Multifidelity kernel regression

Dayoung Kang

Vision

High fidelity data are expensive and scarce

Building blocks

High fidelity mean estimation method
Suffer high variance in prediction

1. **Kernel regression**
2. **Multifidelity Monte Carlo (MFMC)**

Multifidelity approach that minimizes the variance of prediction

Both are mean estimation method

Goal: Apply MFMC mean estimation for kernel regression

Contents

Methods

- Kernel regression
- Multifidelity Monte Carlo (MFMC) estimator
- Multifidelity kernel regression

Show connection between the two

Results

1. Exponential function example
2. Ackely function example

Contents

Methods

- Kernel regression
- Multifidelity Monte Carlo (MFMC) estimator
- Multifidelity kernel regression

Show connection between the two

Results

1. Exponential function example
2. Ackely function example

Kernel regression: definition

- $X \in \mathbb{R}^d$: d-dimensional **input variable**
- $f^{(1)}: \mathbb{R}^d \rightarrow \mathbb{R}$: **high fidelity** input-output map
- $Y^{(1)} = f^{(1)}(X) \in \mathbb{R}$: high fidelity output variable
- $x \in \mathbb{R}^{d\times n}$: **$n$ i.i.d. realizations of X**
- $y^{(1)} \in \mathbb{R}^n$: **n i.i.d. realizations of Y**
- $x^* \in \mathbb{R}^d$: unseen point
- **Kernel regression predicts output by a weighted average of training data**

$$E\left[Y^{(1)} \middle| X = x^*\right] = \int y^{(1)} p_{Y|X=x^*}\left(y^{(1)}\right) dy^{(1)} = \frac{\int y^{(1)} p_{X,Y^{(1)}}\left(x^*, y^{(1)}\right) dy^{(1)}}{p_X(x^*)}$$

Kernel regression: derivation

$$E\left[Y^{(1)} \middle| X = x^*\right] = \int y^{(1)} p_{Y|X=x^*}\left(y^{(1)}\right) dy^{(1)} = \frac{\int y^{(1)} p_{X,Y^{(1)}}\left(x^*, y^{(1)}\right) dy^{(1)}}{p_X(x^*)}$$

$p_{X,Y^{(1)}}\left(x^*, y^{(1)}\right)$ and $p_X(x^*)$ are unknown in practice

Kernel regression introduces kernel density function

$$\hat{p}_X(x^*) = \frac{1}{n}\sum_{i=1}^{n} K_h(x^* - x_i)$$

$$\hat{p}_{X,Y^{(1)}}\left(x^*, y^{(1)}\right) = \frac{1}{n}\sum_{i=1}^{n} K_h(x^* - x_i) K_h(y^{(1)} - y_i^{(1)})$$

where $K_h(\cdot) = \frac{1}{h} K(\frac{\cdot}{h})$ **and** K is a kernel function

Kernel regression: derivation

Conditions for kernel density function K_h

1) Non-negative for all inputs
2) integrated to be 1
3) symmetric

So,

$$\hat{p}_{X,Y^{(1)}}\left(x^*, y^{(1)}\right) = \frac{1}{n}\sum_{i=1}^{n} K_h(x^* - x_i) K_h(y^{(1)} - y_i^{(1)})$$

where $K_h(\cdot) = \frac{1}{h}\, K(\frac{\cdot}{h})$ **and** K is a kernel function

Kernel regression: derivation

$$E\left[Y^{(1)} \middle| X = x^*\right] = \int y^{(1)} p_{Y|X=x^*}\left(y^{(1)}\right) dy^{(1)} = \frac{\int y^{(1)} p_{X,Y^{(1)}}\left(x^*, y^{(1)}\right) dy^{(1)}}{p_X(x^*)}$$

$$\approx \frac{\int y^{(1)} \hat{p}_{X,Y^{(1)}}\left(x^*, y^{(1)}\right) dy^{(1)}}{\hat{p}_X(x^*)}$$

$$\hat{p}_X(x^*) = \frac{1}{n}\sum_{i=1}^{n} K_h(x^* - x_i)$$

$$\hat{p}_{X,Y^{(1)}}\left(x^*, y^{(1)}\right) = \frac{1}{n}\sum_{i=1}^{n} K_h(x^* - x_i) K_h\left(y^{(1)} - y_i^{(1)}\right)$$

