Bayesian Methods for System Identification

Optimisation and Machine Learning in Process Systems Engineering

Tom Savage [1]

[1]University of Cambridge

29th September 2020

Table of Contents

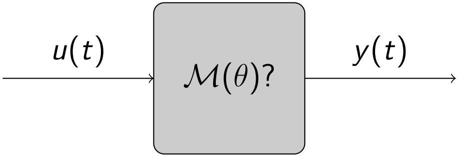
Table of Contents

Problem Statement

Given an input $u(t)$ and output $y(t)$ estimate the model $\mathcal{M}(\theta)$ and associated parameters θ.

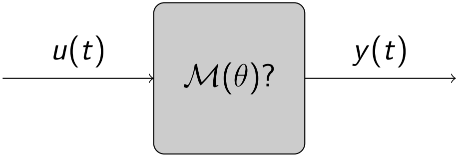
Problem Statement

Given an input $u(t)$ and output $y(t)$ estimate the model $\mathcal{M}(\theta)$ and associated parameters θ.



Problem Statement

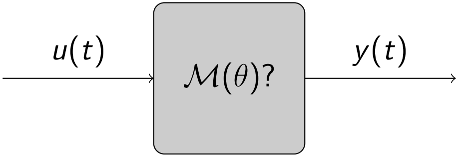
Given an input $u(t)$ and output $y(t)$ estimate the model $\mathcal{M}(\theta)$ and associated parameters θ.



$$y(t) = \overbrace{\theta_1 u(t) + \theta_2}^{\text{Linear model}}$$

Problem Statement

Given an input $u(t)$ and output $y(t)$ estimate the model $\mathcal{M}(\theta)$ and associated parameters θ.



$$y(t) = \overbrace{\theta_1 u(t) + \theta_2}^{\text{Linear model}}$$

$$y(t) = \underbrace{\theta_1 u(t)^n + \theta_2 u(t)^{n-1} + \ldots + \theta_n}_{\text{n-th degree polynomial model}}$$

System Identification

Finite Impulse Response (FIR) Model

System Identification

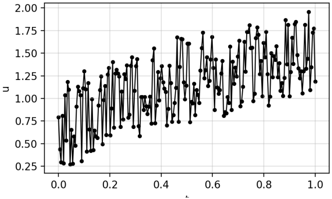
Finite Impulse Response (FIR) Model

Given a time series of inputs $u_0, u_1, u_2, \ldots, u_k$ and outputs $y_0, y_1, y_2, \ldots, y_{k-1}$, how to predict y_k?

System Identification

Finite Impulse Response (FIR) Model

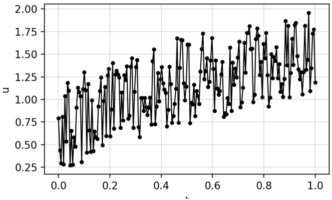
Given a time series of inputs $u_0, u_1, u_2, \dots, u_k$ and outputs $y_0, y_1, y_2, \dots, y_{k-1}$, how to predict y_k?

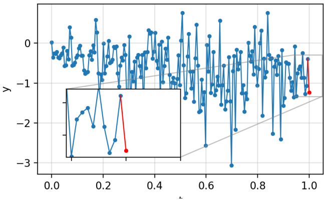


System Identification

Finite Impulse Response (FIR) Model

Given a time series of inputs $u_0, u_1, u_2, \ldots, u_k$ and outputs $y_0, y_1, y_2, \ldots, y_{k-1}$, how to predict y_k?

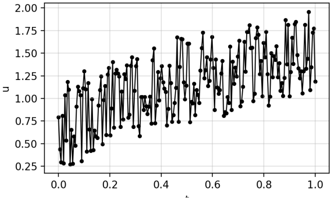


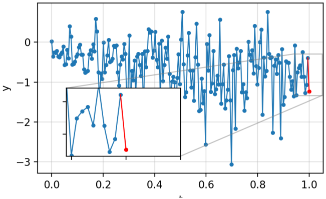


System Identification

Finite Impulse Response (FIR) Model

Given a time series of inputs $u_0, u_1, u_2, \ldots, u_k$ and outputs $y_0, y_1, y_2, \ldots, y_{k-1}$, how to predict y_k?



