ML/AI for the Environment - Tutorial

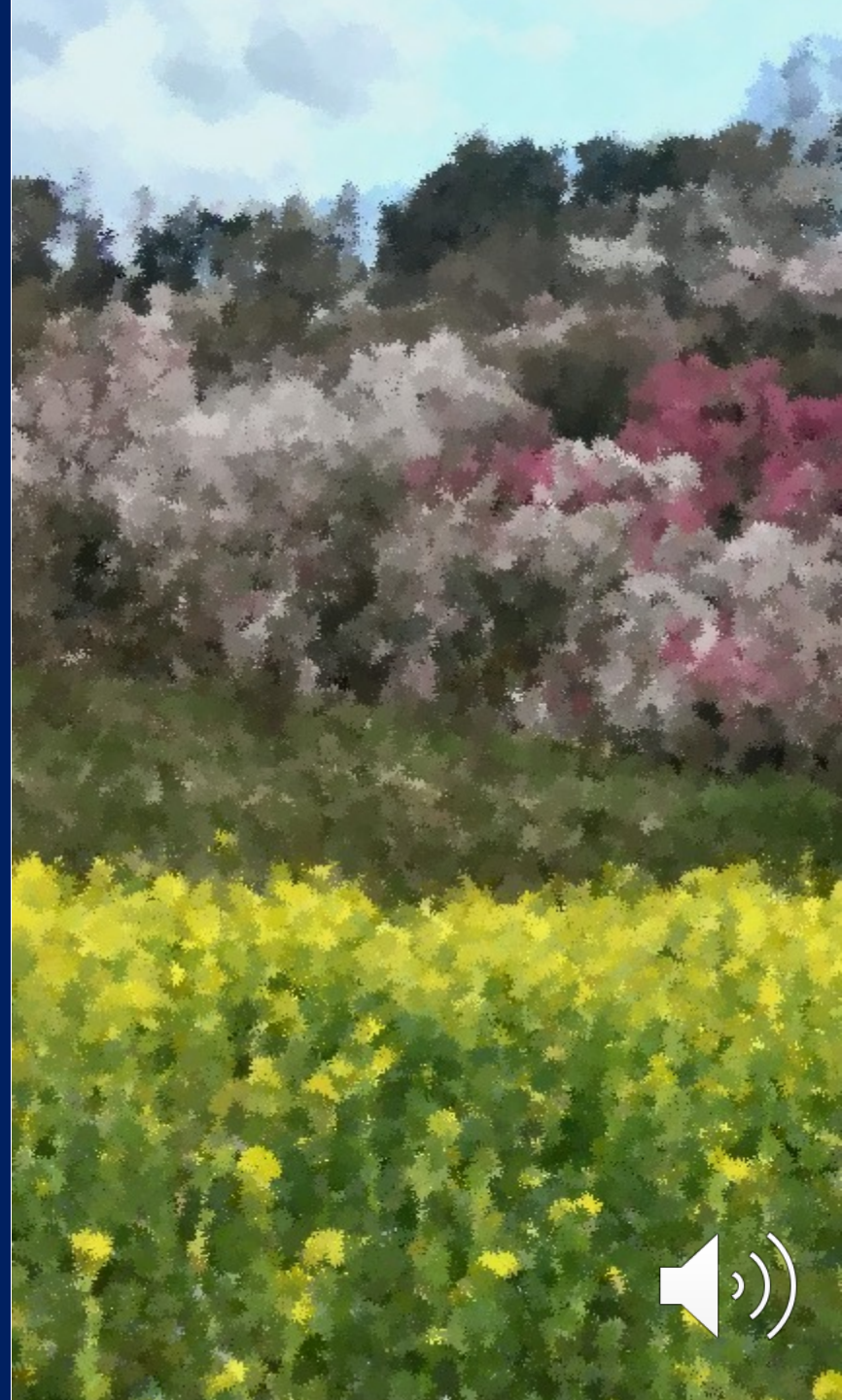
Haruko Wainwright

Nuclear Science and Engineering; Civil and Environmental Engineering

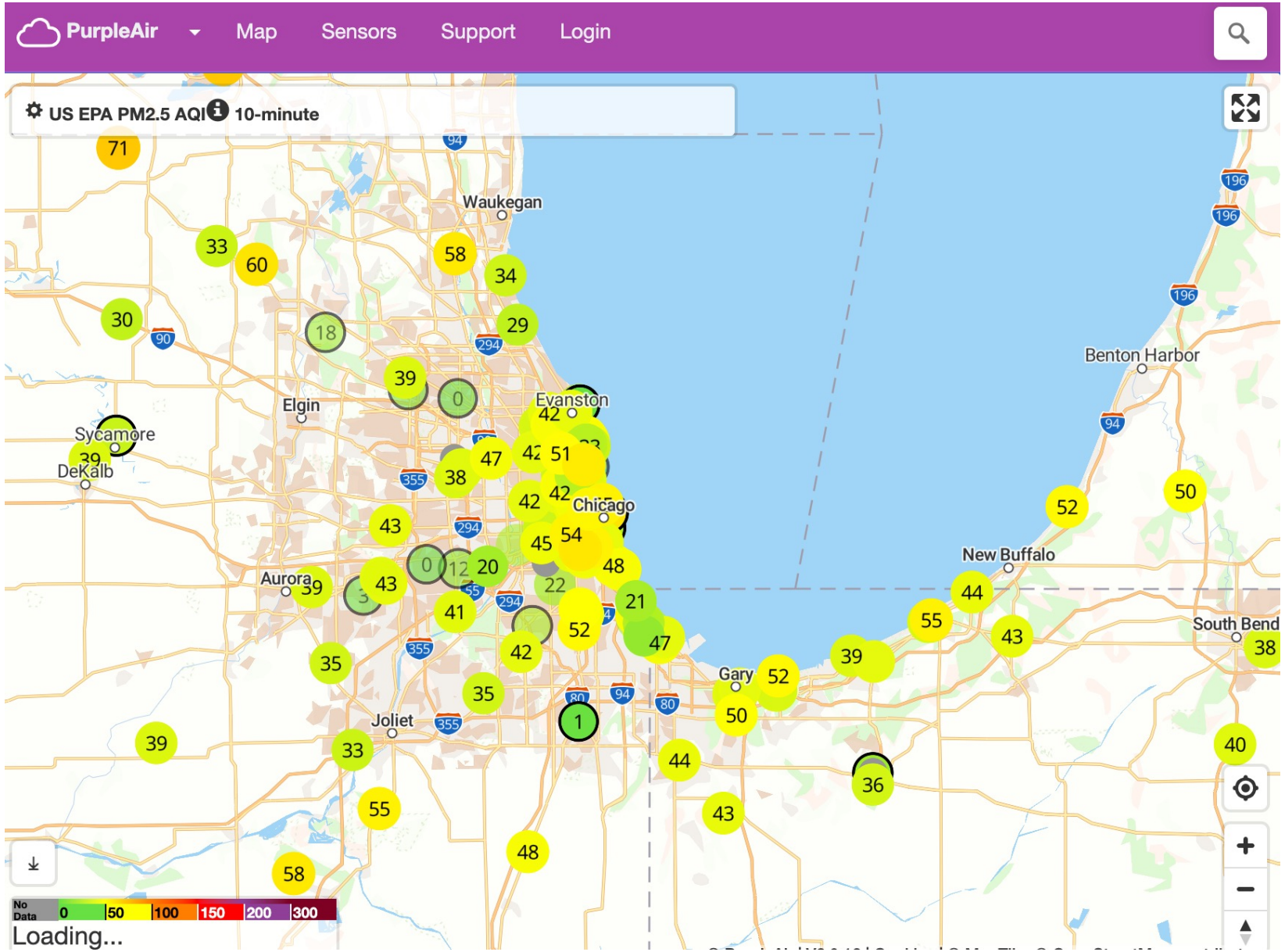
Massachusetts Institute of Technology

NSE
Nuclear Science & Engineering at MIT
science : systems : society

Civil and Environmental Engineering



Tutorial: Air Quality Data Integration with GP



Purple Air Sensor: Perspective

I created the worst air in the world! We started cooking a pizza for kids' lunch.

Air purifier turned on

Air Quality Index

AQI Basics for Ozone and Particle Pollution

Daily AQI Color	Levels of Concern	Values of Index	Description of Air Quality
Green	Good	0 to 50	Air quality is satisfactory, and air pollution poses little or no risk.
Yellow	Moderate	51 to 100	Air quality is acceptable. However, there may be a risk for some people, particularly those who are unusually sensitive to air pollution.
Orange	Unhealthy for Sensitive Groups	101 to 150	Members of sensitive groups may experience health effects. The general public is less likely to be affected.
Red	Unhealthy	151 to 200	Some members of the general public may experience health effects; members of sensitive groups may experience more serious health effects.
Purple	Very Unhealthy	201 to 300	Health alert: The risk of health effects is increased for everyone.
Maroon	Hazardous	301 and higher	Health warning of emergency conditions: everyone is more likely to be affected.

github.com/hmwainw/GP4AQ

GP4AQ

These Jupyter notebooks demonstrate the data integration of EPA and Purple air sensor data and plume simulation results, using the Gaussian Process regression, for interpolating and mapping the air quality index over space.

Note:

- You have to have purpleair_chicago.txt in the Google Drive root directory.
- The same functions appear in multiple notebooks. These notebooks are for the demonstration/teaching purpose.

1. Define the grid and domain for spatial estimation
2. Download EPA air quality data
3. Download Purple air quality data
4. Download plume simulation maps
5. Interpolate EPA air quality data
6. Interpolate Purple air quality data
7. Integrate EPA and Purple air quality data
8. Integrate EPA, Purple air, and plume simulation data

Tutorial: Gaussian Process x Mapping

- **Gaussian process**
- **Kernels**
- **Non-stationary GP**
- **Multi-type Multiscale data integration**
- **Air quality data interpolation**
  - Python mapping and Earth Science suits

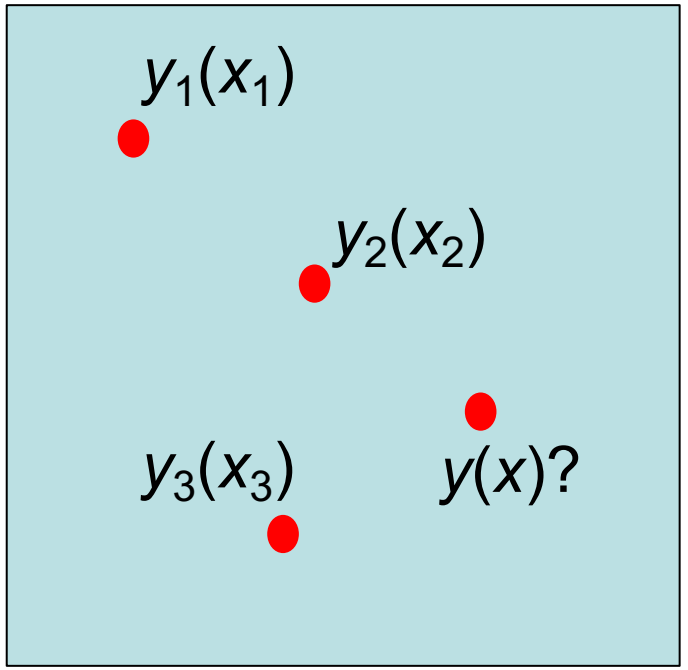
Gaussian Process: Geostatistics

Interpolation in a physical space

$(x_i, y_i) : i = 1, \ldots, n$

x_i : locations (2D, 3D vector)

y_i: measured variables



- **Multivariate Gaussian distribution**

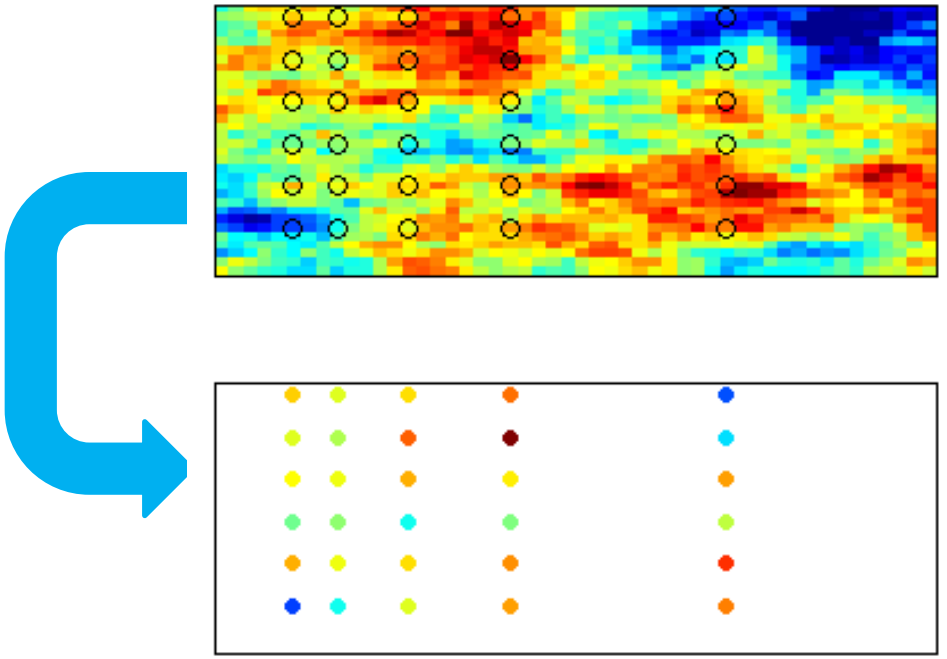
$\boldsymbol{y} = \{\boldsymbol{y}_i : i = 1, \ldots, n\}$