$$= \frac{\cancel{\frac{1}{n}} \int y^{(1)} \sum_{i=1}^{n} K_h(x^* - x_i) K_h\left(y^{(1)} - y_i^{(1)}\right) dy^{(1)}}{\cancel{\frac{1}{n}} \sum_{j=1}^{n} K_h\left(x^* - x_j\right)}$$

$$= \frac{\sum_{i=1}^{n} K_h(x^* - x_i) \cancel{\int y^{(1)} K_h\left(y^{(1)} - y_i^{(1)}\right) dy^{(1)}}}{\sum_{j=1}^{n} K_h\left(x^* - x_j\right)} \qquad y_i^{(1)}$$

$$E\left[Y^{(1)} \middle| X = x^*\right] \approx \frac{\sum_{i=1}^{n} K_h(x^* - x_i)}{\sum_{j=1}^{n} K_h\left(x^* - x_j\right)} y_i^{(1)}$$

Kernel regression: final expression

$$E\left[Y^{(1)} \middle| X = x^*\right] \approx \frac{\sum_{i=1}^{n} K_h\left(x^* - x_i\right)}{\sum_{j=1}^{n} K_h\left(x^* - x_j\right)} y_i^{(1)}$$

$$= \sum_{i=1}^{n} w_i(x^*) y_i^{(1)}$$

where

$$w_i(x^*) = \frac{K_h\left(x^* - x_i\right)}{\sum_{j=1}^{n} K_h\left(x^* - x_j\right)}$$

Note $\sum_{i=1}^{n} w_i = 1$

Kernel regression: final expression

$$E\left[Y^{(1)} \middle| X = x^*\right] \approx \frac{\sum_{i=1}^{n} K_h\left(x^* - x_i\right)}{\sum_{j=1}^{n} K_h\left(x^* - x_j\right)} y_i^{(1)}$$

$$= \sum_{i=1}^{n} w_i(x^*) y_i^{(1)}$$

where

Note $\sum_{i=1}^{n}$

Summary

- Introduced kernel regression
- Kernel regression estimates $E\left[Y^{(1)} \middle| X = x^*\right]$ using weighted average of high fidelity outputs

Q: High fidelity data is expensive. How can we introduce lower fidelity data?

Contents

Methods

- Kernel regression
- Multifidelity Monte Carlo (MFMC) estimator
- Multifidelity kernel regression

Show connection between the two

Results

1. Exponential function example
2. Ackely function example

Multifidelity Monte Carlo estimator

- Kernel regression estimates mean $E\left[Y^{(1)} \middle| X = x^*\right]$
- MFMC estimator is a mean estimator that minimizes variance of prediction
- Idea: **Apply MFMC estimator** for kernel regression problem
- Recall MFMC estimator

$$\mathbb{E}\left[f^{(1)}(X)\right] \approx \frac{1}{n}\sum_{i=1}^{n} f^{(1)}(x_i) + \alpha\left(\frac{1}{m}\sum_{i=1}^{m} f^{(2)}(x_i) - \frac{1}{n}\sum_{i=1}^{n} f^{(2)}(x_i)\right)$$

$f^{(2)}: \mathbb{R}^d \to \mathbb{R}$: low fidelity input-output map

n: number of high fidelity sample

m $(m \gg n)$: number of low fidelity sample

α: weight

Contents

Methods

Results

Multifidelity kernel regression

- Recall MFMC estimator

$$\mathbb{E}\left[f^{(1)}(X)\right] \approx \frac{1}{n}\sum_{i=1}^{n} f^{(1)}(x_i) + \alpha\left(\frac{1}{m}\sum_{i=1}^{m} f^{(2)}(x_i) - \frac{1}{n}\sum_{i=1}^{n} f^{(2)}(x_i)\right)$$

- Kernel regression can be reformulated as

$$E\left[Y^{(1)}\middle|X = x^*\right] = E\left[Y^{(1)}\middle|X = x^*\right] + \alpha\left(E\left[Y^{(k)}\middle|X = x^*\right] - E\left[Y^{(k)}\middle|X = x^*\right]\right)$$

