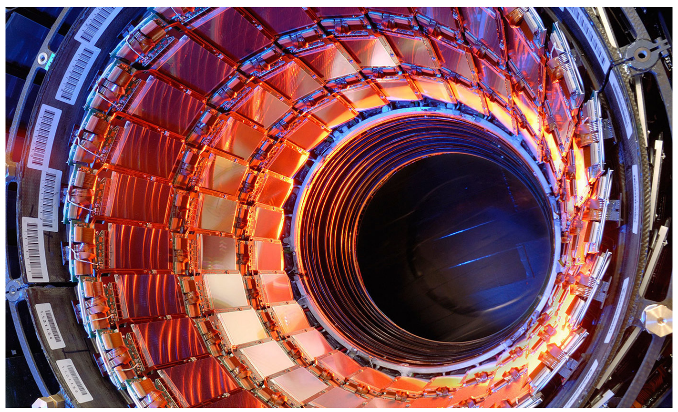
Scalable, reliably accurate Bayesian inference via approximate likelihoods and random features

Jonathan Huggins

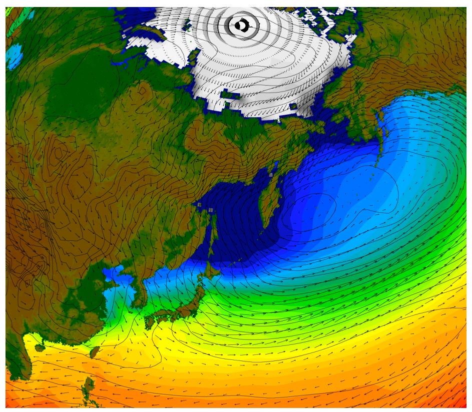
Harvard University

Scalable *and* reliably accurate inference?

Large-scale data analysis for high-impact decision-making is widespread



Physics



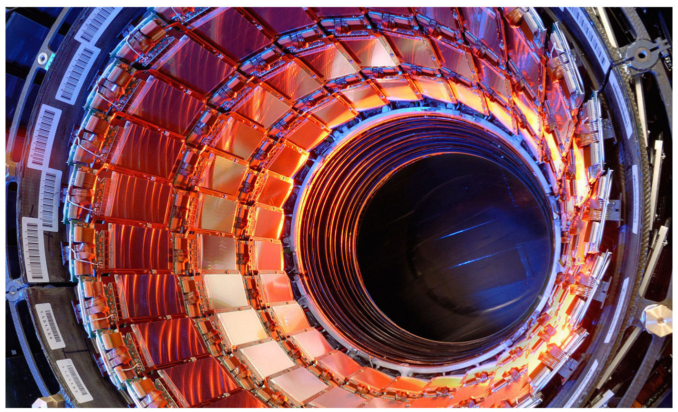
Climate

Medicine

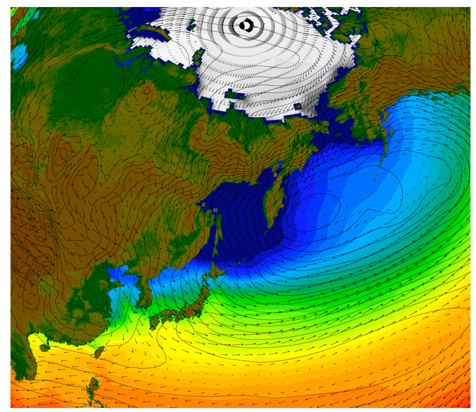
Economics

Scalable *and* reliably accurate inference?

Large-scale data analysis for high-impact decision-making is widespread



Physics



Climate

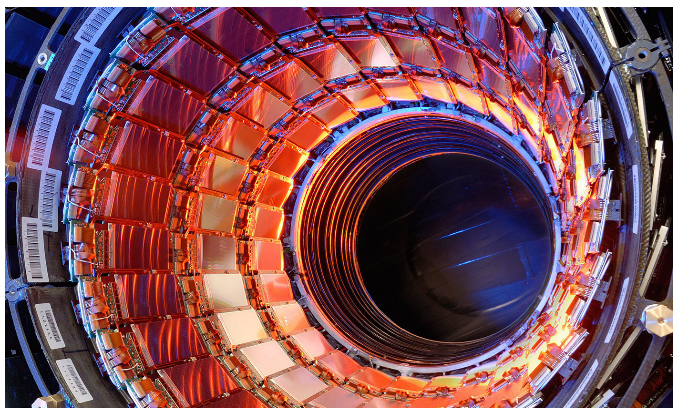
Medicine

Economics

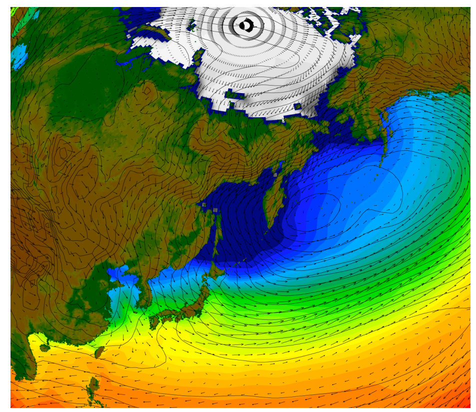
- **Common theme:** need scalability and accuracy

Scalable *and* reliably accurate inference?

