Gaussian Processes for Regression: Models, Algorithms, and Applications

Tamara Broderick

Associate Professor

MIT

http://www.tamarabroderick.com/tutorials.html

Why Gaussian processes (GPs)?

Why Gaussian processes (GPs)?

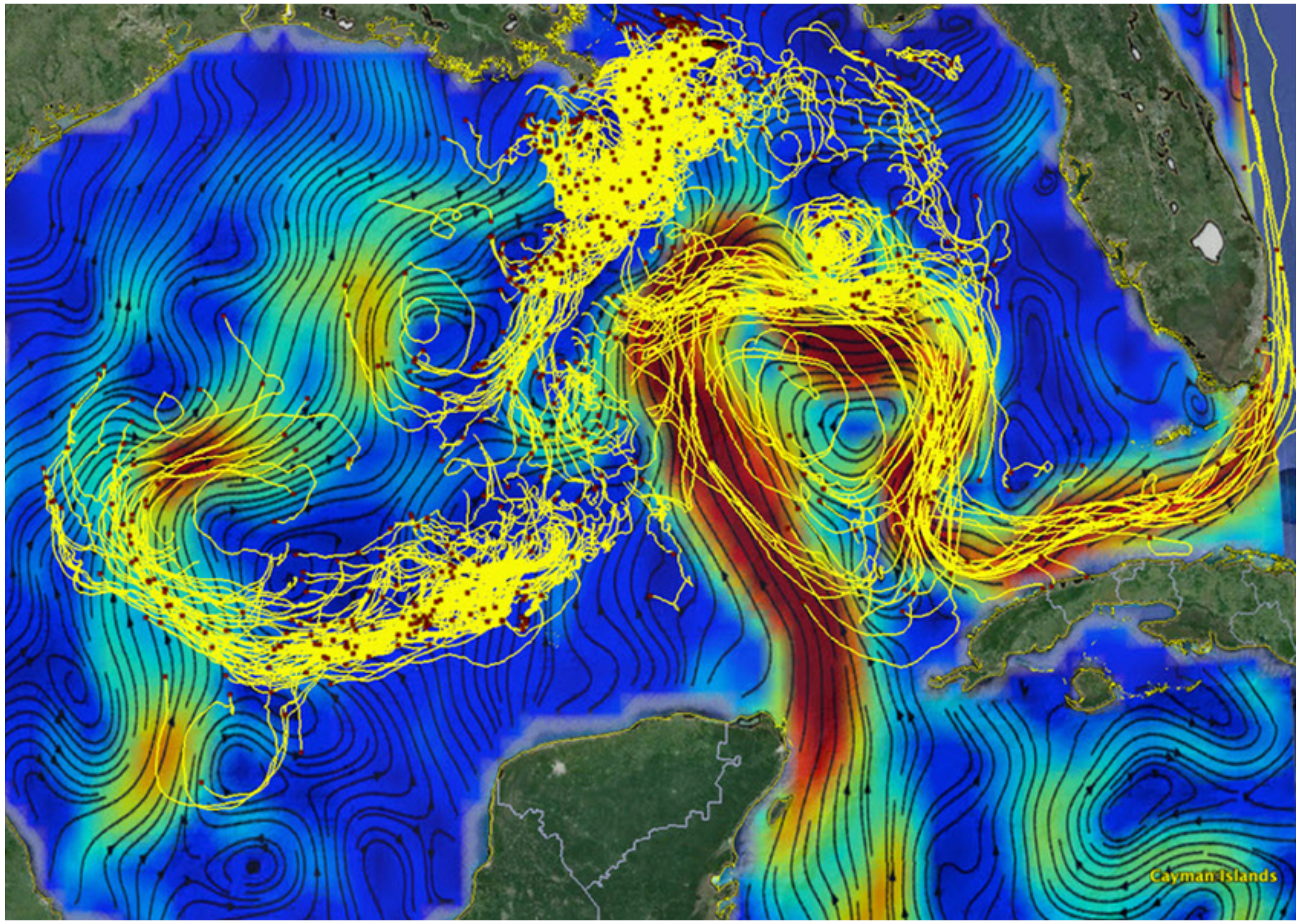
- Often want to estimate/“predict” some continuous outcome as a function of certain inputs (*regression*)

Why Gaussian processes (GPs)?

- Often want to estimate/“predict” some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems

Why Gaussian processes (GPs)?

- Often want to estimate/“predict” some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems

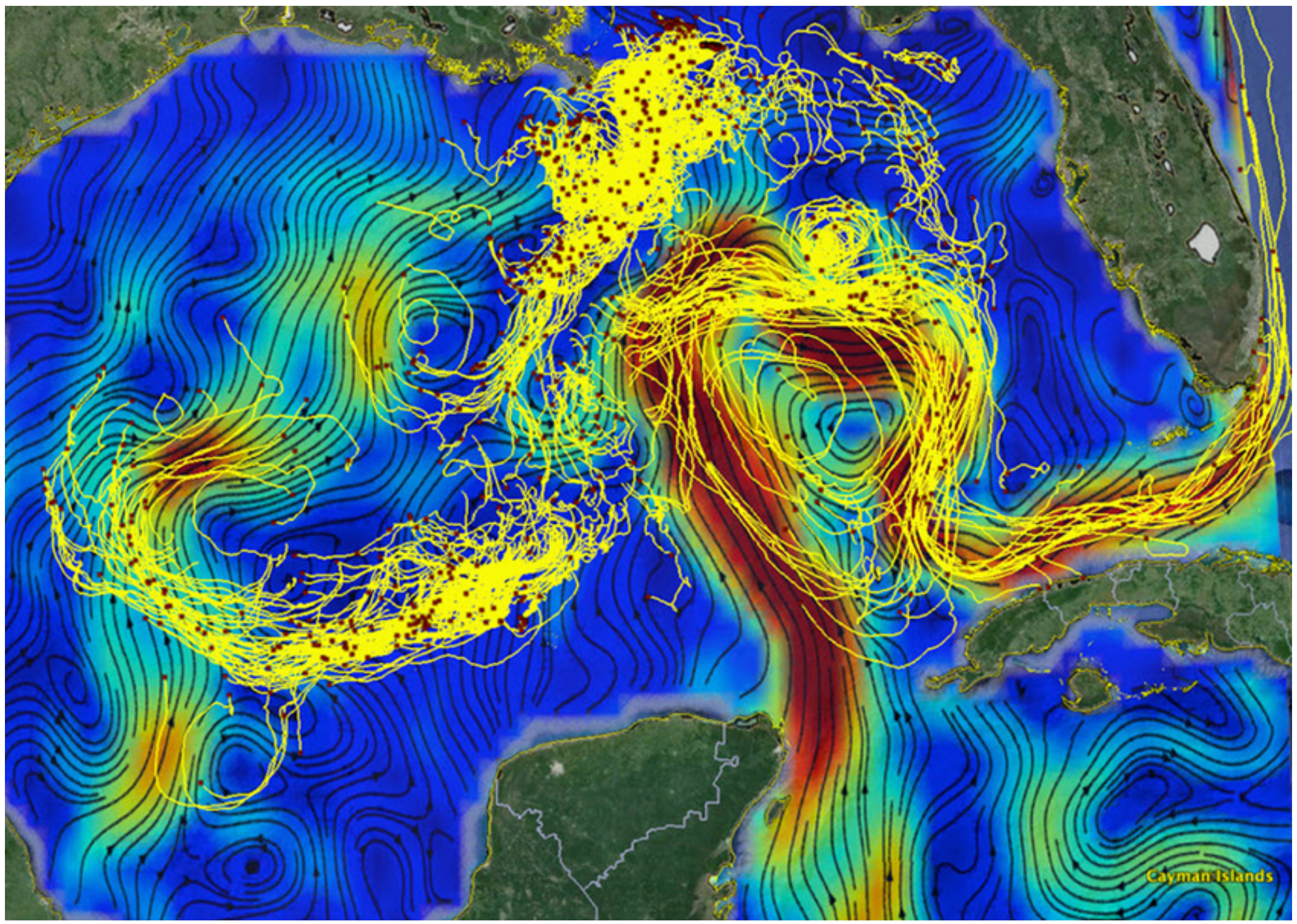


Example:

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Why Gaussian processes (GPs)?

- Often want to estimate/“predict” some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems



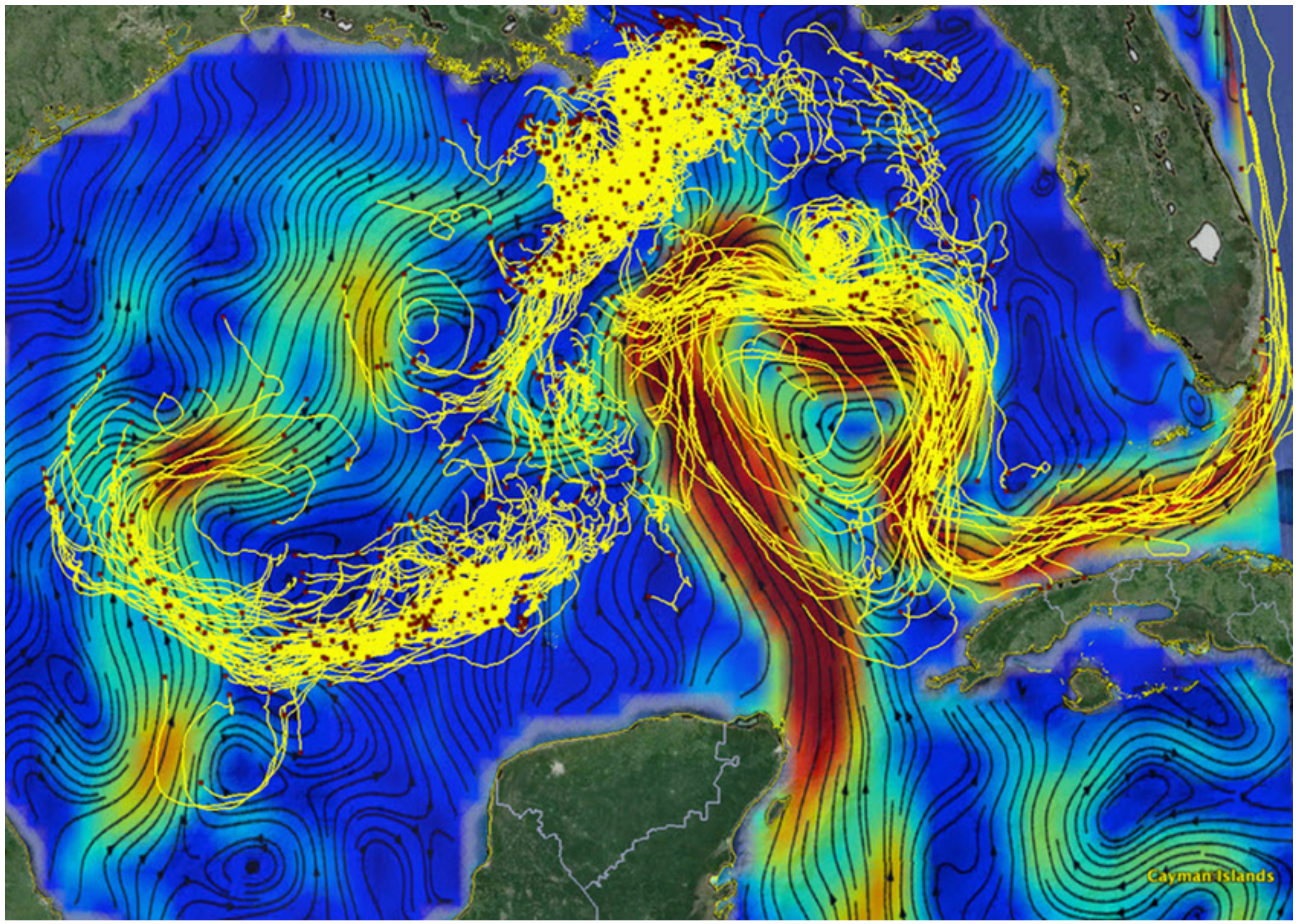
Example:

- The ocean current (velocity vector field) varies by space & time

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Why Gaussian processes (GPs)?

- Often want to estimate/“predict” some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems



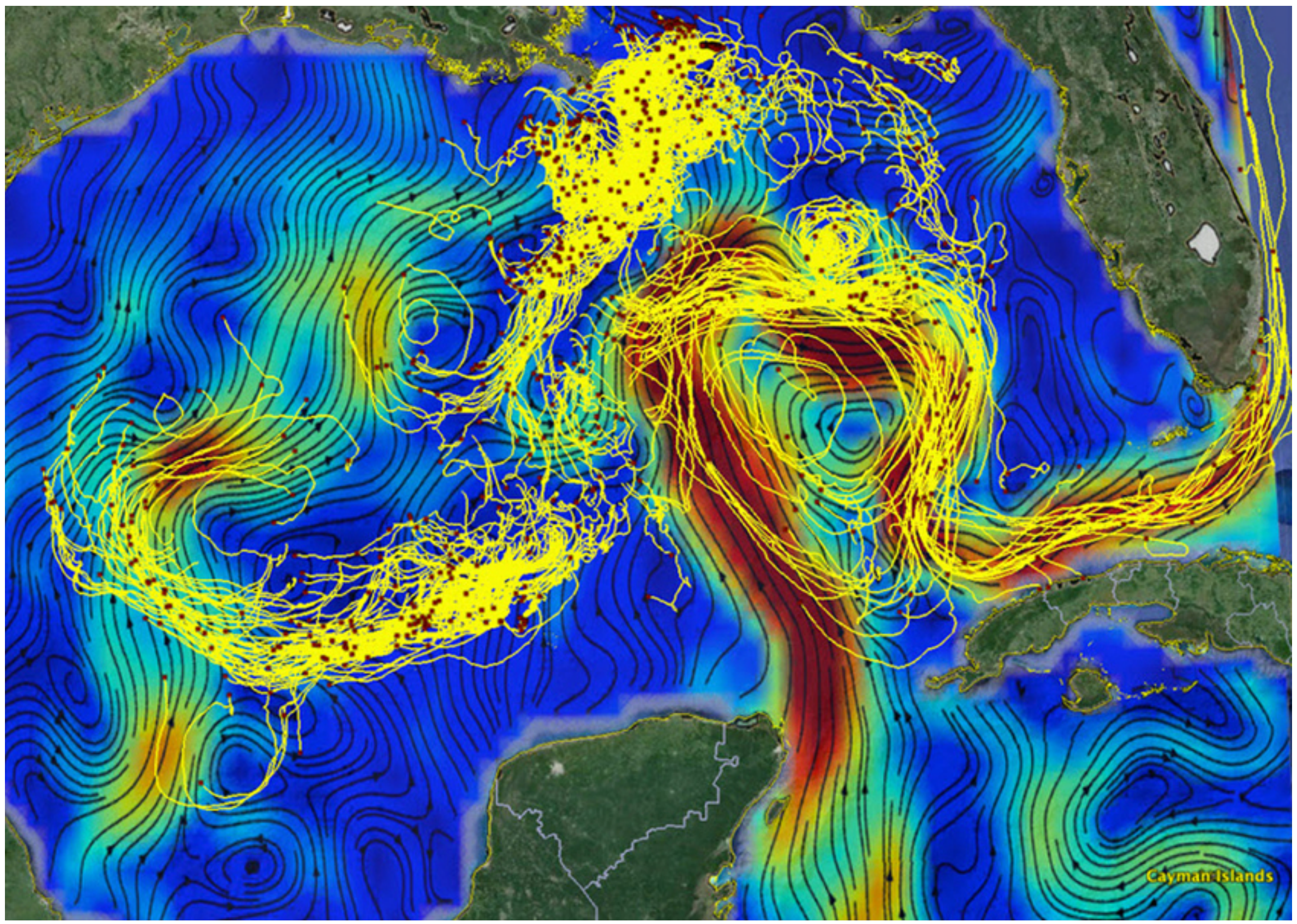
Example:

- The ocean current (velocity vector field) varies by space & time
- Scientists get sparse observations of the current from buoys

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Why Gaussian processes (GPs)?

- Often want to estimate/"predict" some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems

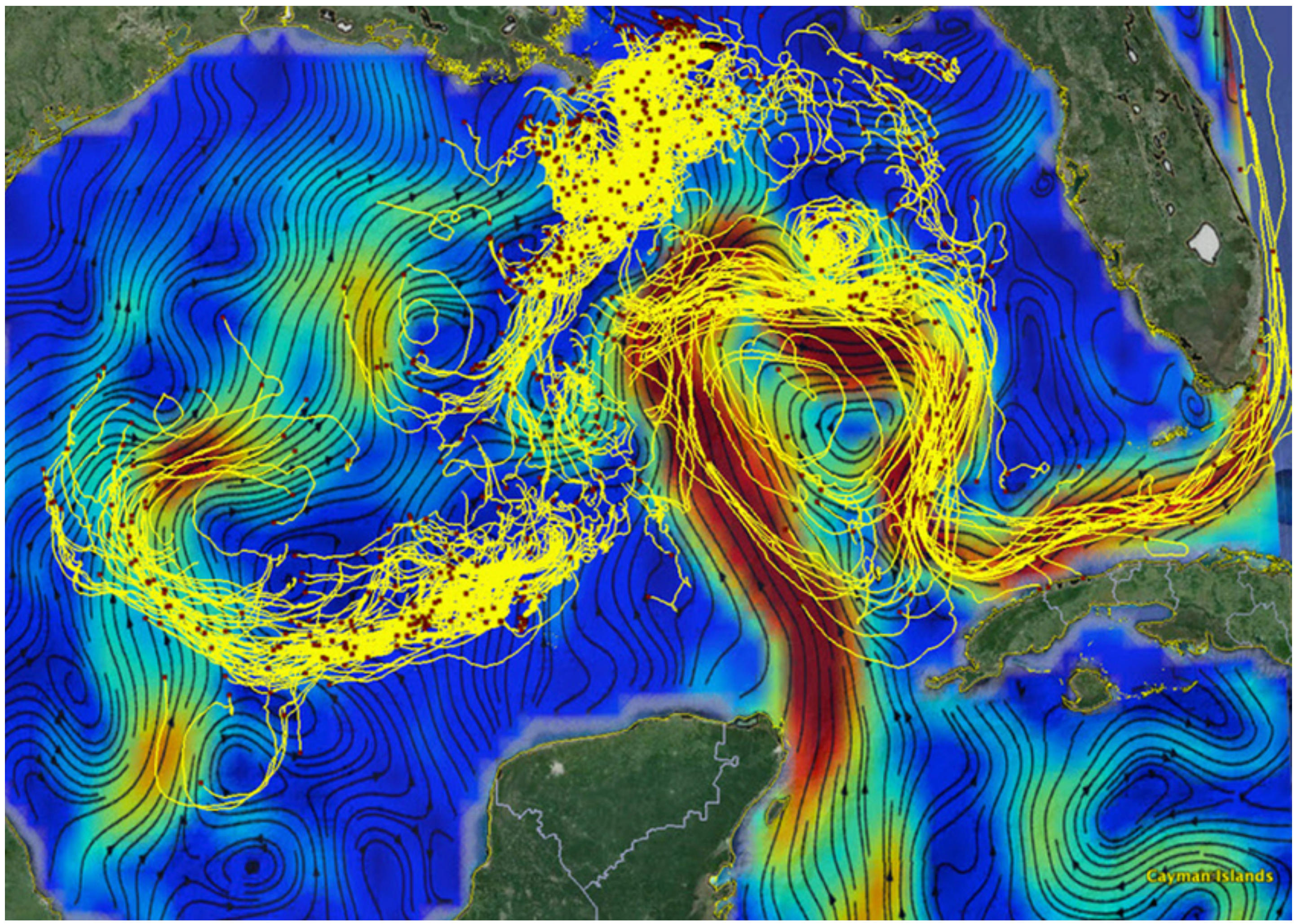


Example:
- The ocean current (velocity vector field) varies by space & time
- Scientists get sparse observations of the current from buoys
- Goal: estimate the current

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Why Gaussian processes (GPs)?

- Often want to estimate/“predict” some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems



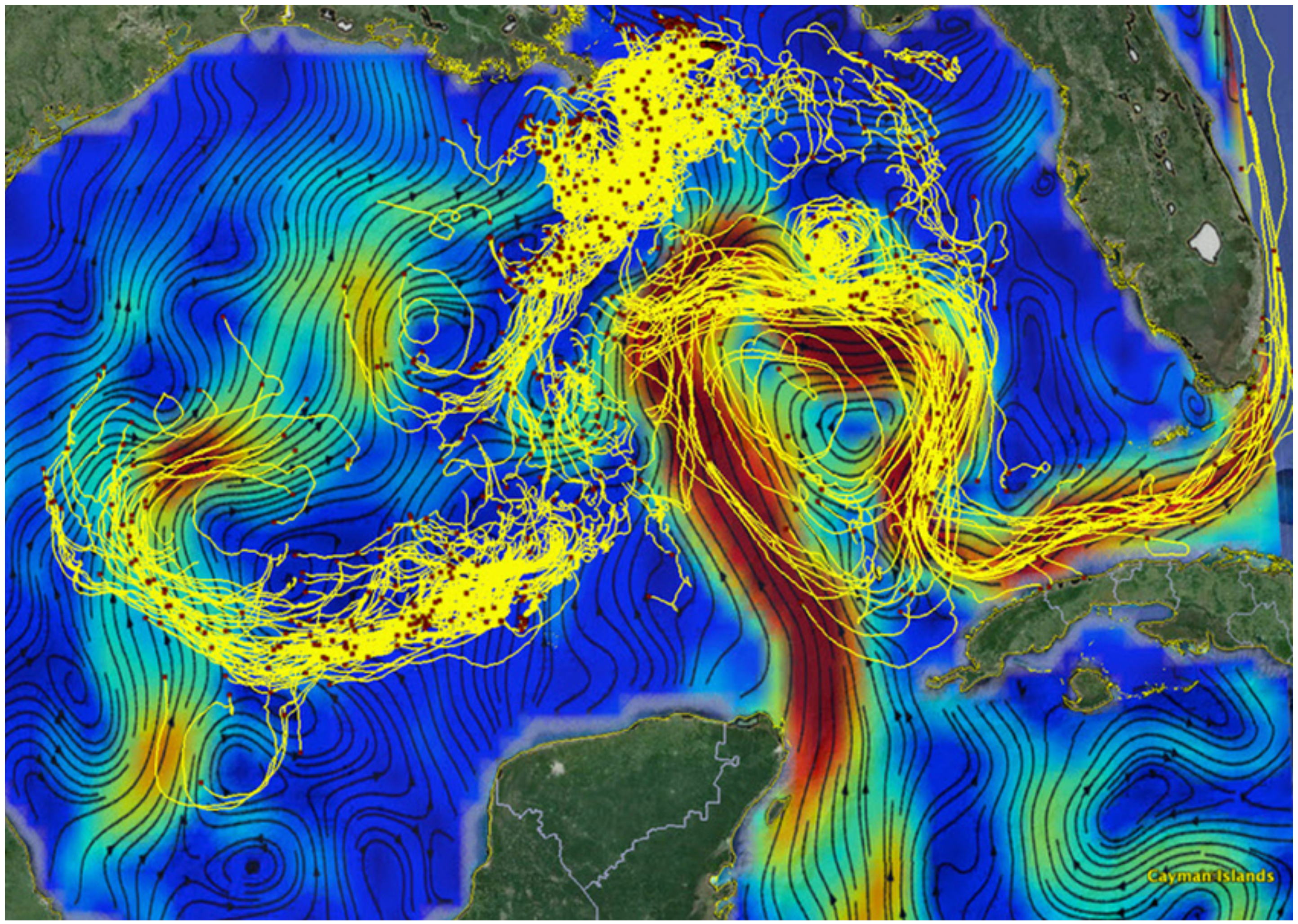
Example:

- The ocean current (velocity vector field) varies by space & time
- Scientists get sparse observations of the current from buoys
- Goal: estimate the current “predict”

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Why Gaussian processes (GPs)?

