DeepKriging: Spatially Dependent Deep Neural Networks for Spatial Prediction

Yue Yu[†]

†: Department of Statistics, Indiana University

Motivation: Why go beyond Kriging?

- Many spatial fields are *non-Gaussian* and *nonstationary*.
- Classical Kriging: optimal under Gaussian/stationary assumptions; struggles otherwise.
- Computational bottlenecks: dense covariance inversion $O(N^3)$ time, $O(N^2)$ memory.
- Goal: a scalable predictor capturing complex spatial dependence.

Kriging recap

Setup: Observations $\{(s_i, Z(s_i))\}_{i=1}^N$ at locations $s_i \in \mathbb{R}^2$. Predict $Z(s_*)$ at a new s_*.

$$\widehat{Z}(s_*) = \mu(s_*) + \mathbf{c}(s_*)^\top \mathbf{C}^{-1}\big(\mathbf{Z} - \mu\big),$$
$$\mathbf{C}_{ij} = \mathrm{Cov}\{Z(s_i), Z(s_j)\}, \quad \mathbf{c}_i(s_*) = \mathrm{Cov}\{Z(s_i), Z(s_*)\}.$$

- *Best Linear Unbiased Predictor* (BLUP) under Gaussian/stationary assumptions.
- Explicit dependence on covariance kernel and inversion.

What breaks in practice

- **Nonstationarity:** spatial dependence varies across domain.
- **Nonlinearity:** response-covariate relations can be nonlinear.
- **Scale:** gridded domains/sensor networks $\Rightarrow$ large N.

Related approaches

- Fixed-Rank Kriging / low-rank GPs: scalable but (mostly) linear structures.
- Local Kriging: handles heterogeneity, can be complex to tune.
- Coordinate-only DNNs: flexible but implicit spatial dependence.

Gap: need an explicit, scalable representation of spatial dependence within a flexible learner.

Problem statement & goals

Task

Given covariates $x(s)$ (optional) and observations at training sites, predict $Y(s_*)$ at unobserved locations.

Requirements

- Explicit spatial dependence modeling
- Nonlinear function class (beyond linear BLUP)
- Scalability to large N (GPU-friendly)

DeepKriging: Core idea at a glance

- Embed location s via a bank of compactly supported basis functions $\phi(s) \in \mathbb{R}^K$.
- Concatenate with covariates: $x_\phi(s) = [\, x(s);\; \phi(s)\,]$.
- Feed to a standard MLP: $\widehat{Y}(s) = f_\theta(x_\phi(s))$.

Key: explicit spatial layer $\Rightarrow$ model learns complex, nonstationary structure.

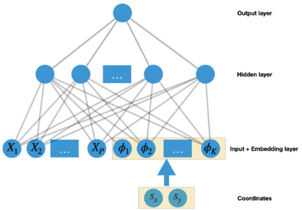

Figure: Visualization of the `DeepKriging` structure in 2D spatial prediction based on a three-layer DNN

Spatial embedding via multi-resolution basis

- Centers $\{c_j^{(\ell)}\}$ on a grid at levels $\ell = 1, \dots, L$ (coarse $\to$ fine).
- Compactly supported Wendland RBFs:

$$\phi_j^{(\ell)}(s) = \psi\left(\frac{\|s - c_j^{(\ell)}\|}{r_\ell}\right)_+, \quad \text{with } \psi(\cdot) \text{ smooth, compact support.}$$

- Locality $\Rightarrow$ nonstationary patterns; multi-resolution $\Rightarrow$ global+local detail.

Input assembly and model

$$x_\phi(s) = \begin{bmatrix} x(s) \\ \phi(s) \end{bmatrix} \in \mathbb{R}^{p+K},$$

$$\widehat{Y}(s) = f_\theta(x_\phi(s)), \quad f_\theta = \mathrm{MLP}(\cdot;\,\theta).$$

- **Regression**: loss $\mathcal{L}(\theta) = \frac{1}{N}\sum_i (Y_i - \widehat{Y}(s_i))^2$ or MAE.
- **Classification** (e.g., exceedance): logistic/cross-entropy on $1\{Y > \tau\}$.
- **Regularization**: dropout, batch-norm, early stopping; prune near-zero basis columns.