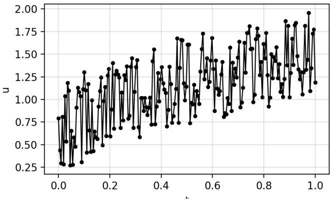


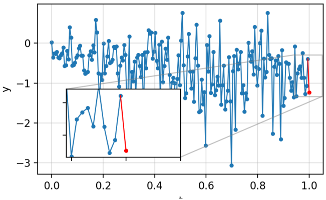
$$y_k = \theta_0 u_k + \theta_1 u_{k-1} + \ldots + \theta_K u_{n-N} = \sum_{i=0}^{N} \theta_i \cdot u_{n-i}$$

System Identification

Finite Impulse Response (FIR) Model

Given a time series of inputs $u_0, u_1, u_2, \ldots, u_k$ and outputs $y_0, y_1, y_2, \ldots, y_{k-1}$, how to predict y_k?





$$y_k = \theta_0 u_k + \theta_1 u_{k-1} + \ldots + \theta_K u_{n-N} = \sum_{i=0}^{N} \theta_i \cdot u_{n-i}$$

Since the order K is finite, response is bounded and stable for a bounded input: bounded input, bounded output (BIBO).

System Identification

Impulse Signal (Dirac delta function)

System Identification

Impulse Signal (Dirac delta function)

$$\delta(t) = \begin{cases} 1, & \text{if } t = 0 \\ 0, & \text{otherwise} \end{cases}$$

System Identification

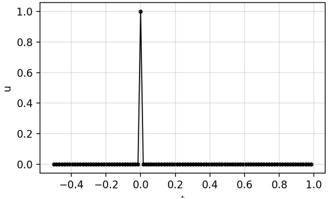
Impulse Signal (Dirac delta function)

$$\delta(t) = \begin{cases} 1, & \text{if } t = 0 \\ 0, & \text{otherwise} \end{cases} \text{ i.e. } \mathbf{u} = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \end{bmatrix}$$

System Identification

Impulse Signal (Dirac delta function)

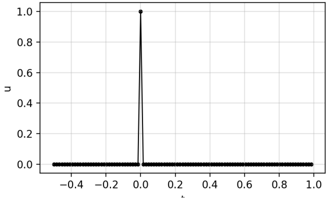
$$\delta(t) = \begin{cases} 1, & \text{if } t = 0 \\ 0, & \text{otherwise} \end{cases} \text{ i.e. } \mathbf{u} = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \end{bmatrix}$$



System Identification

Impulse Signal (Dirac delta function)

$$\delta(t) = \begin{cases} 1, & \text{if } t = 0 \\ 0, & \text{otherwise} \end{cases} \quad \text{i.e. } \mathbf{u} = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \end{bmatrix}$$



Using our previous FIR model, consider the moving horizon of K time-steps it is clear that eventually the output y will eventually become 0 as $t > K$, hence finite.

System Identification

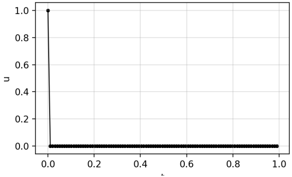
Impulse Signal (Dirac delta function)

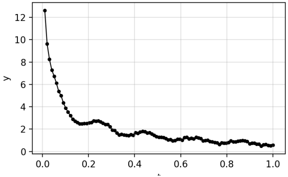
What does the impulse response look like? For a stable system...

System Identification

Impulse Signal (Dirac delta function)

What does the impulse response look like? For a stable system...

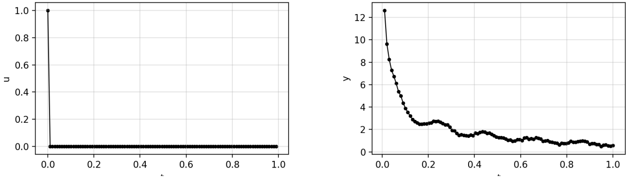




System Identification

Impulse Signal (Dirac delta function)

What does the impulse response look like? For a stable system...

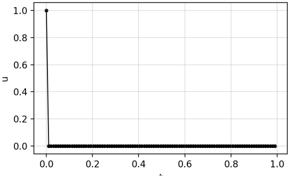


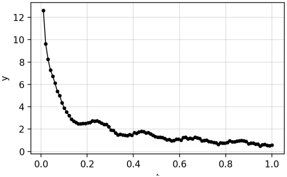
$$y_k = \theta_0 u_k + \theta_1 u_{k-1} + \ldots + \theta_K u_{n-N}$$

System Identification

Impulse Signal (Dirac delta function)

What does the impulse response look like? For a stable system...





