Noise-informed covariance estimation

Matthias Morzfeld

Cecil H. and Ida M. Green Institute of Geophysics and Planetary Physics

Scripps Institution of Oceanography

UC San Diego

Powered by

Caltech

David Vishny
(SIO)

Kyle Gwirtz
(NASA)

Dan Hodyss
(NRL)

Oliver Dunbar
(Caltech)

Eviatar Bach
(Caltech)

Problem formulation

$$\hat{\mathbf{P}} = \frac{1}{n_e - 1} \sum_{i=1}^{n_e} (\mathbf{x}_i - \bar{\mathbf{x}}) \otimes (\mathbf{x}_i - \bar{\mathbf{x}}), \qquad \bar{\mathbf{x}} = \frac{1}{n_e} \sum_{i=1}^{n_e} \mathbf{x}_i$$

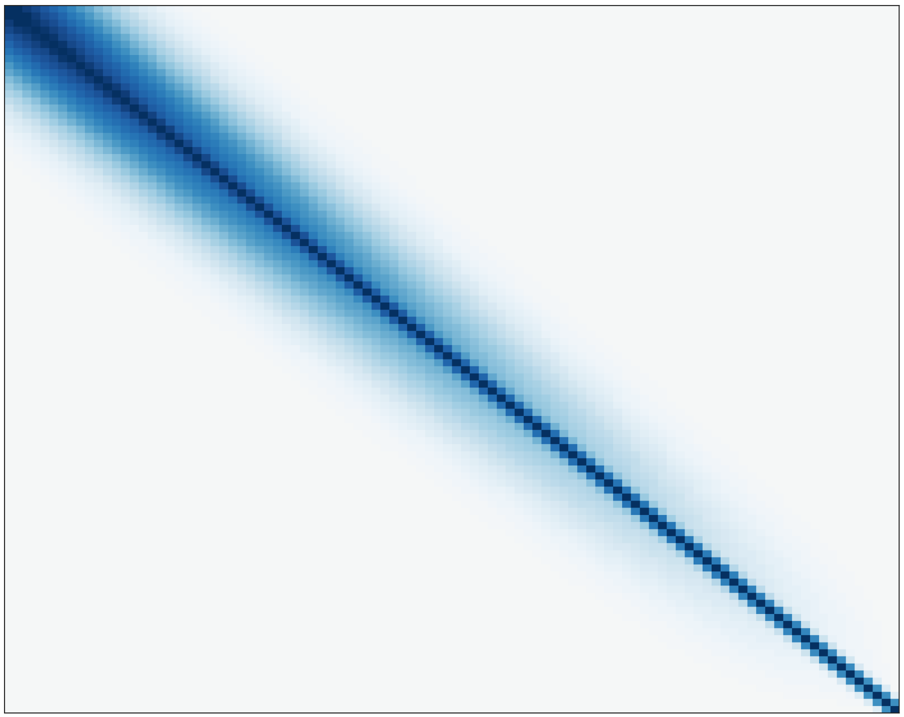
Goal

- Given n_e vectors $\mathbf{x}_i$ of dimension n_x, compute the covariance matrix of the random variable $\mathbf{x}$
- *Caveat*: Few samples, huge dimension, $n_e \ll n_x$

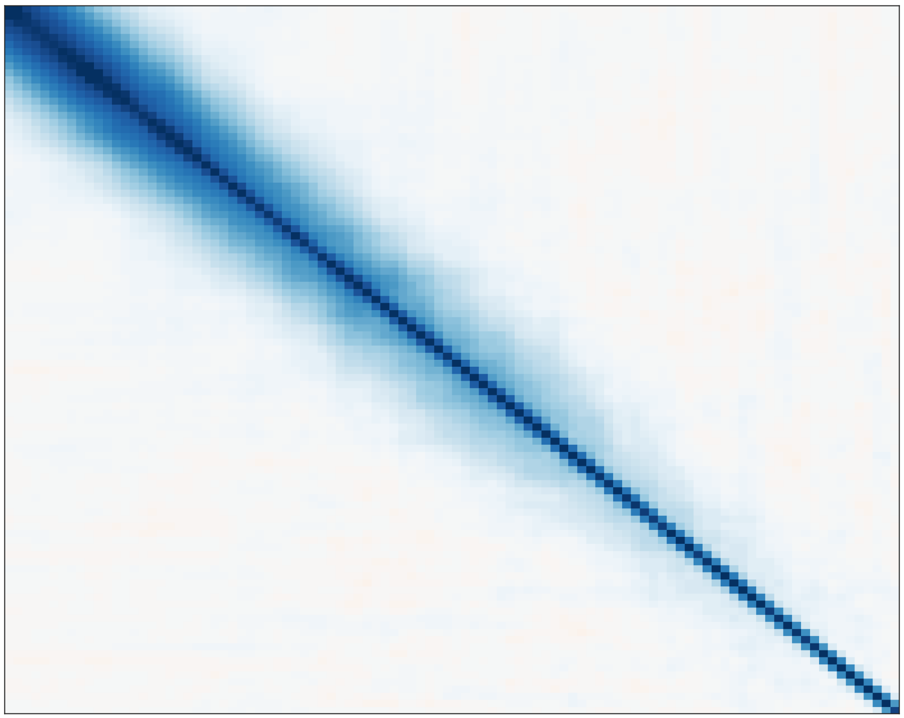
Problem illustration

$$\hat{\mathbf{P}} = \frac{1}{n_e - 1} \sum_{i=1}^{n_e} (\mathbf{x}_i - \bar{\mathbf{x}}) \otimes (\mathbf{x}_i - \bar{\mathbf{x}}), \quad \bar{\mathbf{x}} = \frac{1}{n_e} \sum_{i=1}^{n_e} \mathbf{x}_i$$

True covariance matrix



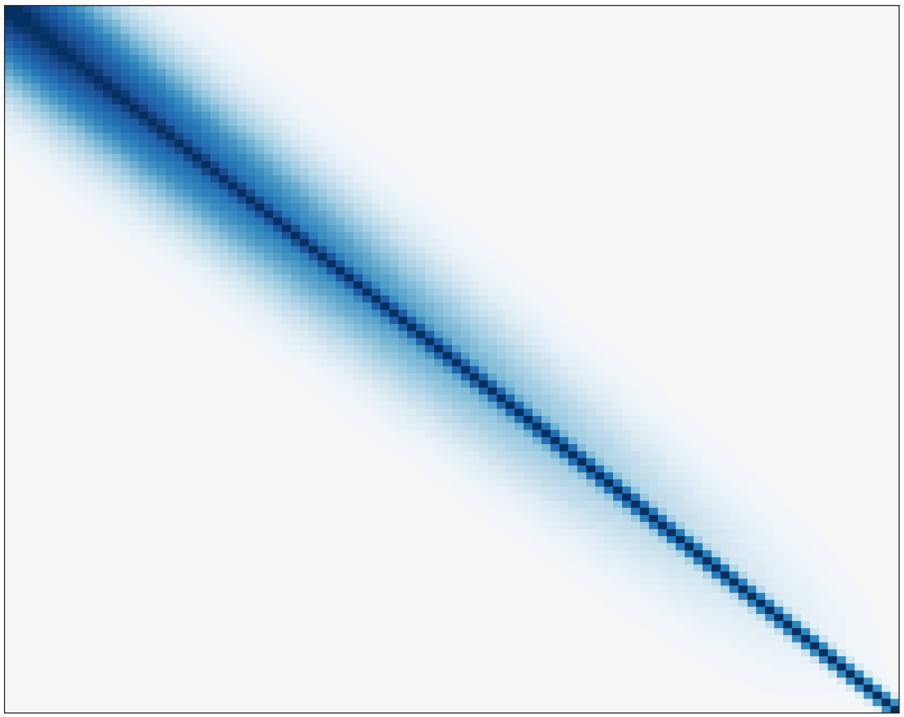
Empirical estimate, $n_x = 100$, $n_e = 2000$