- Recall kernel regression

$$E\left[Y^{(1)}\middle|X = x^*\right] \approx \sum_{i=1}^{n} w_{i,n}(x^*)y_i^{(1)}, \qquad w_{i,n}(x^*) = \frac{K_h(x^* - x_i)}{\sum_{j=1}^{n} K_h\left(x^* - x_j\right)}$$

$$E\left[Y^{(1)}\middle|X = x^*\right] \approx \sum_{i=1}^{n} w_{i,n}(x^*)y_i^{(1)} + \alpha\left(\sum_{i=1}^{m} w_{i,m}(x^*)y_i^{(2)} - \sum_{i=1}^{n} w_{i,m}(x^*)y_i^{(2)}\right)$$

Multifidelity kernel regression

- Recall MFMC estimator

$$\mathbb{E}[f^{(1)}(X)] \approx \frac{1}{n}\sum^{n} f^{(1)}(x_i) + \alpha\left(\frac{1}{m}\sum^{m} f^{(2)}(x_i) - \frac{1}{n}\sum^{n} f^{(2)}(x_i)\right)$$

- Ke

- Re

Summary

- Kernel regression and MFMC estimator both approximate mean
- MFMC estimator achieves variance reduction
- Applied MFMC estimator to kernel regression
- Multifidelity kernel regression achieve variance reduction

$$\sum_{i=1} \qquad \sum_{j=1}^{n} K_h(x^* - x_j)$$

$$E[Y^{(1)}|X = x^*] \approx \sum_{i=1}^{n} w_{i,n}(x^*) y_i^{(1)} + \alpha\left(\sum_{i=1}^{m} w_{i,m}(x^*) y_i^{(2)} - \sum_{i=1}^{n} w_{i,m}(x^*) y_i^{(2)}\right)$$

Contents

Methods

- Kernel regression
- Multifidelity Monte Carlo (MFMC) estimator
- Multifidelity kernel regression

