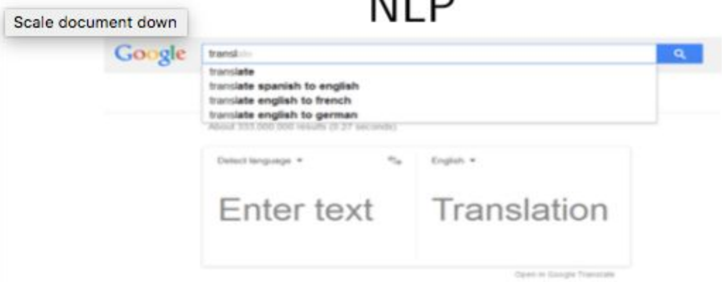
Lecture 5
Hyper-parameter Optimization

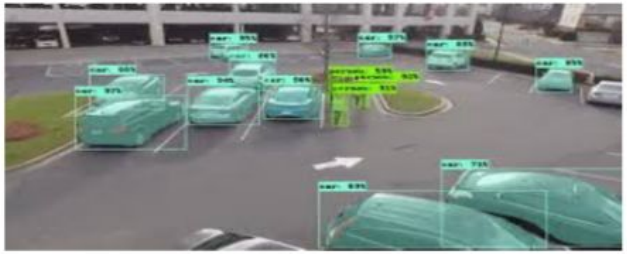
Lisheng Sun-Hosoya, Feb 4

Successes of Machine learning

NLP



Computer vision



Speech recognition

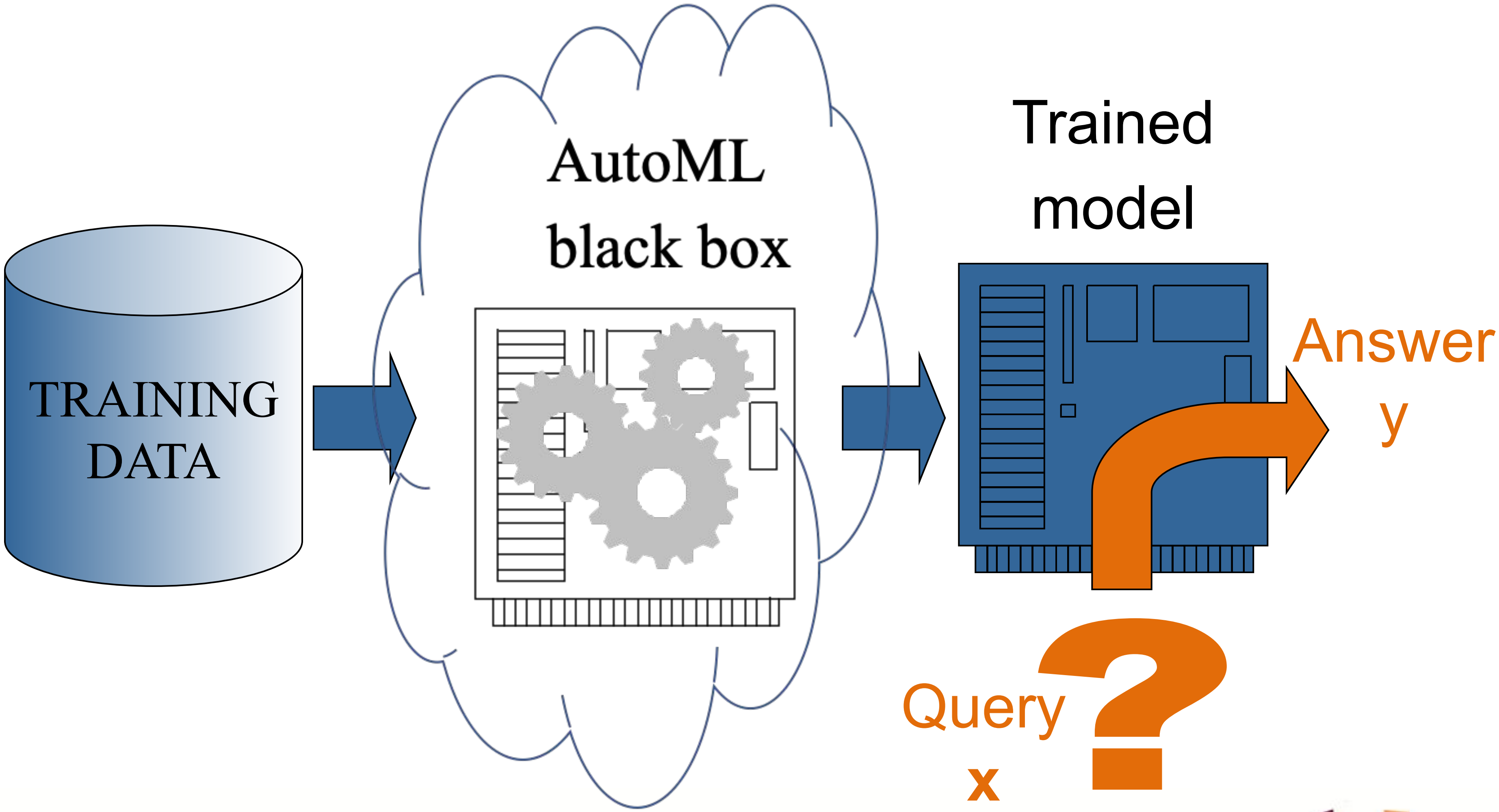
Games

… relies on **extensive** and **manual** tuning of algorithms and their hyperparameters

universite
PARIS-SACLAY

What is Hyper-parameter Optimization (HPO)?

This class!