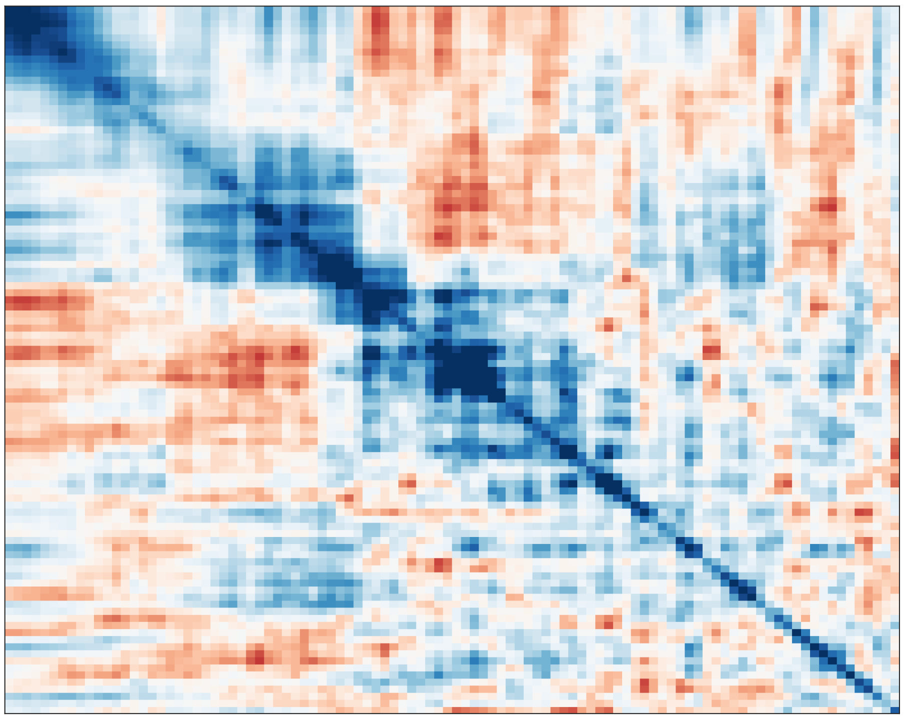
Problem illustration

$$\hat{\mathbf{P}} = \frac{1}{n_e - 1} \sum_{i=1}^{n_e} (\mathbf{x}_i - \bar{\mathbf{x}}) \otimes (\mathbf{x}_i - \bar{\mathbf{x}}), \quad \bar{\mathbf{x}} = \frac{1}{n_e} \sum_{i=1}^{n_e} \mathbf{x}_i$$

True covariance matrix

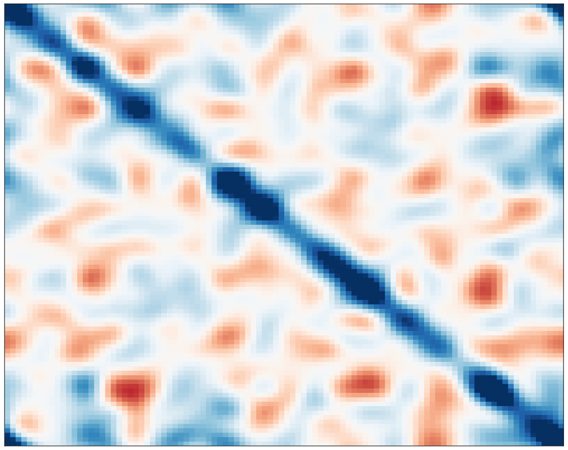


Empirical estimate, $n_x = 100$, $n_e = 20$

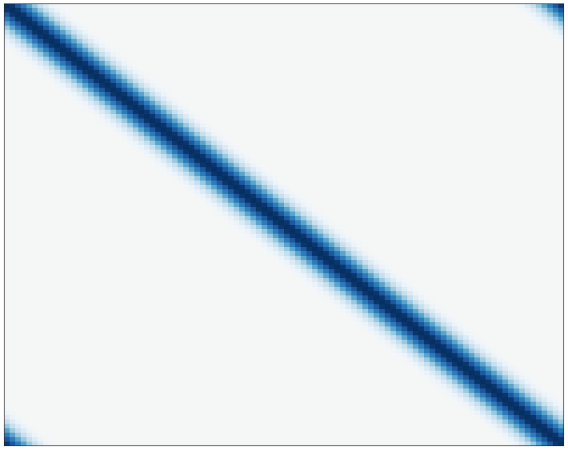


Covariance localization

Empirical covariance



True covariance

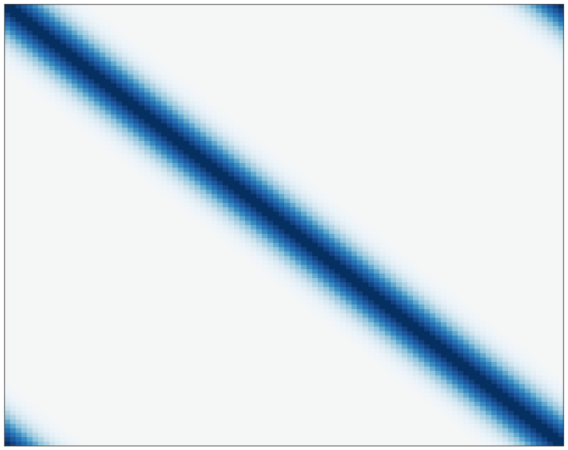


$n_e = 20, \quad n_x = 100$

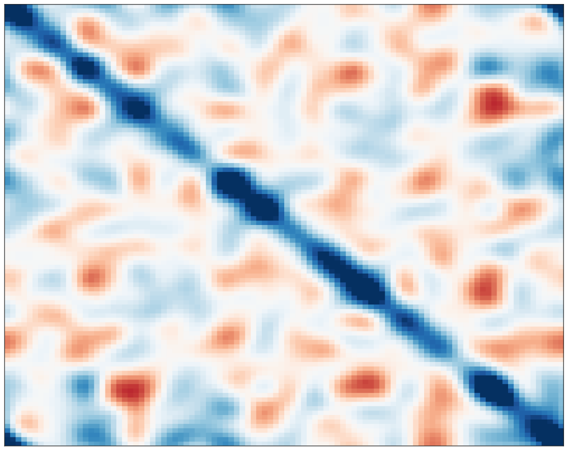
Covariance localization

$$\mathbf{L}(\ell) \circ \hat{\mathbf{P}} = \mathbf{P}_{\text{loc}}$$

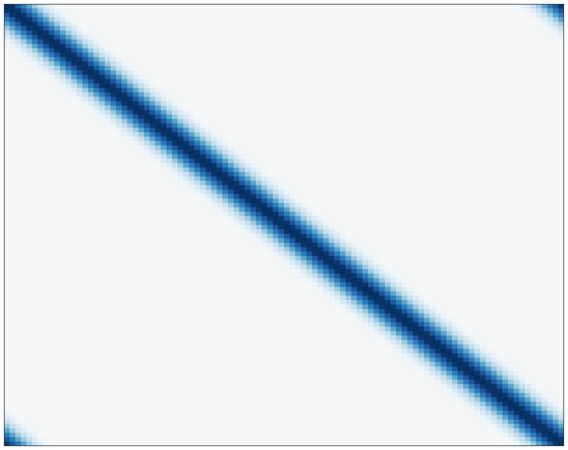
Localization matrix $\circ$ Empirical covariance



