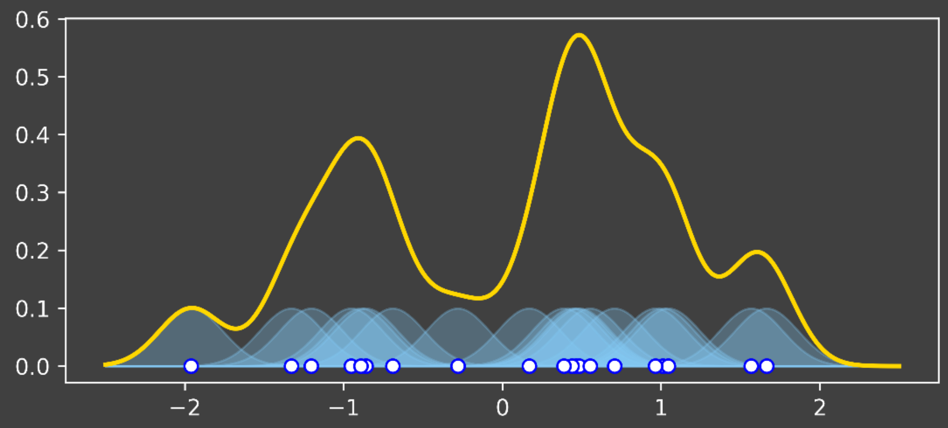
Efficient Interpolation of Density Estimators

Paxton Turner (MIT)
Jingbo Liu (UIUC)
Philippe Rigollet (MIT)

Motivation: Fast evaluation of KDE

- Kernel density estimator (KDE):

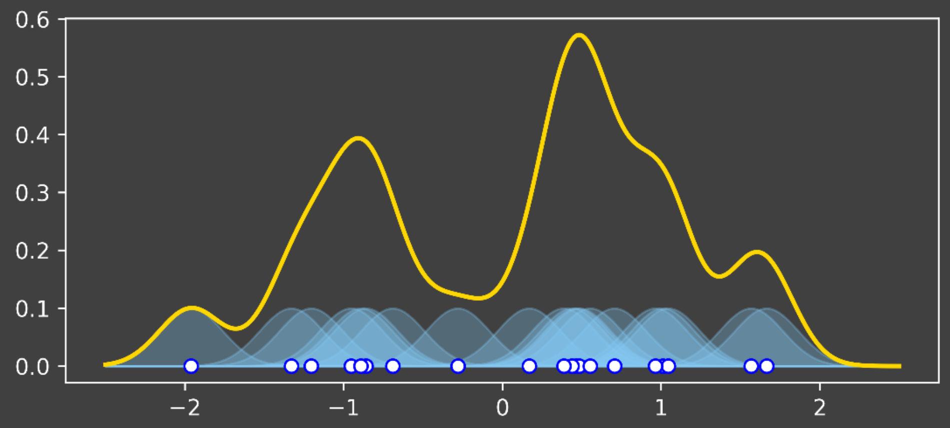
$$\hat{f}(y) = \frac{1}{n}\sum_{j=1}^{n} K_h\left(X_j - y\right)$$



Motivation: Fast evaluation of KDE

- Kernel density estimator (KDE):

