Active Learning for Level Set Estimation

Alkis Gotovos, Nathalie Casati, Gregory Hitz and Andreas Krause

ETH Zurich

IJCAI-13

Problem

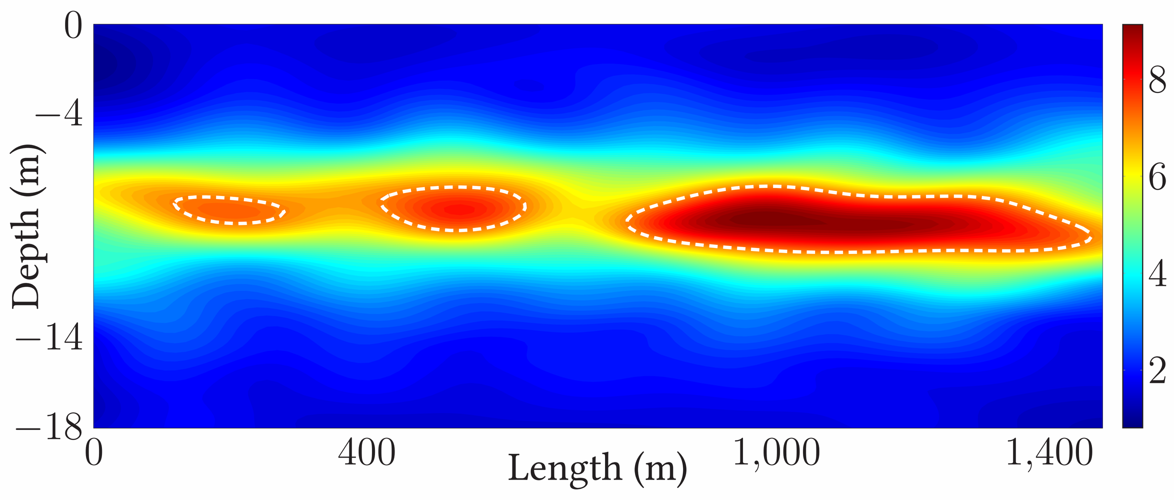
We would like to determine the regions where the value of some unknown function lies above or below a given threshold level.

The above can be posed as a classification problem (into super- and sublevel sets) with *sequential* measurements, which are assumed to be *expensive* and *noisy*.

