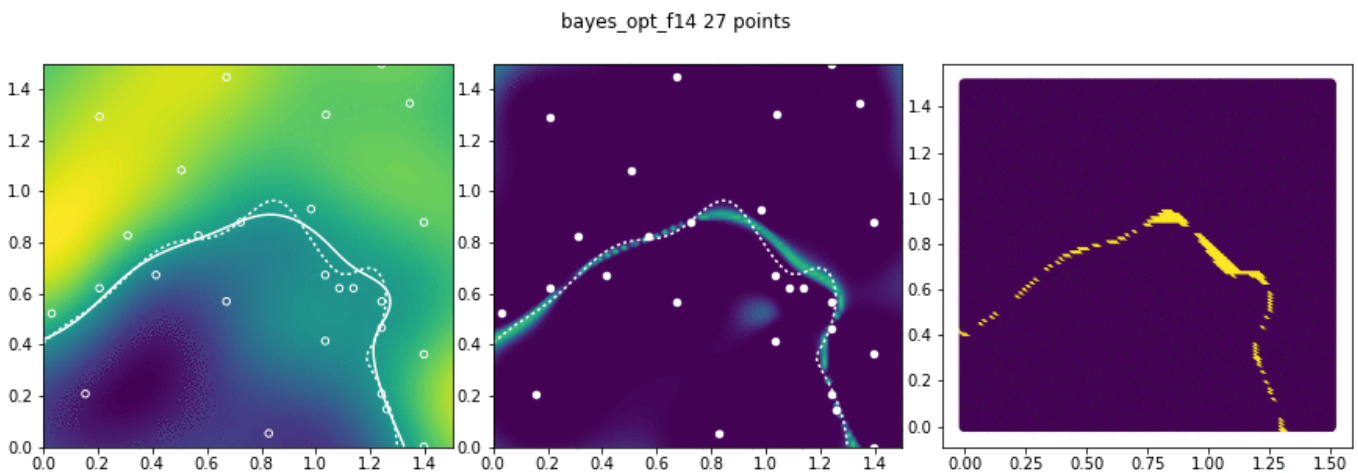




Finding exclusion contours

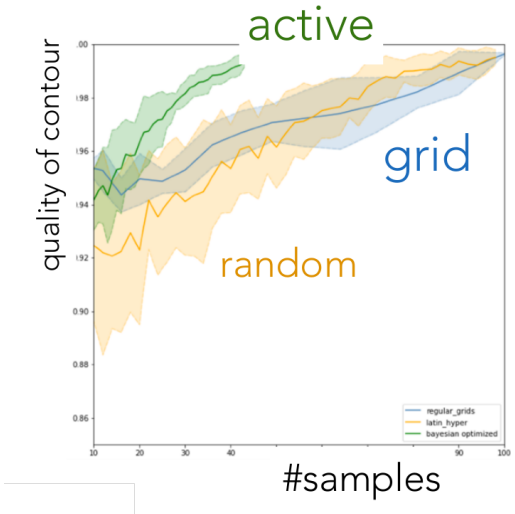
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

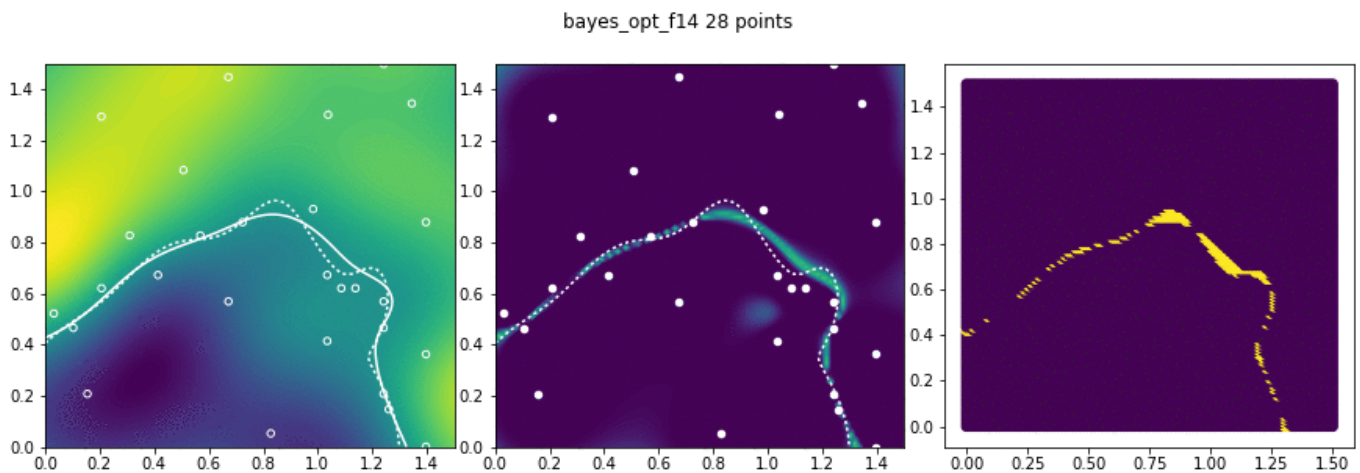




Finding exclusion contours

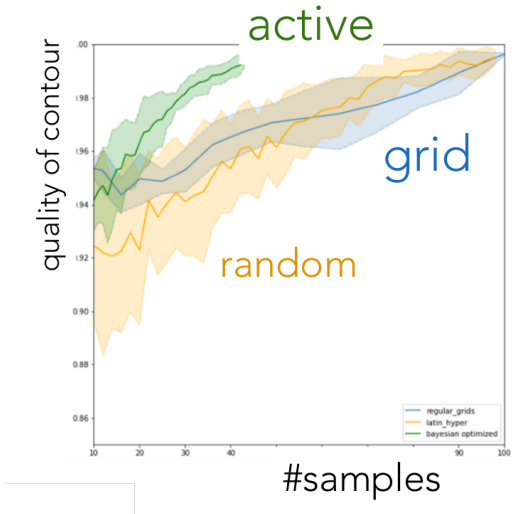
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

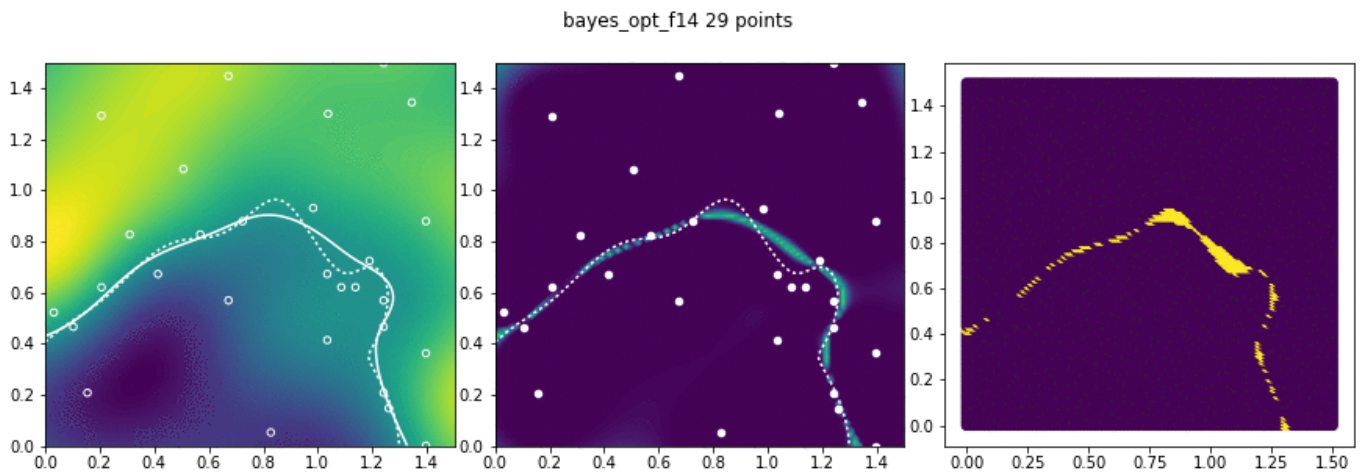




Finding exclusion contours

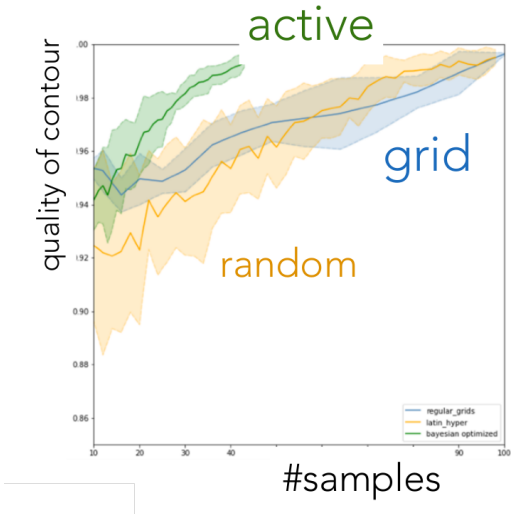
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

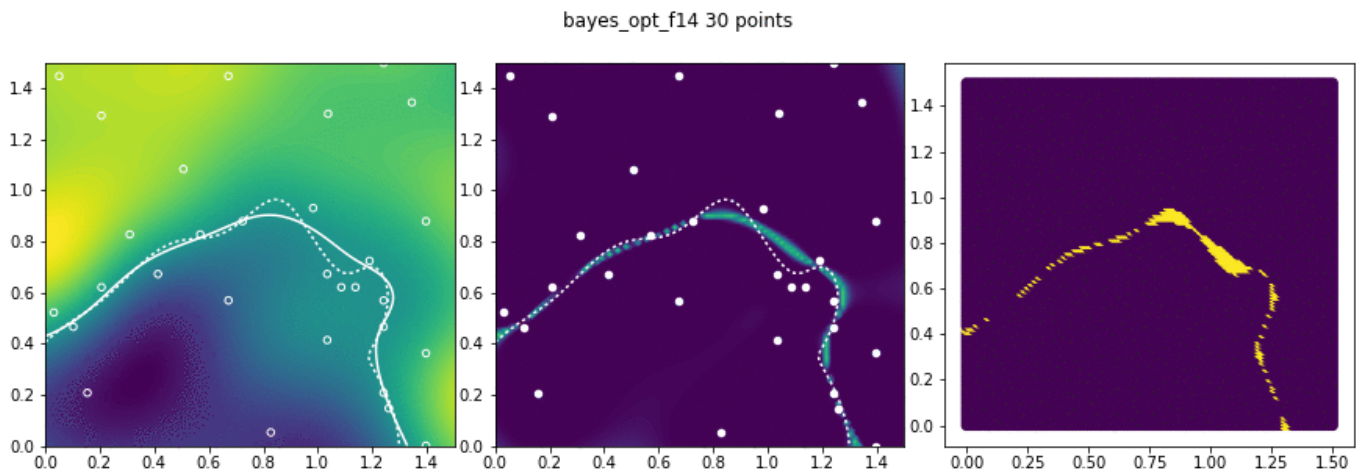




Finding exclusion contours

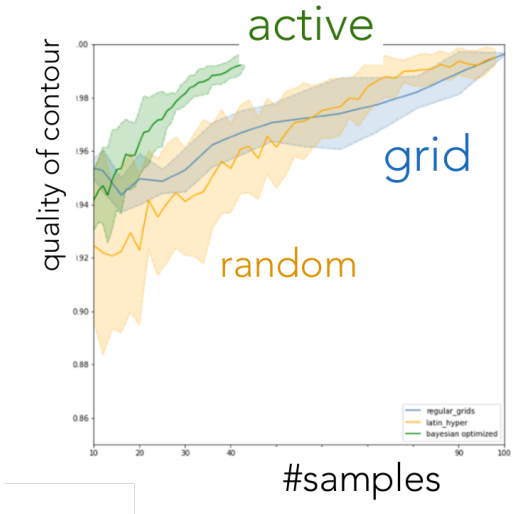
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

