Washington University in St. Louis

Carnegie Mellon University

Active Pointillistic Pattern Search

Yifei Ma*, Dougal J. Sutherland*, Roman Garnett, Jeff Schneider

*: equal contribution

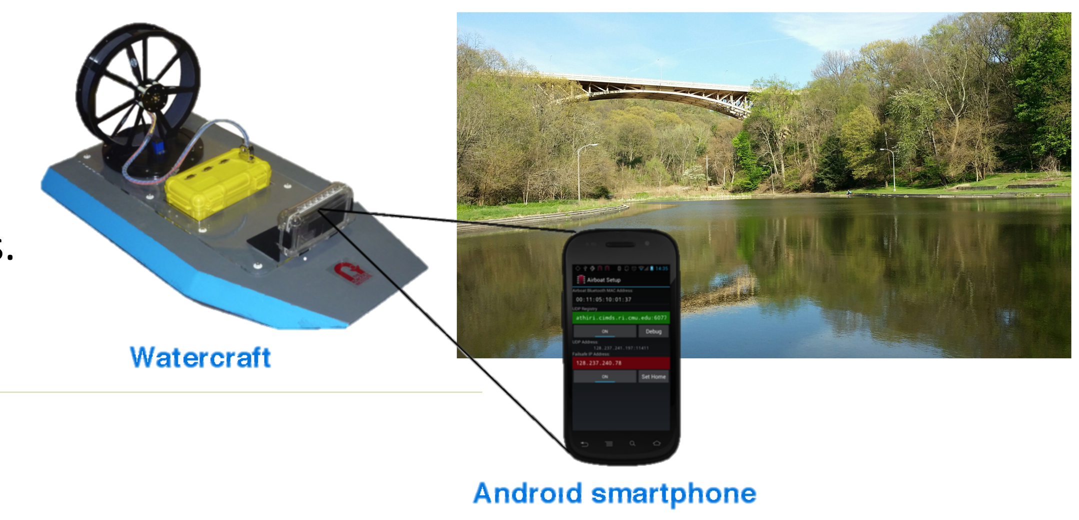
Auton Lab

Motivation

- Active search: filter as many positives as possible.
- Pointillism: point observations, group patterns.
- Flexibility: 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.

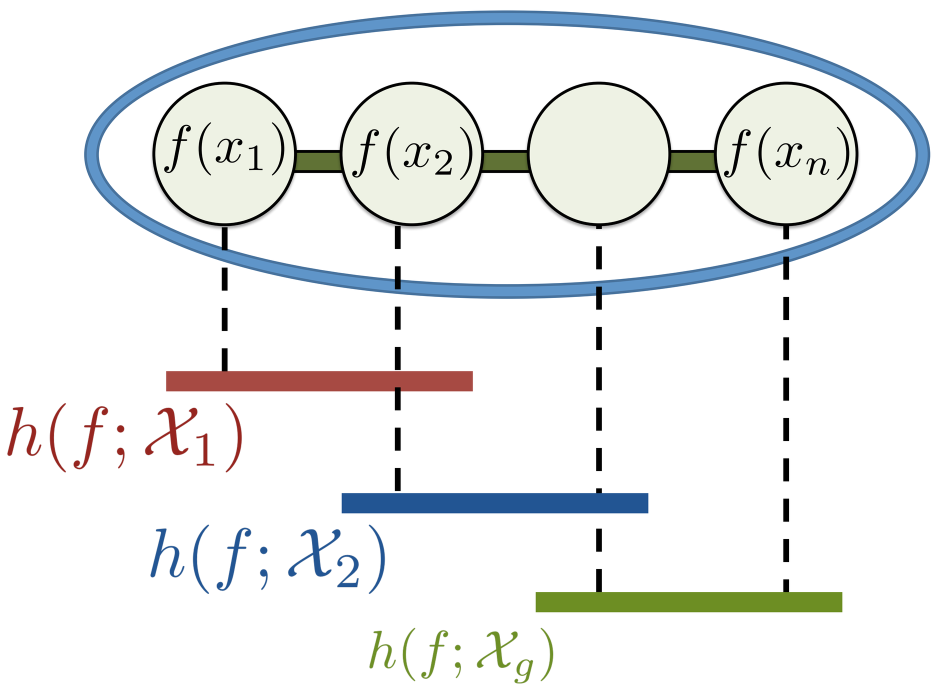


Problem Setup

0 Inputs: Prior $f \sim \mathcal{GP}(\mu(\cdot), \kappa(\cdot,\cdot))$

Regions $\{\mathcal{X}_g\}_{g=1}^G$ Classifier $h(f; \mathcal{X}_g)$

1 Select locations to observe function values



2 Collect reward for region matches

$$r_t = \sum_g \mathbb{1}\{\mathbb{E}[h(f;\mathcal{X}_g) \mid x_{1:t}, y_{1:t}] > \theta\}$$

