Bayesian Optimization with High-Dimensional Outputs

Wesley Maddox[1], Max Balandat[2], Andrew Gordon Wilson[1], Eytan Bakshy[2]

[1] NEW YORK UNIVERSITY [2] Meta

NEURAL INFORMATION PROCESSING SYSTEMS

Paper: https://arxiv.org/abs/2106.12997

Methods

Multi-task Gaussian process (MTGP) posterior covariance matrices have the following form:

$$\Sigma^* = (K_{x^*,x^*} \otimes K_T) - (K_{x^*,X} \otimes K_T)(K_{XX} \otimes K_T + \sigma^2 I_{nT})^{-1}(K_{x^*,X}^{\top} \otimes K_T)$$

So they are of size $n_{test}t \times n_{test}t$

- sampling from these matrices costs n^3t^3 time, which is impractical.

Instead, we use Matheron's rule to draw samples from the MTGP posterior.

$$f^* \mid (Y + \epsilon = y) = f^* + K_{x^*X}(K_{XX} + \sigma^2 I)^{-1}(y - Y - \epsilon),$$

Where $(f^*, Y) \sim \mathcal{N}(0, K_{joint})$

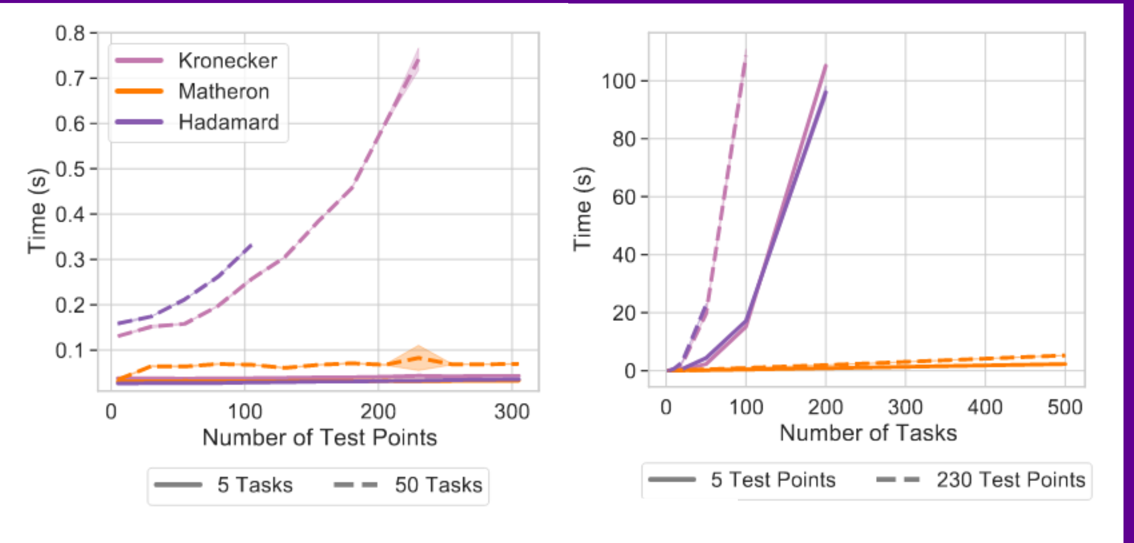
Single task setting provides no speedup

But in multi-task setting, we exploit Kronecker structure to get $n^3 + t^3$ scaling

$$K_{joint} = K_{(x^*,X),(x^*X)} \otimes K_T = (\tilde{R} \otimes L)(\tilde{R} \otimes L)^{\top}$$

Need: $w = (K_{XX} \otimes K_T + \sigma^2 I_{nT})^{-1}(y - Y - \epsilon)$

And $(K_{x^*,X} \otimes K_T)w$



Test points, GPU — Tasks, CPU

Matheron's rule MTGP sampling is significantly faster & more memory efficient.

Code is available in Botorch (https://botorch.org) with tutorials!