- Often want to estimate/“predict” some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems



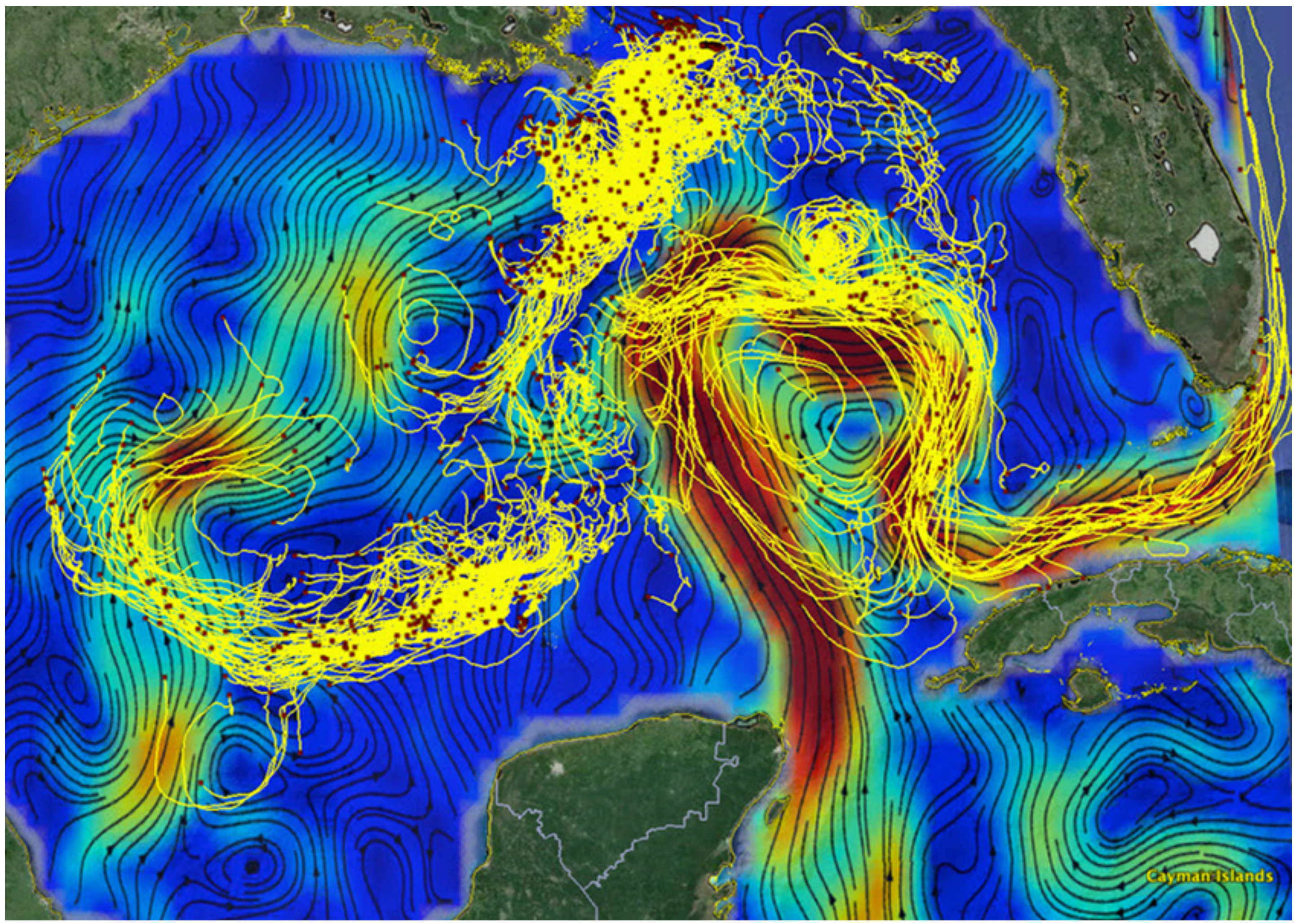
Example:

- The ocean current (velocity vector field) varies by space & time
- Scientists get sparse observations of the current from buoys
- Goal: estimate the current “predict”

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Why Gaussian processes (GPs)?

- Often want to estimate/“predict” some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems



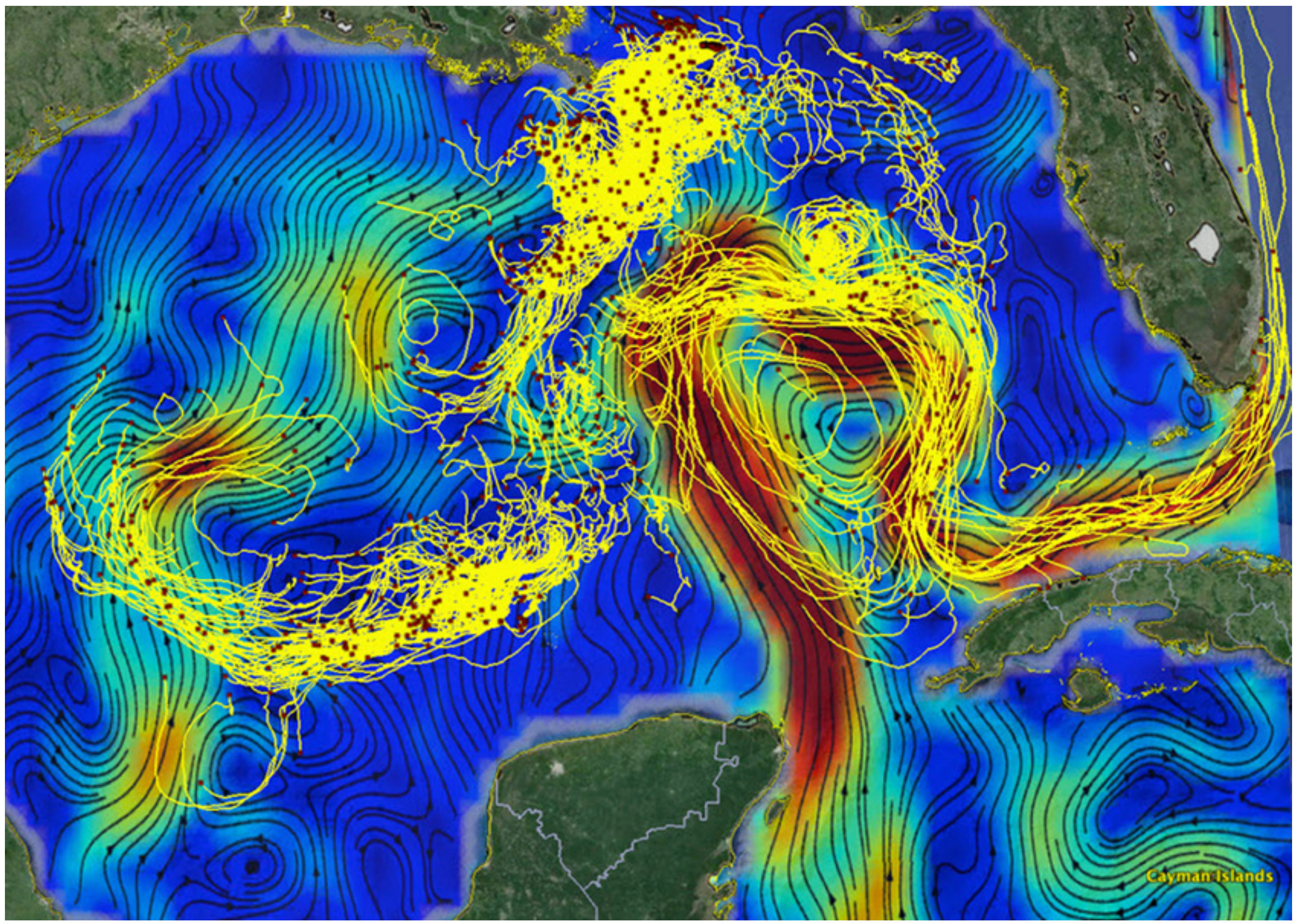
Example:
- The ocean current (velocity vector field) varies by space & time
- Scientists get sparse observations of the current from buoys
- Goal: estimate the current “predict”

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Motif:

Why Gaussian processes (GPs)?

- Often want to estimate/“predict” some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems



Example:

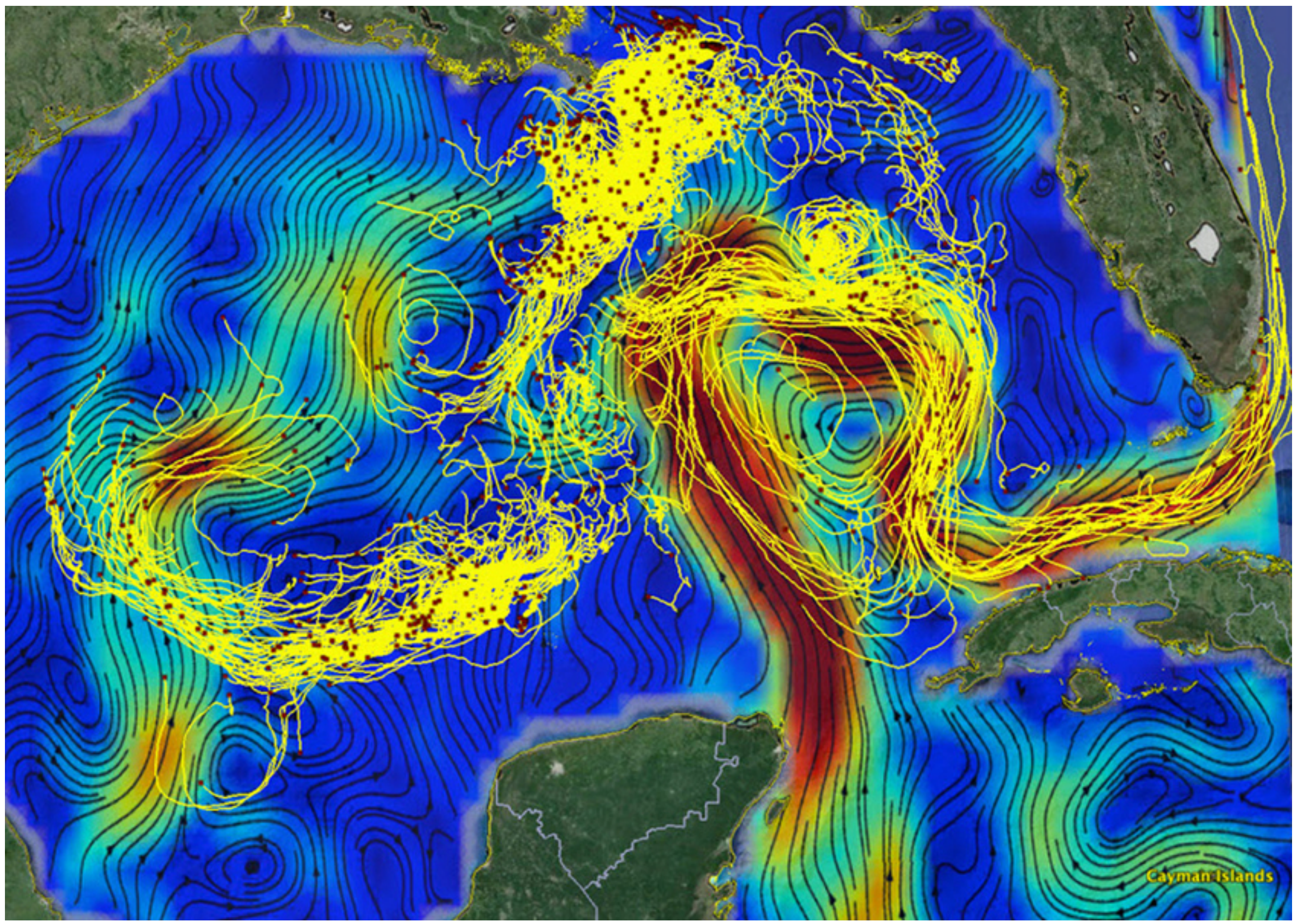
- The ocean current (velocity vector field) varies by space & time
- Scientists get sparse observations of the current from buoys
- Goal: estimate the current “predict”

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Motif: • Expensive → sparse data

Why Gaussian processes (GPs)?

- Often want to estimate/"predict" some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems



Example:

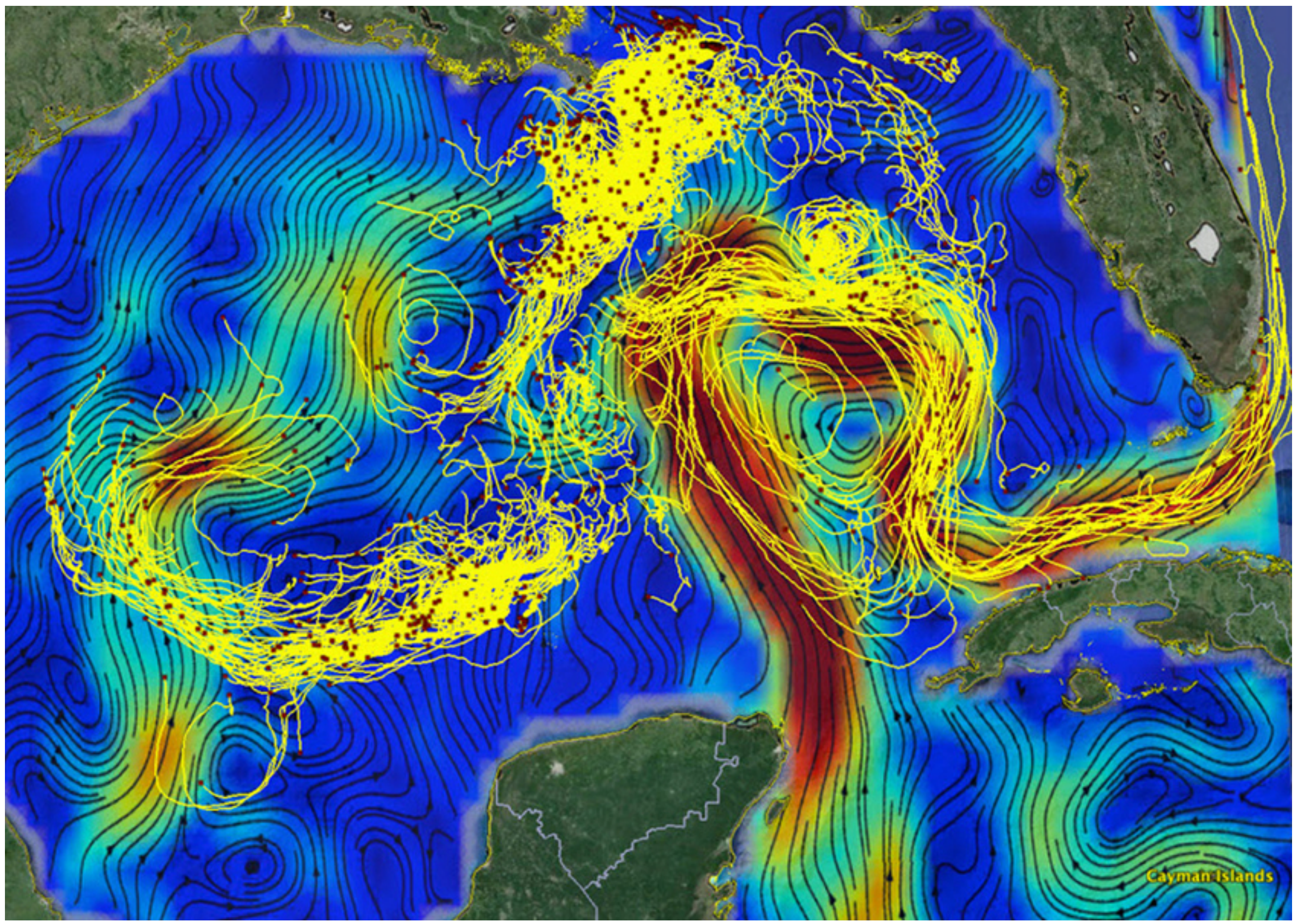
- The ocean current (velocity vector field) varies by space & time
- Scientists get sparse observations of the current from buoys
- Goal: estimate the current "predict"

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Motif: • Expensive → sparse data • Inputs not on a grid

Why Gaussian processes (GPs)?

- Often want to estimate/“predict” some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems



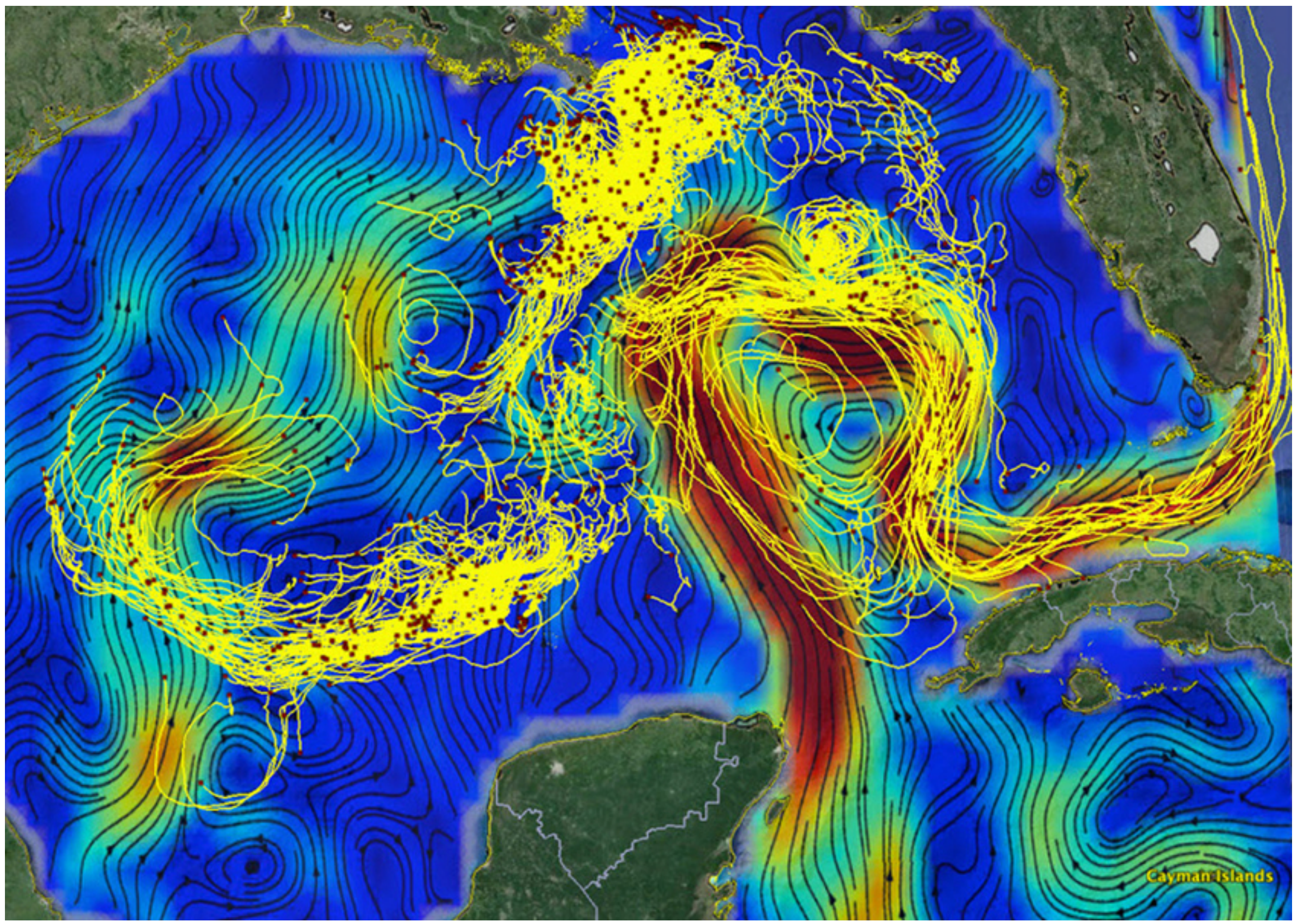
Example:
- The ocean current (velocity vector field) varies by space & time
- Scientists get sparse observations of the current from buoys
- Goal: estimate the current “predict”

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Motif: • Expensive → sparse data • Inputs not on a grid
- Current is highly nonlinear but smooth in space-time

Why Gaussian processes (GPs)?

- Often want to estimate/“predict” some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems



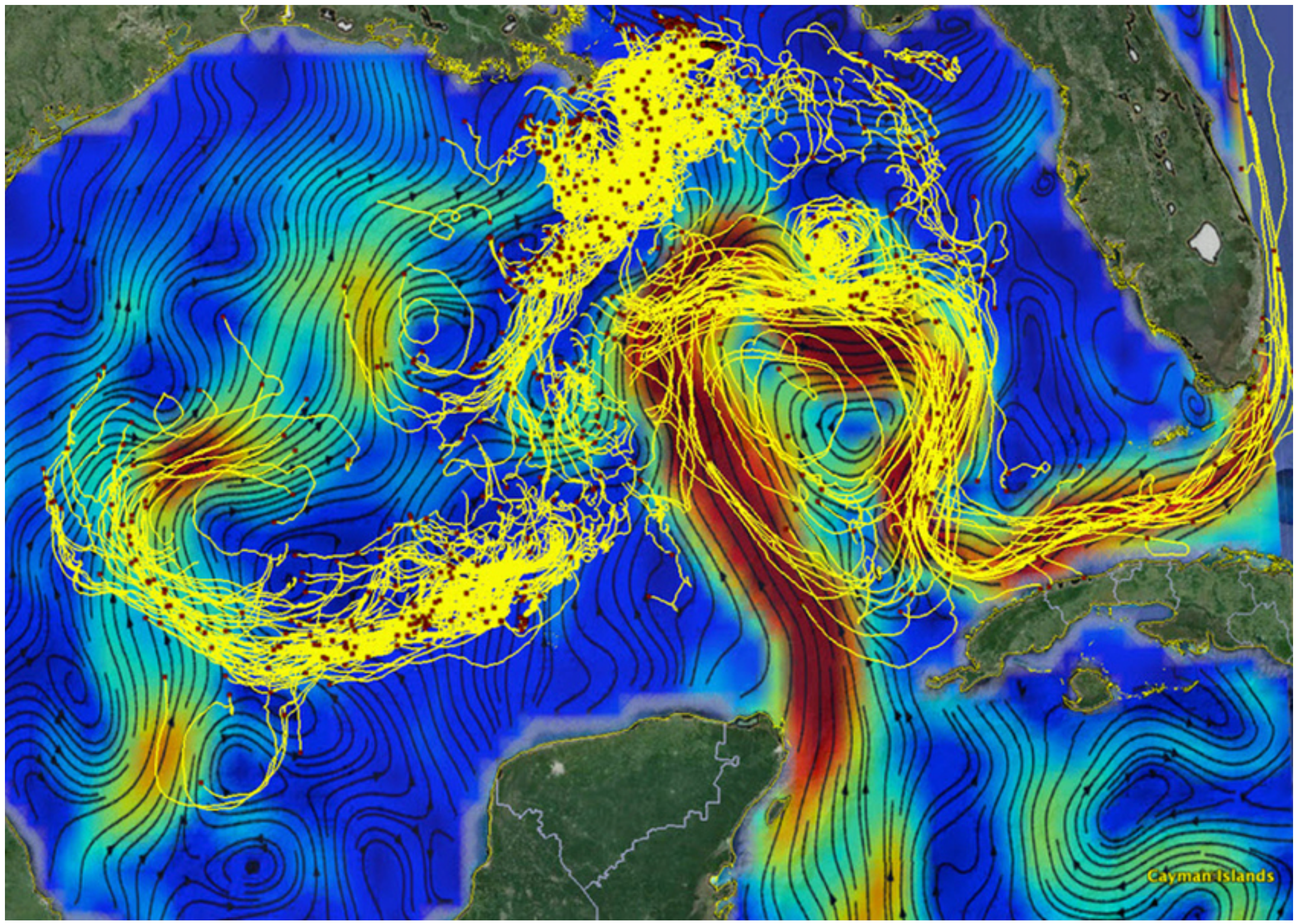
Example:
- The ocean current (velocity vector field) varies by space & time
- Scientists get sparse observations of the current from buoys
- Goal: estimate the current “predict”

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Motif: • Expensive → sparse data • Inputs not on a grid
- Current is highly nonlinear but smooth in space-time
- Want uncertainty quantification

Why Gaussian processes (GPs)?

