# MATH 5470 Project Presentation

## Multiple Kernel Kriging Methods for Aerospace Applications

Kehinde Oyetunde [20548072] [1], Sathi Sarveswara [20647048] [2]

[1]Mechanical and Aerospace Engineering Department,
[2]Electronics and Computer Engineering Department,
The Hong Kong University of Science and Technology, Hong Kong

June 1, 2020

# Computer experiment: Our motivation

- CFD as a computer experiment.
- Computations requires several iterations.
- Could take days to solve problems.
- Heavy computational power required.

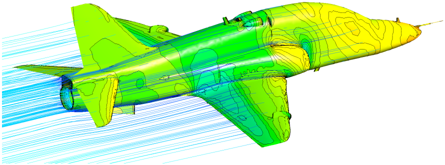


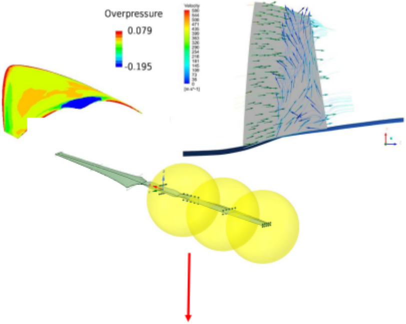
Figure 1: An aircraft[a]

[a] https://cfd2012.com/aircraft-design.html

# Surrogate models

They are techniques devised to significantly reduce the computational costs.

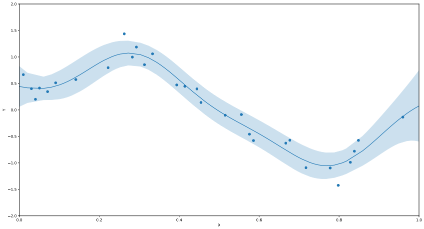
- They are also known as metamodels
- They are simpler, cheaper, and much faster to evaluate
- Design of Experiment (DoE) methods are used to generate training data
- Examples are RBF, `Kriging`




$$\boxed{f(\boldsymbol{x}) \approx \hat{f}(\boldsymbol{x})}$$

# Kriging (Gaussian processes)

Views the outputs of a function as a realization of a stochastic process.




$$m(\boldsymbol{x}) = \mu + \boldsymbol{r}(\boldsymbol{x})^T \boldsymbol{R}^{-1}(\boldsymbol{y} - \mathbf{1}\mu)$$

$$s^2(\boldsymbol{x}) = \hat{\sigma}^2 \left(1 - \boldsymbol{r}(\boldsymbol{x})^T \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x})\right)$$

# Kernels Used

- Gaussian

$$R(\boldsymbol{h}, \boldsymbol{\theta}) = \exp -\frac{1}{2}\left(\frac{\boldsymbol{h}}{\boldsymbol{\theta}}\right)^2 \tag{1}$$

- Exponential

$$R(\boldsymbol{h}, \boldsymbol{\theta}) = \exp -\frac{1}{2}\left(\frac{\boldsymbol{h}}{\boldsymbol{\theta}}\right) \tag{2}$$

- Matern 3/2

$$R(\boldsymbol{h}, \boldsymbol{\theta}) = \left(1 + \frac{\sqrt{3}|\boldsymbol{h}|}{\boldsymbol{\theta}}\right) exp\left(-\frac{\sqrt{3}|\boldsymbol{h}|}{\boldsymbol{\theta}}\right) \tag{3}$$

- Matern 5/2

$$R(\boldsymbol{h}, \boldsymbol{\theta}) = \left(1 + \frac{\sqrt{5}|\boldsymbol{h}|}{\boldsymbol{\theta}} + \frac{5\boldsymbol{h}^2}{3\boldsymbol{\theta}^2}\right) exp\left(-\frac{\sqrt{5}|\boldsymbol{h}|}{\boldsymbol{\theta}}\right) \tag{4}$$

where $h$ is a function of distance $|x - x'|$ and $\boldsymbol{\theta}$ is the length scale