Example applications

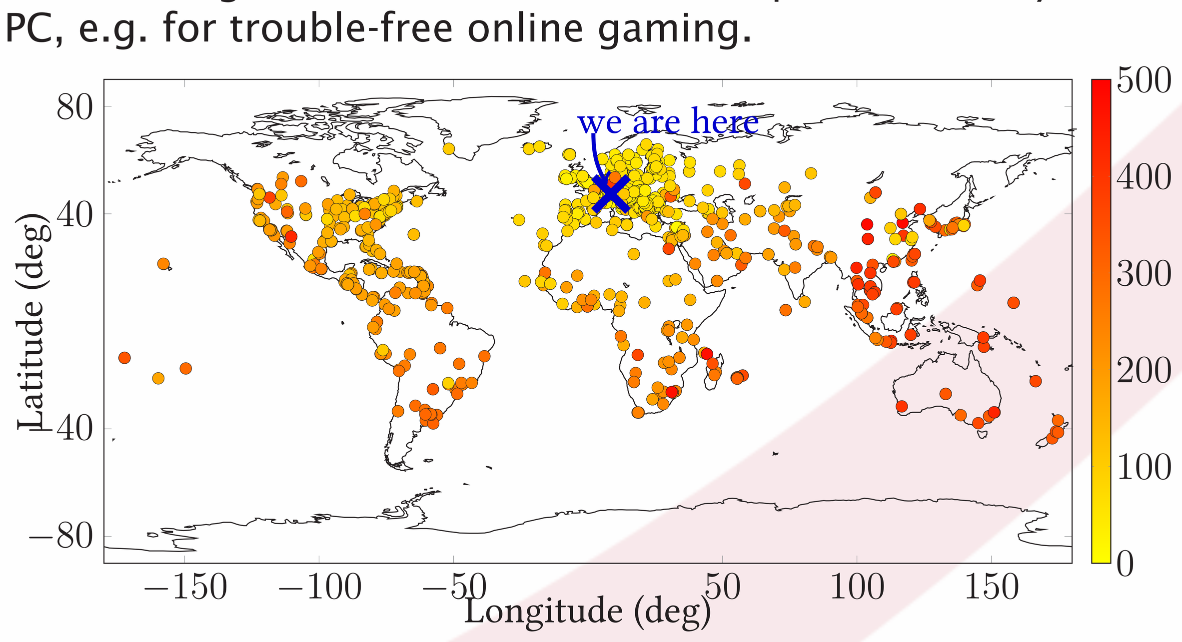
Environmental monitoring

Estimate regions of (a vertical transect of) Lake Zurich where chlorophyll/algal concentration is “abnormally high”.



Geolocating internet latency

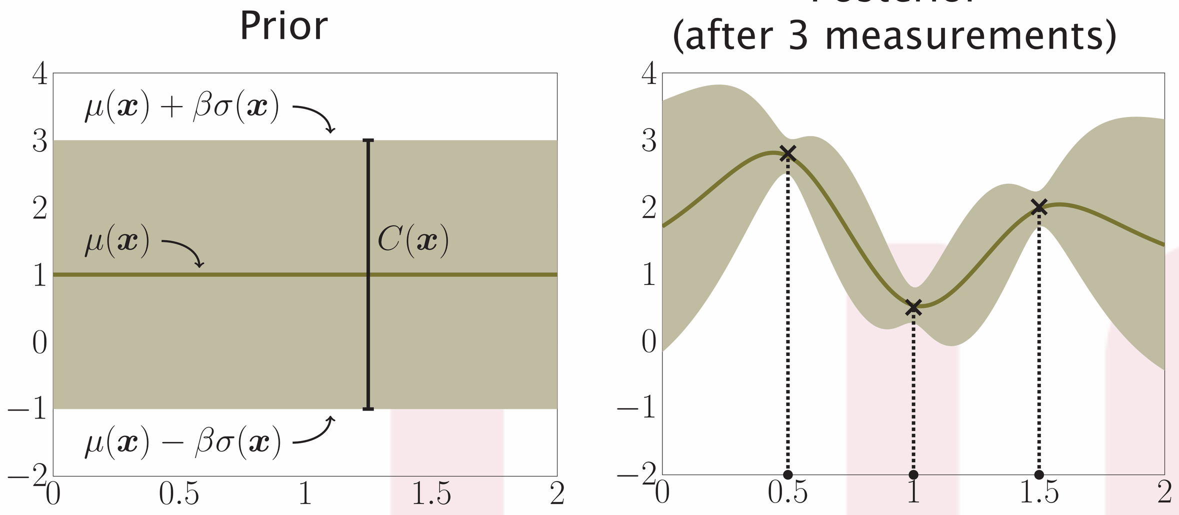
Estimate regions of the world with “acceptable” latency to our PC, e.g. for trouble-free online gaming.