AutoML
black box
Trained
model
TRAINING
DATA
Answer
y
Query
x

The HPO Problem

Given

- Training / validation set: $\mathbf{D}_{train}, \mathbf{D}_{valid} \sim \mathscr{D}$,
- Scoring functions J_1, J_2,
- Hyper-parameters $\boldsymbol{\theta} \in \Theta$
- Trainable parameters $\alpha \in \mathbf{A}$

$f_{\boldsymbol{\theta}}$ is a predictive model such that:

$$\hat{y}_{valid} = f_{\boldsymbol{\theta}}(\mathbf{x}_{valid} | \operatorname*{argmin}_{\alpha \in \mathbf{A}} J_1(\mathbf{D}_{train}, \alpha))$$

we want to

$$\max_{\boldsymbol{\theta} \in \Theta} t J_2(f_{\boldsymbol{\theta}})$$

Ex: J1 = MSE, J2 = AUL / k-fold CV estimator of J1

The HPO S A R I

S: Configuration space $\boldsymbol{\Theta}$

A: Choose a configuration point θ_t

R: $J_2(\theta_t)$

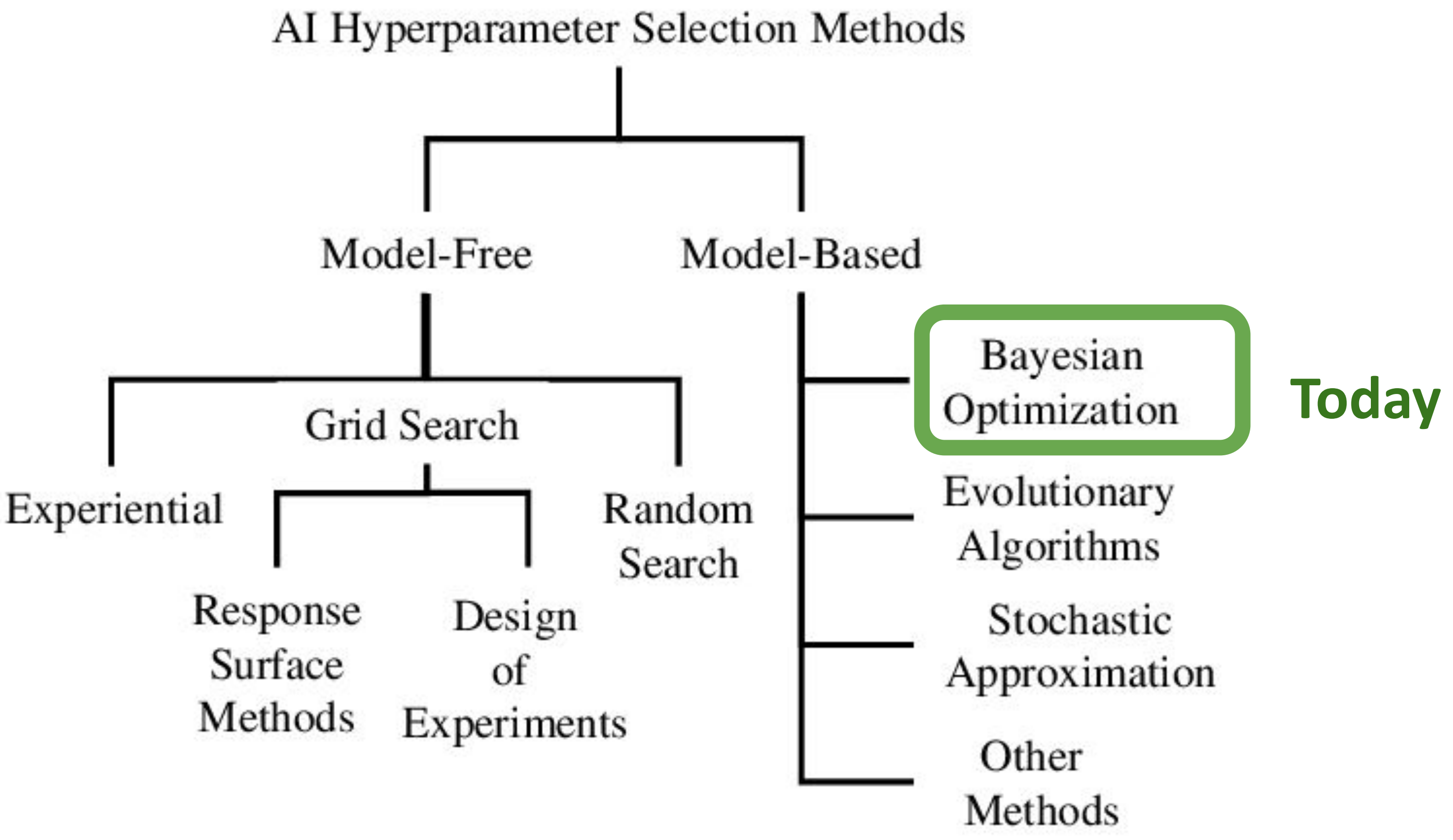
I: J, other meta-information

(dataset meta-features, comp. time, ...)

Challenges of HPO

- Bi-level, black-box optimization
- How to model the complex search space?
- Which sampling strategy?
- How to quickly evaluate a sampled configuration?

HPO Solution Taxonomy



Why Bayesian Optimization (for HPO)?