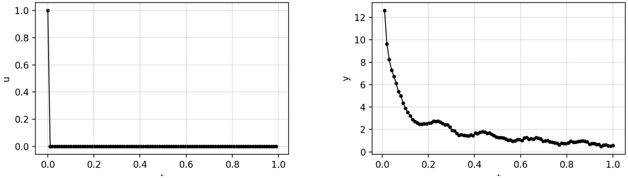
$$y_k = \theta_0 u_k + \theta_1 u_{k-1} + \ldots + \theta_K u_{n-N}$$

$$y_K = \theta_0 \cdot 0 + \theta_1 \cdot 0 + \ldots + \theta_{K-1} \cdot 0 + \theta_K \cdot 1$$

System Identification

Impulse Signal (Dirac delta function)

What does the impulse response look like? For a stable system...



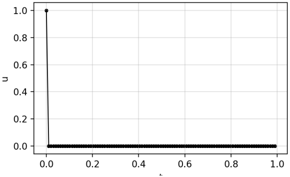
$$y_k = \theta_0 u_k + \theta_1 u_{k-1} + \ldots + \theta_K u_{n-N}$$

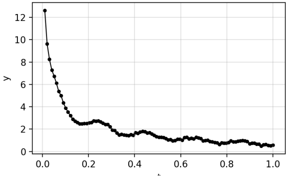
$$y_{K-1} = \theta_0 \cdot 0 + \theta_1 \cdot 0 + \ldots + \theta_{K-1} \cdot 1 + \theta_K \cdot 0$$

System Identification

Impulse Signal (Dirac delta function)

What does the impulse response look like? For a stable system...





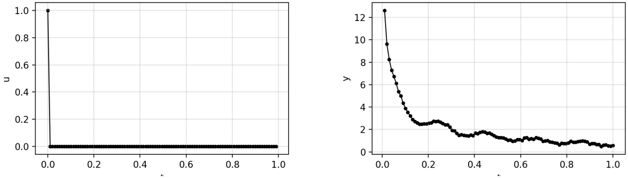
$$y_k = \theta_0 u_k + \theta_1 u_{k-1} + \ldots + \theta_K u_{n-N}$$

$$y_1 = \theta_0 \cdot 0 + \theta_1 \cdot 1 + \ldots + \theta_{K-1} \cdot 0 + \theta_K \cdot 0$$

System Identification

Impulse Signal (Dirac delta function)

What does the impulse response look like? For a stable system...



$$y_k = \theta_0 u_k + \theta_1 u_{k-1} + \ldots + \theta_K u_{n-N}$$

$$y_0 = \theta_0 \cdot 1 + \theta_1 \cdot 0 + \ldots + \theta_{K-1} \cdot 0 + \theta_K \cdot 0$$

System Identification

$$y_k = \theta_0 u_k + \theta_1 u_{k-1} + \ldots + \theta_K u_{n-N}$$

System Identification

$$y_k = \theta_0 u_k + \theta_1 u_{k-1} + ... + \theta_K u_{n-N}$$

- The parameters θ of our FIR model can be seen as the outputs of the system if subjected to an impulse signal $\rightarrow$ predicting the impulse response.

System Identification

$$y_k = \theta_0 u_k + \theta_1 u_{k-1} + ... + \theta_K u_{n-N}$$

- The parameters θ of our FIR model can be seen as the outputs of the system if subjected to an impulse signal $\rightarrow$ predicting the impulse response.
- Useful as if you wanted to find out model parameters you just have to 'kick' the system and observe the response.

System Identification

$$y_k = \theta_0 u_k + \theta_1 u_{k-1} + \ldots + \theta_K u_{n-N}$$

- The parameters θ of our FIR model can be seen as the outputs of the system if subjected to an impulse signal $\rightarrow$ predicting the impulse response.
- Useful as if you wanted to find out model parameters you just have to 'kick' the system and observe the response.
- What if we can't 'kick' the system?

Table of Contents

Linear Regression

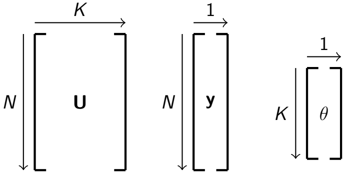
N observations, K inputs:
Given input matrix $\mathbf{U} \in \mathbb{R}^{N \times K}$ and outputs $\mathbf{y} \in \mathbb{R}^{N \times 1}$ along with a model $\mathcal{M}$ in the form $y = \mathbf{u}\theta^T$ where $\theta \in \mathbb{R}^{K \times 1}$:

Linear Regression

N observations, K inputs:
Given input matrix $\mathbf{U} \in \mathbb{R}^{N \times K}$ and outputs $\mathbf{y} \in \mathbb{R}^{N \times 1}$ along with a model $\mathcal{M}$ in the form $y = \mathbf{u}\theta^T$ where $\theta \in \mathbb{R}^{K \times 1}$: *Find* θ