Connection to Kriging / FRK

- If f_θ is *linear*, $\widehat{Y}(s)$ is linear in $[x(s), \phi(s)]$.
- With suitable bases, this recovers fixed-rank Kriging structures.
- Allowing nonlinear layers $\Rightarrow$ **strictly larger** function class than Kriging.

Interpretation

DeepKriging $\approx$ Kriging on a rich, explicit spatial feature map, then *nonlinear* transformation.

Infinite-width view (GP interpretation)

- As hidden-layer width $\to \infty$, many MLPs converge to Gaussian processes.
- On inputs $x_\phi(s)$, the induced kernel $k_\infty\big(x_\phi(s), x_\phi(s')\big)$ can be **nonstationary in space**.
- Intuition: nearby s, s' with similar embeddings $\Rightarrow$ larger covariance.

$$f(s) \sim \mathcal{GP}\big(0,\ k_\infty(x_\phi(s), x_\phi(s'))\big).$$

Complexity & scalability

- Avoids dense covariance inversion.
- Training cost scales like $O(N \times \#\text{neurons})$, mini-batch + GPU friendly.
- Basis matrix is sparse/local $\Rightarrow$ efficient embedding computation.

Implementation notes

- Choose L (levels), centers per level, support radii $\{r_\ell\}$.
- Normalize inputs; standard weight inits; `ReLU`/`ELU` activations work well.
- Early stopping on validation MSE; dropout and batch-norm stabilize training.

When to expect gains

- Nonstationary fields with spatially varying structure.
- Nonlinear covariate effects.
- Moderate to large N where Kriging is computationally expensive.

Experiments: simulation setup

- Synthetic 2D nonstationary fields.
- Train/test split; metrics: RMSE/MAE/MAPE.
- Baselines: Kriging; coordinate-only DNN (no explicit spatial layer).

2D Nonstationary data and boxplot

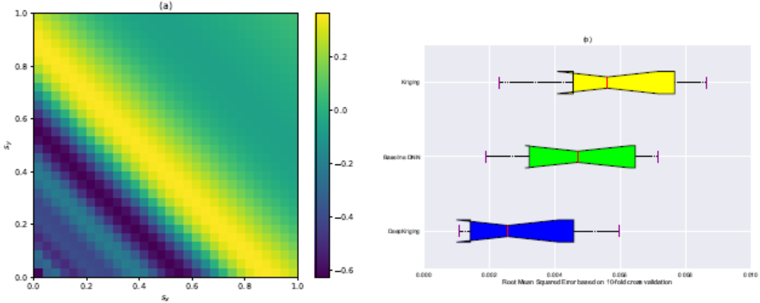

Figure S2: (a) Visualization of the simulated data generated from $Y(\mathrm{s}) = \sin\{30(\bar{s}-0.9)^4\}\cos\{2(\bar{s}-0.9)\} + (\bar{s}-0.9)/2$, where $\mathrm{s} = (s_x, s_y)^T \in [0,1]^2$ and $\bar{s} = (s_x + s_y)/2$. (b) Boxplots of the 10 RMSEs based on the 10-fold cross-validations from DeepKriging (blue), baseline DNN (green), and Kriging (yellow).

2D Nonstationary results comparison with baseline models

Table S2: Model performance for both the training and testing sets based on 10-fold cross-validation. Data are generated from $Y(s) = \sin\{30(\bar{s} - 0.9)^4\}\cos\{2(\bar{s} - 0.9)\} + (\bar{s} - 0.9)/2$, where $s = (s_x, s_y)^T \in [0,1]^2$ and $\bar{s} = (s_x + s_y)/2$. RMSEs and MAPEs are computed, and the mean and standard deviations (SD) for the 10 sets of validation errors are given. DeepKriging prediction is based on the MSE loss. The baseline DNN includes only coordinates s in the features. The Kriging prediction uses an estimated exponential covariance function.