Gaussian processes

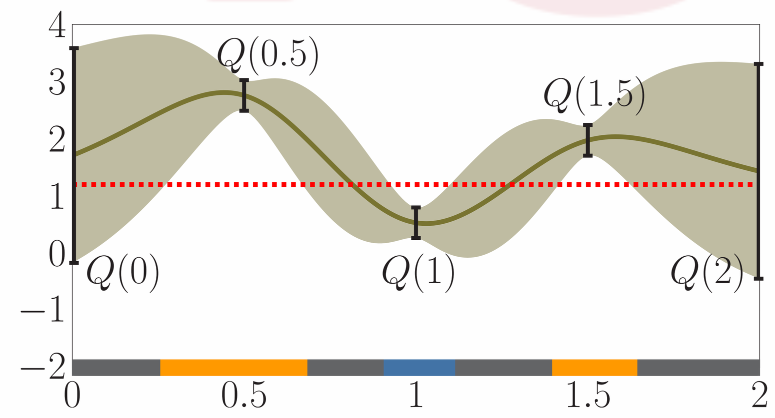
Estimation

Given some measurements, GPs provide *mean and variance* estimates of the unknown function, allowing us to construct *confidence intervals* at each point.



Classification

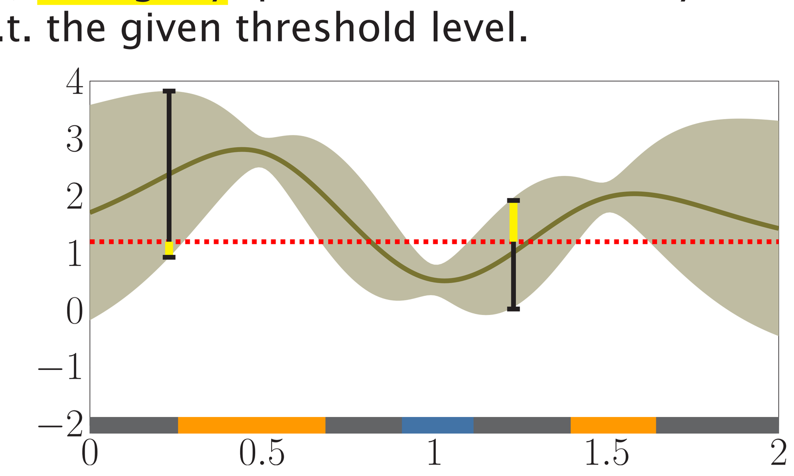
For each point, we use the GP-derived confidence intervals to either classify it into the super- or sublevel sets, or leave it unclassified.



Measurement selection

To obtain informative measurements w.r.t. the problem at hand, at each iteration we select the most *ambiguous* point among the yet unclassified to be measured.

Intuitively, ambiguity quantifies our difficulty in classifying a point w.r.t. the given threshold level.



The LSE algorithm

Given a set of points (e.g. fine grid of the unknown function’s domain) and a threshold level h, our proposed Level Set Estimation (LSE) algorithm iteratively *samples* and *classifies* based on GP-derived confidence intervals.

while there exist unclassified points
- **for each** unclassified point
  - **if** its confidence interval lies above h **then** classify into superlevel set
  - **else if** its confidence interval lies below h **then** classify into sublevel set
  - **else** leave unclassified

  CLASSIFY
- select max. ambiguity (yet unclassified) point
- obtain a noisy measurement at that point

  SAMPLE
- perform GP inference and update conf. intervals

Fine print

- We enforce monotonically shrinking confidence intervals
- We relax classification by an accuracy parameter ϵ