Large-scale data analysis for high-impact decision-making is widespread



Physics



Climate

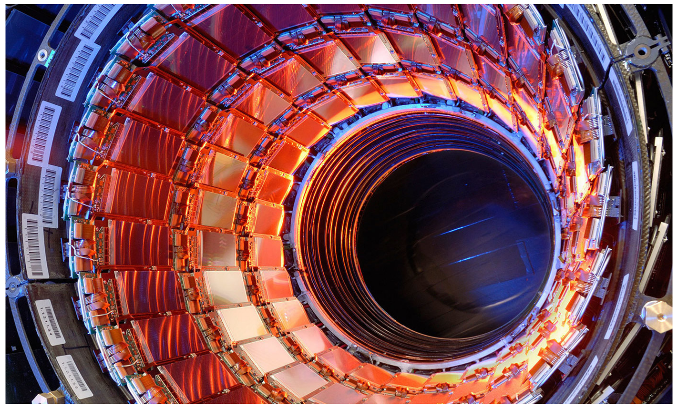
Medicine

Economics

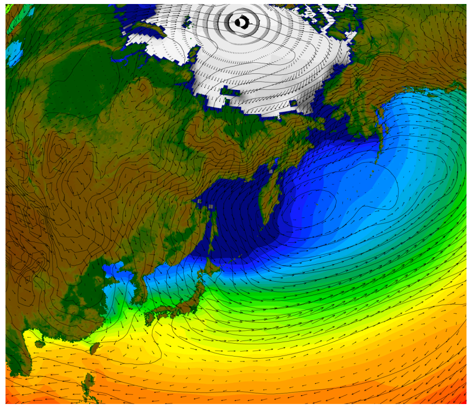
- **Common theme:** need scalability and accuracy
- **Challenge:** scalability and accuracy are competing goals

Scalable *and* reliably accurate inference?

Large-scale data analysis for high-impact decision-making is widespread



Physics



Climate

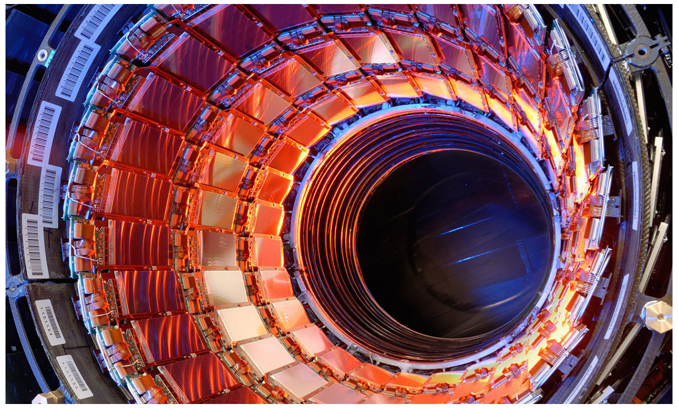
Medicine

Economics

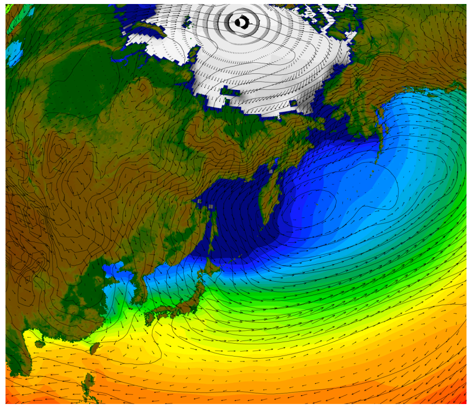
- **Common theme:** need scalability and accuracy
- **Challenge:** scalability and accuracy are competing goals
- "Eager" approach: scalable methods with *pre-specified* guarantees

Scalable *and* reliably accurate inference?

Large-scale data analysis for high-impact decision-making is widespread



Physics



Climate

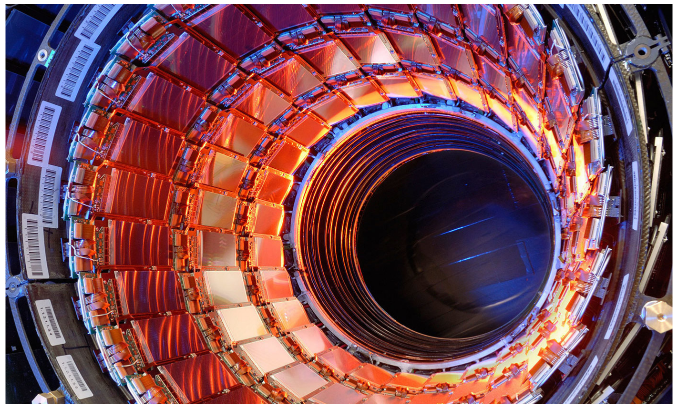
Medicine

Economics

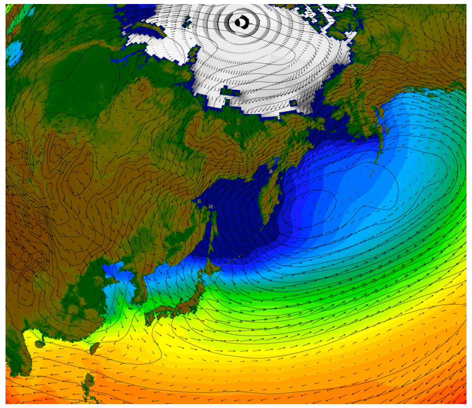
- **Common theme:** need scalability and accuracy
- **Challenge:** scalability and accuracy are competing goals
- "Eager" approach: scalable methods with *pre-specified* guarantees
- "Lazy" approach: validate algorithm's output *post hoc*

Scalable *and* reliably accurate inference?

Large-scale data analysis for high-impact decision-making is widespread



Physics



Climate

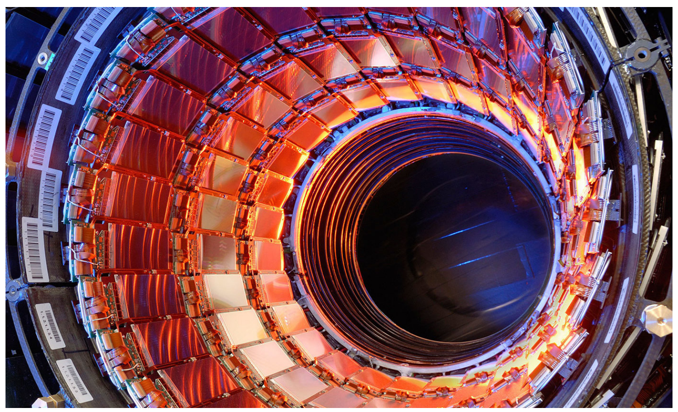
Medicine

Economics

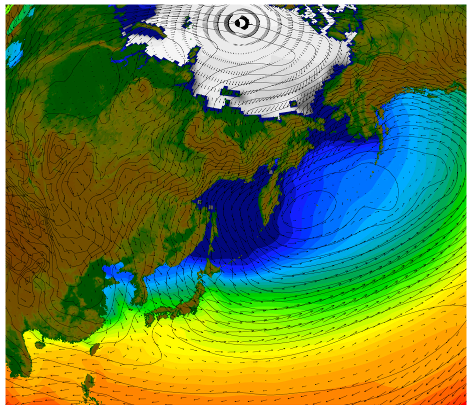
- **Common theme:** need scalability and accuracy
- **Challenge:** scalability and accuracy are competing goals
- “Eager” approach: scalable methods with *pre-specified* guarantees
- “Lazy” approach: validate algorithm’s output *post hoc*
- **Bayesian inference:** flexible modeling of data and uncertainty quantification