$\circ$



True covariance

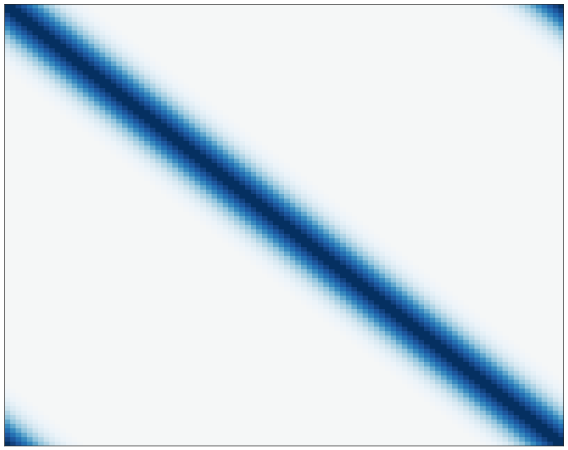


$n_e = 20, \quad n_x = 100$

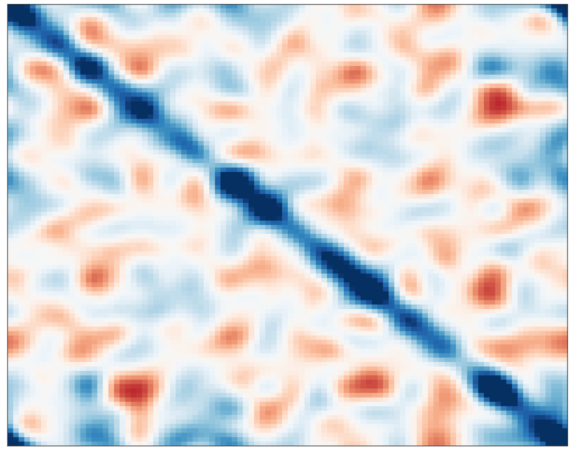
Covariance localization

$$\mathbf{L}(\ell) \circ \hat{\mathbf{P}} = \mathbf{P}_{\mathrm{loc}}$$

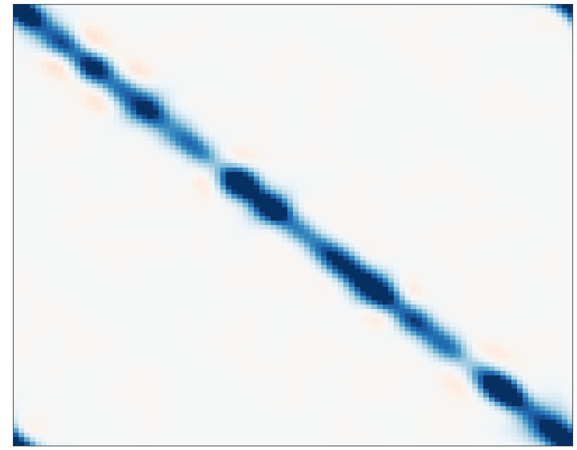
Localization matrix $\circ$ Empirical covariance $=$ Localized estimate



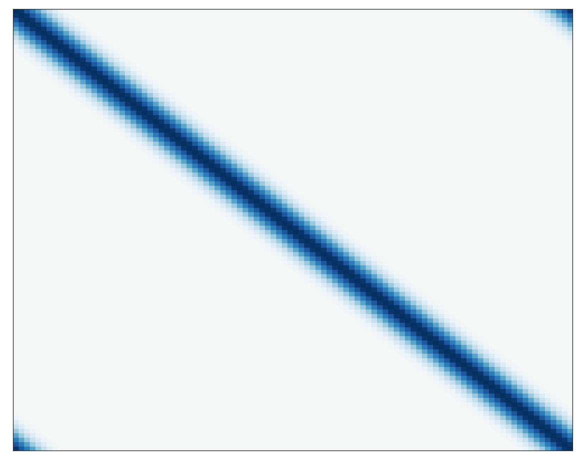
$\circ$



$=$



True covariance

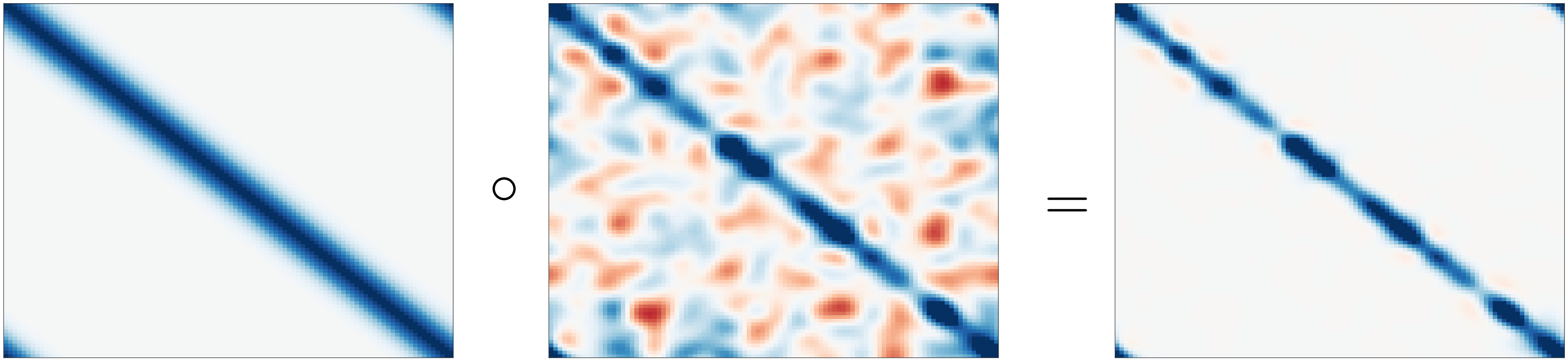


$n_e = 20, \quad n_x = 100$

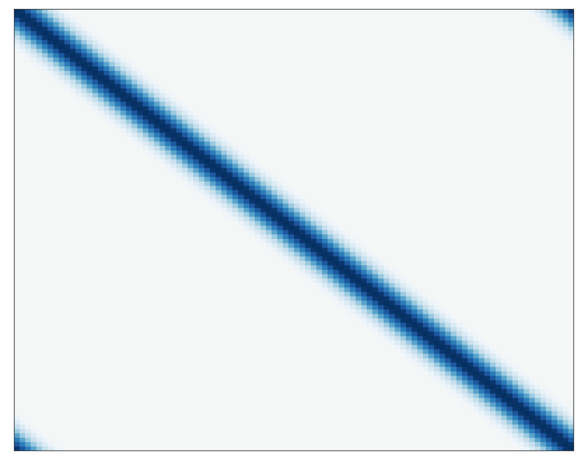
Covariance localization

$$\mathbf{L}(\ell) \circ \hat{\mathbf{P}} = \mathbf{P}_{\text{loc}}$$

Localization matrix $\circ$ Empirical covariance $=$ Localized estimate



True covariance



Math fact

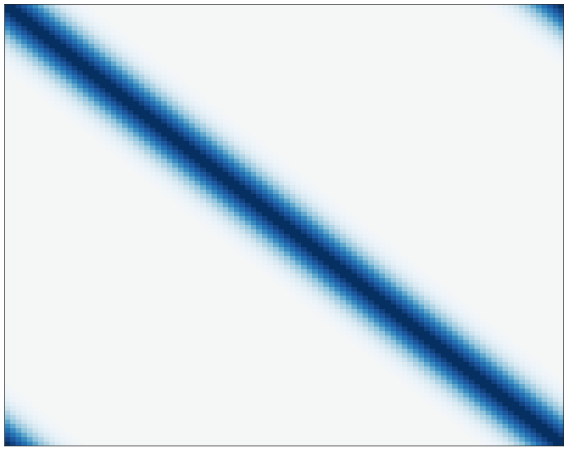
- PSD $\mathbf{L}$ $\rightarrow$ PSD estimate $\mathbf{P}_{\text{loc}}$ (Schur product)

$n_e = 20, \quad n_x = 100$

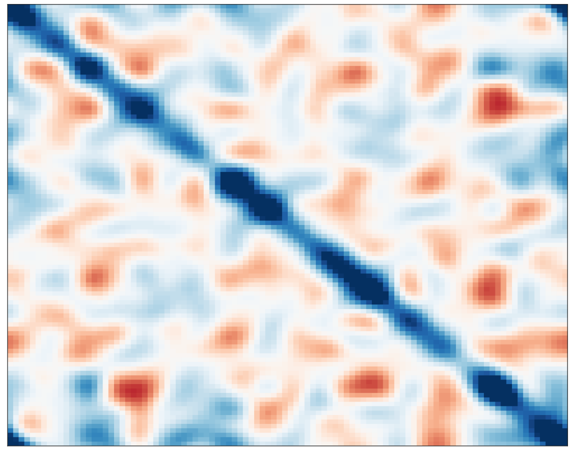
Covariance localization

$$\mathbf{L}(\ell) \circ \hat{\mathbf{P}} = \mathbf{P}_{\text{loc}}$$

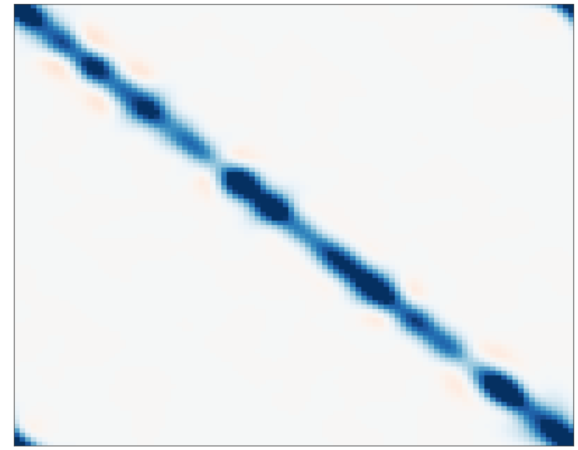
Localization matrix $\circ$ Empirical covariance $=$ Localized estimate



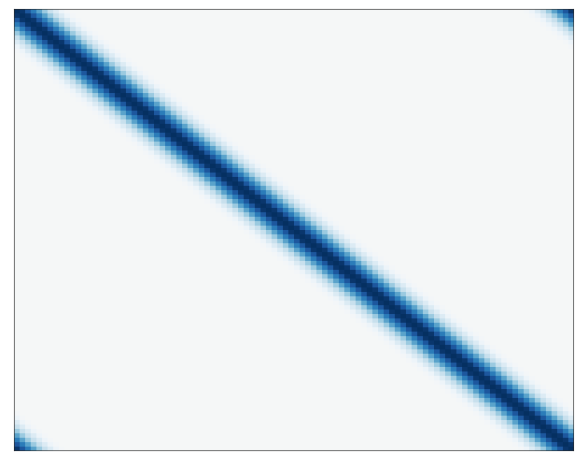
$\circ$



$=$



True covariance



Math fact

- PSD $\mathbf{L}$ → PSD estimate $\mathbf{P}_{\text{loc}}$ (Schur product)

Practical issues

- Pick PSD $\mathbf{L}$ (easy)
- Tune length scale ℓ (hard)

$n_e = 20, \quad n_x = 100$

What if I can't localize?

