Adaptive Conformal Inference in Operator Models with Spectral Localization

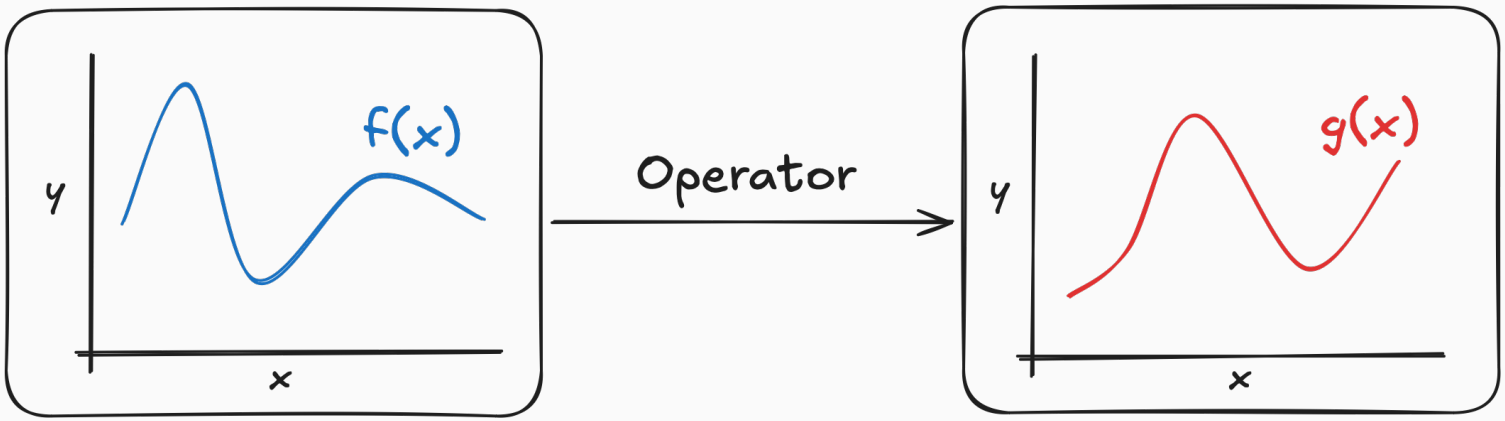
Trevor Harris

March 27, 2025

University of Connecticut
Statistics Department

IMSI, Chicago, IL

Operators and Operator Models

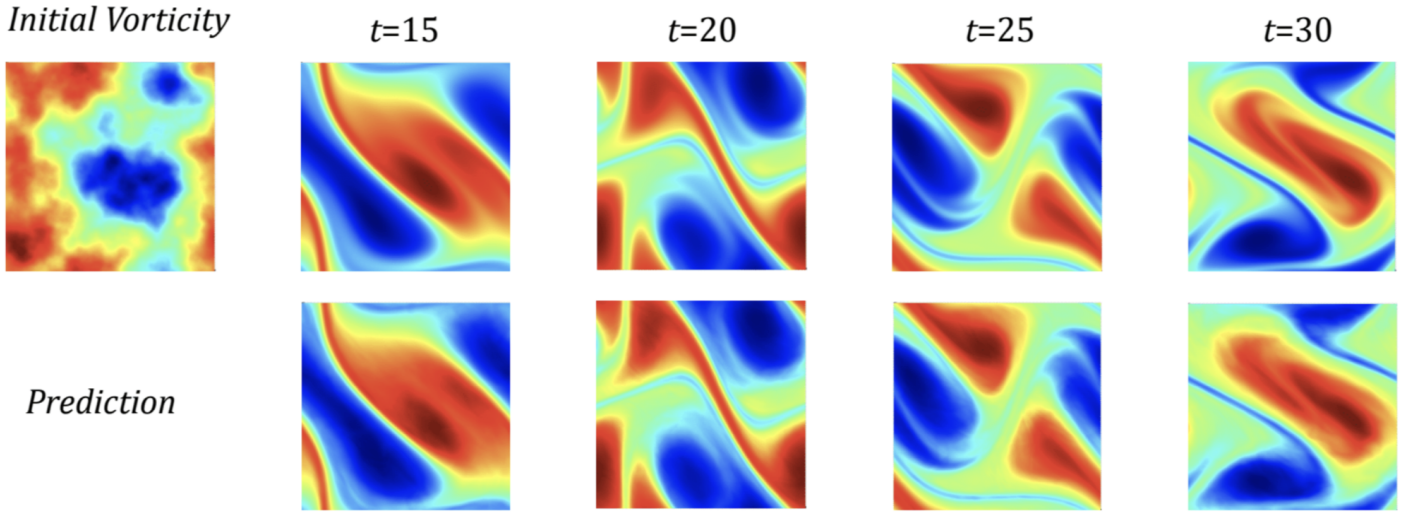


- An **operator** is a map between *functions*
  - Takes a function f and returns a function g (e.x. $\nabla[\sin(x)] = \cos(x)$)
- An **operator model** is a regression model between functions

$$\Gamma_\theta : \mathcal{F} \mapsto \mathcal{G} \tag{1}$$

- $\mathcal{F}$ and $\mathcal{G}$ are function spaces, e.x. $\mathcal{F} = \mathcal{G} = \mathcal{L}^2([0,1]) = \{f : \int_{[0,1]} f(x)^2 dx < \infty\}$
- $\theta \in \Theta$ denotes the model parameters