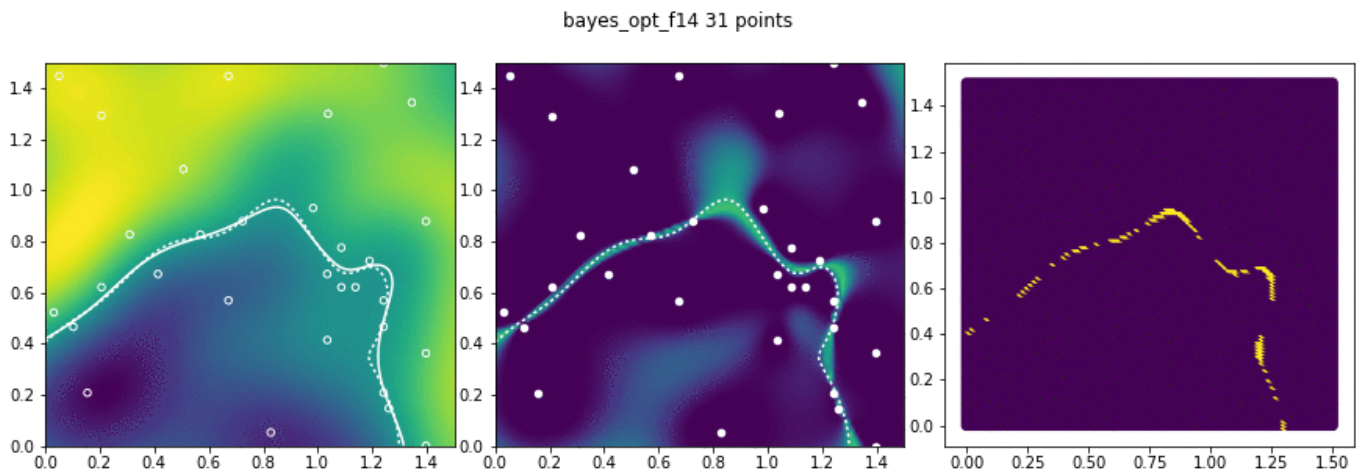




Finding exclusion contours

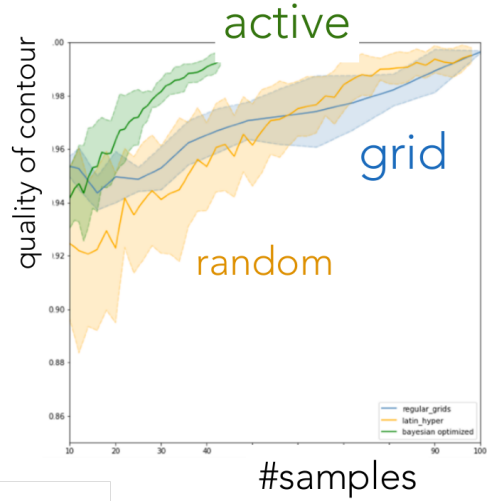
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

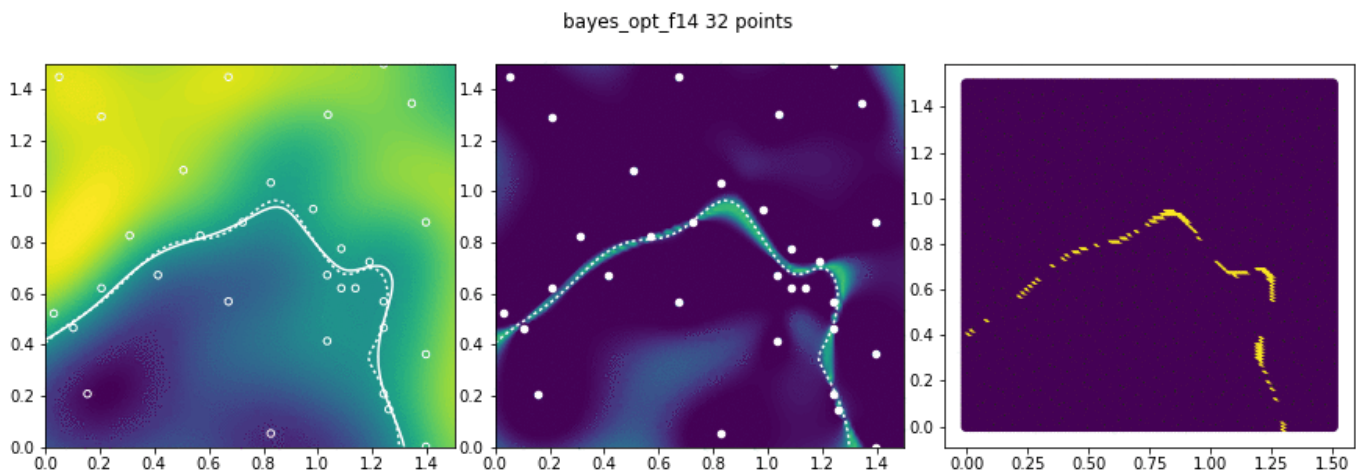




Finding exclusion contours

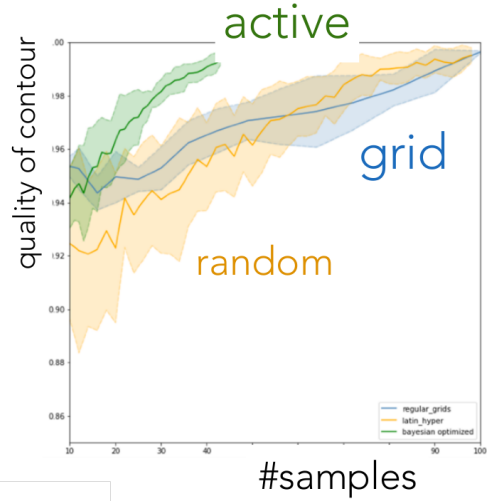
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

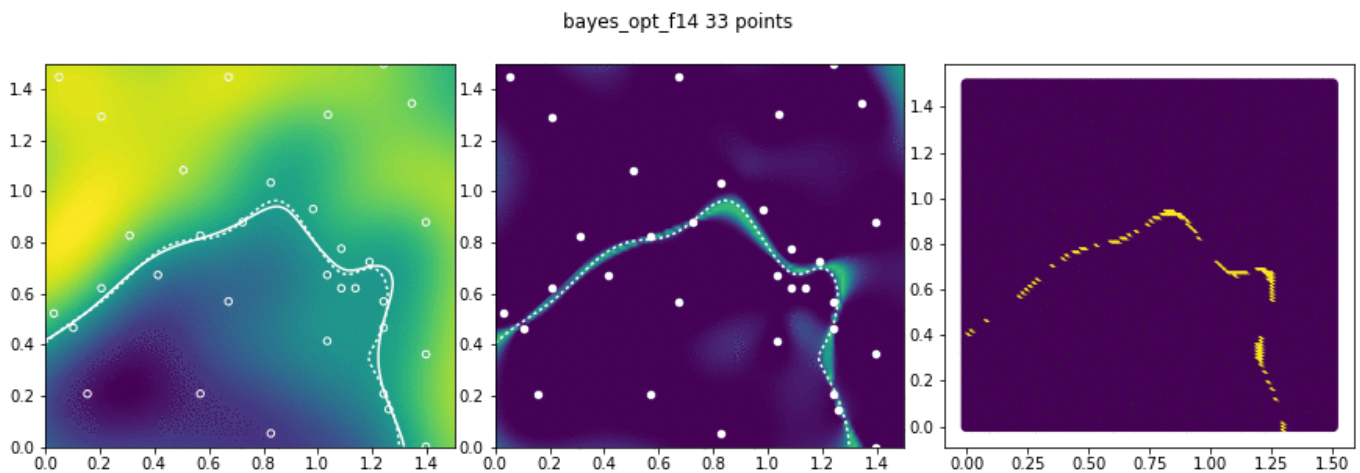




Finding exclusion contours

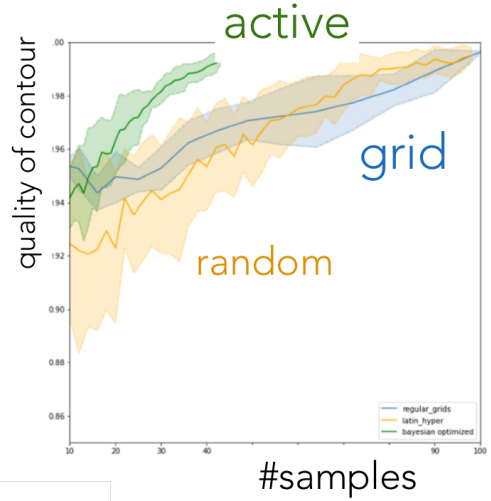
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

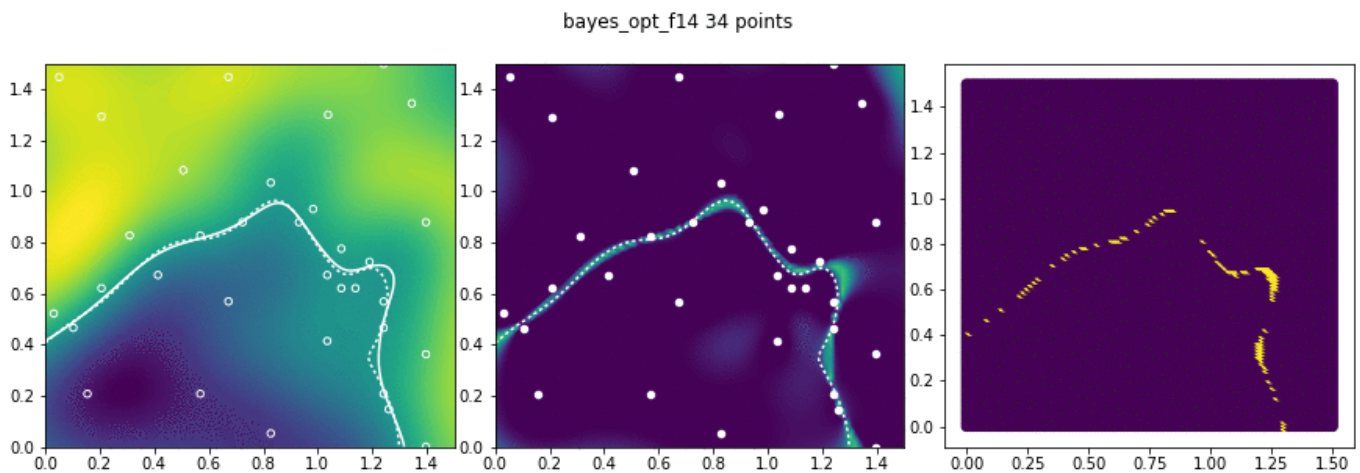




Finding exclusion contours

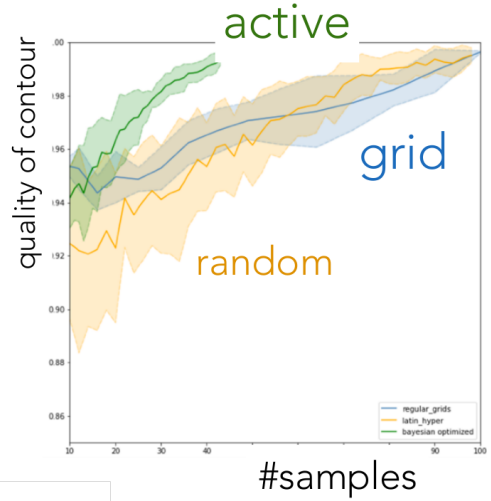
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

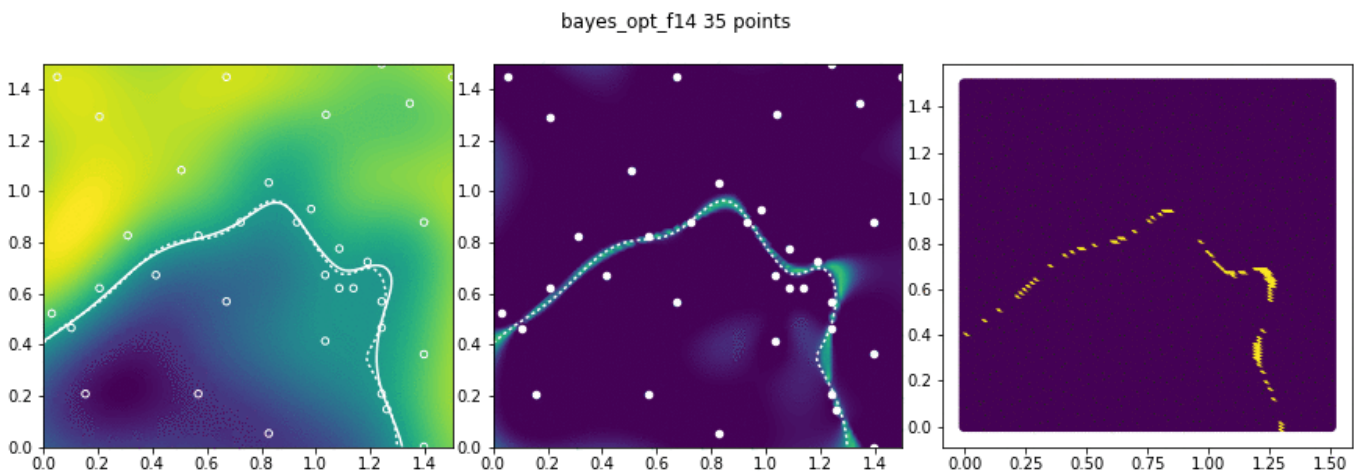




Finding exclusion contours

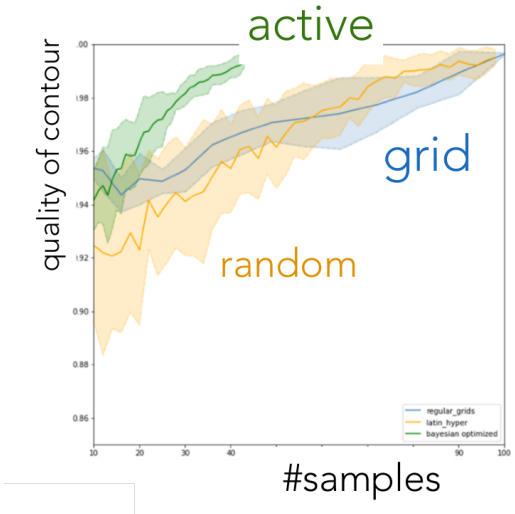
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