Scalable *and* reliably accurate inference?

Large-scale data analysis for high-impact decision-making is widespread



Physics



Climate

Medicine

Economics

- **Common theme:** need scalability and accuracy
- **Challenge:** scalability and accuracy are competing goals
- "Eager" approach: scalable methods with *pre-specified* guarantees
- "Lazy" approach: validate algorithm's output *post hoc*
- **Bayesian inference:** flexible modeling of data and uncertainty quantification
- **This talk:** scalable and accurate Bayesian inference

Bayesian inference

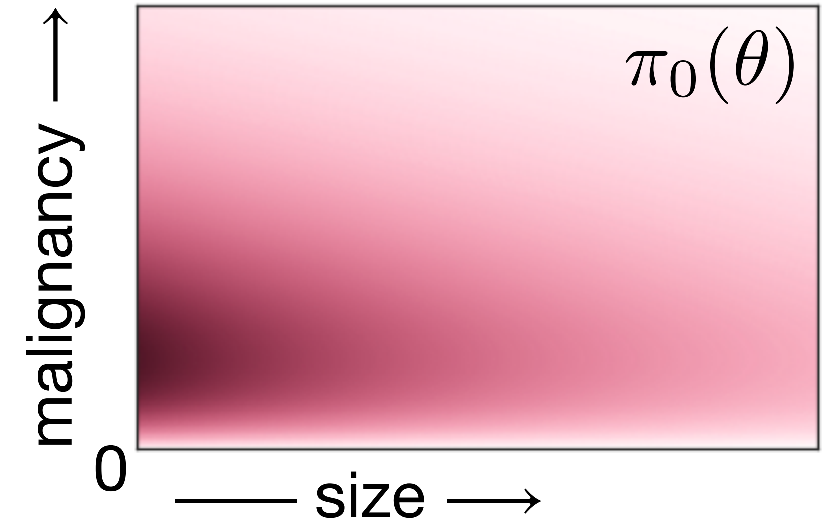
Bayesian inference

- **Goal:** learn about unobserved phenomenon (parameter) of interest θ **[e.g. tumor size & malignancy]**

Bayesian inference

- **Goal:** learn about unobserved phenomenon (parameter) of interest θ **[e.g. tumor size & malignancy]**

- Prior (expert) beliefs $\pi_0(\theta)$ about the phenomenon

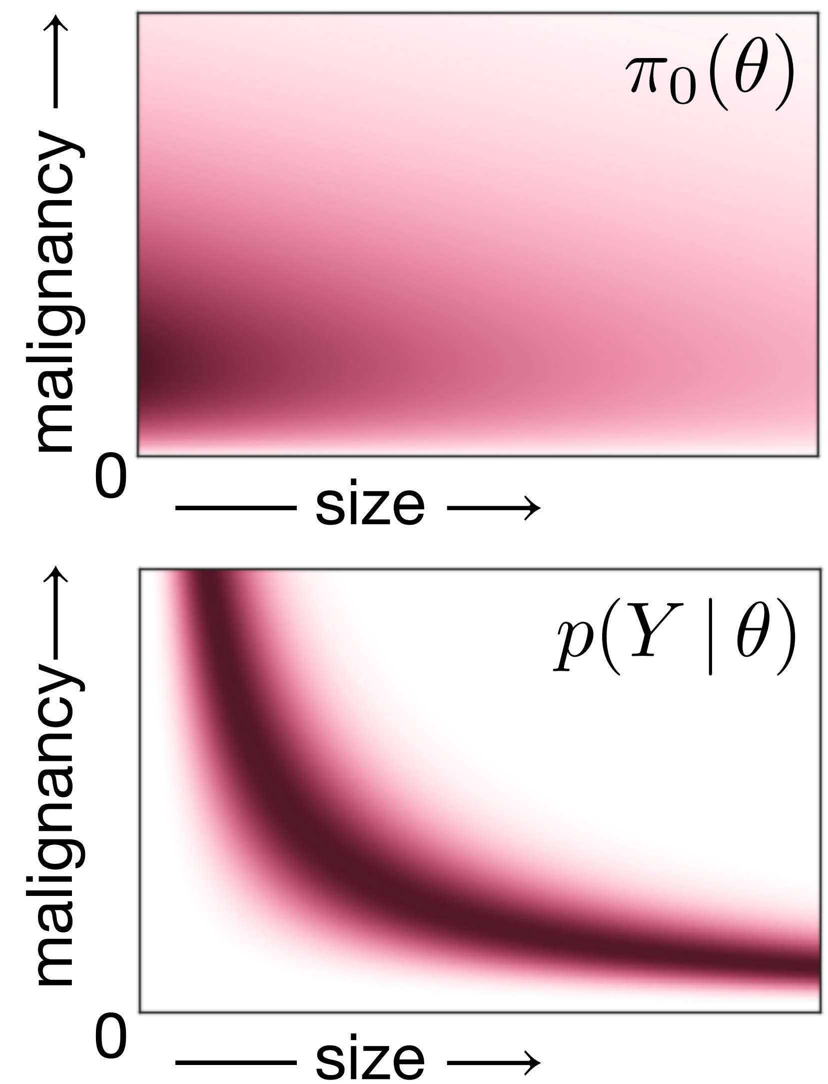


Bayesian inference

- **Goal:** learn about unobserved phenomenon (parameter) of interest θ **[e.g. tumor size & malignancy]**