Results

1. Exponential function example
2. Ackely function example

Exponential function example: Set up

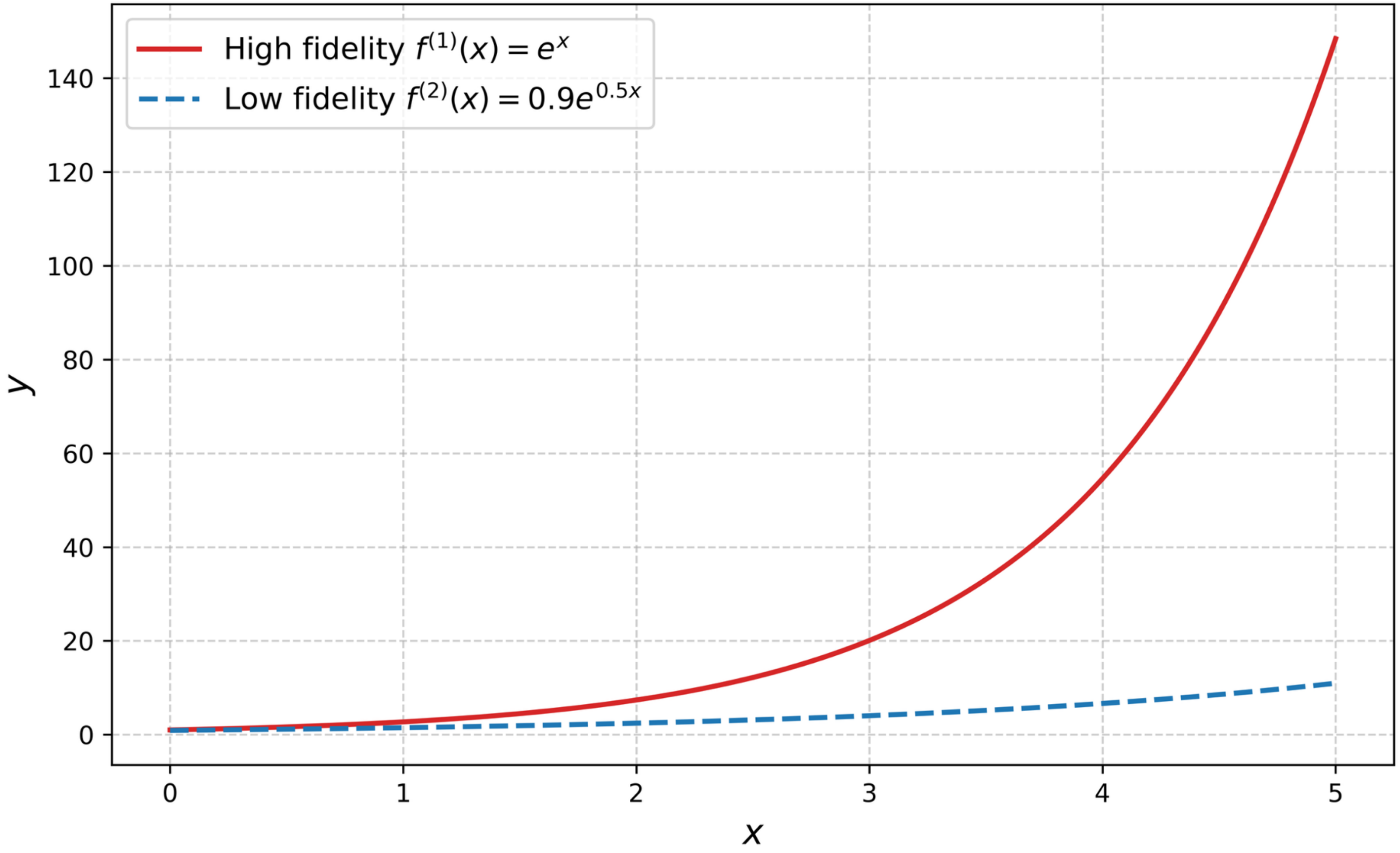
- Bifidelity functions

$$f^{(1)}(x) = \exp x$$

$$f^{(2)}(x) = 0.9\sqrt{\exp x}$$