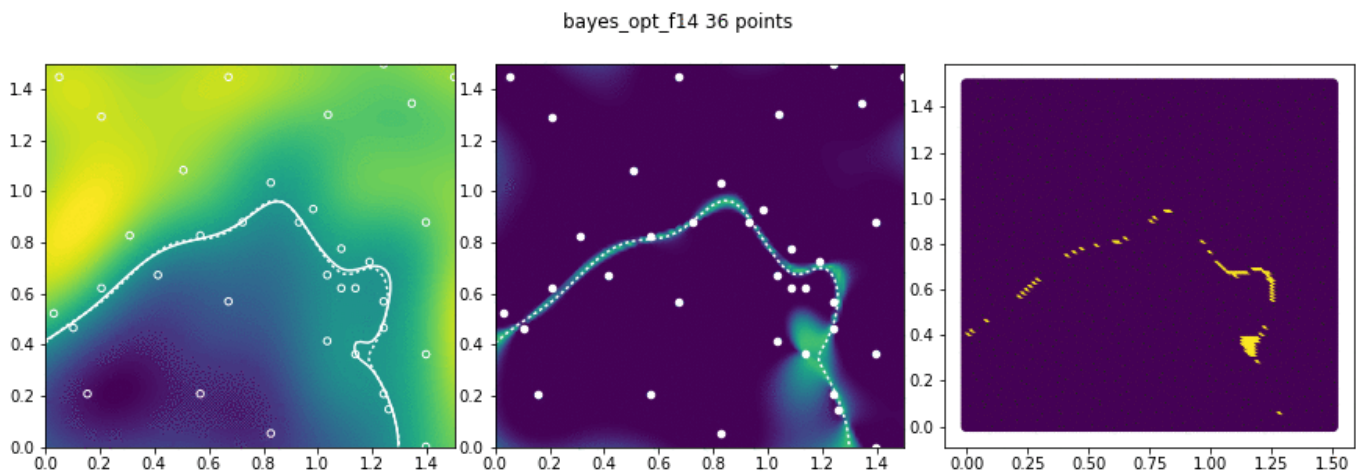




Finding exclusion contours

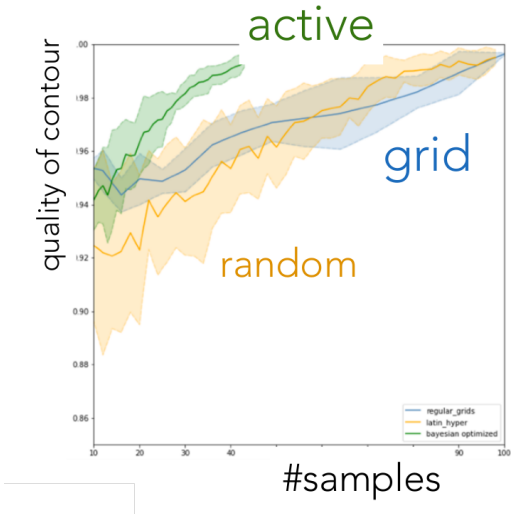
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

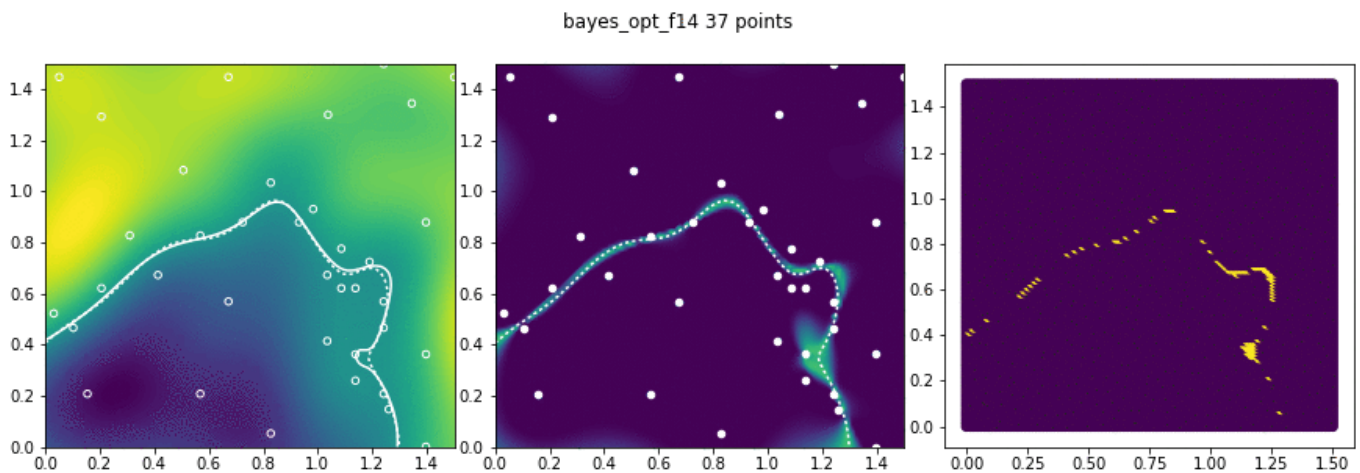




Finding exclusion contours

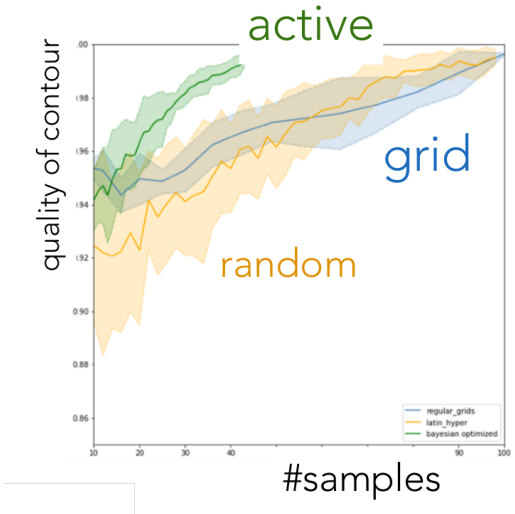
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

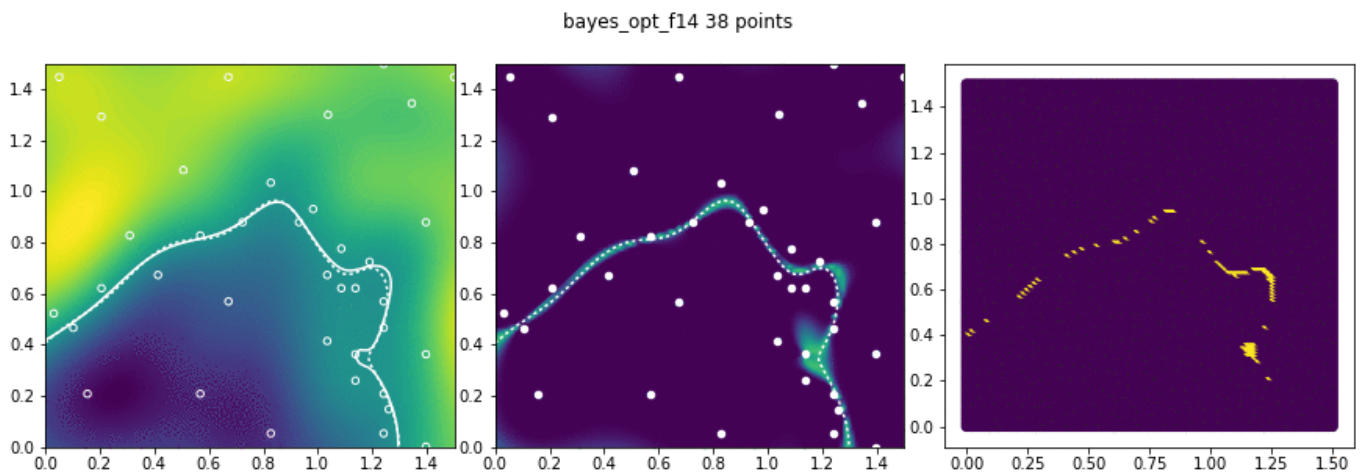




Finding exclusion contours

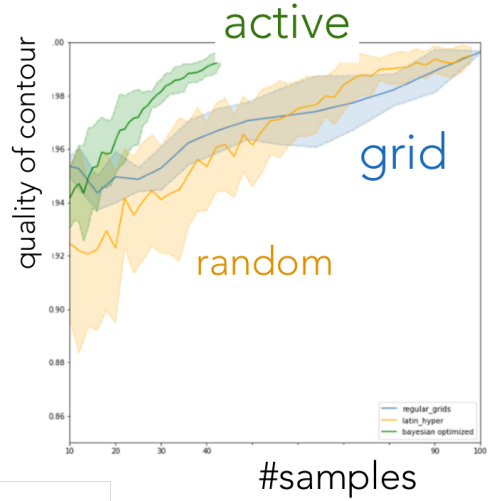
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

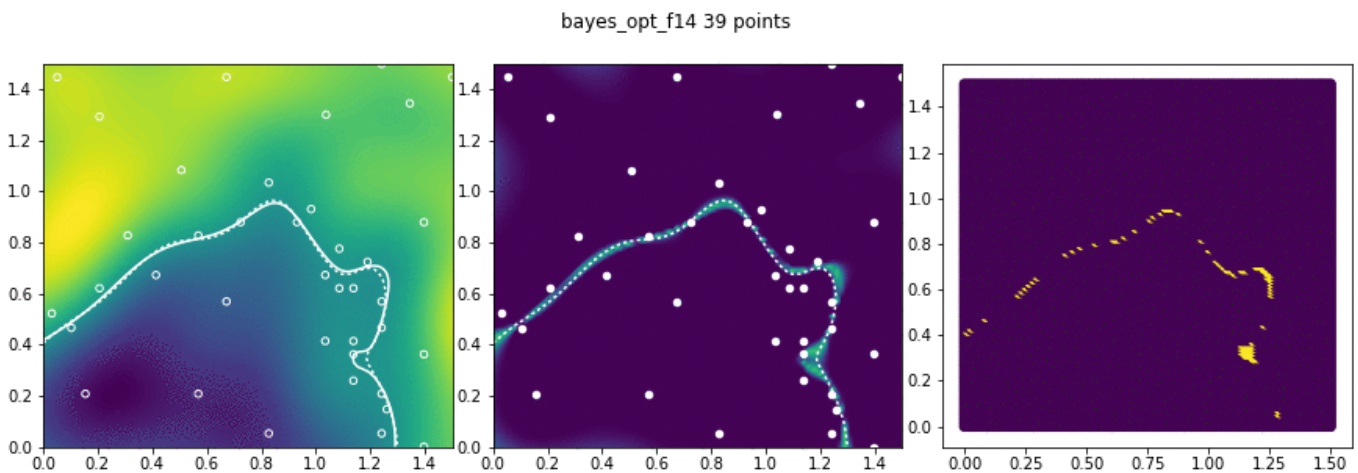




Finding exclusion contours

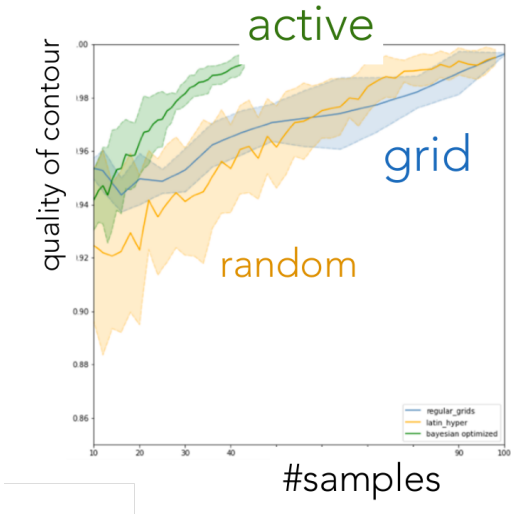
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

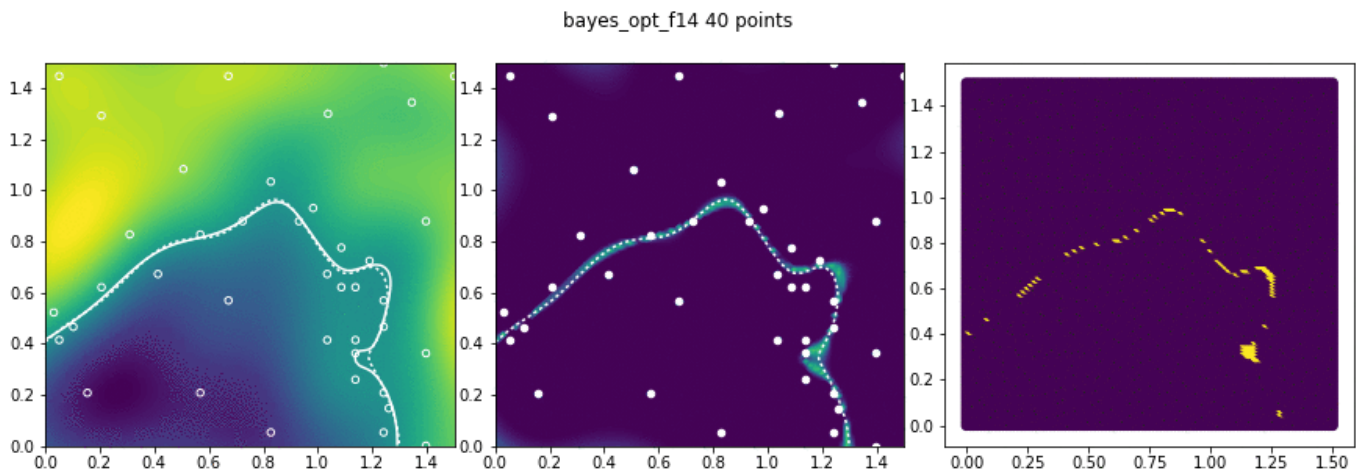




Finding exclusion contours

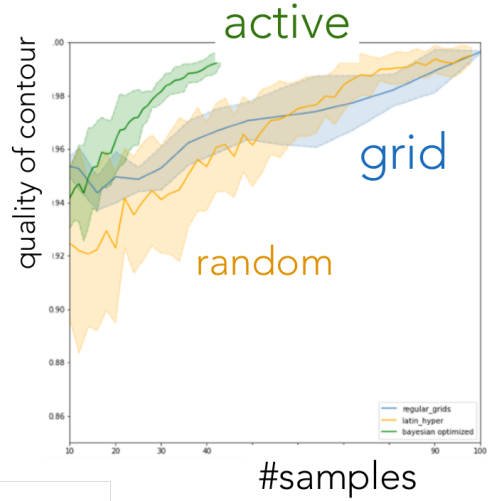
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