Localization not always applicable:

- Parameter estimation
- Training of neural networks for sub grid parameterizations (CliMA)
- Assimilation of nonlocal/integrated observations

What if I can't localize?

Localization not always applicable:

- Parameter estimation
- Training of neural networks for sub grid parameterizations (CliMA)
- Assimilation of nonlocal/integrated observations

Covariance estimation that does not rely on a spatial decay of correlation is useful

What if I can't localize?

Localization not always applicable:

- Parameter estimation
- Training of neural networks for sub grid parameterizations (CliMA)
- Assimilation of nonlocal/integrated observations

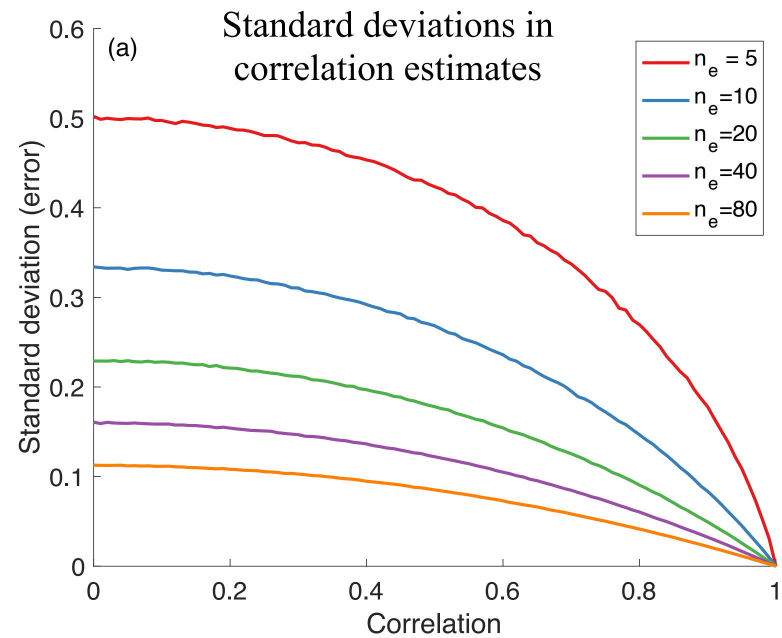
Covariance estimation that does not rely on a spatial decay of correlation is useful

Idea

- Replace spatial decay of correlation with another assumption:
  "Small correlations are noisy, large correlations are trustworthy"
- The method should further be:
  - ***Adaptive*** (no/little tuning)
  - ***Inexpensive*** (no optimization over matrices)
  - ***Guarantee PSD*** estimates (stability of EnKFs)

Foundations of NICE

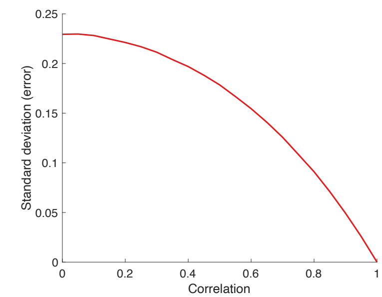
- Near universal truth: *Small correlations are noisy, large correlations are trustworthy*
- Raising correlations to a power implements this idea
- Adaptivity via discrepancy principle (Morozov): *Correct correlations to so that estimates are within expected noise level*



Noise-informed Covariance Estimation (NICE)

1. Compute expected noise level (look-up table)

$$S_\rho = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}(\sigma_{\rho_{ij}})^2}$$



2. Compute a PSD estimate that is "*too strong*": smallest *even* integer that breaks the discrepancy principle

$$\hat{\boldsymbol{\rho}}_\gamma = \hat{\boldsymbol{\rho}}^{\circ\gamma} \circ \hat{\boldsymbol{\rho}},$$
$$\left\|\hat{\boldsymbol{\rho}} - \hat{\boldsymbol{\rho}}_{\gamma^*}\right\|_{\text{Fro}} \geq \delta S_\rho$$

Raise to a power to just break the discrepancy principle

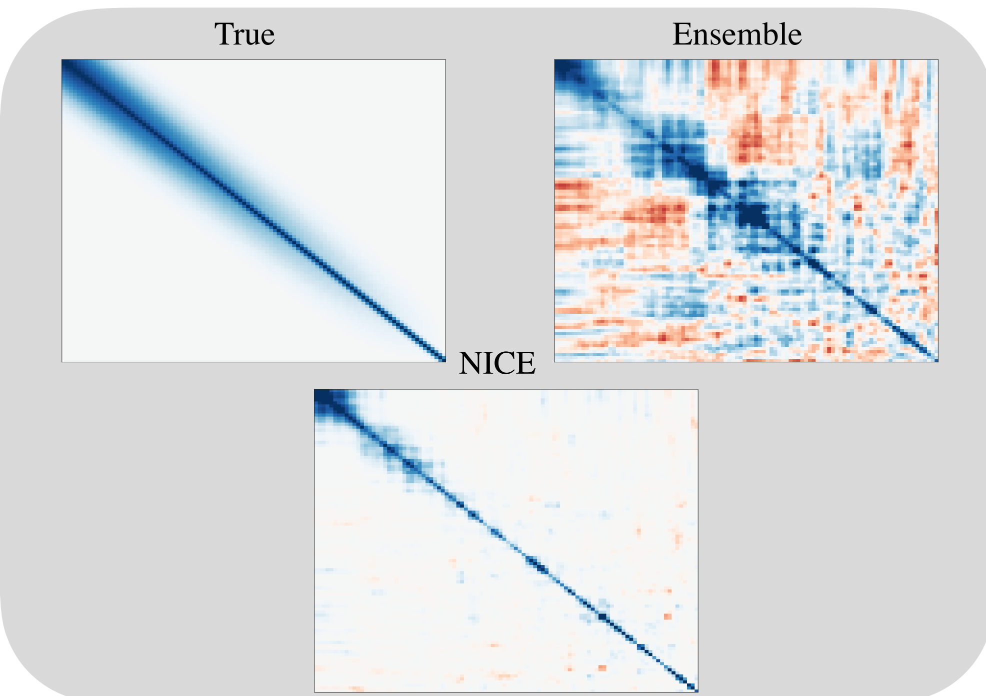
3. Interpolate to find the "right" correction: find the largest interpolation factor that satisfies the discrepancy principle

$$\mathbf{L}(\alpha) = \alpha\hat{\boldsymbol{\rho}}^{\circ\gamma^*} + (1-\alpha)\hat{\boldsymbol{\rho}}^{\circ(\gamma^*-2)}$$
$$\hat{\boldsymbol{\rho}}_\alpha = \mathbf{L}(\alpha) \circ \hat{\boldsymbol{\rho}}$$
$$\|\hat{\boldsymbol{\rho}} - \hat{\boldsymbol{\rho}}_{\alpha^*}\|_{\text{Fro}} \leq \delta S_\rho,$$

Interpolate to satisfy the discrepancy principle

Noise-informed Covariance Estimation (NICE)

1. Looks complicated, but it's a few lines of code
2. Has connections with Jeff Anderson's "*sampling error correction*" but does not require training/ensembles of ensembles and guarantees PSD estimates
3. Can be used within LETKF or EAKF




```
function [Cov_NICE,Corr_NICE] = NICE(X,Y,fac)
Ne = size(X,2) ;
FileName = strcat('std_ro_Ne_',num2str(Ne),'.mat');
load(FileName,'r','stdCrs')

[CorrXY,~] = corr(X',Y');
std_rho = interp1(r,stdCrs,CorrXY,'linear','extrap');
std_rho(CorrXY==1) = 0;
sig_rho = sqrt(sum(sum(std_rho.^2)));

go = 1;
expo2 = 0;
while go == 1
    expo2 = expo2+2;
    L = abs(CorrXY).^expo2;
    Corr_NICER = L.*CorrXY;
    if norm(Corr_NICER - CorrXY,'fro') > fac*sig_rho
        go = 0;
    end
end
expo1 = expo2-2;
rho_exp1 = CorrXY.^expo1;
rho_exp2 = CorrXY.^expo2;

al = 0.1:.1:1;
for kk=1:length(al)
    L = (1-al(kk))*rho_exp1+al(kk)*rho_exp2;
    Corr_NICE = L.*CorrXY;
    if kk>1 && norm(Corr_NICER - CorrXY,'fro') > fac*sig_rho
        Corr_NICE = PrevCorr;
        break
    elseif norm(Corr_NICE - CorrXY,'fro') > fac*sig_rho
        break
    end
    PrevCorr = Corr_NICE;
end

Vy = diag(std(Y,0,2));
Vx = diag(std(X,0,2));

Cov_NICE = Vx*Corr_NICER*Vy;
```

What else can I try?