$$\hat{f}(y) = \frac{1}{n}\sum_{j=1}^{n} K_h\left(X_j - y\right)$$

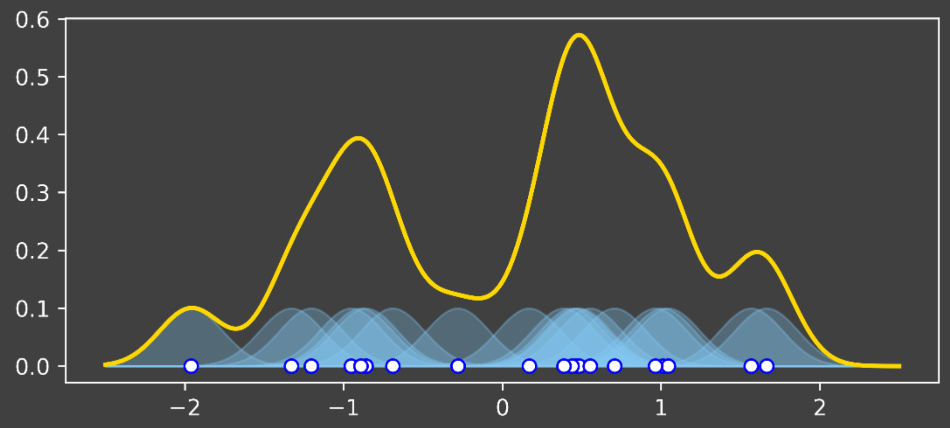


- Statistically accurate, computationally slow

Motivation: Fast evaluation of KDE

- Kernel density estimator (KDE):

$$\hat{f}(y) = \frac{1}{n} \sum_{j=1}^{n} K_h\left(X_j - y\right)$$



- Statistically accurate, computationally slow
- Many known speed-ups:
  - Fast Gauss transform (Greengard-Strain), locality sensitive hashing (Charikar-Siminelakis, Backurs et al), coresets (Phillips-Tai, Karnin-Liberty), binning (Scott-Sheather), interpolation (Jones, Kogure)

Can we speed up accurate estimators ?

Problem

Given: accurate estimator $\hat{f}$ for unknown smooth f ← Hölder smooth of order β

Problem

Given: accurate estimator $\hat{f}$ for unknown smooth f ← Hölder smooth of order β

Goal: Convert $\hat{f}$ to $\hat{g}$ satisfying

1. (Accurate) $\hat{g}$ is a good estimator for f
2. (Low-space) $\hat{g}$ can be stored efficiently
3. (Fast) $\hat{g}$ can be queried efficiently