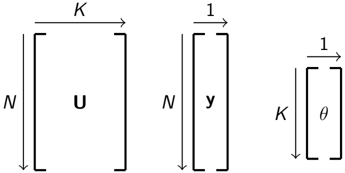
Linear Regression

N observations, K inputs:
Given input matrix $\mathbf{U} \in \mathbb{R}^{N \times K}$ and outputs $\mathbf{y} \in \mathbb{R}^{N \times 1}$ along with a model $\mathcal{M}$ in the form $y = \mathbf{u}\theta^T$ where $\theta \in \mathbb{R}^{K \times 1}$: *Find* θ



Linear Regression

N observations, K inputs:
Given input matrix $\mathbf{U} \in \mathbb{R}^{N \times K}$ and outputs $\mathbf{y} \in \mathbb{R}^{N \times 1}$ along with a model $\mathcal{M}$ in the form $y = \mathbf{u}\theta^T$ where $\theta \in \mathbb{R}^{K \times 1}$: *Find* θ

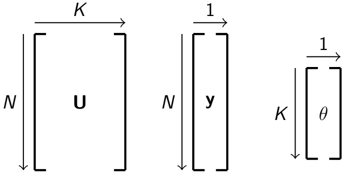


Formulating the problem as least squares...

$$\hat{\theta} = \arg\min_{\theta} ||\mathbf{y} - \mathbf{U}\theta||^2$$

Linear Regression

N observations, K inputs:
Given input matrix $\mathbf{U} \in \mathbb{R}^{N \times K}$ and outputs $\mathbf{y} \in \mathbb{R}^{N \times 1}$ along with a model $\mathcal{M}$ in the form $y = \mathbf{u}\theta^T$ where $\theta \in \mathbb{R}^{K \times 1}$: *Find* θ



Formulating the problem as least squares...

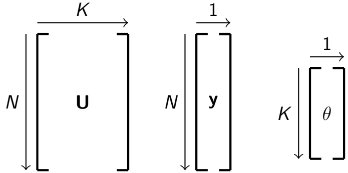
$$\hat{\theta} = \arg\min_{\theta} ||\mathbf{y} - \mathbf{U}\theta||^2$$

Analytical solution of:

$$\hat{\theta} = (\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{y}$$

Linear Regression

N observations, K inputs:
Given input matrix $\mathbf{U} \in \mathbb{R}^{N\times K}$ and outputs $\mathbf{y} \in \mathbb{R}^{N\times 1}$ along with a model $\mathcal{M}$ in the form $y = \mathbf{u}\theta^T$ where $\theta \in \mathbb{R}^{K\times 1}$: *Find* θ



Formulating the problem as least squares...

$$\hat{\theta} = \arg\min_{\theta} ||\mathbf{y} - \mathbf{U}\theta||^2$$

Analytical solution of:

$$\hat{\theta} = (\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{y}$$

What about bias vs variance? $\rightarrow$ Needs to be considered in linear regression.

Linear Regression

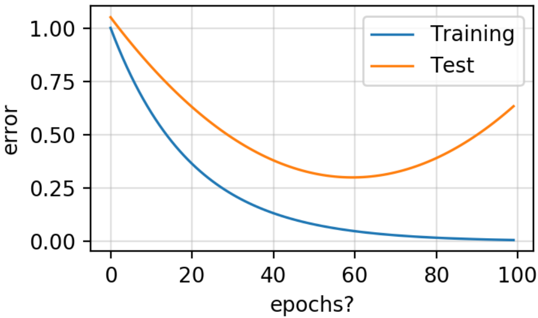
Trading off bias / variance.

For neural networks...

Linear Regression

Trading off bias / variance.

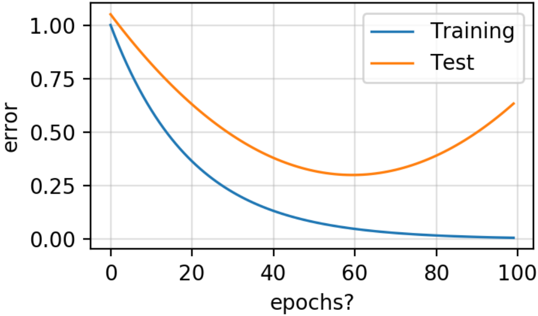
For neural networks...



Linear Regression

Trading off bias / variance.

For neural networks...



What is the equivalent of epochs for a model with analytical parameters?