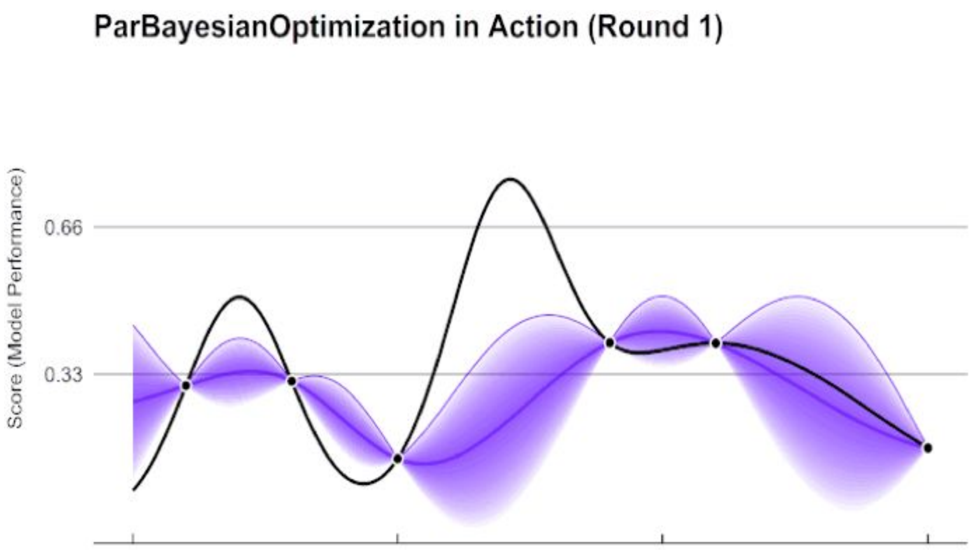
- Promising results
- Active research field
- Good basis for understanding other HPO methods

Rnd	AutoML Ended	AutoML Winners	AutoML $<R>$	AutoML $<S>$	Final Ended	Final Winners	Final $<R>$	Final $<S>$	UP (%)
0	NA	NA	NA	NA	02/14/15	1. ideal 2. abhi 3. aad	1.40 3.60 4.00	0.8159 0.7764 0.7714	NA
1	02/15/15	1. aad 2. jrl44 3. tadej	2.80 3.80 4.20	0.6401 0.6226 0.6456	06/14/15	1. aad 2. ideal 3. amsl	2.20 3.20 4.60	0.7479 0.7324 0.7158	15
2	06/15/15	1. jrl44 2. aad 3. mat	1.80 3.40 4.40	0.4320 0.3529 0.3449	11/14/15	1. ideal 2. djaj 3. aad	2.00 2.20 3.20	0.5180 0.5142 0.4977	35
3	11/15/15	1. djaj 2. NA 3. NA	2.40 NA NA	0.0901 NA NA	02/19/16	1. aad 2. djaj 3. ideal	1.80 2.00 3.80	0.8071 0.7912 0.7547	481
4	02/20/16	1. aad 2. djaj 3. marc	2.20 2.20 2.60	0.3881 0.3841 0.3815	05/1/16	1. aad 2. ideal 3. abhi	1.60 3.60 5.40	0.5238 0.4998 0.4911	31
GPU	NA	NA	NA	NA	05/1/16	1. abhi 2. djaj 3. aad	5.60 6.20 6.20	0.4913 0.4900 0.4884	NA
5	05/1/16	1. aad 2. djaj 3. post	1.60 2.60 4.60	0.5282 0.5379 0.4150	NA	NA	NA	NA	NA

Winners of AutoML challenge 2015-2016. Image source: I Guyon, L Sun-Hosoya, M Boullé, H Escalante, S Escalera, et al.. Analysis of the AutoML Challenge series 2015-2018.

Bayesian Optimization: What and How

Bayesian Optimization: The intuition



source: adapted from
https://en.wikipedia.org/wiki/Bayesian_optimization

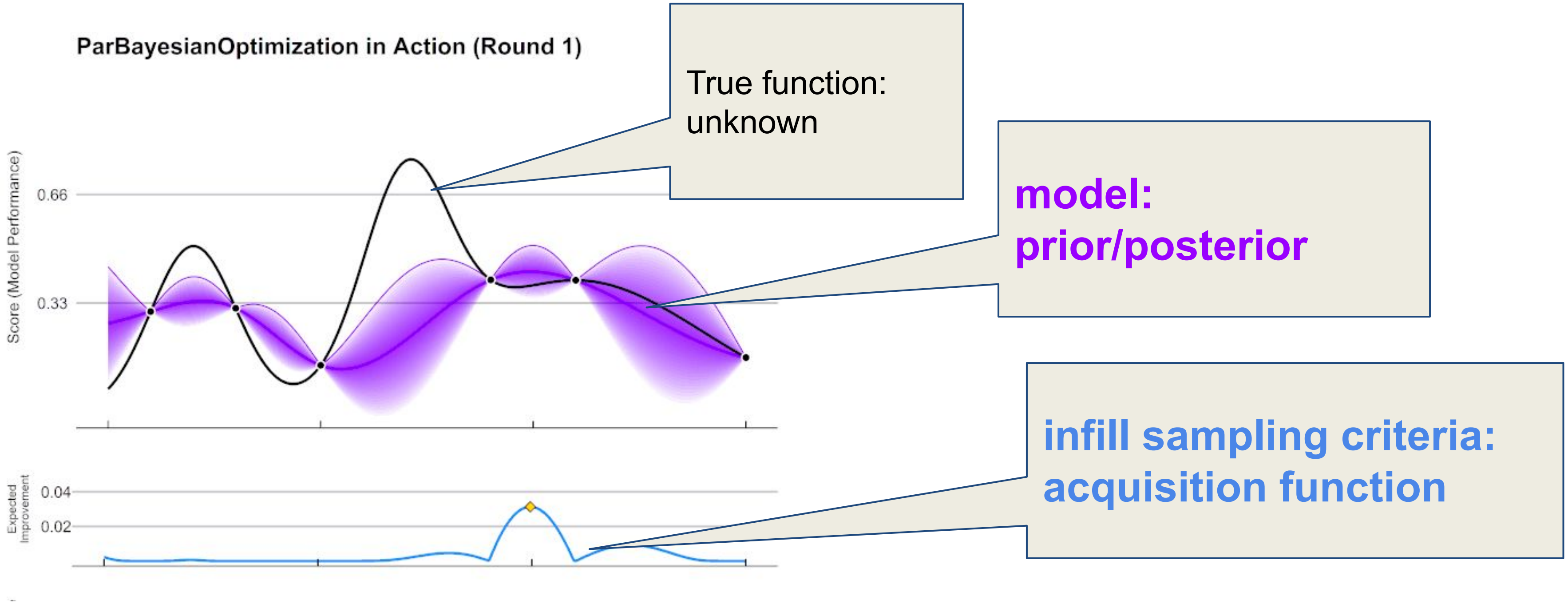
- Goal: find the max as soon as possible
- Iterate:

  (1): model the function

  (2): try the best point according to my model

  (3): update my model with my trials

BO: key components



source: adapted from
https://en.wikipedia.org/wiki/Bayesian_optimization

Step 1&3:

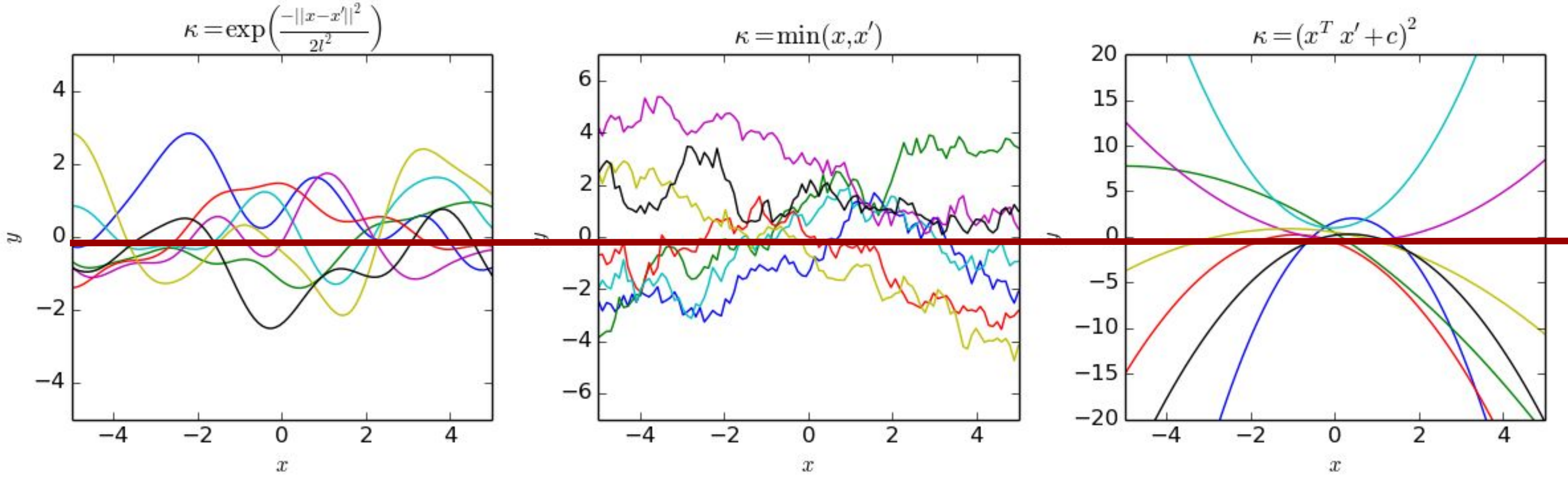
The prior / posterior

The **prior p(f)** captures our belief on f(x), it gets update with our observations {(xi, f(xi))} to form the **posterior p(f|obs.)**

Ex: Gaussian process (GP): $f(\mathbf{x}) \sim GP(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$

- Distribution of random functions
- Fully determined by mean and covariance function

GP: the kernels



$$m(\mathbf{x}) = \mathbf{0}$$

Matern kernels

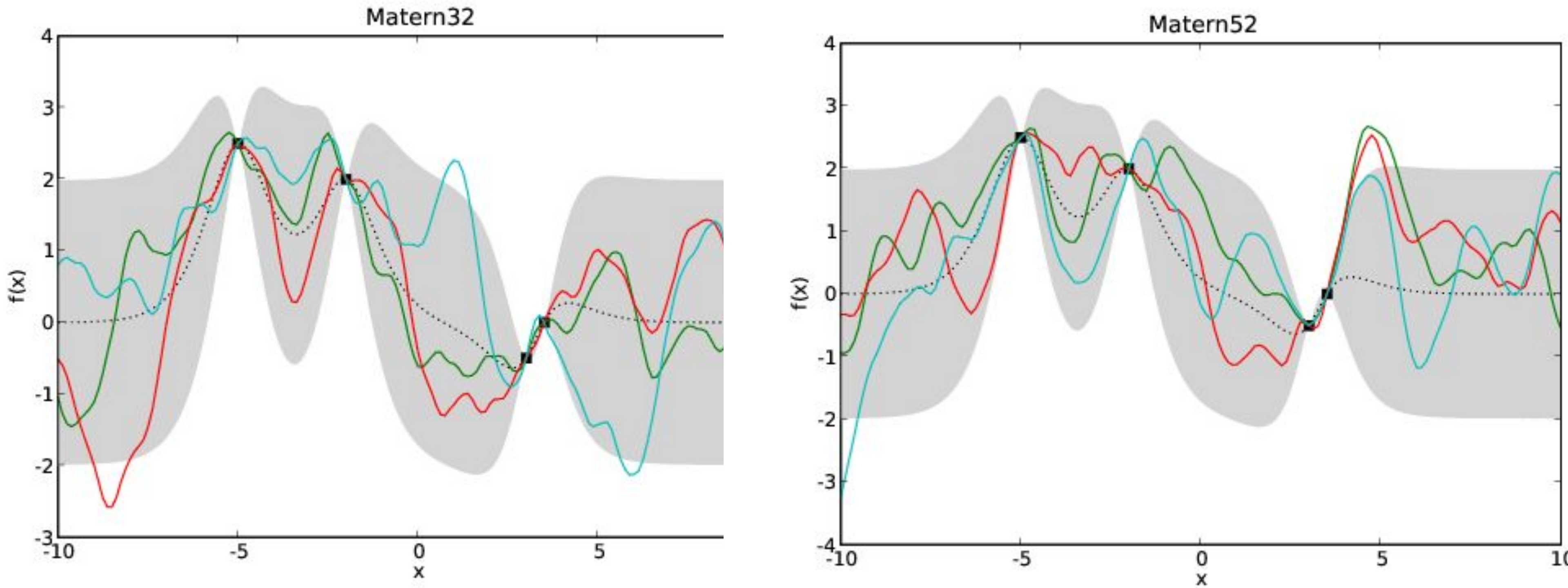
- Quality of GP depends on the kernel
- Good choice: Matern kernels [Matern, 1960, Stein, 1999]