$\boldsymbol{y}(\boldsymbol{x}) \sim \mathrm{MVN}(\boldsymbol{\mu}, \Sigma)$

μ = Mean vector

Σ = Covariance

Sampling



GP: Surrogate Modeling

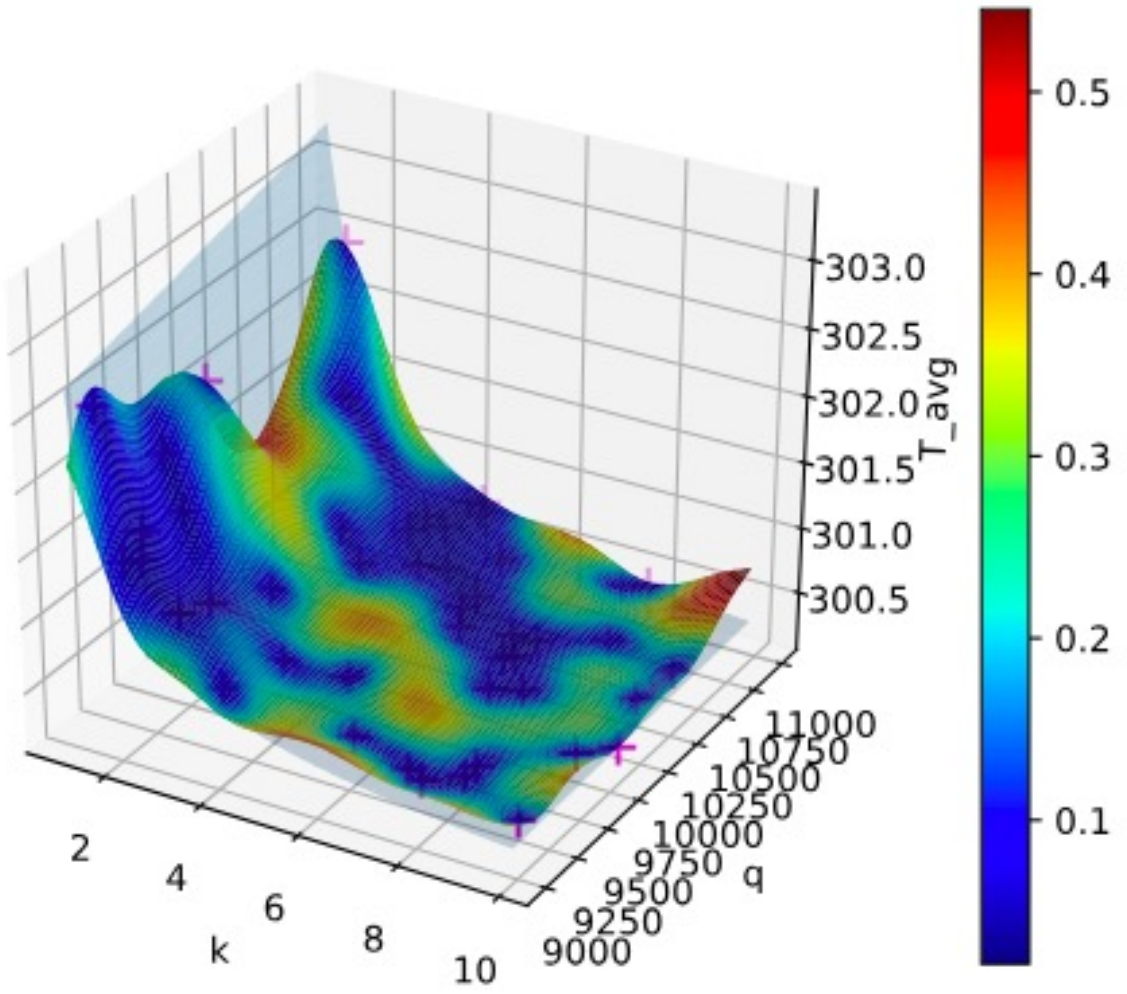
Interpolation in a parameter space

$(x_i, y_i) : i = 1, \ldots . , n$

x_i : parameters (multi-dim. vector)

y_i: simulated target variables

Exa) Temperature ~ heat flux, conductivity



https://mooseframework.inl.gov/

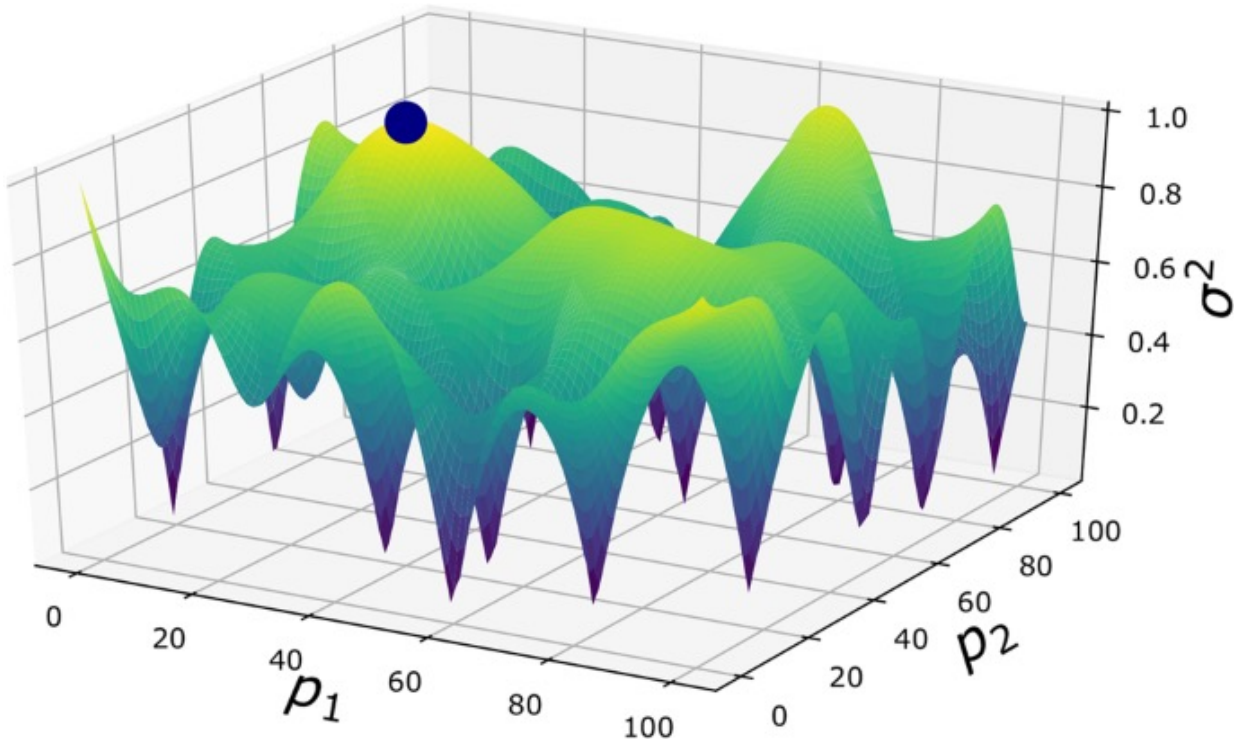
GP: Autonomous Experimentation

- **Interpolation in an experimental parameter space**

$(x_i, y_i) : i = 1, \ldots, n$

x_i : experimental parameters (multi-dim. vector)

y_i: responses



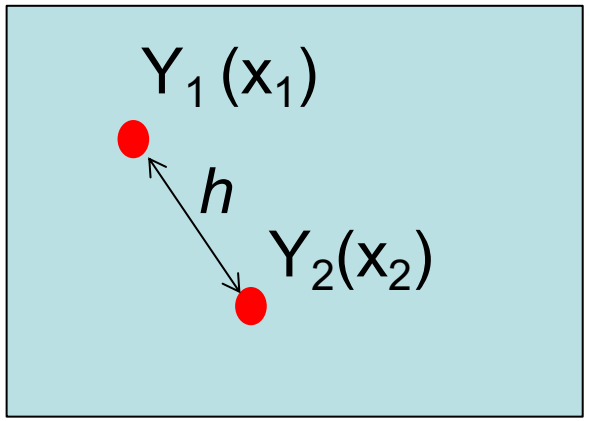
Noack, M. M., Yager, K. G., Fukuto, M., Doerk, G. S., Li, R., & Sethian, J. A. (2019). A kriging-based approach to autonomous experimentation with applications to x-ray scattering. *Scientific reports*, *9*(1), 11809.

Gaussian Process: Covariance

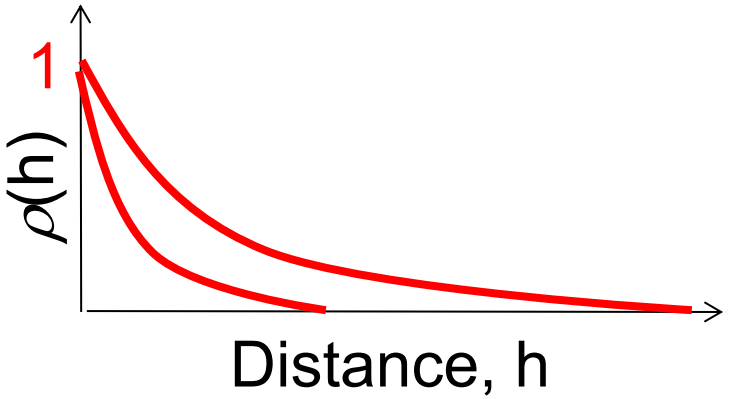
The closer two points are located, the closer values they have.

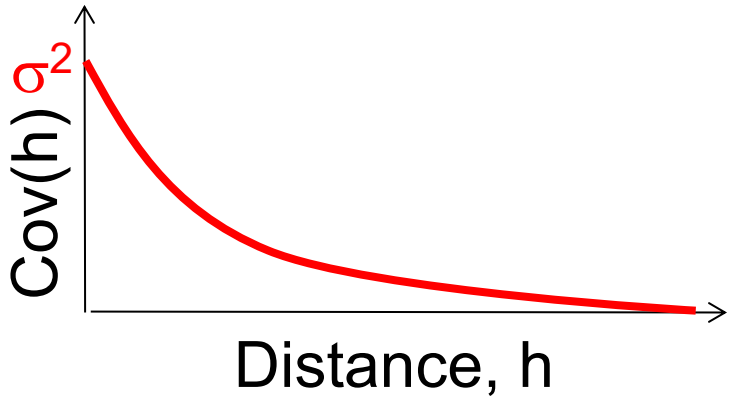
→ Higher correlation as the distance decreases

$$\text{Corr}\,[Y_1, Y_2] = \rho(h)$$
$$\text{Cov}\,[Y_1, Y_2] = \sigma^2\rho(h)$$



Distance: $h = ||x_1 - x_2||$



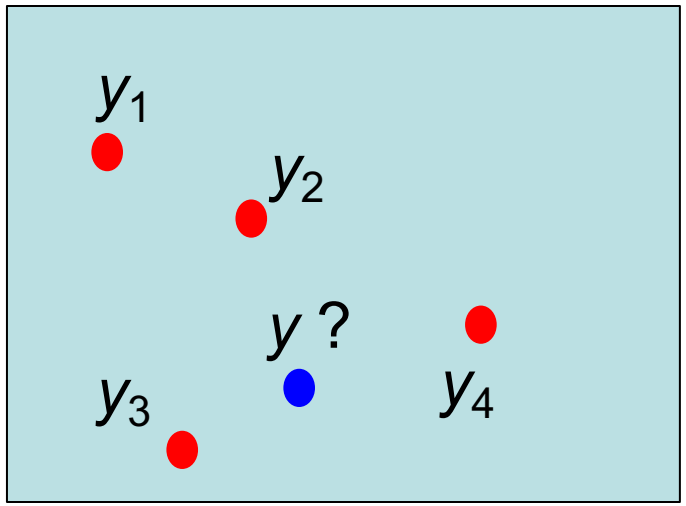


Gaussian Process: Kriging Estimation

Danie G. Krige (1951)

Master thesis on distance-weighted average gold grades

Want to estimate Y at unsampled locations based on y_1,…. y_4



Assumption…

- Stationary field
- Known covariance function

Linear interpolation: Additive Model

$$Y^* = \lambda_0 + \sum_{i=1}^{n} \lambda_i Y_i$$

Best unbiased estimator

Error $Y - Y^*$

(1) Unbiased error

$$E\left[Y - Y^*\right] = 0$$

(2) Minimum variance of the error

$$\min Var\left[Y - Y^*\right]$$

Gaussian Process: Kriging

Kriging estimate

$$\hat{y} = \mu + \Sigma_Y \Sigma^{-1} \left(\boldsymbol{y} - \mu \right)$$

$\boldsymbol{y}$: measurements $y_1, \ldots y_n$

Σ: Auto-covariance matrix

$$\Sigma_{ij} = \mathrm{Cov}(Y_i, Y_j)$$

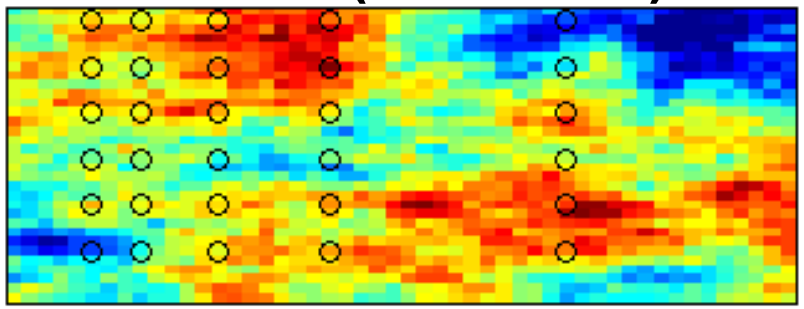
Σ_Y: Cross-covariance matrix

$$\Sigma_j = \mathrm{Cov}(Y_Y, Y_j)$$

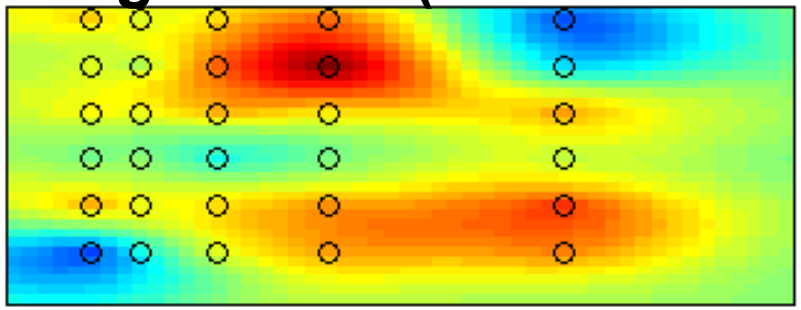
Kriging variance

$$\hat{\sigma}^2 = \sigma^2 - \Sigma_Y \Sigma^{-1} \Sigma_Y^T$$

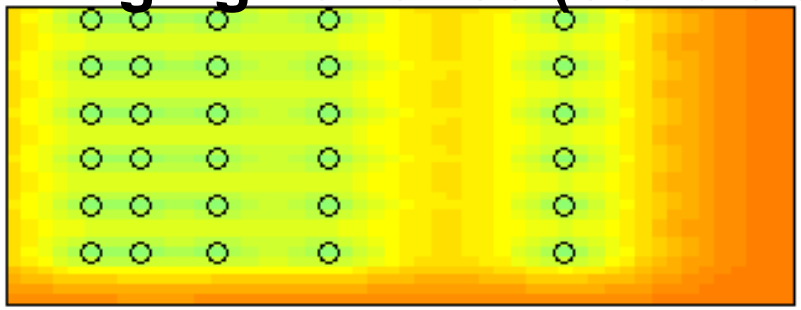
Real Field (unknown)



Kriged Field (conditional mean)



Kriging Variance (conditional var)



