# Bayesian approaches for multiscale modeling

David Dunson

Departments of Statistical Science, Mathematics & ECE, Duke University

Funded from the Office of Naval Research (ONR)

# Bayesian Uncertainty Quantification (UQ)

- Bayesian paradigm provides a useful framework for uncertainty quantification (UQ)

# Bayesian Uncertainty Quantification (UQ)

- Bayesian paradigm provides a useful framework for uncertainty quantification (UQ)
- Let $\theta \in \Theta$ denote unknown parameters & $L(y|\theta)$ denote the likelihood function of data $y$ given these parameters

# Bayesian Uncertainty Quantification (UQ)

- Bayesian paradigm provides a useful framework for uncertainty quantification (UQ)
- Let $\theta \in \Theta$ denote unknown parameters & $L(y|\theta)$ denote the likelihood function of data $y$ given these parameters
- Bayes inference starts with specifying a prior probability distribution $\pi(\theta)$ for the unknowns

# Bayesian Uncertainty Quantification (UQ)

- Bayesian paradigm provides a useful framework for uncertainty quantification (UQ)
- Let $\theta \in \Theta$ denote unknown parameters & $L(y|\theta)$ denote the likelihood function of data $y$ given these parameters
- Bayes inference starts with specifying a prior probability distribution $\pi(\theta)$ for the unknowns
- This probability quantifies our uncertainty prior to observing data $y$ and may include structural constraints

# Bayesian Uncertainty Quantification (UQ)

- Bayesian paradigm provides a useful framework for uncertainty quantification (UQ)
- Let $\theta \in \Theta$ denote unknown parameters & $L(y|\theta)$ denote the likelihood function of data $y$ given these parameters
- Bayes inference starts with specifying a prior probability distribution $\pi(\theta)$ for the unknowns
- This probability quantifies our uncertainty prior to observing data $y$ and may include structural constraints
- Updating the prior distribution with the likelihood function, we obtain the posterior probability distribution:

$$\pi(\theta|y) = \frac{\pi(\theta)L(y|\theta)}{\int_\Theta \pi(\theta)L(y|\theta)d\theta} = \frac{\pi(\theta)L(y|\theta)}{L(y)}.$$

# Bayesian Uncertainty Quantification (UQ)

- Bayesian paradigm provides a useful framework for uncertainty quantification (UQ)
- Let $\theta \in \Theta$ denote unknown parameters & $L(y|\theta)$ denote the likelihood function of data $y$ given these parameters
- Bayes inference starts with specifying a prior probability distribution $\pi(\theta)$ for the unknowns
- This probability quantifies our uncertainty prior to observing data $y$ and may include structural constraints
- Updating the prior distribution with the likelihood function, we obtain the posterior probability distribution:

$$\pi(\theta|y) = \frac{\pi(\theta)L(y|\theta)}{\int_\Theta \pi(\theta)L(y|\theta)d\theta} = \frac{\pi(\theta)L(y|\theta)}{L(y)}.$$

- $\pi(\theta|y)$ quantifies uncertainty about $\theta$ & functionals $f(\theta)$

# Some notes on this framework

- The unknown ‘parameter’ $\theta$ can be very broad - not just scalar inputs but unknown functions, surfaces, tensors, latent data, etc can be included in $\theta$

# Some notes on this framework

- The unknown 'parameter' $\theta$ can be very broad - not just scalar inputs but unknown functions, surfaces, tensors, latent data, etc can be included in $\theta$
- Outside of simple conjugate familes, $\pi(\theta|y)$ is not analytically tractable - even if likelihood $L(y|\theta)$ is available, marginal likelihood $L(y)$ involves a high-dimensional integral

# Some notes on this framework

- The unknown 'parameter' $\theta$ can be very broad - not just scalar inputs but unknown functions, surfaces, tensors, latent data, etc can be included in $\theta$
- Outside of simple conjugate familes, $\pi(\theta|y)$ is not analytically tractable - even if likelihood $L(y|\theta)$ is available, marginal likelihood $L(y)$ involves a high-dimensional integral
- To avoid potentially inaccurate integral approximations to $L(y)$, Monte Carlo methods are widely used