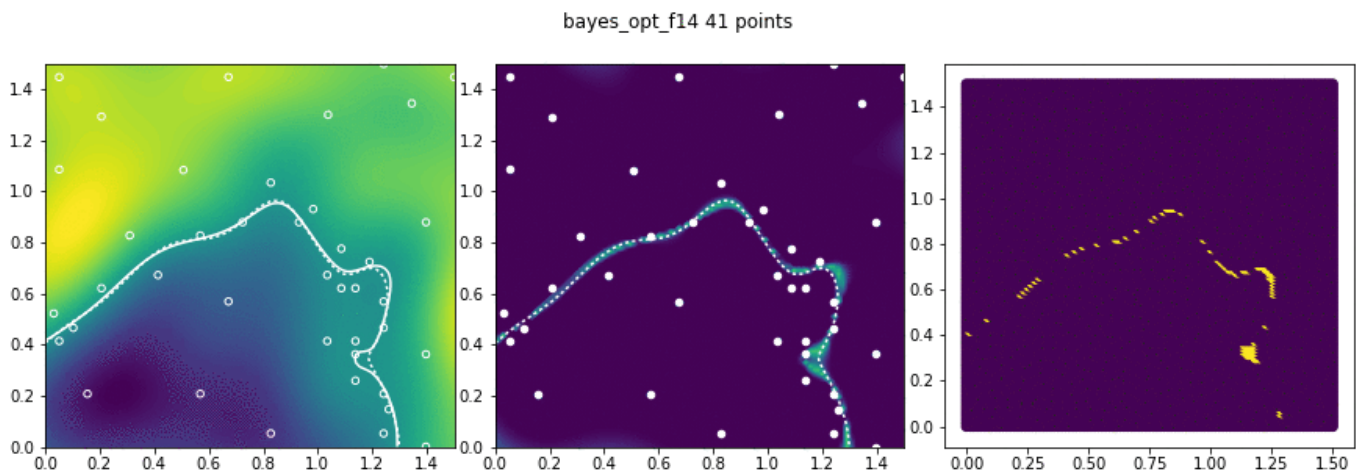




Finding exclusion contours

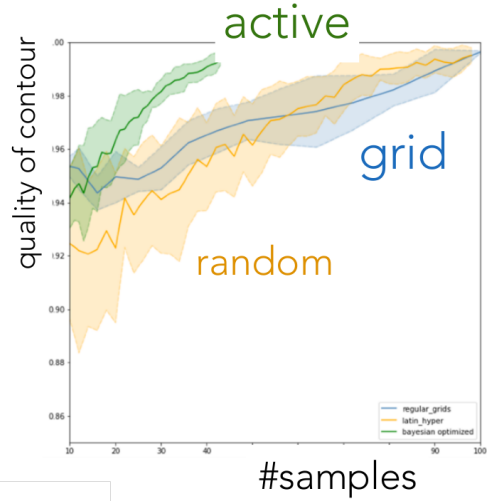
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

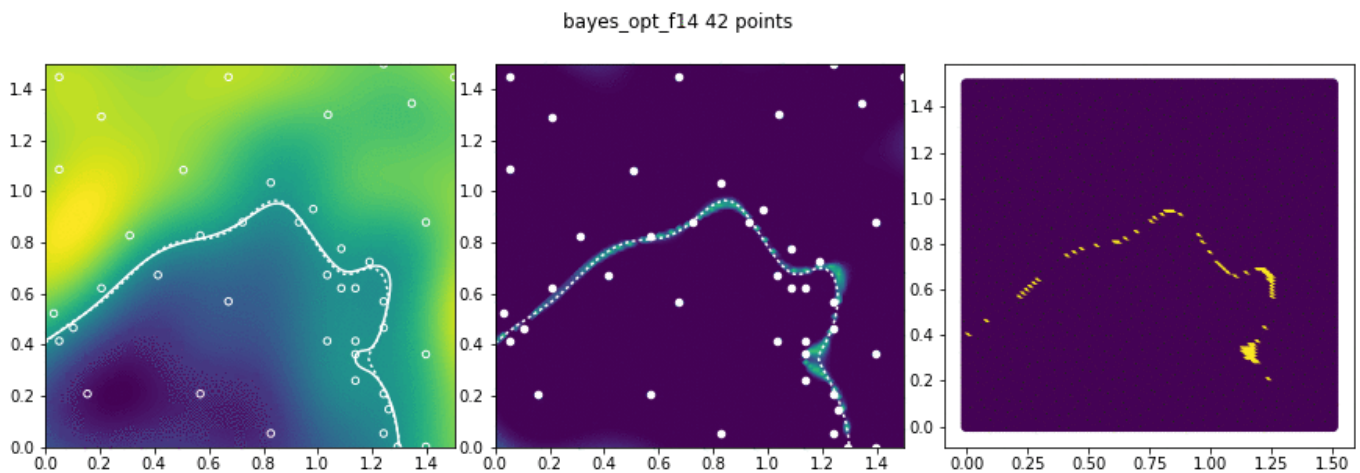




Finding exclusion contours

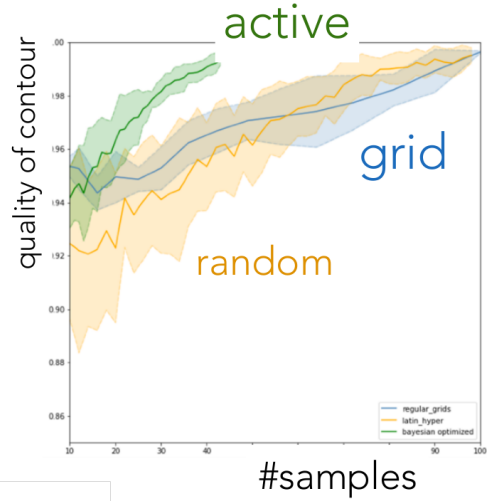
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

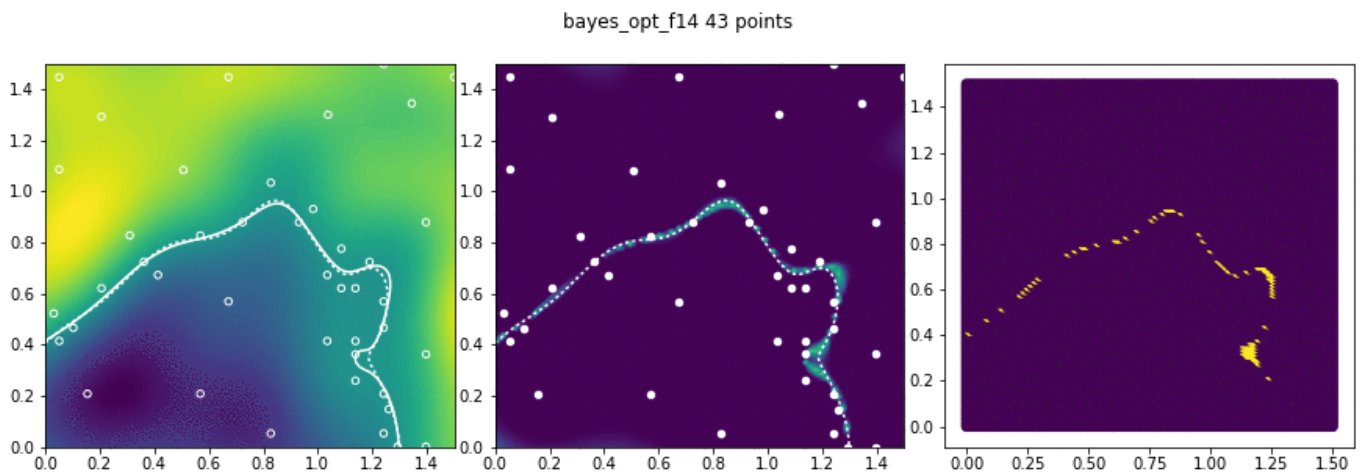




Finding exclusion contours

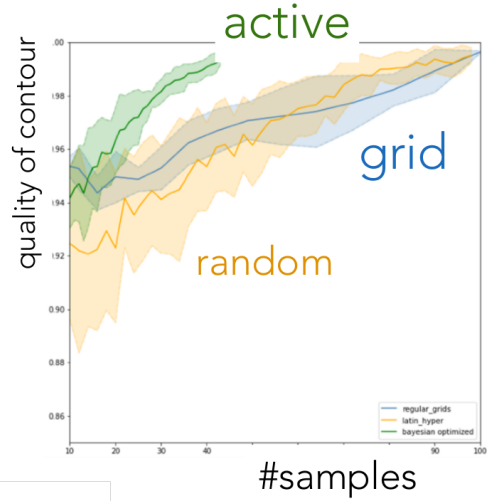
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

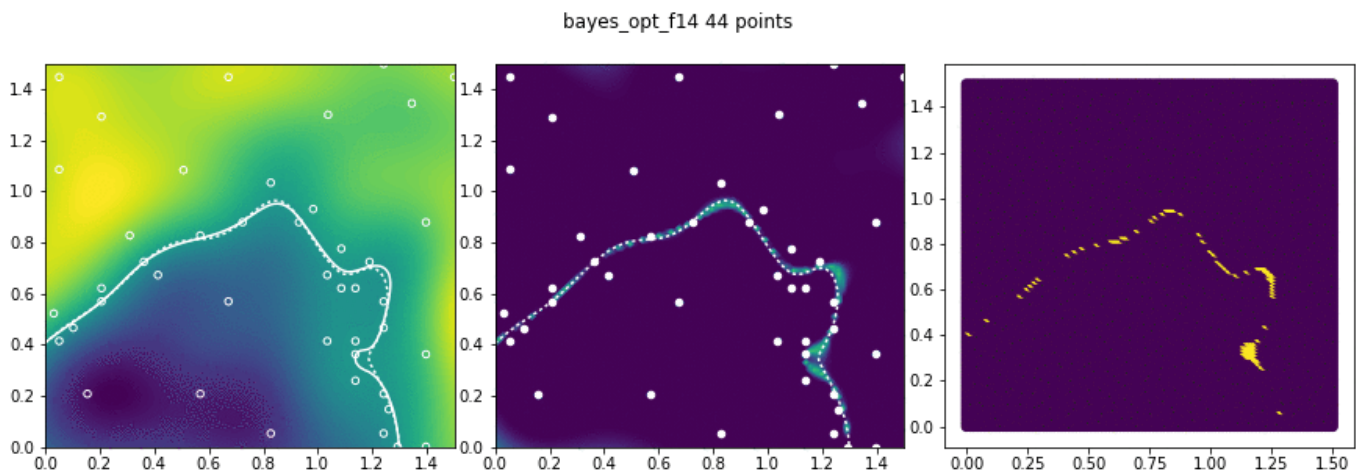




Finding exclusion contours

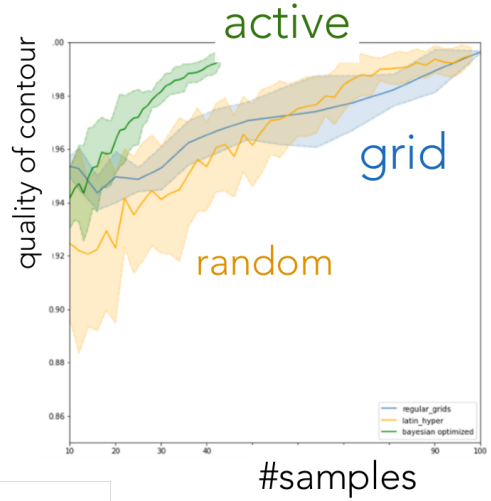
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