- Prior (expert) beliefs $\pi_0(\theta)$ about the phenomenon

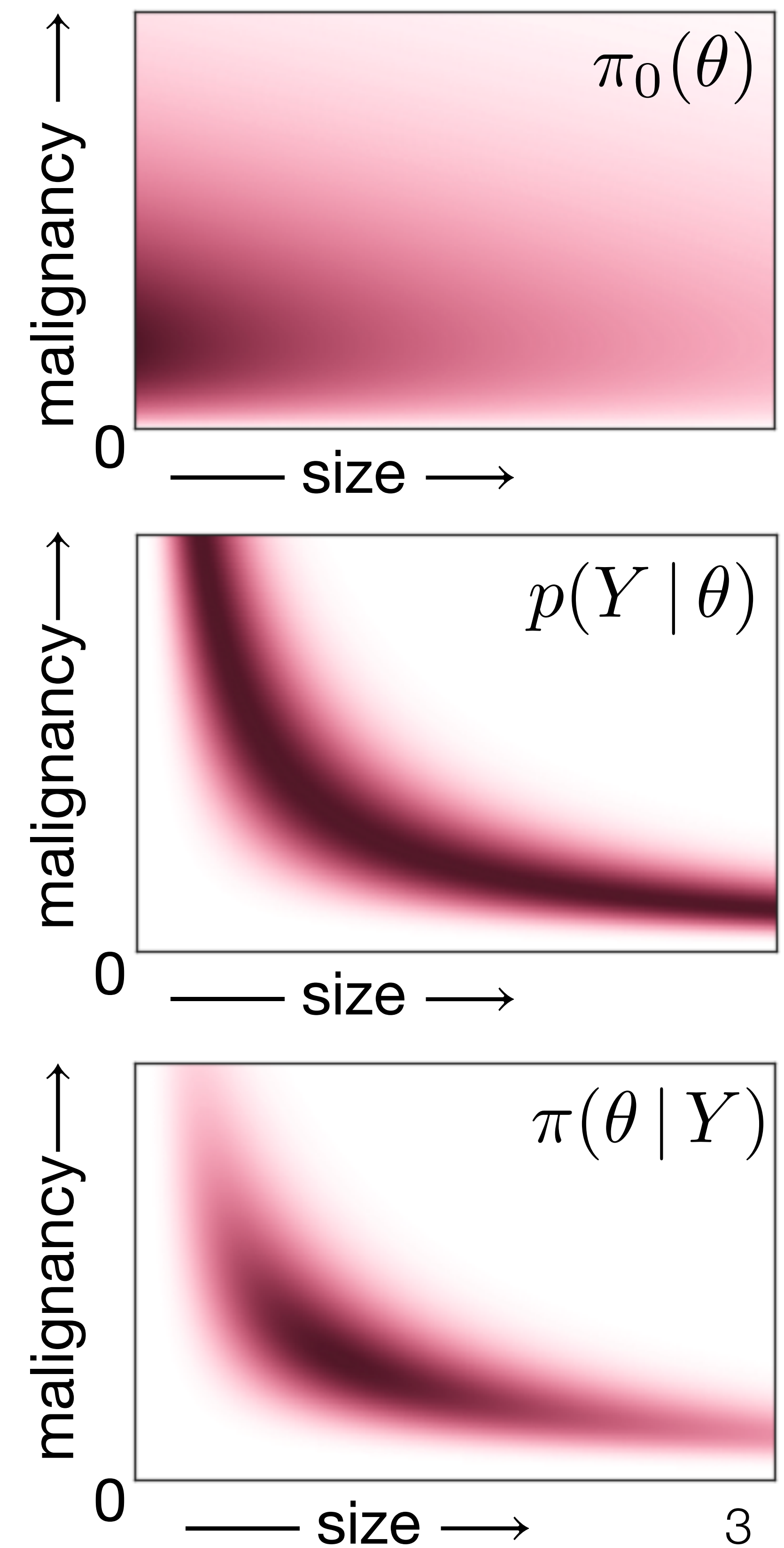
- Observe data Y via measurement process $p(Y \mid \theta)$ **[e.g. ultrasound, biopsy]**



Bayesian inference

- **Goal:** learn about unobserved phenomenon (parameter) of interest θ **[e.g. tumor size & malignancy]**
- Prior (expert) beliefs $\pi_0(\theta)$ about the phenomenon
- Observe data Y via measurement process $p(Y \mid \theta)$ **[e.g. ultrasound, biopsy]**
- Combine prior and observed data to form posterior distribution via **Bayes' Theorem:**

$$\pi(\theta \mid Y) \propto p(Y \mid \theta)\pi_0(\theta)$$



Bayesian inference

- **Goal:** learn about unobserved phenomenon (parameter) of interest θ **[e.g. tumor size & malignancy]**

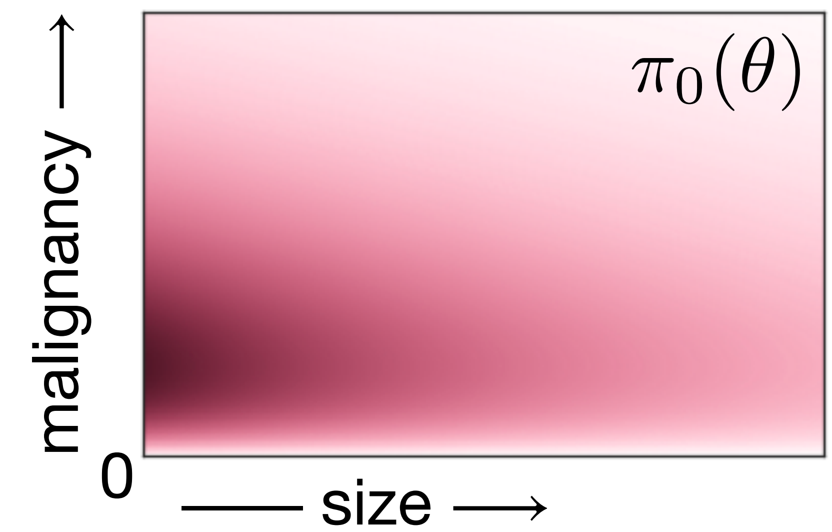
- Prior (expert) beliefs $\pi_0(\theta)$ about the phenomenon