# Some notes on this framework

- The unknown 'parameter' $\theta$ can be very broad - not just scalar inputs but unknown functions, surfaces, tensors, latent data, etc can be included in $\theta$
- Outside of simple conjugate familes, $\pi(\theta|y)$ is not analytically tractable - even if likelihood $L(y|\theta)$ is available, marginal likelihood $L(y)$ involves a high-dimensional integral
- To avoid potentially inaccurate integral approximations to $L(y)$, Monte Carlo methods are widely used
- Markov chain Monte Carlo (MCMC) constructs a Markov chain with stationary distribution $\pi(\theta|y)$

# Some notes on this framework

- The unknown 'parameter' $\theta$ can be very broad - not just scalar inputs but unknown functions, surfaces, tensors, latent data, etc can be included in $\theta$
- Outside of simple conjugate familes, $\pi(\theta|y)$ is not analytically tractable - even if likelihood $L(y|\theta)$ is available, marginal likelihood $L(y)$ involves a high-dimensional integral
- To avoid potentially inaccurate integral approximations to $L(y)$, Monte Carlo methods are widely used
- Markov chain Monte Carlo (MCMC) constructs a Markov chain with stationary distribution $\pi(\theta|y)$
- Bypasses ever needing to calculate $L(y)$ & highly complex models can be considered

# UQ in applied math mechanistic modeling

- Ok so how is this useful in multscale/mechanistic modeling in applied math?

# UQ in applied math mechanistic modeling

- Ok so how is this useful in multscale/mechanistic modeling in applied math?
- In such settings, one has some intricate system of equations (e.g., ODEs, PDEs, etc)

# UQ in applied math mechanistic modeling

- Ok so how is this useful in multscale/mechanistic modeling in applied math?
- In such settings, one has some intricate system of equations (e.g., ODEs, PDEs, etc)
- For known inputs, data can be forward simulated from the mechanistic model

# UQ in applied math mechanistic modeling

- Ok so how is this useful in multscale/mechanistic modeling in applied math?
- In such settings, one has some intricate system of equations (e.g., ODEs, PDEs, etc)
- For known inputs, data can be forward simulated from the mechanistic model
- Also, there are commonly solvers available that can be used to 'fit' data

# UQ in applied math mechanistic modeling

- Ok so how is this useful in multscale/mechanistic modeling in applied math?
- In such settings, one has some intricate system of equations (e.g., ODEs, PDEs, etc)
- For known inputs, data can be forward simulated from the mechanistic model
- Also, there are commonly solvers available that can be used to ‘fit’ data
- Multiple types of uncertainty: (i) unknown inputs; (ii) model may not exactly characterize observed data; (iii) may be difficult to model ‘everything’ (e.g., variability across subjects or conditions) mechanistically

# UQ in applied math mechanistic modeling

- Ok so how is this useful in multscale/mechanistic modeling in applied math?
- In such settings, one has some intricate system of equations (e.g., ODEs, PDEs, etc)
- For known inputs, data can be forward simulated from the mechanistic model
- Also, there are commonly solvers available that can be used to 'fit' data
- Multiple types of uncertainty: (i) unknown inputs; (ii) model may not exactly characterize observed data; (iii) may be difficult to model 'everything' (e.g., variability across subjects or conditions) mechanistically
- Bayesian paradigm potentially very useful for solving such problems

# Bayes UQ - issues & typical approaches

- Typically not possible to analytically express the likelihood $L(y|\theta)$ of the data under an interesting mechanistic model

# Bayes UQ - issues & typical approaches

- Typically not possible to analytically express the likelihood $L(y|\theta)$ of the data under an interesting mechanistic model
- Unclear how to use the Bayes paradigm for UQ without a likelihood

# Bayes UQ - issues & typical approaches

- Typically not possible to analytically express the likelihood $L(y|\theta)$ of the data under an interesting mechanistic model
- Unclear how to use the Bayes paradigm for UQ without a likelihood
- Common solutions to likelihood specification problem:

# Bayes UQ - issues & typical approaches

- Typically not possible to analytically express the likelihood $L(y|\theta)$ of the data under an interesting mechanistic model
- Unclear how to use the Bayes paradigm for UQ without a likelihood
- Common solutions to likelihood specification problem:
  1. Use a usual solver without considering UQ & put solution in as 'center' of simple statistical distribution

# Bayes UQ - issues & typical approaches

- Typically not possible to analytically express the likelihood $L(y|\theta)$ of the data under an interesting mechanistic model
- Unclear how to use the Bayes paradigm for UQ without a likelihood
- Common solutions to likelihood specification problem:
  1. Use a usual solver without considering UQ & put solution in as 'center' of simple statistical distribution
  2. Use approximate Bayes computation (ABC) methods, which only require a forward simulator & not an explicit likelihood