- Often want to estimate/"predict" some continuous outcome as a function of certain inputs (*regression*)
- GPs are good at certain types of regression problems



Example:
- The ocean current (velocity vector field) varies by space & time
- Scientists get sparse observations of the current from buoys
- Goal: estimate the current "predict"

[Ryan, Özgökmen 2023; Zewe 2023; Gonçalves et al 2019; Lodise et al 2020; Berlinghieri et al 2023]

Motif: • Expensive → sparse data • Inputs not on a grid
- Current is highly nonlinear but smooth in space-time
- Want uncertainty quantification • Few inputs

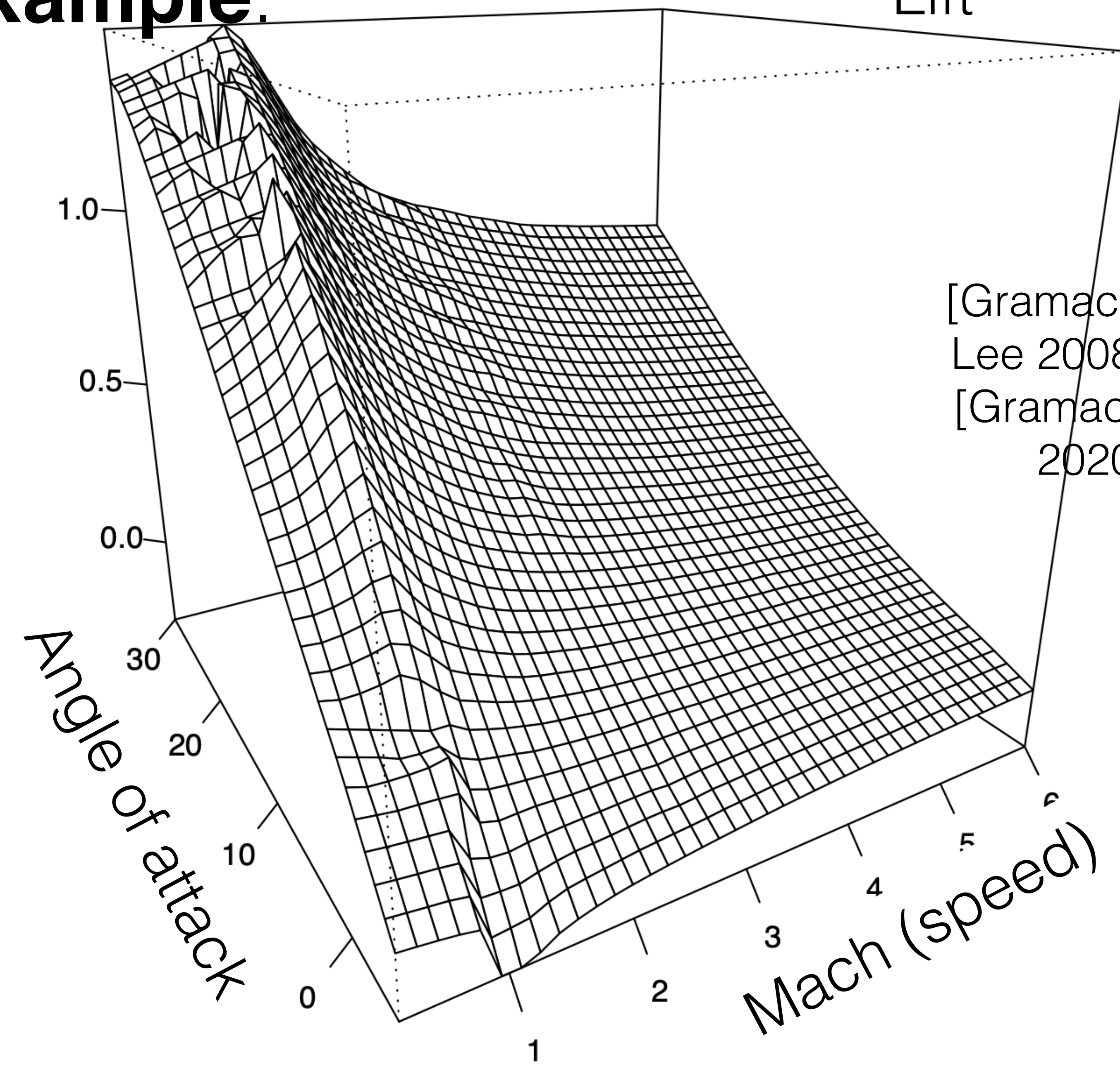
Why Gaussian processes?

Surrogate model/emulator example:

Why Gaussian processes?

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle

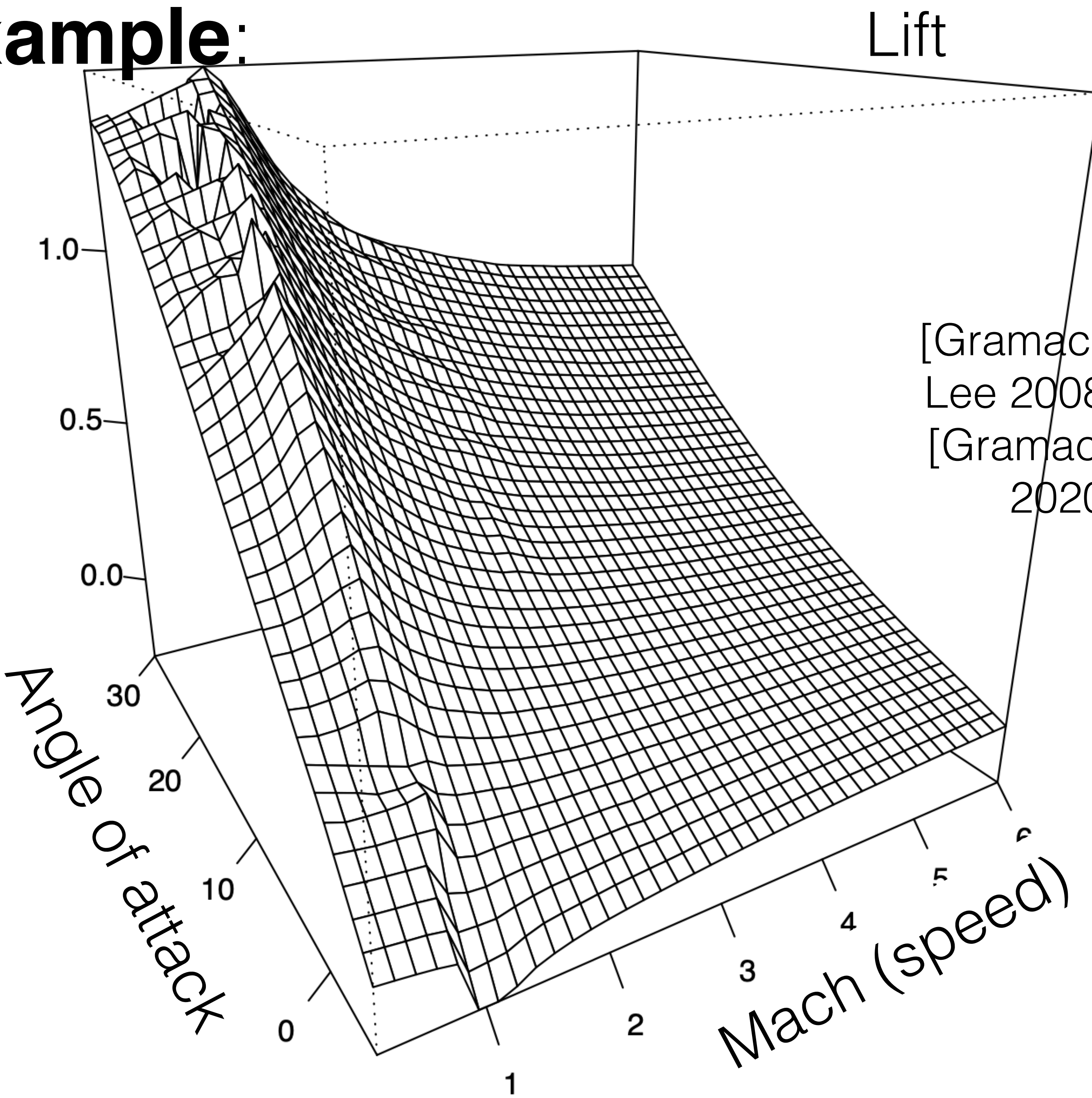


[Gramacy, Lee 2008] [Gramacy 2020]

Why Gaussian processes?

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle
- Scientists can run expensive simulations at chosen input settings

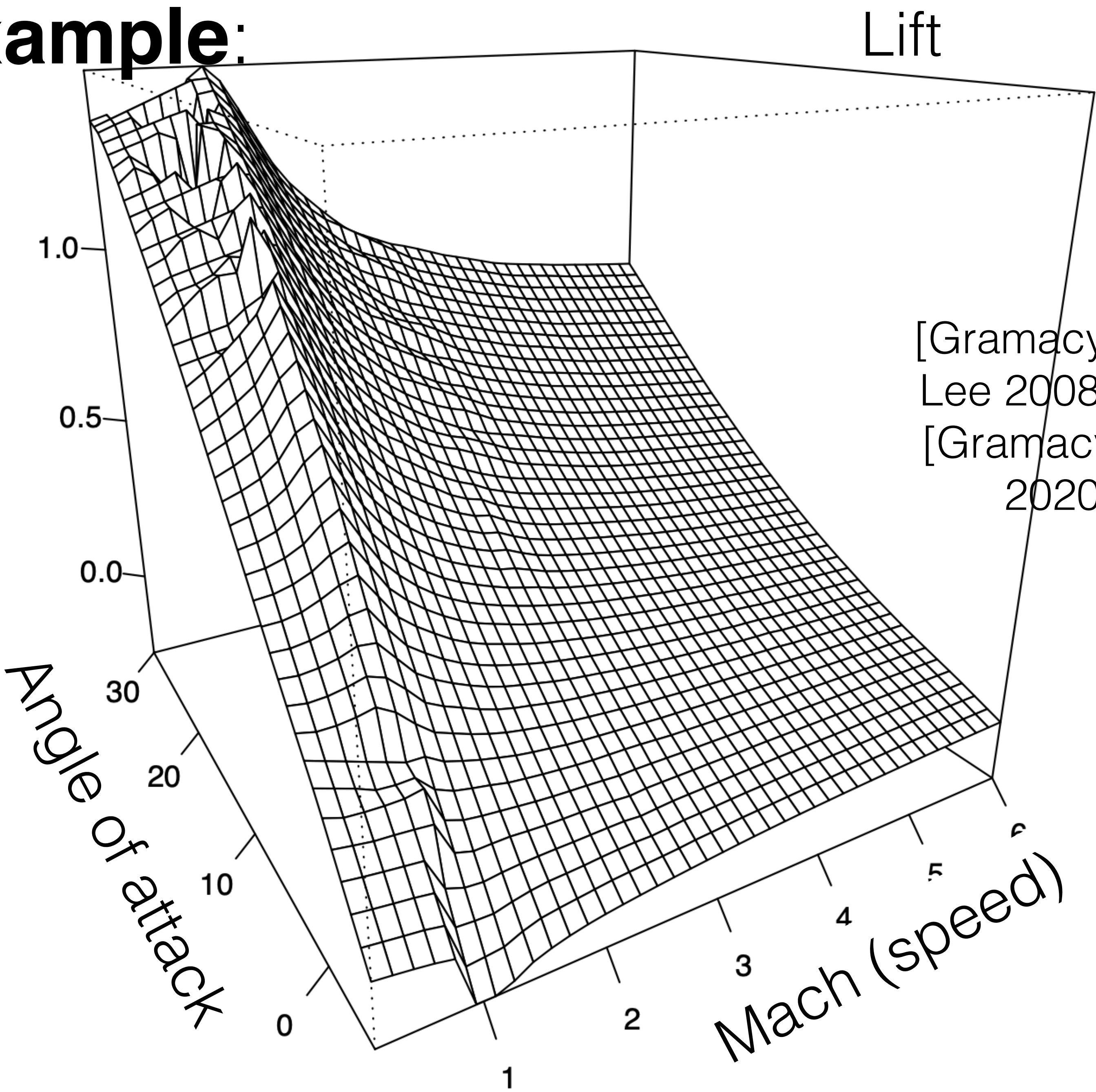


[Gramacy, Lee 2008] [Gramacy 2020]

Why Gaussian processes?

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle
- Scientists can run expensive simulations at chosen input settings
- Goal: estimate how lift varies as function of inputs

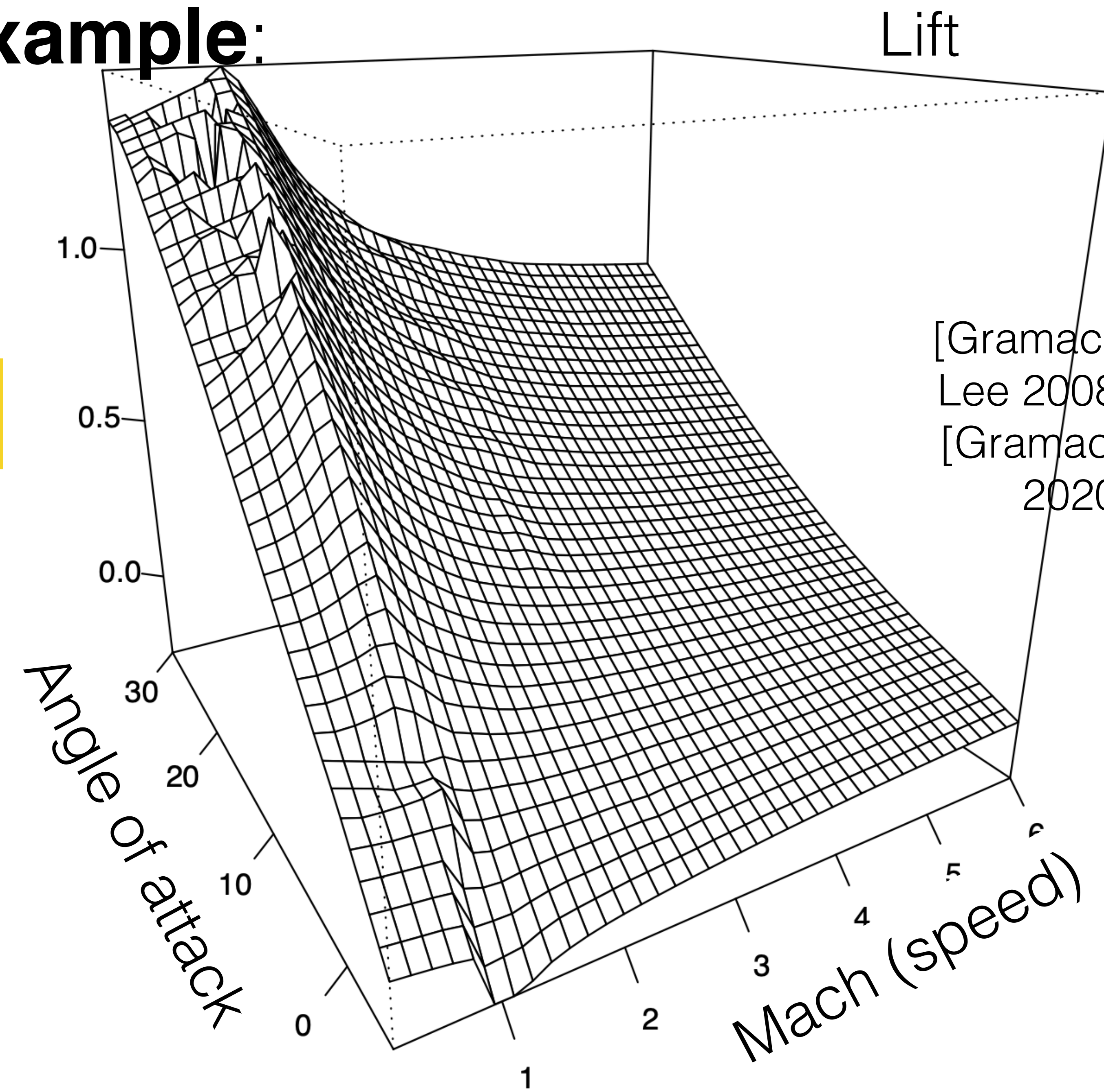


[Gramacy, Lee 2008] [Gramacy 2020]

Why Gaussian processes?

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle
- Scientists can run expensive simulations at chosen input settings
- Goal: estimate how lift varies as function of inputs

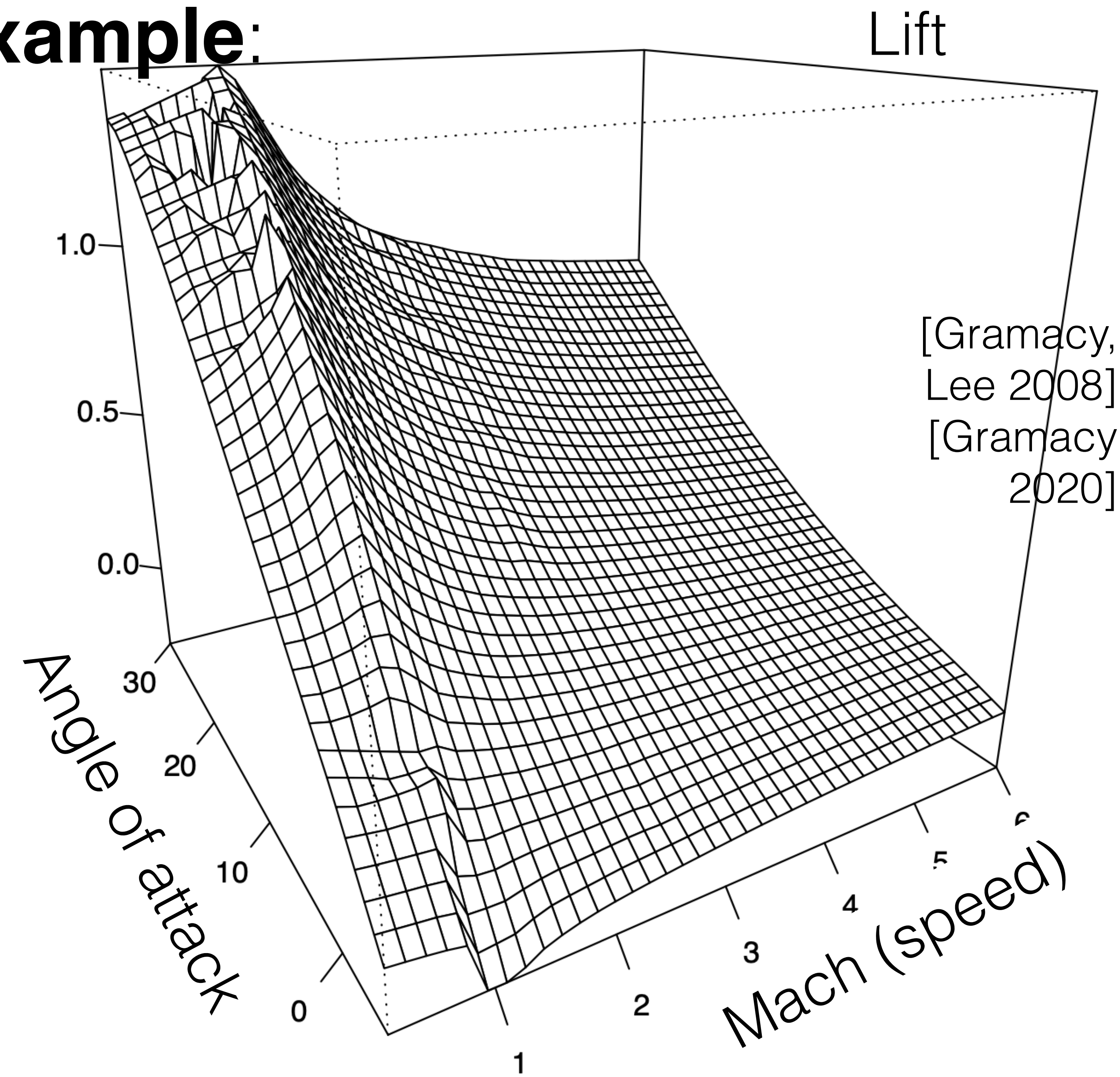


[Gramacy, Lee 2008] [Gramacy 2020]

Why Gaussian processes?

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle
- Scientists can run expensive simulations at chosen input settings
- Goal: estimate how lift varies as function of inputs

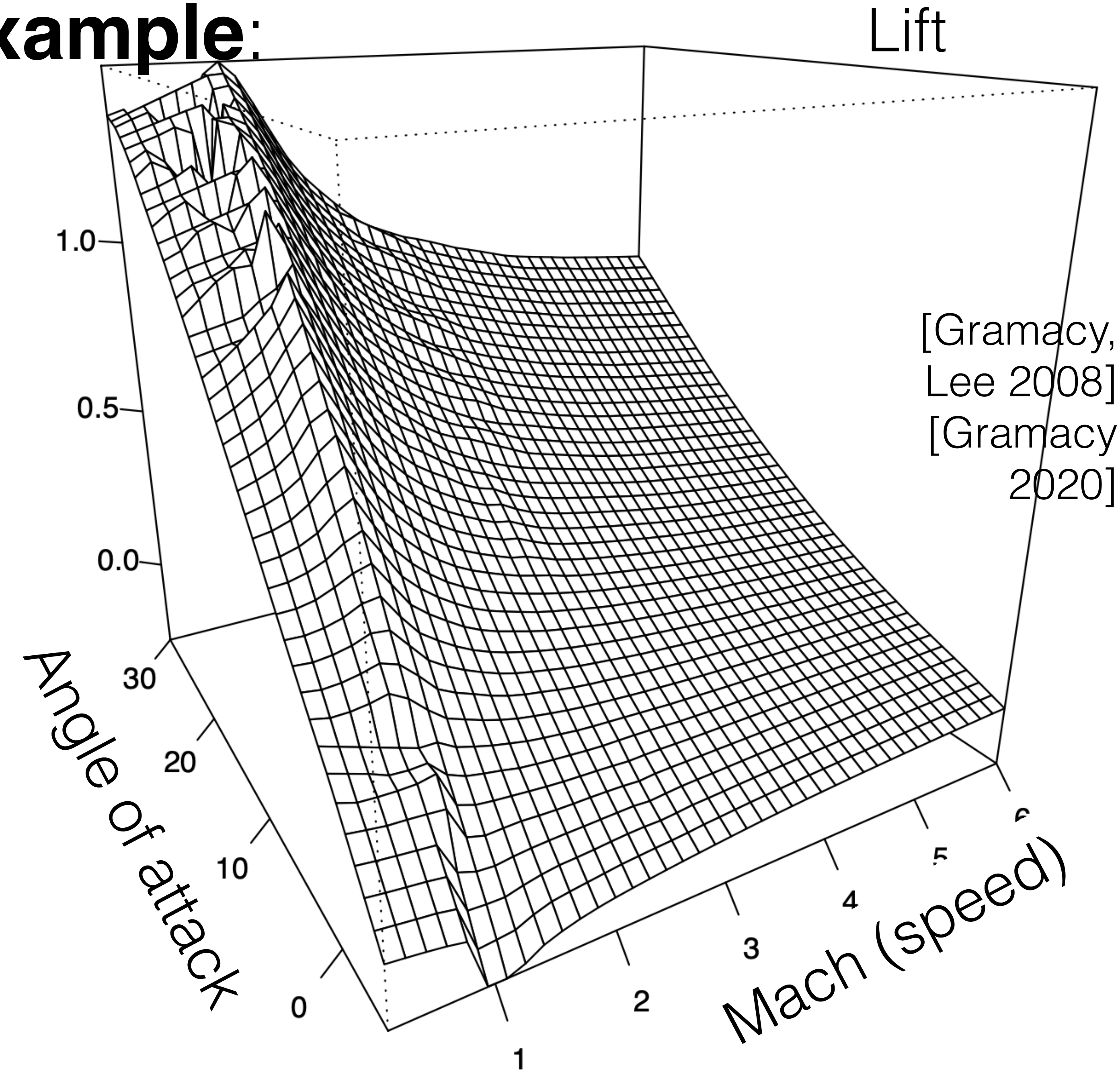


[Gramacy, Lee 2008] [Gramacy 2020]

Why Gaussian processes?