Table of Contents

Ridge Regression

Regularisation technique

Ridge Regression

Regularisation technique

$$\hat{\theta} = \arg\min_{\theta} ||\mathbf{y} - \mathbf{U}\theta||^2$$

Ridge Regression

Regularisation technique

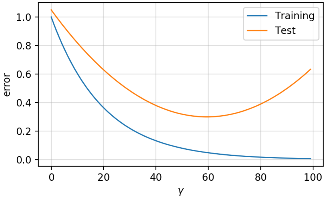
$$\hat{\theta} = \arg\min_{\theta} ||\mathbf{y} - \mathbf{U}\theta||^2 + \underbrace{\gamma||\theta||^2}_{\text{Ridge term}}$$

Ridge Regression

Regularisation technique

$$\hat{\theta} = \arg\min_{\theta} ||\mathbf{y} - \mathbf{U}\theta||^2 + \underbrace{\gamma||\theta||^2}_{\text{Ridge term}}$$

By varying γ, bias and variance can be traded off.



For any γ, analytical solution of $\hat{\theta} = (\mathbf{U}^T\mathbf{U} + \gamma\mathbf{I})^{-1}\mathbf{U}^T\mathbf{y}$

For more information visit `https://tom-savage.co.uk/Linear-Regression/`

Table of Contents

Extension to Gaussian processes

$$\underbrace{\gamma||\theta||^2 = \gamma\theta^T \mathbf{I}\theta}_{\text{Ridge Term}}$$

Extension to Gaussian processes

$$\underbrace{\gamma||\theta||^2 = \gamma\theta^T \mathbf{I}\theta}_{\text{Ridge Term}}$$

What if instead of the identity matrix we used some other matrix?

Extension to Gaussian processes

$$\underbrace{\gamma||\theta||^2 = \gamma\theta^T \mathbf{I}\theta}_{\text{Ridge Term}}$$

What if instead of the identity matrix we used some other matrix?

$$\arg\min_{\theta} ||\mathbf{y} - \mathbf{U}\theta||^2 + \gamma\theta^T \mathbf{I}\theta$$

Extension to Gaussian processes

$$\underbrace{\gamma||\theta||^2 = \gamma\theta^T \mathbf{I}\theta}_{\text{Ridge Term}}$$

What if instead of the identity matrix we used some other matrix?

$$\arg\min_{\theta} ||\mathbf{y} - \mathbf{U}\theta||^2 + \gamma\theta^T K^{-1}\theta$$

Extension to Gaussian processes

$$\underbrace{\gamma||\theta||^2 = \gamma\theta^T \mathbf{I}\theta}_{\text{Ridge Term}}$$

What if instead of the identity matrix we used some other matrix?

$$\arg\min_{\theta} ||\mathbf{y} - \mathbf{U}\theta||^2 + \gamma\theta^T K^{-1}\theta$$

This has an analytical solution of:

$$\hat{\theta} = (K\mathbf{U}^T\mathbf{U} + \gamma I_K)^{-1} K\mathbf{U}^T\mathbf{y}$$

Extension to Gaussian processes

$$\hat{\theta} = \underbrace{(K\mathbf{U}^T\mathbf{U} + \gamma I_K)^{-1}K\mathbf{U}^T\mathbf{y}}_{\text{regularised least squares estimate}}$$

Extension to Gaussian processes

$$\hat{\theta} = \underbrace{(K\mathbf{U}^T\mathbf{U} + \gamma I_K)^{-1} K\mathbf{U}^T\mathbf{y}}_{\text{regularised least squares estimate}}$$

For jointly Gaussian variables ($\theta \sim \mathcal{N}(0, K)$, $e_{\mathbf{y}} \sim \mathcal{N}(0, \sigma)$)

$$\begin{bmatrix} \theta \\ \mathbf{y} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K & K\mathbf{U}^T \\ \mathbf{U}K & \mathbf{U}K\mathbf{U}^T + \sigma^2 I_N \end{bmatrix} \right)$$

Extension to Gaussian processes

$$\hat{\theta} = \underbrace{(K\mathbf{U}^T\mathbf{U} + \gamma I_K)^{-1} K\mathbf{U}^T\mathbf{y}}_{\text{regularised least squares estimate}}$$

For jointly Gaussian variables ($\theta \sim \mathcal{N}(0, K)$, $e_{\mathbf{y}} \sim \mathcal{N}(0, \sigma)$)

$$\begin{bmatrix} \theta \\ \mathbf{y} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K & K\mathbf{U}^T \\ \mathbf{U}K & \mathbf{U}K\mathbf{U}^T + \sigma^2 I_N \end{bmatrix} \right)$$