# Bayes UQ - Gaussian process emulators

- A very widely used approach is Gaussian process (GP) ‘emulation’

# Bayes UQ - Gaussian process emulators

- A very widely used approach is Gaussian process (GP) 'emulation'
- Let $y = g(x; \theta)$ denote the input-output relationship under some posited mechanistic (perhaps multiscale) model with conditions $\theta$

# Bayes UQ - Gaussian process emulators

- A very widely used approach is Gaussian process (GP) 'emulation'
- Let $y = g(x; \theta)$ denote the input-output relationship under some posited mechanistic (perhaps multiscale) model with conditions $\theta$
- Based on data $(x_i, y_i)$, $i = 1, \ldots, n$, we apply a solver to obtain $\hat{g}$

# Bayes UQ - Gaussian process emulators

- A very widely used approach is Gaussian process (GP) 'emulation'
- Let $y = g(x; \theta)$ denote the input-output relationship under some posited mechanistic (perhaps multiscale) model with conditions $\theta$
- Based on data $(x_i, y_i)$, $i = 1, \ldots, n$, we apply a solver to obtain $\hat{g}$
- There can be errors in the solver & biases in the model

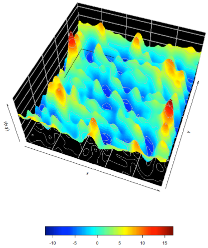
# Bayes UQ - Gaussian process emulators

- A very widely used approach is Gaussian process (GP) 'emulation'
- Let $y = g(x;\theta)$ denote the input-output relationship under some posited mechanistic (perhaps multiscale) model with conditions $\theta$
- Based on data $(x_i, y_i)$, $i = 1, \ldots, n$, we apply a solver to obtain $\hat{g}$
- There can be errors in the solver & biases in the model
- To account for this, let $y_i = \mu(x_i) + \epsilon_i, \epsilon_i \sim N(0, \sigma^2)$

# Bayes UQ - Gaussian process emulators

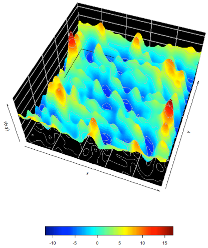
- A very widely used approach is Gaussian process (GP) 'emulation'
- Let $y = g(x; \theta)$ denote the input-output relationship under some posited mechanistic (perhaps multiscale) model with conditions $\theta$
- Based on data $(x_i, y_i)$, $i = 1, \ldots, n$, we apply a solver to obtain $\hat{g}$
- There can be errors in the solver & biases in the model
- To account for this, let $y_i = \mu(x_i) + \epsilon_i, \epsilon_i \sim N(0, \sigma^2)$
- $\mu \sim \text{GP}(\hat{g}, c)$ = unknown function, $\epsilon_i$ = measurement error

# Gaussian process (GP) overview



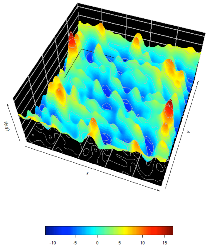

- Gaussian process provides a ‘prior’ for an unknown function $\mu$ mapping from inputs to outputs

# Gaussian process (GP) overview




- Gaussian process provides a ‘prior’ for an unknown function $\mu$ mapping from inputs to outputs
- Realizations $\mu \sim \text{GP}(\hat{g}, c)$ are random functions/stochastic processes centered on $\hat{g}$ on average

# Gaussian process (GP) overview




- Gaussian process provides a ‘prior’ for an unknown function $\mu$ mapping from inputs to outputs
- Realizations $\mu \sim \mathrm{GP}(\hat{g}, c)$ are random functions/stochastic processes centered on $\hat{g}$ on average
- Variance and smoothness of the realizations controlled by the covariance function:

$$\mathrm{cov}\{\mu(x), \mu(x')\} = c_\phi(x, x'),$$

where $\phi$ are (potentially unknown) parameters

# Some comments on GPs

- Often a default covariance is used that doesn't include mechanistic information; eg,

$$c_\phi(x, x') = \phi_1 \exp\{-\phi_2 ||x, x'||_2^2\},$$

$\phi_1$ controls amplitude variability & $\phi_2$ smoothness

# Some comments on GPs

- Often a default covariance is used that doesn't include mechanistic information; eg,

$$c_\phi(x, x') = \phi_1 \exp\{-\phi_2 ||x, x'||_2^2\},$$

$\phi_1$ controls amplitude variability & $\phi_2$ smoothness

- If the 'true' function $\mu_0$ is smooth with a global level of smoothness $\alpha$, the GP has excellent theoretical properties