Competitors

Statistics

- Graphical lasso (*Tibshirani*) — ~~adaptive~~, ~~efficient~~, PSD
- Soft thresholding (*Wainwright*) — ~~adaptive~~, efficient, ~~PSD~~
- Sparse Covariance estimation (*Xue, Ma, Zou*) — ~~adaptive~~, efficient, PSD
- Convex sparse Cholesky selection (*Rajaratnam*) — ~~adaptive~~, ~~efficient~~, PSD
- Optimal localization (*benchmark*) — infeasible
- Sampling error correction (*Anderson, Lee*) — ~~adaptive~~, efficient, ~~PSD~~

Stats methods solve optimization problems

Graphical Lasso

$$F_{\text{G-Lasso}}(\boldsymbol{\Theta}) = \operatorname{tr}(\hat{\mathbf{P}}\boldsymbol{\Theta}) - \log\det(\boldsymbol{\Theta}) + \lambda \sum_{j \neq k} |\boldsymbol{\Theta}_{jk}|$$

Convex Sparse Cholesky Selection

$$F_{\text{CSCS}}(\mathbf{A}) = \operatorname{tr}(\mathbf{A}^T\mathbf{A}\hat{\mathbf{P}}) - 2\log\det(\mathbf{A}) + \lambda \sum_{1 \leq j < i} |\mathbf{A}_{ij}|$$

Sparse covariance estimation

$$F_{\text{SCE}}(\mathbf{P}) = \frac{1}{2}\|\mathbf{P} - \hat{\mathbf{P}}\|_{\text{Fro}}^2 + \lambda \sum_{j \neq k} |\mathbf{P}_{jk}|$$

Test cases

- Synthetic covariance examples (sanity check)
- Cycling DA with EnKF (synthetic and semi-real)
- Inversion of electromagnetic field data
- Training of feed-forward neural net on time averaged data from a chaotic dynamics

NICE is just as good or better than other, tuned methods, but computationally more efficient

Does it work?

Test cases

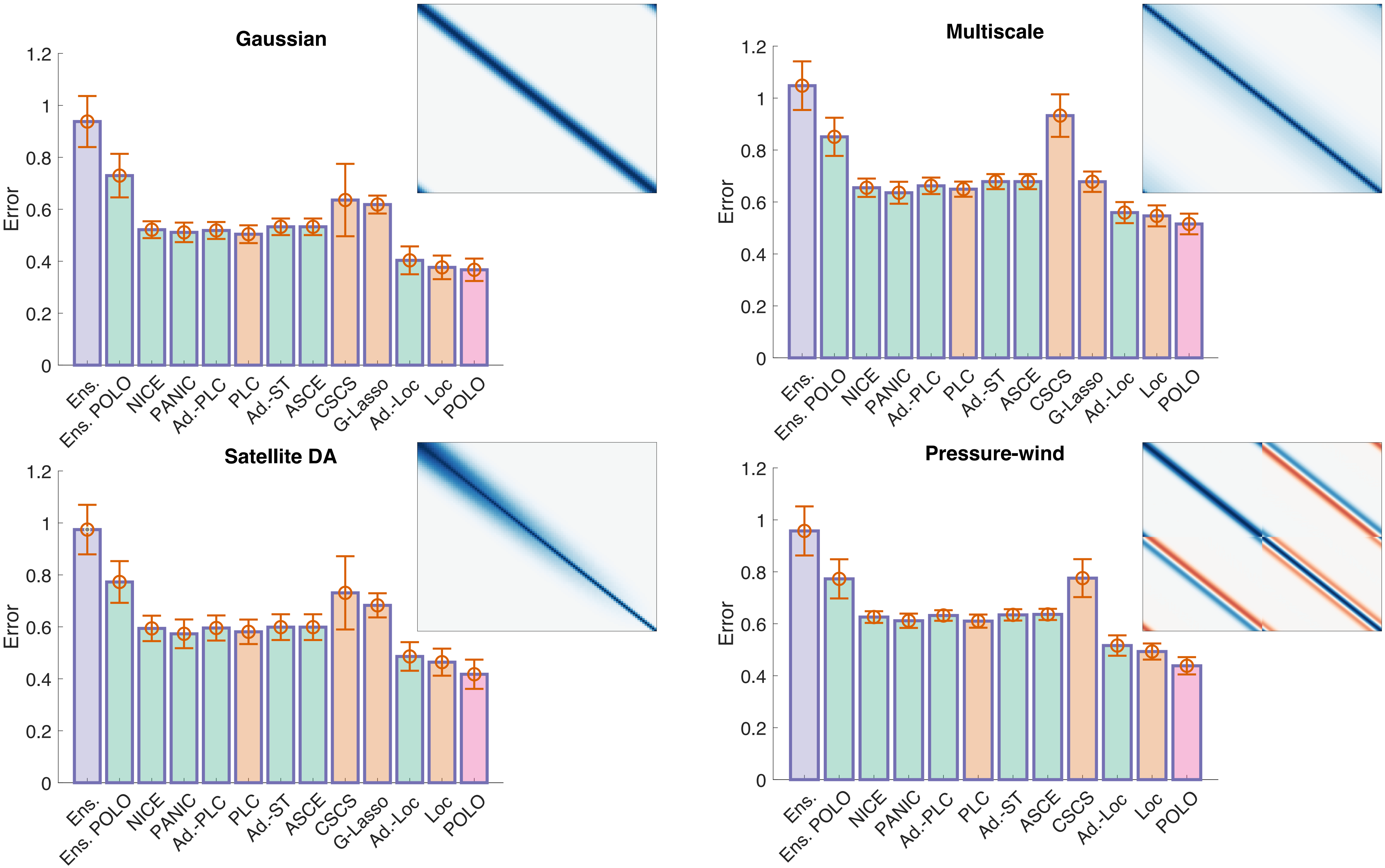
- Synthetic covariance examples (sanity check)
- Cycling DA with EnKF (synthetic and semi-real)
- Inversion of electromagnetic field data
- Training of feed-forward neural net on time averaged data from a chaotic dynamics

NICE is just as good or better than other, tuned methods, but computationally more efficient