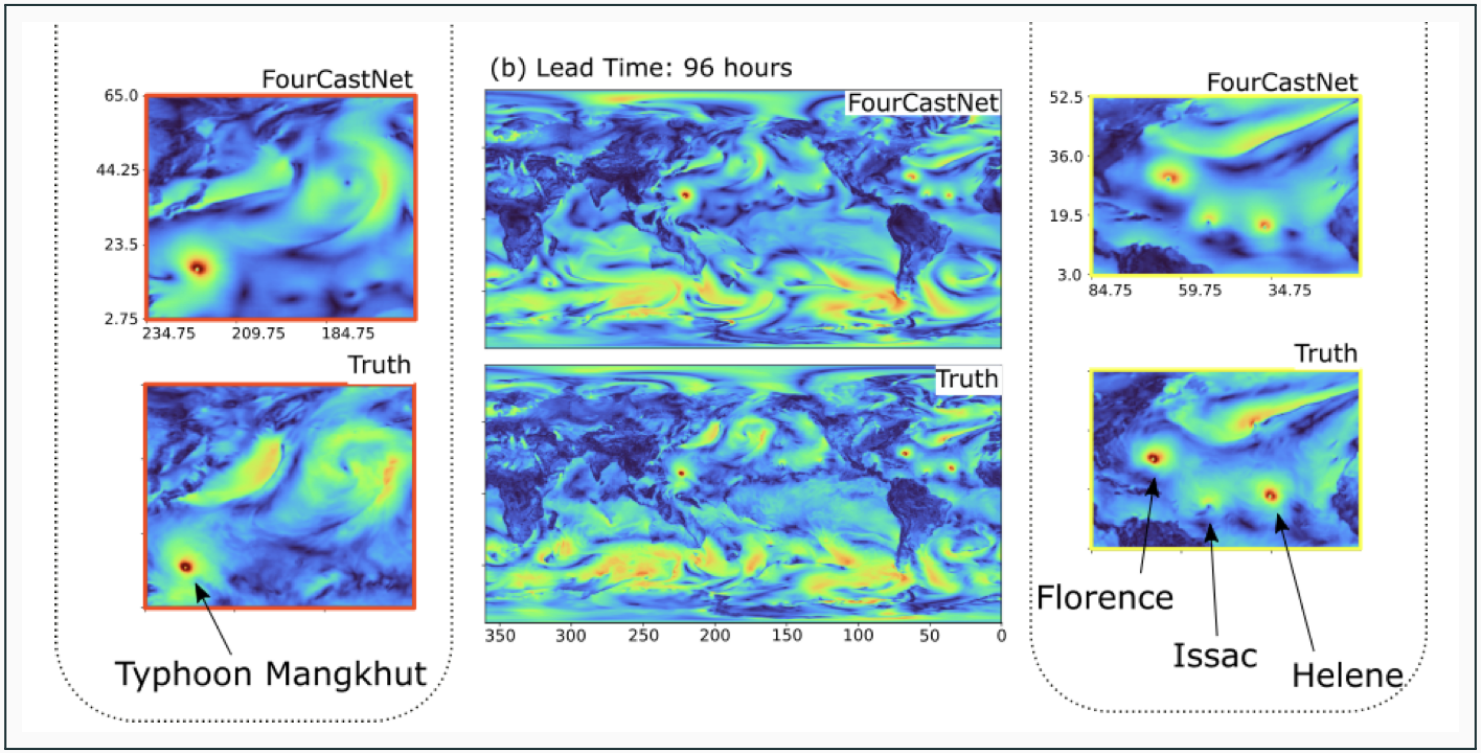
Operator learning



Zero-shot super-resolution: Navier-Stokes Equation with viscosity $\nu = 1e{-}4$; Ground truth on top and prediction on bottom; trained on $64 \times 64 \times 20$ dataset; evaluated on $256 \times 256 \times 80$ (see Section 5.4).

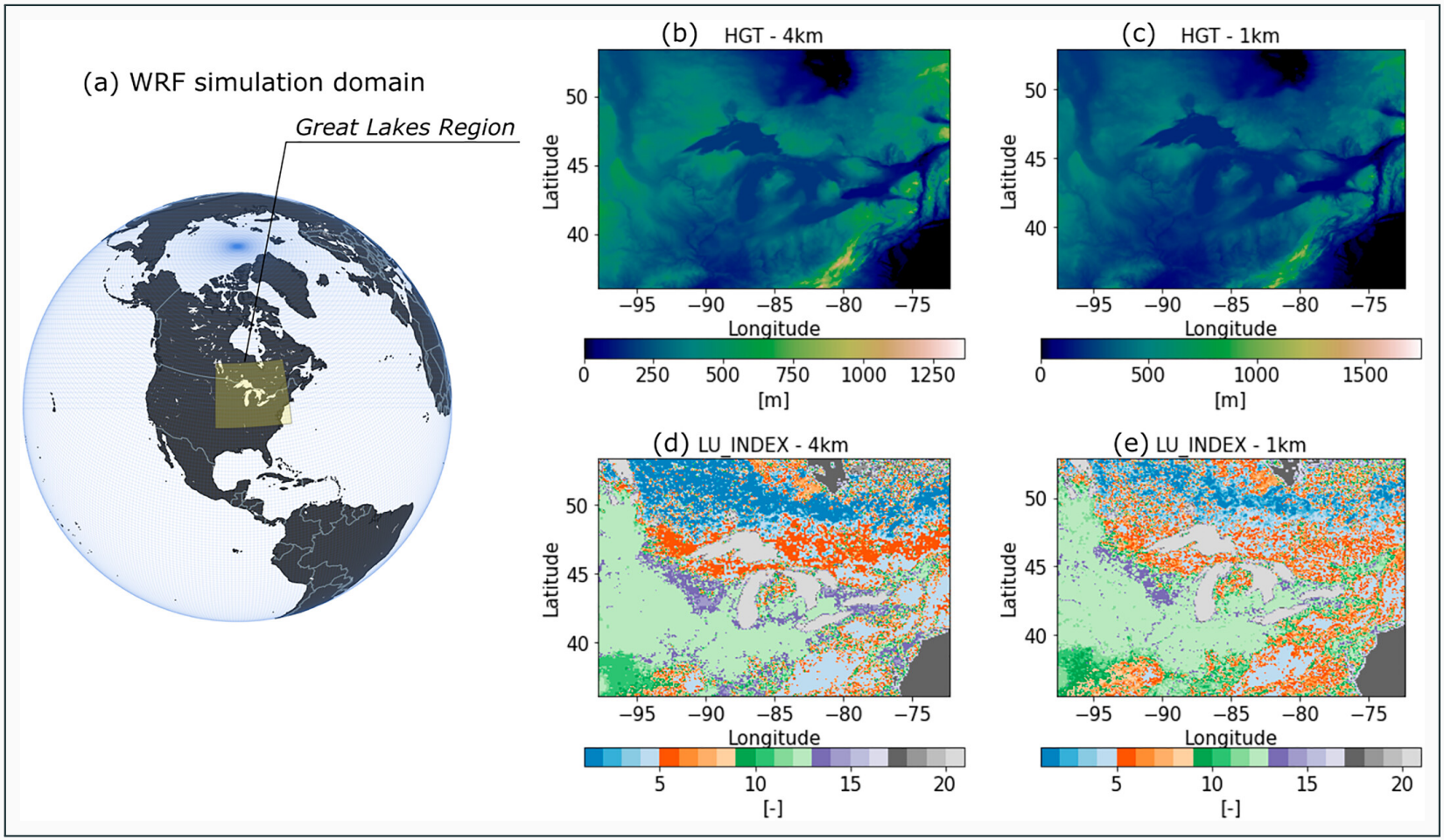
- [Li et al. (2020)] used neural operators to solve high resolution Navier-Stokes equations from low resolution training data
- Neural operator models runs 1000x times faster with nearly free zero-shot super-resolution

Operator learning



- [Pathak et al. (2022)] trained a neural operator model to forecast global weather patterns (temperature, humidity, wind velocity, etc)
- Matches the skill of ECMWF Integrated Forecasting System (IFS) at a fraction of the cost

Operator learning



- [Jiang et al. (2023)] used neural operators for high resolution downscaling of regional climate model output
- Matches the skill of dynamically downscaled (WRF) output

Operator learning

- Functional data analysis - Inference and regression
  - Linear models - [Ramsay and Dalzell (1991); Besse and Cardot (1996); Yao et al. (2005); Wu and Müller (2011); Ivanescu et al. (2015)], etc.
  - Nonlinear models - [Yao and Müller (2010); Giraldo et al. (2010); Ferraty et al. (2011); McLean et al. (2014); Scheipl et al. (2015)], etc.
- Neural operators models - Deep predictive models
  - Neural Operator Models - [Chen and Chen (1995); Lu et al. (2021); Bhattacharya et al. (2021); Nelsen and Stuart (2021)], etc.
  - Computationally efficient: Fourier Neural operators [Li et al. (2020)], Spherical Operators [Bonev et al. (2023)], Wavelet Operators [Tripura and Chakraborty (2023)], and others [Lanthaler et al. (2023a); Kovachki et al. (2024)]
- Neural operators lack uncertainty quantification (UQ). UQ critical for many applications (e.x. weather forecasting).