Sample complexity bound

Theorem

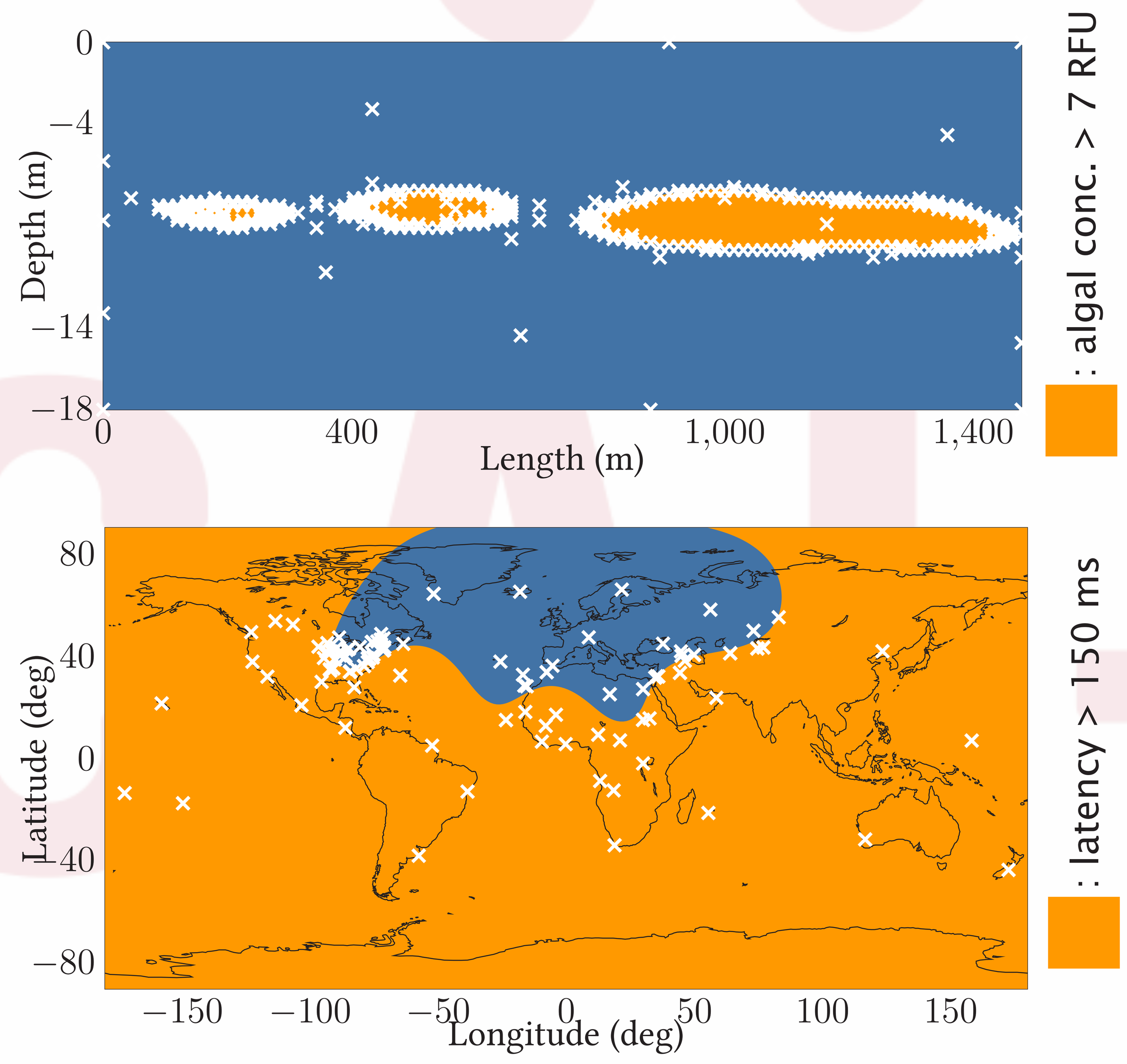
For any $h \in \mathbb{R}$, $\delta \in (0,1)$, and $\epsilon > 0$, if $\beta_t = 2\log(|D|\pi^2 t^2/(6\delta))$, LSE terminates after at most T iterations, where T is the smallest positive integer satisfying