The posterior distribution of θ given $\mathbf{y}$ is given as:

$$\theta | \mathbf{y} \sim \mathcal{N}(\hat{\theta}, K^*)$$

Extension to Gaussian processes

$$\hat{\theta} = \underbrace{(K\mathbf{U}^T\mathbf{U} + \gamma I_K)^{-1}K\mathbf{U}^T\mathbf{y}}_{\text{regularised least squares estimate}}$$

For jointly Gaussian variables ($\theta \sim \mathcal{N}(0, K)$, $e_{\mathbf{y}} \sim \mathcal{N}(0, \sigma)$)

$$\begin{bmatrix} \theta \\ \mathbf{y} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K & K\mathbf{U}^T \\ \mathbf{U}K & \mathbf{U}K\mathbf{U}^T + \sigma^2 I_N \end{bmatrix} \right)$$

The posterior distribution of θ given $\mathbf{y}$ is given as:

$$\theta | \mathbf{y} \sim \mathcal{N}(\hat{\theta}, K^*)$$

$$\hat{\theta} = \underbrace{(K\mathbf{U}^T\mathbf{U} + \sigma^2 I_N)^{-1}K\mathbf{U}^T\mathbf{y}}_{\text{Posterior distribution of } \theta}$$

Extension to Gaussian processes

$$\underbrace{(K\mathbf{U}^T\mathbf{U} + \gamma I_K)^{-1} K\mathbf{U}^T\mathbf{y}}_{\text{kernel-regularised least squares estimate}} \quad \underbrace{(K\mathbf{U}^T\mathbf{U} + \sigma^2 I_N)^{-1} K\mathbf{U}^T\mathbf{y}}_{\text{Posterior distribution of } \theta}$$

Extension to Gaussian processes

$$\underbrace{(K\mathbf{U}^T\mathbf{U} + \gamma I_K)^{-1} K\mathbf{U}^T\mathbf{y}}_{\text{kernel-regularised least squares estimate}} \quad \underbrace{(K\mathbf{U}^T\mathbf{U} + \sigma^2 I_N)^{-1} K\mathbf{U}^T\mathbf{y}}_{\text{Posterior distribution of } \theta}$$

They're equivalent! What does this mean?

Extension to Gaussian processes

$$\underbrace{(K\mathbf{U}^T\mathbf{U} + \gamma I_K)^{-1} K\mathbf{U}^T\mathbf{y}}_{\text{kernel-regularised least squares estimate}} \quad \underbrace{(K\mathbf{U}^T\mathbf{U} + \sigma^2 I_N)^{-1} K\mathbf{U}^T\mathbf{y}}_{\text{Posterior distribution of } \theta}$$

They're equivalent! What does this mean?

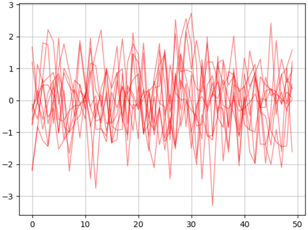
- Ridge regression and Gaussian processes are equivalent when the covariance matrix $= I$. e.g. when prior samples look like this...

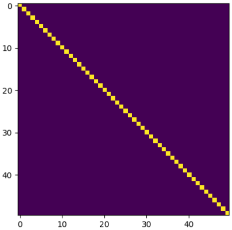
Extension to Gaussian processes

$$\underbrace{(K\mathbf{U}^T\mathbf{U} + \gamma I_K)^{-1} K\mathbf{U}^T\mathbf{y}}_{\text{kernel-regularised least squares estimate}} \quad \underbrace{(K\mathbf{U}^T\mathbf{U} + \sigma^2 I_N)^{-1} K\mathbf{U}^T\mathbf{y}}_{\text{Posterior distribution of } \theta}$$

They're equivalent! What does this mean?

- Ridge regression and Gaussian processes are equivalent when the covariance matrix $= I$. e.g. when prior samples look like this...

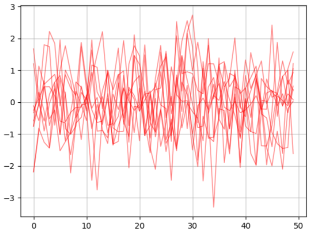


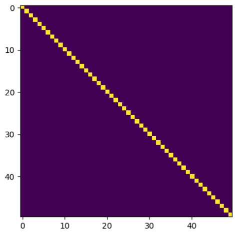


Extension to Gaussian processes

They're equivalent! What does this mean?

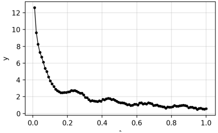
- Ridge regression and Gaussian processes are equivalent when the covariance matrix $= I$. e.g. when prior samples look like this...