Our Approach

Problem

Given: good estimator $\hat{f}$ for smooth f

Goal: Convert $\hat{f}$ to $\hat{g}$ that is accurate, low-space, and fast

- **Fact:** Hölder β functions $\approx$ degree β piecewise polynomials

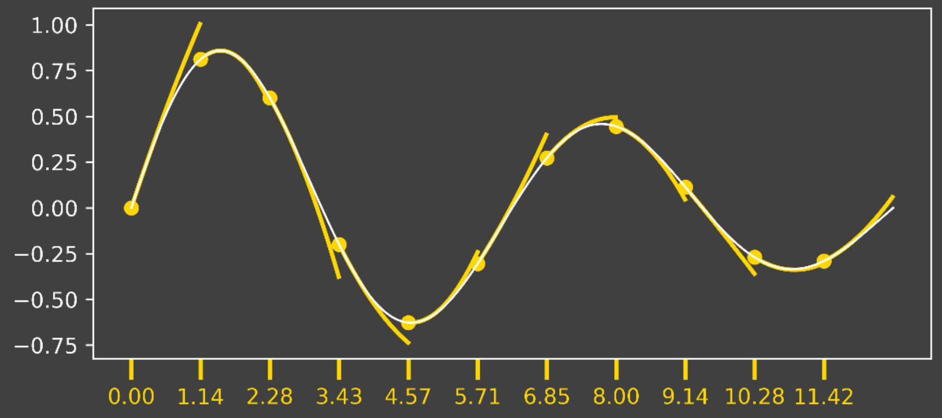
Our Approach

Problem

Given: good estimator $\hat{f}$ for smooth f

Goal: Convert $\hat{f}$ to $\hat{g}$ that is accurate, low-space, and fast

- **Fact:** Hölder β functions $\approx$ degree β piecewise polynomials



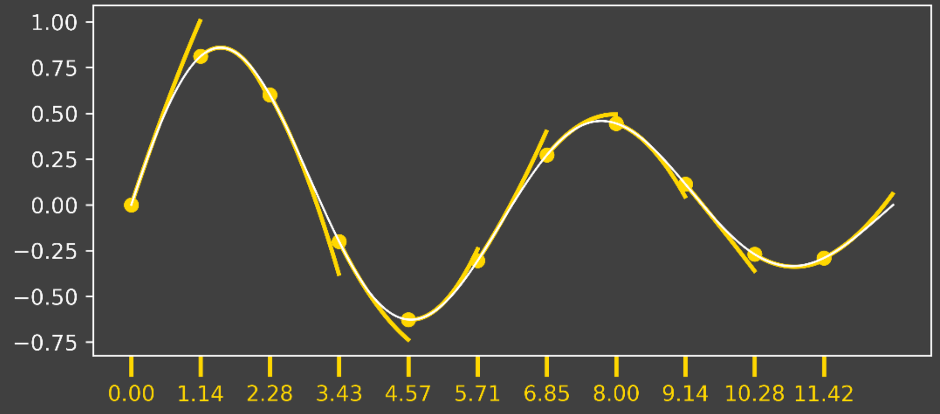
Our Approach

Problem

Given: good estimator $\hat{f}$ for smooth f

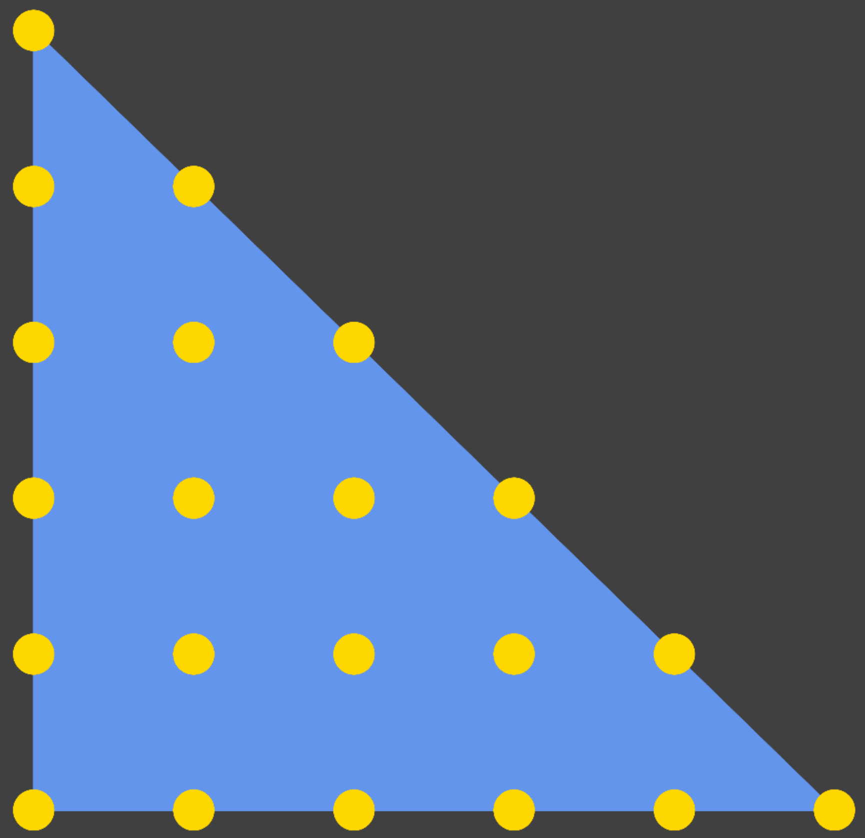
Goal: Convert $\hat{f}$ to $\hat{g}$ that is accurate, low-space, and fast