Models	DeepKriging		Baseline DNN		Kriging	
	Mean	SD	Mean	SD	Mean	SD
Training set						
RMSE ($\times 10^{-3}$)	**.585**	.869	4.255	2.384	5.969e-4	7.287e-5
MAPE	**1.269**	1.313	5.296	3.170	5.749e-7	5.200e-8
Testing set						
RMSE ($\times 10^{-3}$)	**3.466**	3.417	6.934	7.672	8.552	9.562
MAPE	**5.330**	3.885	6.152	3.221	0.007	0.005

Real Dataset & task: U.S. $PM_{2.5}$

- Gridded CONUS domain (single date); $PM_{2.5}$ observations at $\sim$604 cells.
- Predict over remaining $\sim$7,102 grid cells.
- Covariates available on grid: NARR meteorological variables (e.g., temp, wind, humidity).

Preprocessing & protocol

- Standardize covariates; handle missingness.
- 10-fold cross-validation.
- Two tasks:
  1. **Regression**: predict $PM_{2.5}$ concentration.
  2. **Classification**: exceedance of $12g/m^3$ threshold.

Quantitative results — Regression

Parameters	DeepKriging		Baseline DNN		Kriging	
	Mean	SD	Mean	SD	Mean	SD
MSE	**1.632**	.572	3.632	.925	3.361	.773
MAE	**.892**	.103	1.448	.162	1.365	.178
ACC	**95.2%**	2.6%	89.6%	4.8%	88.5%	4.6%

Figure: Model performance based on the 10-fold cross-validation. MSEs and MAEs of the predictions, as well as classification accuracy (ACC) for predicting $PM_{2.5}$ concentrations above $12.0\ \mu g = m^3$ are used as the validation criteria. Mean and standard deviation (SD) of the 10 sets of validation errors or accuracy are provided in the table.

Qualitative maps

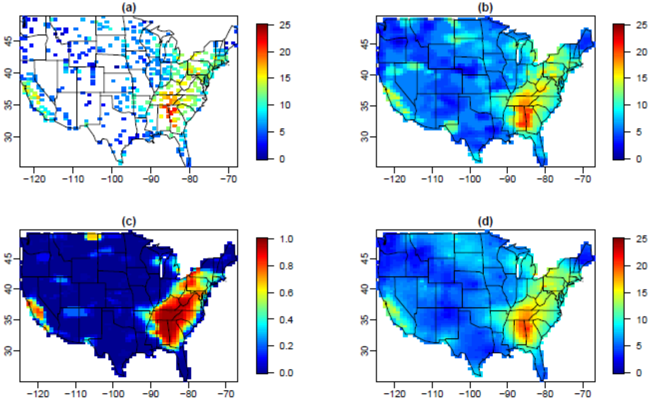

Figure 2: (a) $PM_{2.5}$ concentration ($\mu g/m^3$) collected from monitoring stations. (b) Predicted $PM_{2.5}$ concentration using DeepKriging. (c) Predicted risk of high pollution $\mathbb{P}\{PM_{2.5} > 12\ \mu g/m^3\}$ based on distribution prediction using DeepKriging. (d) Predicted $PM_{2.5}$ concentration using Kriging.

Theoretical Insights: Summary

- **Connection to Kriging**: DeepKriging approximates Kriging under certain linear settings.
- **Function Space Extension**: Nonlinear MLPs allow DeepKriging to operate in a strictly larger function class.
- **Gaussian Process Interpretation**: Infinite-width networks converge to GPs with nonstationary kernels.
- **Covariance Approximation**: Basis functions in DeepKriging approximate standard covariance structures.

Why DeepKriging? Practical Advantages

- **Scalable**: Avoids $O(N^3)$ complexity of traditional Kriging.
- **Flexible**: Handles non-Gaussian, nonstationary, and nonlinear data.
- **Interpretable Spatial Embedding**: Multi-resolution basis functions give explicit spatial structure.
- **Unified Framework**: Regression, classification, and uncertainty quantification in one model.

Limitations and Challenges

- **Model Design**: Selection of basis function types, resolutions, and network depth is non-trivial.
- **Missing Covariates**: Missing features at prediction sites may bias inference.
- **Interpretability**: Like other DNNs, internal representations can be opaque.
- **Hyperparameter Tuning**: Training requires careful regularization and validation.

Questions?

Thank you!

References I