## Washington University in St. Louis



## **Active Pointillistic Pattern Search**

## Yifei Ma<sup>\*</sup>, Dougal J. Sutherland<sup>\*</sup>, Roman Garnett, Jeff Schneider

\*: equal contribution



#### Motivation

- <u>Active search</u>: filter as many positives as possible.
- <u>Pointillism</u>: point observations, group patterns.
- <u>Flexibility</u>: allow for arbitrarily defined patterns.

#### Applications

- Environment: autonomous boats searching a pond for polluted areas.
- Astronomy: choosing where to point a telescope to find interesting objects.
- Polling: carefully surveying to find electoral races that need attention.





#### Problem Setup

0	Inputs:	$\mathbf{Prior}f$	$\sim \mathcal{GP}(\mu(\cdot),\kappa(\cdot,\cdot))$
		( <b>1</b> , <b>1</b> )G	
	Regions	$\{\mathcal{X}_g\}_{g=1}^{G}$	Classifier $h(f; \mathcal{X}_q)$

### Algorithm

Choose point to greedily maximize expected reward  $\max_{x_*} \mathbb{E}_{y_*} \left[ \sum_{g \in \mathcal{G}_t} \mathbb{1} \left\{ \mathbb{E} \left[ h(f; \mathcal{X}_g) \mid x_{1:t}, x_*, y_{1:t}, y_* \right] > \theta \right\} \right]$ 



#### Select locations to observe function values

# $(f(x_1)) f(x_2)$ $(f(x_n))$ $h(f; \mathcal{X}_1)$ $h(f; \mathcal{X}_2)$ $h(f; \mathcal{X}_q)$

2 Collect reward for region matches  $r_t = \sum_{q} \mathbb{1} \{ \mathbb{E} [h(f; \mathcal{X}_g) \mid x_{1:t}, y_{1:t}] > \theta \}$ 

#### Estimate expected reward with Monte Carlo:

## Sample observation $z_* \sim \mathcal{N}\left(\mu_t(x_*), \kappa_t(x_*, x_*) + \sigma^2\right)$

Sample enough of *f* to get *h* output  $f \sim \mathcal{GP}\left(\mu_*, \kappa_*\right)$ 

#### → Analytical form:

If  $h_g(f) = \Phi(L_g f + b)$  where  $\Phi$  is normal cdf and  $L_g$  is linear, e.g.  $L_g f = \int_{x \in \mathcal{X}_g} w_g(x)^\mathsf{T} f(x) \, \mathrm{d}x$ then expected reward has a closed form:

 $\mathbb{E}_{y_*} r_* = \sum_g \Phi \left( \frac{L_g \mu_t + b - \sqrt{1 + L_g^2 \kappa_* \Phi^{-1}(\theta)}}{\left| L_g \left[ \kappa_t(\cdot, x_*) \right] \right| / \sqrt{\kappa_t(x_*, x_*) + \sigma^2}} \right)$ 

If using probit-linear classifiers and regions are independent: Analysis

The algorithm picks regions with a certain tradeoff between:

- Ratio of variance reduction with one more point.
- High posterior mean.

Within a region, it picks the point that:

#### Finding Vortices with a Black-Box Classifier

## We want to find vortices in a 2d map of fluid flow by

Used a 2-layer neural network learned

#### observing point vectors. Regions are overlapping squares.



The velocity dataset; each arrow represents the mean of a 2x2 square. This run was initialized with the points at the green circles and selected the ones at the red circles.

#### from a small training set:

	1111777777777777777777777	
1	111177777	
1,		· · · / / / / / / / / / /
11	112222	
A - XXXXXXX	1111	
1112		11111
111		
/ MARGERENN		
		1111111
1. EEVVEENN		
		11111
······································	555-11111	1111111
······································		11111111
······································		11111111



Mean and standard error of recall for matching regions, over 15 runs. True labels are determined by using the classifier on the full velocity dataset.

- Is most correlated to region's label.
- Equivalently, most increases the power of the hypothesis test.

#### Finding Polluted Regions



This is the actual data of dissolved oxygen (DO) measurements that is densely-collected from a pond. We define regions to be the rectangles with black borders that cover the map; the real polluted regions are colored in black.



#### Finding Winning Districts

By doing precinct-level ("point") polling, can we find districts (green bordered regions) where we'll win?



Using demographic data of each precinct, made Gaussian process for all precincts. District result will be an average from its precincts, weighted by number of voters. The classifier is probit-linear.



Recall for election prediction. Color bands show standard errors after 15 runs. Uneven weights cause uncertainty sampling to underperform random.