References:

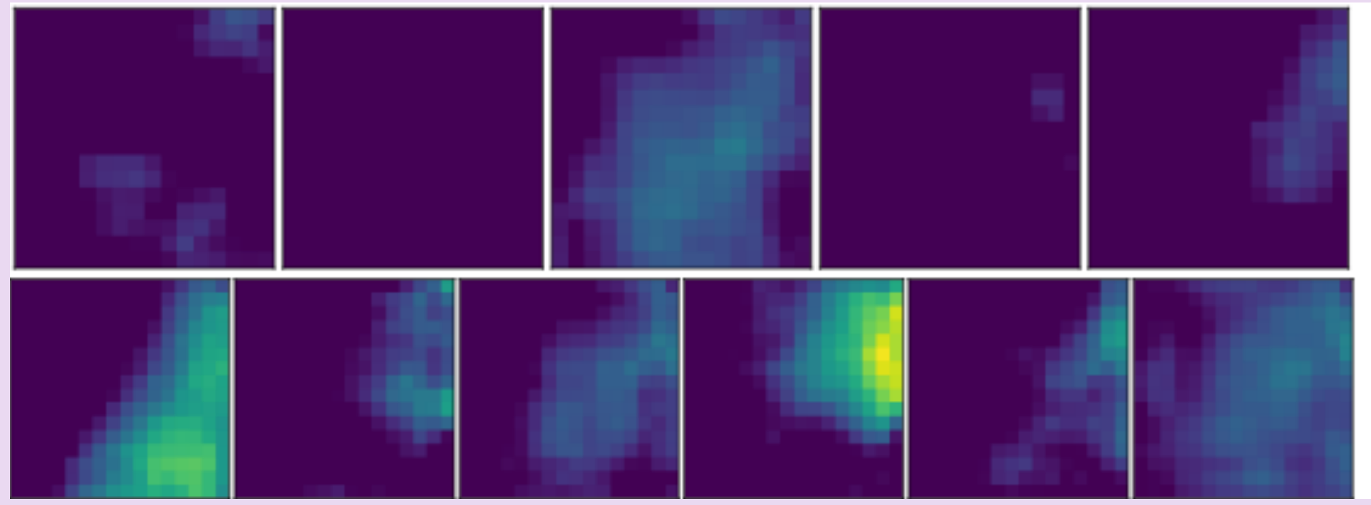
Multi-task Gaussian Process Prediction, Bonilla et al, NeurIPS, '07

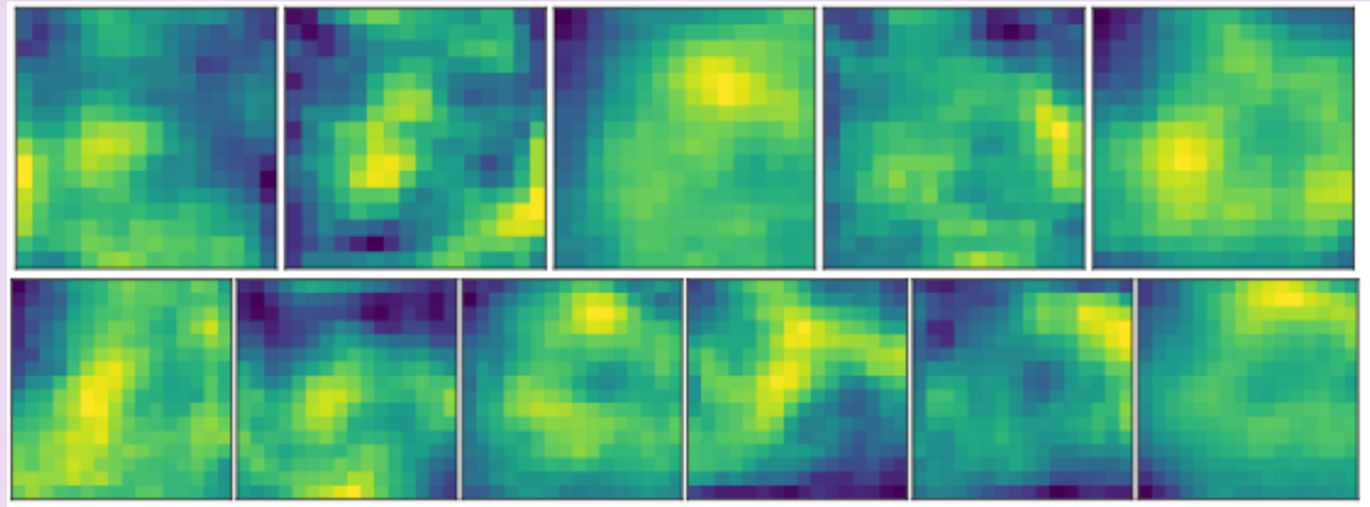
Efficiently sampling functions from GP Posteriors, Wilson et al, ICML, '20