Kernels: Covariance

- **Exponential**

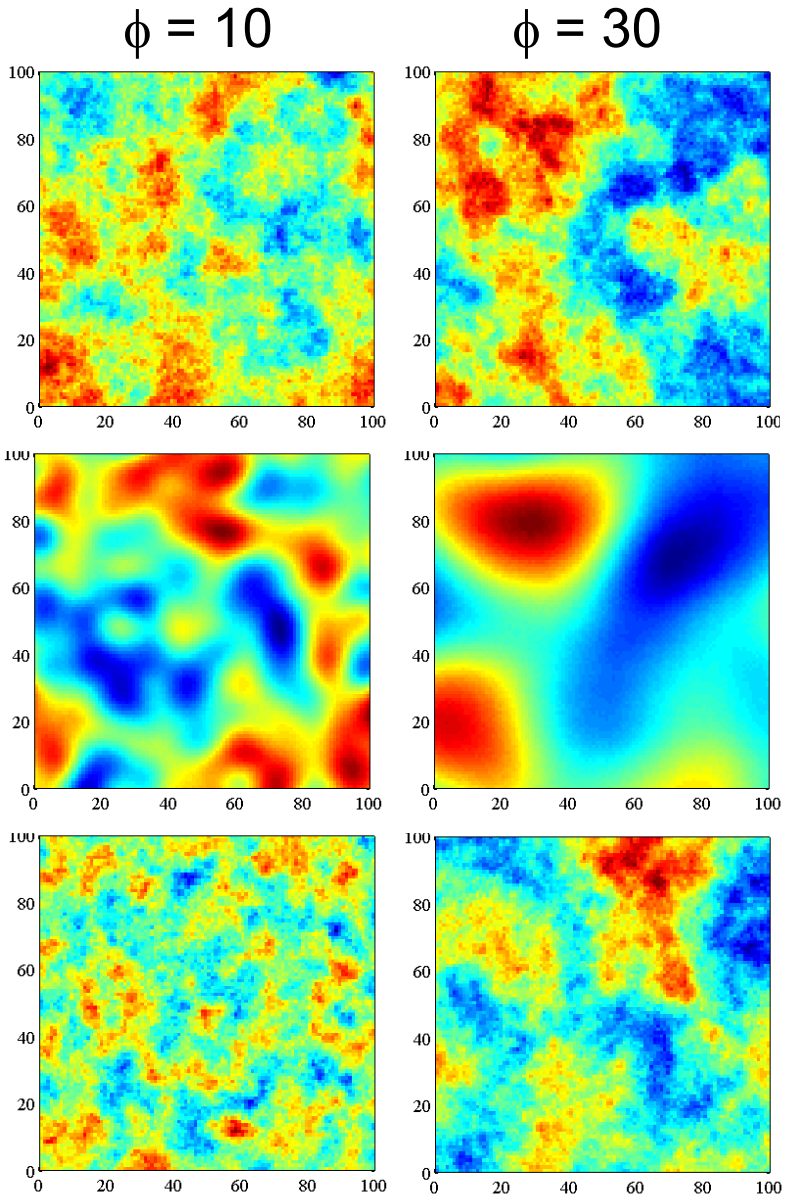
$$\mathrm{Cov}(h) = \sigma^2 \exp\left(-\frac{h}{\phi}\right)$$

- **Gaussian (radial basis functions)**

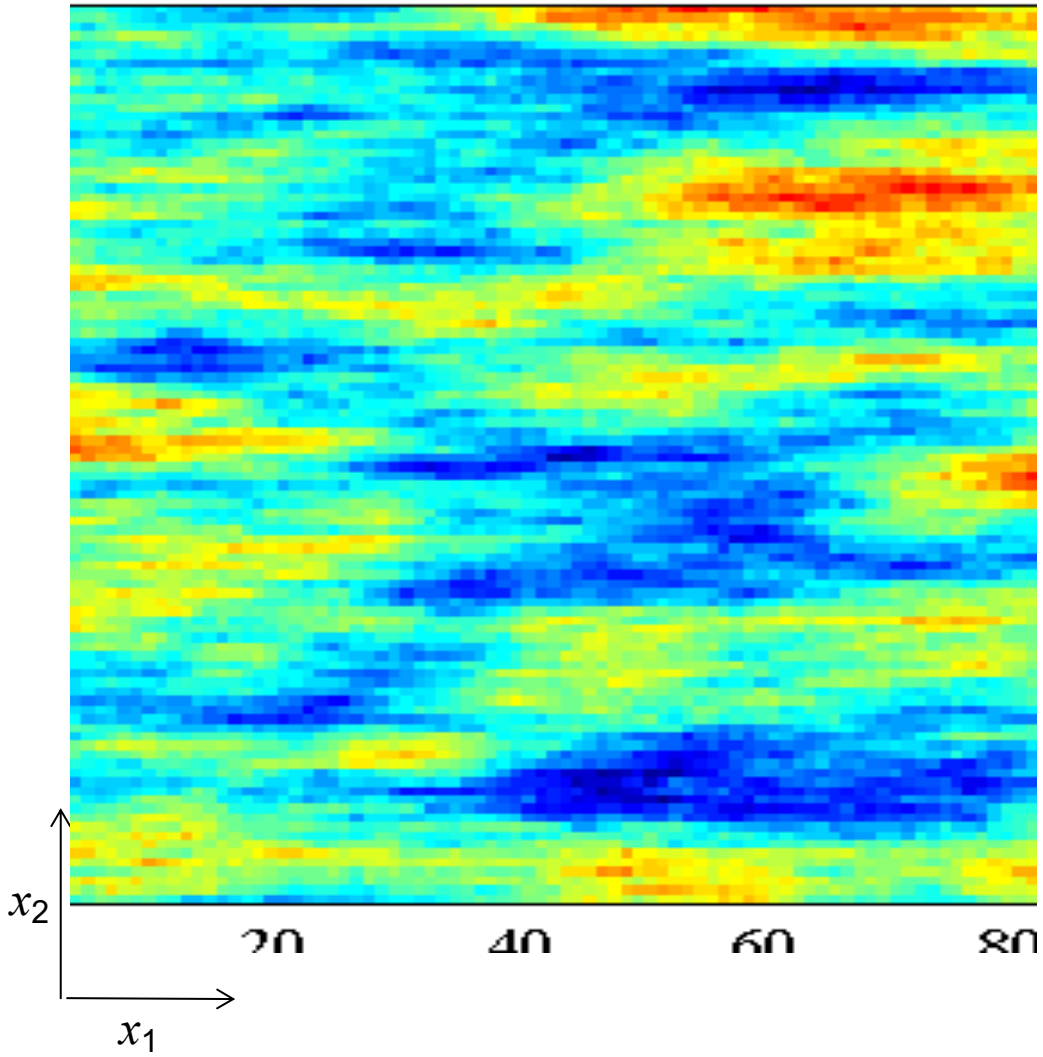
$$\mathrm{Cov}(h) = \sigma^2 \exp\left(-\frac{h^2}{\phi^2}\right)$$

- **Spherical**

$$\mathrm{Cov}(h) = \begin{cases} \sigma^2 \exp\left(-\frac{h^2}{\phi^2}\right), 0 < h < \phi \\ \sigma^2, \phi \leq h \end{cases}$$



Kernels: Anisotropy



$$\mathrm{Cov}(h_1, h_2) = \sigma^2 \exp\left(-\sqrt{\frac{h_1^2}{\phi_1^2} + \frac{h_2^2}{\phi_2^2}}\right)$$

Kernels: Nugget

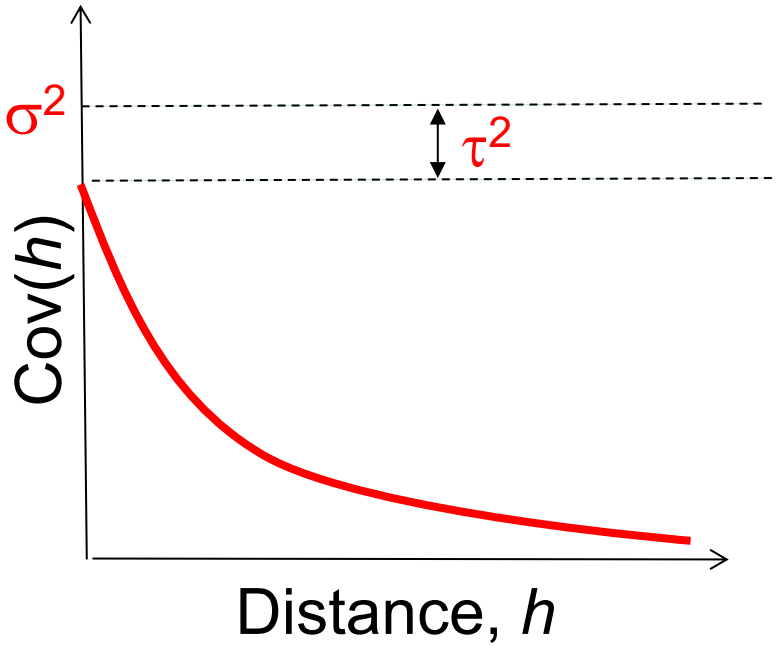
Origin: gold nugget

→Spatial discontinuity

→Uncorrelated variability, noise

- Spatial variation on a scale smaller than the closest distance
- Variability within a grid cell
- Measurement errors

* They are inseparable

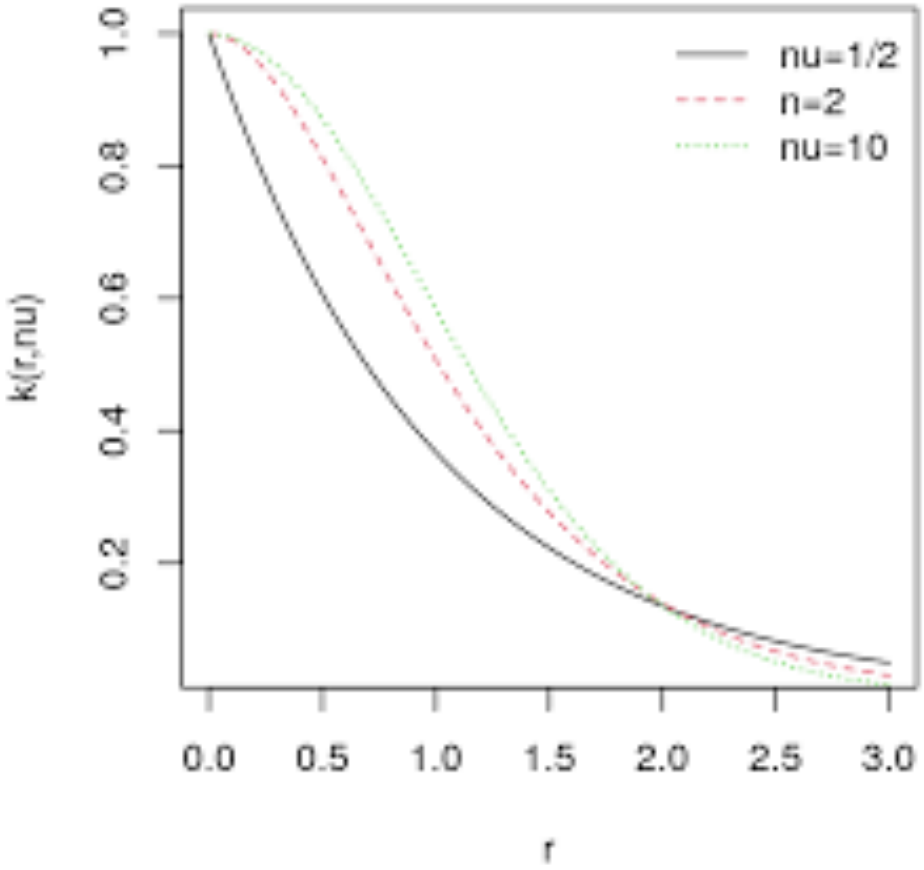


Kernels: Matern

Flexible covariance (model uncertainty)

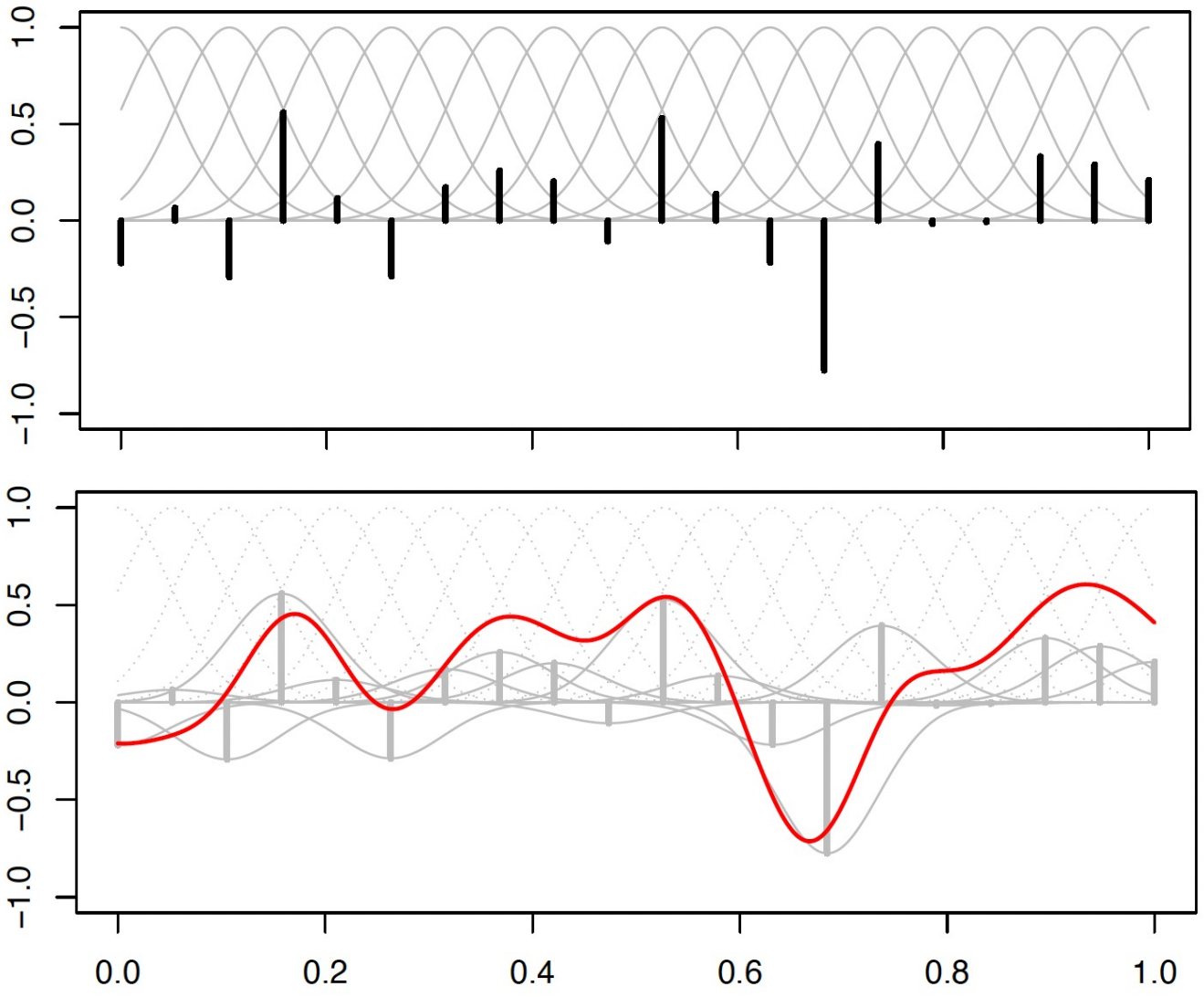
$$\mathrm{Cov}(h) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\frac{h}{\phi}\right)^{\nu} K_\nu \left(\sqrt{2\nu}\frac{h}{\phi}\right)$$

- ν: smoothness parameter
- $\nu = 1/2$: exponential
- $\nu = \infty$: gaussian



https://bookdown.org/rbg/surrogates/chap5.html

Non-stationary GP



Courtesy to Mark D. Risser

$$f(\mathbf{x}) = \int_G K(\mathbf{x} - \mathbf{u}) dW(\mathbf{u})$$

$$f(\mathbf{x}) \approx \sum_{m=1}^{M} K(\mathbf{x} - \mathbf{u}_m) W(\mathbf{u}_m)$$

Risser, Mark D., and Daniel Turek. "Bayesian inference for high-dimensional nonstationary Gaussian processes." *Journal of Statistical Computation and Simulation* 90.16 (2020): 2902-2928.