$$k_{\nu=p+1/2}(r) = \exp\left(-\frac{\sqrt{2\nu}r}{\ell}\right)\frac{\Gamma(p+1)}{\Gamma(2p+1)}\sum_{i=0}^{p}\frac{(p+i)!}{i!(p-i)!}\left(\frac{\sqrt{8\nu}r}{\ell}\right)^{p-i}.$$

$\nu = 3/2 \text{ or } 5/2$ $\rightarrow$ Matern 3/2 and 5/2 kernel

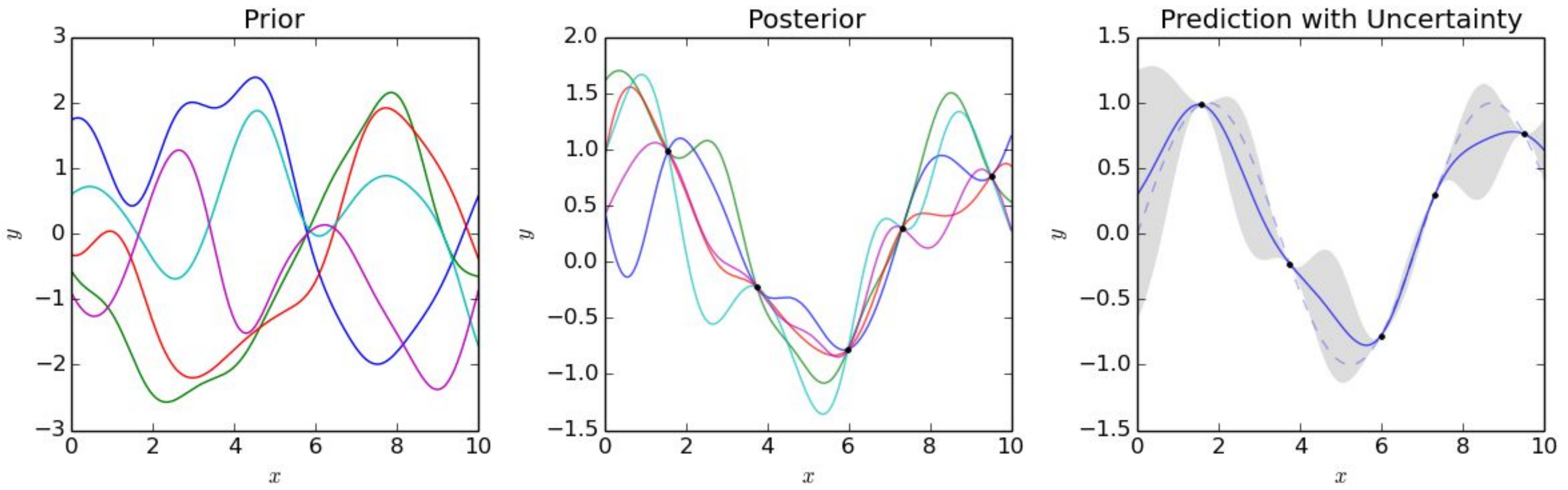
$$k_{\nu=3/2}(r) = \left(1+\frac{\sqrt{3}r}{\ell}\right)\exp\left(-\frac{\sqrt{3}r}{\ell}\right),$$

$$k_{\nu=5/2}(r) = \left(1+\frac{\sqrt{5}r}{\ell}+\frac{5r^2}{3\ell^2}\right)\exp\left(-\frac{\sqrt{5}r}{\ell}\right)$$



function samples with mean = 0 and matern 3/2 (5/2) kernels
[image source: https://pythonhosted.org/infpy/gps.html]

GP: modeling -> sampling -> predicting



Gaussian Process Prediction
[image source: https://en.wikipedia.org/wiki/Gaussian_process]