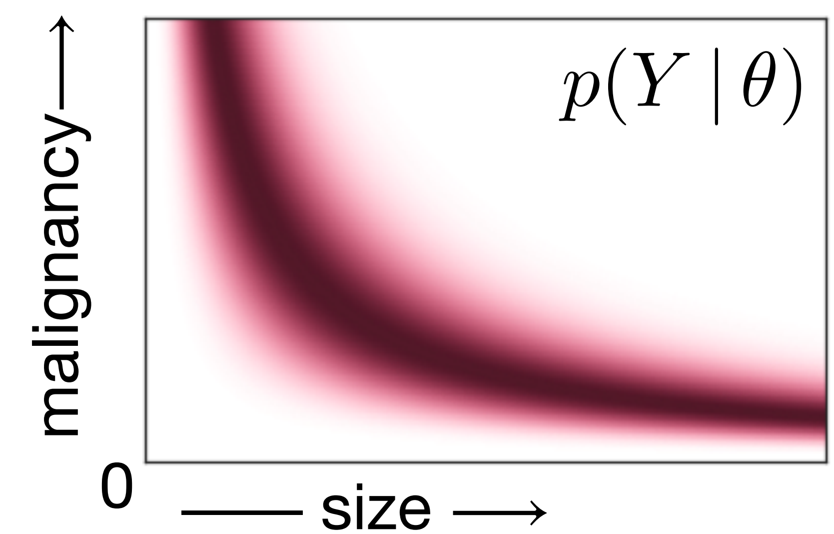
- Observe data Y via measurement process $p(Y \mid \theta)$ **[e.g. ultrasound, biopsy]**

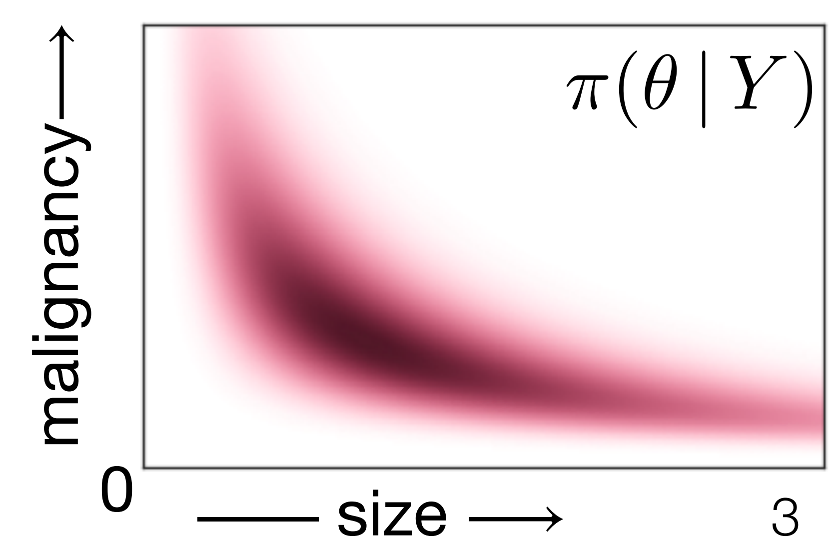
- Combine prior and observed data to form posterior distribution via **Bayes' Theorem:**

$$\pi(\theta \mid Y) \propto p(Y \mid \theta)\pi_0(\theta)$$

- **Benefits:** coherent belief updates, uncertainty quantification, flexible modeling, and more