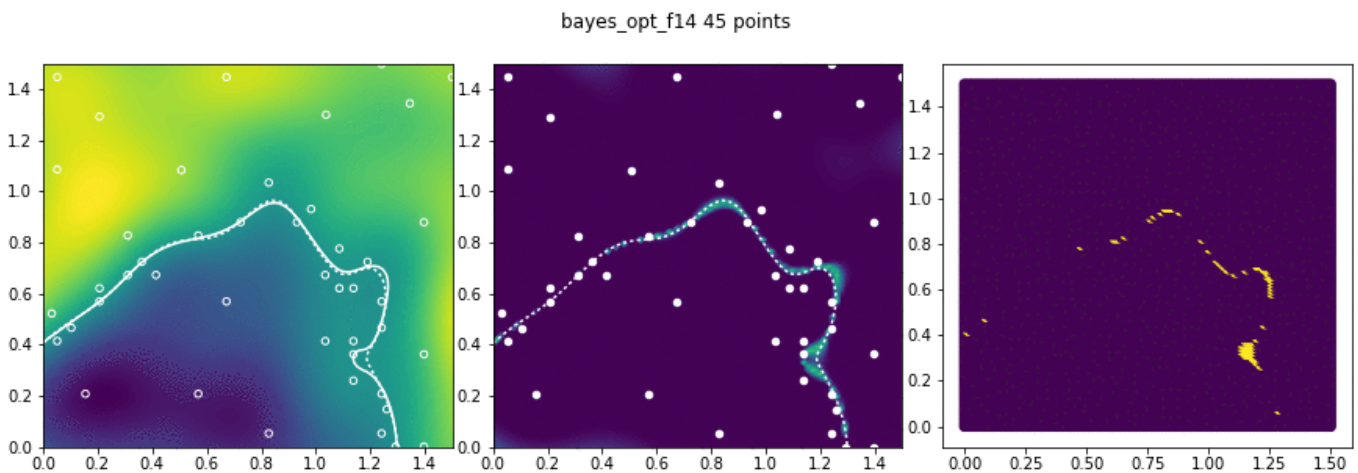




Finding exclusion contours

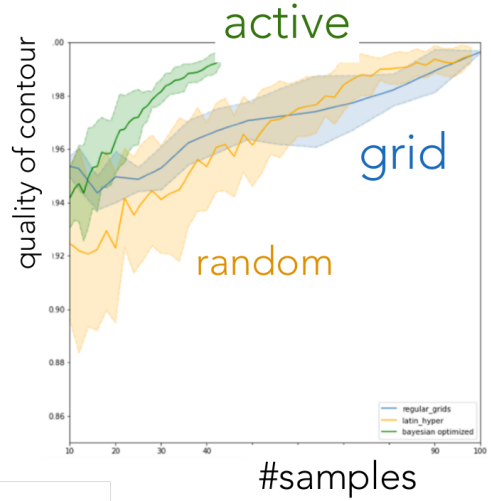
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

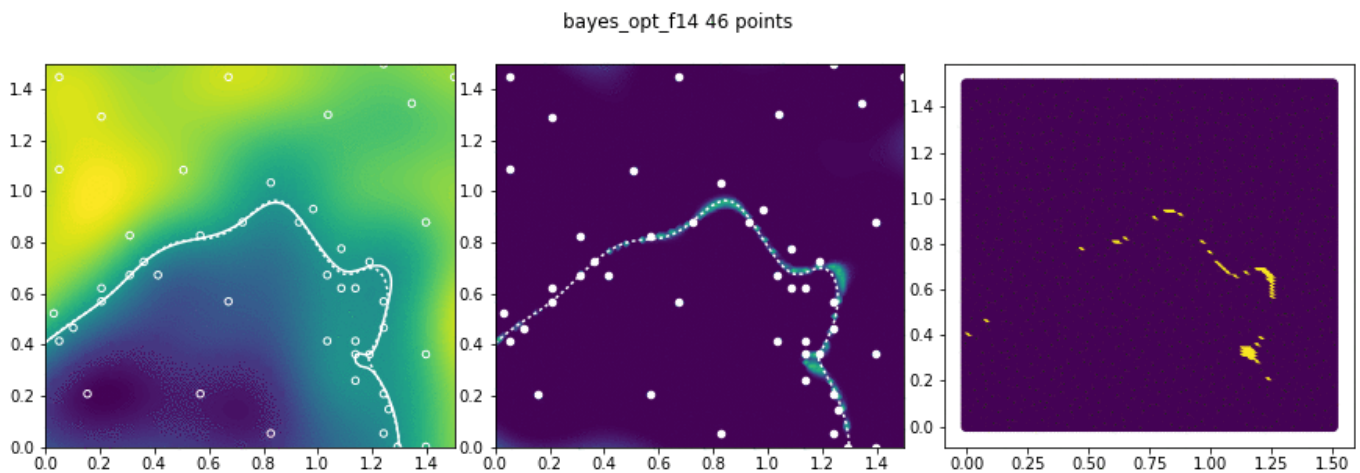




Finding exclusion contours

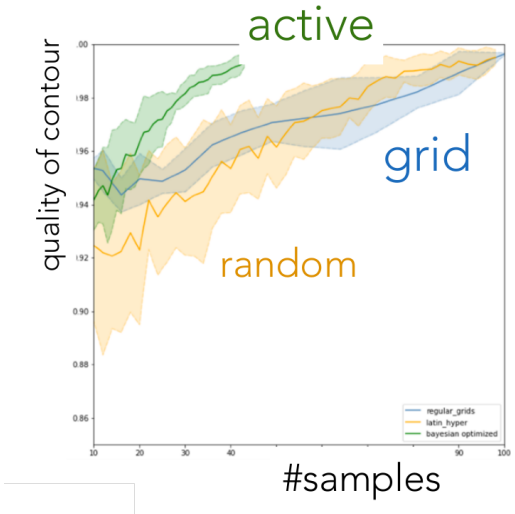
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

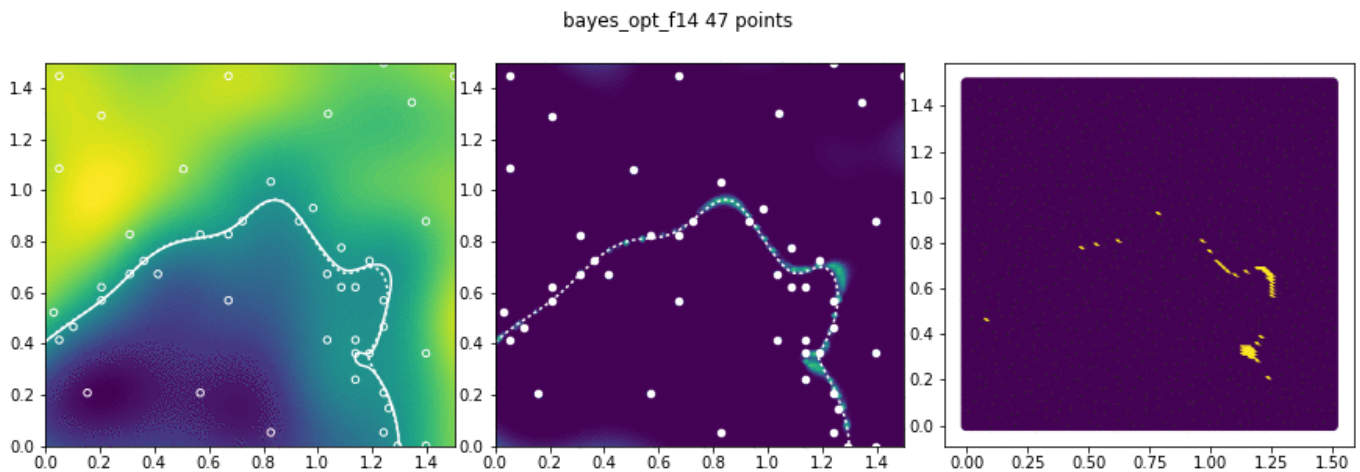




Finding exclusion contours

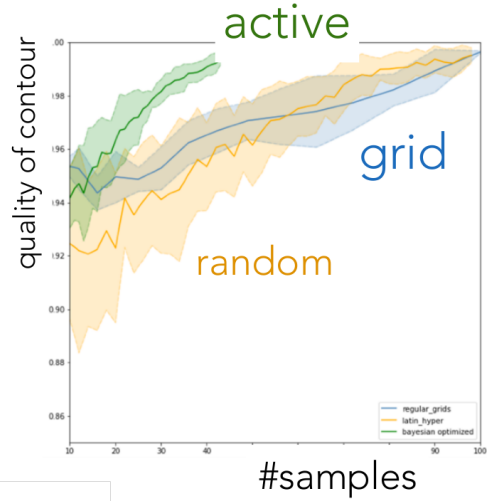
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

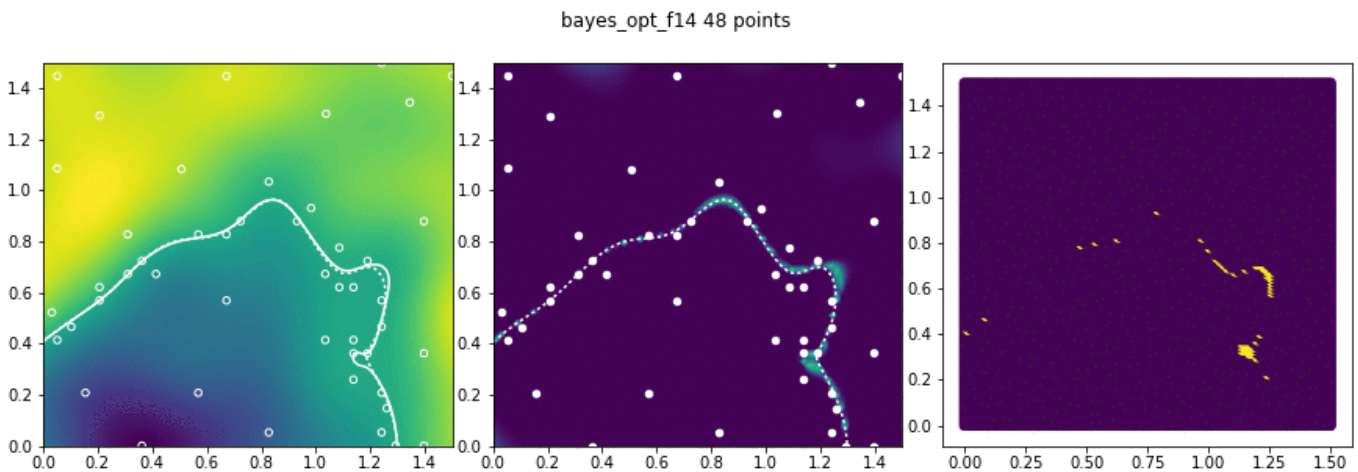




Finding exclusion contours

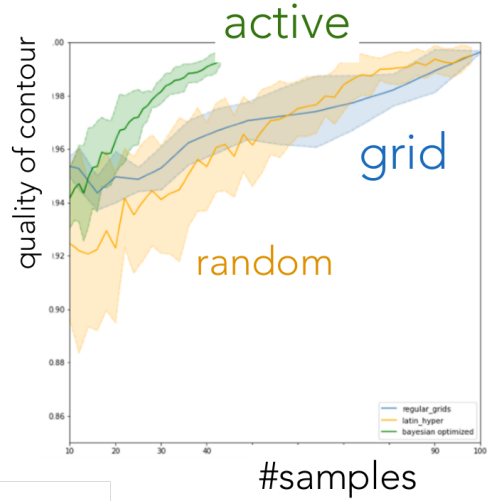
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

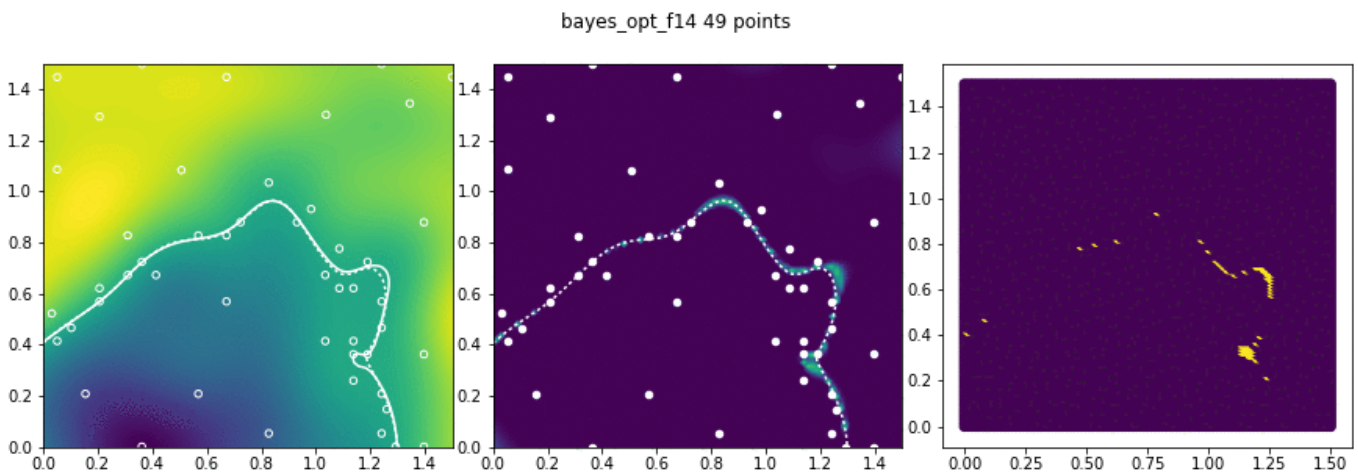




Finding exclusion contours

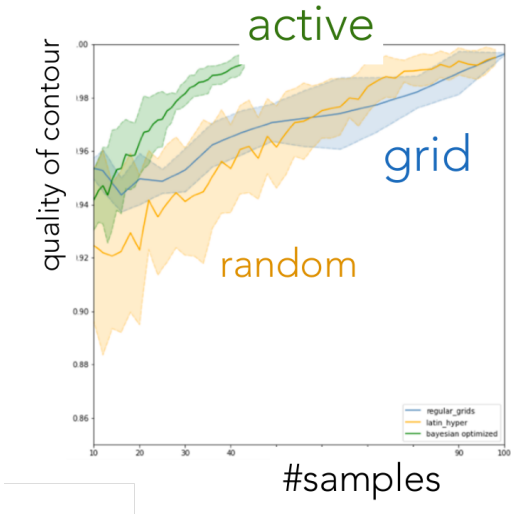
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