# Some comments on GPs

- Often a default covariance is used that doesn't include mechanistic information; eg,

$$c_\phi(x, x') = \phi_1 \exp\{-\phi_2 ||x, x'||_2^2\},$$

  $\phi_1$ controls amplitude variability & $\phi_2$ smoothness
- If the 'true' function $\mu_0$ is smooth with a global level of smoothness $\alpha$, the GP has excellent theoretical properties
- If we choose a prior for the smoothness $\phi_2$, the posterior for $\mu$ concentrates around $\mu_0$ at the minimax optimal adaptive rate

# Some comments on GPs

- Often a default covariance is used that doesn't include mechanistic information; eg,

$$c_\phi(x, x') = \phi_1 \exp\{-\phi_2 ||x, x'||_2^2\},$$

  $\phi_1$ controls amplitude variability & $\phi_2$ smoothness
- If the 'true' function $\mu_0$ is smooth with a global level of smoothness $\alpha$, the GP has excellent theoretical properties
- If we choose a prior for the smoothness $\phi_2$, the posterior for $\mu$ concentrates around $\mu_0$ at the minimax optimal adaptive rate
- Also very convenient computationally: $\mu \sim \mathrm{GP}(\hat{g}, c)$ implies

$$\{\mu(x_1), \ldots, \mu(x_n)\} \sim N_n(\{\hat{g}(x_1), \ldots, \hat{g}(x_n)\}, C_n),$$

  where $C_n \sim n \times n$ covariance matrix with elements $c_\phi(x_i, x_j)$.

# Some comments on GPs

- Often a default covariance is used that doesn't include mechanistic information; eg,

$$c_\phi(x, x') = \phi_1 \exp\{-\phi_2 ||x, x'||_2^2\},$$

  $\phi_1$ controls amplitude variability & $\phi_2$ smoothness
- If the 'true' function $\mu_0$ is smooth with a global level of smoothness $\alpha$, the GP has excellent theoretical properties
- If we choose a prior for the smoothness $\phi_2$, the posterior for $\mu$ concentrates around $\mu_0$ at the minimax optimal adaptive rate
- Also very convenient computationally: $\mu \sim \text{GP}(\hat{g}, c)$ implies

$$\{\mu(x_1), \dots, \mu(x_n)\} \sim N_n(\{\hat{g}(x_1), \dots, \hat{g}(x_n)\}, C_n),$$

  where $C_n \sim n \times n$ covariance matrix with elements $c_\phi(x_i, x_j)$.
- This prior for $\mu$ evaluated at a finite number of inputs is *conjugate* to the normal likelihood of the measurements $\{y_i\}$

# Some comments on GPs

- Often a default covariance is used that doesn't include mechanistic information; eg,

$$c_\phi(x, x') = \phi_1 \exp\{-\phi_2 ||x, x'||_2^2\},$$

$\phi_1$ controls amplitude variability & $\phi_2$ smoothness
- If the 'true' function $\mu_0$ is smooth with a global level of smoothness $\alpha$, the GP has excellent theoretical properties
- If we choose a prior for the smoothness $\phi_2$, the posterior for $\mu$ concentrates around $\mu_0$ at the minimax optimal adaptive rate
- Also very convenient computationally: $\mu \sim \mathrm{GP}(\hat{g}, c)$ implies

$$\{\mu(x_1), \ldots, \mu(x_n)\} \sim N_n(\{\hat{g}(x_1), \ldots, \hat{g}(x_n)\}, C_n),$$

where $C_n \sim n \times n$ covariance matrix with elements $c_\phi(x_i, x_j)$.
- This prior for $\mu$ evaluated at a finite number of inputs is *conjugate* to the normal likelihood of the measurements $\{y_i\}$
- The posterior for $\mu | y_1, \ldots, y_n, \hat{g}, \phi$ has a simple analytic form

# Some issues with ‘vanilla’ GPs for UQ

- The GP approach to UQ has been widely used & successful but has some issues

# Some issues with ‘vanilla’ GPs for UQ

- The GP approach to UQ has been widely used & successful but has some issues
- Off the shelf generic covariance functions can lead to overly erratic & unstructured deviations from $\hat{g}$.