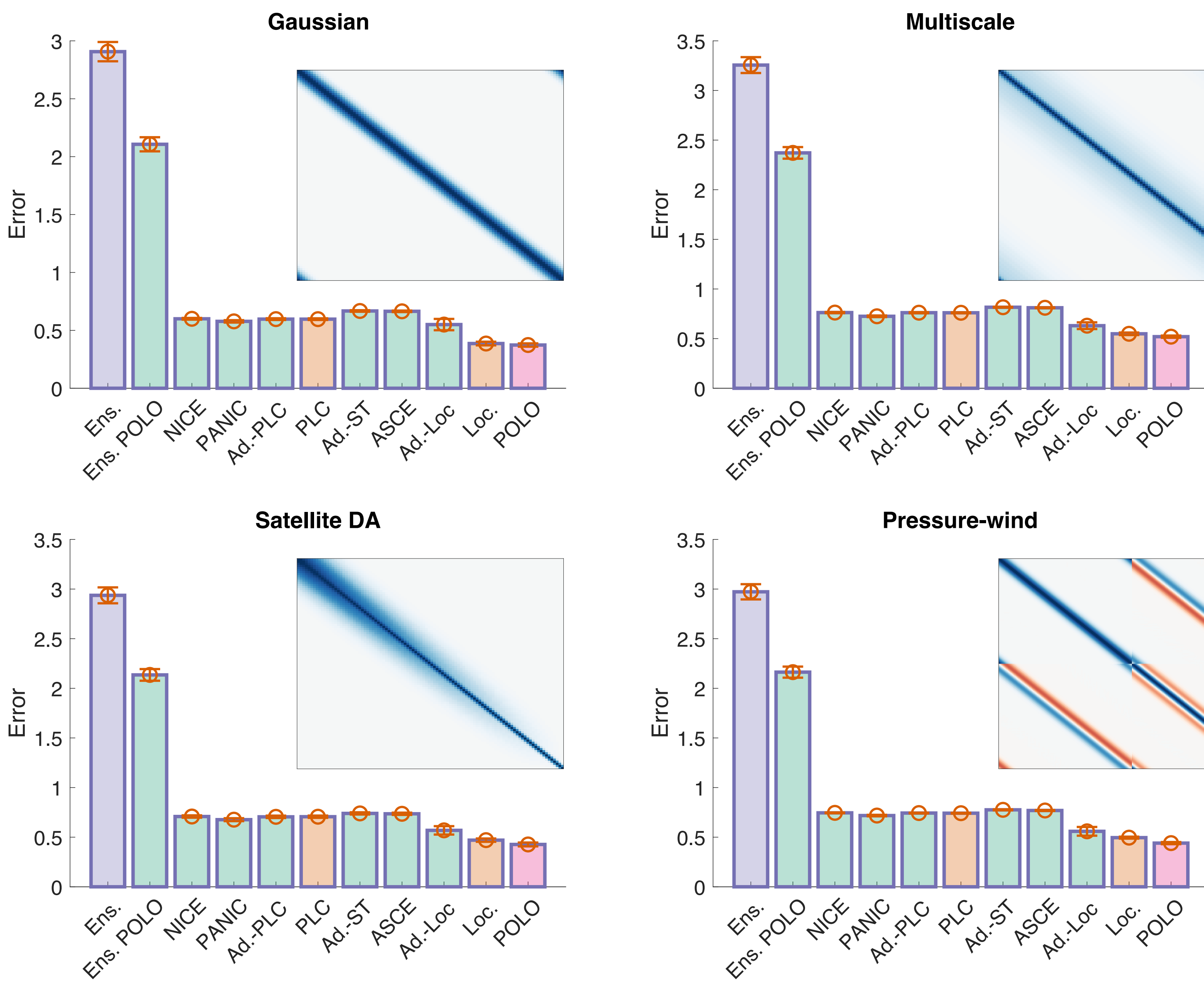
Bonus:

- Discrepancy principle can make methods adaptive
- We created **5** other adaptive schemes
  - *Adaptive localization (Ad.-Loc)*
  - *Adaptive power law corrections (Ad.-PLC)*
  - *Adaptive soft thresholding (Ad.-ST)*
  - *Adaptive sparse covariance estimation (ASCE)*

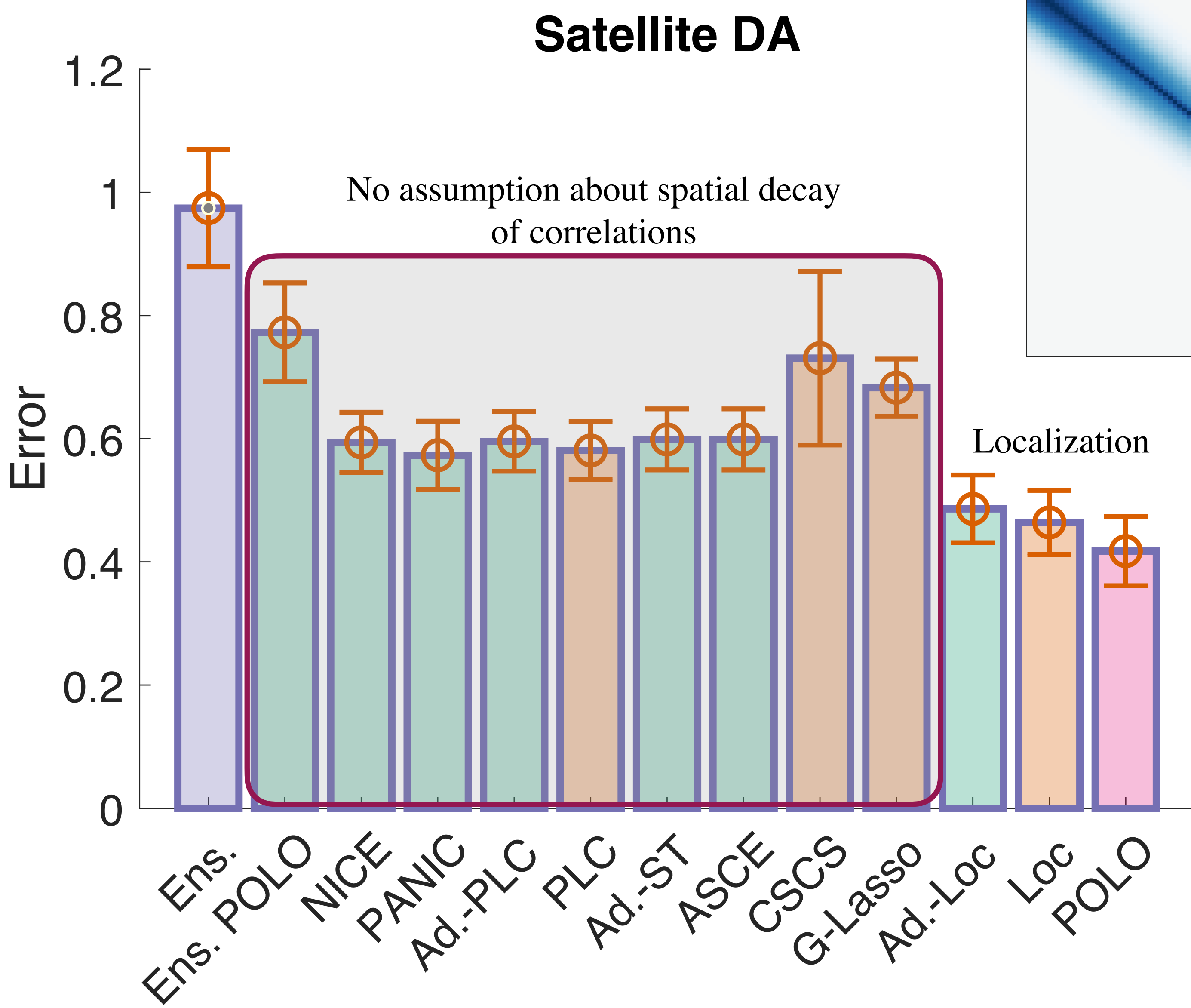
Sanity checks on Gaussians: $n = 100,\ n_e = 20$