# Ensemble Methods

- Proposed to overcome misspecification in optimization problems
- The goal of optimization is not to pursue global accuracy
- Ensemble techniques improved the robustness and performance of EGO
- Uncertainty vanishes (not useful in BO)
- Can either be local or global
- Uses weight to combine models
- Weight is computed
  - **`Akaike information criterion (AIC)`**
  - Bayesian information criterion (BIC)

$$\hat{y}_{ens}(x) = \sum_{i=1}^{K} w_i(x)\hat{y}_i(x)$$

# EM formulations

$$\hat{y}_{ens}(x) = \sum_{i=1}^{K} w_i(x)\hat{y}_i(x) \tag{5}$$

The weight is computed;

$$w_i = \frac{exp(-0.5\Delta AIC_i)}{\sum_{j=1}^{K} exp(-0.5\Delta AIC_j)} \tag{6}$$

where,

$$AIC = -2\ln(L) + 2N_f \tag{7}$$

$$AIC_c = AIC + \frac{2N_f^2 + 2N_f}{n - N_f - 1} \tag{8}$$

$$\Delta AIC_i = AIC_i - AIC_{min} \tag{9}$$

$$N_f = len(\theta) + 2 \tag{10}$$

# Composite kernel learning (CKL)

Current state-of-the-art method. Proposed by Palar *et. al.* (2019).

- It involves the construction of new kernels by combinations of existing kernels.
- Primarily focuses on the discovery of new kernels
- The new kernel is constructed so as to further optimize the likelihood function
- Uses weight to combine kernels
- Weight is usually optimized simultaneously with the hyperparameters.

$$R_{CKL}(h) = \sum_{i=1}^{K} w_i(x) R_i(h) \tag{11}$$

where $\mathbf{w} = w_1, w_2, ..., w_K{}^T$ is the non-negative weight vector that should satisfies the equality constraint $\sum_{i=1}^{K} w_i = 1$.

# Composite kernel learning (CKL)

- Capture complex data (when one kernel could be insufficient)
- Improve global accuracy of models
- Avoid misspecification

$$R_{CKL}(h) = \sum_{i=1}^{K} w_i(x) R_i(h)$$

# Multidimensional Composite kernel learning methods

- Capture even more complex data (when one kernel could be insufficient)
- Improve global accuracy of models
- Avoid misspecification

# Multidimensional composite kernel learning (MCKL)

- Take the sample data, $\boldsymbol{x}$ and $\mathbf{y}$, kernel list $\boldsymbol{K} = [k_1, k_2, ...k_m]$, $\boldsymbol{\theta_{initial}}$
- Compute the euclidean distance between the sample input, $h = |x_i - x_j|$
- Initialize $\boldsymbol{w}$ as $\left[\frac{1}{m}\right] \times n$, where m is the number of available kernels and n is the dimension of the data.

$$\mathbf{w} = \begin{pmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,m} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n,1} & w_{n,2} & \cdots & w_{n,m} \end{pmatrix}$$

- Compute $R_{ij}(\boldsymbol{\theta}, h) = \prod_{d=1}^{n} \sum_{i=1}^{m} w_{nm} R\left(\theta^{(d)}, h_{ij}^{(d)}\right)$

# Multidimensional composite kernel learning (MCKL) cont'd

- Complete the model training by optimizing the hyperparameter $\boldsymbol{\theta}$ and weights $\boldsymbol{w}$ by minimizing the Maximum Likelihood Estimate (MLE) to obtain $\boldsymbol{\theta_{final}}$ and $\boldsymbol{w_{optimum}}$
- Compute $R(\boldsymbol{\theta}, h) = \prod_{d=1}^{n} \sum_{i=1}^{m} w_i R\left(\theta^{(d)}, h^{(d)}\right)$
- Complete model training