- **Fact:** Hölder β functions $\approx$ degree β piecewise polynomials



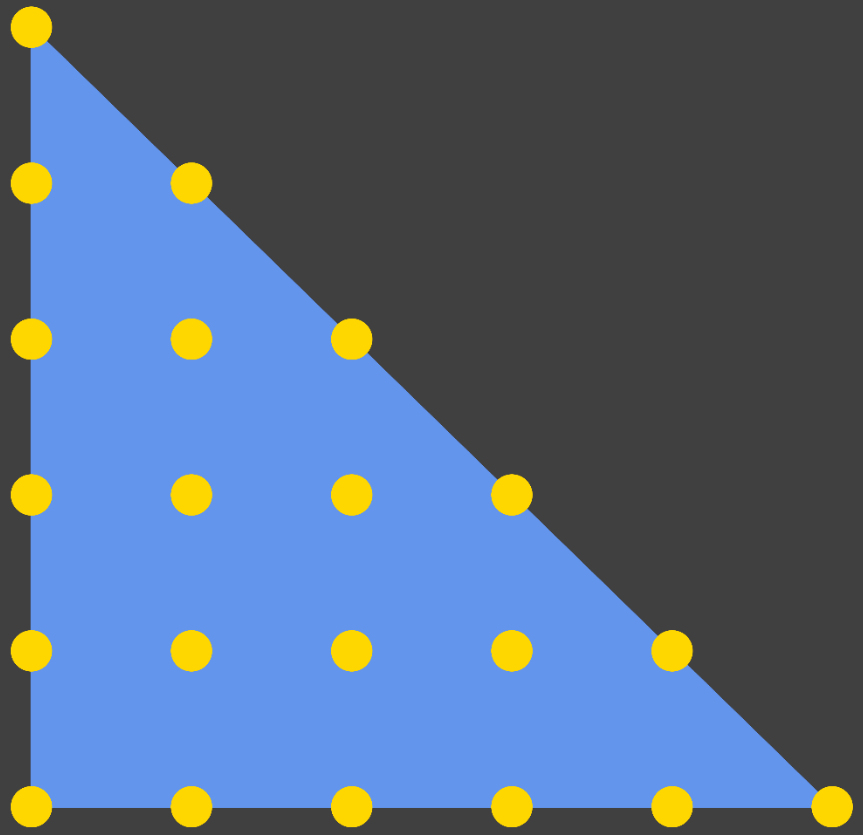
- **Strategy:** recover these polynomials from $\hat{f}$

Principal Lattice Interpolation



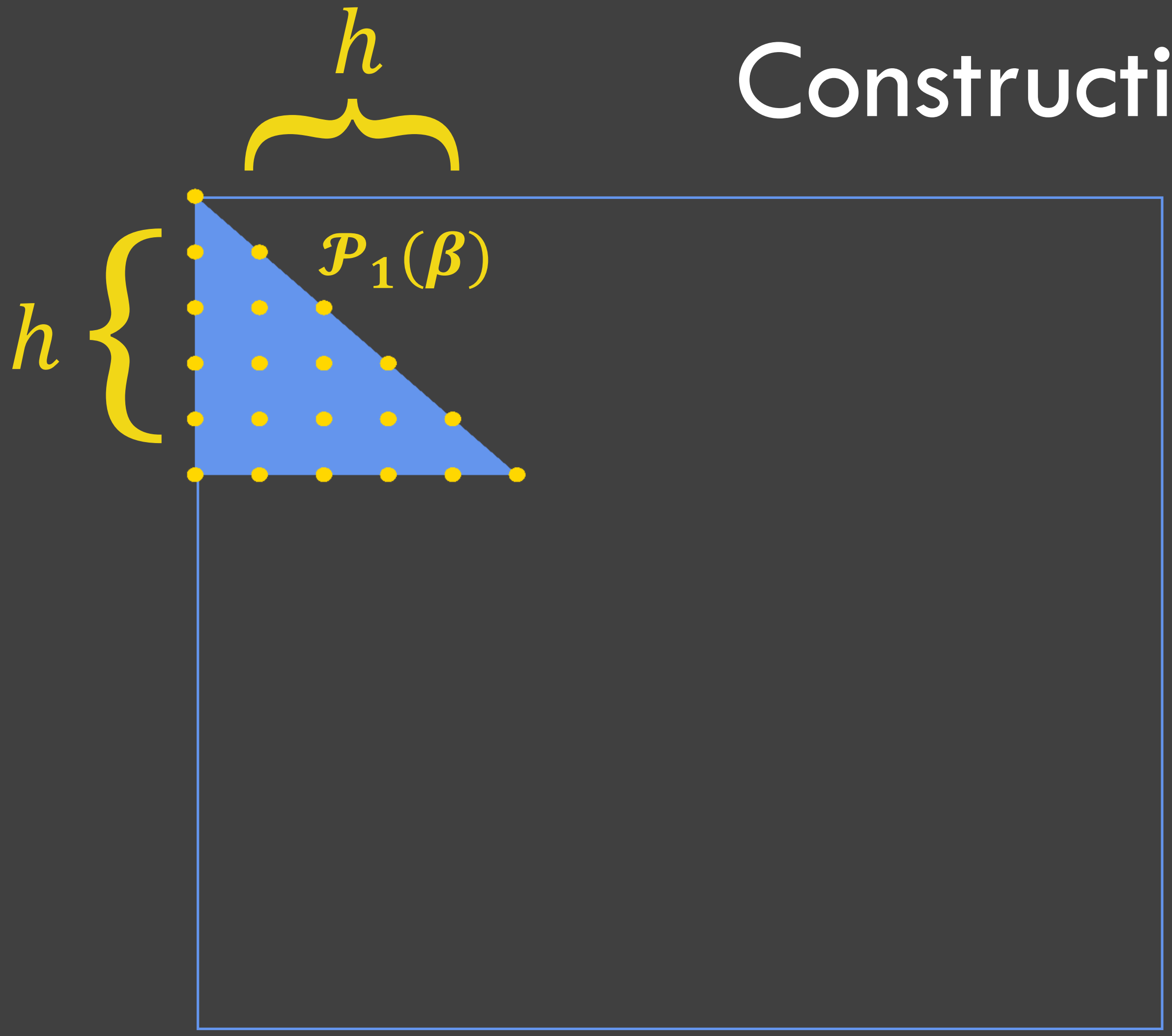
- $\mathcal{P}(\beta) = \text{simplex} \cap \frac{1}{\beta} \mathbb{Z}^d$