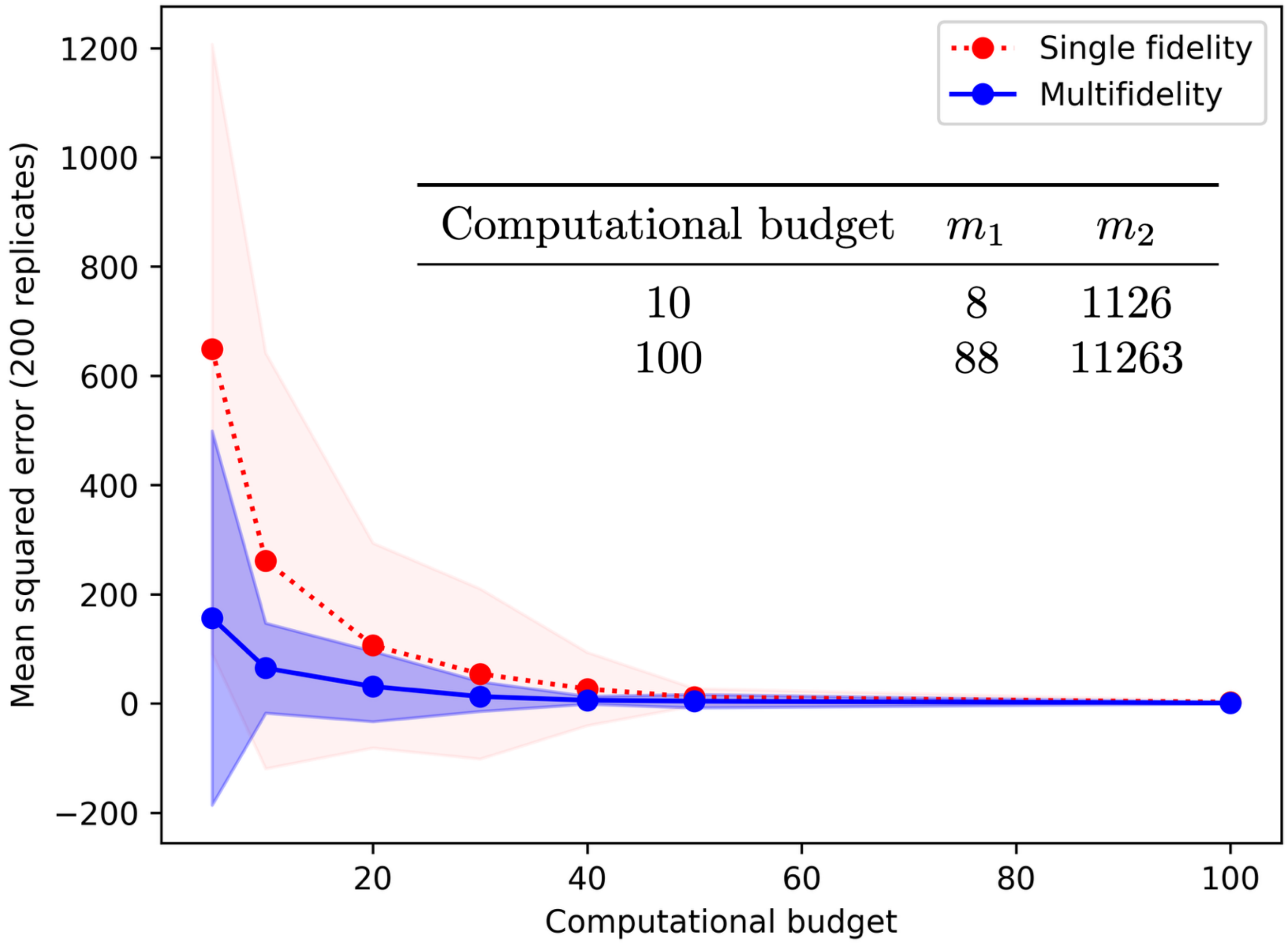
$$x \sim \mathcal{U}(0,5)$$

- Correlation coefficient: 0.97
- Cost: [1, 0.001]



Exponential function example: Results

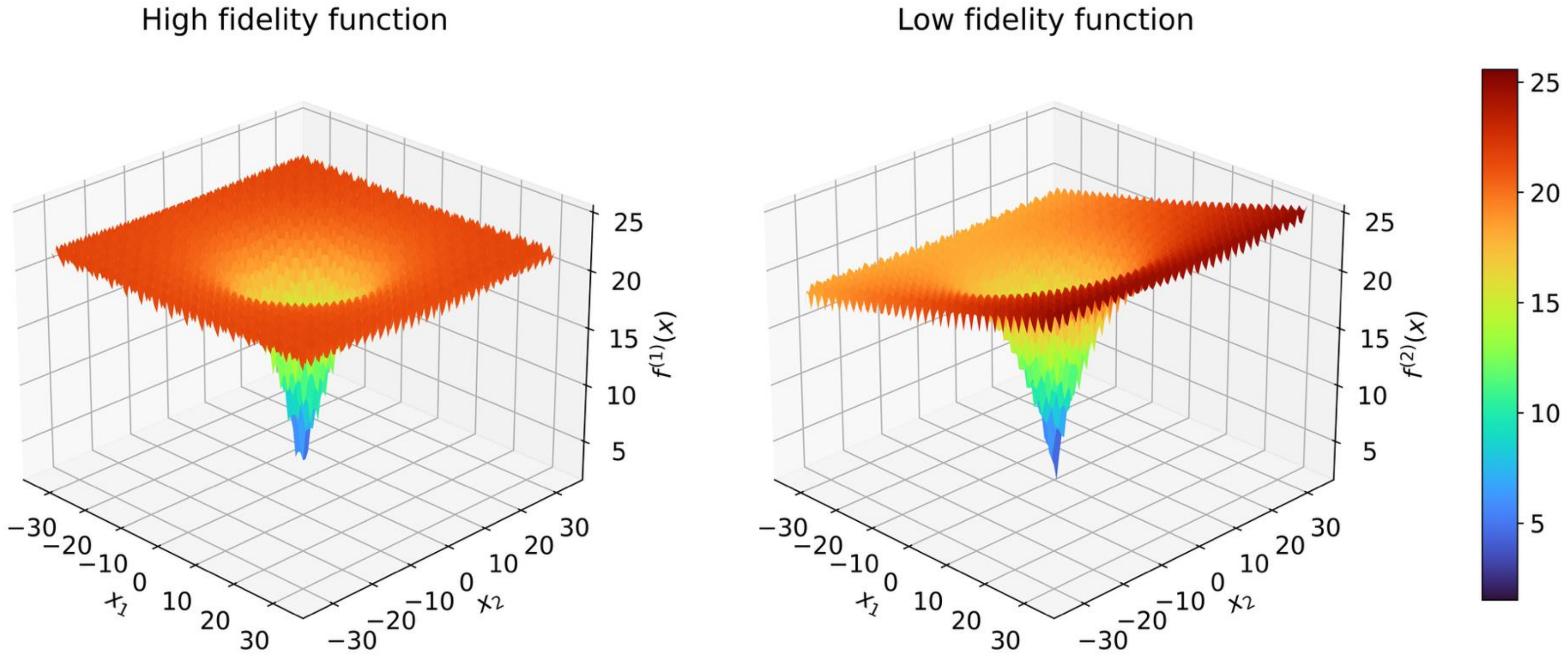
- Multifidelity kernel regression is more robust