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle
- Scientists can run expensive simulations at chosen input settings
- Goal: estimate how lift varies as function of inputs



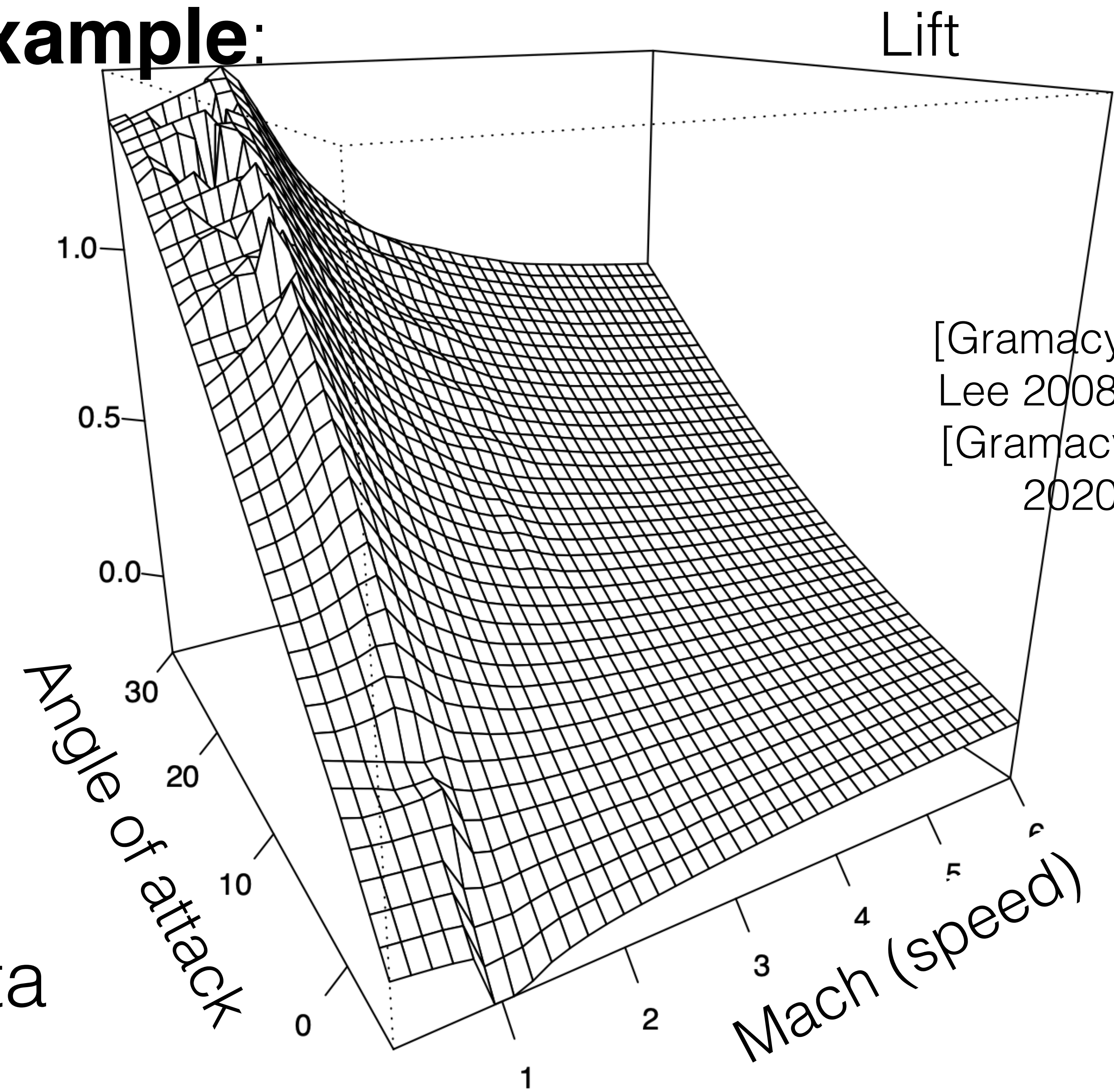
[Gramacy, Lee 2008] [Gramacy 2020]

Motif:

Why Gaussian processes?

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle
- Scientists can run expensive simulations at chosen input settings
- Goal: estimate how lift varies as function of inputs



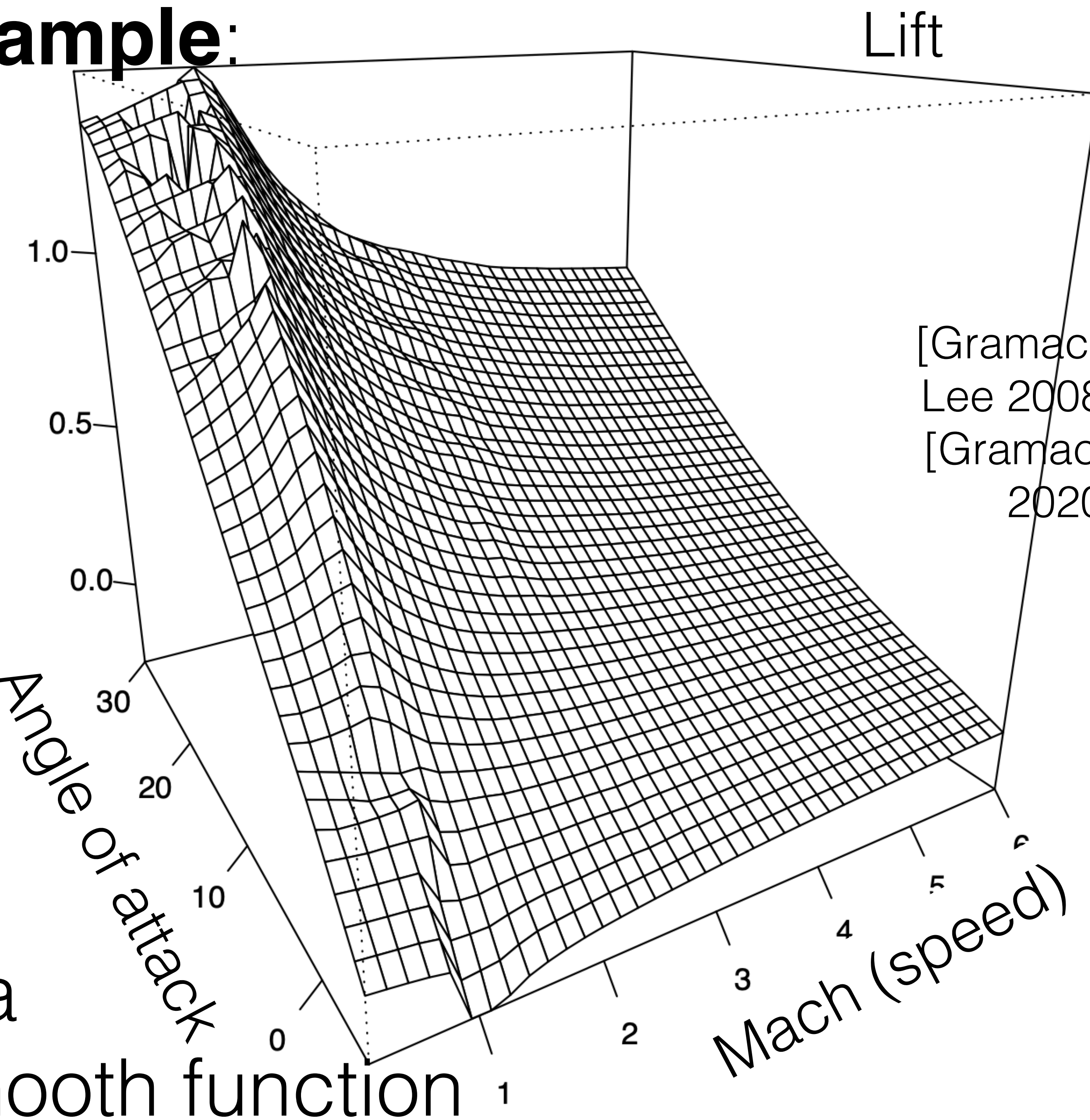
[Gramacy, Lee 2008] [Gramacy 2020]

Motif: • Expensive, sparse data

Why Gaussian processes?

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle
- Scientists can run expensive simulations at chosen input settings
- Goal: estimate how lift varies as function of inputs



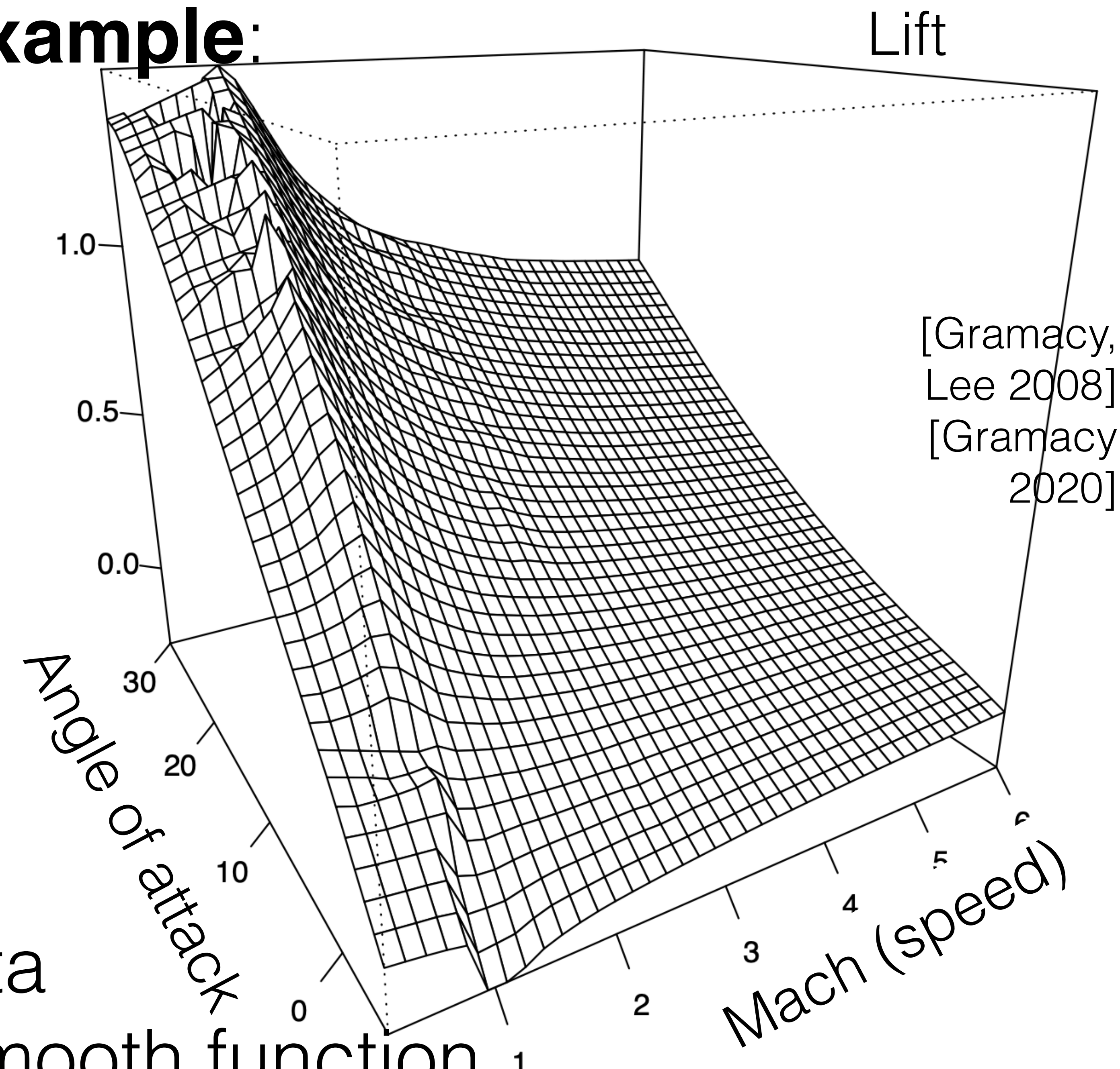
[Gramacy, Lee 2008] [Gramacy 2020]

Motif:
- Expensive, sparse data
- Potentially nonlinear but smooth function

Why Gaussian processes?

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle
- Scientists can run expensive simulations at chosen input settings
- Goal: estimate how lift varies as function of inputs



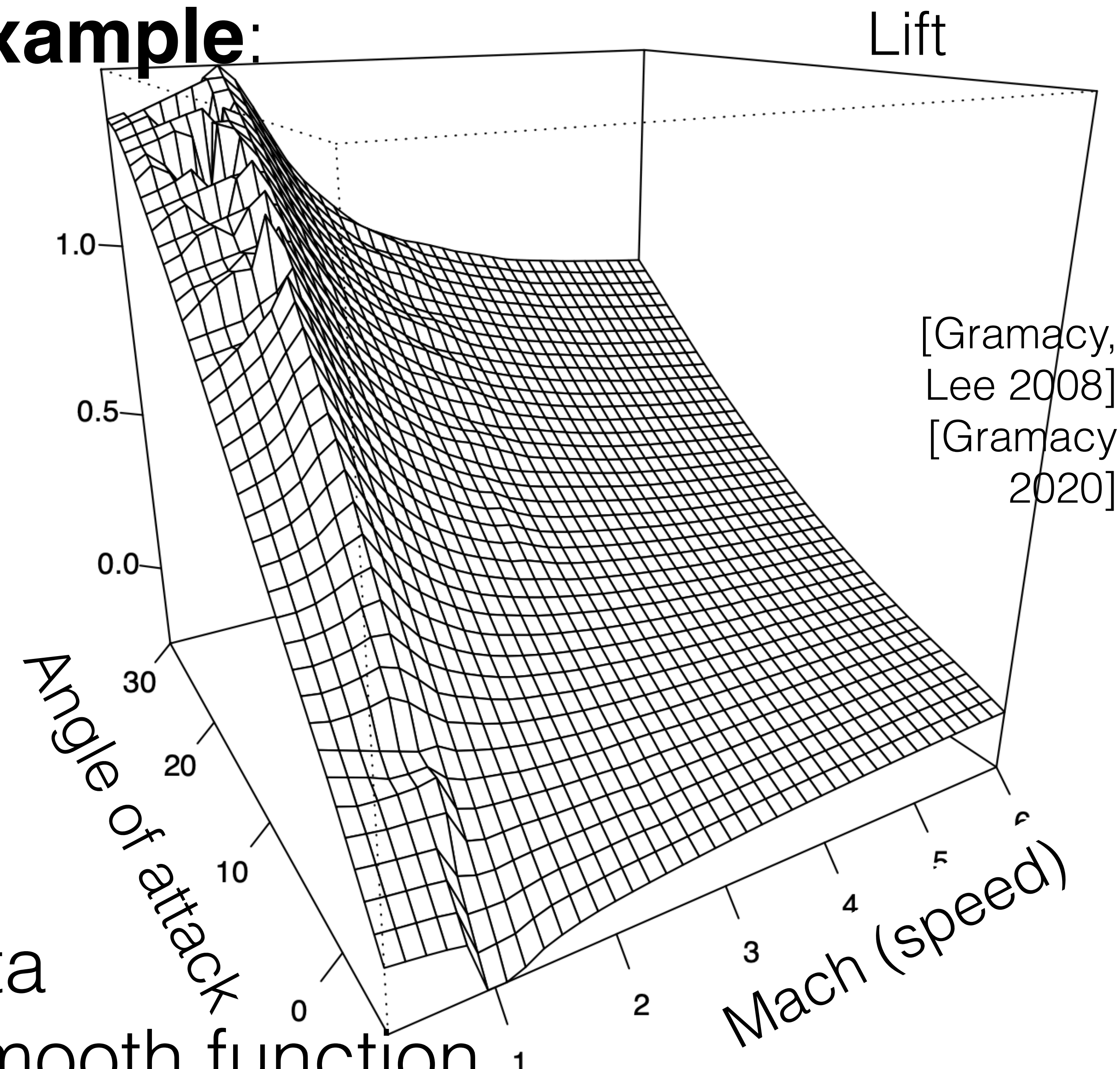
[Gramacy, Lee 2008] [Gramacy 2020]

Motif:
- Expensive, sparse data
- Potentially nonlinear but smooth function
- Few inputs, not on grid

Why Gaussian processes?

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle
- Scientists can run expensive simulations at chosen input settings
- Goal: estimate how lift varies as function of inputs



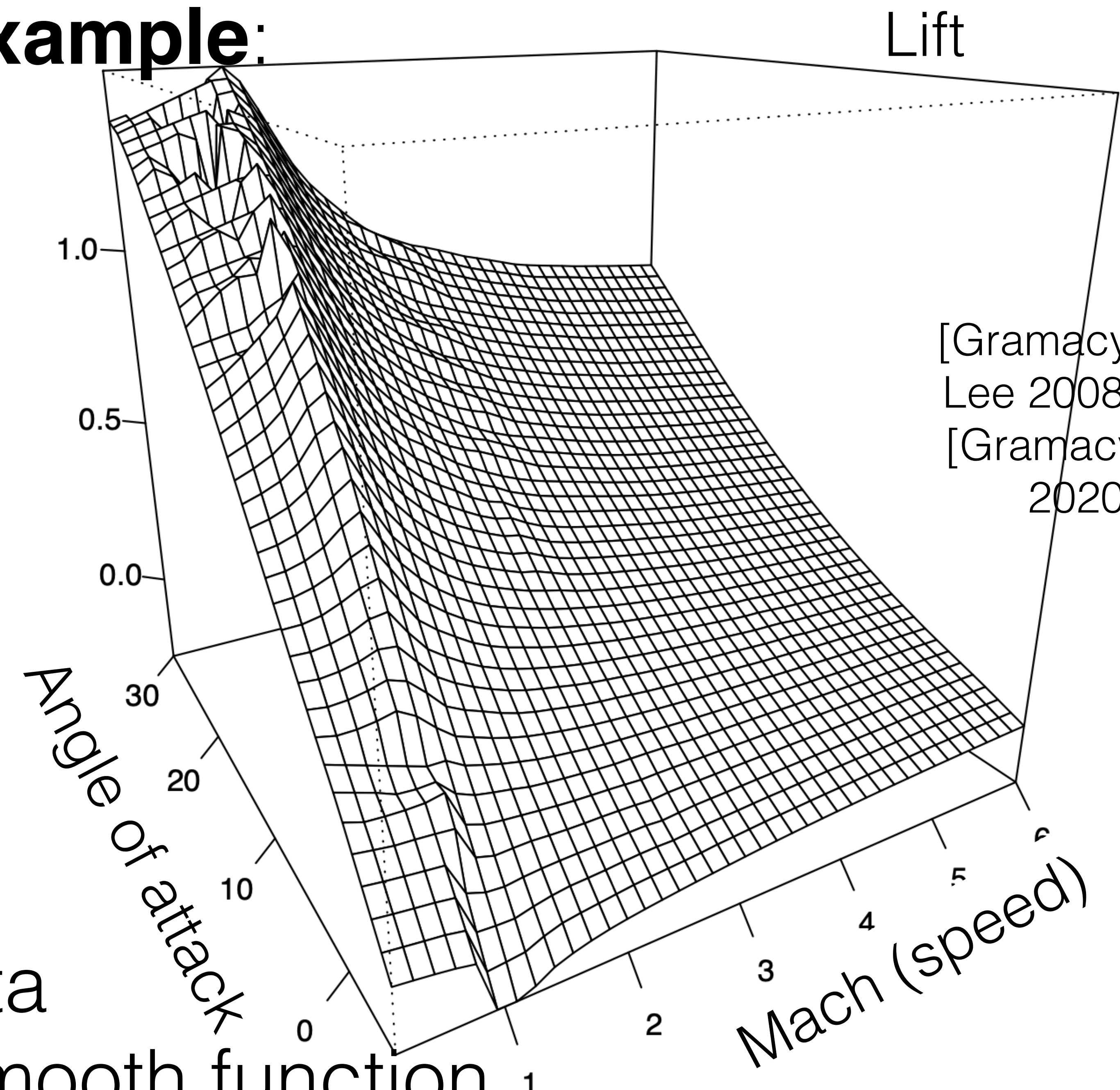
[Gramacy, Lee 2008] [Gramacy 2020]

Motif:
- Expensive, sparse data
- Potentially nonlinear but smooth function
- Few inputs, not on grid
- Want uncertainty quantification

Why Gaussian processes?

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle
- Scientists can run expensive simulations at chosen input settings
- Goal: estimate how lift varies as function of inputs



[Gramacy, Lee 2008] [Gramacy 2020]

Motif:
- Expensive, sparse data
- Potentially nonlinear but smooth function
- Few inputs, not on grid
- Want uncertainty quantification

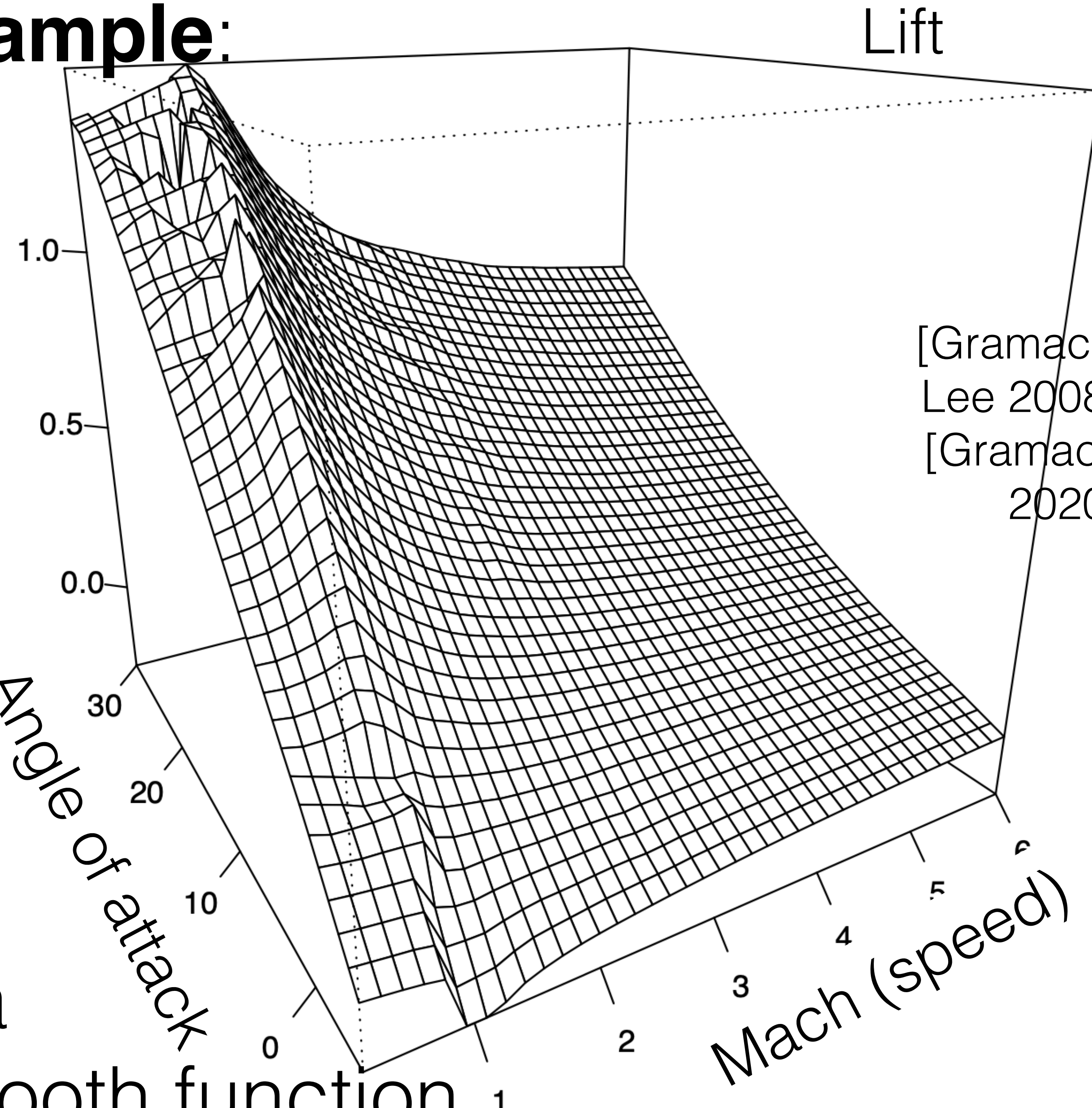
Myren, Lawrence 2021: “the Gaussian process emulators perform better overall than the neural network emulators. [But e]ach of these examples is a good match for the assumptions of the Gaussian process.”