Principal Lattice Interpolation

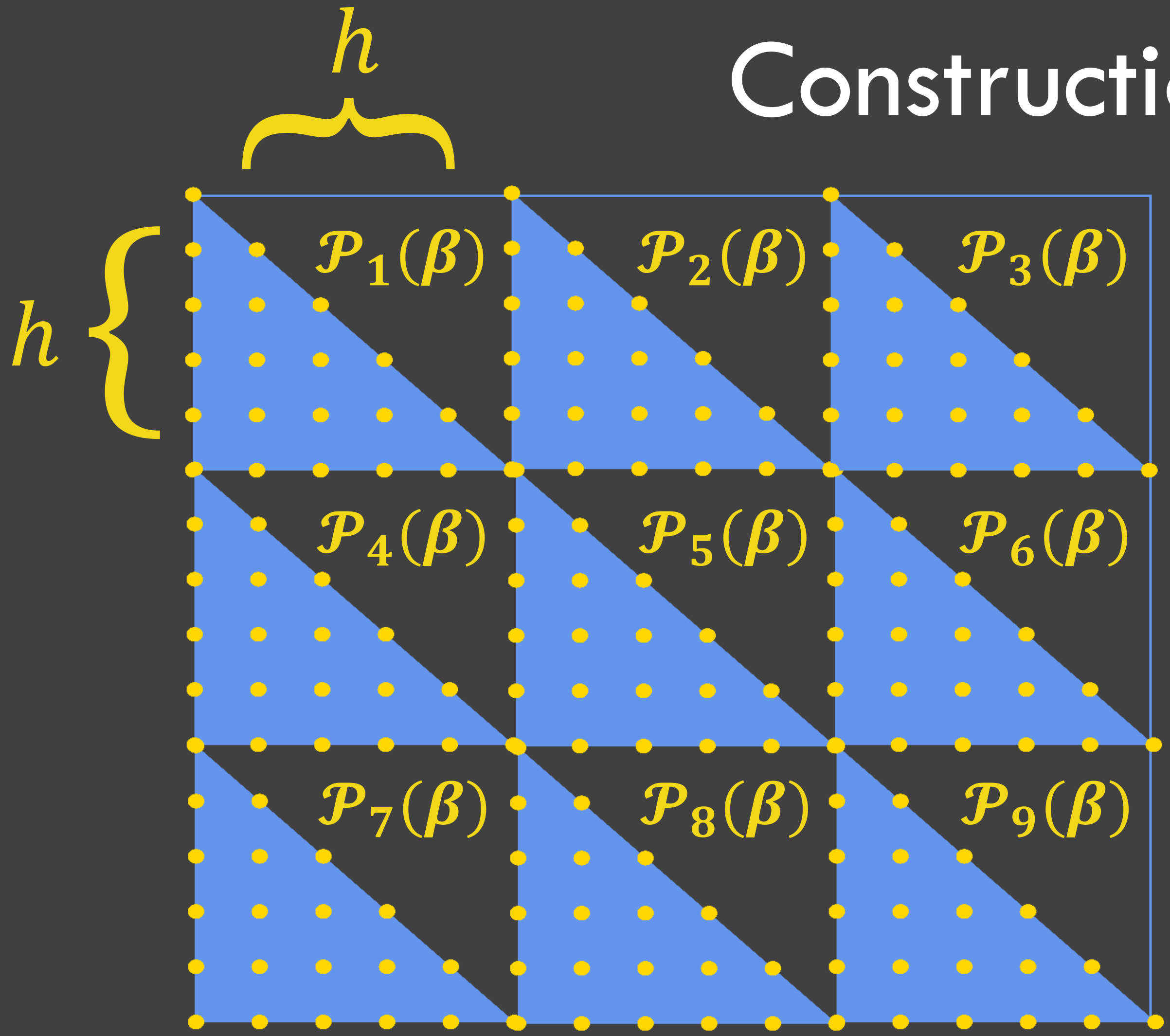


- $\mathcal{P}(\beta) = \text{simplex} \cap \frac{1}{\beta}\,\mathbb{Z}^d$
- $\mathcal{P}(\beta)$ has unique interpolants of degree β