Algorithm

Choose point to greedily maximize expected reward

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

Estimate expected reward with Monte Carlo:

Sample observation

$$z_* \sim \mathcal{N}\Big(\mu_t(x_*),\ \kappa_t(x_*, x_*) + \sigma^2\Big)$$

Sample enough of f to get h output

$$f \sim \mathcal{GP}(\mu_*, \kappa_*)$$

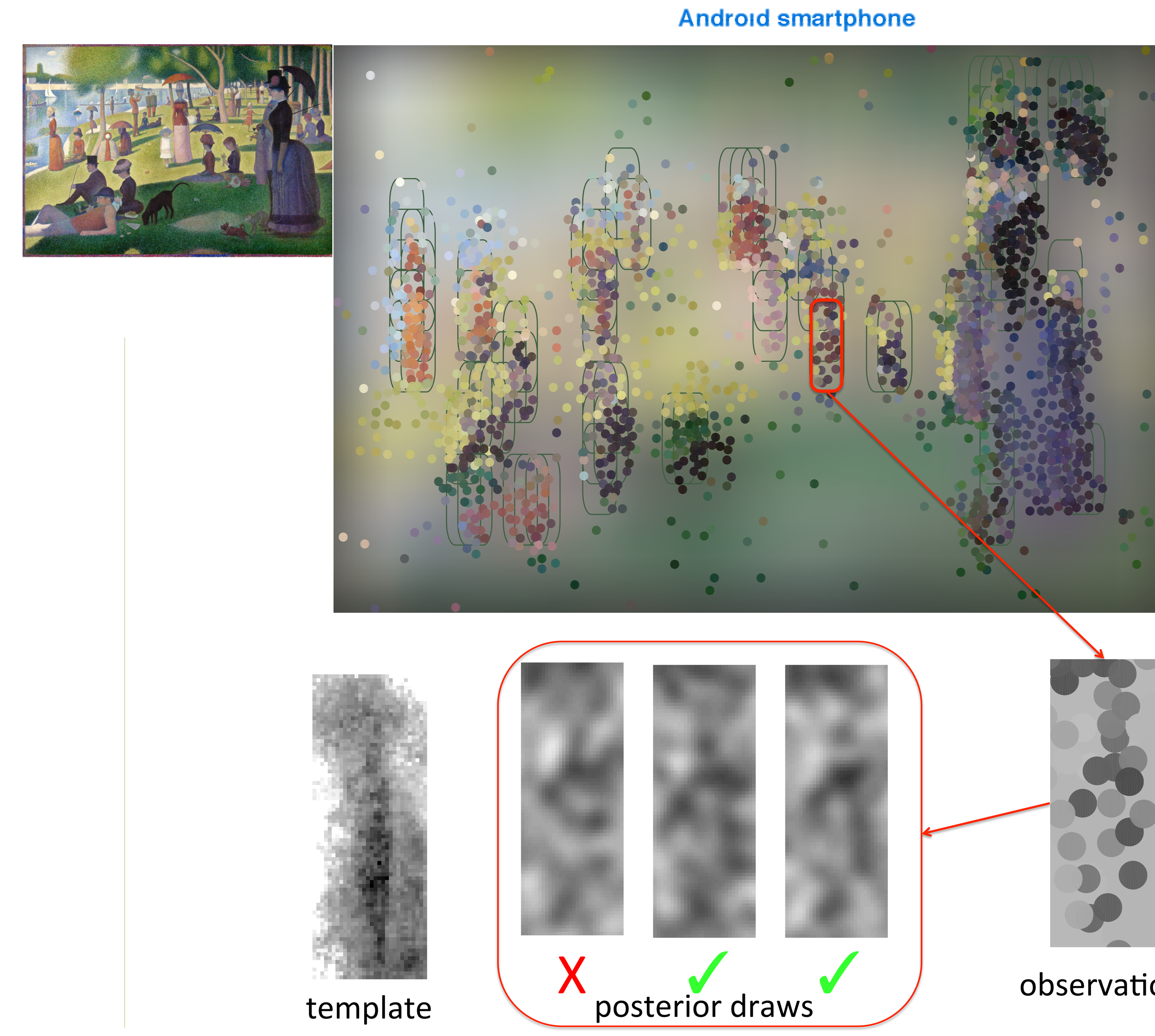
Analytical form:

If $h_g(f) = \Phi\Big(L_g f + b\Big)$ where Φ 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:

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



Analysis

If using probit-linear classifiers and regions are independent:

The algorithm picks regions with a certain tradeoff between:

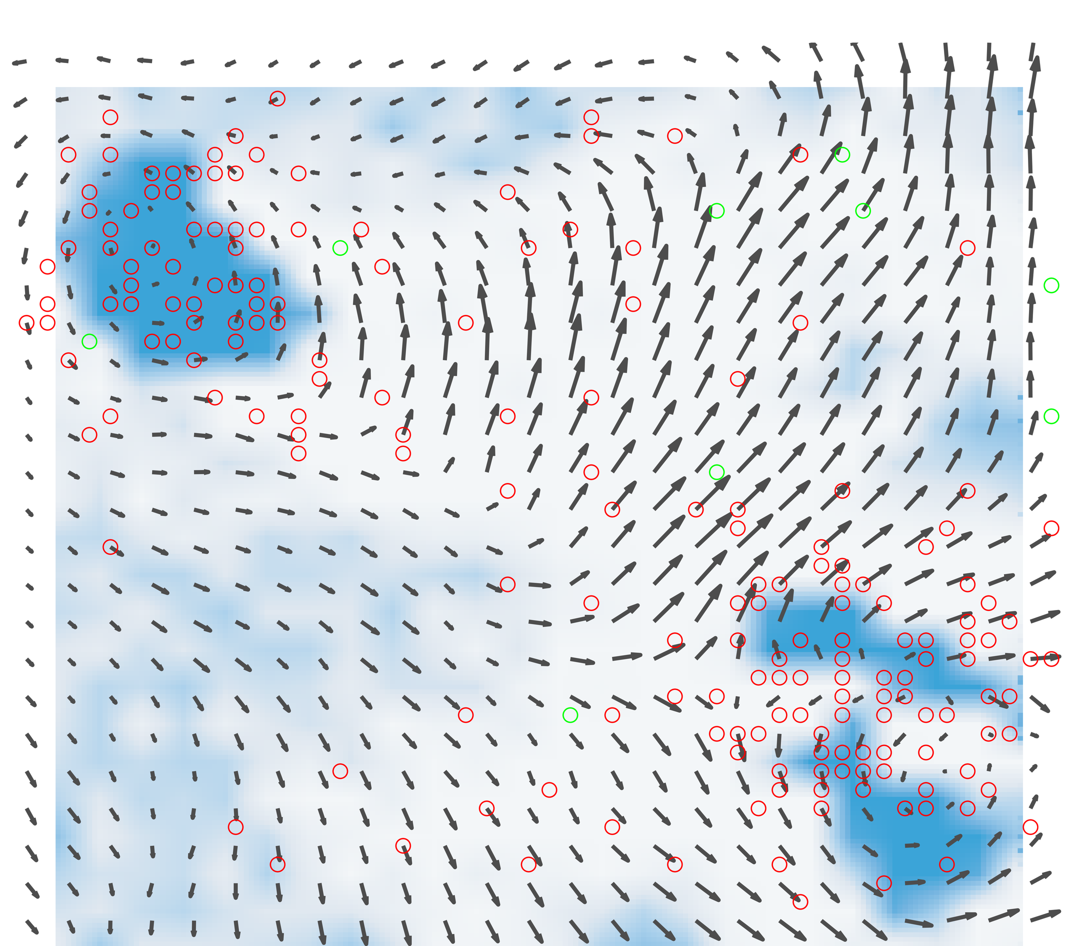
- Ratio of variance reduction with one more point.
- High posterior mean.