Non-stationary Covariance

Changing variance over space

$$\mathrm{Cov}(\boldsymbol{x}, \boldsymbol{x}') = \sigma(\boldsymbol{x})\sigma(\boldsymbol{x}') \frac{|\Sigma(\boldsymbol{x})|^{1/4}|\Sigma(\boldsymbol{x})|^{1/4}}{\left|\frac{\Sigma(\boldsymbol{x}) + \Sigma(\boldsymbol{x}')}{2}\right|^{1/2}} M_\nu\big(Q(\boldsymbol{x}, \boldsymbol{x}')\big)$$

where

$$Q(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x} - \boldsymbol{x}')^T \left\{\frac{\Sigma(\boldsymbol{x}) + \Sigma(\boldsymbol{x}')}{2}\right\}^{-1} (\boldsymbol{x} - \boldsymbol{x}')$$

BayesNSGP Package for R

Risser, Mark D., and Daniel Turek. "Bayesian inference for high-dimensional nonstationary Gaussian processes." *Journal of Statistical Computation and Simulation* 90.16 (2020): 2902-2928.

Non-stationary GP for Large Datasets

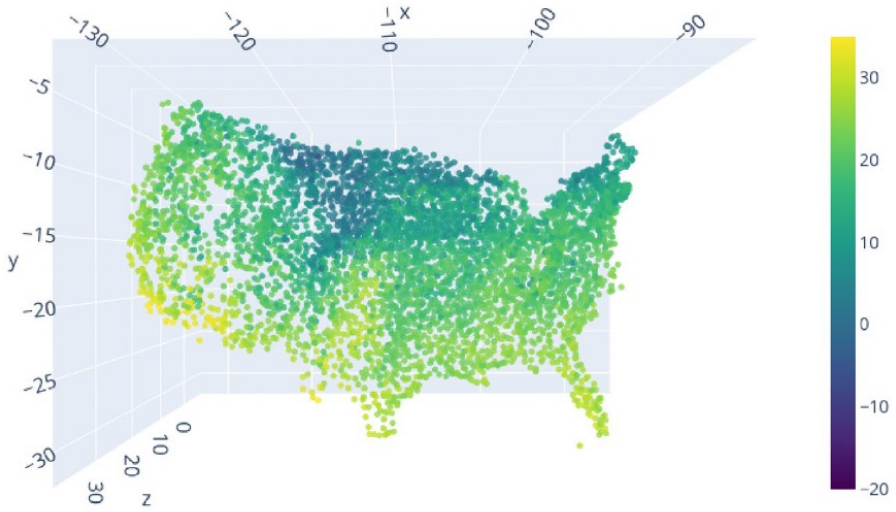
scientific reports

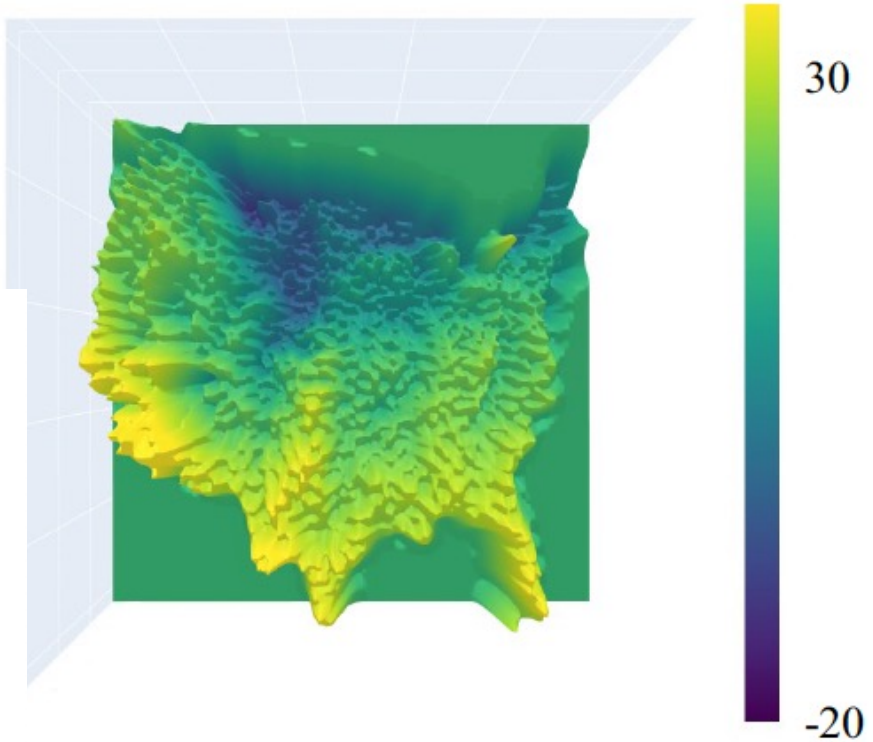
Check for updates

OPEN

Exact Gaussian processes for massive datasets via non-stationary sparsity-discovering kernels

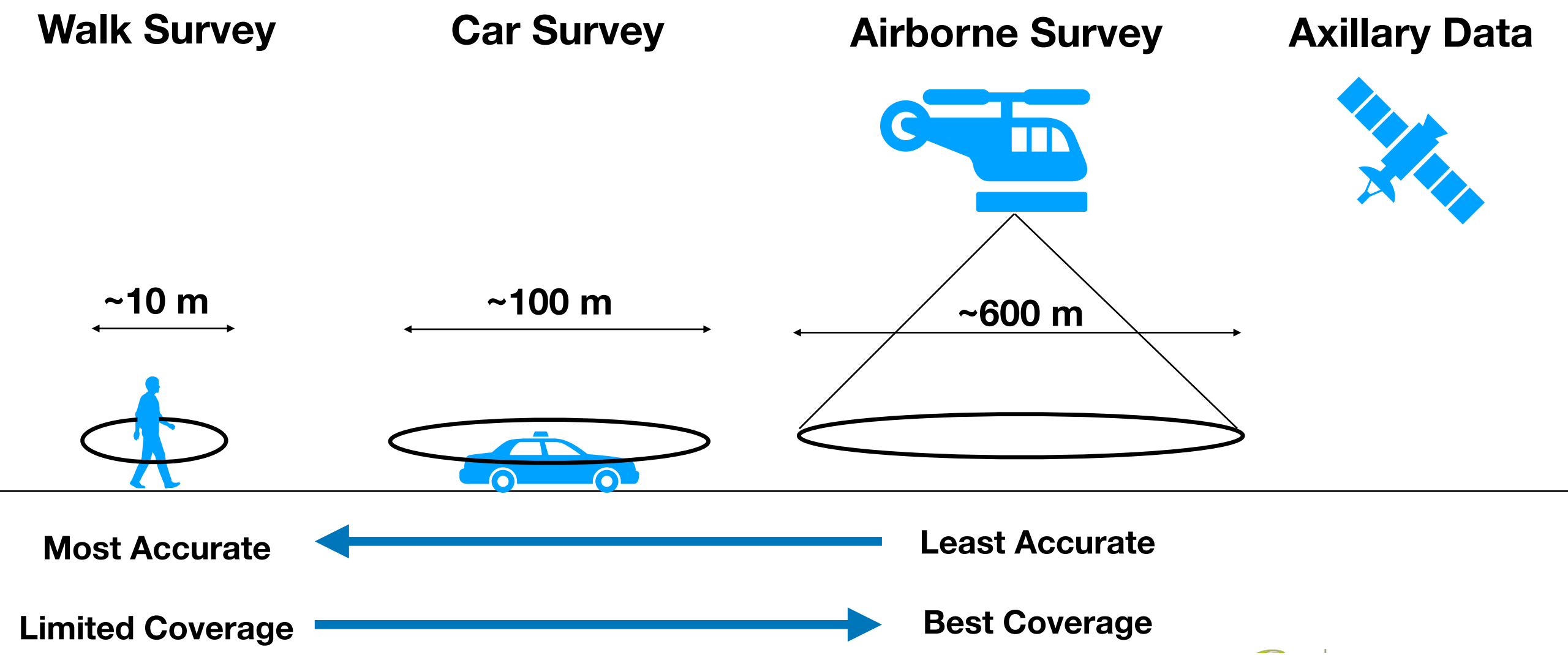
Marcus M. Noack[1]✉, Harinarayan Krishnan[1], Mark D. Risser[2] & Kristofer G. Reyes[3]





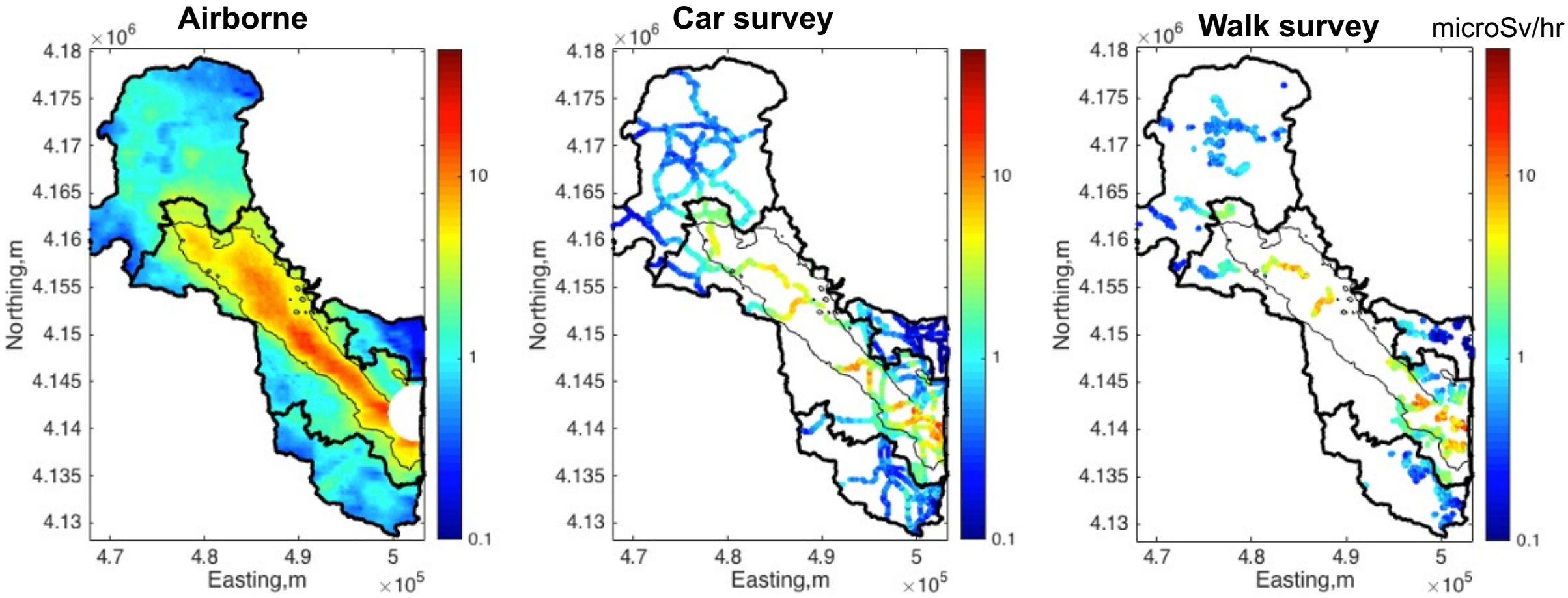
(c)

Data Integration: Fukushima Radiation Air Dose Rates



Wainwright, H. M., Seki, A., Chen, J., & Saito, K. (2017). A multiscale Bayesian data integration approach for mapping air dose rates around the Fukushima Daiichi Nuclear Power Plant. *Journal of environmental radioactivity*, *167*, 62-69.

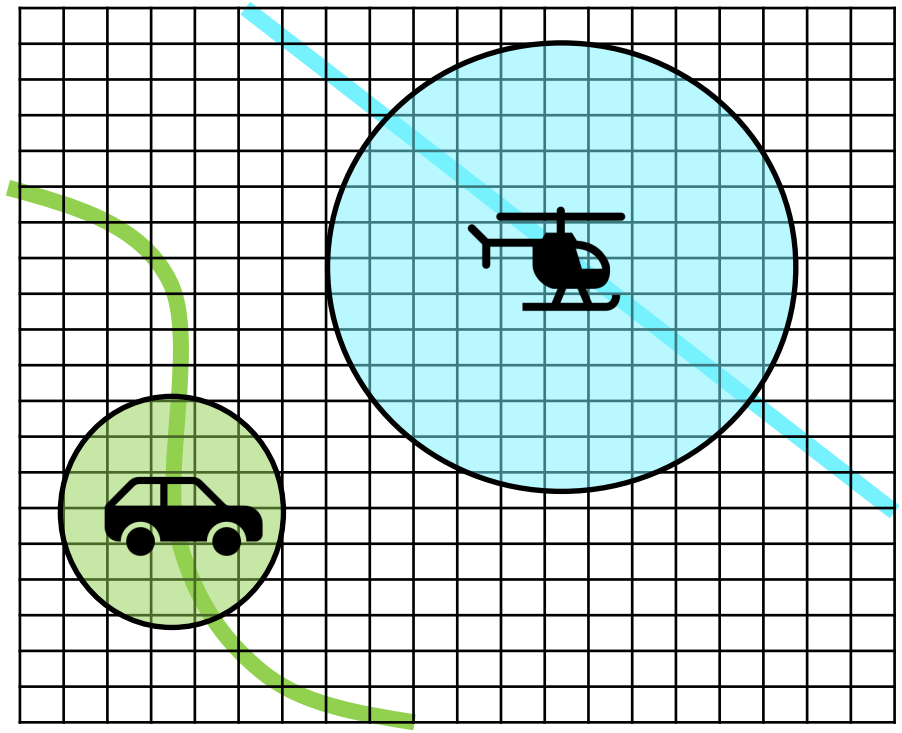
2016 Data within the Evacuation Zone (2017)



Airborne Survey: Over-estimation
Area＞20mSv/y ＝ 264 km²

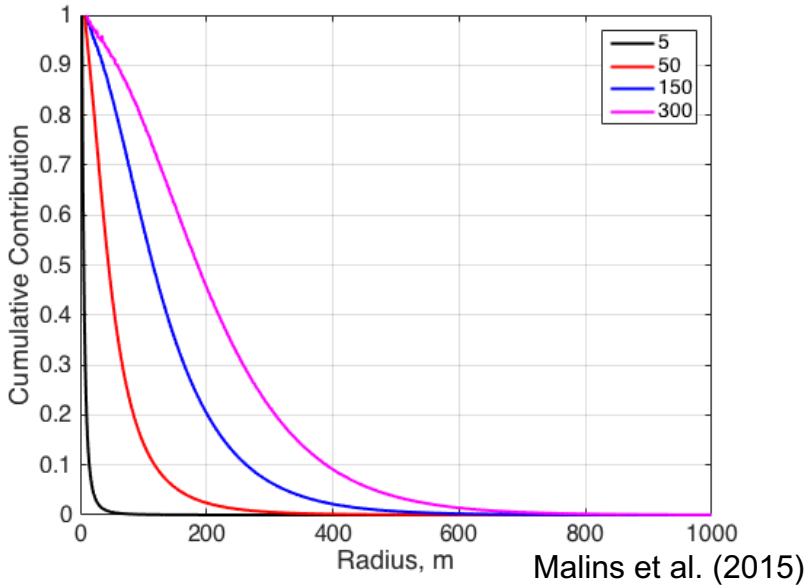
Multiscale Representation

Map of air dose rates at 1m above the ground surface $\boldsymbol{y} = \{y_1, y_2, \ldots . y_N\}$