# Some issues with 'vanilla' GPs for UQ

- The GP approach to UQ has been widely used & successful but has some issues
- Off the shelf generic covariance functions can lead to overly erratic & unstructured deviations from $\hat{g}$.
- Would be appealing for the realizations of the unknown $\mu$ to inherit some characteristics of the mechanistic model

# Some issues with 'vanilla' GPs for UQ

- The GP approach to UQ has been widely used & successful but has some issues
- Off the shelf generic covariance functions can lead to overly erratic & unstructured deviations from $\hat{g}$.
- Would be appealing for the realizations of the unknown $\mu$ to inherit some characteristics of the mechanistic model
- For example, the realizations may resemble realizations of SDEs/SPDEs or solutions of ODEs/PDEs

# Some issues with 'vanilla' GPs for UQ

- The GP approach to UQ has been widely used & successful but has some issues
- Off the shelf generic covariance functions can lead to overly erratic & unstructured deviations from $\hat{g}$.
- Would be appealing for the realizations of the unknown $\mu$ to inherit some characteristics of the mechanistic model
- For example, the realizations may resemble realizations of SDEs/SPDEs or solutions of ODEs/PDEs
- May also have a multiscale/non-smooth character

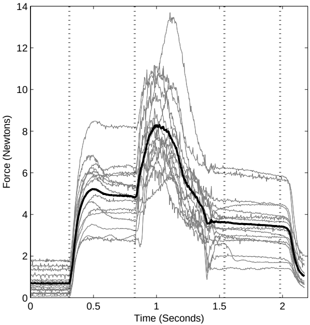
# Some issues with 'vanilla' GPs for UQ

- The GP approach to UQ has been widely used & successful but has some issues
- Off the shelf generic covariance functions can lead to overly erratic & unstructured deviations from $\hat{g}$.
- Would be appealing for the realizations of the unknown $\mu$ to inherit some characteristics of the mechanistic model
- For example, the realizations may resemble realizations of SDEs/SPDEs or solutions of ODEs/PDEs
- May also have a multiscale/non-smooth character
- Broadly *mechanistic nonparametric Bayes* models can be designed for these problems

# Some issues with 'vanilla' GPs for UQ

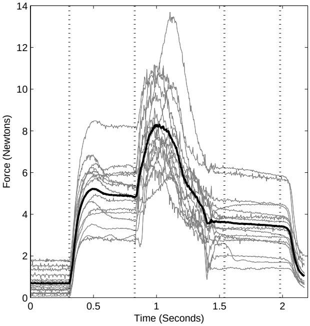
- The GP approach to UQ has been widely used & successful but has some issues
- Off the shelf generic covariance functions can lead to overly erratic & unstructured deviations from $\hat{g}$.
- Would be appealing for the realizations of the unknown $\mu$ to inherit some characteristics of the mechanistic model
- For example, the realizations may resemble realizations of SDEs/SPDEs or solutions of ODEs/PDEs
- May also have a multiscale/non-smooth character
- Broadly *mechanistic nonparametric Bayes* models can be designed for these problems
- Literature is under-developed- will give a simple case study here

# Mechanistic GPs - muscle activation example



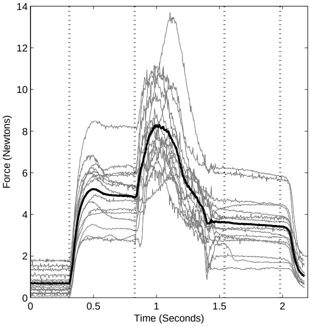

- Rich literature collecting data & modeling muscle contractions

# Mechanistic GPs - muscle activation example



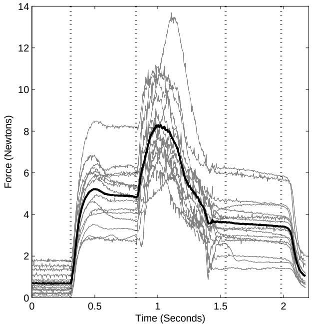

- Rich literature collecting data & modeling muscle contractions
- For each subject under each condition, a force tracing curve is collected