Within a region, it picks the point that:

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

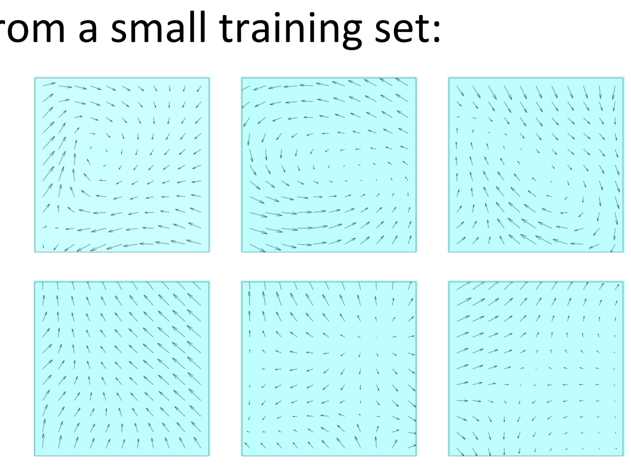
Finding Vortices with a Black-Box Classifier

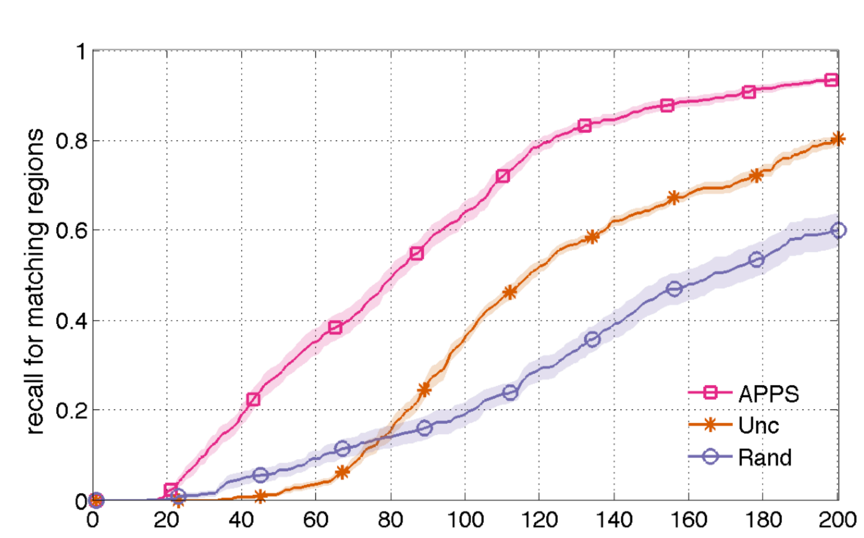
We want to find vortices in a 2d map of fluid flow by observing point vectors. Regions are overlapping squares.



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.

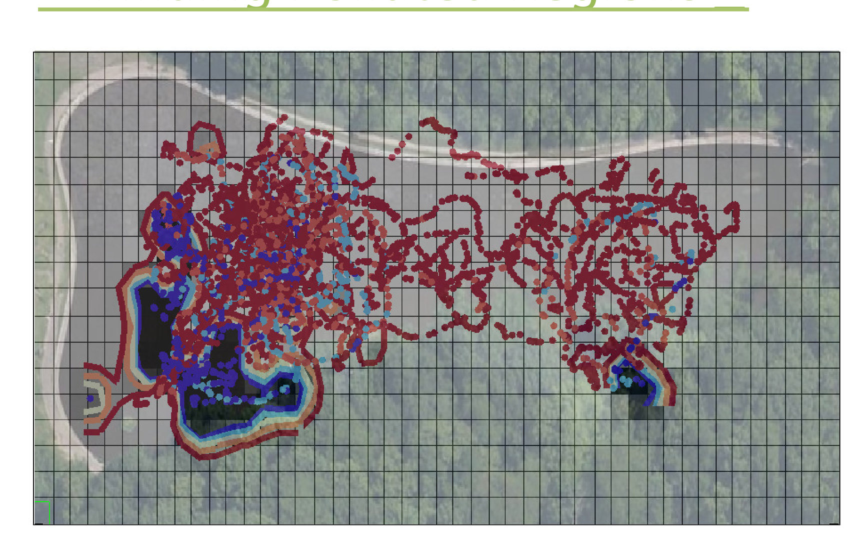
Used a 2-layer neural network learned from a small training set:





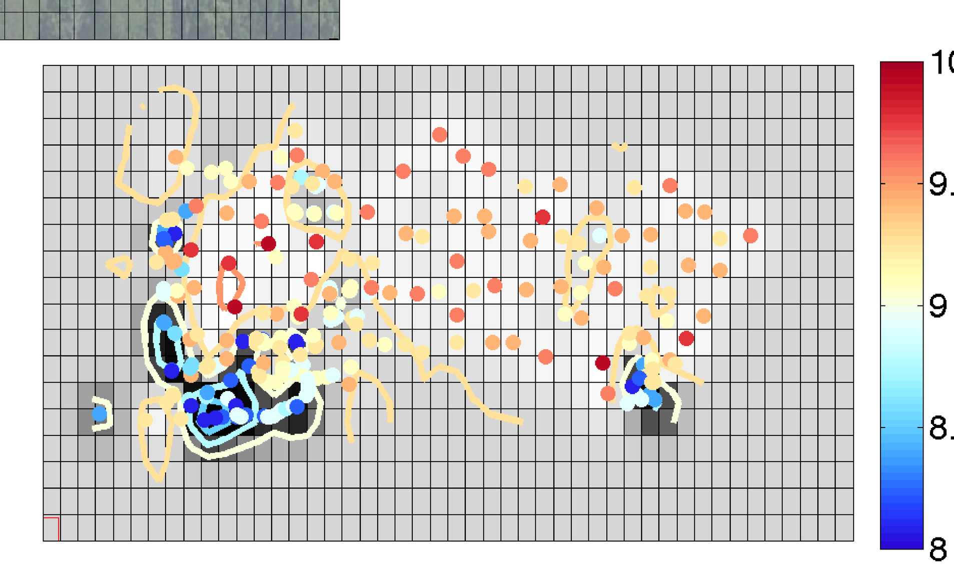
Mean and standard error of recall for matching regions, over 15 runs. True labels are determined by using the classifier on the full velocity dataset.

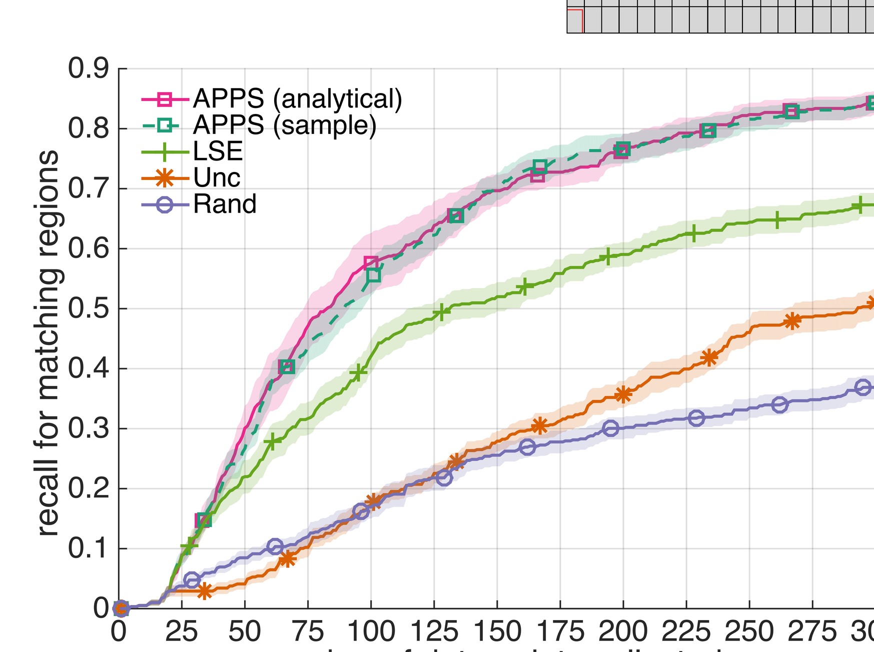
Finding Polluted Regions



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.

We could recall 75% of the polluted regions (black) with 200 actively collected measurements. The dot colors indicate DO values.