Grid: 50 m x 50 m

Radiation Transport Simulations



Malins et al. (2015)

Airborne survey data

$$z_{A,j} = f_A\left(\sum w_{A,i,j} y_i\right) + \varepsilon_{A,j},$$

$$\boldsymbol{z}_A = A\boldsymbol{y} + \boldsymbol{\varepsilon}_A$$

$$\boldsymbol{\varepsilon}_A = N(\boldsymbol{\mu}, \mathrm{D_A})$$

Car survey data

$$z_{C,j} = f_C\left(\sum w_{C,i,j} y_i\right) + \varepsilon_{C,j}.$$

$$\boldsymbol{z}_C = C\boldsymbol{y} + \boldsymbol{\varepsilon}_C$$

$$\boldsymbol{\varepsilon}_C = N(\boldsymbol{\mu}, \mathrm{D_C})$$

Bayesian Data Integration

- Estimate the spatial distribution of radiation dose rate y (Sv/hr) conditioned on walk (z_W), car (z_C), and airborne survey (z_A)

$$p(\boldsymbol{y}|\boldsymbol{z}_W, \boldsymbol{z}_C, \boldsymbol{z}_A)$$

- Target dose rate is the walk survey data: health risk of a person walking on streets

- Posterior distribution

$$p(\boldsymbol{y}|\boldsymbol{z}_W, \boldsymbol{z}_A, \boldsymbol{z}_C) \propto \int p(\boldsymbol{z}_A|\boldsymbol{y}) p(\boldsymbol{z}_C|\boldsymbol{y}) p(\boldsymbol{y}|\boldsymbol{z}_W, \theta) p(\theta) d\theta$$

$$= MVN(Q^{-1}\boldsymbol{g}, Q^{-1})$$

$$\mathrm{Q} = \Sigma^{-1} + \mathrm{A}^{\mathrm{T}}\mathrm{D}_{\mathrm{A}}^{-1}\mathrm{A} + \mathrm{C}^{\mathrm{T}}\mathrm{D}_{\mathrm{C}}^{-1}\mathrm{C}$$

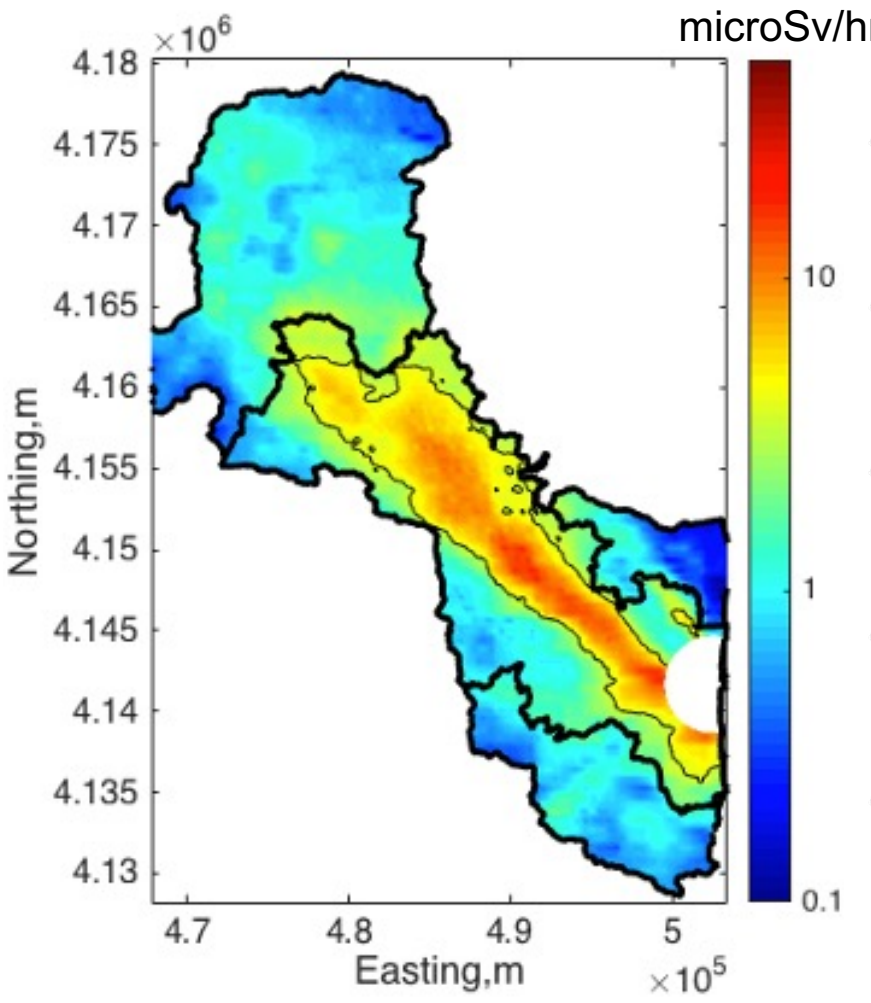
$$\boldsymbol{g} = \boldsymbol{\mu} + A^T D_A^{-1} \boldsymbol{z}_A + C^T D_C^{-1} \boldsymbol{z_c}$$