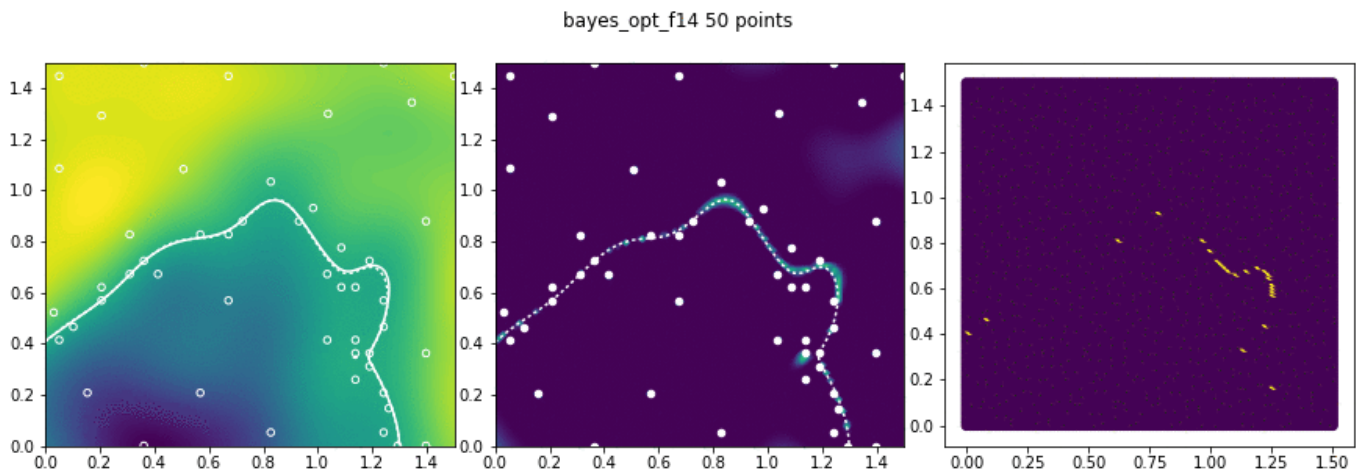




Finding exclusion contours

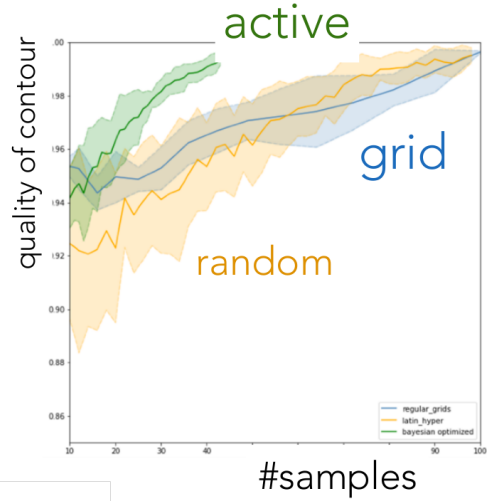
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

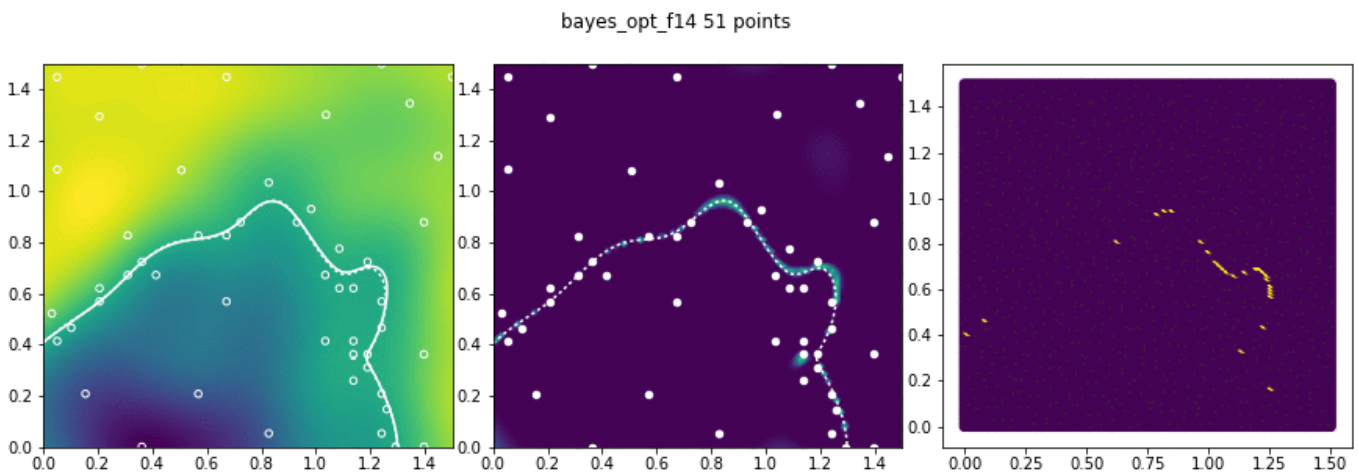




Finding exclusion contours

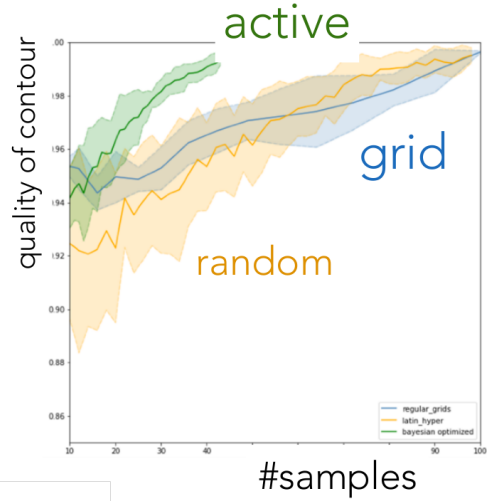
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

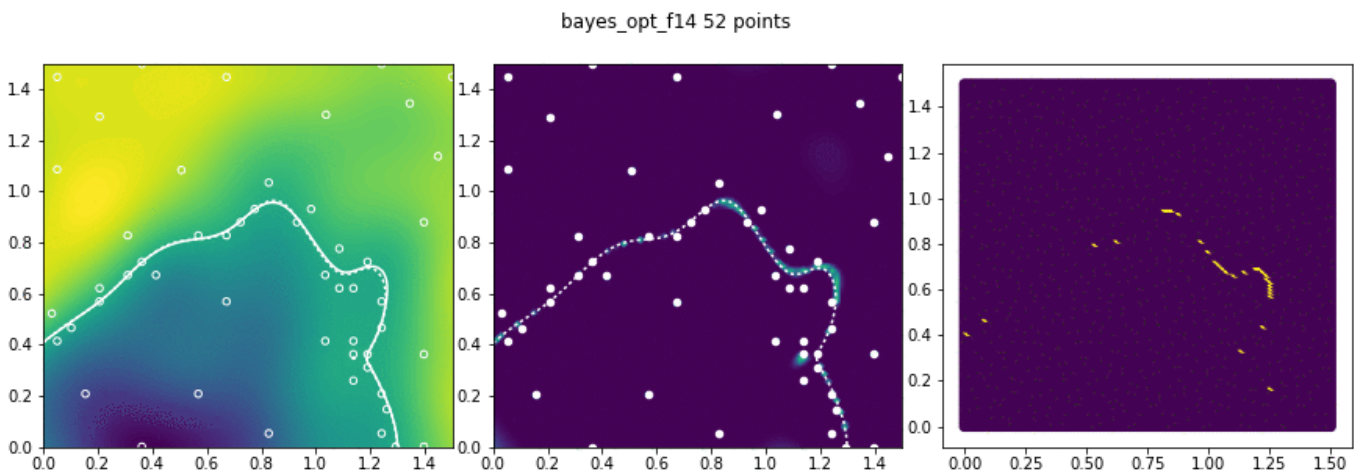




Finding exclusion contours

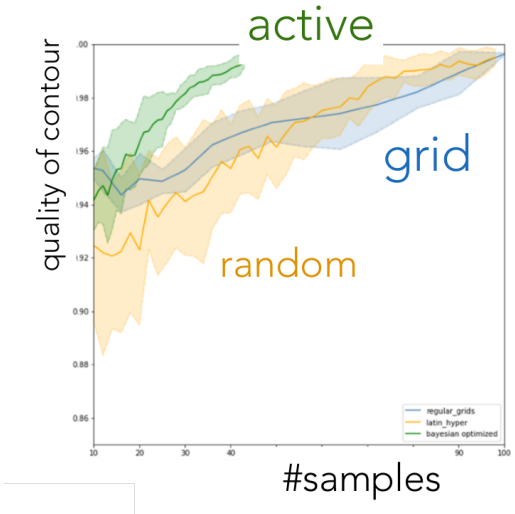
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

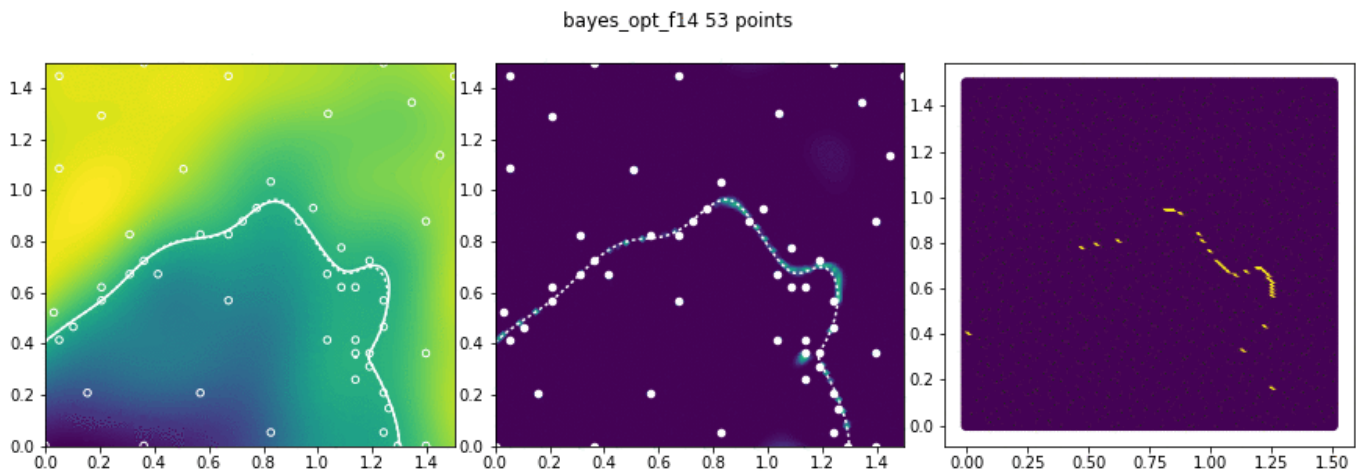




Finding exclusion contours

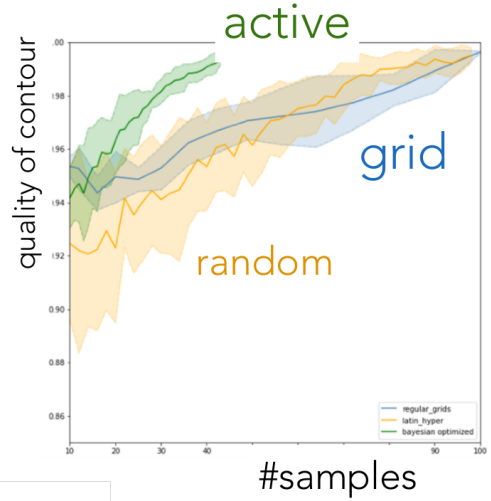
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