Contents

Ackely function example: Set up

- Bifidelity functions

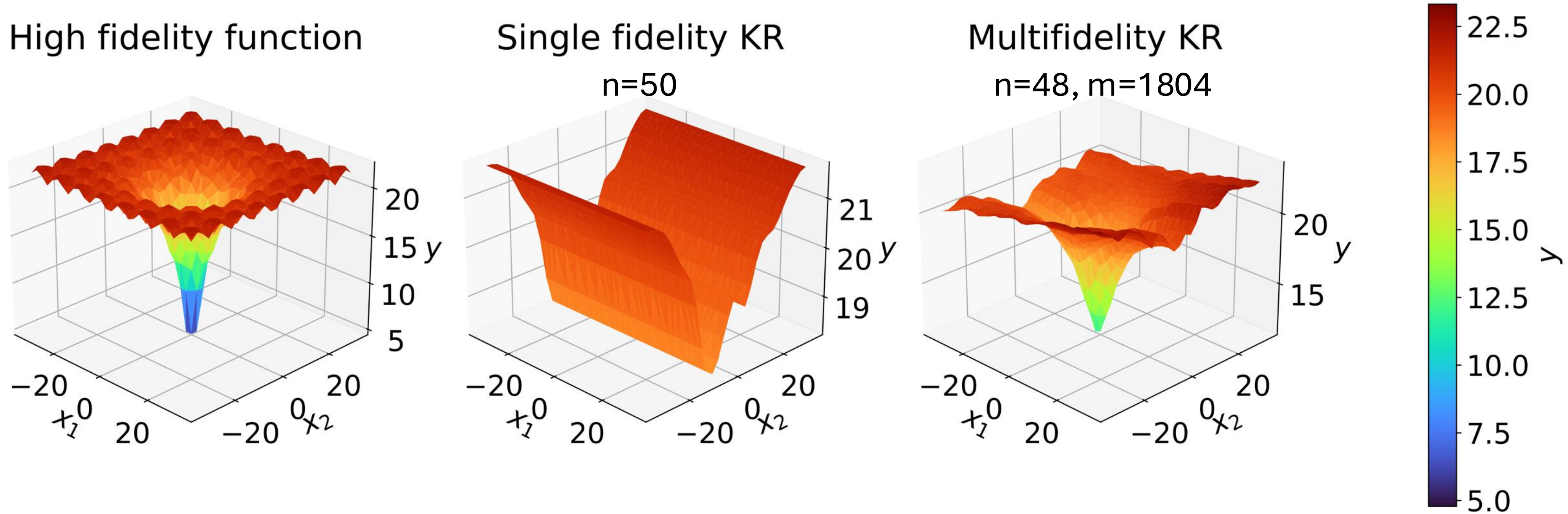


$x \sim \mathcal{U}(-32.768, 32.768)$

- Correlation coefficient: 0.76
- Cost: [1, 0.001]

Ackely function example: Results

- **Computational budget 50**
- Mean squared error
  - Single fidelity: 4.21
  - Multifidelity: 1.77



Summary and conclusion

- Defined multifidelity kernel regression model
- Ingredients
  - Kernel regression
  - MFMC estimator
- Multifidelity kernel regression model is more robust than the single fidelity counterpart
- Variance reduction is more significant in low budget regime