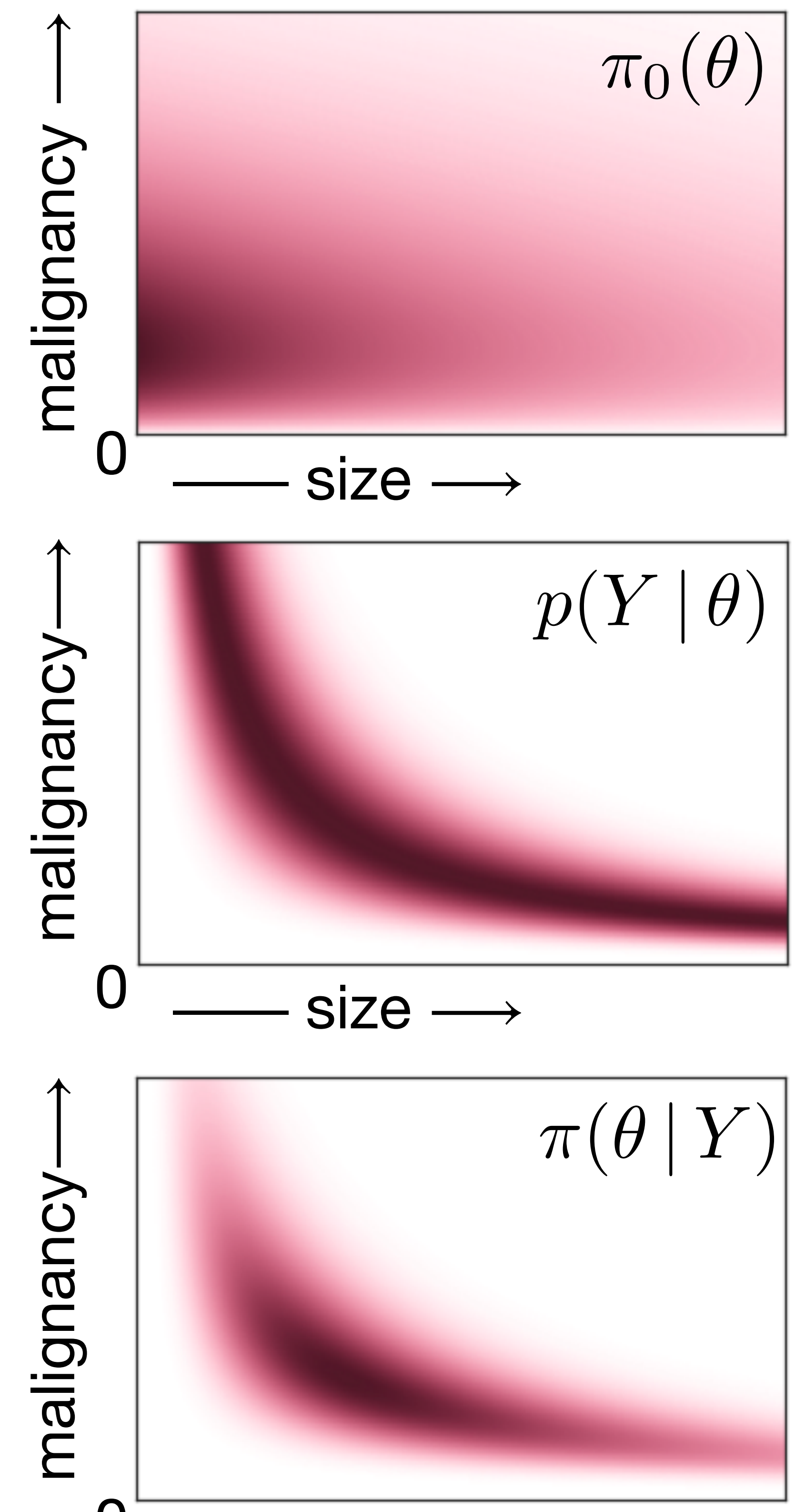
Bayesian inference

- **Goal:** learn about unobserved phenomenon (parameter) of interest θ **[e.g. tumor size & malignancy]**
- Prior (expert) beliefs $\pi_0(\theta)$ about the phenomenon
- Observe data Y via measurement process $p(Y \mid \theta)$ **[e.g. ultrasound, biopsy]**
- Combine prior and observed data to form posterior distribution via **Bayes' Theorem:**

$$\pi(\theta \mid Y) \propto p(Y \mid \theta)\pi_0(\theta)$$

- **Benefits:** coherent belief updates, uncertainty quantification, flexible modeling, and more
- Extract actionable information by **computing expectations** **[e.g. means and standard deviations]**:

$$\mathbb{E}[f(\theta) \mid Y] = \int f(\theta)\pi(\theta \mid Y)\mathrm{d}\theta$$



Bayesian inference

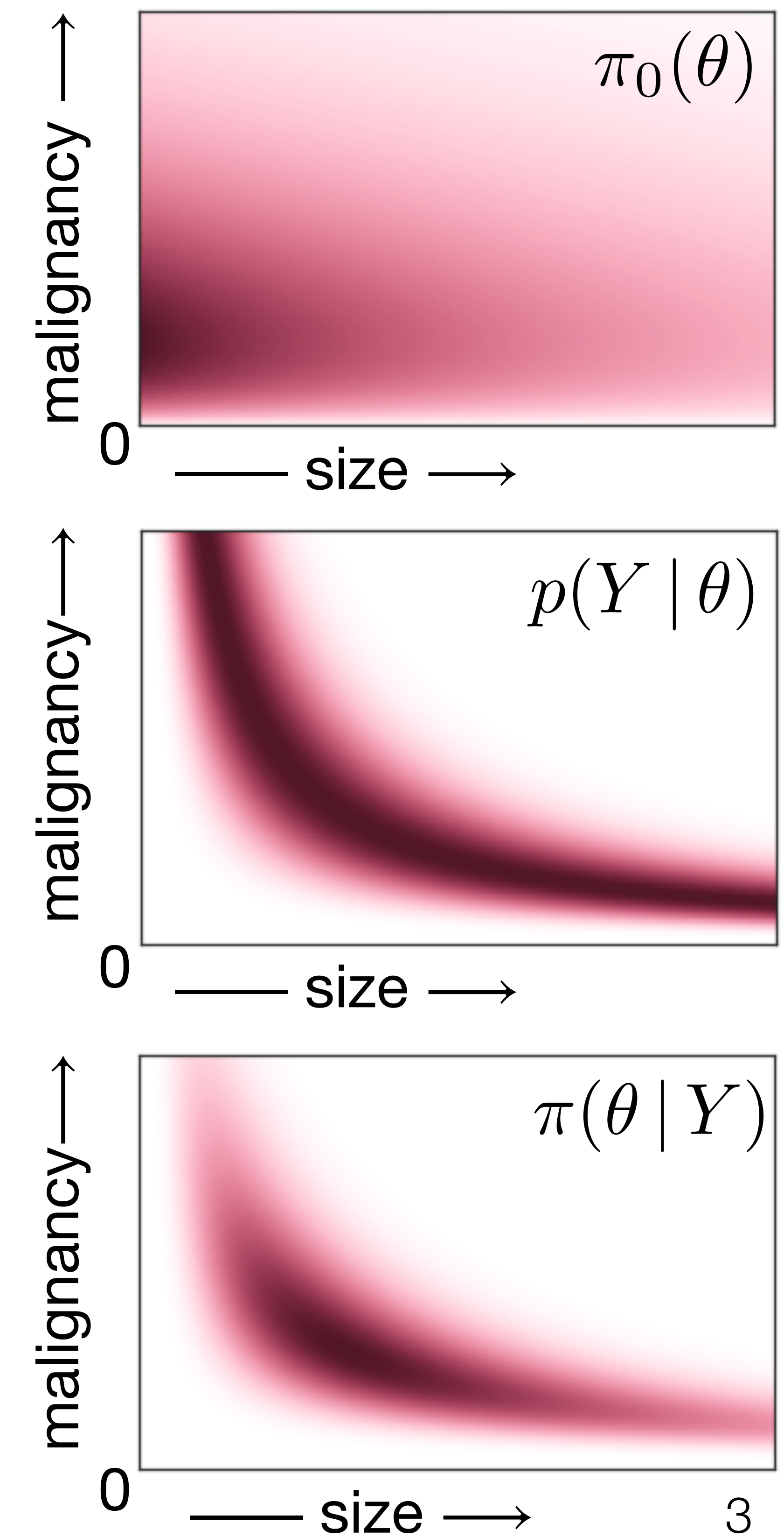
- **Goal:** learn about unobserved phenomenon (parameter) of interest θ **[e.g. tumor size & malignancy]**
- Prior (expert) beliefs $\pi_0(\theta)$ about the phenomenon
- Observe data Y via measurement process $p(Y \mid \theta)$ **[e.g. ultrasound, biopsy]**
- Combine prior and observed data to form posterior distribution via **Bayes' Theorem:**