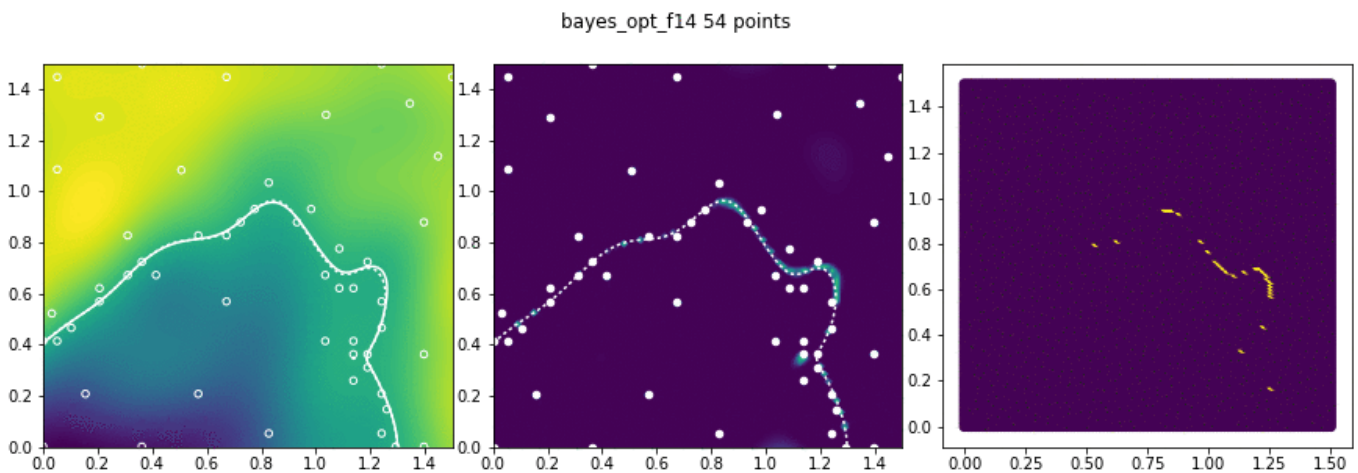




Finding exclusion contours

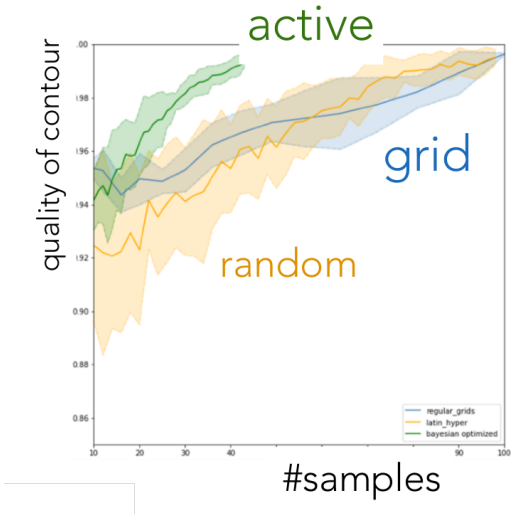
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

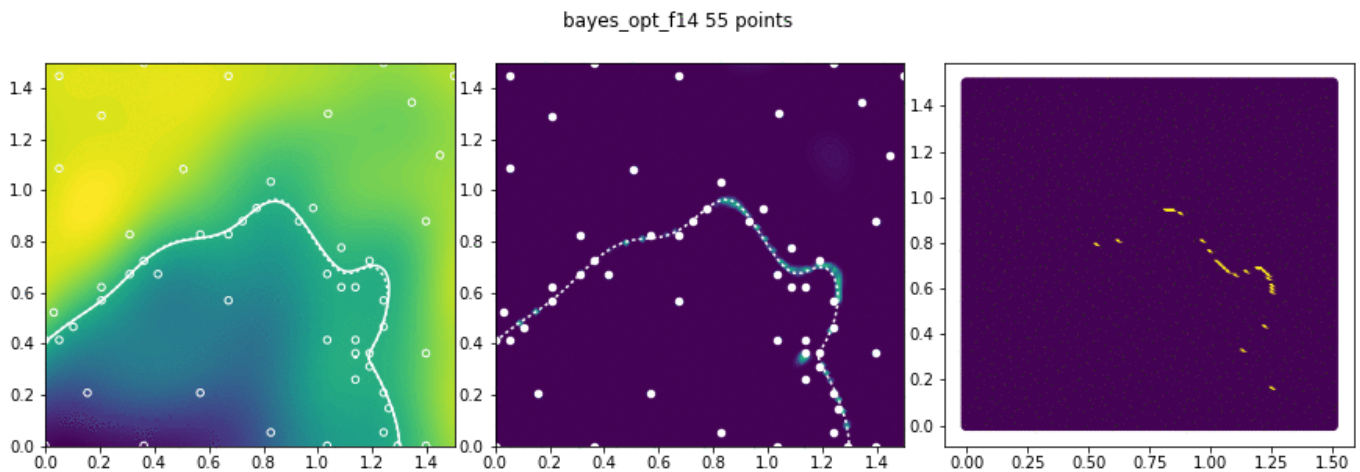




Finding exclusion contours

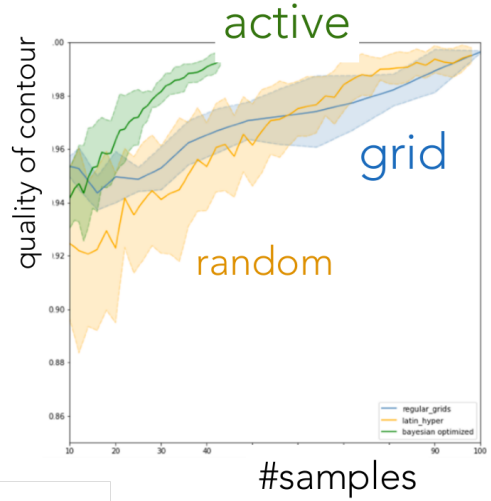
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

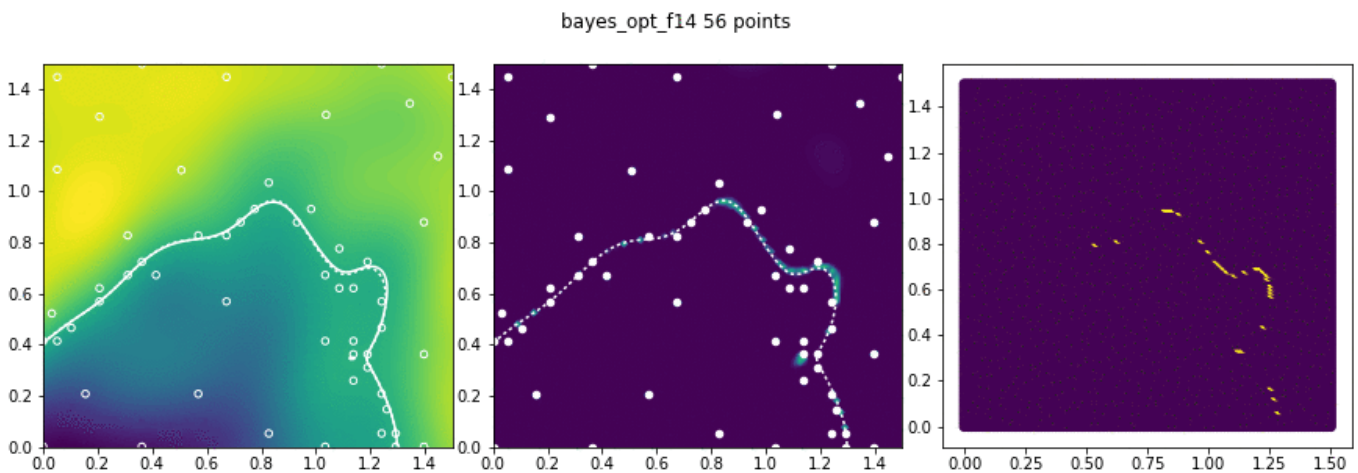




Finding exclusion contours

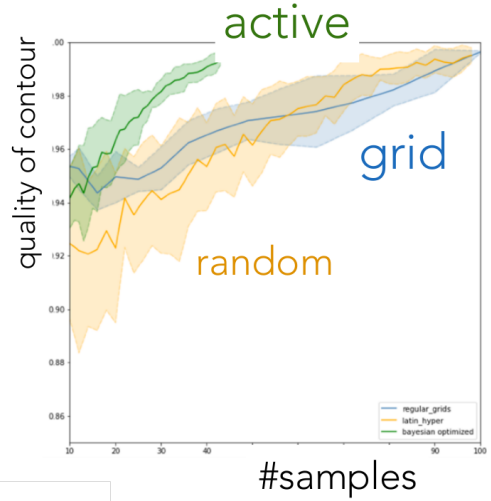
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

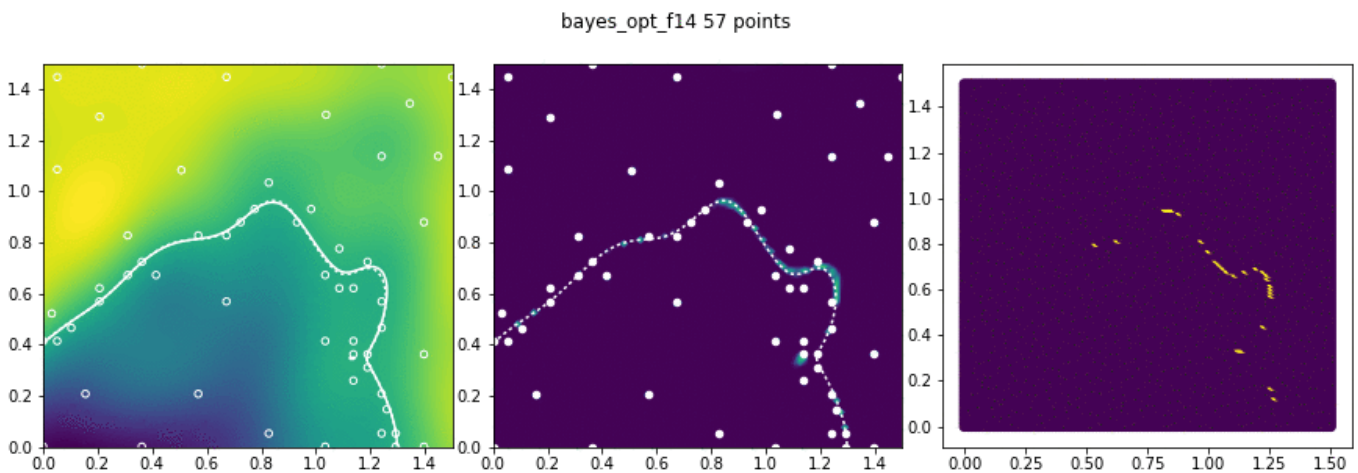




Finding exclusion contours

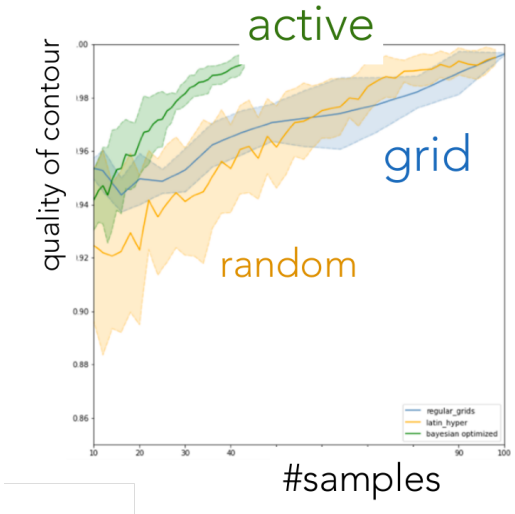
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

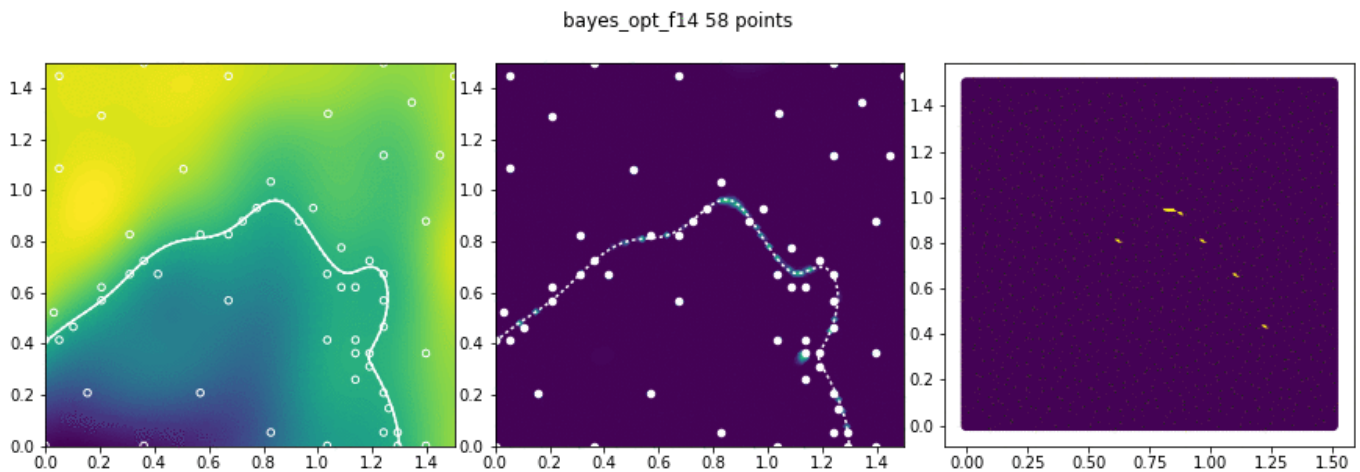




Finding exclusion contours

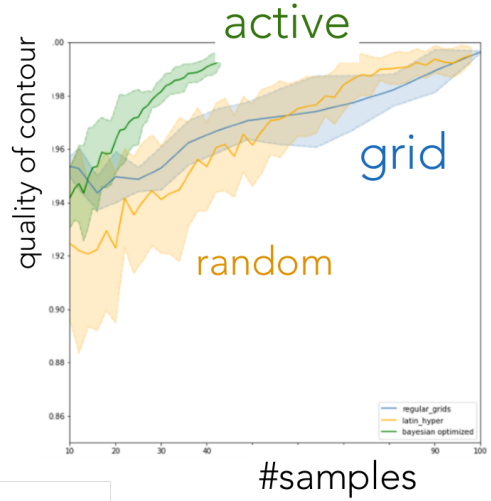
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