$$p(\boldsymbol{y}|\boldsymbol{z}_W, \theta) = MVN(\boldsymbol{\mu}, \Sigma)$$

2016 Integrated Map: Evacuation zone (2017)

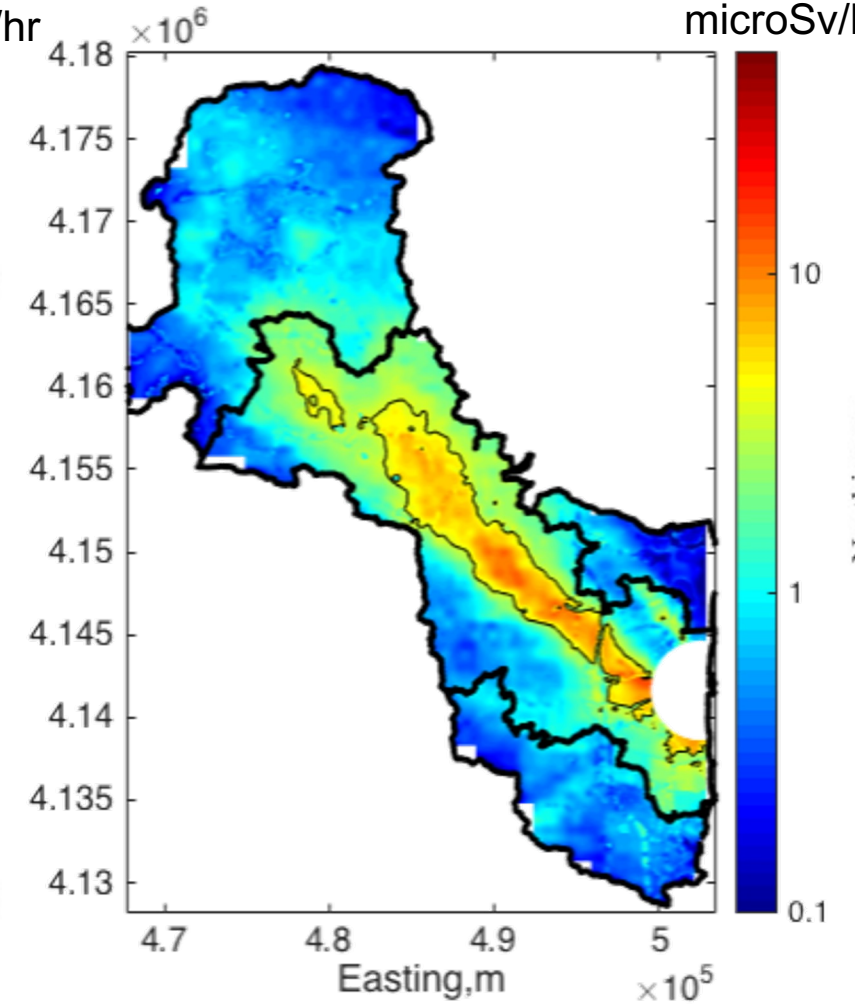
Airborne Data

Area＞20mSv/y ＝ 264 km²



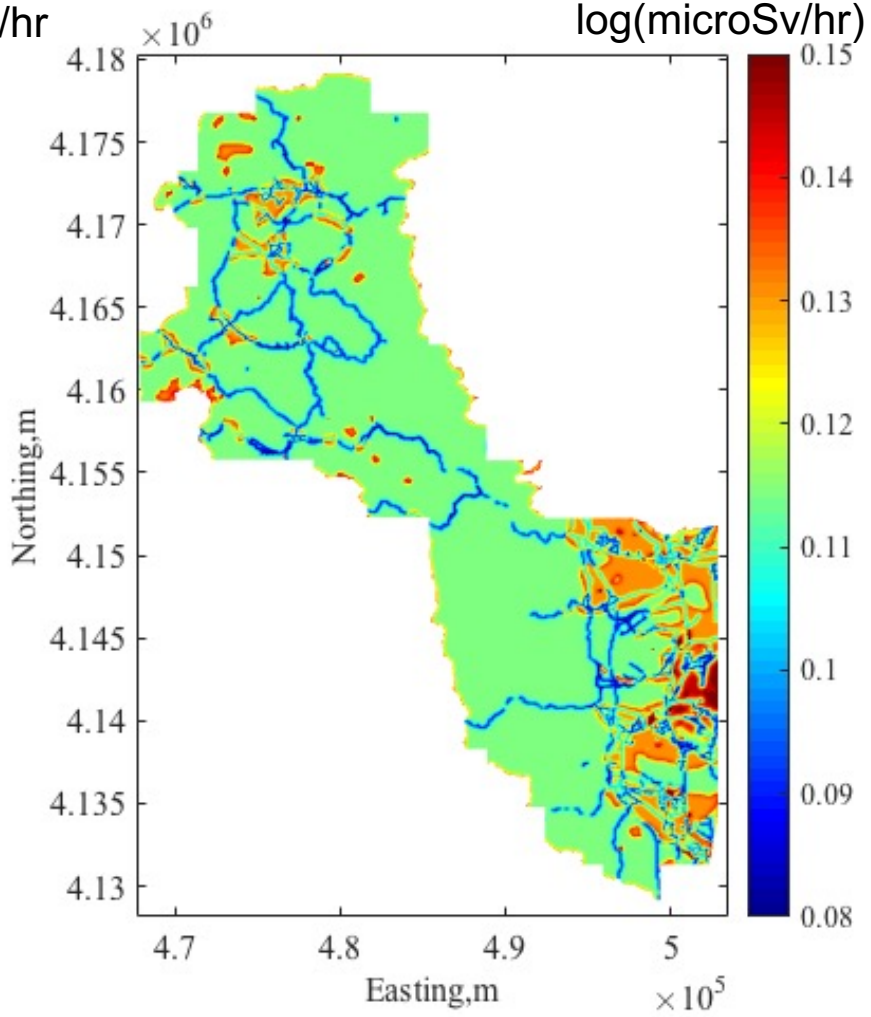
Integrated Map

Area＞20mSv/y ＝ 218km²

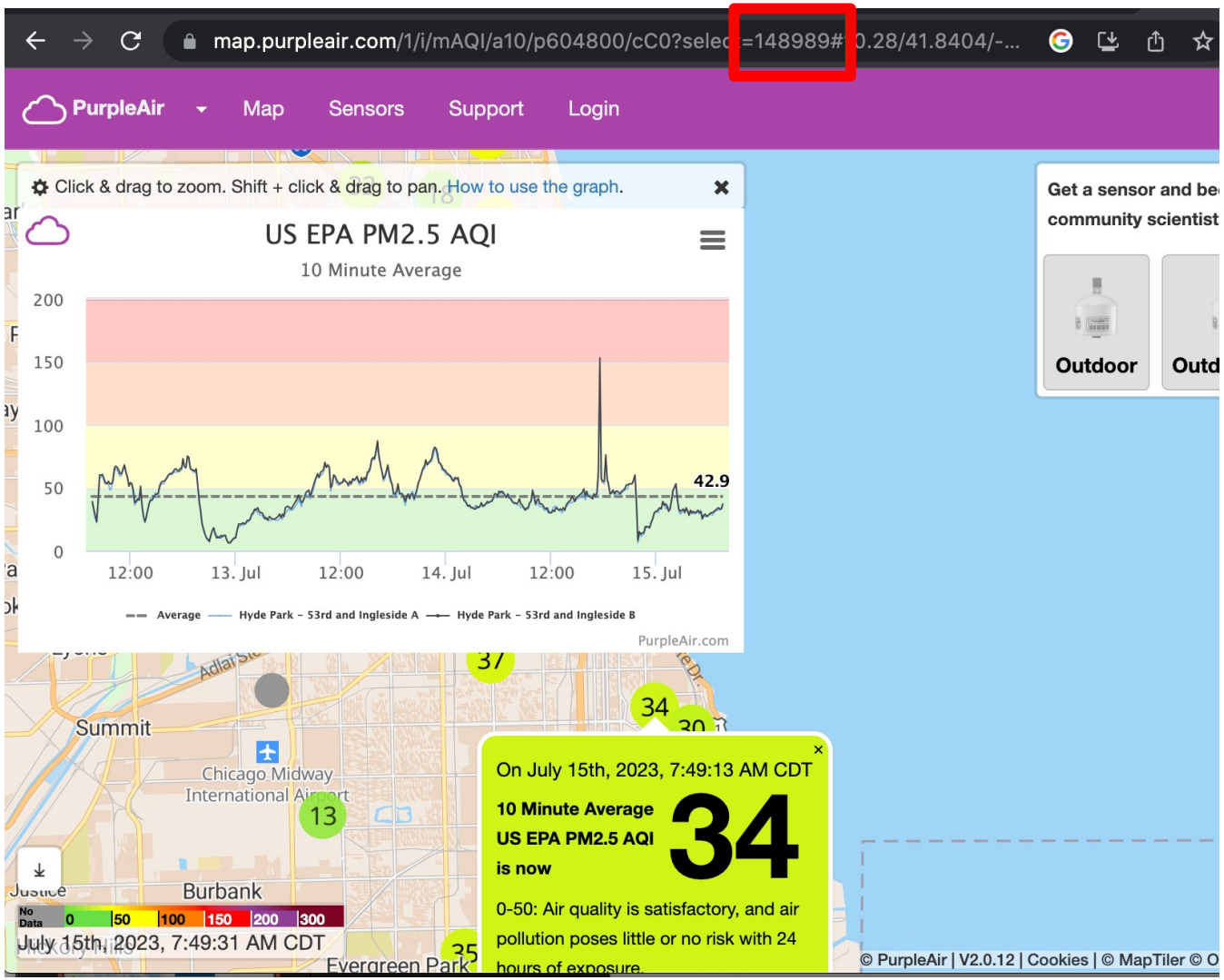


Uncertainty Estimate

Standard deviation



Environmental Data API: Purple Air



purpleair 0.0.4

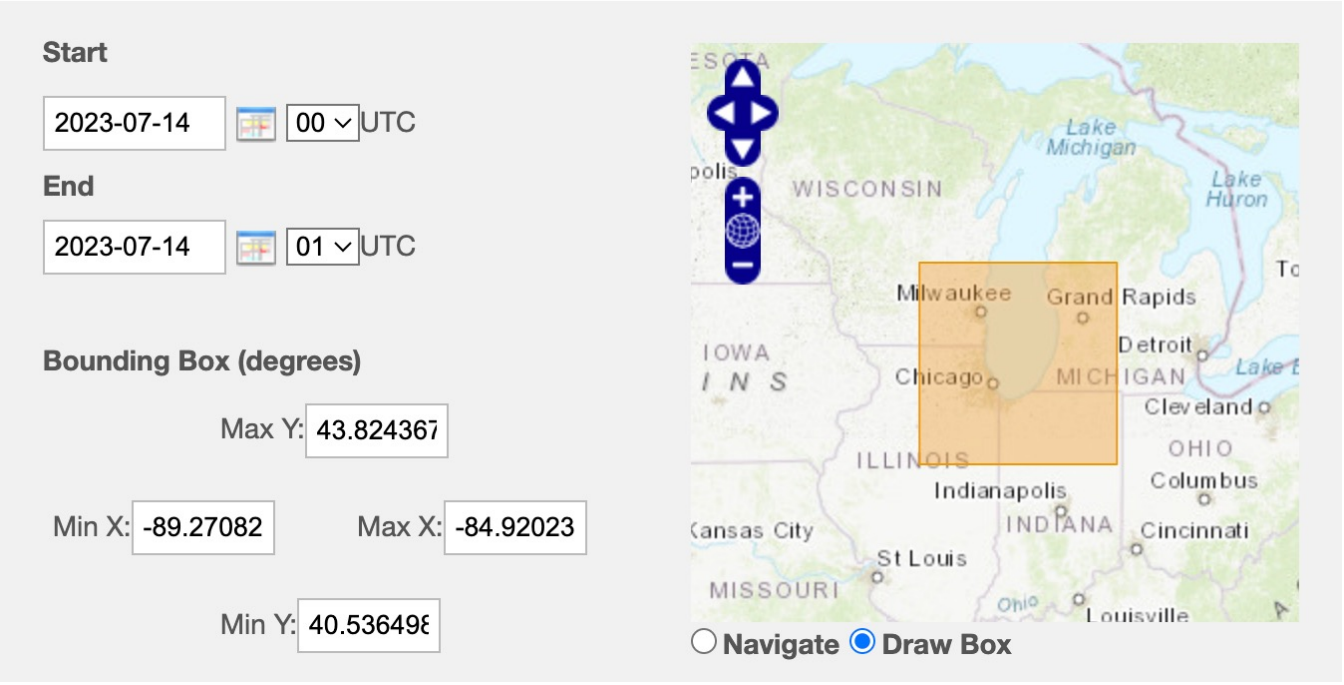
```
pip install purpleair
```

```
In [1]: from purpleair import PurpleAir

In [2]: p = PurpleAir('READ_KEY_HERE')

# See help(p.get_sensor_data) for param names, and see https://api.purpleair.com f

In [3]: p.get_sensor_data('99999')
Out[3]:
{'api_version': 'V1.0.10-0.0.17',
 'time_stamp': 1657577251,
 'data_time_stamp': 1657577238,
 'sensor': {'sensor_index': 99999,
  'last_modified': 1628736055,
  'date_created': 1624389476,
  'last_seen': 1657577216,
  'private': 0,
  'is_owner': 0,
  'name': '99999',
  'icon': 0,
  'location_type': 0,
  'model': 'PA-II',
  'hardware': '2.0+BME280+PMSX003-B+PMSX003-A',
  'led_brightness': 35,
  'firmware_version': '7.00',
```

API key is not free

Environmental Data API: EPA Air Quality

Air Now API: docs.airnowapi.org/ (free)



Generate URL

```
url =
'https://www.airnowapi.org/aq/data/?startDat
e=2023-06-26T14&endDate=2023-06-
26T15&parameters=PM25&BBOX=-
89.615860,41.33,-
84.606094,44.3&dataType=A&format=text/csv&ve
rbose=0&monitorType=0&includerawconcentratio
ns=0&API_KEY=3EDB1ADE-7637-4F22-A5B3-
05218A41D98D'
```

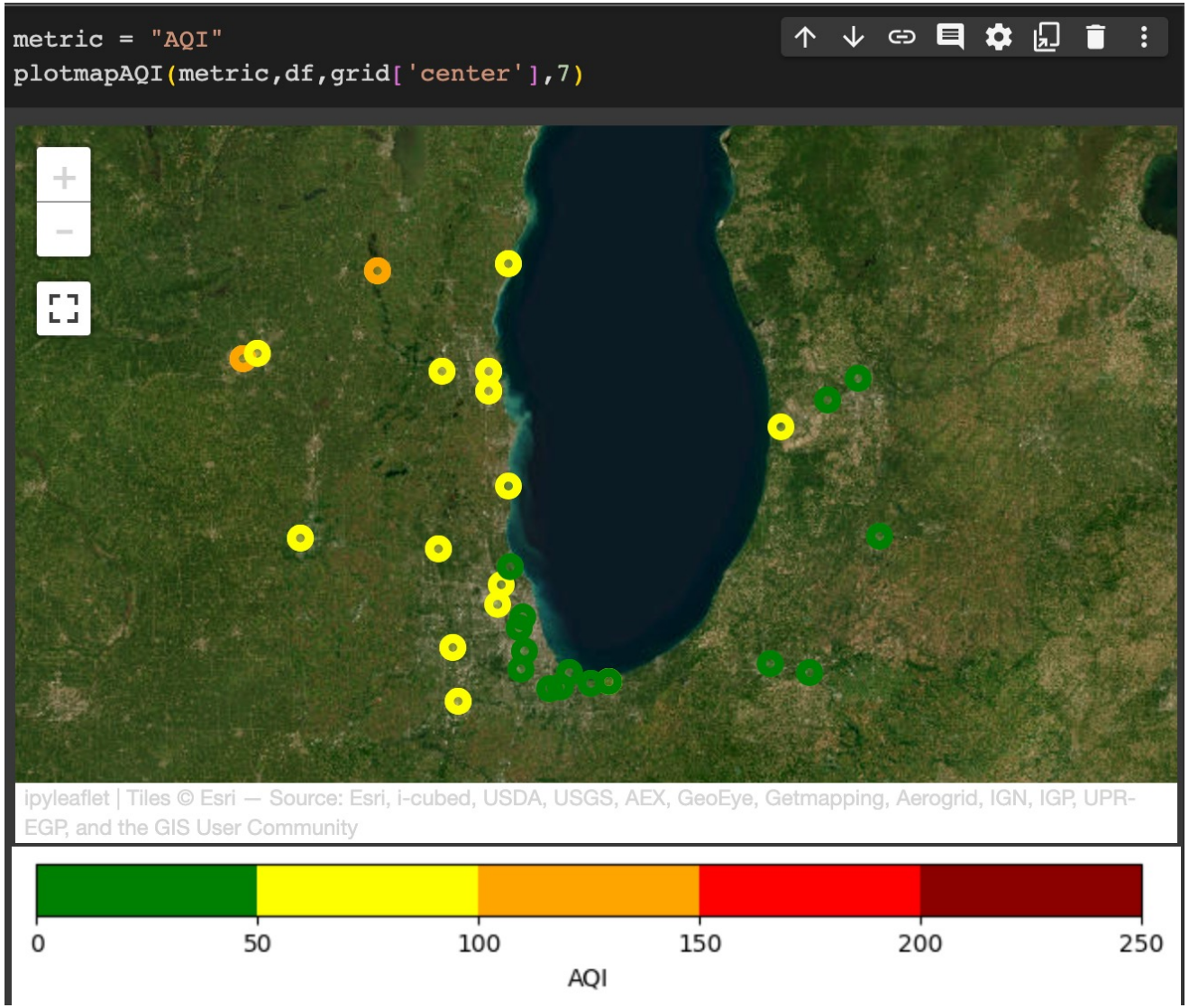
```
import requests

r = requests.get(url, allow_redirects=True)
open('epa_aqi_temp.csv', 'wb').write(r.content)

3531
```

Map Visualization: ipyleaflet/Folium

Example

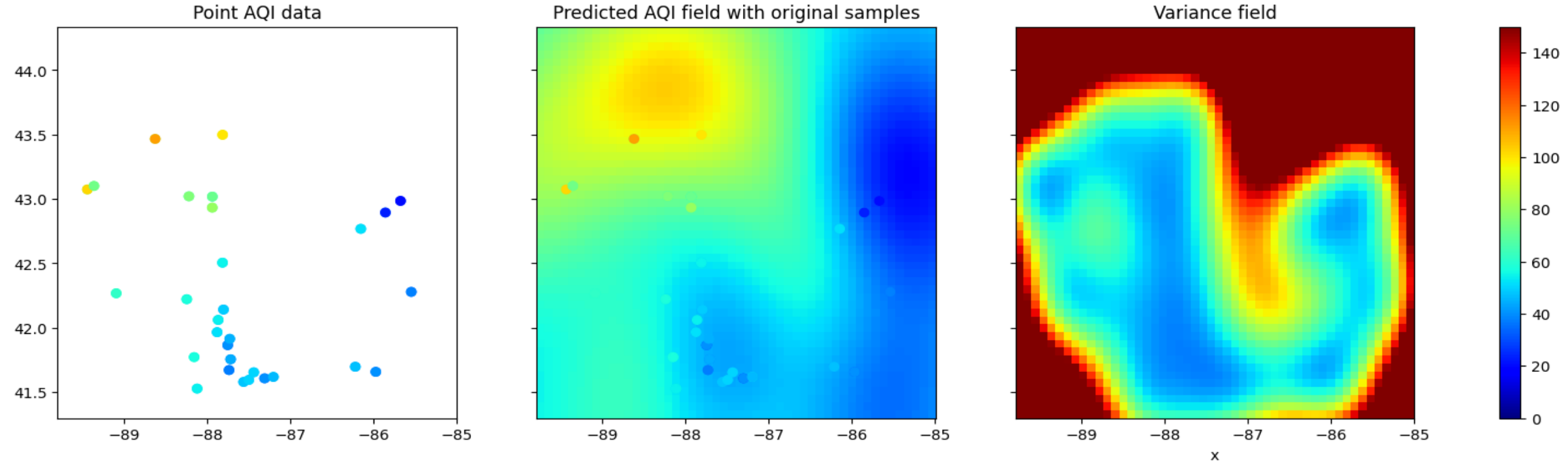
```
from ipyleaflet import Map, basemaps, basemap_to_tiles

m = Map(
    basemap=basemap_to_tiles(basemaps.NASAGIBS.ModisTerraTrueColorCR
    center=(52.204793, 360.121558),
    zoom=4
)

m
```


```
metric = "AQI"
plotmapAQI(metric,df,grid['center'],7)
```

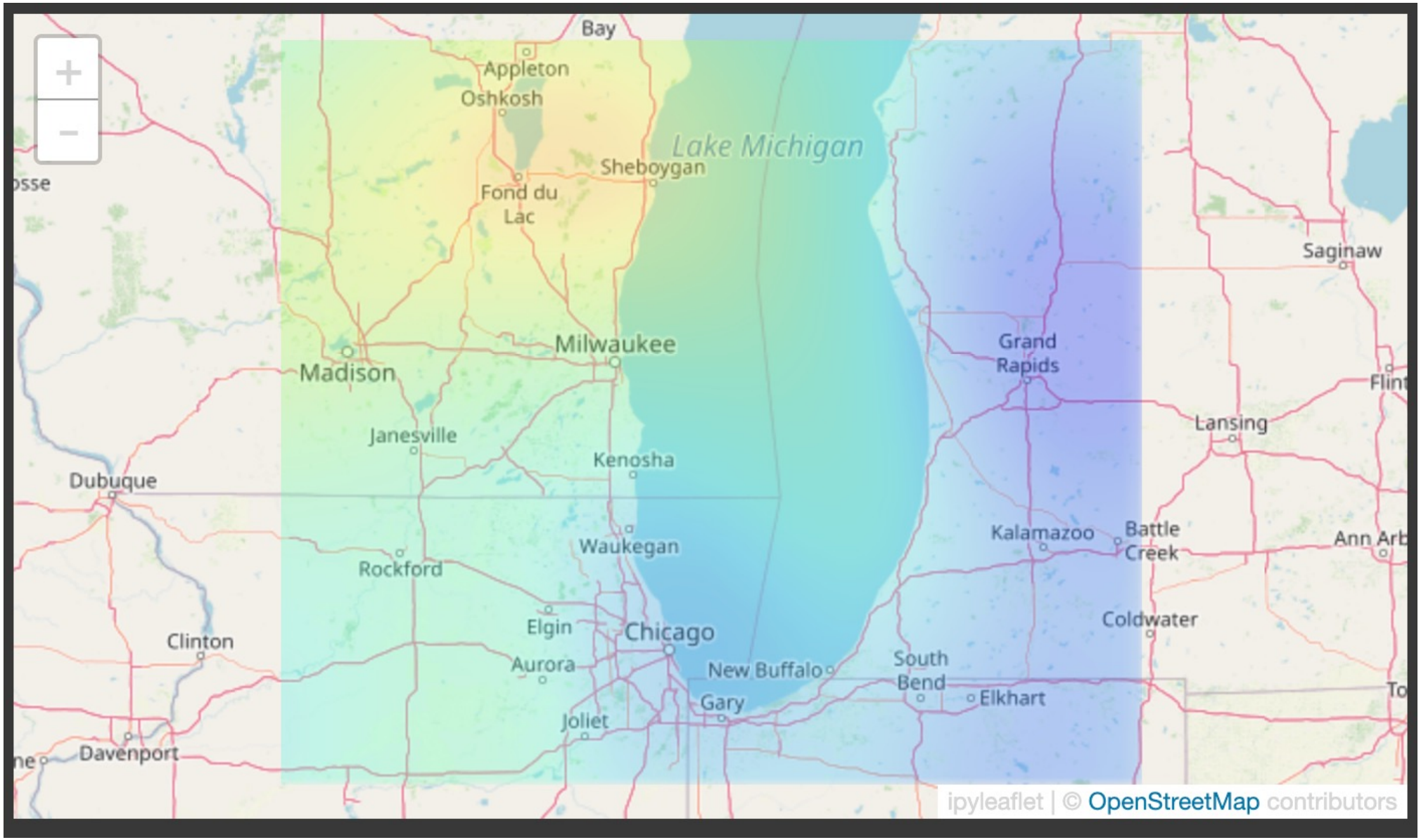


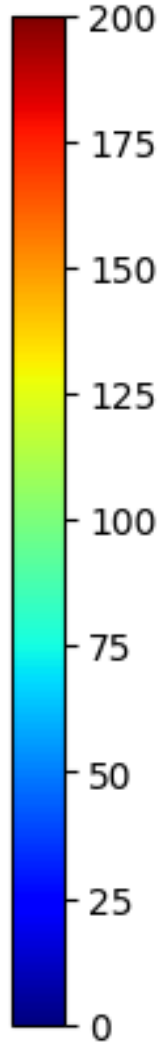
Interpolate EPA data: GP

```
my_ae = gpcam.autonomous_experimenter.AutonomousExperimenterGP(param_bounds,
                                    init_hp,
                                    hp_bounds,
                                    init_dataset_size= 100,
                                    x_data=x_train,
                                    y_data=y_train,
                                    kernel_func = kernel_RBF_noise,
                                    use_inv = True,
                                    communicate_full_dataset = False,
                                    ram_economy = True)#, info = False, prior_mean_func = optional_mean_func)
my_ae.train(max_iter=10000)
```