  $\Rightarrow$ Develop UQ for general operator models

Uncertainty quantification

- What do we want to achieve with UQ?
- Many approaches to UQ. We take a prediction *set* approach.
  - $\Gamma_\theta(f)$ returns a single function $\hat{g}$
  - Lift $\Gamma_\theta(f)$ to return a *set of functions* $C_\alpha(f)$
- Properties of the UQ procedure
  1. **Valid:** Resulting $C_\alpha(f)$ should cover g with high probability, i.e. $(1-\alpha)$

$$P(g \in C_\alpha(f)) \geq 1 - \alpha$$

  2. **Minimal:** $C_\alpha(f)$ should be as small as possible without violating validity

$$P(g \in C_\alpha(f)) < 1 - \alpha + \epsilon$$

  3. **Adaptive:** Responsive to aleatoric, epistemic uncertainty, and shape
     - $C_\alpha(f)$ should expand or contract to reflect different levels of certainty
     - Elements of $C_\alpha(f)$ should have the same shape as g_{n+1}
  4. **General:** should work *regardless of the underlying prediction algorithm* Γ_θ.

(Split) Conformal inference

- General framework for quantifying predictive uncertainty [Vovk et al. (2005); Lei et al. (2018)]
  - No asymptotic assumptions, prior elicitation, or modification to the training procedure. Computationally efficient.
  - Default assumption: data $\{(f_t, g_t)\}_{t=1}^n$ are exchangeable with (f_{n+1}, g_{n+1})
- Basic algorithm: given training data $\mathcal{D}_{tr} = \{(f_t, g_t)\}_{t=1}^n$, calibration data $\mathcal{D}_{cal} = \{(f_s, g_s)\}_{s=1}^m$, and confidence level $\alpha \in (0, 1)$ and test input f_{n+1}
  1. Train on $\mathcal{D}_{tr}$: $\hat{\Gamma}_\theta = \arg\min_\theta \mathcal{L}(\Gamma_\theta, (f, g))$
  2. Score residuals on $\mathcal{D}_{cal}$: $|r_s| = \mid g_s - \hat{\Gamma}_\theta(f_s) \mid \quad \forall s \in 1, ...m$
  3. Estimate dispersion on $\mathcal{D}_{cal}$ as quantile of the residual score distribution

$$\begin{aligned} k_m &= \text{Quantile}(|r_s|, \lceil (1-\alpha)m + 1 \rceil / m) \\ &= \inf\{k \in \mathbb{R} : \lceil (1-\alpha)m + 1 \rceil / m \leq \hat{G}(k)\} \end{aligned}$$

  4. Construct prediction set on f_{n+1}

$$C_\alpha(f_{n+1}) = \hat{\Gamma}_\theta(f_{n+1}) \pm k_m$$

Adaptive Conformal inference

- Standard split conformal inference works well marginally
  - $P(g \in C_\alpha(f)) \geq 1 - \alpha$ even in *finite samples* (valid)
  - $P(g \in C_\alpha(f)) \leq 1 - \alpha + O(m)$ (minimal)
  - Works for any prediction algorithm $\hat{\Gamma}_\theta$ (general)
- But it is **non-adaptive** to any heterogeneity in $g \mid f$.
  - Size and shape of prediction sets fixed ($\pm k_m$) at calibration time
  - $C_\alpha(f_{n+1}) = \hat{\Gamma}_\theta(f_{n+1}) \pm k_m$, $C_\alpha(f_{n+2}) = \hat{\Gamma}_\theta(f_{n+2}) \pm k_m$, etc.

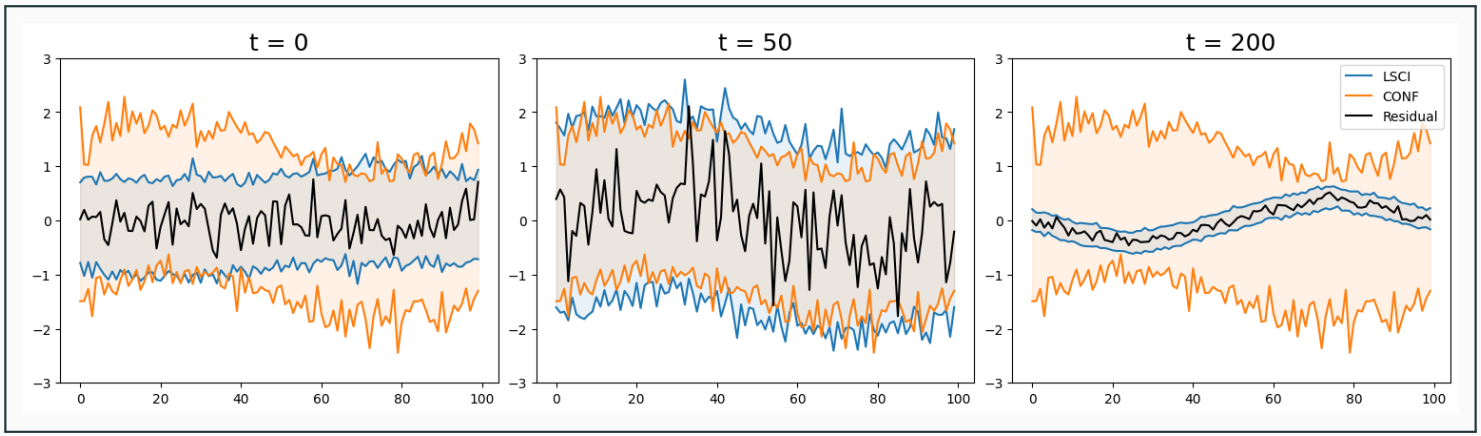


Figure 1: E.x. heterogenous residual functions. Not captured by standard CI interval

Adaptive Conformal inference

- Adaptive Scoring Rules - Weight calibration residuals by the predicted variance [Lei et al. (2015, 2018); Diquigiovanni et al. (2022)]

$$C_\alpha(f_{n+1}) = \hat{\Gamma}_\theta(f_{n+1}) \pm \hat{\sigma}_\phi(f_{n+1}) k_m$$

- Conformalized Quantile regression - Adjust quantile regression to have conformal guarantee [Romano et al. (2019); Angelopoulos et al. (2022); Ma et al. (2024)]

$$C_\alpha(f_{n+1}) = (\lambda \hat{\Gamma}_\theta^{\alpha/2}(f_{n+1}), \lambda \hat{\Gamma}_\phi^{1-\alpha/2}(f_{n+1}))$$

- Local Conformal Inference - Weight the residual quantile function by similarity between f_{n+1} and f_t [Guan (2023); Hore and Barber (2023)]

$$C_\alpha(f_{n+1}) = \hat{\Gamma}_\theta(f_{n+1}) \pm k_m(f_{n+1})$$

- Local conformal inference assumes that the conditional distribution $g \mid f = f_t$ evolves *smoothly* over f