# Mechanistic GPs - muscle activation example




- Rich literature collecting data & modeling muscle contractions
- For each subject under each condition, a force tracing curve is collected
- Observed function $h(t) = Q(t)F(t)$ is product of isometric & stretch shortening components defined by ODEs

# Mechanistic GPs - muscle activation example




- Rich literature collecting data & modeling muscle contractions
- For each subject under each condition, a force tracing curve is collected
- Observed function $h(t) = Q(t)F(t)$ is product of isometric & stretch shortening components defined by ODEs
- Solutions to ODEs would need to be specific to each replicate & do not fit observed data perfectly

# Mechanistic Gaussian Processes

- Nonparametric hierarchical model - mechanistic knowledge through ODEs; allow bias, UQ & systematic & random deviations among subjects

# Mechanistic Gaussian Processes

- Nonparametric hierarchical model - mechanistic knowledge through ODEs; allow bias, UQ & systematic & random deviations among subjects
- Mechanistic information is expressed via linear ODE

$$Lh(t) = \frac{d^m h(t)}{dt^m} + a_{m-1}(t)\frac{d^{m-1}h(t)}{dt^{m-1}} + \ldots a_1(t)\frac{dh(t)}{dt} + a_0(t)h(t),$$

$= r(t)$; $\{a_0(t), \ldots, a_{m-1}(t)\}$ = known non-zero functions.

# Mechanistic Gaussian Processes

- Nonparametric hierarchical model - mechanistic knowledge through ODEs; allow bias, UQ & systematic & random deviations among subjects
- Mechanistic information is expressed via linear ODE

$$Lh(t) = \frac{d^m h(t)}{dt^m} + a_{m-1}(t)\frac{d^{m-1}h(t)}{dt^{m-1}} + \dots a_1(t)\frac{dh(t)}{dt} + a_0(t)h(t),$$

$= r(t)$; $\{a_0(t), \dots, a_{m-1}(t)\}$ = known non-zero functions.

- The solution exists & can be expressed as

$$h(t) = \int_{t_0}^{t} G(t,\xi) r(\xi) d\xi,$$

$G(t,\xi)$=Green's function

# Mechanistic Gaussian Processes

- Nonparametric hierarchical model - mechanistic knowledge through ODEs; allow bias, UQ & systematic & random deviations among subjects
- Mechanistic information is expressed via linear ODE

$$Lh(t) = \frac{d^m h(t)}{dt^m} + a_{m-1}(t)\frac{d^{m-1}h(t)}{dt^{m-1}} + \ldots a_1(t)\frac{dh(t)}{dt} + a_0(t)h(t),$$

$= r(t)$; $\{a_0(t), \ldots, a_{m-1}(t)\}$ = known non-zero functions.

- The solution exists & can be expressed as

$$h(t) = \int_{t_0}^{t} G(t,\xi)r(\xi)d\xi,$$

$G(t,\xi)$=Green's function

- As the integral operator is linear, if $r(t) \sim \mathrm{GP}(0, c)$ then $h(t)$ is also a GP

# Mechanistic Gaussian Processes

- Nonparametric hierarchical model - mechanistic knowledge through ODEs; allow bias, UQ & systematic & random deviations among subjects
- Mechanistic information is expressed via linear ODE

$$Lh(t) = \frac{d^m h(t)}{dt^m} + a_{m-1}(t)\frac{d^{m-1}h(t)}{dt^{m-1}} + \dots a_1(t)\frac{dh(t)}{dt} + a_0(t)h(t),$$

$= r(t)$; $\{a_0(t), \dots, a_{m-1}(t)\}$ = known non-zero functions.

- The solution exists & can be expressed as

$$h(t) = \int_{t_0}^{t} G(t,\xi)r(\xi)d\xi,$$

$G(t,\xi)$=Green's function

- As the integral operator is linear, if $r(t) \sim \mathrm{GP}(0, c)$ then $h(t)$ is also a GP
- Covariance kernel of induced GP is obtained by the convolution of Green's function for the ODE & the covariance kernel of $r(t)$.