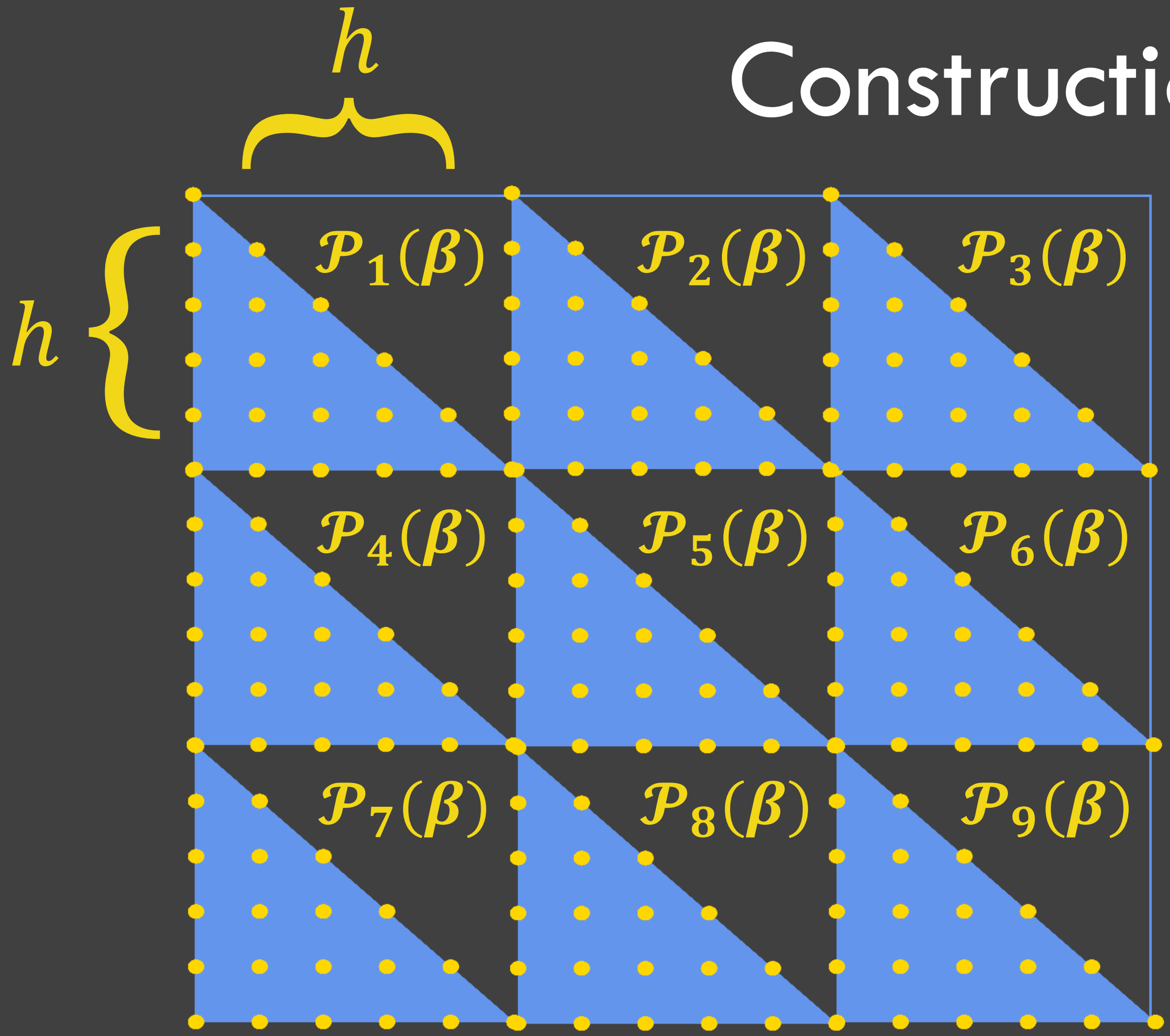
Construction of $\hat{g}$



Construction of $\hat{g}$

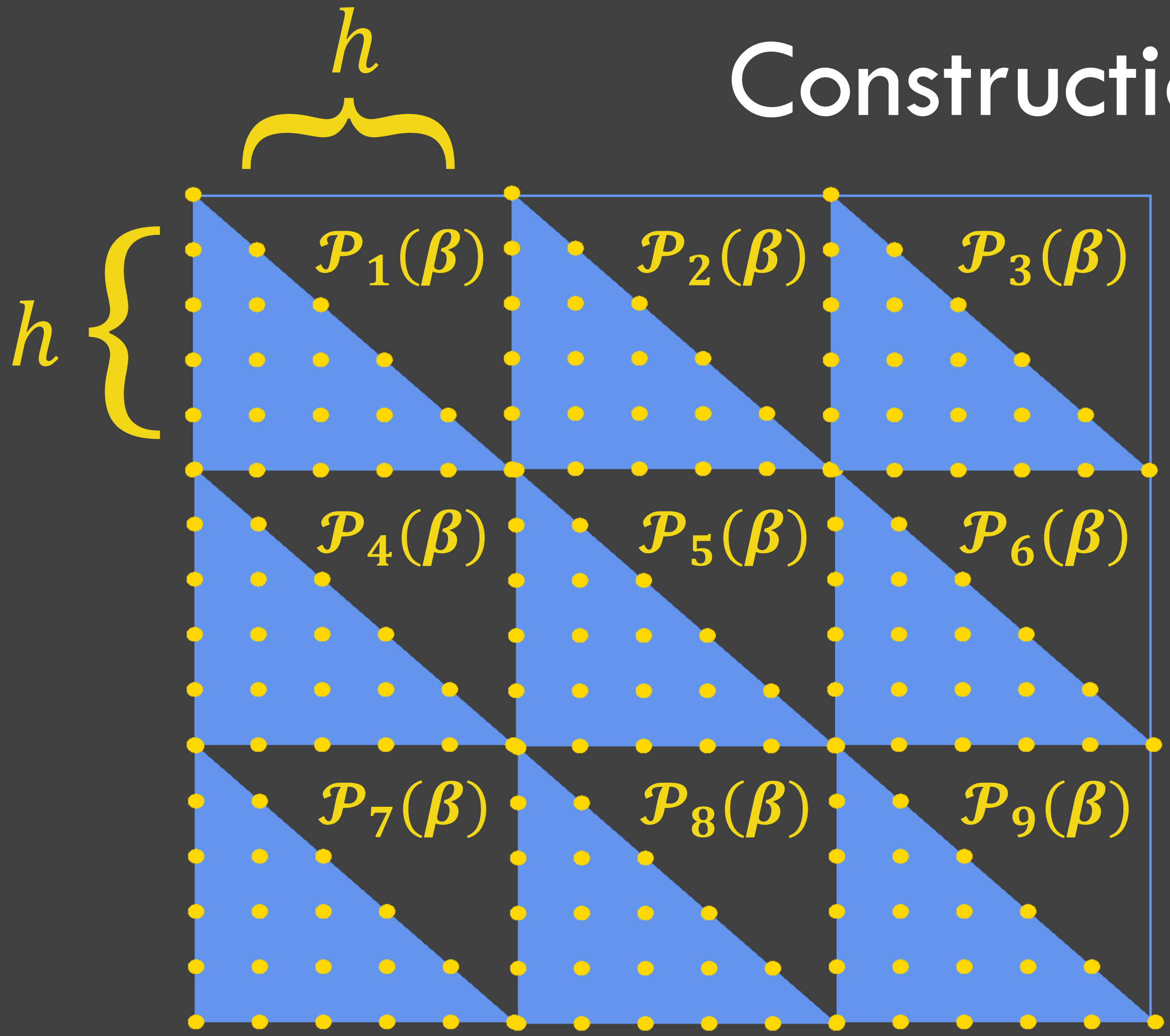


Construction of $\hat{g}$



For each box B_j, compute the interpolant $q_j(y)$ on $\mathcal{P}_j(\beta)$

Construction of $\hat{g}$



For each box B_j, compute the interpolant $q_j(y)$ on $\mathcal{P}_j(\beta)$

↓

$$\hat{g}(y) = \sum_{B_j} \mathbf{1}\left(y \in B_j\right) \sum_{x \in \mathcal{P}_j(\beta)} \hat{f}(x) \prod_{j=1}^{\beta} h_j^x(y)$$

Main Result

Theorem Suppose $\hat{f}$ is pointwise minimax optimal:

$$\sup_{y\in[0,1]^d} \mathbb{P}\left[\,\left|\hat{f}(y) - f(y)\right| > t\, n^{-\frac{\beta}{2\beta+d}}\right] \lesssim e^{-ct^2}$$