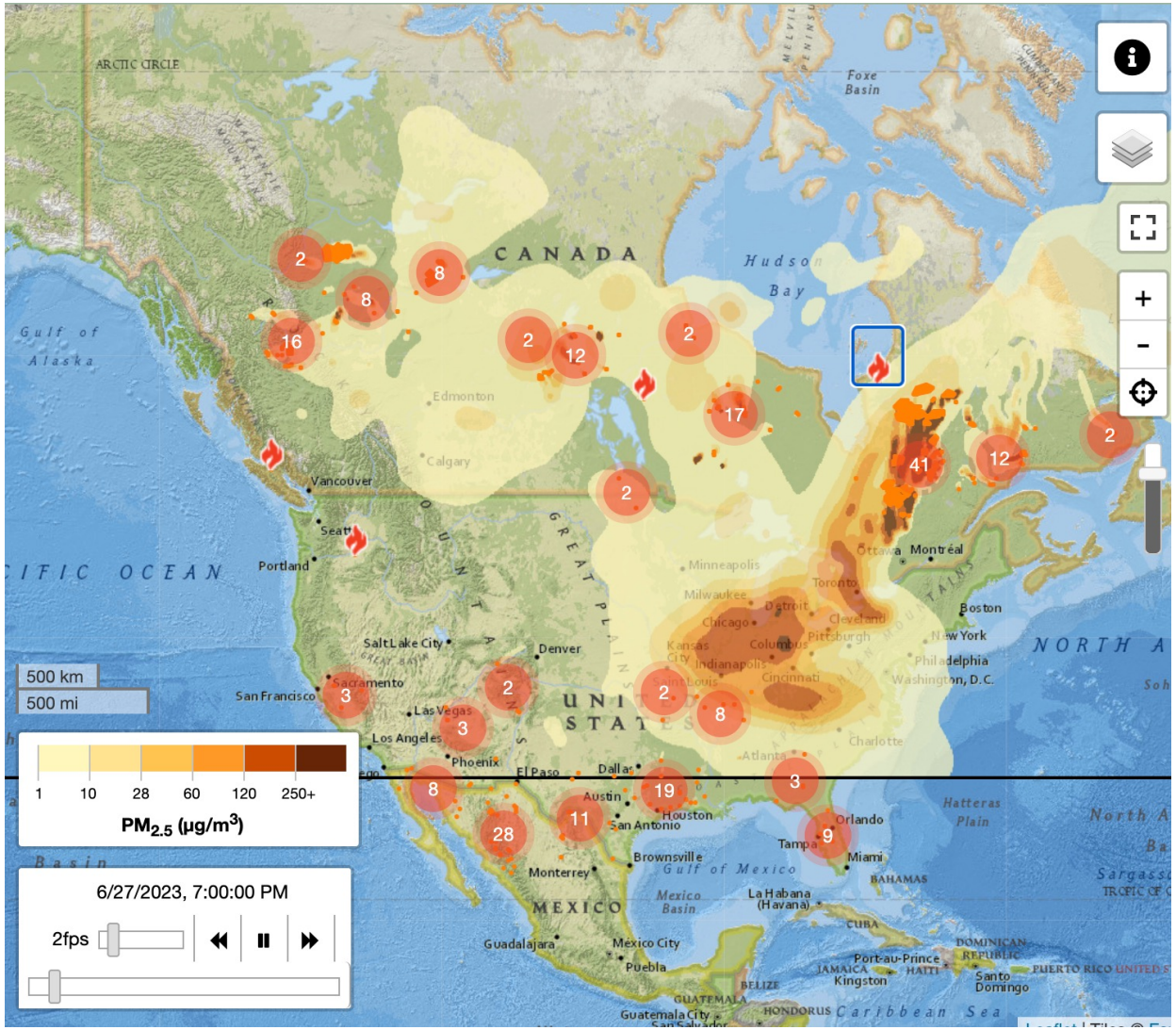
Interpolate EPA data: Visualization





Plume Simulation

firesmoke.ca/



Weather Forecast Research Team at the University of British Columbia

```
import requests
import pickle

url = 'https://firesmoke.ca/forecasts/BSC06CA12-01/2023062614/dispersion.nc'
r = requests.get(url, allow_redirects=True)
open('dispersion.nc', 'wb').write(r.content)
```

NetCDF format:

```
XCENT:  -106.0
YCENT:  51.0
XORIG:  -160.0
YORIG:  32.0
XCELL:  0.10000000149011612
YCELL:  0.10000000149011612
```

→ Re-interpolate on the grid

EPA vs Purple Air: US-wide Study

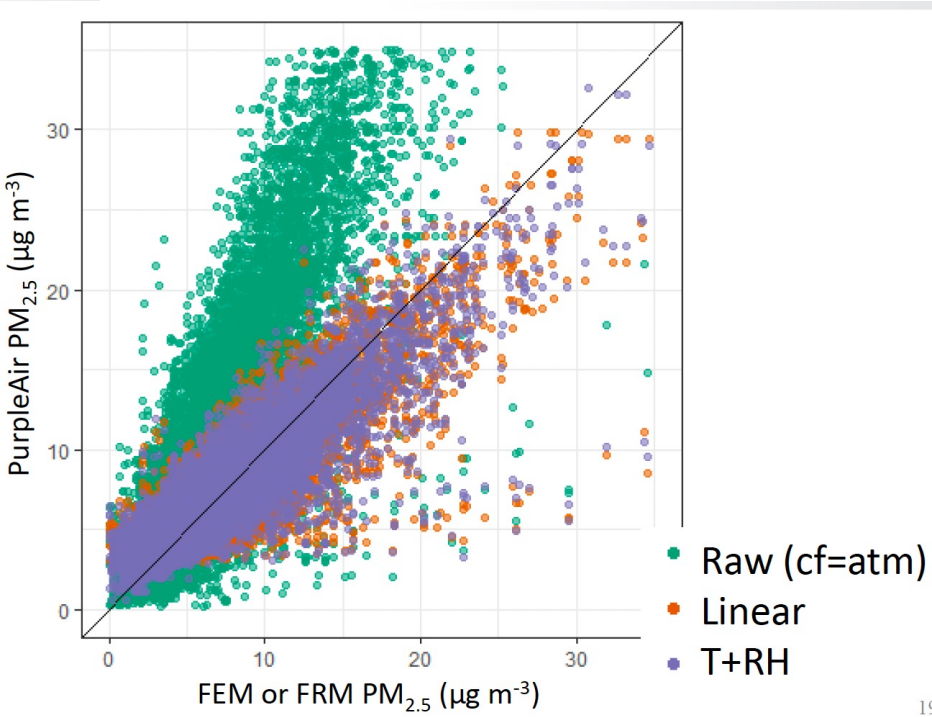
Equations considered

1. **Raw:** $PM_{2.5}$ = PA (raw PurpleAir $PM_{2.5}$ cf=atm)

2. **Linear:** $PM_{2.5} = 0.38*PA + 2.94$, $R^2=0.69$

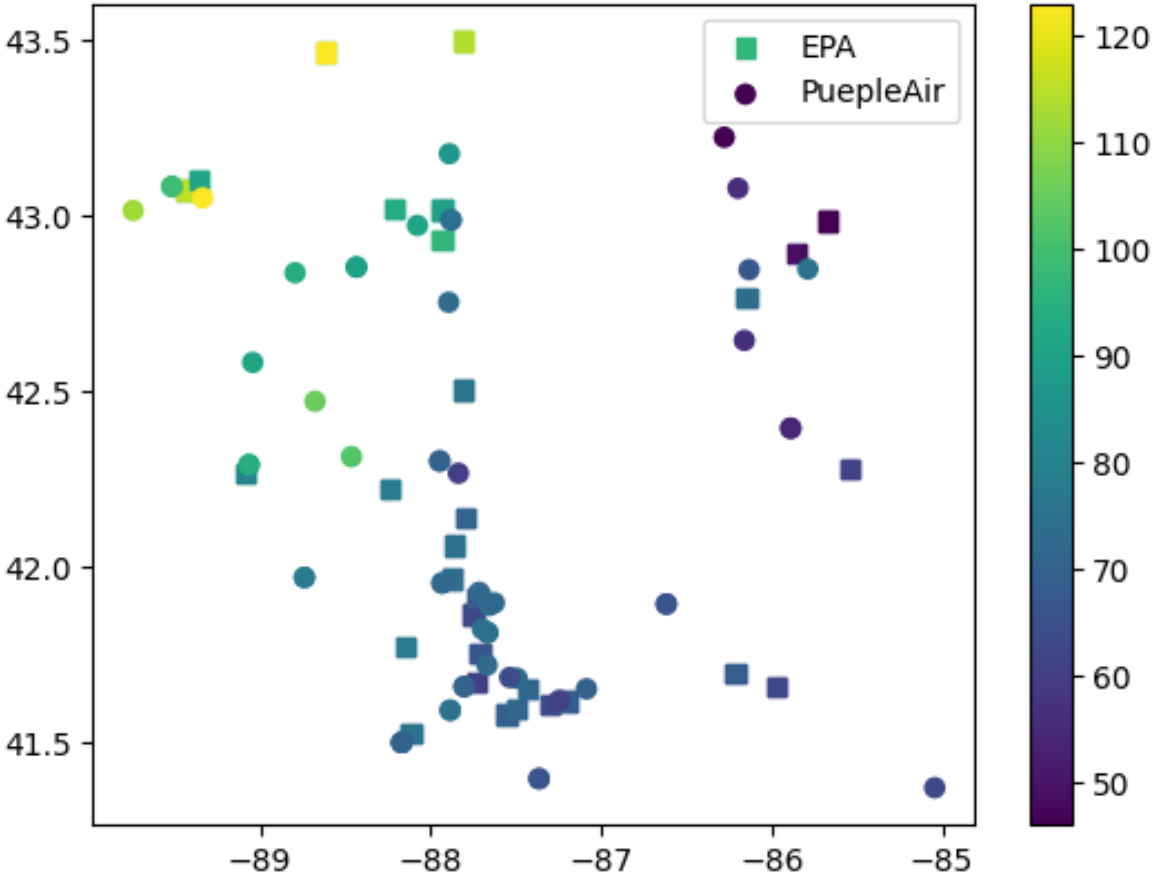
3. **T & RH:** $PM_{2.5} = 0.39*PA + 0.0024*T - 0.050*RH + 5.19$, $R^2=0.72$

Units:
$PM_{2.5}$= µg m^{-3}
T=°F
RH=%

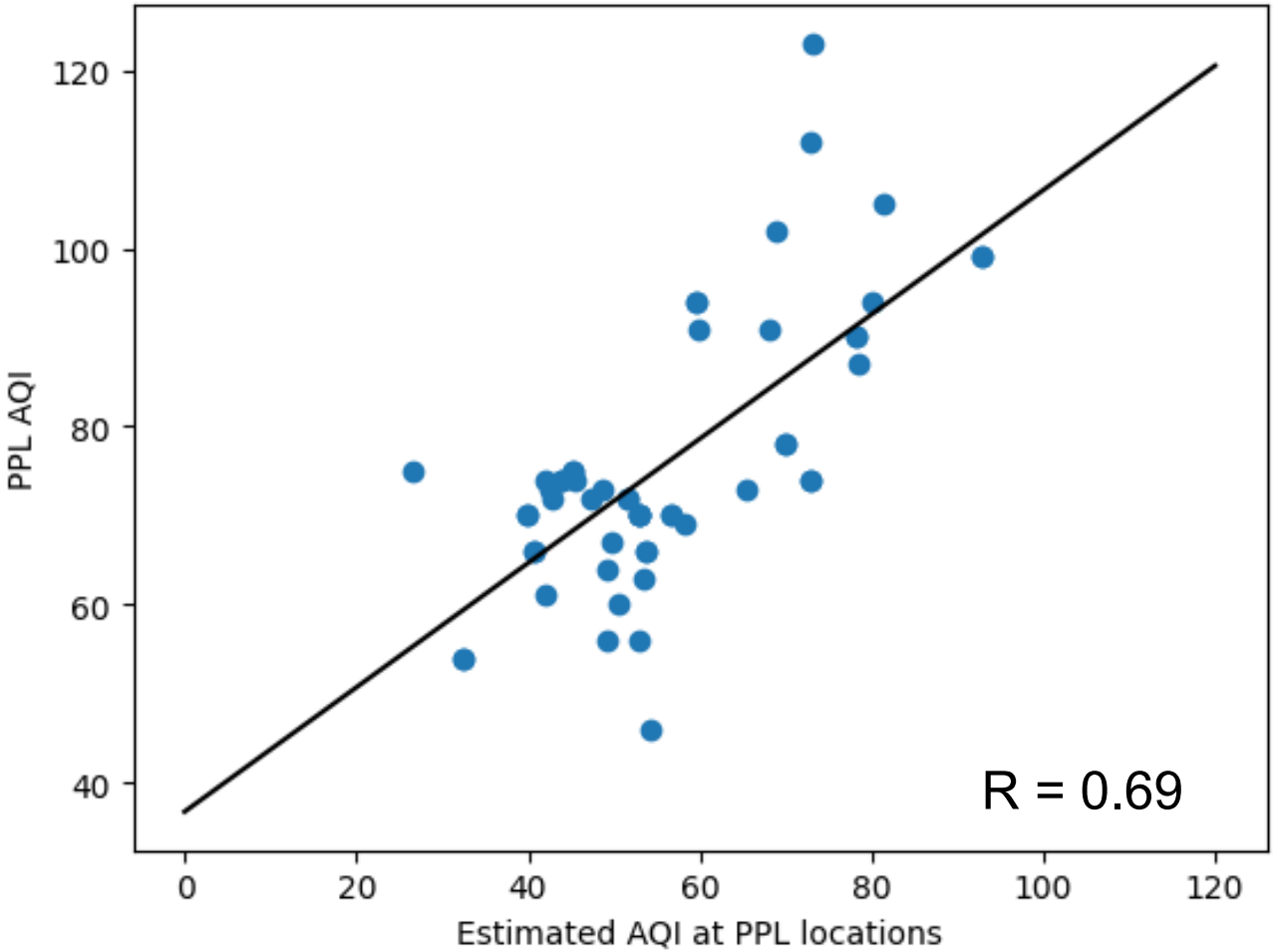


Johnson, K., B. Gantt, I. VonWald, AND A. Clements. PurpleAir PM2.5 Performance Across the U.S. Webinar Presentation with State/Local/Tribal Project Partners, RTP, NC, December 09, 2019.