Geostatistics for natural resources evaluation, Goovaerts, '94

Scalable High-Order Gaussian Process Regression, Zhe, AISTATS, '19

Un-optimized images from an optical design problem





Optimized images

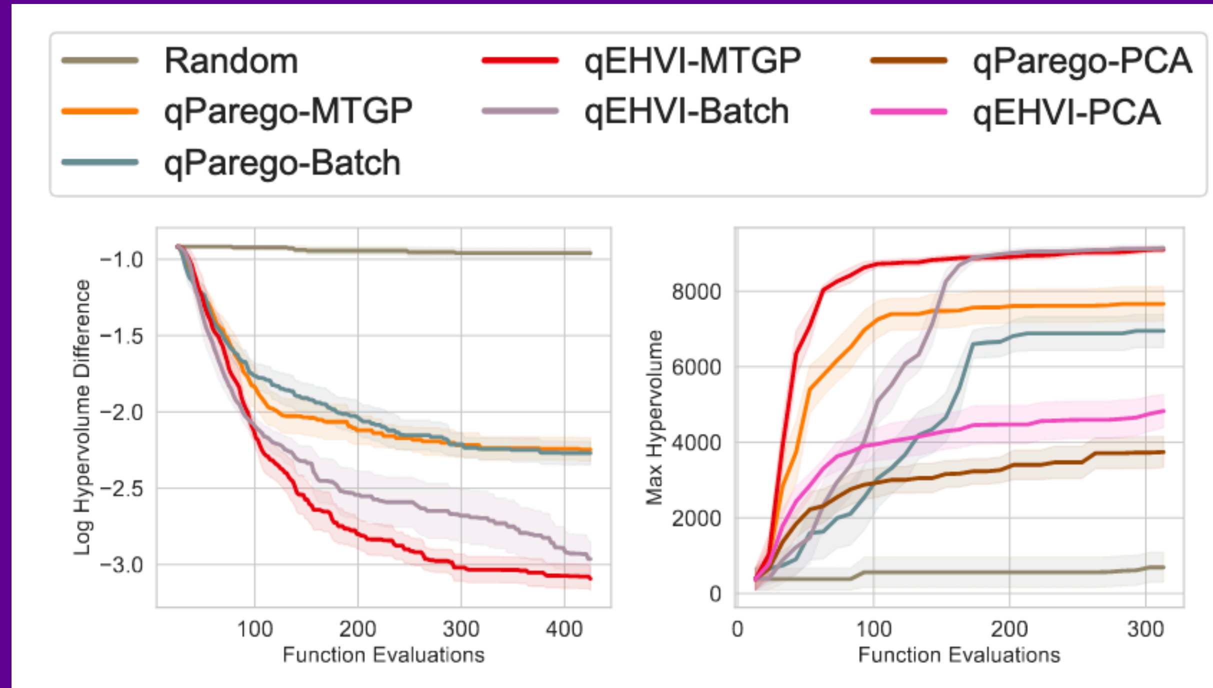
"Optimizing High-Dimensional Physics Simulations via Composite Bayesian Optimization" https://arxiv.org/abs/2111.14911

Multi-task Gaussian process posterior distributions cost n^3t^3 to sample from, preventing their usage in many Bayesian optimization problems.

Using Matheron's rule to instead sample from the MTGP posterior distribution costs $n^3 + t^3$ time.