Why Gaussian processes?

see also "kriging," "optimal interpolation (OI)"

Surrogate model/emulator example:

- The lift force of a rocket booster varies as a function of speed at re-entry, angle of attack, and sideslip angle
- Scientists can run expensive simulations at chosen input settings
- Goal: estimate how lift varies as function of inputs



[Gramacy, Lee 2008] [Gramacy 2020]

Motif: • Expensive, sparse data

- Potentially nonlinear but smooth function
- Few inputs, not on grid • Want uncertainty quantification

Myren, Lawrence 2021: "the Gaussian process emulators perform better overall than the neural network emulators. [But e]ach of these examples is a good match for the assumptions of the Gaussian process."

Why Gaussian processes?

One last **example**:

Why Gaussian processes?

One last **example**:

- Estimate (and optimize) deep learning performance as a function of learning rate, momentum, and other settings

Why Gaussian processes?

One last **example**:

- Estimate (and optimize) deep learning performance as a function of learning rate, momentum, and other settings
- Deep Learning Tuning Playbook: "Currently, there is an astonishing amount of toil and guesswork involved in actually getting deep neural networks to work well in practice. Even worse, the actual recipes people use to get good results with deep learning are rarely documented. […] There is a vast gulf between the results achieved by deep learning experts and less skilled practitioners using superficially similar methods."

[Godbole et al 2023]

Why Gaussian processes?

One last **example**:

- Estimate (and optimize) deep learning performance as a function of learning rate, momentum, and other settings
- Deep Learning Tuning Playbook: "Currently, there is an astonishing amount of toil and guesswork involved in actually getting deep neural networks to work well in practice. Even worse, the actual recipes people use to get good results with deep learning are rarely documented. [...] There is a vast gulf between the results achieved by deep learning experts and less skilled practitioners using superficially similar methods."
- **Motif**: sparse, no grid, nonlinear, smooth, few inputs, UQ

[Godbole et al 2023]

Why Gaussian processes?

One last **example**:

- Estimate (and optimize) deep learning performance as a function of learning rate, momentum, and other settings
- Deep Learning Tuning Playbook: "Currently, there is an astonishing amount of toil and guesswork involved in actually getting deep neural networks to work well in practice. Even worse, the actual recipes people use to get good results with deep learning are rarely documented. […] There is a vast gulf between the results achieved by deep learning experts and less skilled practitioners using superficially similar methods."
- **Motif**: sparse, no grid, nonlinear, smooth, few inputs, UQ

Bonus benefits of GPs:

[Godbole et al 2023]

Why Gaussian processes?

One last **example**:

- Estimate (and optimize) deep learning performance as a function of learning rate, momentum, and other settings
- Deep Learning Tuning Playbook: "Currently, there is an astonishing amount of toil and guesswork involved in actually getting deep neural networks to work well in practice. Even worse, the actual recipes people use to get good results with deep learning are rarely documented. […] There is a vast gulf between the results achieved by deep learning experts and less skilled practitioners using superficially similar methods."
- **Motif**: sparse, no grid, nonlinear, smooth, few inputs, UQ

Bonus benefits of GPs: • Ease of use (software, tuning)

[Godbole et al 2023]

Why Gaussian processes?

One last **example**:

- Estimate (and optimize) deep learning performance as a function of learning rate, momentum, and other settings
- Deep Learning Tuning Playbook: "Currently, there is an astonishing amount of toil and guesswork involved in actually getting deep neural networks to work well in practice. Even worse, the actual recipes people use to get good results with deep learning are rarely documented. […] There is a vast gulf between the results achieved by deep learning experts and less skilled practitioners using superficially similar methods."
- **Motif**: sparse, no grid, nonlinear, smooth, few inputs, UQ

Bonus benefits of GPs: • Ease of use (software, tuning)
• Supports optimization of outcome

[Godbole et al 2023]

Why Gaussian processes?

One last **example**:

- Estimate (and optimize) deep learning performance as a function of learning rate, momentum, and other settings
- Deep Learning Tuning Playbook: "Currently, there is an astonishing amount of toil and guesswork involved in actually getting deep neural networks to work well in practice. Even worse, the actual recipes people use to get good results with deep learning are rarely documented. [...] There is a vast gulf between the results achieved by deep learning experts and less skilled practitioners using superficially similar methods."
- **Motif**: sparse, no grid, nonlinear, smooth, few inputs, UQ

Bonus benefits of GPs: • Ease of use (software, tuning) • Supports optimization of outcome • Predictions & uncertainties over derivatives & integrals

[Godbole et al 2023]

Why Gaussian processes?

One last **example**:

- Estimate (and optimize) deep learning performance as a function of learning rate, momentum, and other settings
- Deep Learning Tuning Playbook: "Currently, there is an astonishing amount of toil and guesswork involved in actually getting deep neural networks to work well in practice. Even worse, the actual recipes people use to get good results with deep learning are rarely documented. […] There is a vast gulf between the results achieved by deep learning experts and less skilled practitioners using superficially similar methods."
- **Motif**: sparse, no grid, nonlinear, smooth, few inputs, UQ

Bonus benefits of GPs: • Ease of use (software, tuning) • Supports optimization of outcome • Predictions & uncertainties over derivatives & integrals • Module within more-complex methods

[Godbole et al 2023]

Roadmap

Roadmap

- A Bayesian approach

Roadmap

- A Bayesian approach
- What is a Gaussian process (GP)?

Roadmap

- A Bayesian approach
- What is a Gaussian process (GP)?
  - A particularly popular version in practice

Roadmap

- A Bayesian approach
- What is a Gaussian process (GP)?
  - A particularly popular version in practice
- Gaussian process inference

Roadmap

- A Bayesian approach
- What is a Gaussian process (GP)?
  - A particularly popular version in practice
- Gaussian process inference
  - Prediction & uncertainty quantification

Roadmap

- A Bayesian approach
- What is a Gaussian process (GP)?
  - A particularly popular version in practice
- Gaussian process inference
  - Prediction & uncertainty quantification
- Using GPs to optimize a function (Bayesian optimization)

Roadmap

- A Bayesian approach
- What is a Gaussian process (GP)?
  - A particularly popular version in practice
- Gaussian process inference
  - Prediction & uncertainty quantification
- Using GPs to optimize a function (Bayesian optimization)
- What are the limits? What can go wrong?

Roadmap

- A Bayesian approach
- What is a Gaussian process (GP)?
  - A particularly popular version in practice
- Gaussian process inference
  - Prediction & uncertainty quantification
- Using GPs to optimize a function (Bayesian optimization)
- What are the limits? What can go wrong?

- Goals:

Roadmap

- A Bayesian approach
- What is a Gaussian process (GP)?
  - A particularly popular version in practice
- Gaussian process inference
  - Prediction & uncertainty quantification
- Using GPs to optimize a function (Bayesian optimization)
- What are the limits? What can go wrong?

- Goals:
  - Learn the mechanism behind standard GPs to identify benefits and pitfalls

Roadmap

- A Bayesian approach
- What is a Gaussian process (GP)?
  - A particularly popular version in practice
- Gaussian process inference
  - Prediction & uncertainty quantification
- Using GPs to optimize a function (Bayesian optimization)
- What are the limits? What can go wrong?

- Goals:
  - Learn the mechanism behind standard GPs to identify benefits and pitfalls
  - Learn the skills to be responsible users of standard GPs (transferable to other artificial intelligence/ machine learning methods)

A Bayesian approach

A Bayesian approach

- p(unknowns | data)

A Bayesian approach

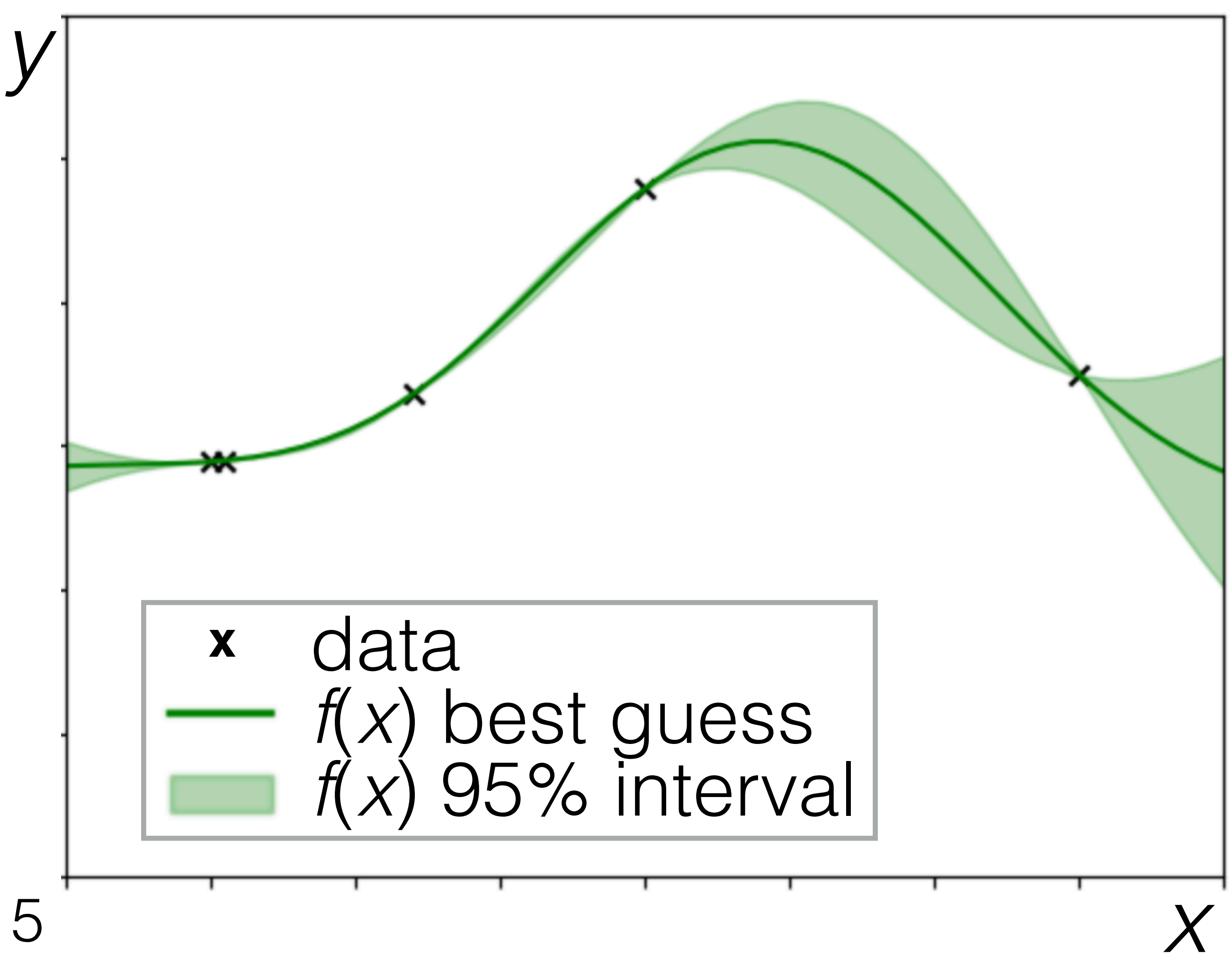
- p(unknowns | data)

Given the data we've seen, what do we know about the underlying function?

A Bayesian approach

- p(unknowns | data)

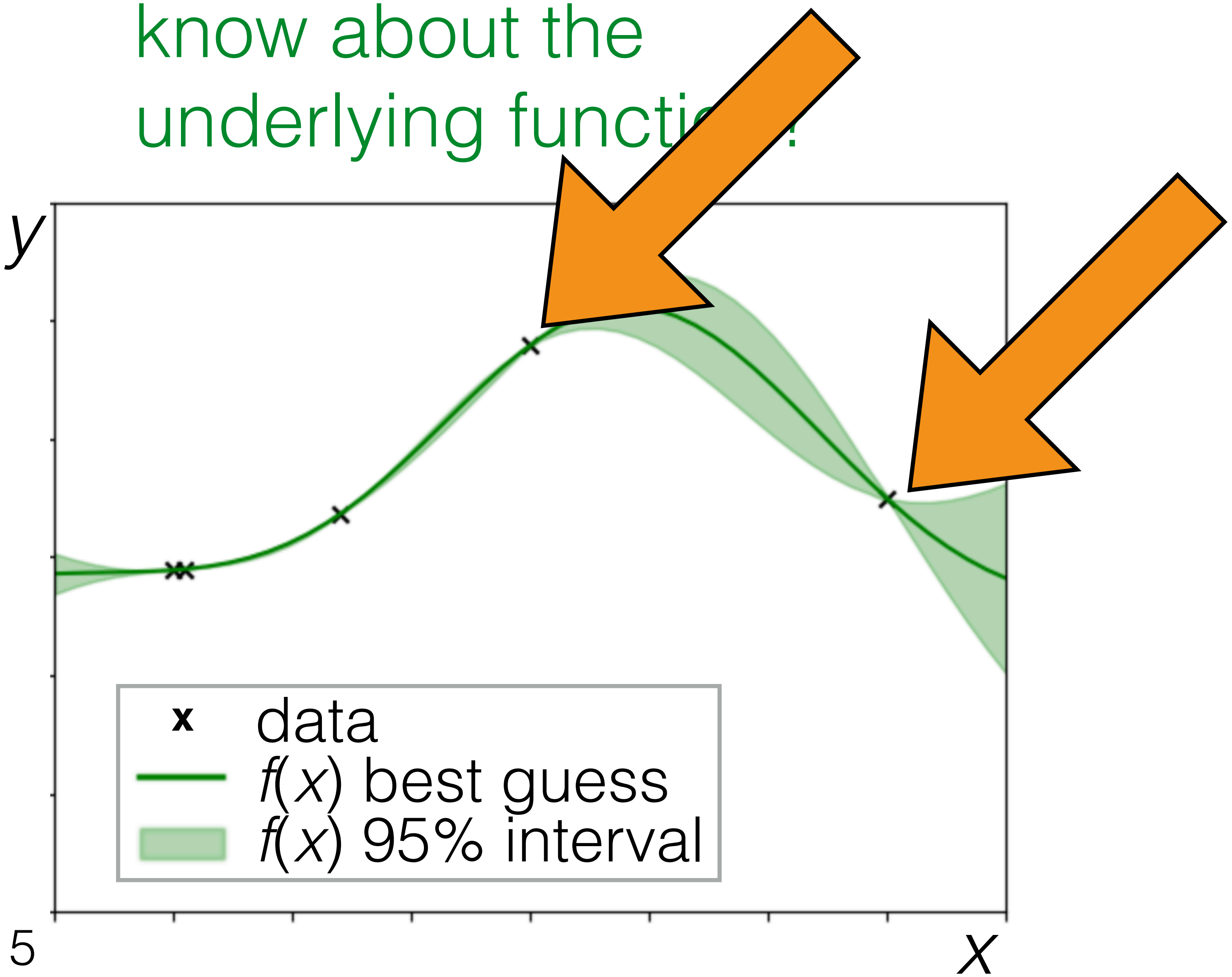
Given the data we've seen, what do we know about the underlying function?



A Bayesian approach

- p(unknowns | data)

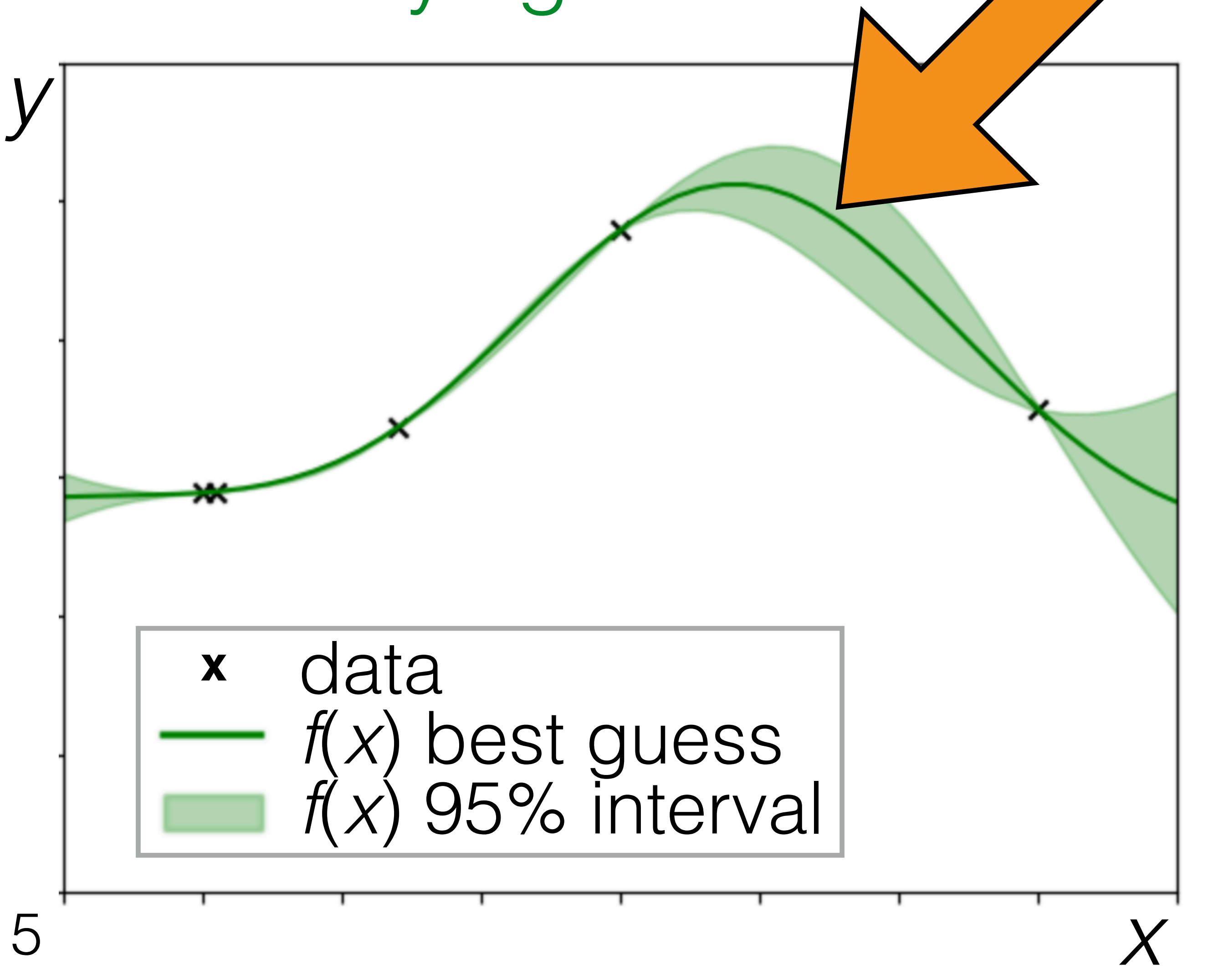
Given the data we've seen, what do we know about the underlying functi ?



A Bayesian approach

- p(unknowns | data)

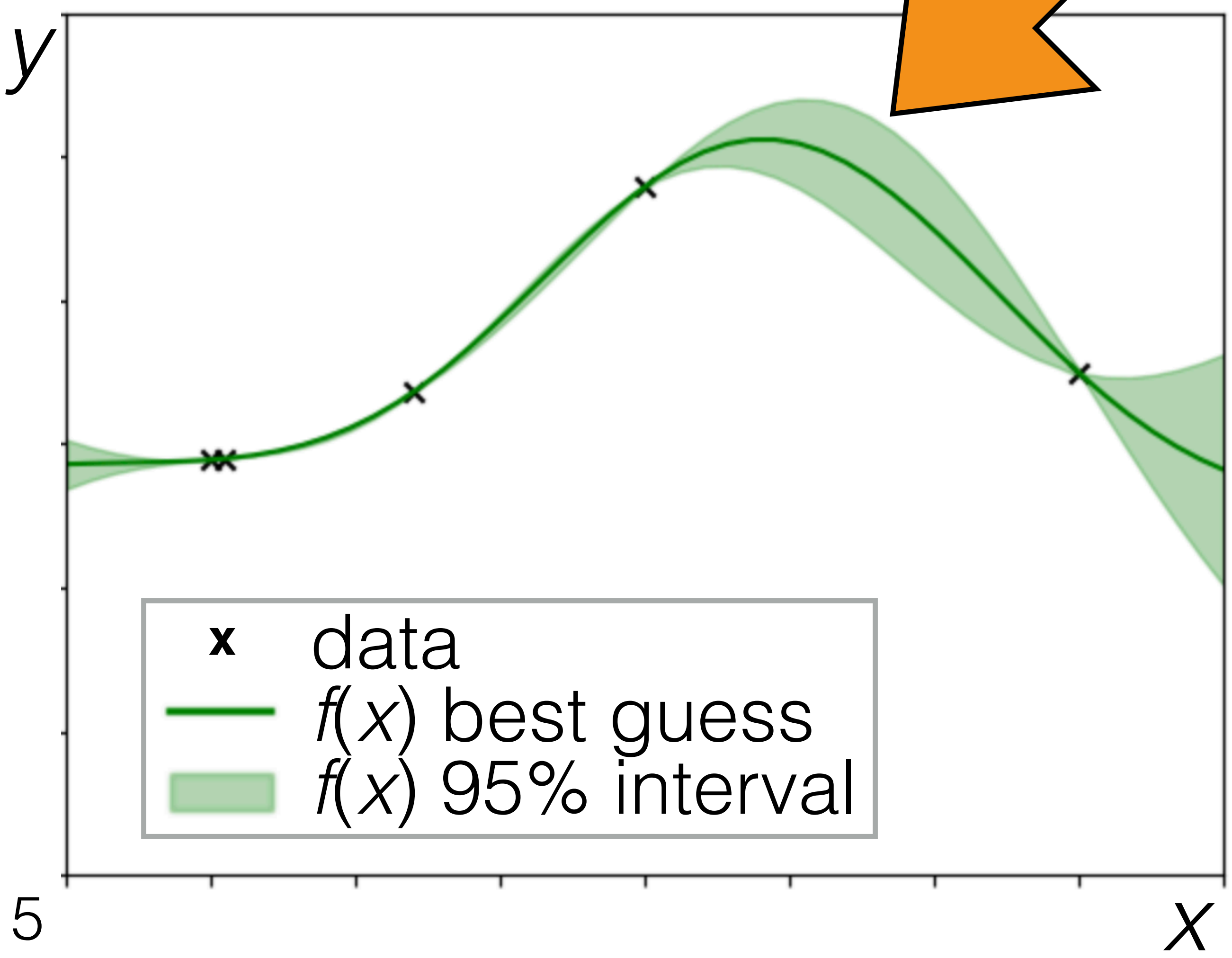
Given the data we've seen, what do we know about the underlying function?



A Bayesian approach

- p(unknowns | data)

Given the data we've seen, what do we know about the underlying function?