EPA vs Purple Air: Local relationship



$$z_{\mathrm{PPL}} = a\tilde{y}(x_{\mathrm{PPL}}) + b + \varepsilon$$



based on EPA data

AQI Data Integration: EPA + PPL + Simulation

$$p(\boldsymbol{y}|\boldsymbol{z}_{\mathrm{PPL}}, \boldsymbol{z}_{\mathbf{EPA}}, \boldsymbol{z}_{\mathrm{sim}}) \propto \int p(\boldsymbol{z}_{\mathrm{PPL}}|\boldsymbol{y}) p(\boldsymbol{y}|\boldsymbol{z}_{\mathrm{sim}}, \boldsymbol{\theta}, \boldsymbol{z}_{\mathrm{EPA}}) d\boldsymbol{\theta}$$

$$\boldsymbol{z}_{\mathrm{PPL}} = \mathrm{A}\boldsymbol{y} + \boldsymbol{\varepsilon}, \boldsymbol{\varepsilon} \sim \boldsymbol{MVN}(D_A)$$

- Jeffrey's prior for $\boldsymbol{\theta}$
- Remove the trend $f(\boldsymbol{z}_{\mathrm{sim}})$ at data locations, using a regression
- Estimate the residual field $\boldsymbol{r} = \boldsymbol{y} - f(\boldsymbol{z}_{\mathrm{sim}})$
- Max. likelihood estimation for $\boldsymbol{\theta}$ first using $\boldsymbol{z}_{\mathrm{EPA}}$

$$p(\boldsymbol{r}|\boldsymbol{z}_{\mathrm{sim}}, \boldsymbol{\theta}, \boldsymbol{z}_{\mathrm{EPA}}) = MVN(\boldsymbol{\mu}_{\mathrm{C}}, \Sigma_c)$$

$$\begin{cases} \boldsymbol{\mu}_{\mathrm{C}} = \Sigma_{\mathrm{cross}} \Sigma^{-1} \boldsymbol{z}_{\mathrm{EPA}} \\ \Sigma_c = \Sigma_{\mathrm{cross}} \Sigma^{-1} \Sigma_{\mathrm{cross}} \end{cases}$$

$$p(\boldsymbol{r}|\boldsymbol{z}_{\mathrm{PPL}}, \boldsymbol{z}_{\mathbf{EPA}}, \boldsymbol{z}_{\mathrm{sim}}) = MVN(Q^{-1} g, Q)$$

$$\begin{cases} \mathrm{Q} = \Sigma_C^{-1} + \mathrm{A}^{\mathrm{T}} \mathrm{D}_{\mathrm{A}}^{-1} \mathrm{A} \\ \boldsymbol{g} = \boldsymbol{\mu}_{\boldsymbol{C}} + A^T D_A^{-1} \boldsymbol{z}_A \end{cases}$$

AQI Data Integration: EPA + PPL + Simulation

Alternatively, in this case, the PPL data are point measurements

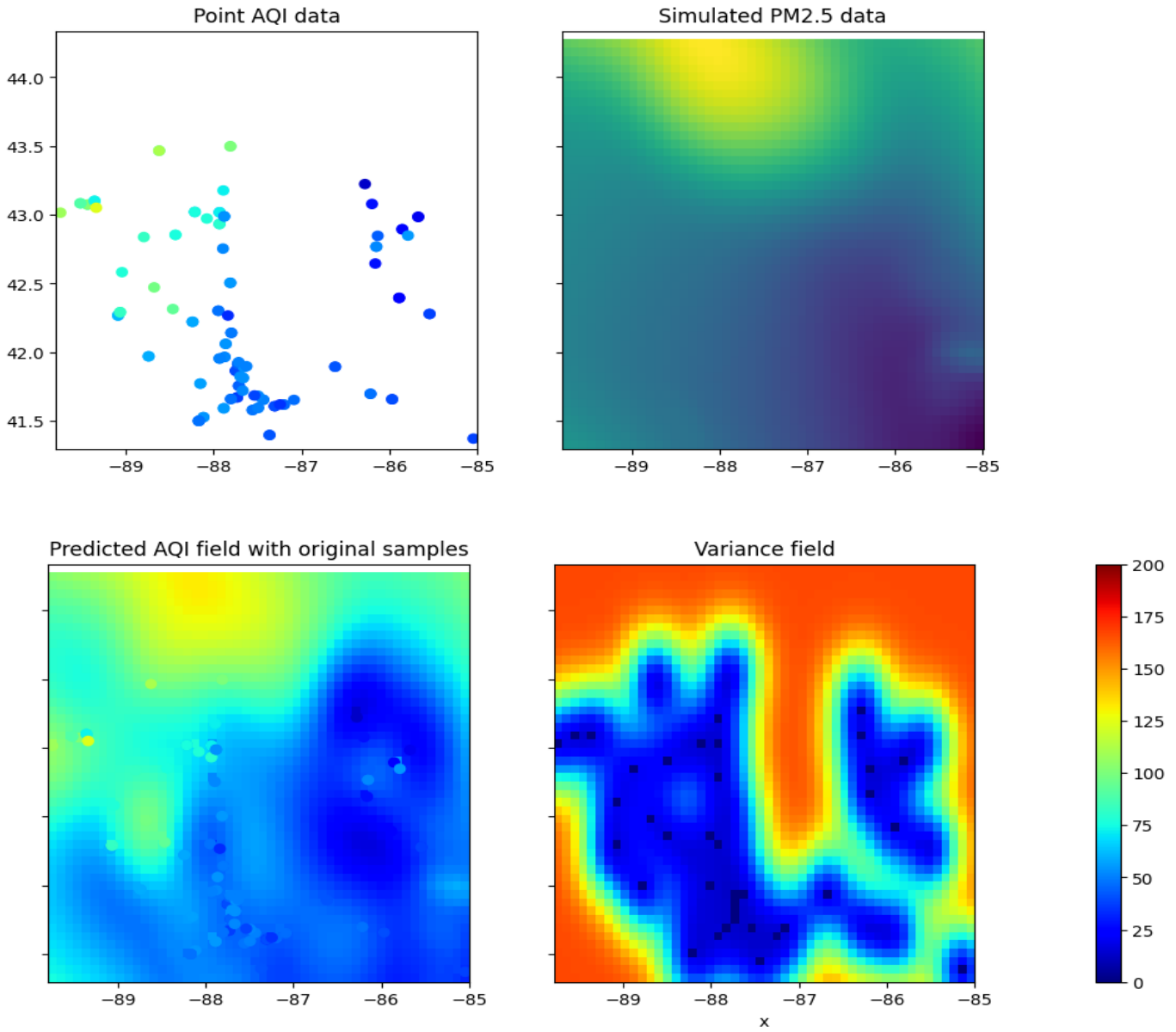
$$\Sigma' = \begin{pmatrix} \Sigma(\boldsymbol{x}_{EPA}, \boldsymbol{x}_{EPA}) & \Sigma(\boldsymbol{x}_{EPA}, \boldsymbol{x}_{PPL}) \\ \Sigma(\boldsymbol{x}_{EPA}, \boldsymbol{x}_{PPL})^T & \Sigma(\boldsymbol{x}_{PPL}, \boldsymbol{x}_{PPL}) + D_A \end{pmatrix}$$

$$\boldsymbol{z}' = [\boldsymbol{z}_{EPA}, (\boldsymbol{z}_{PPL} - a)/b]^{\boldsymbol{T}}$$

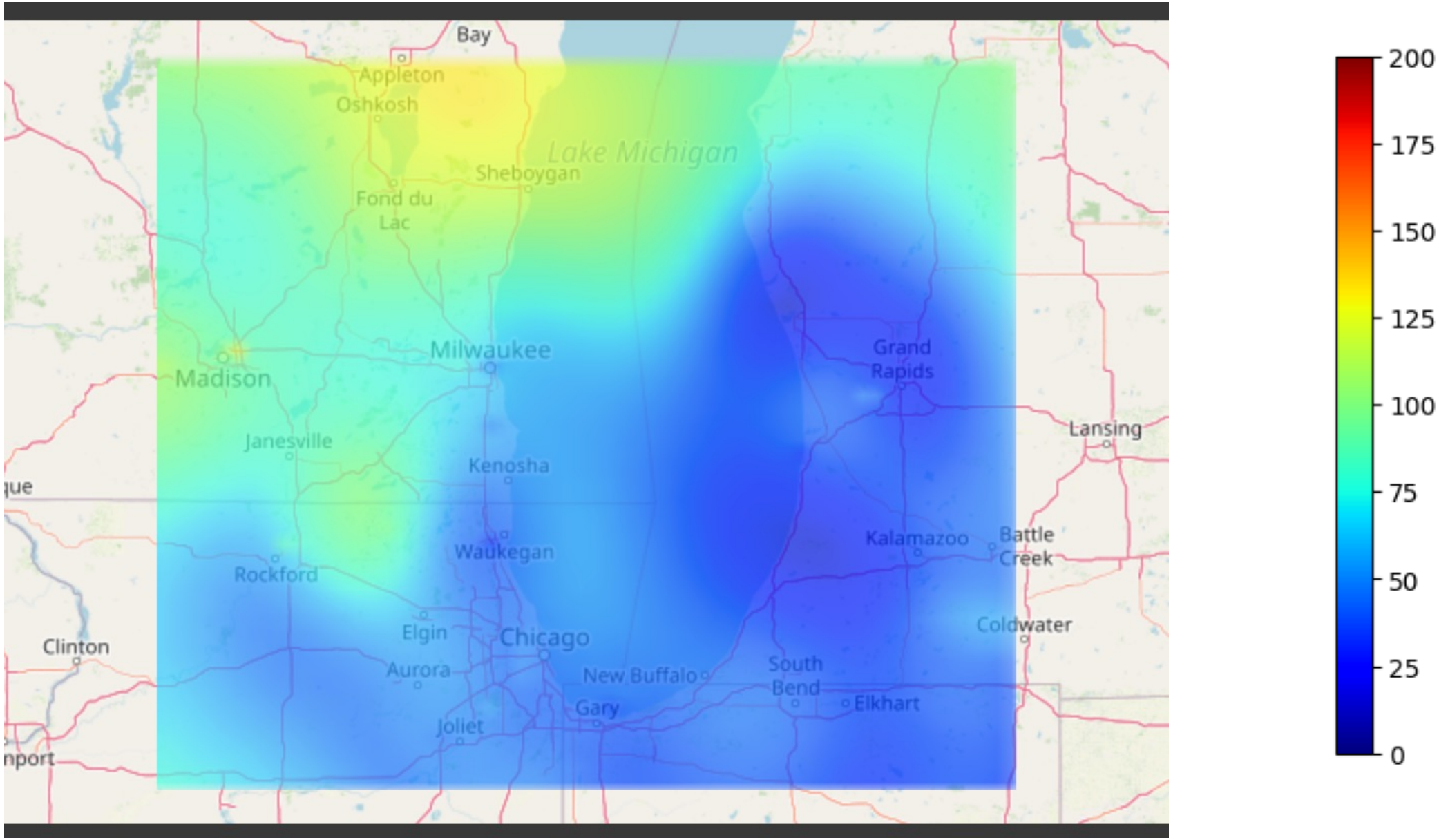
$$p(\boldsymbol{y}|\boldsymbol{z}_{\mathrm{PPL}}, \boldsymbol{z}_{\mathbf{EPA}}, \boldsymbol{z}_{\mathrm{sim}}) = MVN(\boldsymbol{g}, Q)$$

$$\begin{cases} g = f(\boldsymbol{z}_{sim}) + \Sigma_{cross}\Sigma'^{-1}(\boldsymbol{z}' - \boldsymbol{f}(\boldsymbol{z}_{sim}(\boldsymbol{x}_{EPA}, \boldsymbol{x}_{\mathrm{PPL}})) \\ Q = \Sigma_{\mathrm{cross}}\Sigma^{-1}\Sigma_{\mathrm{cross}} \end{cases}$$

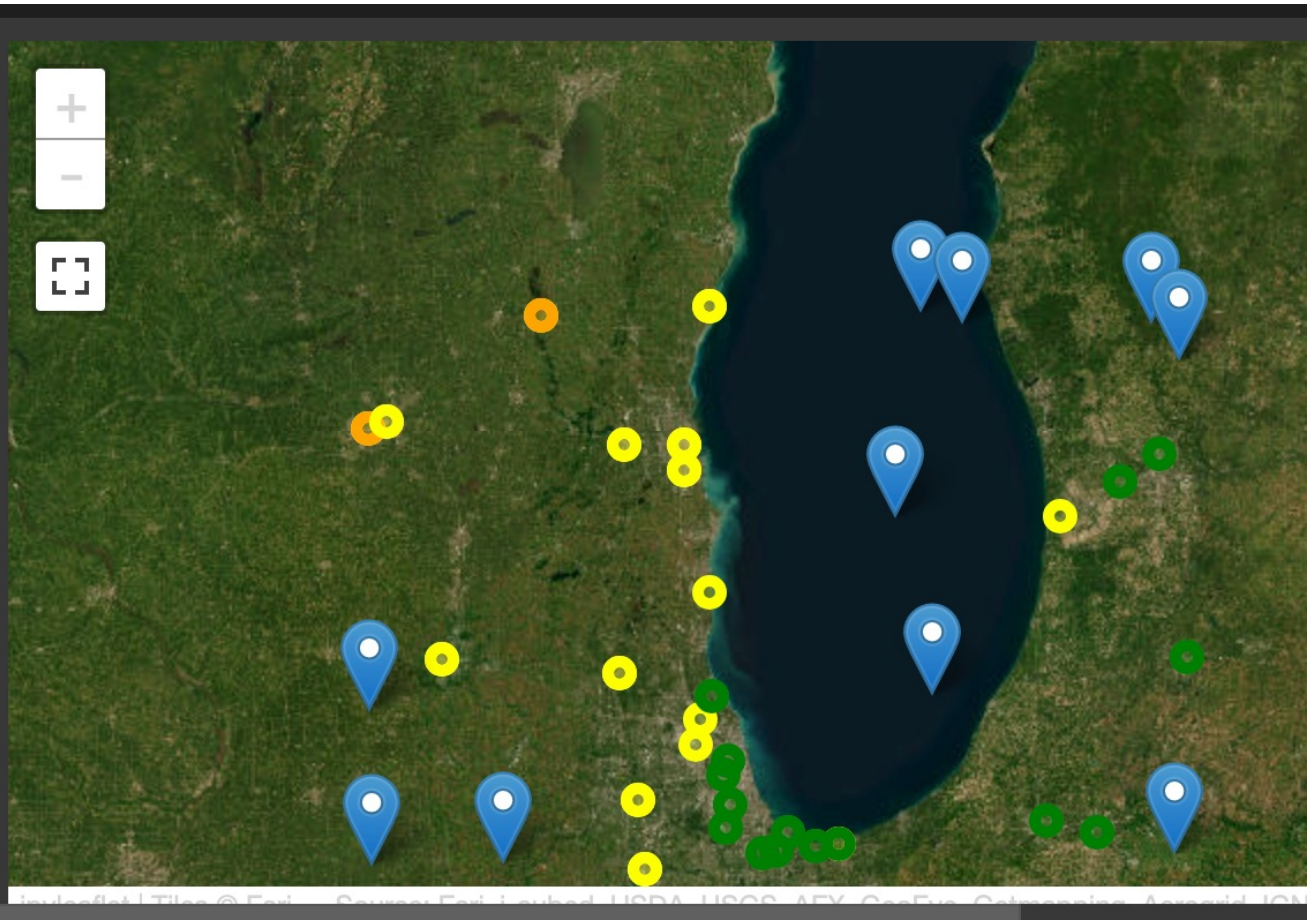
AQI Data Integration: EPA + PPL + Simulation



AQI Data Integration: EPA + PPL + Simulation



Sensor Placement Optimization: GPCAM



```
# AE
my_ae = AutonomousExperimenterGP(
        param_bounds,
        init_hp,
        hp_bounds,
        x_data=x_train,
        y_data=y_train,
        instrument_func = instrument,
        kernel_func =
        kernel_RBF_noise,
        use_inv = True,
        ram_economy = False)
my_ae.train()
my_ae.go(N = 70)
```

Thank You!

Contact
Haruko Wainwright
HMWainw@MIT.EDU

Acknowledgment
DOE Office of Environmental Management
DOE Office of Science