# Dataset (Airfoil Coordinates)

- **Input:** Shape parameters, $x_i$
- **Output:** Coefficient of drag ($c_d$)
- **8D:** Size = 500 points
- **16D:** Size = 1000 points
- **Airfoil:** FFAST
- **Mach:** 0.80
- **AOA:** 1.25 deg
- **Gridsize:** Medium (20,000 cells)

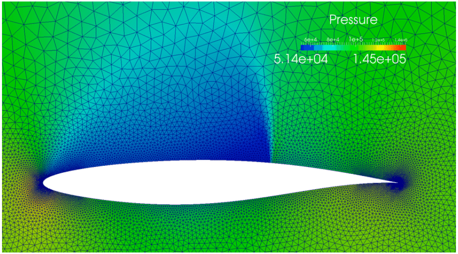
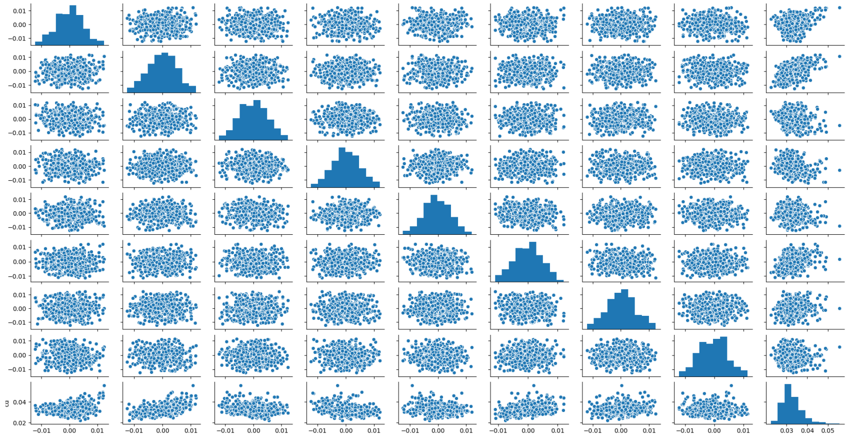



Figure 2: CFD result

# Data distribution (8D)



# Approach

1. Divide the data into training and truth sets (80-20 plan).
2. Draw out samples using Halton-based sequence ($N_s$ = 20,40,60) from the training set.
3. Train the model.
4. Predict using truth set inputs.
5. Estimate model performance on prediction of the truth set output using different metrics.

# Halton-based sampling plan

**Input:** Sampling size, $N_s$
**Output:** $x_{sample}$,$y_{sample}$
**Data:** $\mathbf{x}$, $y$

1. Get k from the data
2. Design a Halton sequence with dimension,k and size,$N_s$
3. Normalize $\mathbf{x}$ : $x_{norm}$
4. Map the generated sequence on $x_{norm}$ and select the nearest points: $x_{s,norm}$
5. Denormalize $x_{s,norm}$ : $x_{sample}$
6. Get the corresponding data from y : $y_{sample}$

# Illustration: Sampling plan

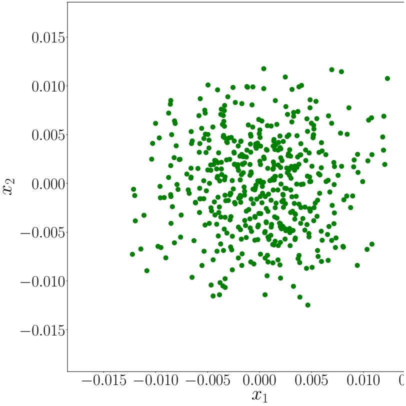



Figure 3: Data

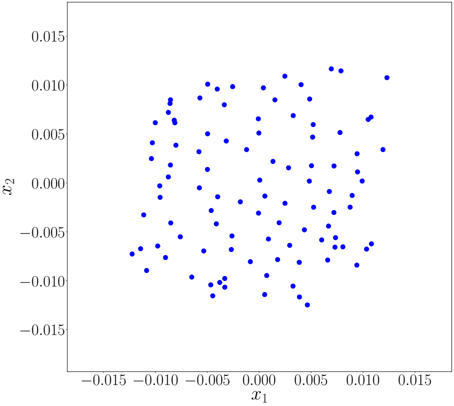



Figure 4: Samples (100)