$$\frac{T}{\beta_T \gamma_T} \geq \frac{C_1}{4\epsilon^2},$$

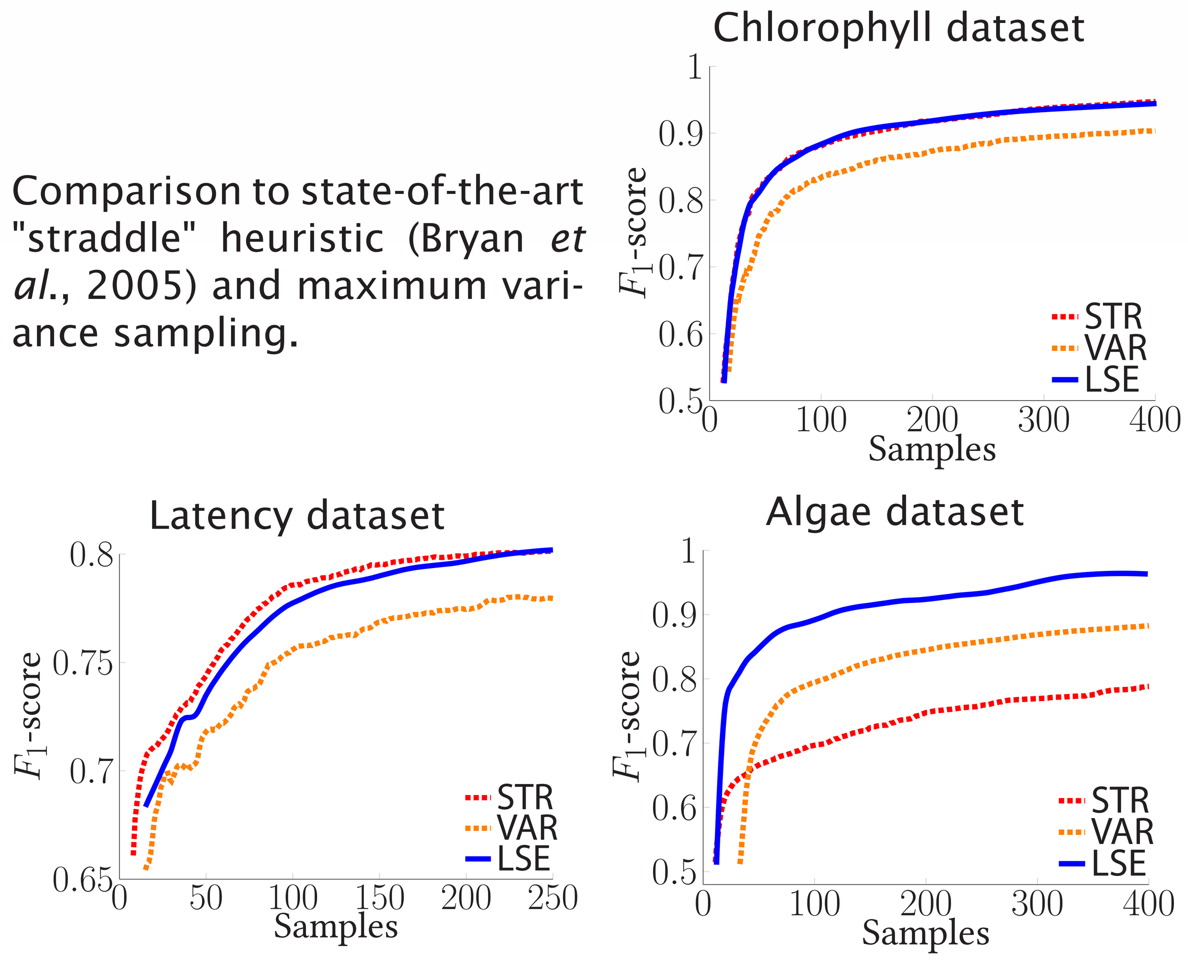
where $C_1 = 8/\log(1+\sigma^{-2})$.
Furthermore, with probability at least $1-\delta$, the algorithm returns an ϵ-accurate solution, that is

$$\Pr\left\{\max_{x \in D} \ell_h(x) \leq \epsilon\right\} \geq 1-\delta.$$

Experimental results



Comparison to state-of-the-art "straddle" heuristic (Bryan *et al.*, 2005) and maximum variance sampling.