Local Conformal Inference

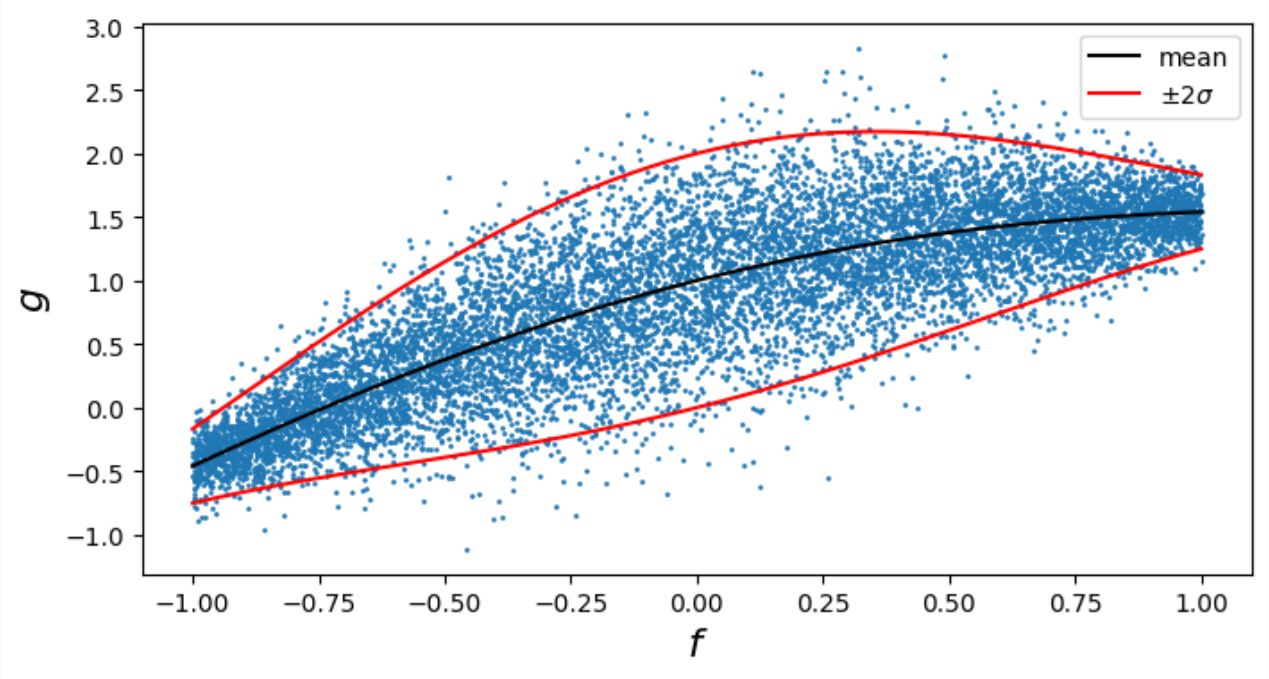


Figure 2: 1D example of local smoothness / local exchangeability

- Idea: place more weight on data near f_{n+1} when estimating intervals
- [Guan (2023)] showed strong performance of local methods in univariate problems $\Rightarrow$ We take a local approach for *operator* problems.

Local Spectral Conformal Inference

- Suppose we have n training samples $\{(f_t, g_t)\}_{t=1}^{n}$ and m calibration samples $\{(f_s, g_s)\}_{s=1}^{m}$ and a test function f_{n+1}.
- Replace global k_m from global residual score distribution $\hat{G}$ with a local k_m from the local distribution residual score $\hat{G}_{n+1}$

$$\hat{G} = m^{-1} \sum_{s=1}^{m} \delta(|r_s|) \quad \Rightarrow \quad \hat{G}_{n+1} = \sum_{s=1}^{m} \eta_s \delta(|r_s|) \tag{2}$$

- $\delta(\cdot)$ is the indicator function and η_s is the localization weight.
- η_s increases as $d(f_s, f_{n+1})$ decreases.
  - Covariate localization: $d(f_s, f_{n+1}) = ||f_s - f_{n+1}||_2$ or $d(f_s, f_{n+1}) = \max_{x \in D} |f_s(x) - f_{n+1}(x)|$, etc.
  - Semantic localization: $d(f_s, f_{n+1}) = ||\Gamma_\theta^L(f_s) - \Gamma_\theta^L(f_{n+1})||_2$ etc.
- Can't estimate G_{n+1} directly on functional data. Project to eigenbasis.

Spectral Projection

- Karhunen-Loeve (KL) decomposition of any $h_t \in \mathcal{H} = \mathcal{L}^2(D)$

$$h_t = \mu_t + \sum_{k=1}^{\infty} \xi_{k,t} \phi_k \tag{3}$$

  - Each $\phi_k \in \mathcal{H}$ is an eigenfunction of the covariance kernel defining the stochastic process that generated h_t (eigenbasis)
  - Spectral coefficients: $\xi_{k,t} = \int_D \phi_k(x) h_t(x) dx$
- Spectral representation of h_t : $\xi_t = \{\xi_{k,t}\}_{k=1}^{\infty}$ (functional principal components - FPCs)
- Express as a (invertible) linear operator: $\Phi : \mathcal{H} \mapsto \mathcal{H}'$

$$\xi_t = \Phi(h_t), \quad h_t = \Phi^{-1}(\xi_t), \tag{4}$$

- In practice: finite n_c eigenfunctions and matrix multiplication

$$h_t = \mu_t + \sum_{k=1}^{n_c} \xi_{k,t} \phi_k \quad \text{and} \quad \xi_t = \Phi h_t, \quad h_t = \Phi^{-1} \xi_t \tag{5}$$

Spectral Localization

- **Goal:** Approximate the distribution of the unknown residual $r_{n+1} = g_{n+1} - \hat{\Gamma}_\theta(f_{n+1})$ from the calibration residuals.
- Instead work with $\xi_{n+1} = \Phi(r_{n+1})$. Estimate the eigenbasis Φ on the calibration residuals and project.

$$\begin{array}{ccc} r_1 = g_1 - \hat{\Gamma}_\theta(f_1) & & \xi_1 = \Phi(r_1) \\ r_2 = g_2 - \hat{\Gamma}_\theta(f_2) & & \xi_2 = \Phi(r_2) \\ \vdots & \xrightarrow{\Phi} & \vdots \\ r_m = g_m - \hat{\Gamma}_\theta(f_m) & & \xi_m = \Phi(r_m) \end{array}$$

- Locally approximate the marginals of $\xi_{n+1} = (\xi_{k,n+1})_{k=1}^{\infty}$ as

$$G_{k,n+1} \approx \hat{G}_{k,n+1} = \sum_{s=1}^{m} \eta_{k,s} \delta(\xi_{k,s}) \quad \forall k = 1, 2, ...$$

- Localization weight $\eta_{k,s} \uparrow$ as $d(f_s, f_{n+1}) = \max_{x \in D} |f_s(x) - f_{n+1}(x)| \downarrow$.
- Marginals not sufficient to estimate joint, but it can get us level sets*

Spectral Conformity

- Depth functions quantify centrality wrt a reference distribution [Nagy et al. (2016)]

- Integrated Spectral Tukey (IST) depth

$$d(\xi_s \mid \hat{G}_{n+1}) = 2\mathbb{E}_{k\in\mathbb{N}} \left|1 - 2\hat{G}_{k,n+1}(\xi_{k,s})\right|$$

$$\approx \frac{2}{n_c} \sum_{k=1}^{n_c} \left|1 - 2\hat{G}_{k,n+1}(\xi_{k,s})\right|$$

- $d(\xi \mid \hat{G}_{n+1})$ varies continuously 0 to 1
  - $d(\xi \mid \hat{G}_{n+1}) = 1$ is the median of $\hat{G}_{n+1}$
  - $d(\xi \mid \hat{G}_{n+1}) = 0$ is infinitely outlying

- $d(\xi \mid \hat{G}_{n+1})$ based on marginals. Represents joint by assuming $\hat{G}_{n+1}$ is convex and unimodal.