# Performance Metric

- Normalized root mean square, NRMSE (%)

$$NRMSE = 100 * \sqrt{\frac{1}{m}\sum_{i}^{m}\left(\frac{y_i - \hat{y}_i}{y_i}\right)^2} \tag{12}$$

- Coefficient of determination, R2 score, $R^2$

$$R^2 = 1 - \frac{\sum_i^m (y_i - \hat{y}_i)^2}{\sum_i^m (y_i - \bar{y})^2} \tag{13}$$

# Model weights (8D)

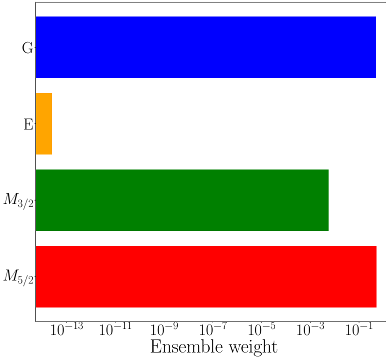



Figure 5: Ensemble model

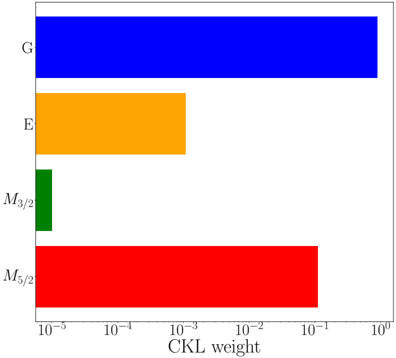



Figure 6: CKL model

# Results (8D)

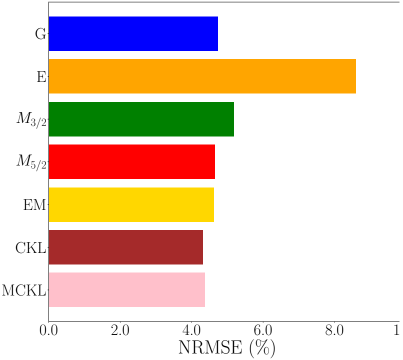



Figure 7: NRMSE (%)

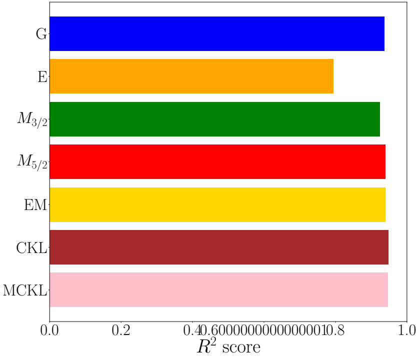



Figure 8: Coefficient of determination

# Training time (8D)

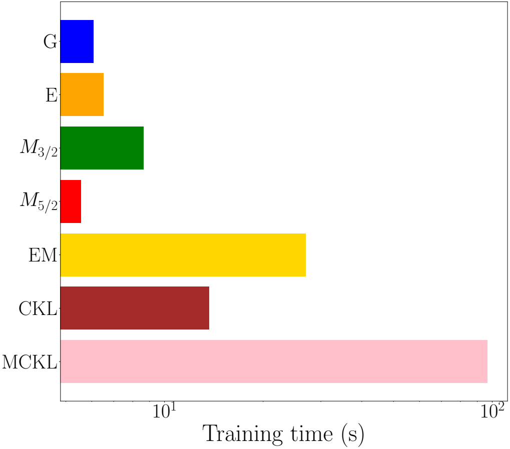



Figure 9: Training time (seconds)