$$\pi(\theta \mid Y) \propto p(Y \mid \theta)\pi_0(\theta)$$

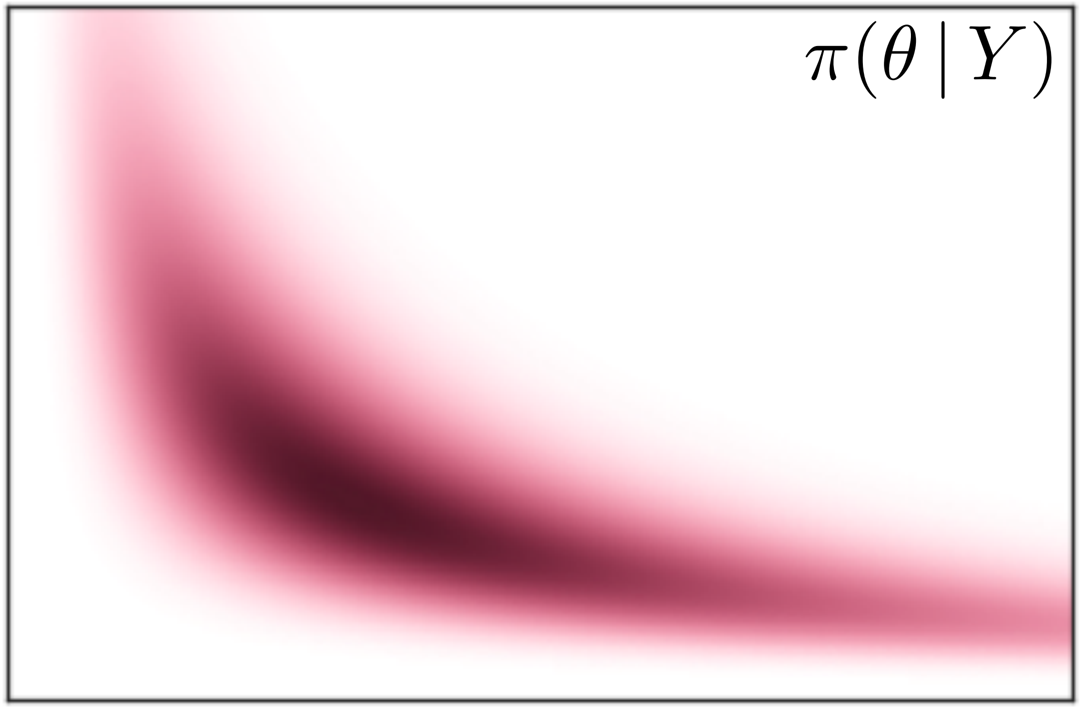
- **Benefits:** coherent belief updates, uncertainty quantification, flexible modeling, and more
- Extract actionable information by **computing expectations [e.g. means and standard deviations]**:

$$\mathbb{E}[f(\theta) \mid Y] = \int f(\theta)\pi(\theta \mid Y)\mathrm{d}\theta$$

- **Computational challenges:** posterior unnormalized, high-dimensional integral



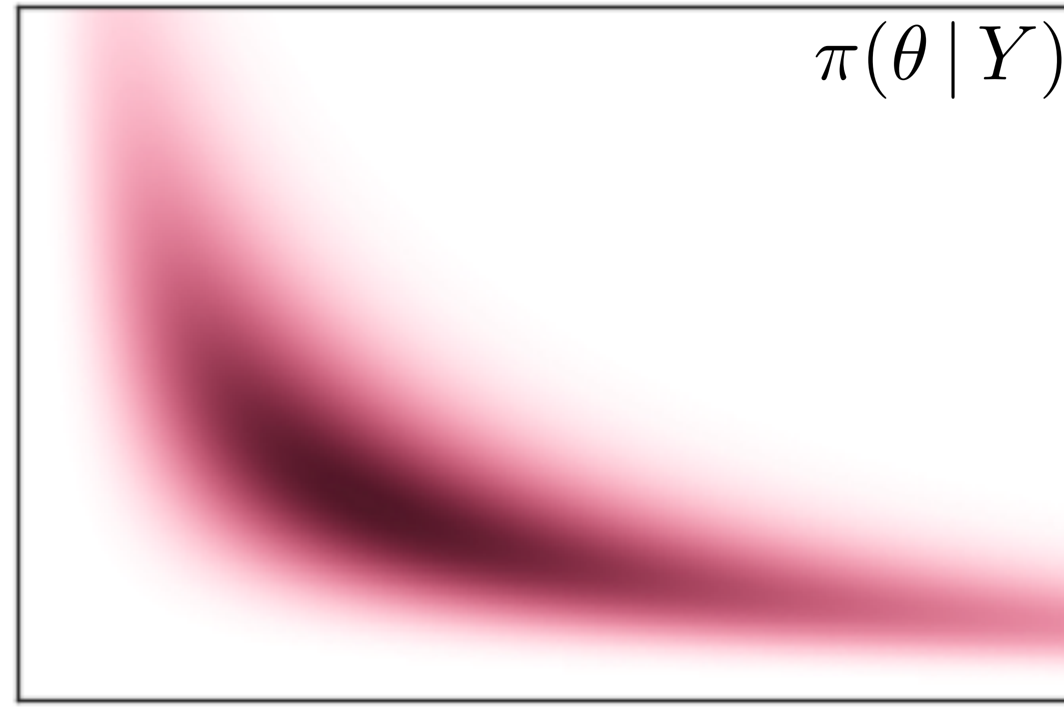
A scalable inference framework



A scalable inference framework

- **Canonical, reliable approximate inference:** Markov chain Monte Carlo (MCMC)