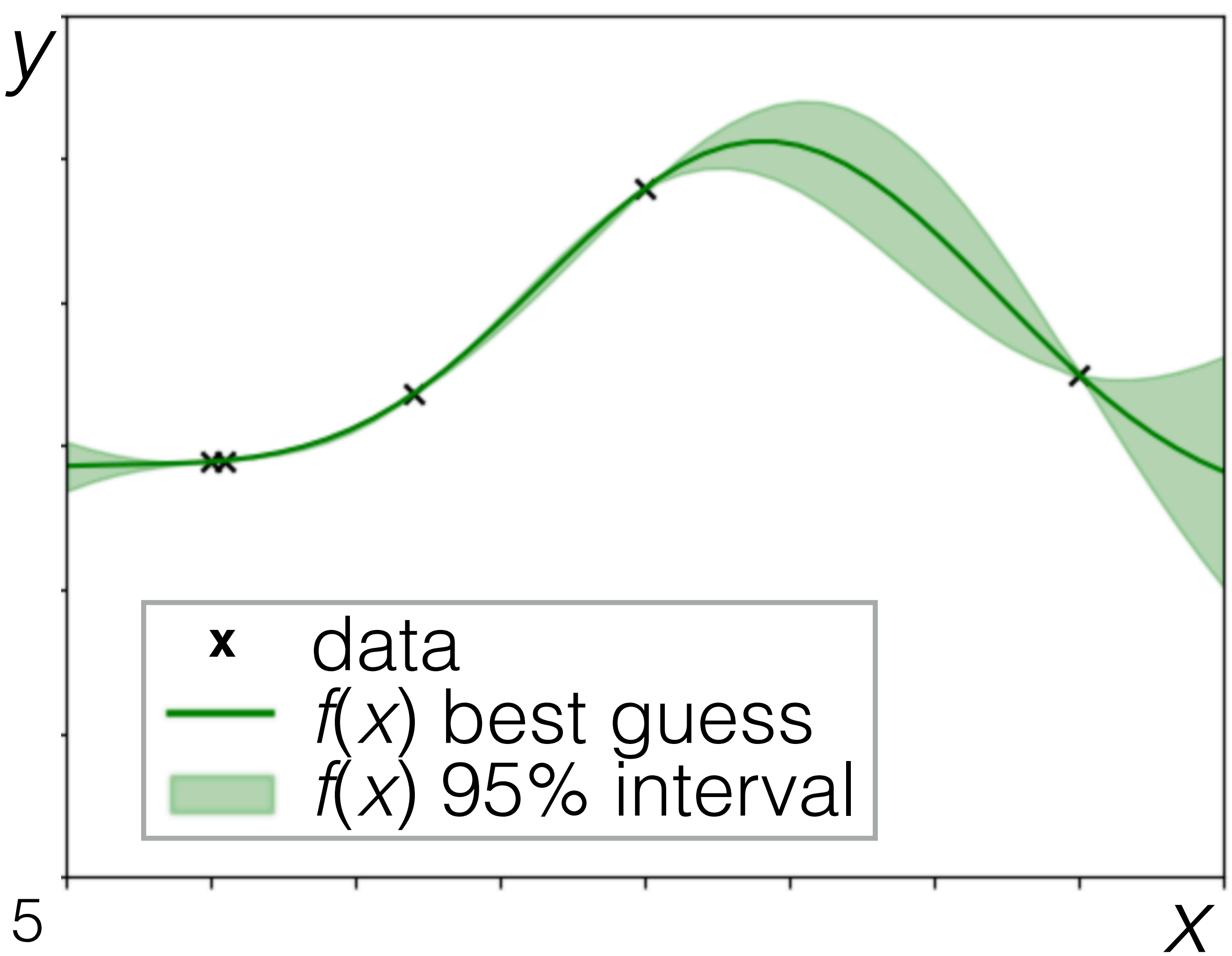


5

A Bayesian approach

- p(unknowns | data) ∝ p(data | unknowns) p(unknowns)

Given the data we've seen, what do we know about the underlying function?

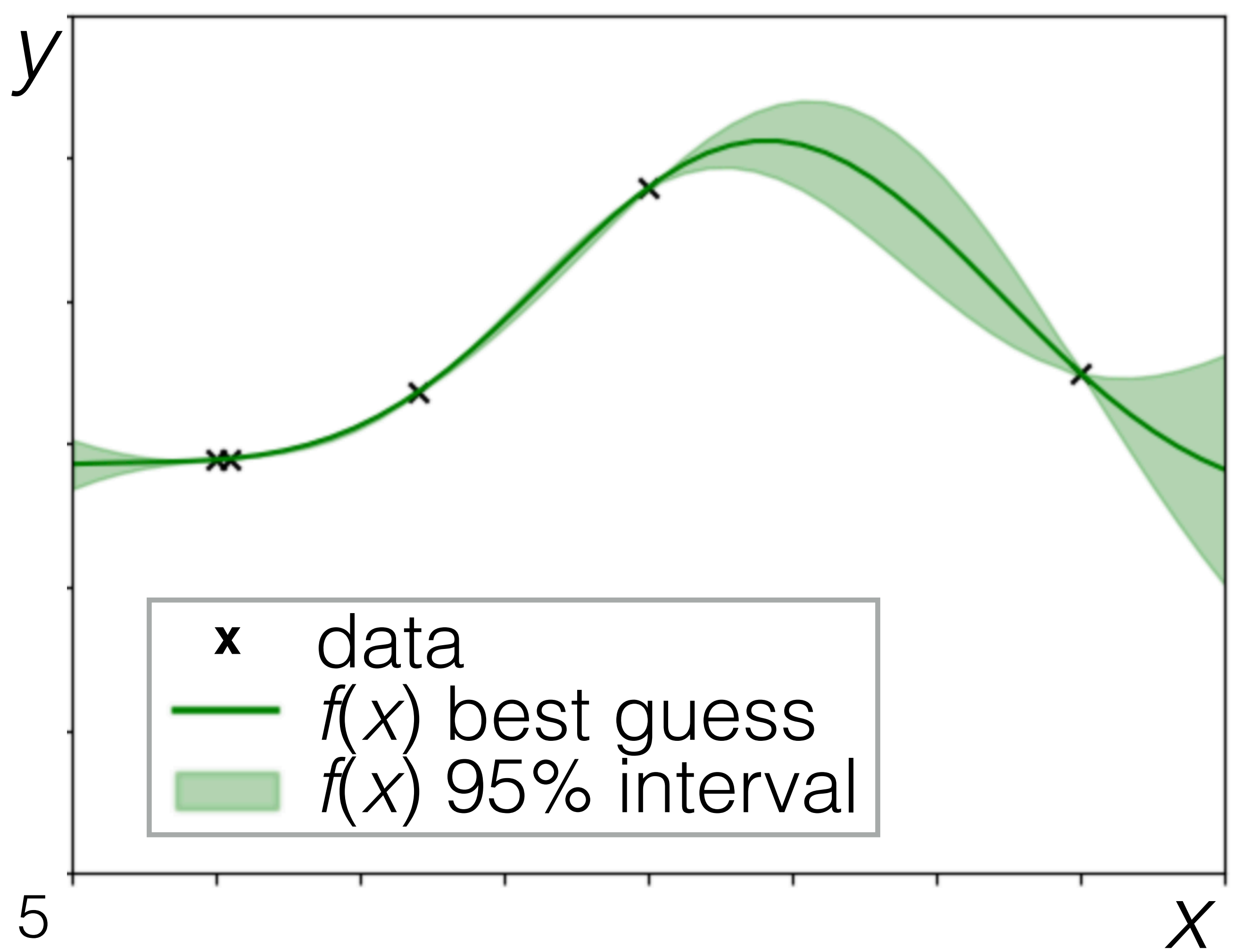


A Bayesian approach

- p(unknowns | data) $\propto$ p(data | unknowns) p(unknowns)

Given the data we've seen, what do we know about the underlying function?

A (stats/AI/ML) model that can generate functions and data of interest

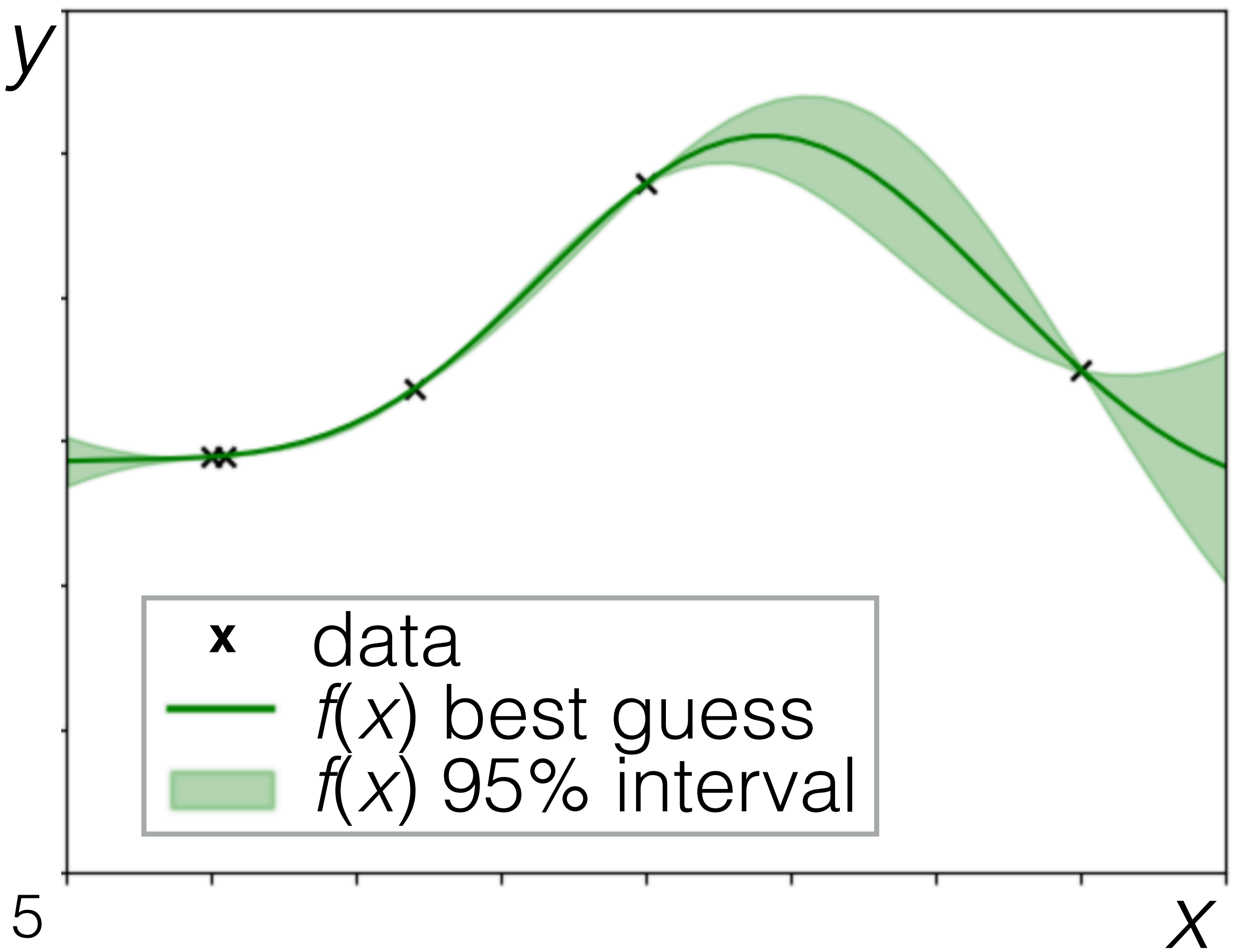


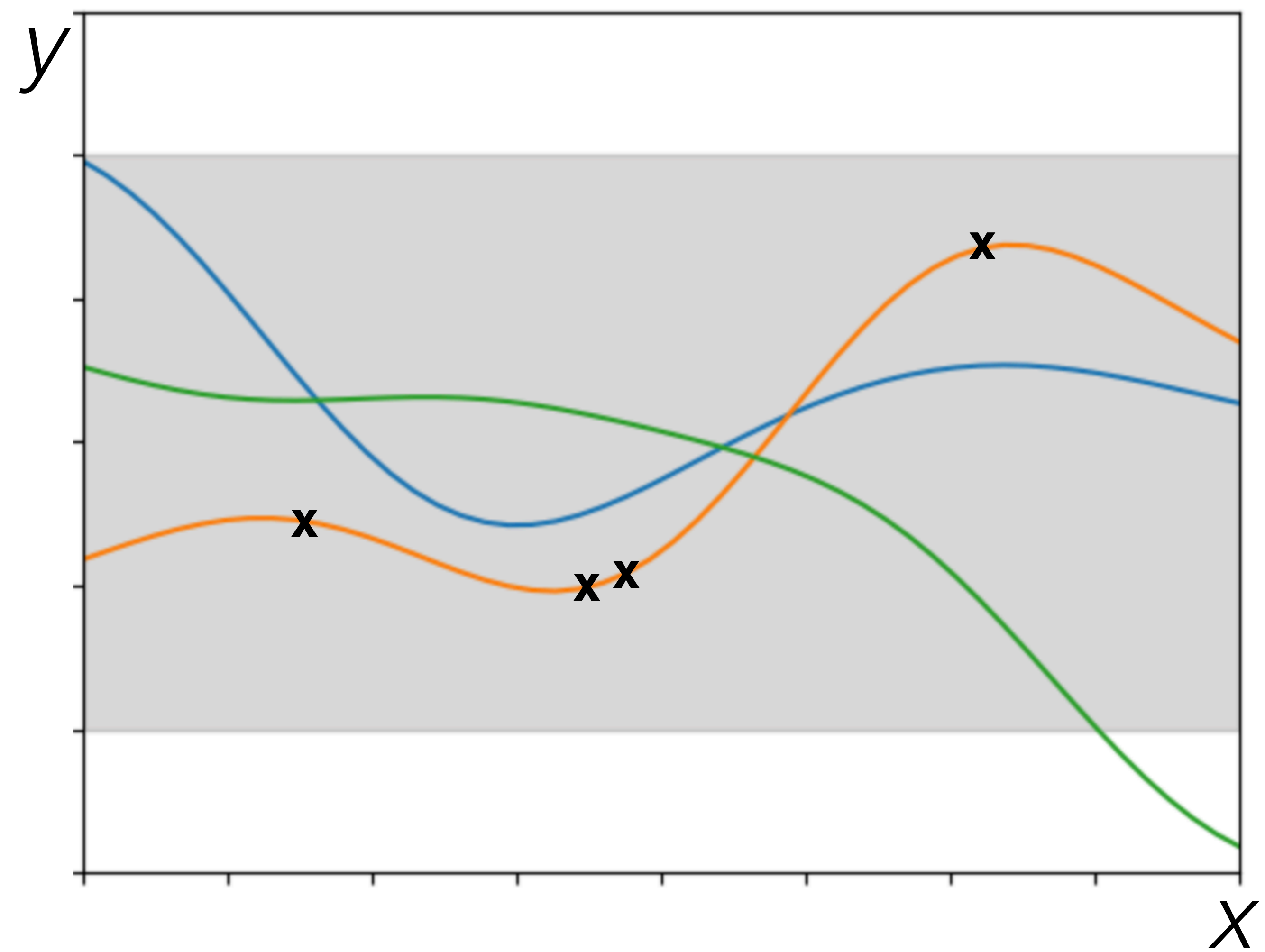
A Bayesian approach

- p(unknowns | data) ∝ p(data | unknowns) p(unknowns)

Given the data we've seen, what do we know about the underlying function?

A (stats/AI/ML) model that can generate functions and data of interest





Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix *K* is symmetric, positive semidefinite

Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix *K* is symmetric, positive semidefinite
  - A common choice:

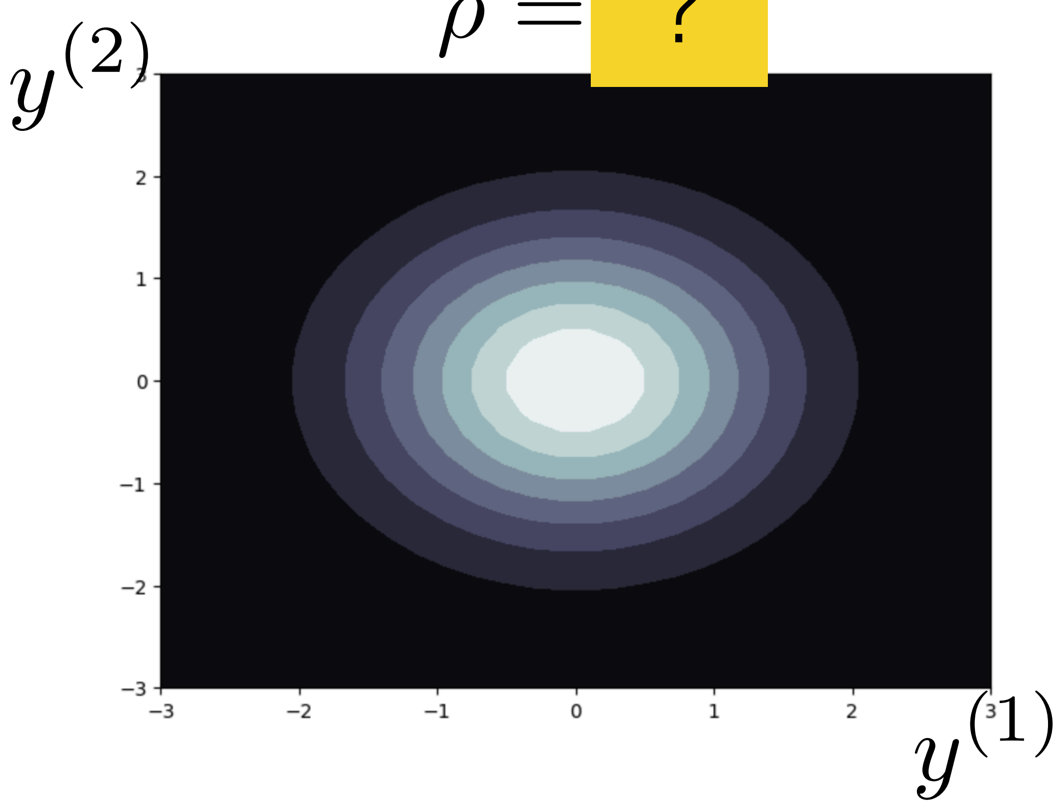
$$K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$$

correlation

Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix *K* is symmetric, positive semidefinite
  - A common choice: $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation
  - Probability density:

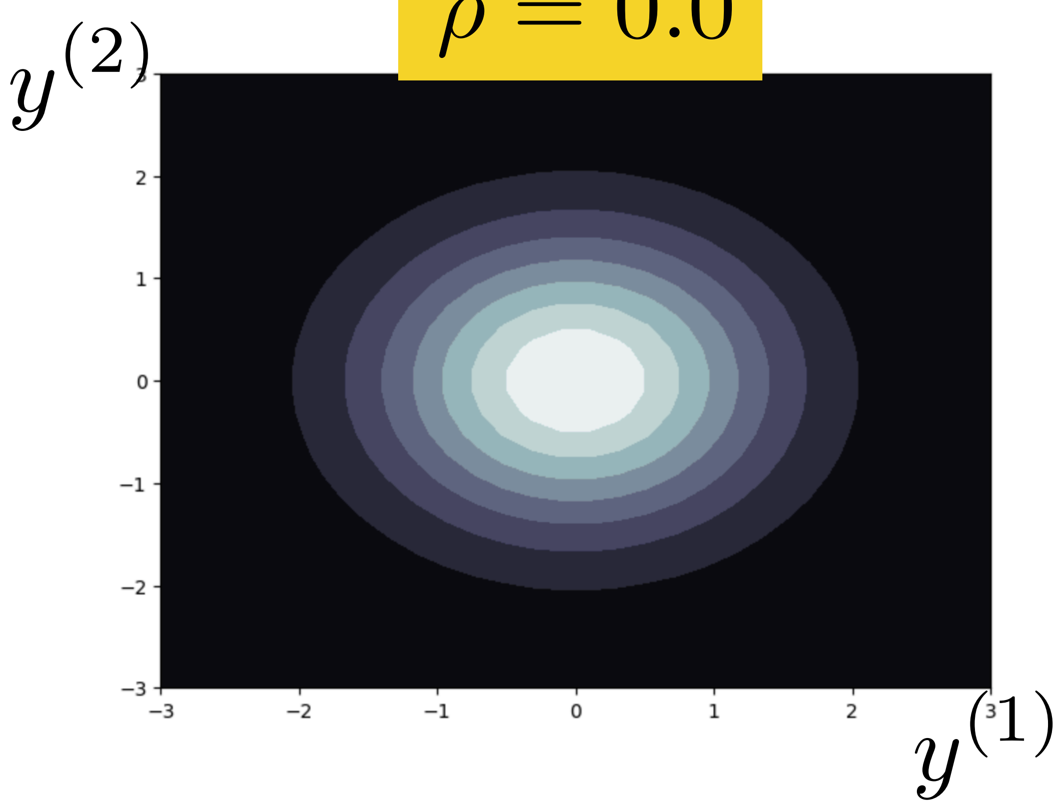
$\rho =$?



Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix *K* is symmetric, positive semidefinite
  - A common choice: $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation
  - Probability density:

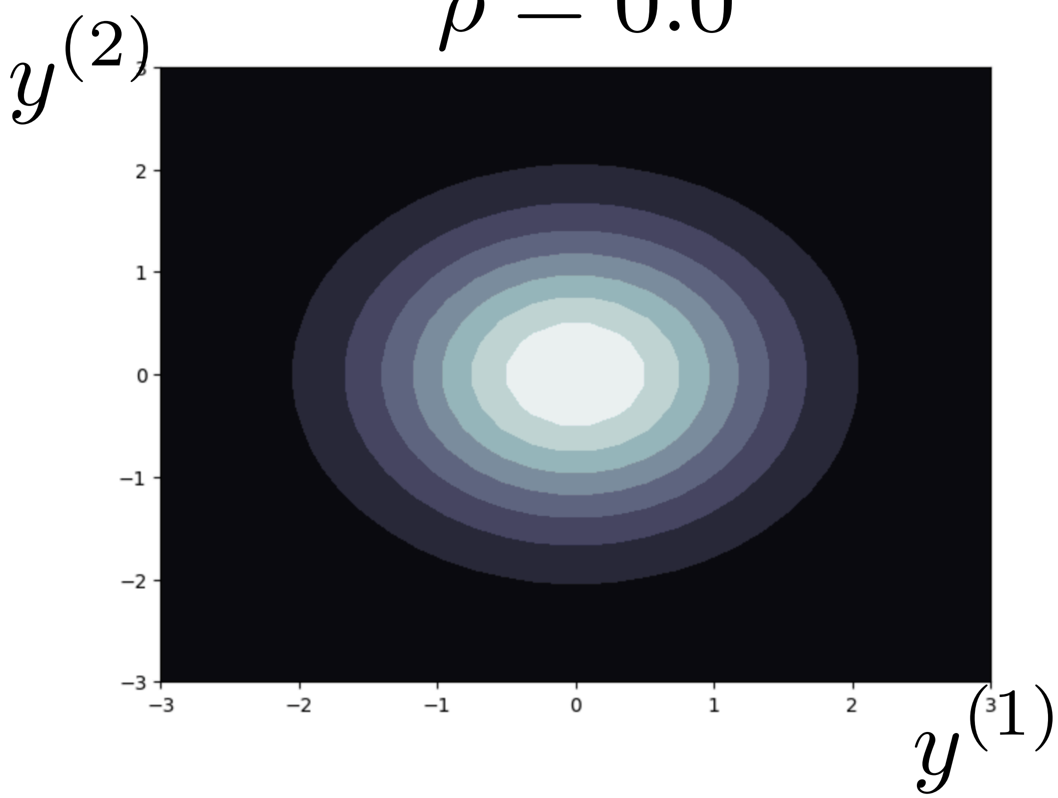
$\rho = 0.0$



Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix *K* is symmetric, positive semidefinite
  - A common choice: $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation
  - Probability density:

$\rho = 0.0$

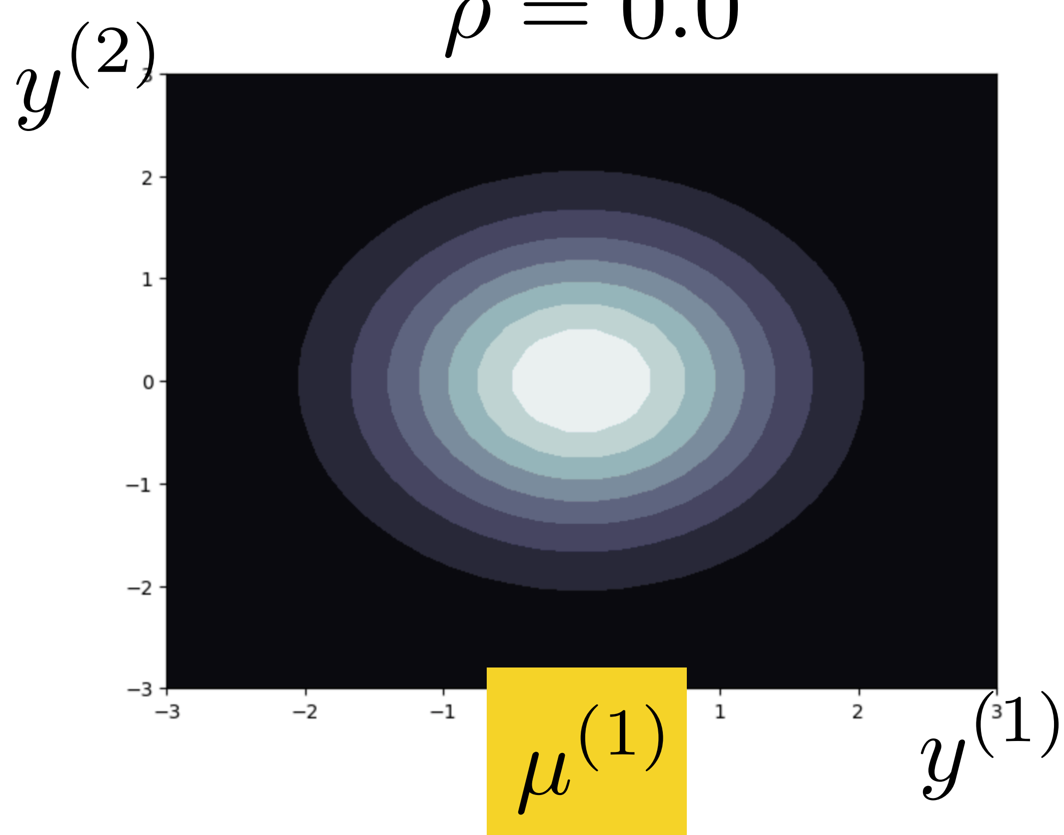


$\mu^{(1)}$?

Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix *K* is symmetric, positive semidefinite
  - A common choice: $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation
  - Probability density:

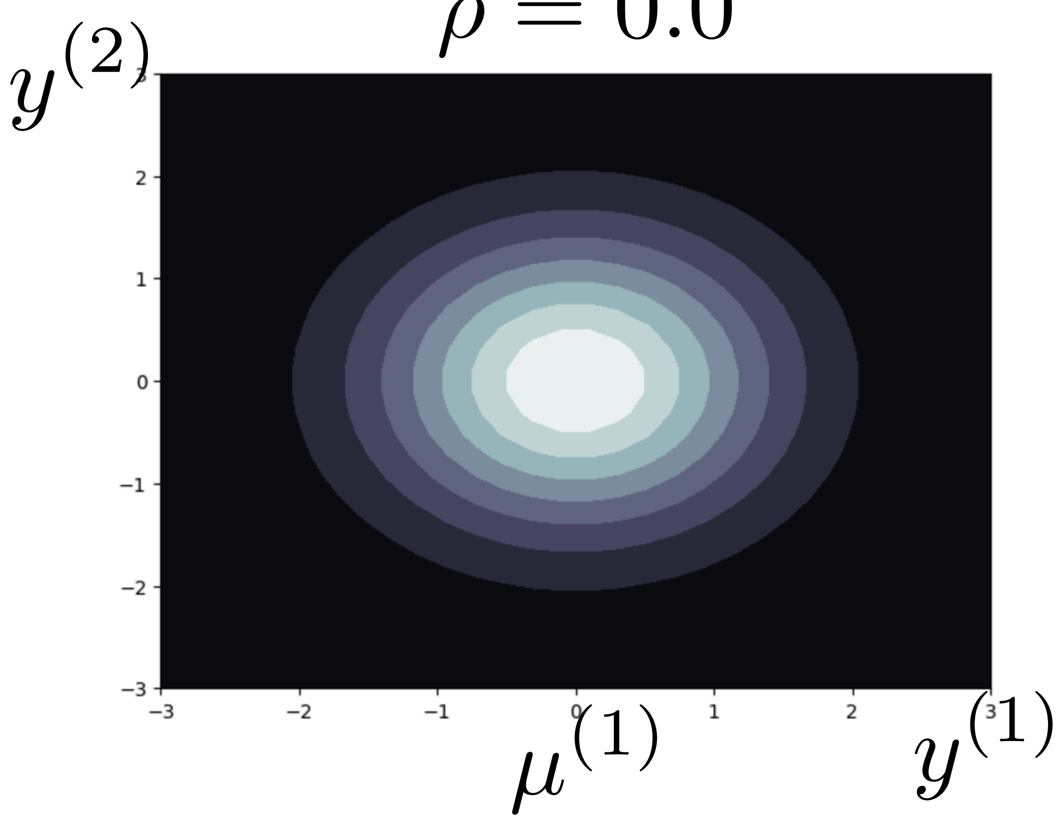
$\rho = 0.0$



Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix *K* is symmetric, positive semidefinite
  - A common choice: $$K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$$ correlation
  - Probability density:

$\rho = 0.0$



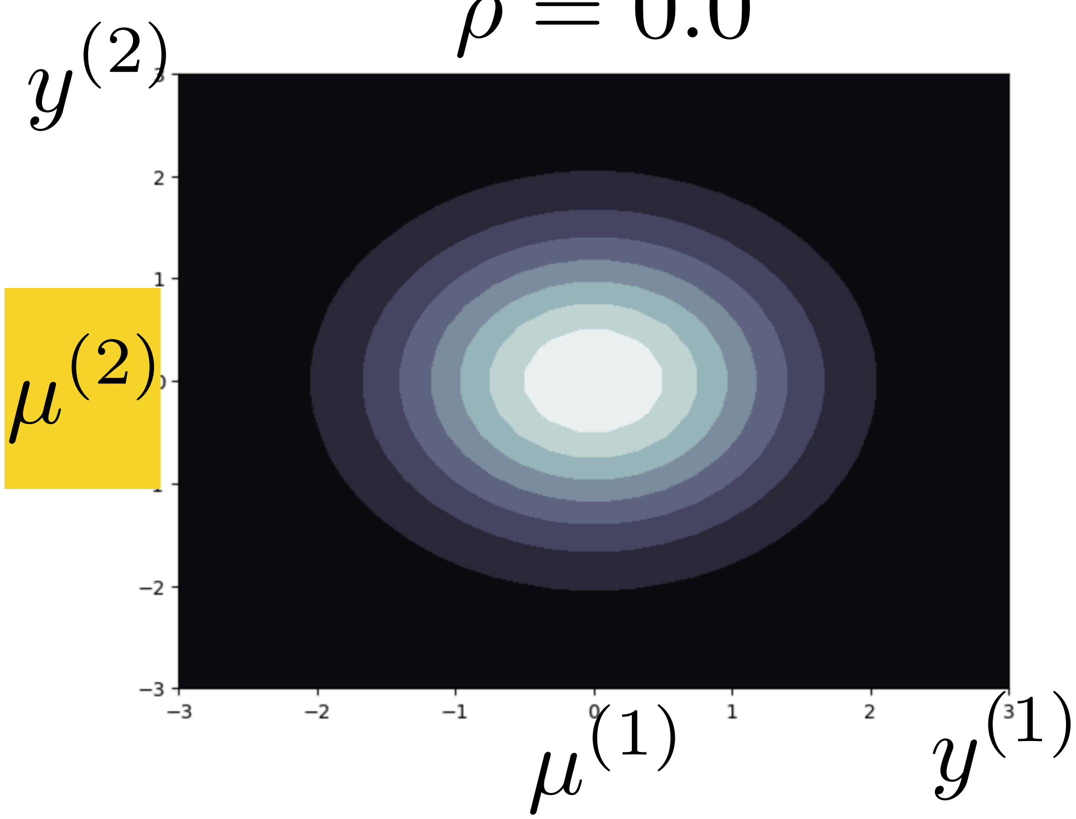
$\mu^{(2)}$?

Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix K is symmetric, positive semidefinite
  - A common choice: $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation

  - Probability density:

$\rho = 0.0$



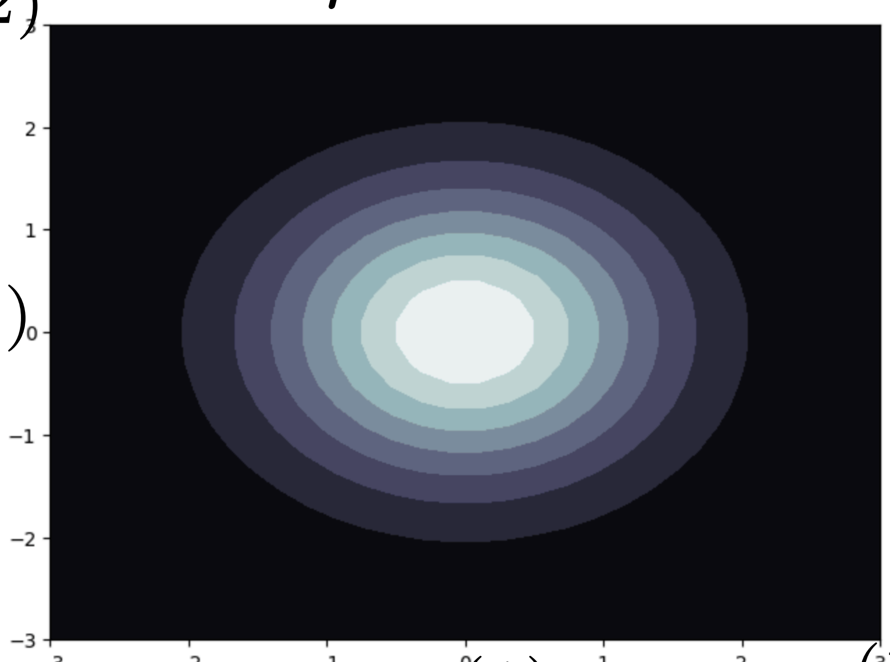
Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix *K* is symmetric, positive semidefinite
  - A common choice: $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation
  - Probability density:

$\rho = 0.0$

$y^{(2)}$

$\mu^{(2)}$



$\mu^{(1)}$ $y^{(1)}$

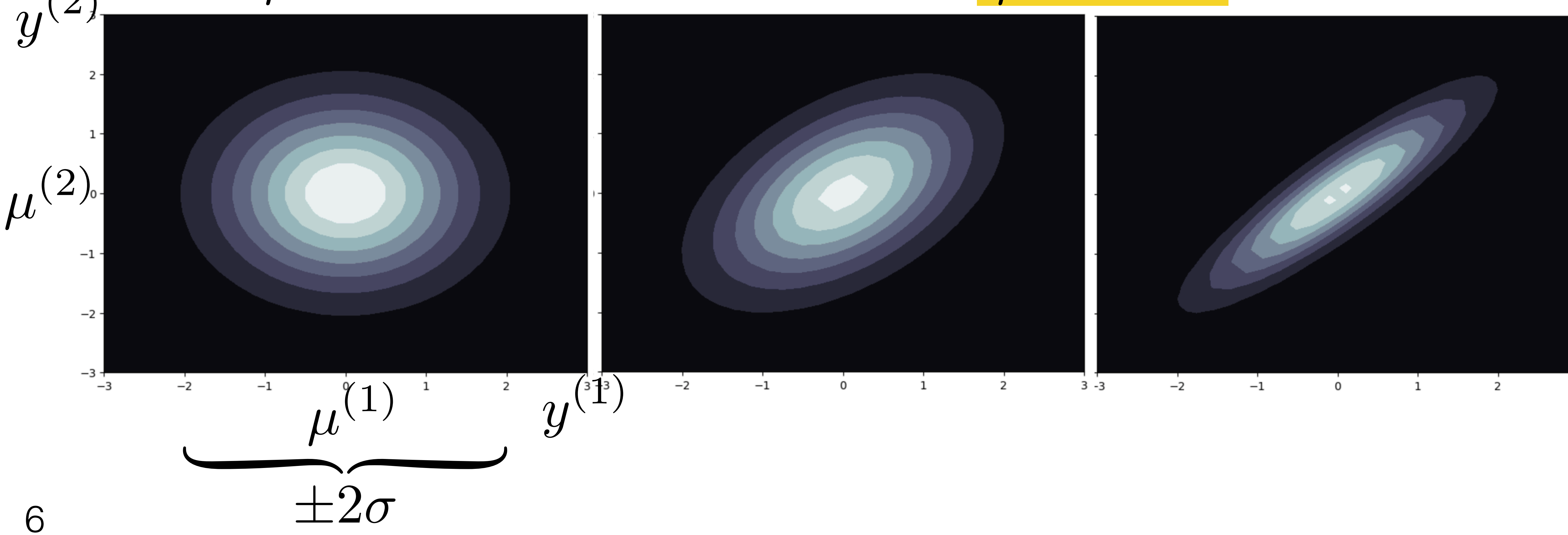
$\underbrace{\qquad\qquad}_{\pm 2\sigma}$

Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix K is symmetric, positive semidefinite
  - A common choice: $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation
  - Probability density:

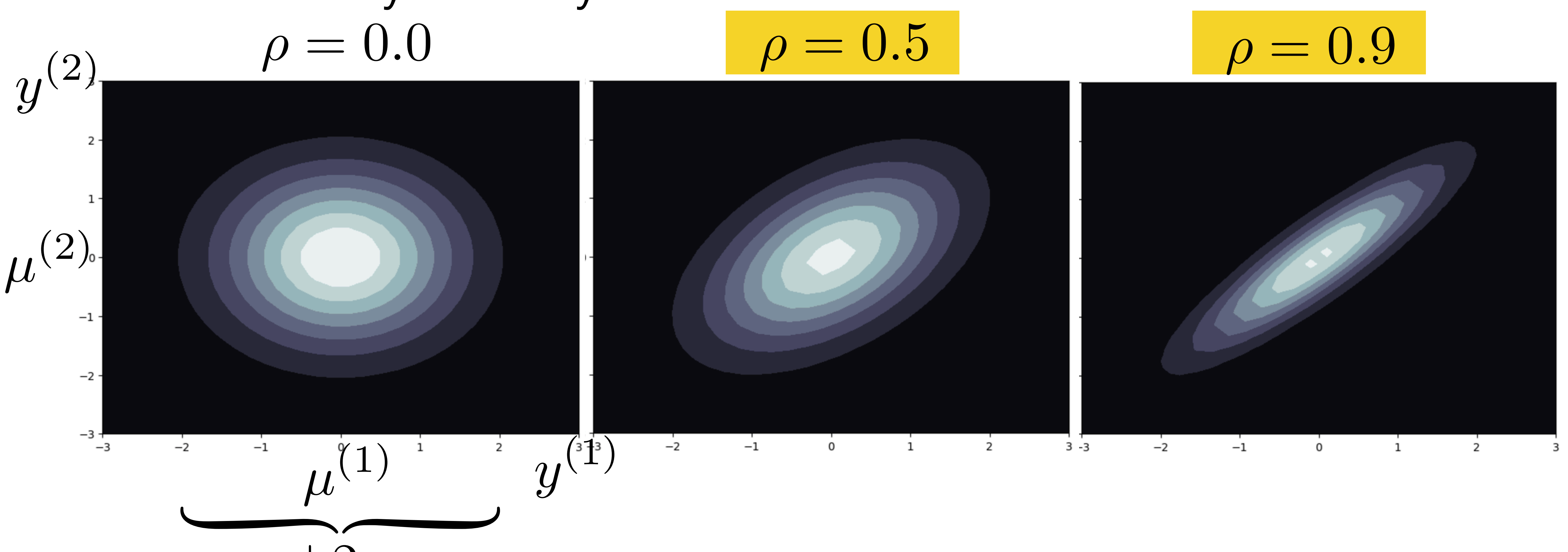
$\rho = 0.0$

$\rho = 0.5$?
$\rho = 0.9$



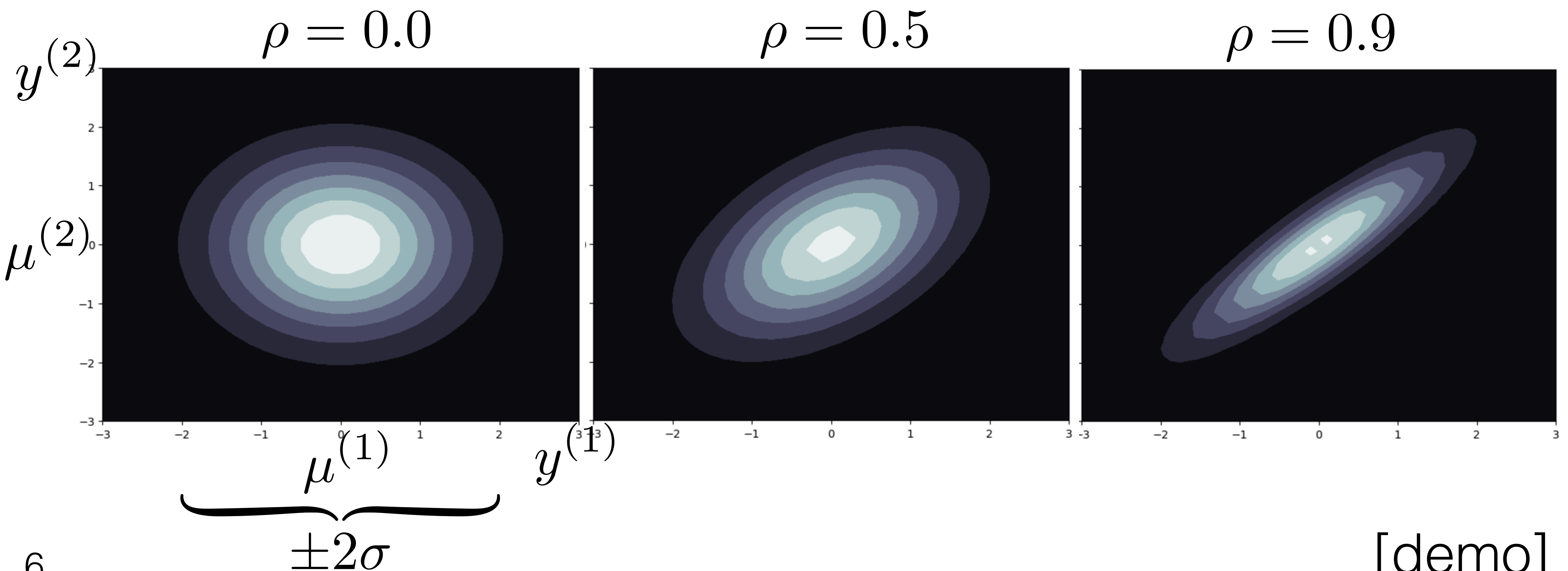
Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix *K* is symmetric, positive semidefinite
  - A common choice: $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation
  - Probability density:



Bivariate Gaussian distribution review

- Gaussian: $y \sim \mathcal{N}(\mu, K)$
  - Observations $y = [y^{(1)}, y^{(2)}]^\top$; mean $\mu = [\mu^{(1)}, \mu^{(2)}]^\top$
  - Covariance matrix *K* is symmetric, positive semidefinite
  - A common choice: $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation
  - Probability density:



[demo]

Bivariate Gaussian using locations

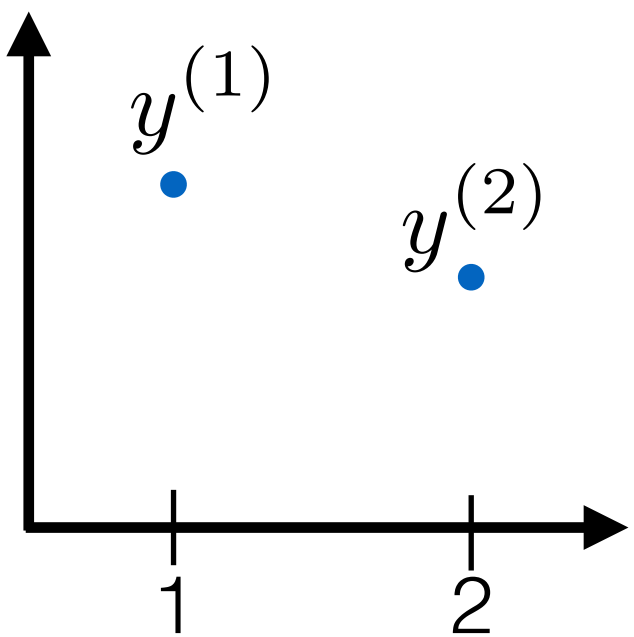
- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$

Bivariate Gaussian using locations

- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$
  - With $\mu = [0, 0]^\top$ and $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation

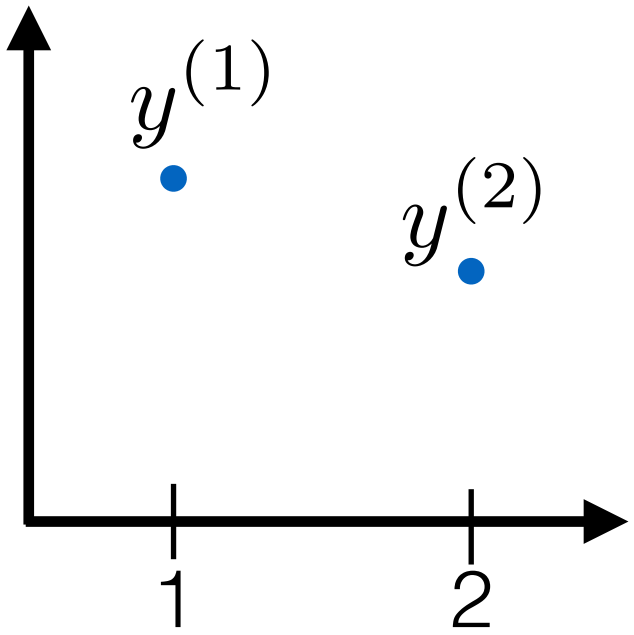
Bivariate Gaussian using locations

- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$
  - With $\mu = [0, 0]^\top$ and $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation



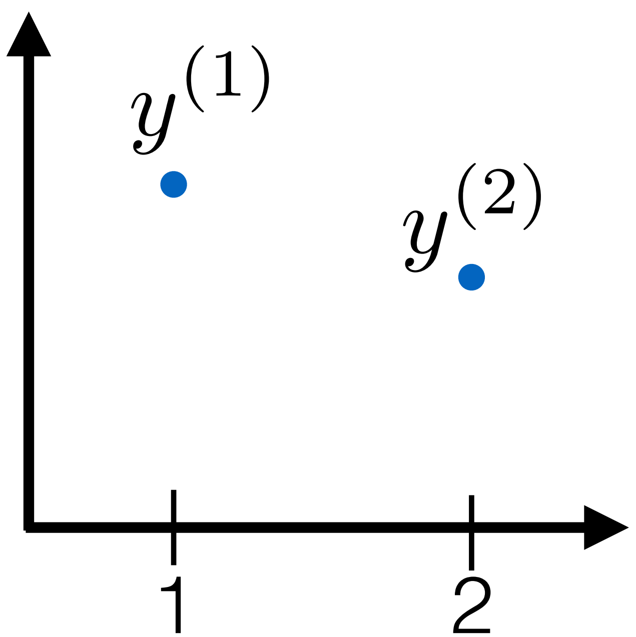
Bivariate Gaussian using locations

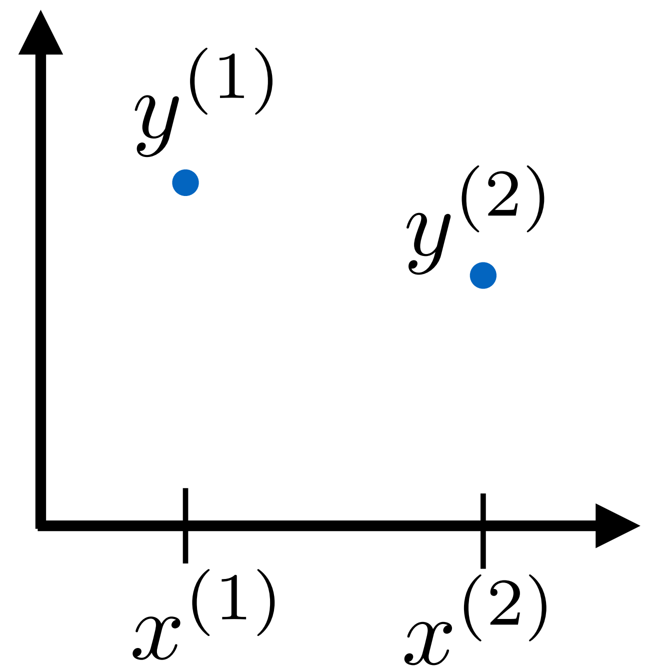
- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$
  - With $\mu = [0, 0]^\top$ and $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation



Bivariate Gaussian using locations

- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$
  - With $\mu = [0, 0]^\top$ and $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation

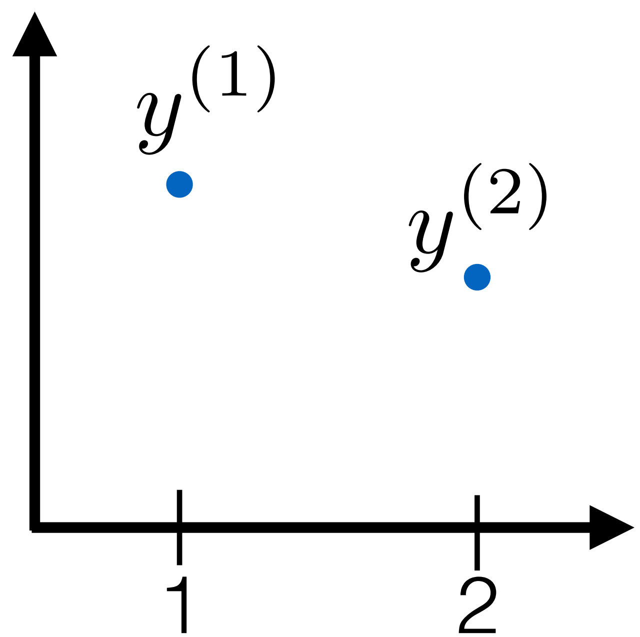


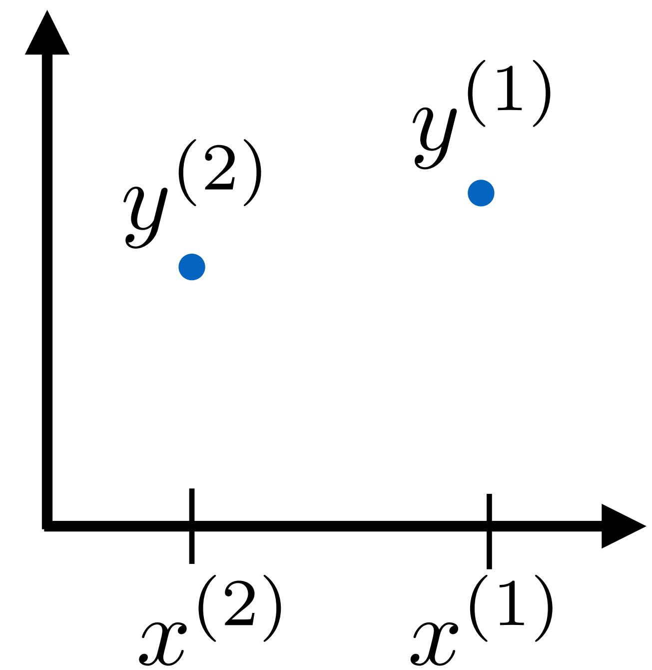


Bivariate Gaussian using locations

- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$
  - With $\mu = [0, 0]^\top$ and $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$

correlation

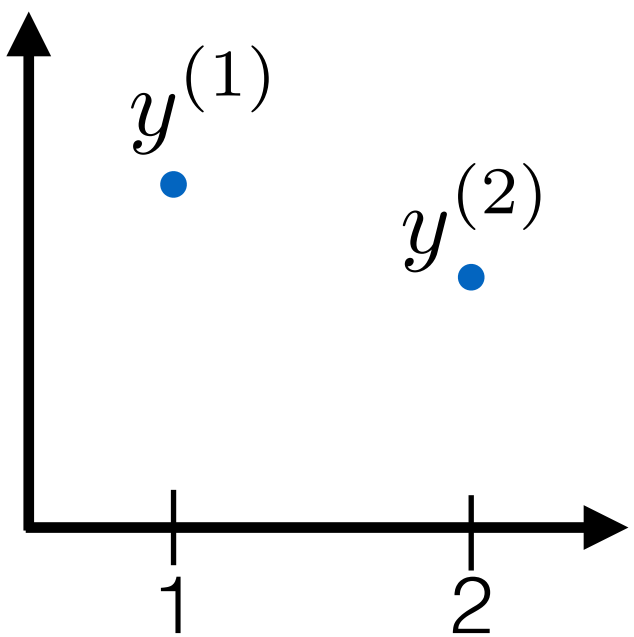


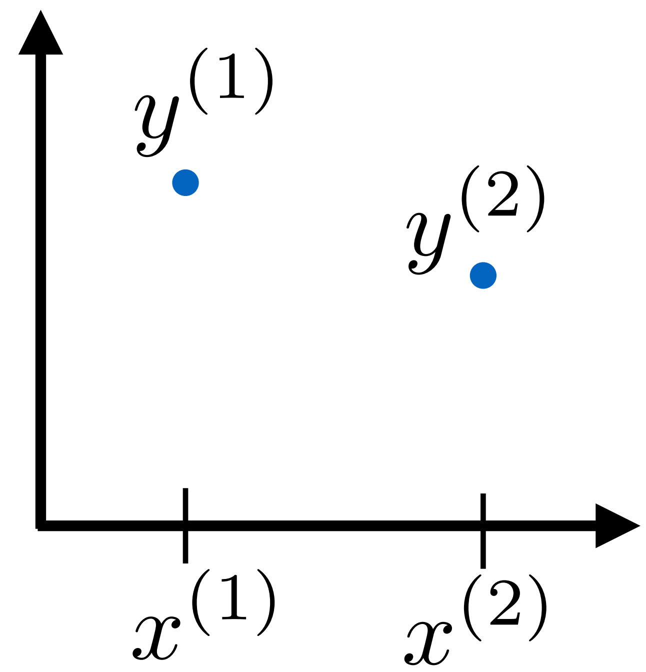


Bivariate Gaussian using locations

- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$
  - With $\mu = [0, 0]^\top$ and $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$

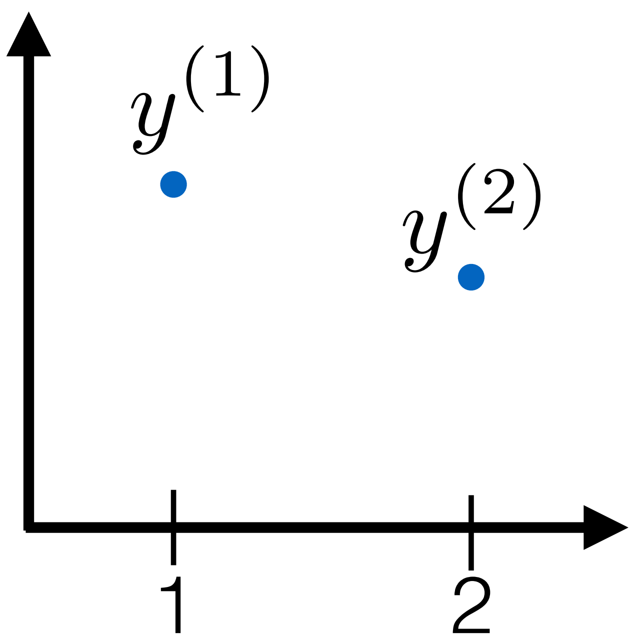
correlation

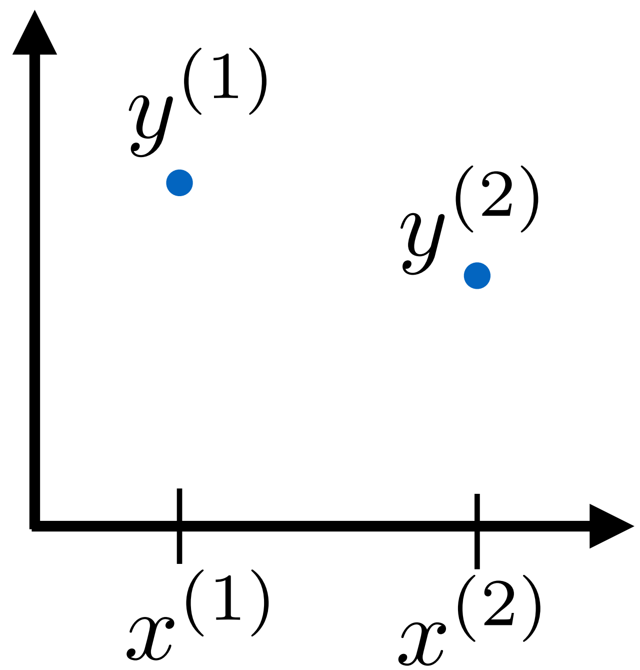




Bivariate Gaussian using locations

- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$
  - With $\mu = [0, 0]^\top$ and $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation

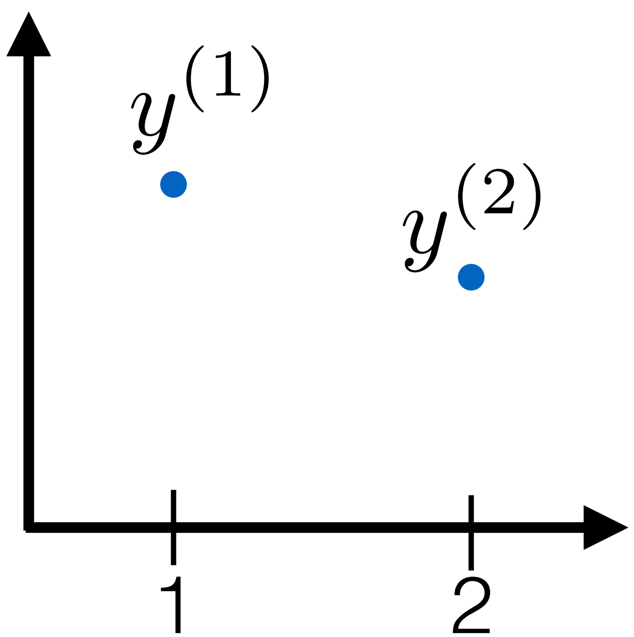


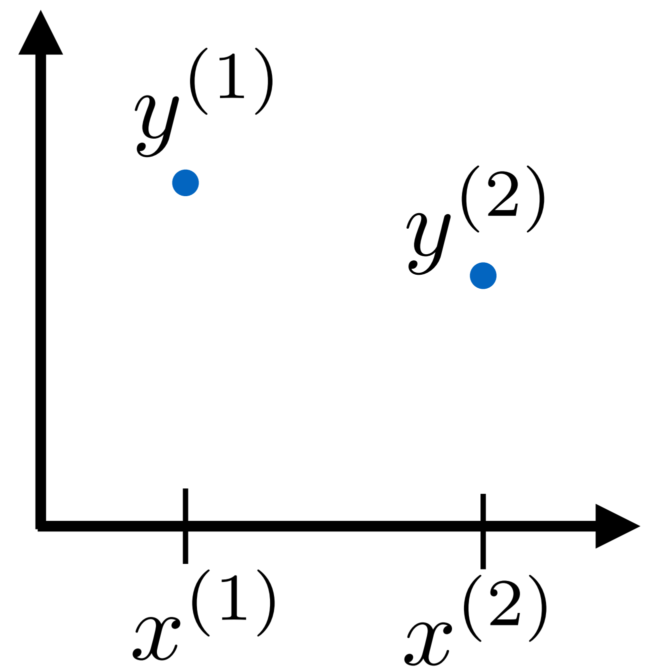


- What if we let the correlation depend on the *x*'s?

Bivariate Gaussian using locations

- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$
  - With $\mu = [0, 0]^\top$ and $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation

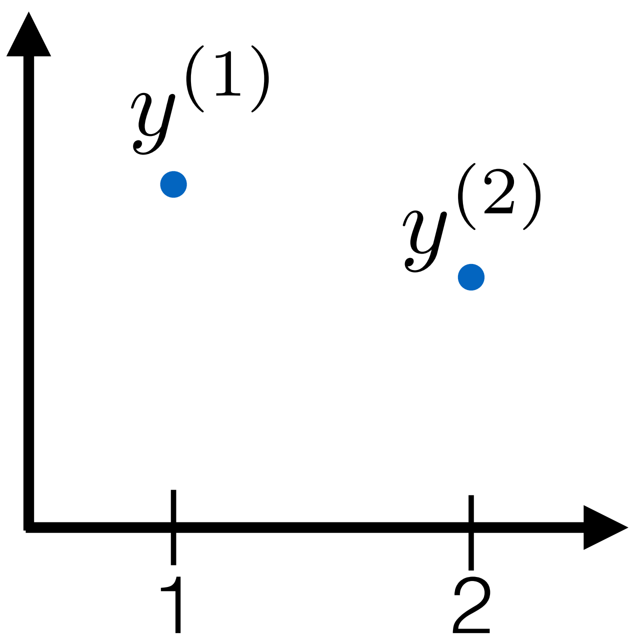


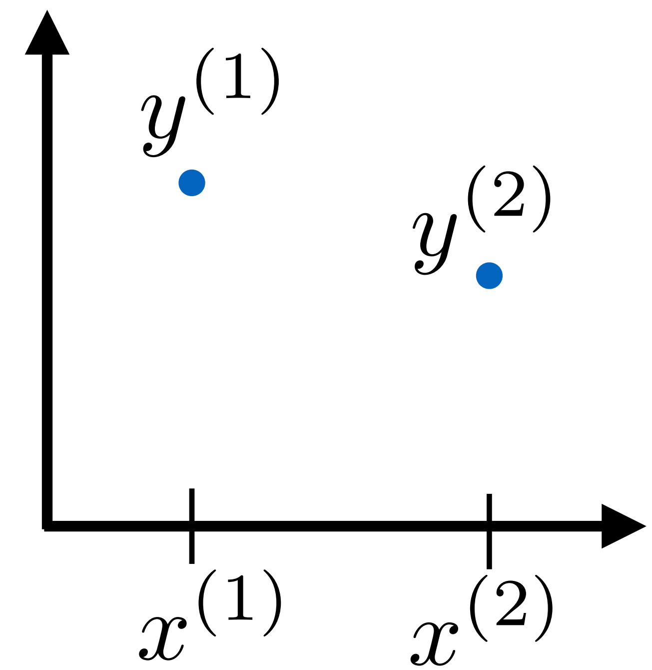


- What if we let the correlation depend on the *x*'s?
  - Let $\rho = \rho(|x^{(1)} - x^{(2)}|)$

Bivariate Gaussian using locations

- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$
  - With $\mu = [0, 0]^\top$ and $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation

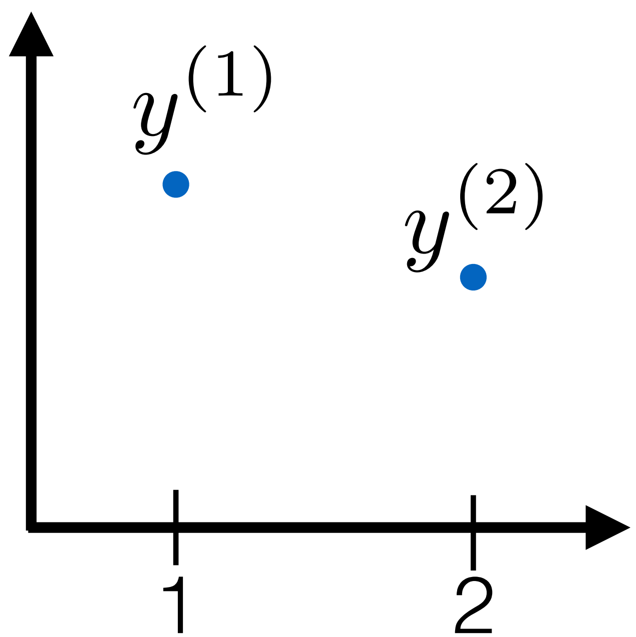


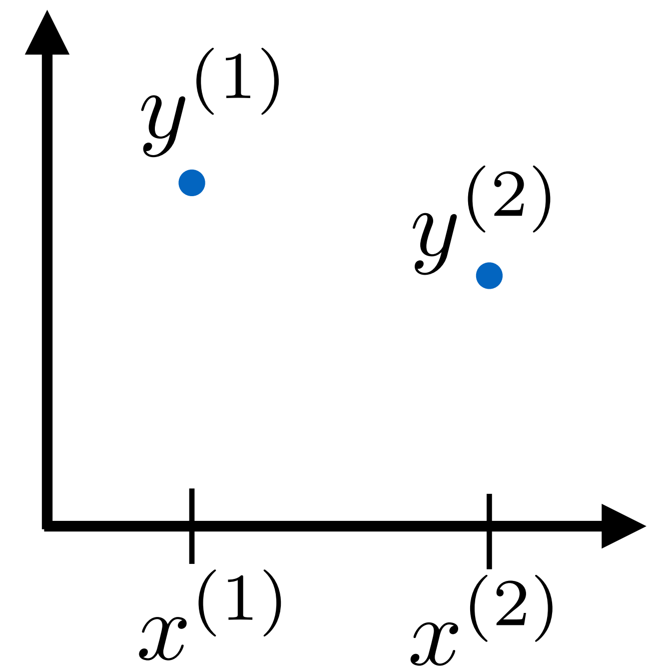


- What if we let the correlation depend on the *x*'s?
  - Let $\rho = \rho(|x^{(1)} - x^{(2)}|)$
    - Where the correlation goes to 1 as the *x*'s get close

Bivariate Gaussian using locations

- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$
  - With $\mu = [0,0]^\top$ and $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation

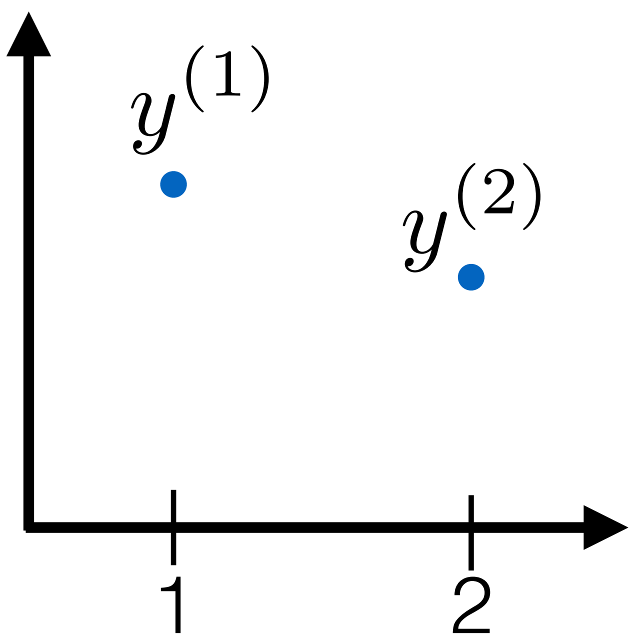


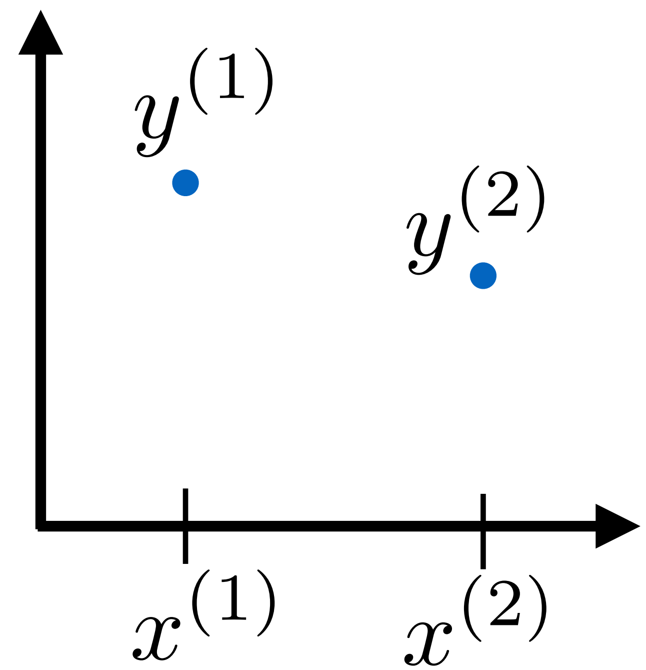


- What if we let the correlation depend on the *x*'s?
  - Let $\rho = \rho(|x^{(1)} - x^{(2)}|)$
    - Where the correlation goes to 1 as the *x*'s get close
    - And goes to 0 as the *x*'s get far

Bivariate Gaussian using locations

- Bivariate Gaussian: $[y^{(1)}, y^{(2)}]^\top \sim \mathcal{N}(\mu, K)$
  - With $\mu = [0, 0]^\top$ and $K = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ correlation





- What if we let the correlation depend on the *x*'s?
  - Let $\rho = \rho(|x^{(1)} - x^{(2)}|)$
    - Where the correlation goes to 1 as the *x*'s get close
    - And goes to 0 as the *x*'s get far

[demo]

Multivariate Gaussian using locations

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):

$$x^{(m)} \in (-\infty, \infty)$$

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
    $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
    $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
  $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
  $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
    $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that
    $$K_{m,m'} = \sigma^2 \rho(|x^{(m)} - x^{(m')}|)$$

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
    $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that
      $$K_{m,m'} = \sigma^2 \rho(|x^{(m)} - x^{(m')}|)$$
    - Let's try $\rho(\Delta) = \exp(-\frac{1}{2}\Delta^2)$

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
  $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that
    $$K_{m,m'} = \sigma^2 \rho(|x^{(m)} - x^{(m')}|)$$
    - Let's try $\rho(\Delta) = \exp(-\frac{1}{2}\Delta^2)$
      - Check: $\rho(0) =$?

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
  $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that
    $$K_{m,m'} = \sigma^2 \rho(|x^{(m)} - x^{(m')}|)$$
    - Let's try $\rho(\Delta) = \exp(-\frac{1}{2}\Delta^2)$
      - Check: $\rho(0) = 1$

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
    $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that
      $$K_{m,m'} = \sigma^2 \rho(|x^{(m)} - x^{(m')}|)$$
    - Let's try $\rho(\Delta) = \exp(-\frac{1}{2}\Delta^2)$
      - Check: $\rho(0) = 1$, $\rho(\Delta)$ is ? as Δ increases

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
    $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that
      $$K_{m,m'} = \sigma^2 \rho(|x^{(m)} - x^{(m')}|)$$
    - Let's try $\rho(\Delta) = \exp(-\frac{1}{2}\Delta^2)$
      - Check: $\rho(0) = 1$, $\rho(\Delta)$ is decreasing as Δ increases

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
    $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that
      $$K_{m,m'} = \sigma^2 \rho(|x^{(m)} - x^{(m')}|)$$
    - Let's try $\rho(\Delta) = \exp(-\frac{1}{2}\Delta^2)$
      - Check: $\rho(0) = 1$, $\rho(\Delta)$ is decreasing as Δ increases
      - And $\rho(\Delta) \to$? as $\Delta \to \infty$

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
    $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that
      $$K_{m,m'} = \sigma^2 \rho(|x^{(m)} - x^{(m')}|)$$
    - Let's try $\rho(\Delta) = \exp(-\frac{1}{2}\Delta^2)$
      - Check: $\rho(0) = 1$, $\rho(\Delta)$ is decreasing as Δ increases
      - And $\rho(\Delta) \to 0$ as $\Delta \to \infty$

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
    $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that
      $$K_{m,m'} = \sigma^2 \rho(|x^{(m)} - x^{(m')}|)$$
    - Let's try $\rho(\Delta) = \exp(-\frac{1}{2}\Delta^2)$
      - Check: $\rho(0) = 1$, $\rho(\Delta)$ is decreasing as Δ increases
      - And $\rho(\Delta) \to 0$ as $\Delta \to \infty$

Note: our previous example was the special case where *M*=2

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
    $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that
      $$K_{m,m'} = \sigma^2 \rho(|x^{(m)} - x^{(m')}|)$$
    - Let's try $\rho(\Delta) = \exp(-\frac{1}{2}\Delta^2)$
      - Check: $\rho(0) = 1$, $\rho(\Delta)$ is decreasing as Δ increases
      - And $\rho(\Delta) \to 0$ as $\Delta \to \infty$

[demo1, demo2]

Note: our previous example was the special case where *M*=2

Multivariate Gaussian using locations

- Next: Similar setup but an *M*-dimensional Gaussian instead of just 2 dimensions
  - We have *M* locations (need not be in order):
  $$x^{(m)} \in (-\infty, \infty)$$
  - We're going to generate a random $y = [y^{(1)}, \ldots, y^{(M)}]^\top$
    - Let $y \sim \mathcal{N}(\mu, K)$ with $\mu = \mathbf{0}_M$ and *K* such that
    $$K_{m,m'} = \sigma^2 \rho(|x^{(m)} - x^{(m')}|)$$
    - Let's try $\rho(\Delta) = \exp(-\frac{1}{2}\Delta^2)$
      - Check: $\rho(0) = 1$, $\rho(\Delta)$ is decreasing as Δ increases
      - And $\rho(\Delta) \to 0$ as $\Delta \to \infty$

[demo1, demo2]

Note: our previous example was the special case where *M*=2

We just drew random functions from a type of Gaussian process that is very commonly used in practice!