# Model weights (16D)

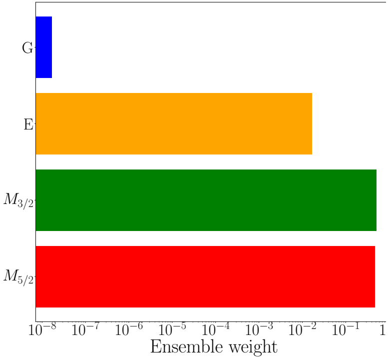



Figure 10: Ensemble model

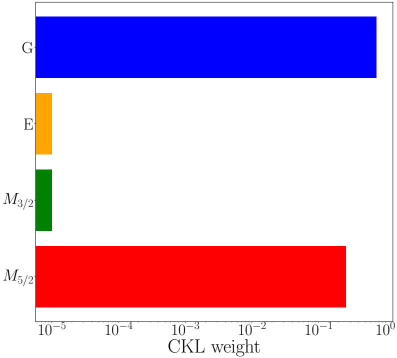



Figure 11: CKL model

# Results (16D)

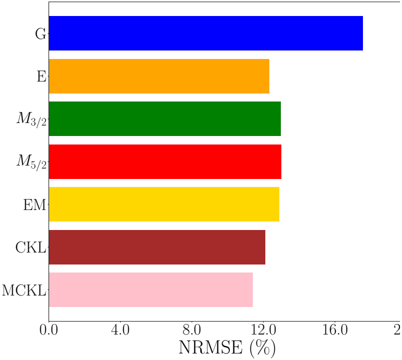



Figure 12: NRMSE (%)

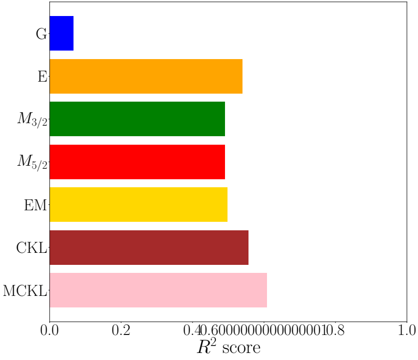



Figure 13: Coefficient of determination

# Training time (16D)

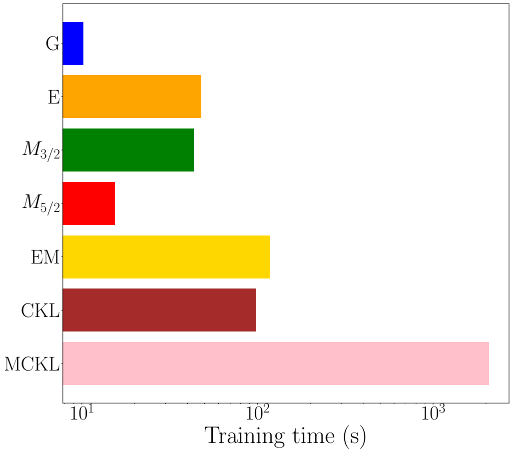



Figure 14: Training time (seconds)

# Summary

- Multiple kernel methods are more robust with better performance
- MCKL incurred the highest computational cost due to complexity of model
- CKL method is considered best for building models to predict drag coefficients
  - Least training time for multiple kernel methods
  - Reasonable model accuracy

# Contributions

1. We developed a Halton-based sampling technique
2. We developed three multiple kernel learning methods based on weighting
3. We benchmarked performance of methods with single-kernel models

# Reference


1. Jay D. Martin and Timothy W. Simpson. Use of kriging models to approximate deterministic computer models. 43:853–863, 2005.
2. Pramudita S. Palar and Koji Shimoyama. Kriging with composite kernel learning for surrogate modeling in computer experiments. 2019.
3. Pramudita Satria Palar and Koji Shimoyama. Ensemble of kriging with multiple kernel functions for engineering design optimization, 2018.
4. Pramudita Satria Palar and Koji Shimoyama. Efficient global optimization with ensemble and selection of kernel functions for engineering design. 59:93–116, 2019.