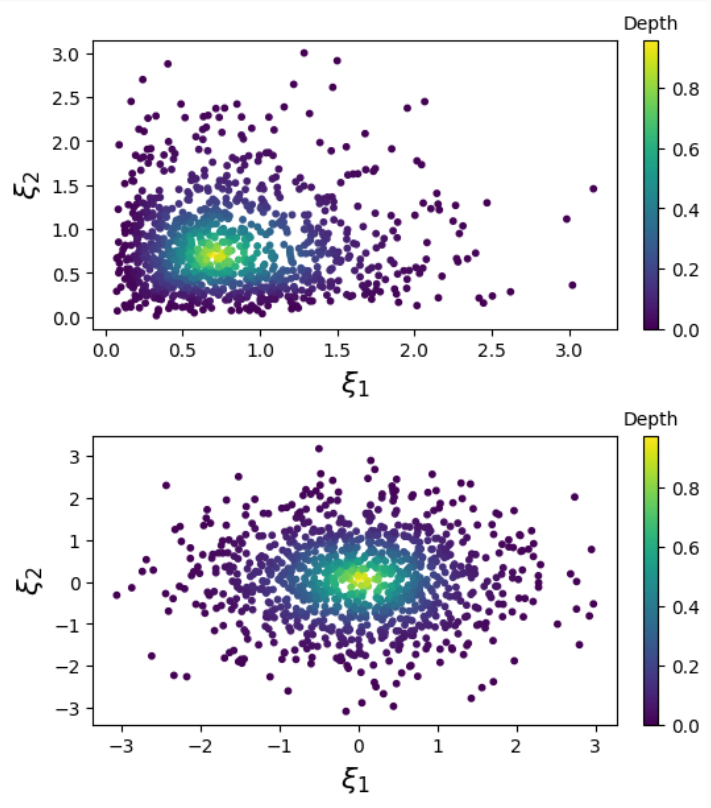


Figure 3: 2D examples of depth

Spectral Conformity

- Depth defines **central regions**

$$D(\xi \mid \beta) = \{\xi : d(\xi \mid \hat{G}_{n+1}) \geq \beta\}$$

- Grows monotonically outward from the median of G_{n+1} as $\beta \to 0$
- If $\beta_1 < \beta_2$ then $D(\xi \mid \beta_2) \subset D(\xi \mid \beta_1)$
- Characterizes the location, scale, and shape of G_{n+1}
- For all coverage levels $\alpha \in (0, 1)$ $\exists \beta$ st

$$P(\xi \in D(\xi \mid \beta)) \geq 1 - \alpha$$

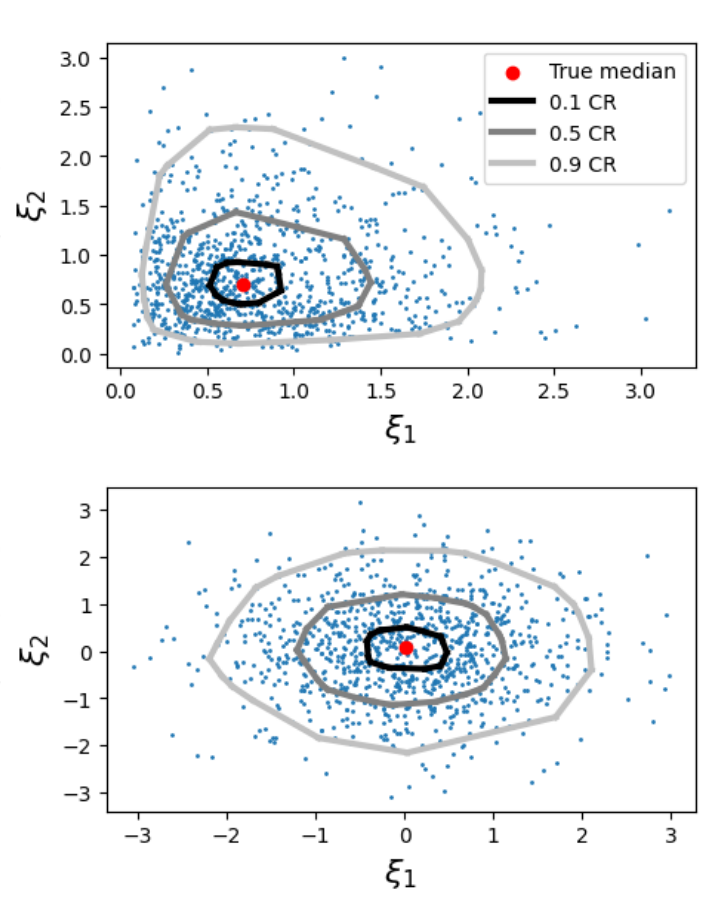


Figure 4: 2D examples of depth based central regions

Prediction Sets

- Denote $D_\alpha(\xi)$ as the smallest central region $D(\xi \mid \beta)$ such that

$$P(\xi' \in D(\xi \mid \beta)) \geq 1 - \alpha \tag{6}$$

- $D_\alpha(\xi)$ is sufficient to define a prediction set.
  - $D_\alpha(\xi)$ contains (e.x.) 90% most typical residual spectra under $\hat{G}_{n+1}$
  - $\Phi^{-1}[D_\alpha(\xi)]$ contains the 90% most typical looking residuals
  - Translate by $\hat{\Gamma}_\theta(f_{n+1})$ to get 90% most typical looking predictions

- Define our prediction set $C_\alpha(f_{n+1})$ as

$$C_\alpha(f_{n+1}) = \{\Gamma_\theta(f_{n+1}) + \Phi^{-1}(\xi') : \xi' \in D_\alpha(\xi)\} \tag{7}$$

- In theory requires mapping all (infinite) $\xi' \in D_\alpha(\xi)$ back through Φ^{-1}. Instead use random sampling.

Inverse sampling

- Random sampling to approximate $C_\alpha(f_{n+1})$
  1. Sample spectral coefficients with inverse local CDFs

$$\tilde{\xi}_{n+1} = \{\tilde{\xi}_{k,n+1}\}_{k=1}^{\infty} = \{\hat{G}_{k,n+1}^{-1}(u_k)\}_{k=1}^{\infty} \quad u_k \sim U(0,1) \tag{8}$$

  2. Accept if $\tilde{\xi}_{n+1} \in D_\alpha(\xi)$.
  3. Map through $\Phi^{-1}(\cdot)$

$$\tilde{r}_{n+1} = \Phi^{-1}(\tilde{\xi}_{n+1}) = \sum_{k=1}^{\infty} \tilde{\xi}_{k,n+1}\phi_k \tag{9}$$

  4. Repeat.

- Large ensemble of residuals $\tilde{r}_{n+1}^1, ..., \tilde{r}_{n+1}^m$
  - Prediction set: $C_\alpha(f_{n+1}) = \{\Gamma_\theta(f_{n+1}) + \tilde{r}_{n+1}^i : i \in 1, ..., m\}$
  - Prediction band: (L_{n+1}, U_{n+1}) – pointwise min and max of the $C_\alpha(f_{n+1})$.