GP prior / posterior in BO

In BO, we want to 'fit' the GP with observations

$$\mathcal{D}_{1,\ldots,t} = \{\mathbf{x}_{1:t}, \mathbf{f}_{1:t}\}$$

And use the GP posterior to 'predict' $f(\mathbf{x}_{t+1})$

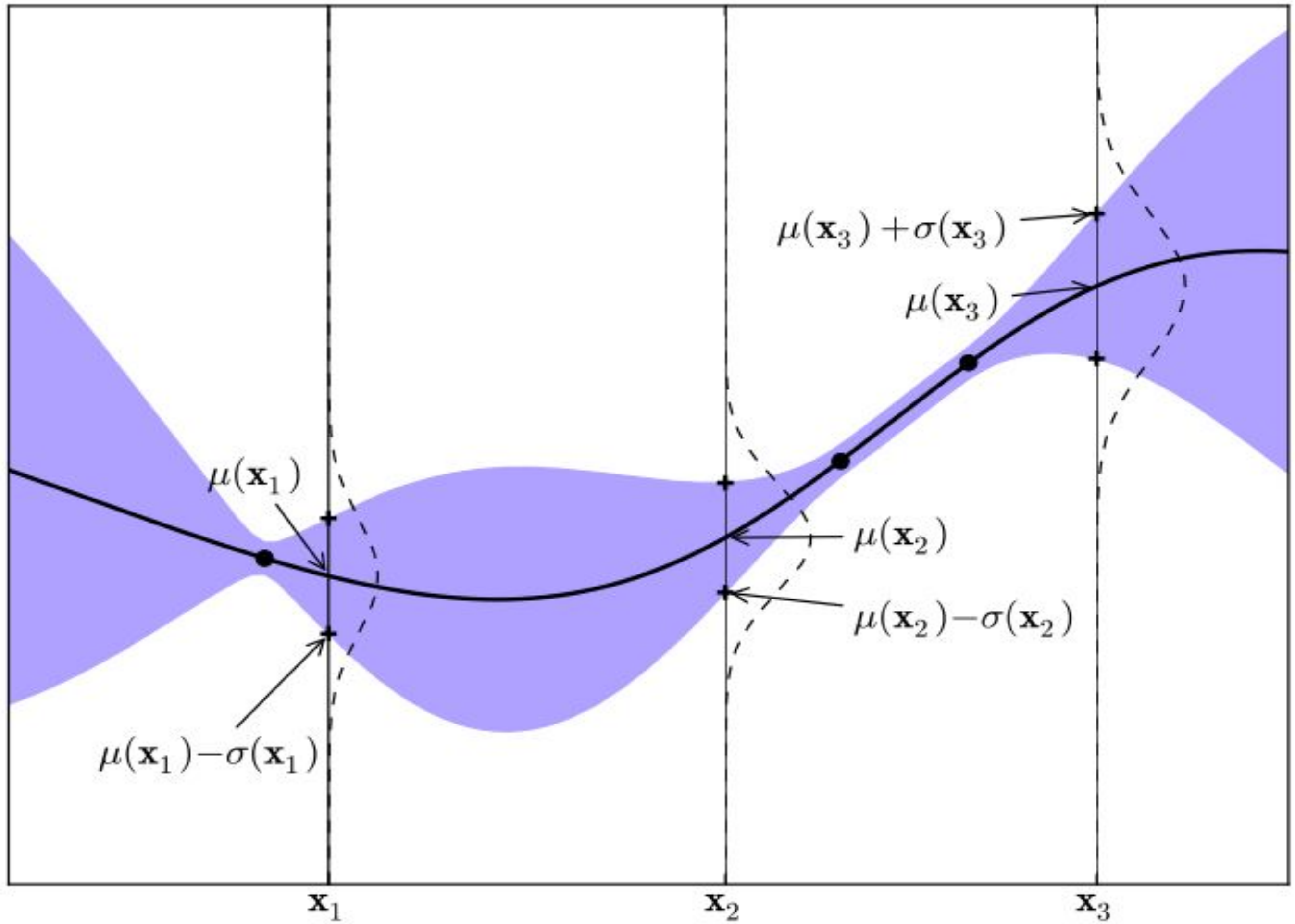
$$p(f_{t+1}|\mathcal{D}_{1,\ldots,t}, \mathbf{x}_{t+1}) = \mathcal{N}(\mu_t(\mathbf{x}_{t+1}), \sigma_t^2(\mathbf{x}_{t+1}))$$

$$\mu_t(\mathbf{x}_{t+1}) = \mathbf{k}^T \mathbf{K}^{-1} \mathbf{f}_{1:t}$$

$$\sigma_t^2(\mathbf{x}_{t+1}) = k(\mathbf{x}_{t+1}, \mathbf{x}_{t+1}) - \mathbf{k}^T \mathbf{K}^{-1} \mathbf{k}$$

$\mathbf{k}$ = covariance between $\mathbf{x}_{t+1}$ and all previous samples $\mathbf{x}_{1:t}$

$\mathbf{K}$ = covariance matrix of all previous samples $\mathbf{x}_{1:t}$



1D GP with 3 observations, the surrogate mean prediction of f(x) given the data (black line), the variance (shaded area).

[image source: Brochu et al., 2010, A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning]

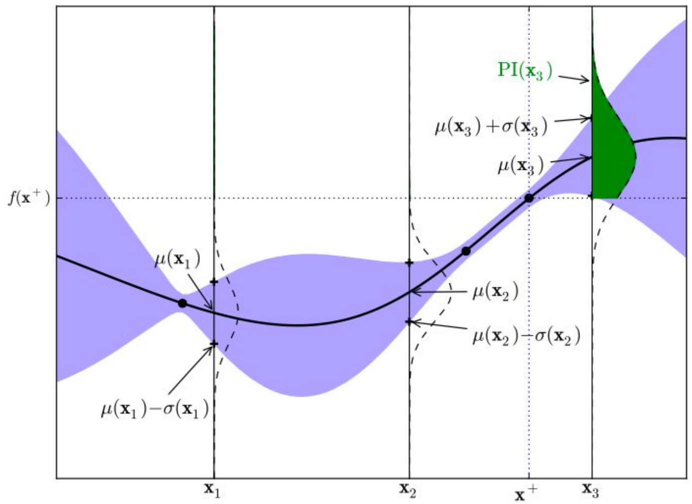
Step 2:
The acquisition function

Determines the next query point, trading-off exploration - exploitation

Ex: Probability of improvement
[Kushner et al. 1964]

$$PI(\mathbf{x}) = Pr(f(\mathbf{x}) \geq f(\mathbf{x}^+))$$
$$= \Phi(\frac{\mu(\mathbf{x}) - f(\mathbf{x}^+)}{\sigma(x)})$$