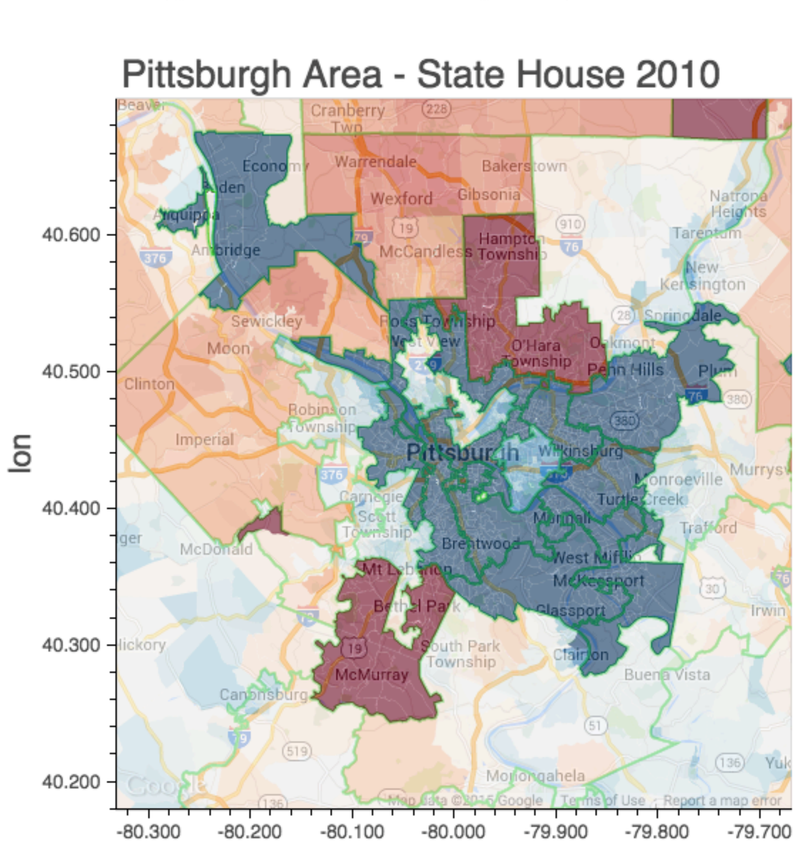




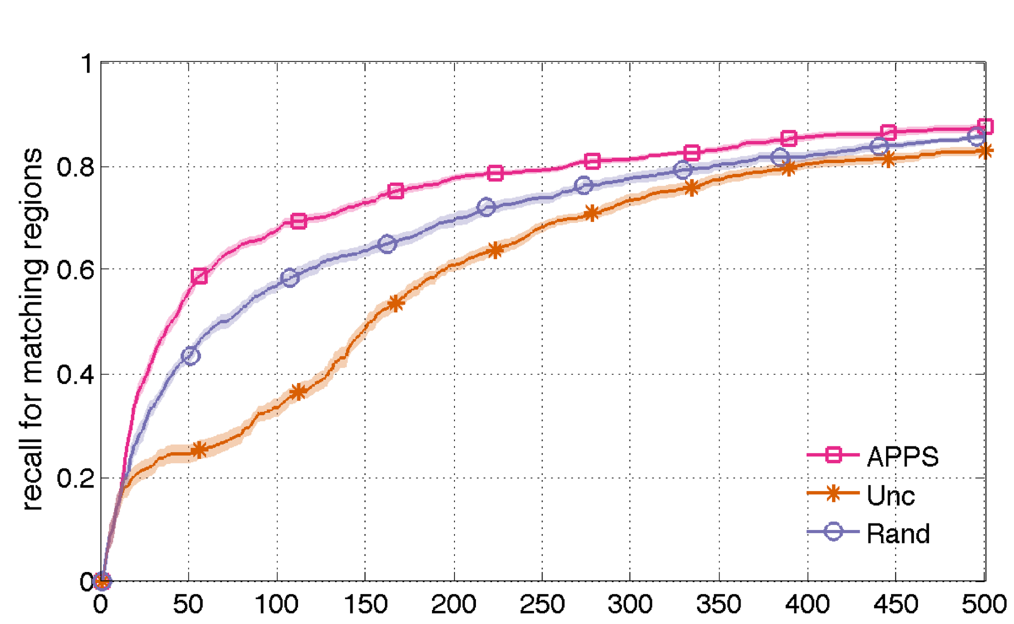
Recall curves for pond monitoring experiment. Shaded bands show standard errors after 15 runs.

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.

Related Work

Most Bayesian optimization:

- Models functions with GPs.
- Maximizes observable point values.

Active search (e.g. Garnett et al., ICML 2012)

- Usually assumes that labels are directly observable and correspond to single points.

Active Area Search (Ma et al., AISTATS 2014)

- Similar setup and algorithm, but can only detect thresholds on mean of a region.
- APPS generalizes to *any* pattern.

Level set estimation (Gotovos et al., IJCAI '13; Low et al., AAMAS '12)

- Actively finds a particular level set in a function.
- Related to AAS; can't model arbitrary patterns.