Local Spectral Conformal Inference

Input: Trained operator model Γ_θ, calibration data $\{(f_s, g_s)\}_{t=1}^m$, test point f_{n+1}, and confidence level $\alpha \in (0,1)$

1. Estimate the spectral operator Φ from the calibration residuals $r_t = g_s - \Gamma_\theta(f_s)\ \forall s \in 1, ..., m$ and map to the spectral domain

$$\xi_s = \Phi(r_s) \quad \forall s \in 1, ..., m$$

2. Estimate the local marginal distributions of $\xi_{n+1} = \Phi(r_{n+1})$

$$G_{k,n+1} \approx \hat{G}_{k,n+1} = \sum_{s=1}^{m} \eta_{k,s}\delta(\xi_{k,s}) \quad \forall k = 1, 2, ...$$

3. Compute the adjusted $1-\alpha$ central region of $\hat{G}_{n+1}$

$$D_\alpha(\xi_{n+1}) = \{\xi : \hat{G}_{n+1}(\xi) \leq \lceil (1-\alpha)(m+1) \rceil / m\}$$

4. Map back to real space and translate

$$C_\alpha(f_{n+1}) = \{\Gamma_\theta(f_{n+1}) + \Phi^{-1}(\xi') : \xi' \in D_\alpha(\xi_{n+1})\} \tag{10}$$

Return: $C_\alpha(f_{n+1})$

Sparse weighting

- Recall: localization weights $\eta_{k,t}$ in the local CDF based on $d(f_s, f_{n+1})$
  - Uses all of the calibration data $f_1, f_2, ..., f_m$
  - Prediction sets vary smoothly over $f_{n+1}, f_{n+2}, ...$
  - Not responsive to rapid changes

- Alternative: define the un-normalized weights as

$$\eta_{k,t} = \max\{0, d_k - d(f_t, f_{n+1})\}, \tag{11}$$

  where d_k is the k'th largest distance among $d(f_1, f_{n+1}), d(f_2, f_{n+1}), ...$

- Diff with original weighting scheme
  - \+ k-nearest neighbors to estimate the local CDF.
  - \+ Apply tree-based methods to reduce lookup time
  - \+ Hard thresholds most weights to zero
  - − *Only* uses the k-nearest neighbors

Numerical experiments

1. **1D Gaussian process**: 1D GPs, Matèrn covariance, time varying mean and error variance. $n_{tr} = n_{cal} = 1000$, $p = 100$. One step ahead forecast.

2. **2D Spherical Gaussian process**: 2D GPs on a sphere, Matèrn covariance, time varying mean and error variance. $n_{tr} = n_{cal} = 200$, $p = (30, 60)$. One step ahead forecast.

3. **Energy demand forecasting**: Energy demand curve forecasting (ERCOT). $n_{tr} = 1500$, $n_{cal} = 2500$, $p = 24$. One step ahead forecast.

4. **Temperature forecasting**: Daily ERA5 2m surface temperature fields $n_{tr} = 3650$, $n_{cal} = 1825$, $p = (32, 64)$. One step ahead forecast.

Methods

- For all problems we train a simple Residual Averaging Neural Operator (ANO) [Lanthaler et al. (2023b)]
  - Four ANO layers with width 100
  - Skip connection from input to output
  - Train with Adam (lr = 1e-3) for 50 epochs on NVIDIA V100.

- Baseline methods
  - Oracle - True prediction residual
  - Conf - Marginal conformal for functional data [Lei et al. (2015)]
  - Supr - Marginal conformal with data depth [Diquigiovanni et al. (2022)]
  - UQNO - Operator / functional variant of CQR [Ma et al. (2024)]

- LSCI - using $k = 0.1m$ and $k = 0.05m$.

Metrics

- Metrics
  1. **Risk** - % of targets that are, at least, $\delta \times 100\%$ contained in the prediction set. Set $\delta = 0.01$. Should be close to $1 - \alpha$.
  2. **Width** - Distance between upper and lower bound of prediction set. Computed as mean pointwise difference.
  3. **Risk Correlation** (RC) - Correlation between risk and error variance. Should be close to 0.
  4. **Width Correlation** (WC) - Correlation between width and error variance Should be close to 1.

- Risk and width measure the marginal behavior of the prediction sets over the test data.

- RC and WC measure how adaptive the prediction sets are.

1D GP on $[0, 1]$

	Risk ($\uparrow$)	RC ($\rightarrow 0$)	Width ($\downarrow$)	WC ($\uparrow$)
Oracle	1.000	0.000	0.613	0.984
Conf.	0.852	-0.568	1.285	0.000
Supr.	0.888	-0.505	1.422	0.000
UQNO	0.968	-0.297	1.602	0.182
$\text{LSCI}_{0.1}$	0.924	-0.013	0.912	0.928
$\text{LSCI}_{0.05}$	0.896	-0.035	0.876	0.926

Table 1: Uncertainty metrics on synthetic GP sims

- LSCI has near perfect RC and high WC (high adaptivity)
- Valid risk control (≥ 0.9) and smallest width

1D GP on $[0, 1]$

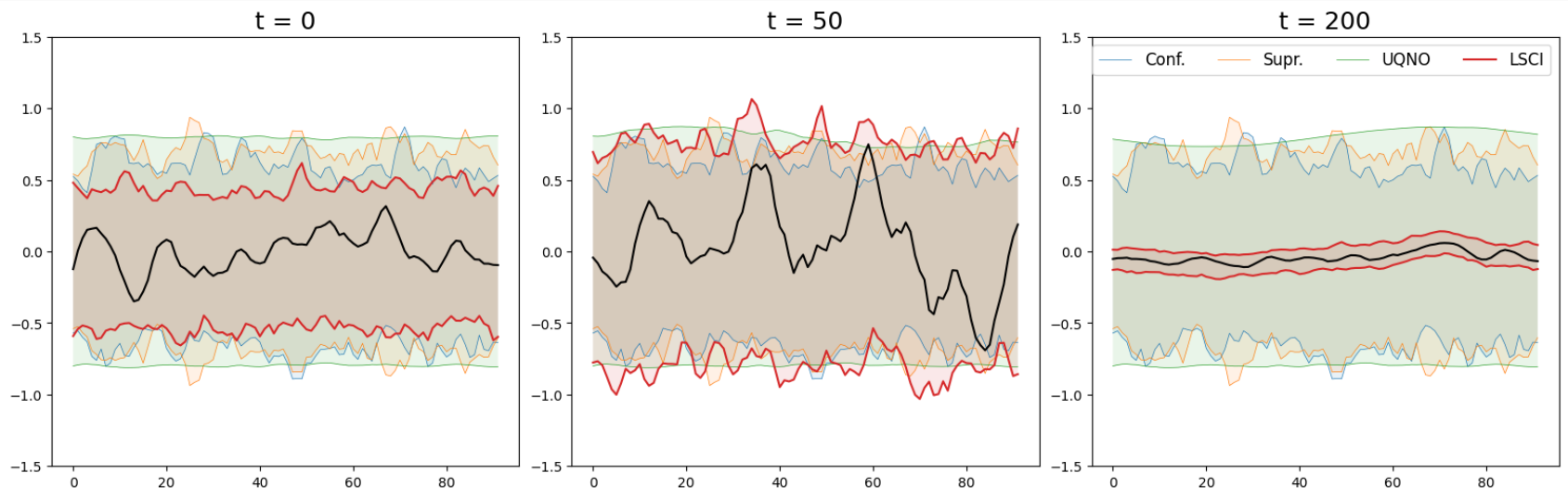


Figure 5: Example prediction sets using each of the baseline methods and the proposed method (LSCI).