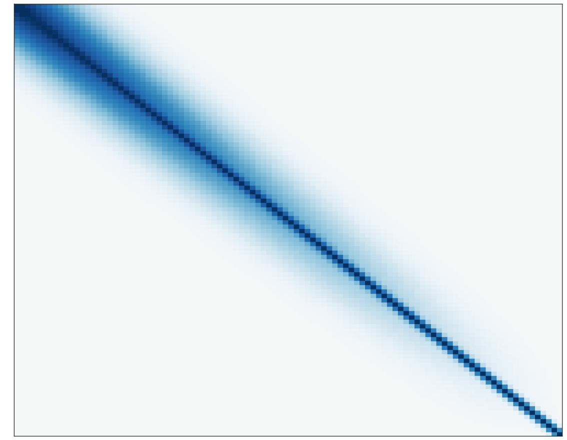


Sanity checks on Gaussians: $n = 1000,\ n_e = 20$

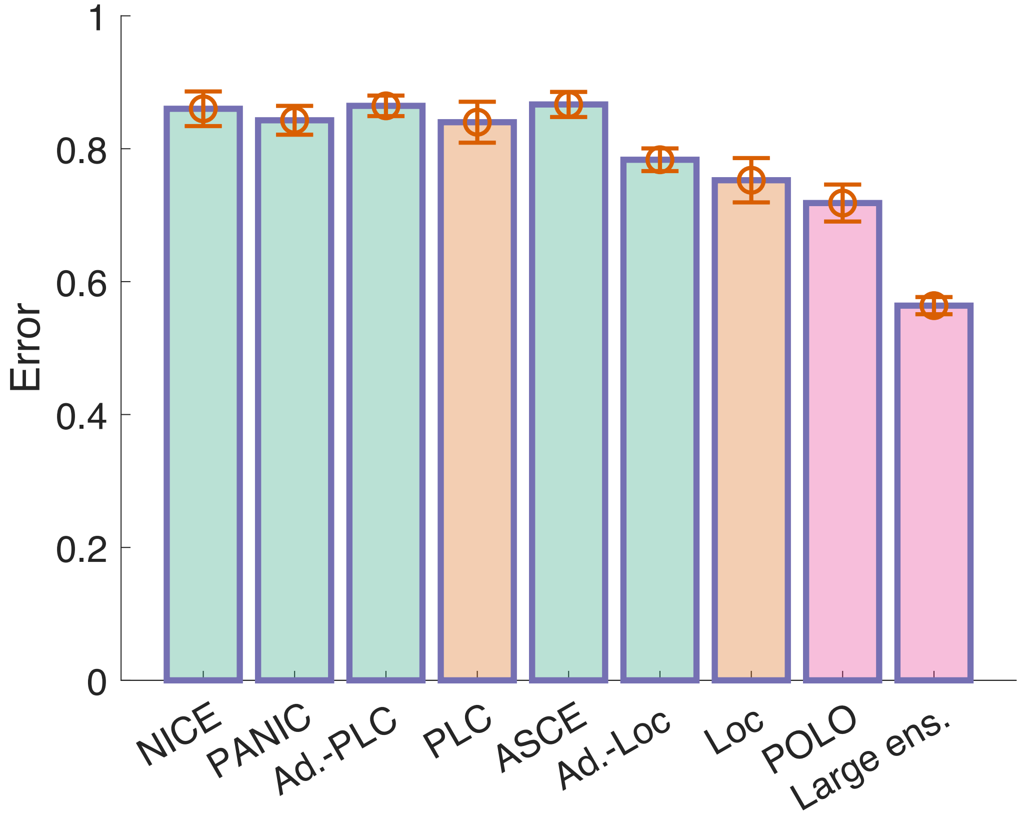


Sanity checks on Gaussians: $n = 100,\ n_e = 20$

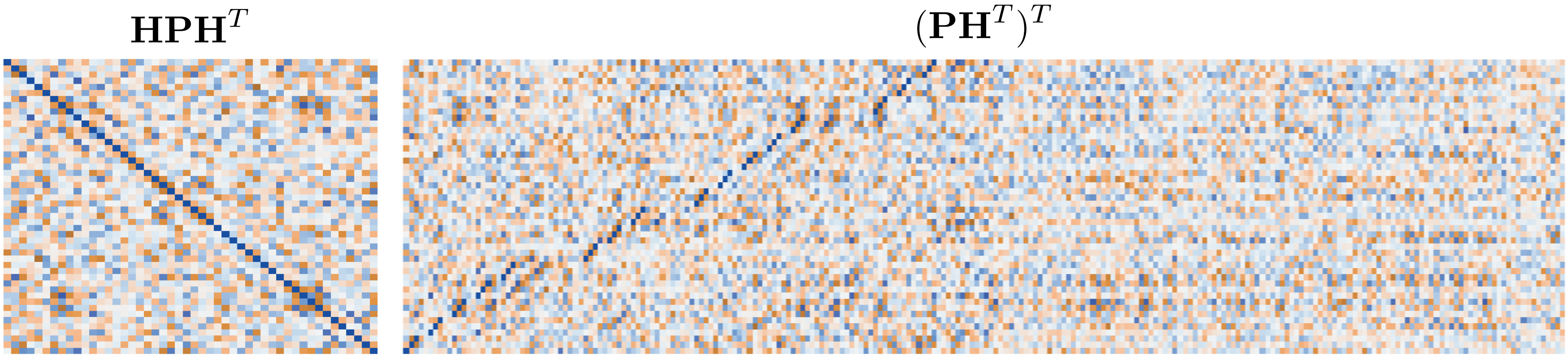




Sanity checks on Lorenz ’95 (*yawn*)

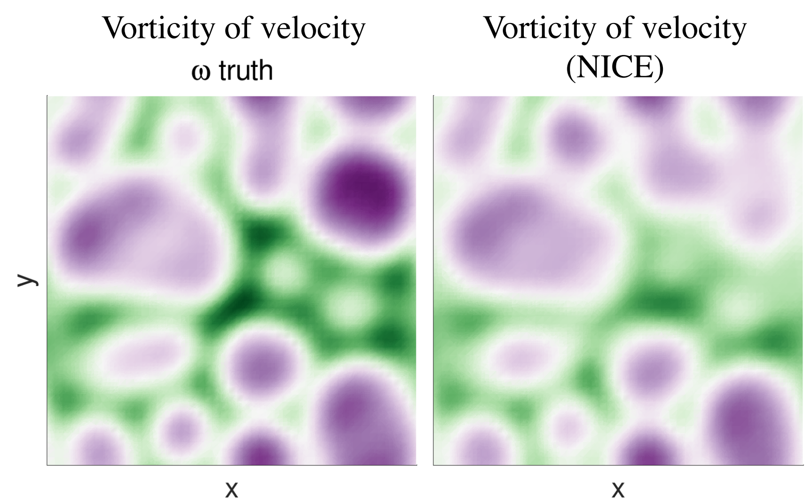


Geomagnetic DA

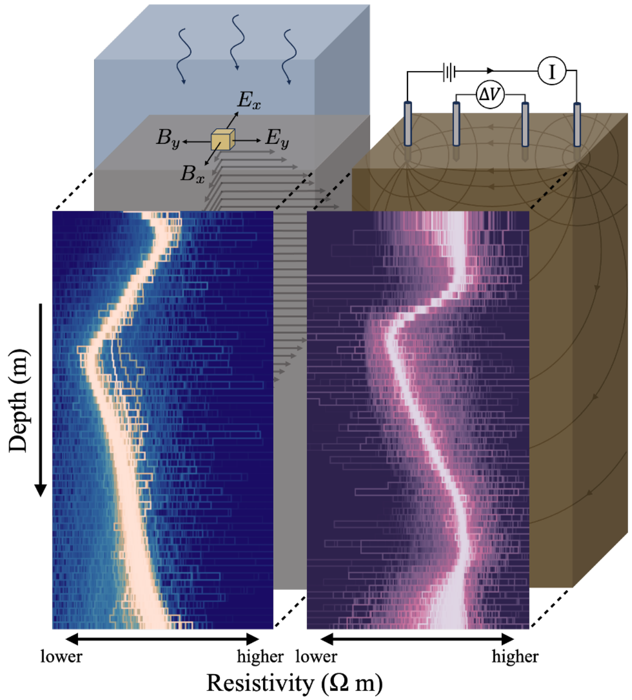


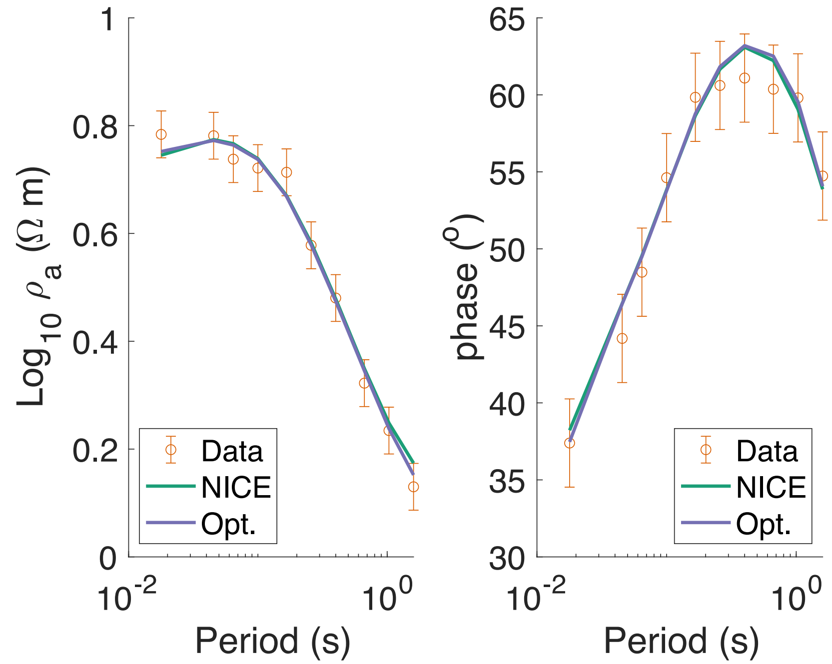
- Correlation structure is erratic/hard to anticipate
- Localization fails on this example
- NICE outperformed a heavily tuned shrinkage scheme
- NICE is not tuned in any way