- We want to find the pt. w. max. area above the best obs. f(x+)
- This corresponds to the max of PI



[image source: Brochu et al., 2010]

Other priors

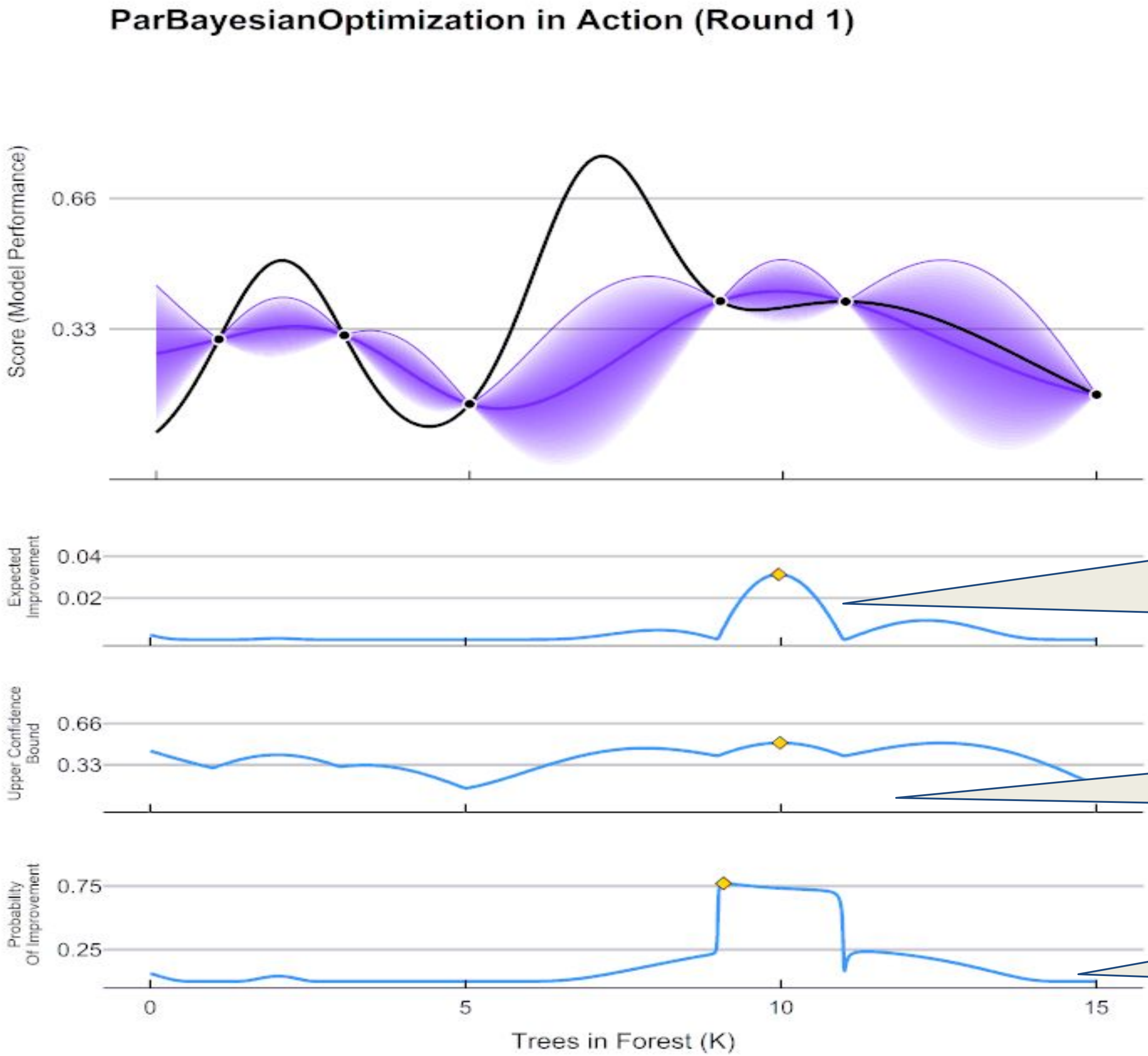
Problems with GP:

- Scales cubically in data points
- High dim. and categorical HP space: need to adapt the kernel

→ Frequentist solution: Random forest

- a collection of regression trees
- input: x;
- output: ^f(x)
- mean and variance over trees

Other acquisition functions



Expected improvement [Mockus et al., 1978, Jones et al., 1998]

UCB-GP [Srinivas et al. 2010]

PI

HPO algorithms using BO

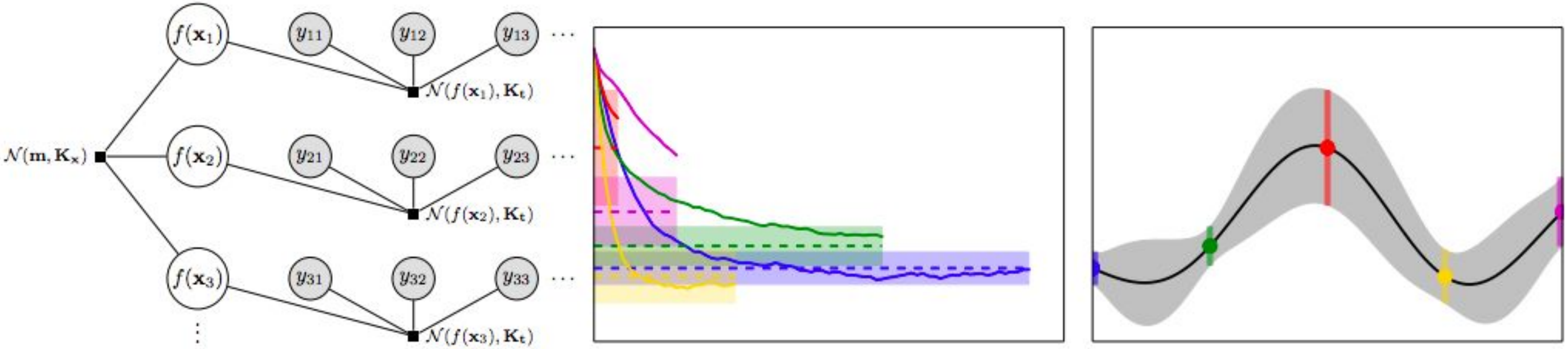
Ex 1: Freeze-Thaw BO

[Swersky et al. 2014]