Extension 1: Implicit threshold level

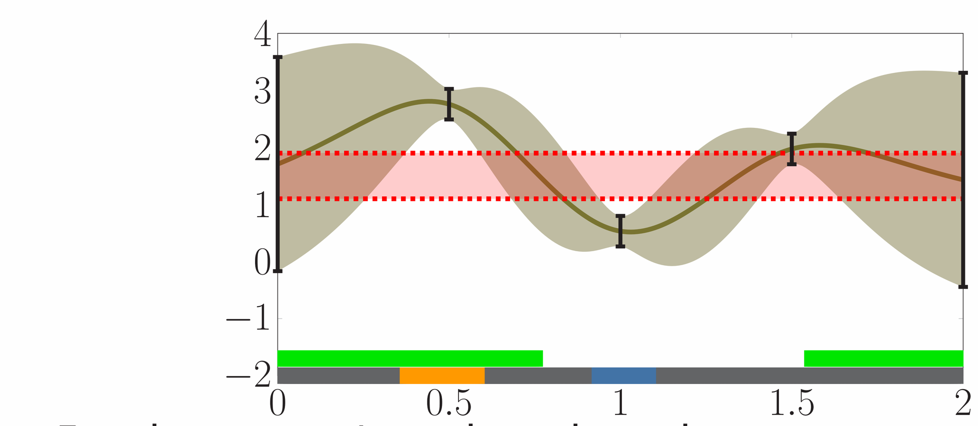
What if we do not have a predefined threshold level h? For example, we want to determine relative "hotspots" of algal concentration.

Implicitly defined thr. level: $h = \omega \max f(x), \ 0 < \omega < 1$

For this setting, we propose the LSE_{imp} algorithm with similar theoretical guarantees to LSE.

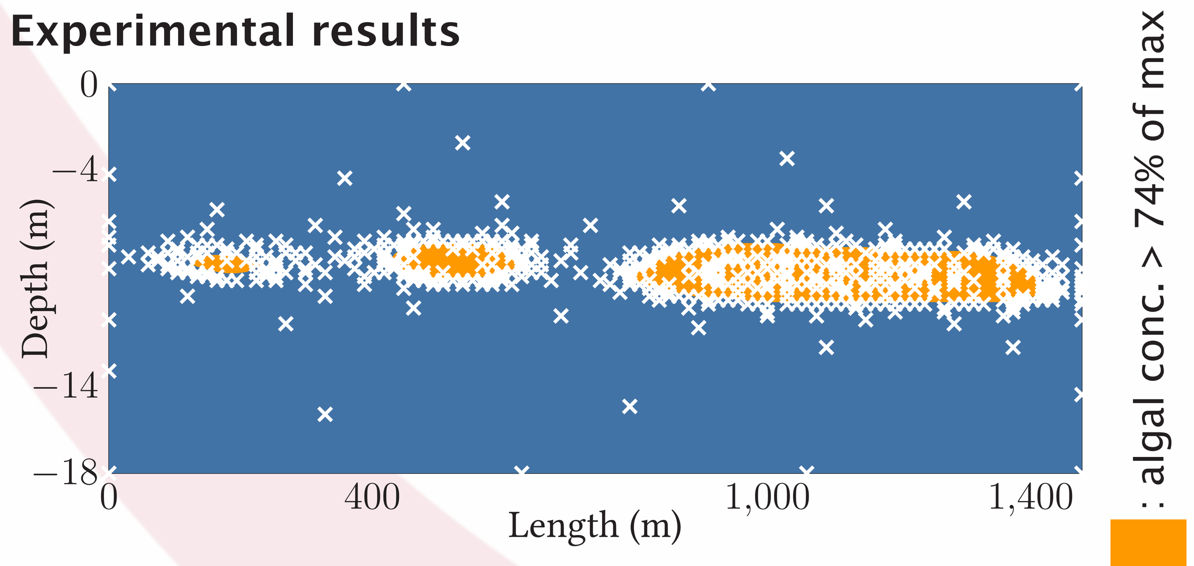
Main novelties of LSE_{imp}:

- h is now an estimated quantity with associated uncertainty, which leads to slower classification.