- LSCI prediction sets match the target variability
- *note: optimal conditions for LSCI*

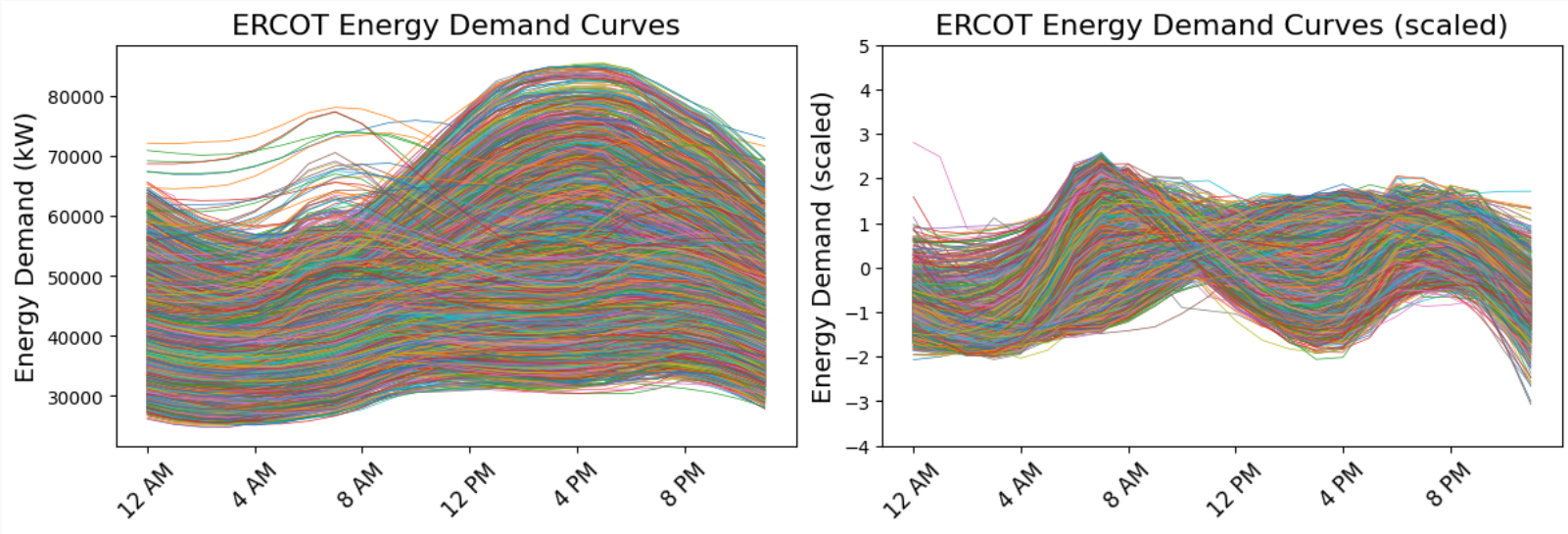
2D GP on $\mathbb{S}^2$

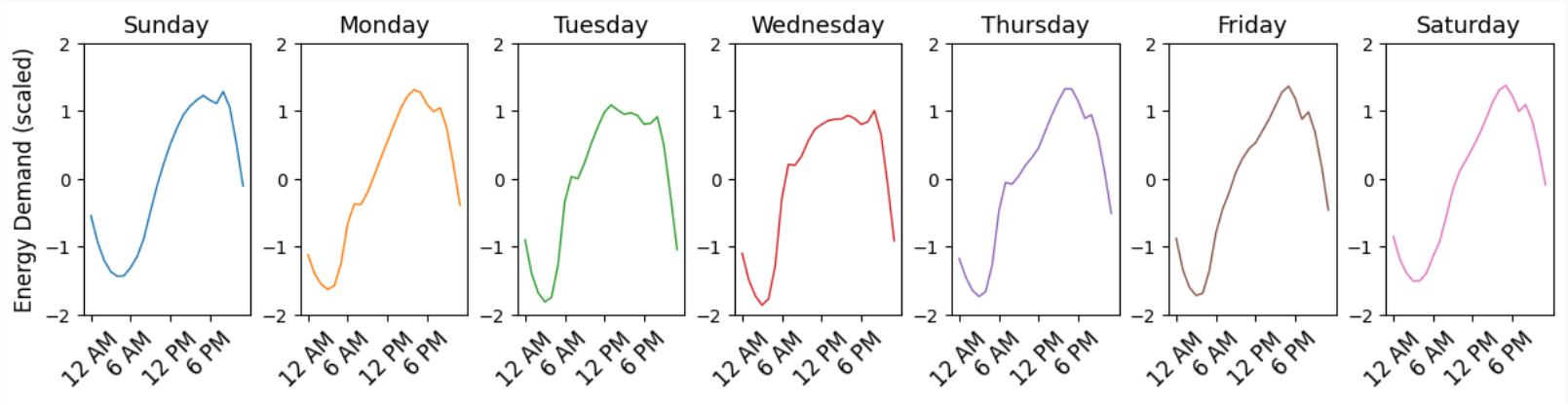
	Risk ($\uparrow$)	RC ($\rightarrow 0$)	Width ($\downarrow$)	WC ($\uparrow$)
Oracle	1.000	0.000	0.730	0.979
Conf.	0.727	-0.748	1.003	0.000
Supr.	0.932	-0.531	1.224	0.000
UQNO	0.709	-0.774	0.867	0.968
$\text{LSCI}_{0.1}$	0.934	-0.140	0.794	0.967
$\text{LSCI}_{0.05}$	0.884	-0.152	0.780	0.966

Table 2: Uncertainty metrics on synthetic gp2d (spherical) sims.

- 1D Euclidean v.s. 2D Spherical doenst change LSCI behavior
- UQNO no longer controls risk

Energy Demand - ERCOT





- Predict tomorrow's demand curve given today's demand curve

Energy Demand - ERCOT

	Risk ($\uparrow$)	RC ($\rightarrow 0$)	Width ($\downarrow$)	WC ($\uparrow$)
Oracle	0.974	0.000	1.085	0.979
Conf.	0.965	-0.549	3.789	0.000
Supr.	0.912	-0.709	3.013	0.000
UQNO	0.910	-0.562	4.279	0.596
$\text{LSCI}_{0.1}$	0.932	-0.236	1.856	0.725
$\text{LSCI}_{0.05}$	0.921	-0.260	1.690	0.733

Table 3: Uncertainty metrics on synthetic gaussian process data.