Main Result

Theorem Suppose $\hat{f}$ is pointwise minimax optimal:

$$\sup_{y\in[0,1]^d} \mathbb{P}\left[\,\left|\hat{f}(y) - f(y)\right| > t\, n^{-\frac{\beta}{2\beta+d}}\right] \lesssim e^{-ct^2}$$

Set $h \asymp n^{-\frac{1}{2\beta+d}}$. Then $\hat{g}$ has these properties.

- Sublinear space: $\tilde{O}_{\beta,d}(n^{\frac{d}{2\beta+d}})$

Main Result

Theorem Suppose $\hat{f}$ is pointwise minimax optimal:

$$\sup_{y \in [0,1]^d} \mathbb{P}\left[\left|\hat{f}(y) - f(y)\right| > t\, n^{-\frac{\beta}{2\beta+d}} \right] \lesssim e^{-ct^2}$$

Set $h \asymp n^{-\frac{1}{2\beta+d}}$. Then $\hat{g}$ has these properties.

- Sublinear space: $\tilde{O}_{\beta,d}(n^{\frac{d}{2\beta+d}})$
- Near-constant query time: $\tilde{O}_{\beta,d}(1)$

Main Result

Theorem Suppose $\hat{f}$ is pointwise minimax optimal:

$$\sup_{y\in[0,1]^d} \mathbb{P}\left[\,|\hat{f}(y) - f(y)\,| > t\, n^{-\frac{\beta}{2\beta+d}}\right] \lesssim e^{-ct^2}$$

Set $h \asymp n^{-\frac{1}{2\beta+d}}$. Then $\hat{g}$ has these properties.

- Sublinear space: $\tilde{O}_{\beta,d}(n^{\frac{d}{2\beta+d}})$
- Near-constant query time: $\tilde{O}_{\beta,d}(1)$
- Near-minimax error: $\|f - \hat{g}\|_\infty \lesssim \tilde{O}_{\beta,d}(n^{-\frac{\beta}{2\beta+d}})$

Demo