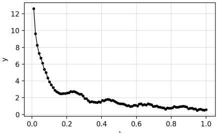
- Now we can look at this system identification problem from a Bayesian point of view; can we fully embrace the framework and incorperate prior knowledge into our prediction of $\hat{\theta}$?

Extension to Gaussian processes

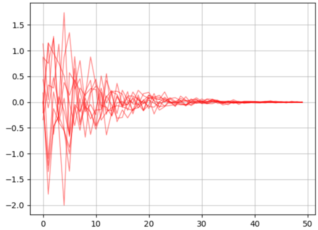


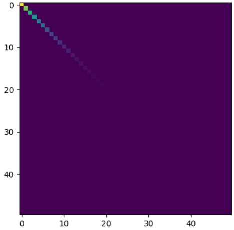
Instead of doing Ridge regression (i.e. using the identity prior) or some other covariance function (i.e. squared exponential) what about if we let it decay exponentially?

Extension to Gaussian processes

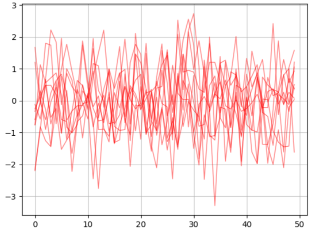


Instead of doing Ridge regression (i.e. using the identity prior) or some other covariance function (i.e. squared exponential) what about if we let it decay exponentially?

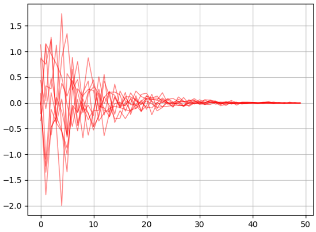




Extension to Gaussian processes

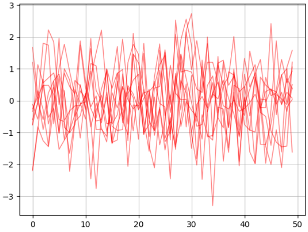


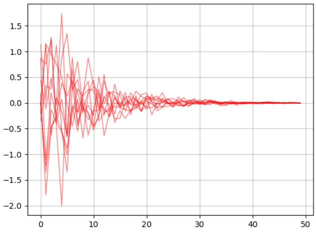
$$K_{ij} = \delta_{ij}$$



$$K_{ij} = \alpha^i \delta_{ij}$$

Extension to Gaussian processes

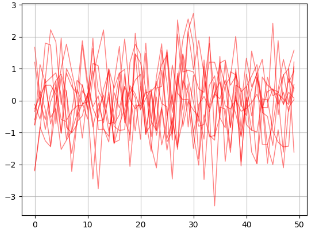


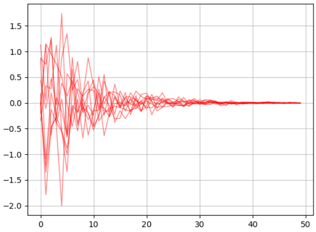


$$K_{ij} = \delta_{ij} \qquad\qquad K_{ij} = \alpha^i \delta_{ij}$$

Remaining issue:
Each neighbouring value of θ, i.e. $\theta_1, \theta_2 \ldots$ is uncorrelated (off diagonal terms in the covariance matrix are all 0) $\rightarrow$ non-smooth.

Extension to Gaussian processes



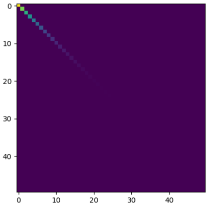


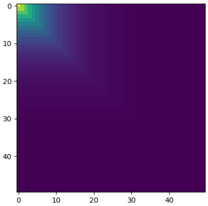
$$K_{ij} = \delta_{ij} \qquad\qquad K_{ij} = \alpha^i \delta_{ij}$$

Remaining issue:
Each neighbouring value of θ, i.e. $\theta_1, \theta_2 ...$ is uncorrelated (off diagonal terms in the covariance matrix are all 0) $\rightarrow$ non-smooth.
This is a false assumption!