- LSCI prediction sets expand and contract with increasing prediction variance (high RC, WC)
- Valid risk control (≥ 0.9) and smallest width

Energy Demand - ERCOT

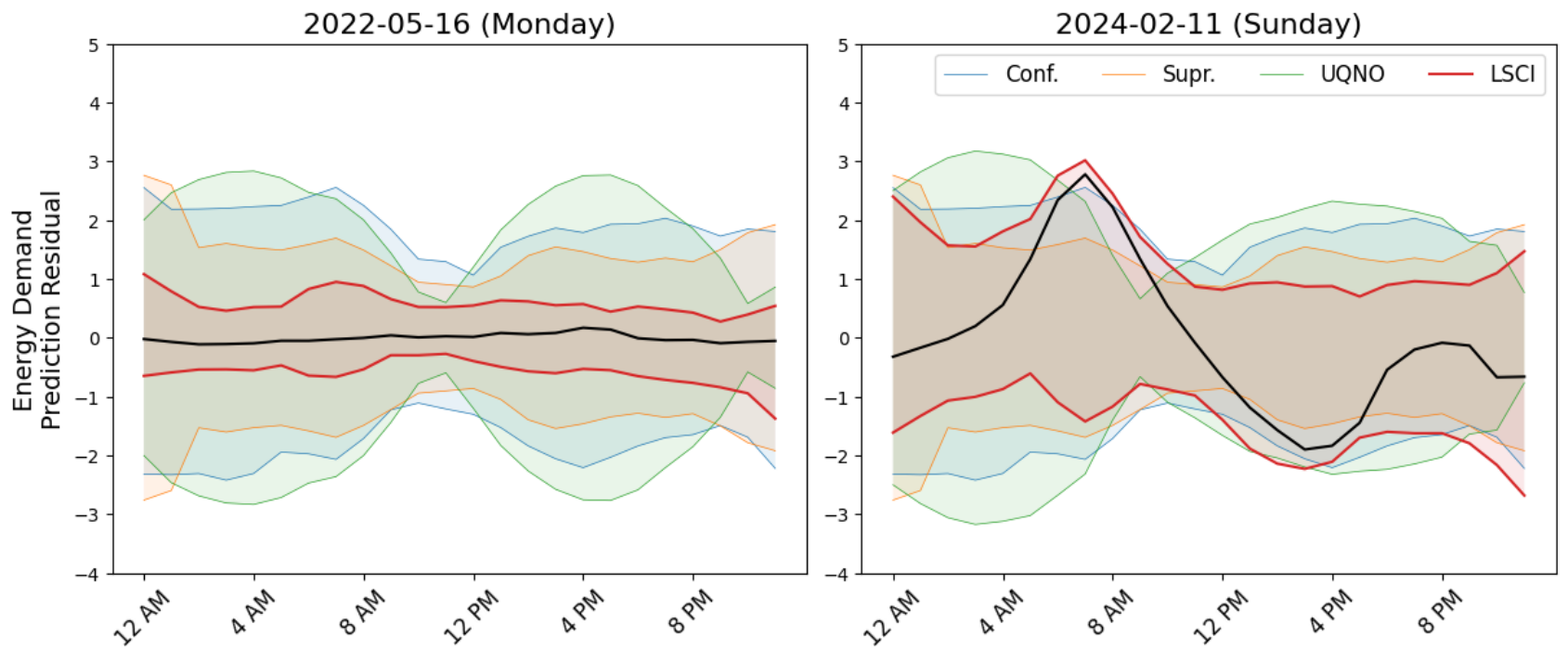


Figure 6: Example prediction sets using each of the baseline methods and the proposed method (LSCI).

- LSCI prediction sets adapt to the shape and variability of each target
- Conf and Supr do not adapt, UQNO weakly adapts

Weather forecasting

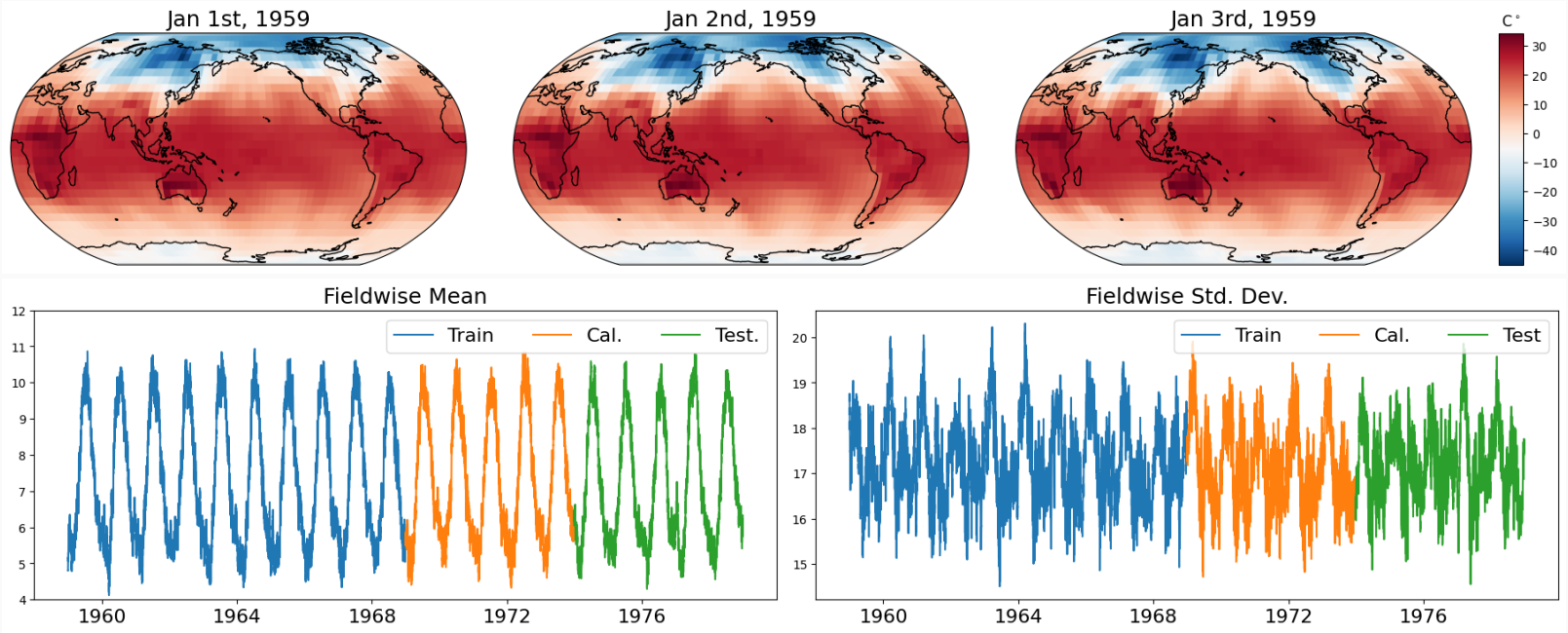


Figure 7: Weather forecasting. Predict daily average temperature field. Data is non-stationary.

- One day ahead weather forecasting (AR(1))
- Train on 10 years of data, calibrate on 5, test on 5. (further data introduces covariate shift)
- Data is highly non-stationary

Weather forecasting

	Risk (↑)	RC ($\rightarrow 0$)	Width (↓)	WC (↑)
Oracle	1.000	0.000	1.190	0.498
Conf.	0.995	-0.511	11.224	0.000
Supr.	0.995	-0.481	12.235	0.000
UQNO	1.000	-0.570	47.695	0.162
$\text{LSCI}_{0.1}$	0.984	-0.521	9.947	0.247
$\text{LSCI}_{0.05}$	0.984	-0.533	9.928	0.239

Table 4: Uncertainty metrics on synthetic gaussian process data.

- No methods seems to perform well relative to the oracle, although all control risk and have acceptable width (except UQNO).
- LSCI exhibits low WC, possibly leading to high negative RC

Discussion

- Motivation
  - Operator models are an important development in ML
  - Neural operators lack inherent UQ
  - UQ is critical to many applications
- Method
  - Local Spectral Conformal inference
  - Transform to spectral domain, localize each spectra *marginally*
  - Use depth to estimate *joint* spectral prediction sets
  - Sample to get approximate prediction sets in real space
- Results
  - Good risk control and width
  - Significantly more adaptive (size and shape) than baselines
  - Finalize theoretical and empirical results
- Future work
  - Improve computational efficiency!