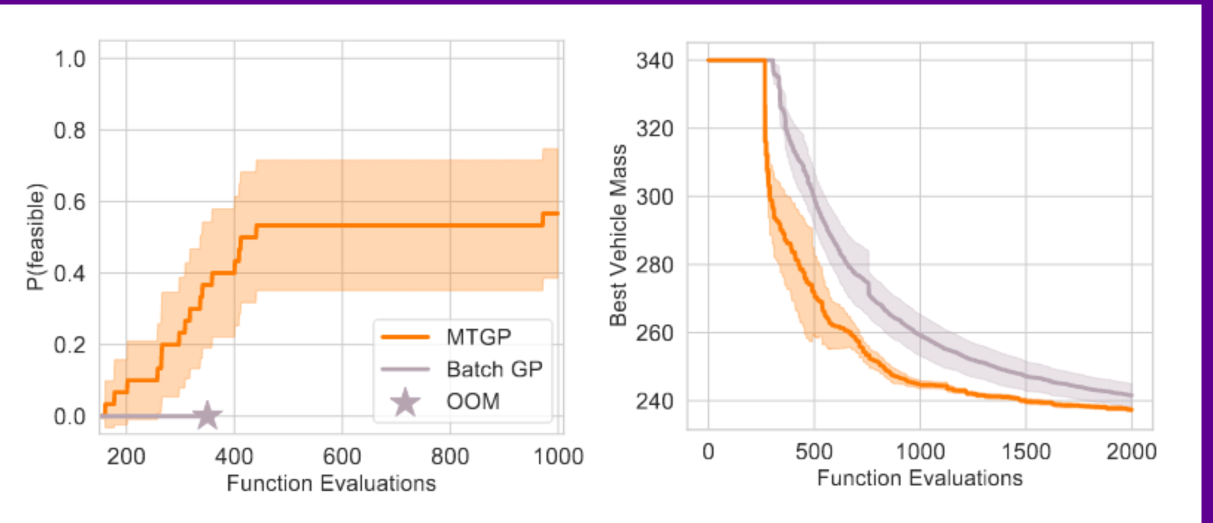
Enables **Bayesian optimization with tens of thousands of outputs** vs tens of outputs.

Experiments



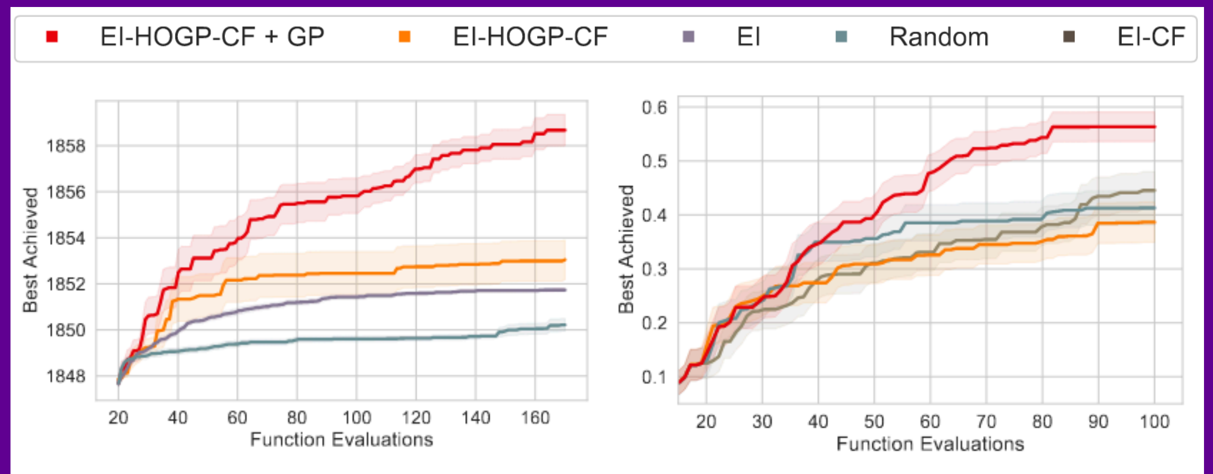
(a) C2DTLZ2, $q = 2$ (b) OSY, $q = 10$

MTGPs are better for constrained multi-objective problems and scale to large batch sizes.



(b) Feasibility, Lunar Lander, $m = 100$ (d) Vehicle mass, MOPTA08, $m = 68$

MTGPs are more memory efficient for large scale Thompson sampling & perform better.



(c) Cell-tower coverage $(t = 2 \times 50 \times 50, d = 30)$ (d) Optics $(t = 16 \times 64 \times 64, d = 4)$

The HOGP (a more structured MTGP) enables scaling to problems with >50,000 output dimensions.