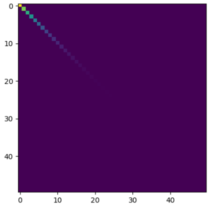
Table of Contents

Stable spline kernel

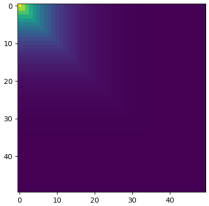




Stable spline kernel

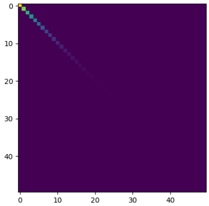


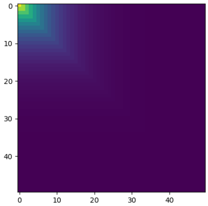
$$K_{ij} = \alpha^{i}\delta_{ij}$$



$$K_{ij} = \alpha^{\max(i,j)}$$

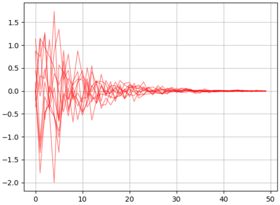
Stable spline kernel

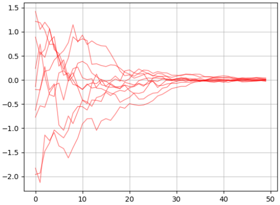




$$K_{ij} = \alpha^i \delta_{ij}$$

$$K_{ij} = \alpha^{\max(i,j)}$$





Stable spline kernel

The stable spline kernel allows for prior information of the FIR parameters θ to be included within a Bayesian estimation framework.

Stable spline kernel

The stable spline kernel allows for prior information of the FIR parameters θ to be included within a Bayesian estimation framework.

- Stable (parameters decay to 0)

Stable spline kernel

The stable spline kernel allows for prior information of the FIR parameters θ to be included within a Bayesian estimation framework.

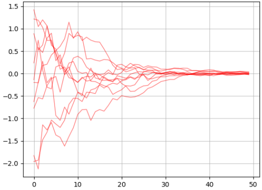
- Stable (parameters decay to 0)
- Smooth (neighbouring parameters are correlated)

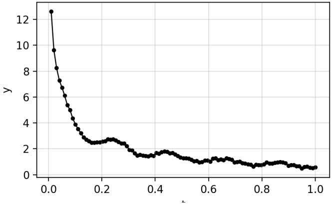
Stable spline kernel

The stable spline kernel allows for prior information of the FIR parameters θ to be included within a Bayesian estimation framework.

- Stable (parameters decay to 0)
- Smooth (neighbouring parameters are correlated)

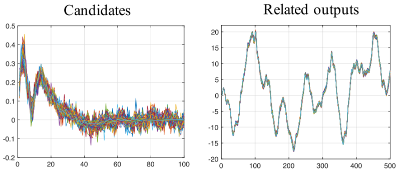
Can also extend the estimation of the parameter vector to continuous time. Continuous sytem identification $\rightarrow$ function approximation using an informative prior.

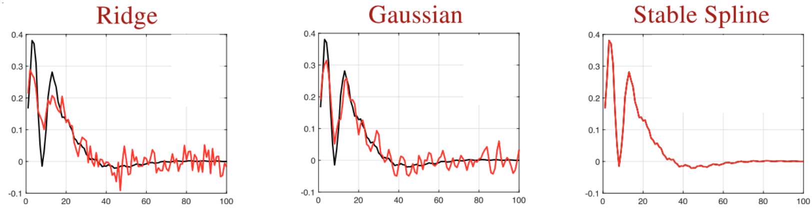




Results

100 different impulse responses generated, with a single 'true' response. Note how the predicted outputs with each impulse response varies by very little.





Conclusions

Conclusions

- System identification allows for a model to be constructed based on input and output data.

Conclusions

- System identification allows for a model to be constructed based on input and output data.
- Regularised linear regression and Gaussian processes inherently linked

Conclusions

- System identification allows for a model to be constructed based on input and output data.
- Regularised linear regression and Gaussian processes inherently linked
- Incorperating correct assumptions into prior distributions is extremely effective

Conclusions

- System identification allows for a model to be constructed based on input and output data.
- Regularised linear regression and Gaussian processes inherently linked
- Incorperating correct assumptions into prior distributions is extremely effective
- Continuous system identification enabled using GPs as a function approximator.

Thanks!

References

- Pillonetto, G., Dinuzzo, F., Chen, T., De Nicolao, G., & Ljung, L. (2014). Kernel methods in system identification, machine learning and function estimation: A survey. Automatica, 50(3), 657–682. https://doi.org/10.1016/j.automatica.2014.01.001
- Pillonetto, G. (n.d.). REGULARIZED KERNEL-BASED APPROACHES TO SYSTEM IDENTIFICATION. Retrieved from https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjWse28_Y3sAhUKSxUIHXVNCWOQFjABegQIARAB&url=https%3A%2F%2Fwww.kth.se%2Fsocial%2Ffiles%2F5b716a5b56be5bd565e07e4f%2Fgianluigi_pillonetto_sysid2018.pdf&usg=AOvVaw2uRQRyV2Tb9jo3lO4TO6zj