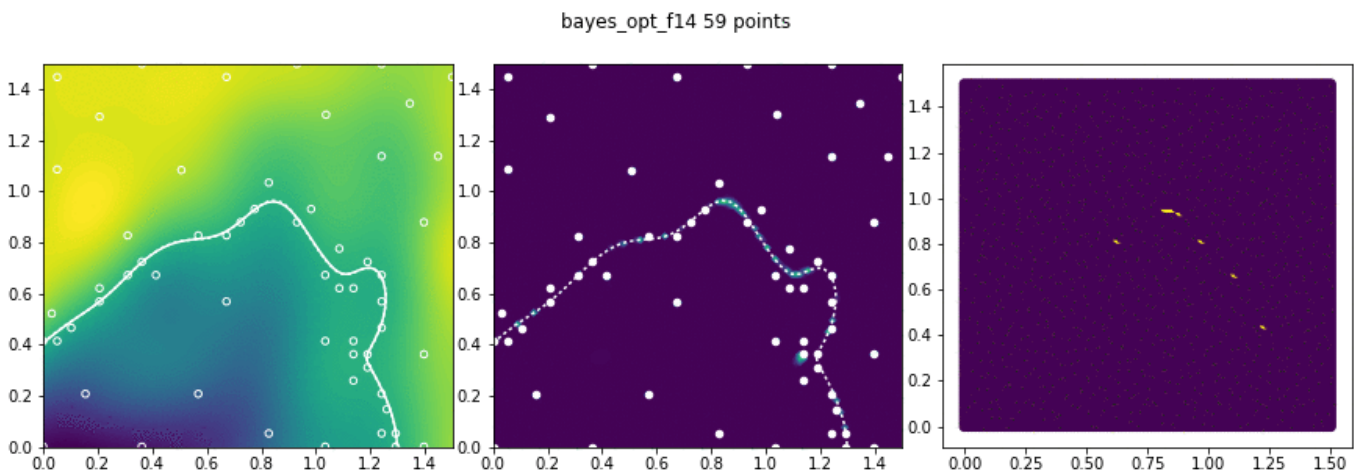




Finding exclusion contours

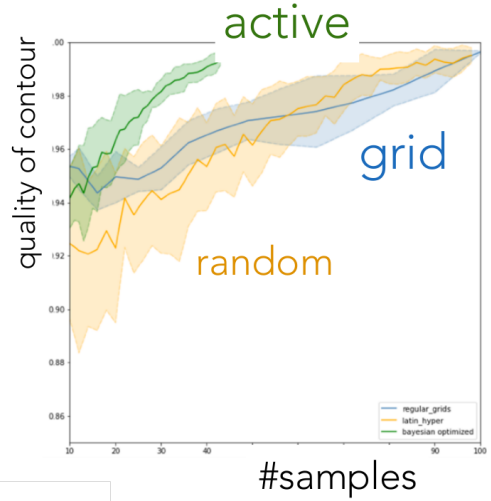
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

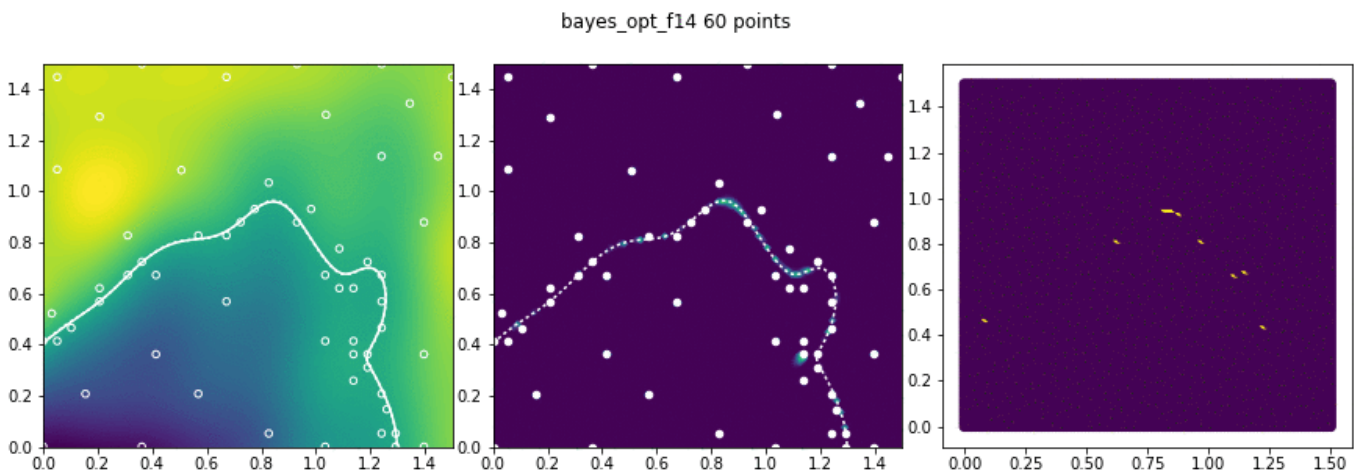




Finding exclusion contours

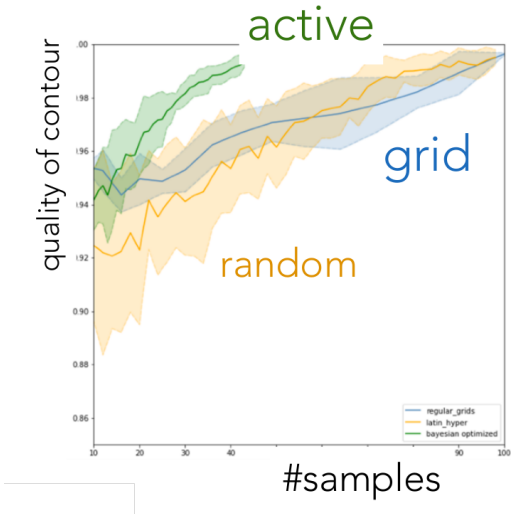
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

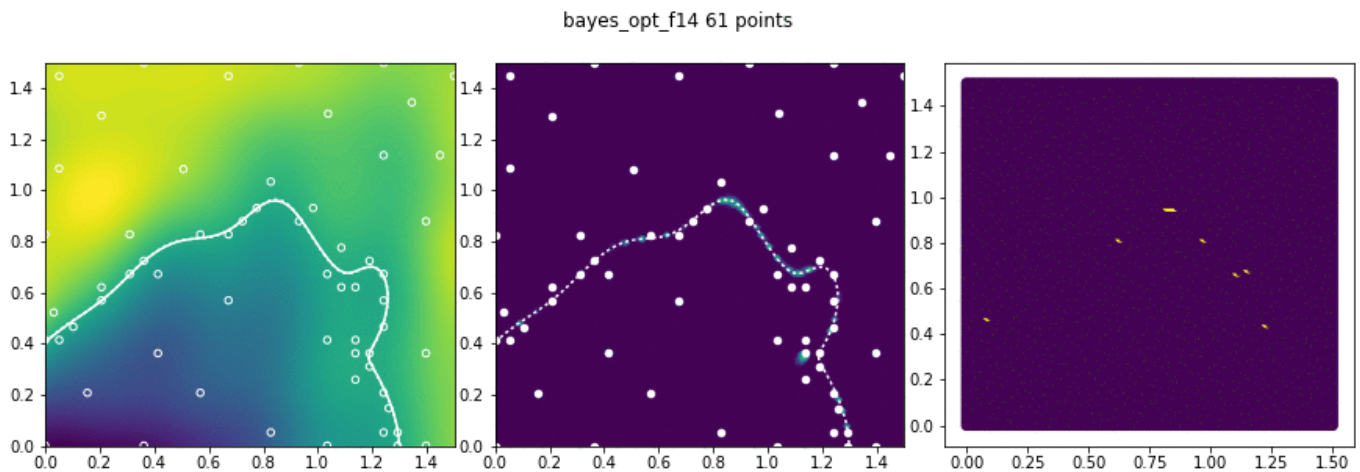




Finding exclusion contours

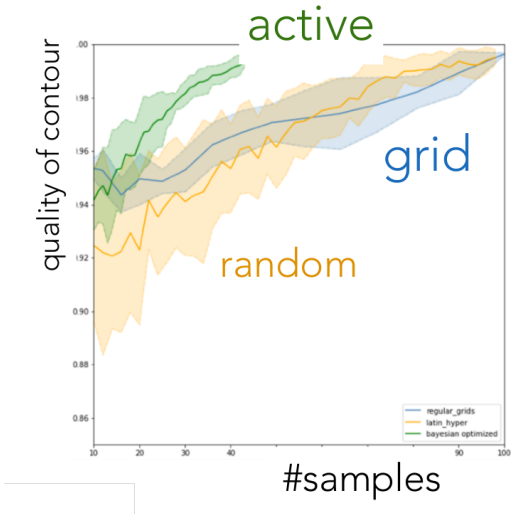
- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.

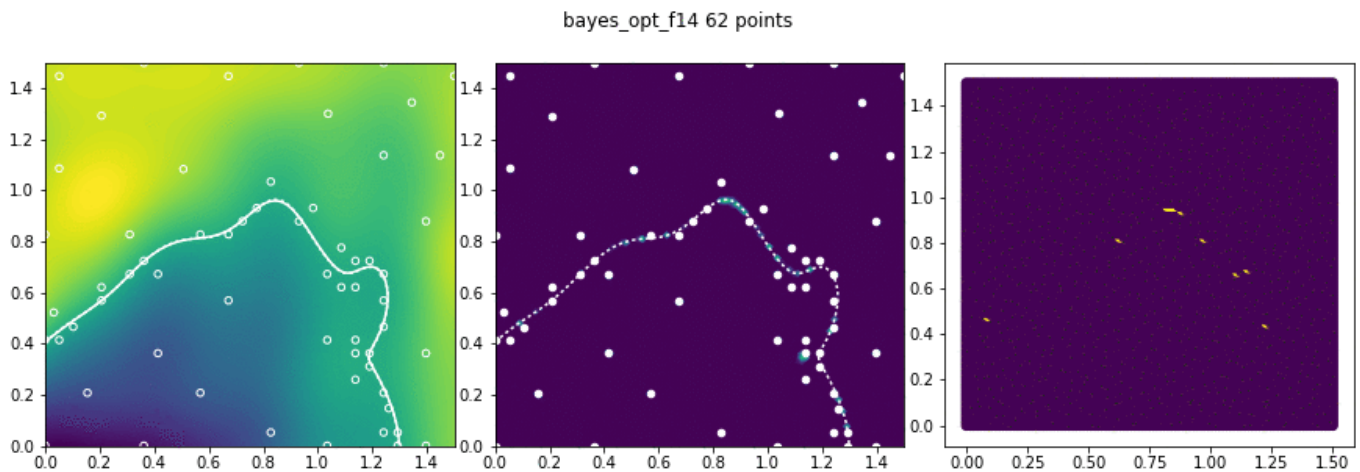




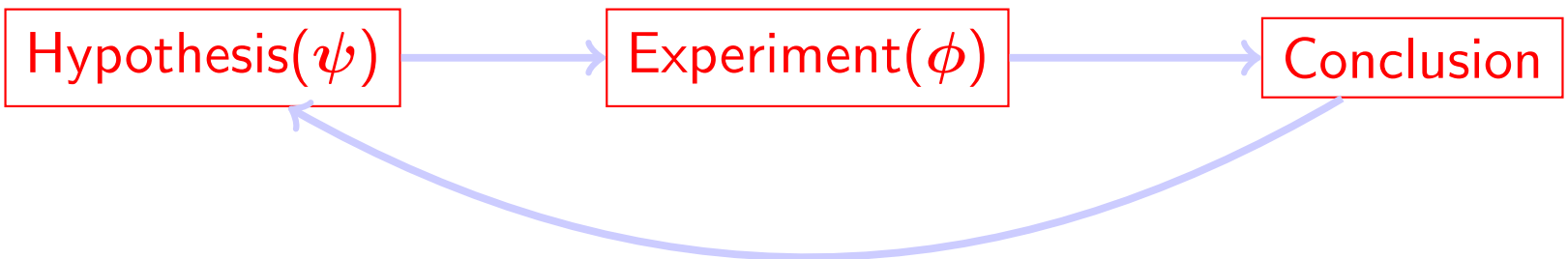
Finding exclusion contours

- Do not generate Monte Carlo a priori, generate it on demand only where it is relevant.
  - where the value of the test statistic (e.g., CLs) to be above/below the threshold is uncertain.
- Embed and instrument the full experimental pipeline through RECAST.
- Drastically more efficient use of computing resources.





AI recipe for understanding Nature



Find $\{\boldsymbol{\psi} | \rho(p_r(\mathbf{x}|\boldsymbol{\phi}), p(\mathbf{x}|\boldsymbol{\psi}, \boldsymbol{\phi}, \boldsymbol{\theta}^*)) < \epsilon, \forall \boldsymbol{\phi}\}$

Summary

- Likelihood-free inference algorithms are modern generalizations of histogram-based inference.
- Very active field of research, which connects many related problems and algorithms.
- Particle physics provide a unique testbed for the ambitous development of ML/AI methods, as enabled by the precise mechanistic understanding of physical processes.

Joint work with