# Implementation & hierarchical extension

- Exact solution to resulting covariance matrix is very ill-conditioned

# Implementation & hierarchical extension

- Exact solution to resulting covariance matrix is very ill-conditioned
- Rely on a Runge-Kutta (RK) method to approximate the ODE solution

# Implementation & hierarchical extension

- Exact solution to resulting covariance matrix is very ill-conditioned
- Rely on a Runge-Kutta (RK) method to approximate the ODE solution
- Use an Euler-Cauchy second order approximation; higher order RK also possible

# Implementation & hierarchical extension

- Exact solution to resulting covariance matrix is very ill-conditioned
- Rely on a Runge-Kutta (RK) method to approximate the ODE solution
- Use an Euler-Cauchy second order approximation; higher order RK also possible
- We obtain a simple posterior sampling algorithm & applied to the muscle force data

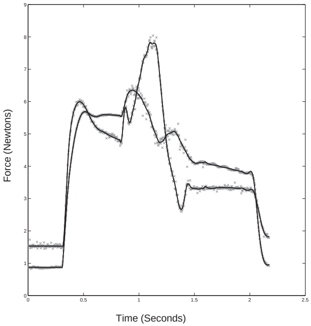
# Implementation & hierarchical extension

- Exact solution to resulting covariance matrix is very ill-conditioned
- Rely on a Runge-Kutta (RK) method to approximate the ODE solution
- Use an Euler-Cauchy second order approximation; higher order RK also possible
- We obtain a simple posterior sampling algorithm & applied to the muscle force data
- To model differences among subjects & across experimental groups (pre/post exercise protocol), we use a hierarchical GP

# Implementation & hierarchical extension

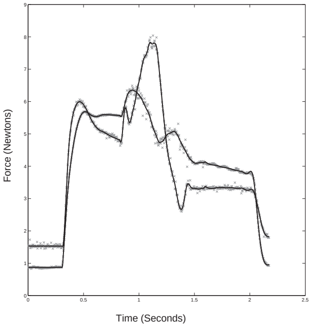
- Exact solution to resulting covariance matrix is very ill-conditioned
- Rely on a Runge-Kutta (RK) method to approximate the ODE solution
- Use an Euler-Cauchy second order approximation; higher order RK also possible
- We obtain a simple posterior sampling algorithm & applied to the muscle force data
- To model differences among subjects & across experimental groups (pre/post exercise protocol), we use a hierarchical GP
- Sample individual curves from GPs with mean curve specific to each group.

# Muscle force application



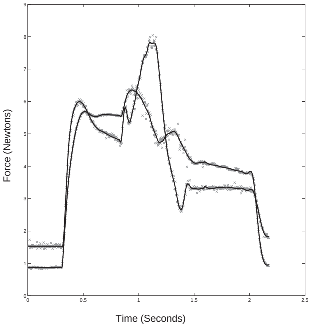

- Analyzed effect of repetitive muscle contractions on muscle force

# Muscle force application



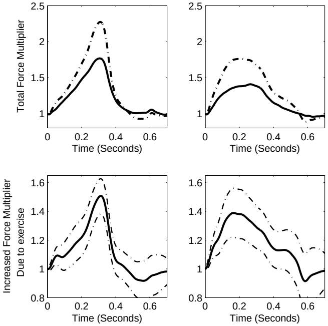

- Analyzed effect of repetitive muscle contractions on muscle force
- Data on 13 sessions for 15 young & 27 old rats, with 565 observations per session

# Muscle force application



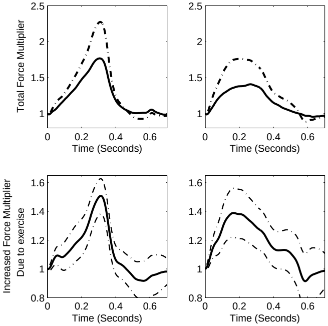

- Analyzed effect of repetitive muscle contractions on muscle force
- Data on 13 sessions for 15 young & 27 old rats, with 565 observations per session
- The above figure shows our model fits for one animal pre- and post-

# Some results



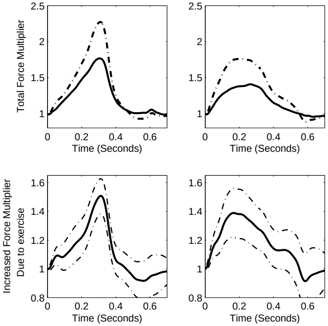

- Bayesian mechanistic hierarchical model can produce uncertainty estimates in any functional of interest