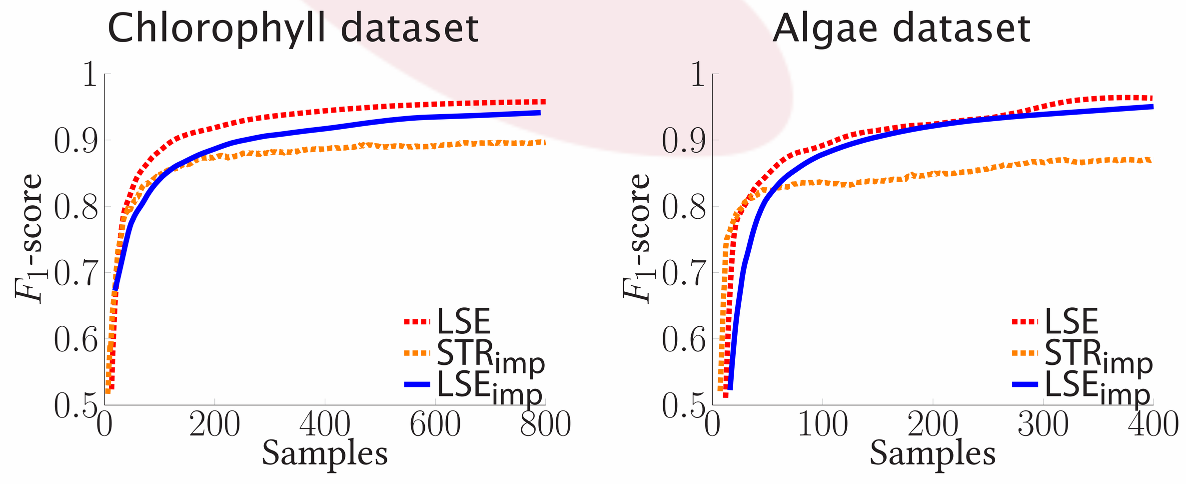


- For the uncertainty about h to decrease we need to accurately estimate the function maximum, therefore we need to keep sampling at regions where the maximum may lie.

Experimental results



Comparison to LSE and to a naive extension of "straddle" for implicit threshold levels.



Extension 2: Batch sampling

We propose the $\text{LSE}_{\text{batch}}$ extension of LSE, which, instead of selecting a single measurement at each iteration, selects a *batch* of B of them at a time.

Why?

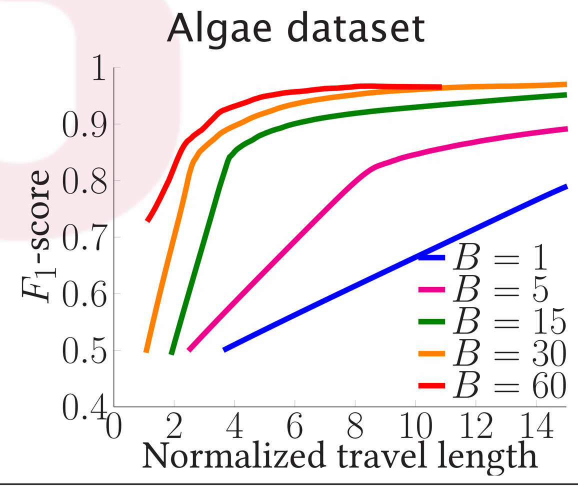
Latency geolocation

Send multiple ping requests in parallel at essentially the same cost as a single request, thus increasing sampling throughput.

Environmental monitoring

Reduce the total traveling distance by planning ahead:

- Select a batch of sampling locations.
- Connect them using a Euclidean TSP path.
- Traverse path and collect measurements.



Extra: Proof outline of LSE bound