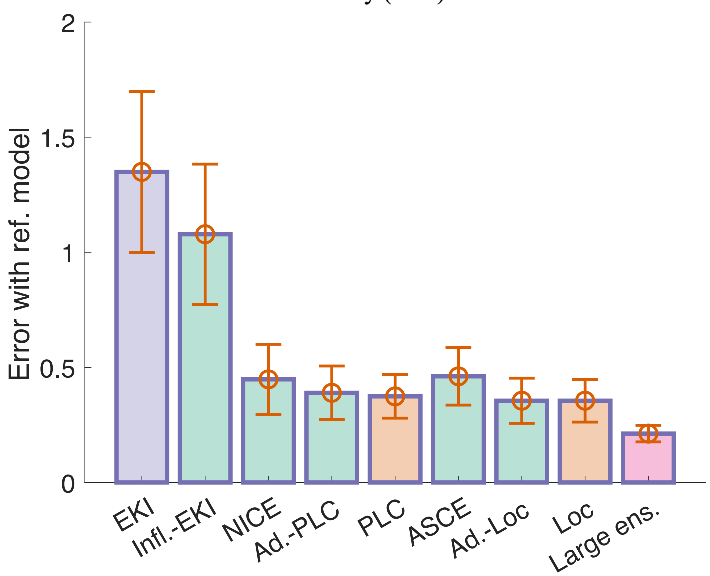
	Error in mag. field	Error in vel. field
Shrinkage (tuned)	1.2	2.0
Ad.-PLC	1.2	2.5
NICE	1.0	1.8
Large ens.	0.7	1.1

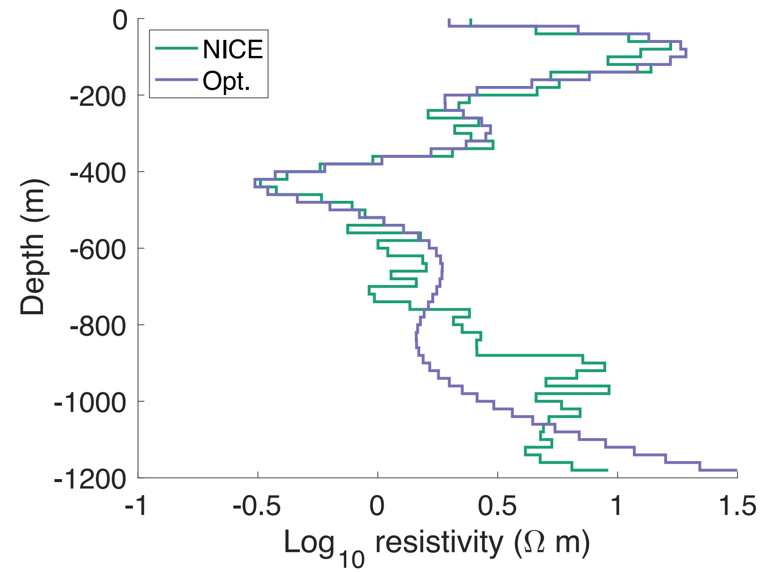


Geophysical inversion of marine magentotelluric data

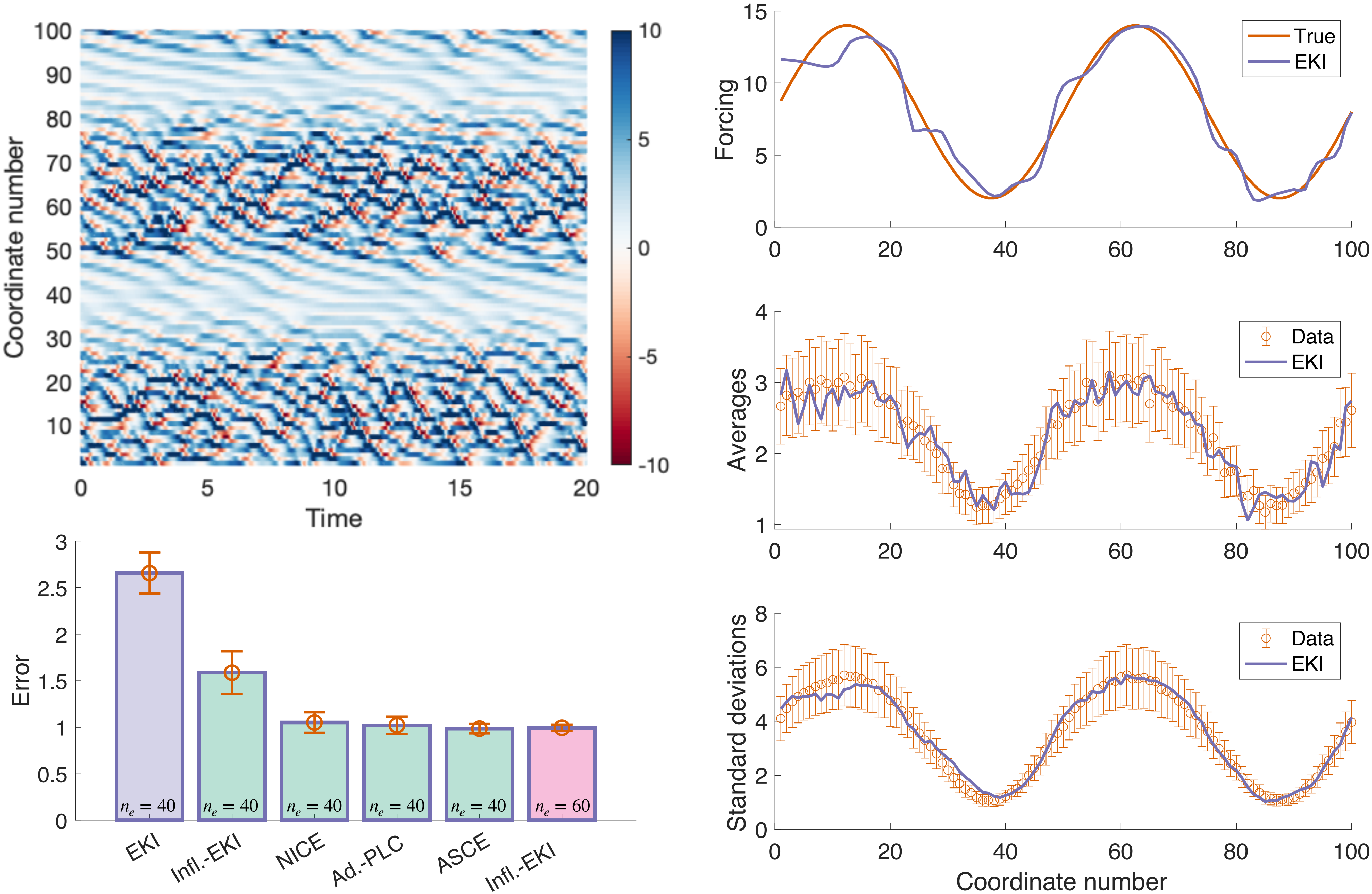








Training a NN with time-averaged data of chaotic dynamics



Facts

- High-dimensional covariance estimation from a small number of samples is (*a*) important; and (*b*) difficult.
- NWP has solved this problem under the assumption that correlation decays with distance (*but this result has not been recognized by statisticians*)

Contribution

- Assumption: *Small/medium correlations are noisy and should be more heavily damped than large ones*
- We turned this idea into a simple, robust, adaptive algorithm (NICE)

Properties of NICE

- Adaptive and tuning-free
- PSD guarantees (but many other, sophisticated schemes don't)
- Works just as well or better than more sophisticated alternatives
- Has proven useful in a large number of diverse test problems

Thank you.

Email: *matti@ucsd.edu*

Thanks to the
Office of Naval Research
for generous support

Facts

- Applied successfully to 3 non-trivial test problems, one involving field data (***we go beyond toy problems***)
- ***Works out of the box*** with *no tuning*, and is easy to implement/use
- ***Early adopters***: CliMA (already part of their Julia package), NASA geomagnetic forecasting — NICE outperformed a scheme that was my former student's main Ph.D. work
- ***As accurate or better*** than state-of-the-art methods from statistics (but computationally less expensive). Did careful comparison to 5 competing methods (serious ones, not non-starters)
- Along the way, ***created 5 other adaptive covariance estimation schemes***, but none is as good as NICE