# Some results



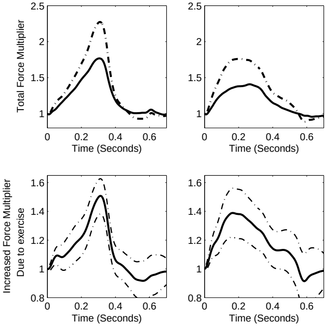

- Bayesian mechanistic hierarchical model can produce uncertainty estimates in any functional of interest
- Top row: mean isometric contraction for pre- (dashed) & post- (solid) exercise in old (top left) & young (top right)

# Some results



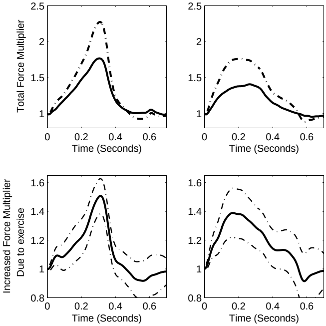

- Bayesian mechanistic hierarchical model can produce uncertainty estimates in any functional of interest
- Top row: mean isometric contraction for pre- (dashed) & post- (solid) exercise in old (top left) & young (top right)
- Figure bottom row = differences & 95% intervals are shown in the bottom row (old left & young right)

# Some results




- Bayesian mechanistic hierarchical model can produce uncertainty estimates in any functional of interest
- Top row: mean isometric contraction for pre- (dashed) & post- (solid) exercise in old (top left) & young (top right)
- Figure bottom row = differences & 95% intervals are shown in the bottom row (old left & young right)
- Increased muscle performance after exercise in young & old

# Some results




- Bayesian mechanistic hierarchical model can produce uncertainty estimates in any functional of interest
- Top row: mean isometric contraction for pre- (dashed) & post- (solid) exercise in old (top left) & young (top right)
- Figure bottom row = differences & 95% intervals are shown in the bottom row (old left & young right)
- Increased muscle performance after exercise in young & old
- Can also do inferences on individual differences

# Summary & discussion

- There is great potential for hybrid approaches combining Bayesian nonparametric models & mechanistic models in applied math

# Summary & discussion

- There is great potential for hybrid approaches combining Bayesian nonparametric models & mechanistic models in applied math
- Illustrated in a simple application to studying muscle contractions but similar approaches can be developed much more broadly

# Summary & discussion

- There is great potential for hybrid approaches combining Bayesian nonparametric models & mechanistic models in applied math
- Illustrated in a simple application to studying muscle contractions but similar approaches can be developed much more broadly
- Bayes approach appealing for not just UQ but also for allow modeling of hierarchical structure & statistical inferences

# Summary & discussion

- There is great potential for hybrid approaches combining Bayesian nonparametric models & mechanistic models in applied math
- Illustrated in a simple application to studying muscle contractions but similar approaches can be developed much more broadly
- Bayes approach appealing for not just UQ but also for allow modeling of hierarchical structure & statistical inferences
- Appealing to consider more complex mechanistic models (e.g. PDEs) & nonparametric Bayes models other than Gaussian processes

# Summary & discussion

- There is great potential for hybrid approaches combining Bayesian nonparametric models & mechanistic models in applied math
- Illustrated in a simple application to studying muscle contractions but similar approaches can be developed much more broadly
- Bayes approach appealing for not just UQ but also for allow modeling of hierarchical structure & statistical inferences
- Appealing to consider more complex mechanistic models (e.g. PDEs) & nonparametric Bayes models other than Gaussian processes
- However, GPs are remarkably flexible and can incorporate quite rich dynamics including multiscale structure

# Facilitating implementation - STAN

- In implementing the mechanistic hierarchical GP, we developed our own MCMC sampling algorithm & coded it ourselves

# Facilitating implementation - STAN

- In implementing the mechanistic hierarchical GP, we developed our own MCMC sampling algorithm & coded it ourselves
- The software package STAN provides a black box for efficient posterior sampling in a broad variety of Bayesian models

# Facilitating implementation - STAN

- In implementing the mechanistic hierarchical GP, we developed our own MCMC sampling algorithm & coded it ourselves
- The software package STAN provides a black box for efficient posterior sampling in a broad variety of Bayesian models
- There is an effort by the STAN development team to accommodate mechanistic models

# Facilitating implementation - STAN

- In implementing the mechanistic hierarchical GP, we developed our own MCMC sampling algorithm & coded it ourselves
- The software package STAN provides a black box for efficient posterior sampling in a broad variety of Bayesian models
- There is an effort by the STAN development team to accommodate mechanistic models
- One of the STAN developers (Michael Betancourt) is here & interested in helping facilitate implementation