- Intuition:

  Maintains a set of “frozen” (partially completed but not being actively trained) models and uses an information-theoretic criterion to determine which ones to “thaw” and continue training

- Use BO for:
  - learning curve prediction → offers quick evaluations
  - HP space modeling



(a) Graphical Model (b) Training curve predictions (c) Asymptotic GP

- GP for learning curve (b):
  - Exponential decay kernel
  - $p(f_{t+1}|\mathcal{D}_{1:t})$
- GP for HP space (c):
  - Matern 5/2 kernel
  - $p(f_{new}|\mathcal{D}_{1:t}, x_{new})$

Run some models

THINK

Algorithm 1 Entropy Search Freeze-Thaw Bayesian Optimization

Given a basket $\{(\mathbf{x}, \mathbf{y})\}_{B_{\text{old}}} \cup \{(\mathbf{x})\}_{B_{\text{new}}}$
$a = (0, 0, \ldots, 0)$
Compute $P_{\min}$ over the basket using Monte Carlo simulation and Equation 19.
for each point $\mathbf{x}_k$ in the basket **do**
    // n_{fant} is some specified number, e.g., 5.
    for $i = 1 \ldots n_{\text{fant}}$ **do**
        if the point is old **then**
            Fantasize an observation y_{t+1} using Equation 20.
        end if
        if the point is new **then**
            Fantasize an observation y_1 using Equation 21.
        end if
        Conditioned on this observation, compute $P^y_{\min}$ over the basket using Monte Carlo simulation and Equation 19.
        $a(k) \leftarrow a(k) + \frac{H(P^y_{\min}) - H(P_{\min})}{n_{\text{fant}}}$ // information gain.
    end for
end for
Select $\mathbf{x}_k$, where $k = \text{argmax}_k\, a(k)$ as the next model to run.

Run next model

Ex 2: Auto-sklearn

- Intuition:

  Warm start the BO with meta-learning techniques, ensemble the top models.

- Use BO for:

  HP space modeling

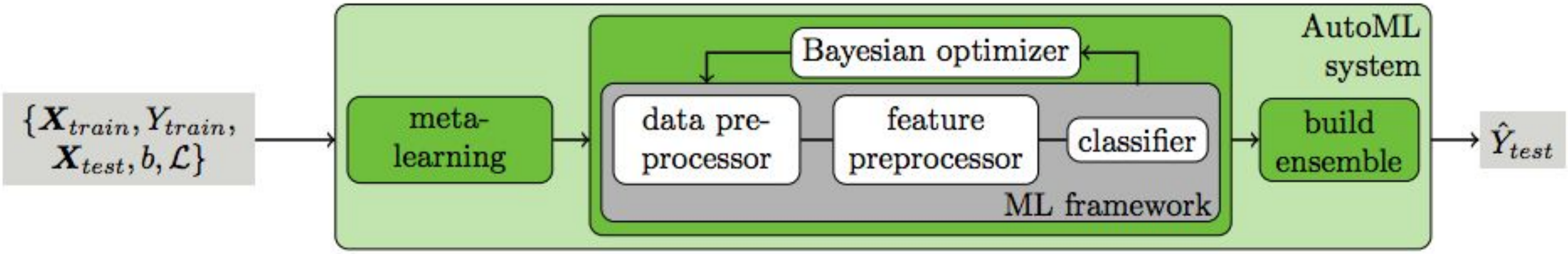


Figure 1: Our improved approach to AutoML. We add two components to Bayesian hyperparameter optimization of an ML framework: meta-learning for initializing the Bayesian optimizer and automated ensemble construction from configurations evaluated during optimization.

Meta-learning [Brazdil et al., 2009]:

- characterize the dataset using meta-features,
- initialize BO with config. that performed well on old similar dataset

BO subroutine: SMAC [Hutter et al., 2011]

- Random Forest prior
- Expected improvement acquisition
- 1 fold quick evaluation

universite
PARIS-SACLAY

Today’s
Take-home messages

Take-home messages (1)

What you have learned:

- HPO: bi-level, black-box optimization problem

- Bayesian Optimization: a powerful solution w. 2 key ingredients:
  - a prior: to model the space
  - an acquisition function: to guide the sampling

Take-home messages (2)

What you can use in your projects:

- Auto-sklearn: open-source, active community
  https://automl.github.io/auto-sklearn/master/

- NNI: more than BO, good for deep learning models
  https://github.com/microsoft/nni

- Hyperopt:
- https://github.com/hyperopt/hyperopt

Take-home messages (3)

Open Question and research directions:

- Benchmarks and Comparability

  eg. Black-box Optimization Benchmarking, AutoML and AutoDL challenges

- Gradient-Based Optimization

  eg. Maclaurin et al., 2015, Franceschi et al., 2017, Pedregosa, 2016, etc.

- Scalability and parallezation

  Bergstra et al., 2011, Desautels et al., 2014, Falkner et al., 2018, etc.

- Towards meta-learning (coming lecture)