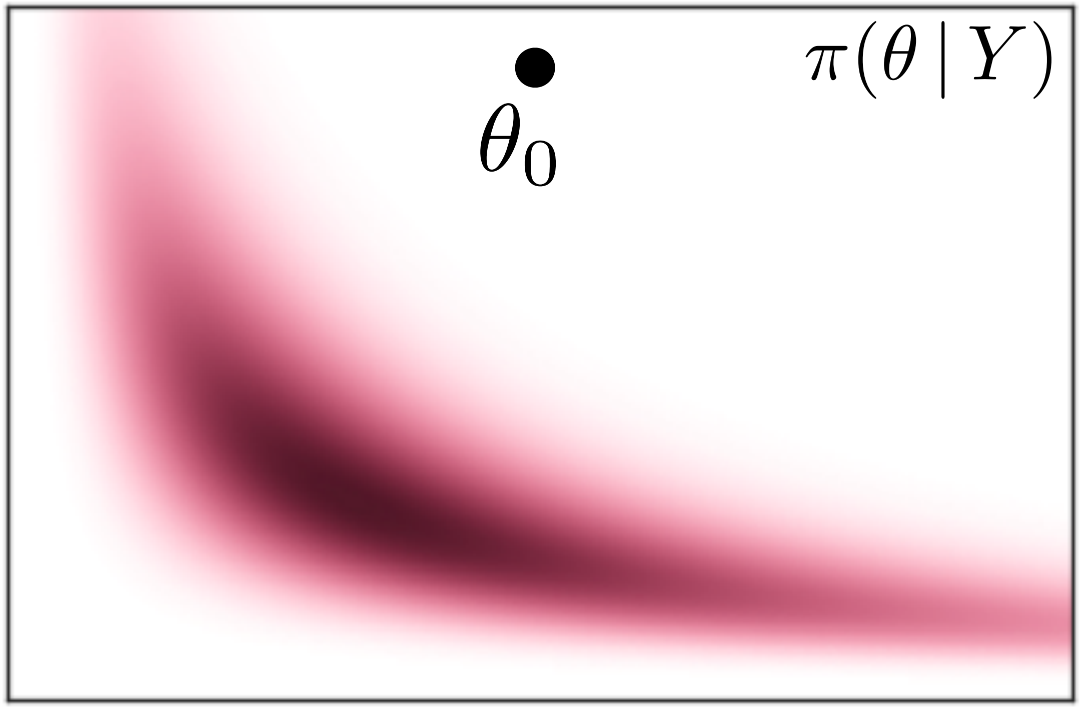
  ➡ A top 10 algorithm of the 20th century
  [Dongarra & Sullivan, *Computing in Science & Engineering*]



A scalable inference framework

- **Canonical, reliable approximate inference:** Markov chain Monte Carlo (MCMC)

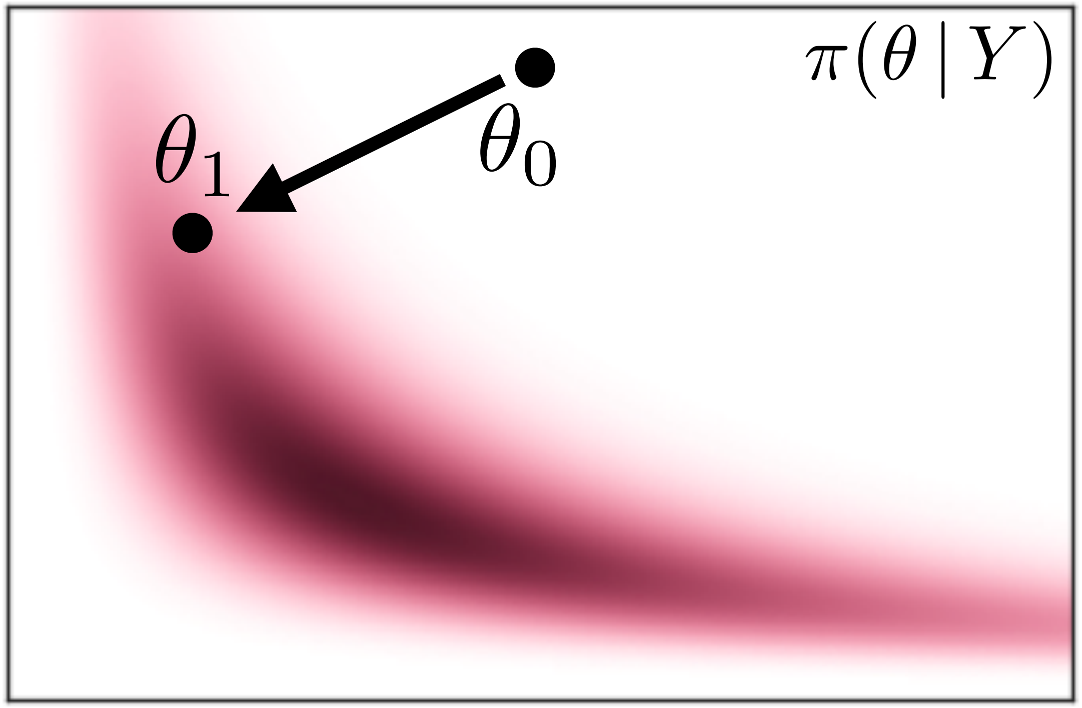
  ➡ A top 10 algorithm of the 20th century
  [Dongarra & Sullivan, *Computing in Science & Engineering*]



A scalable inference framework

- **Canonical, reliable approximate inference:** Markov chain Monte Carlo (MCMC)

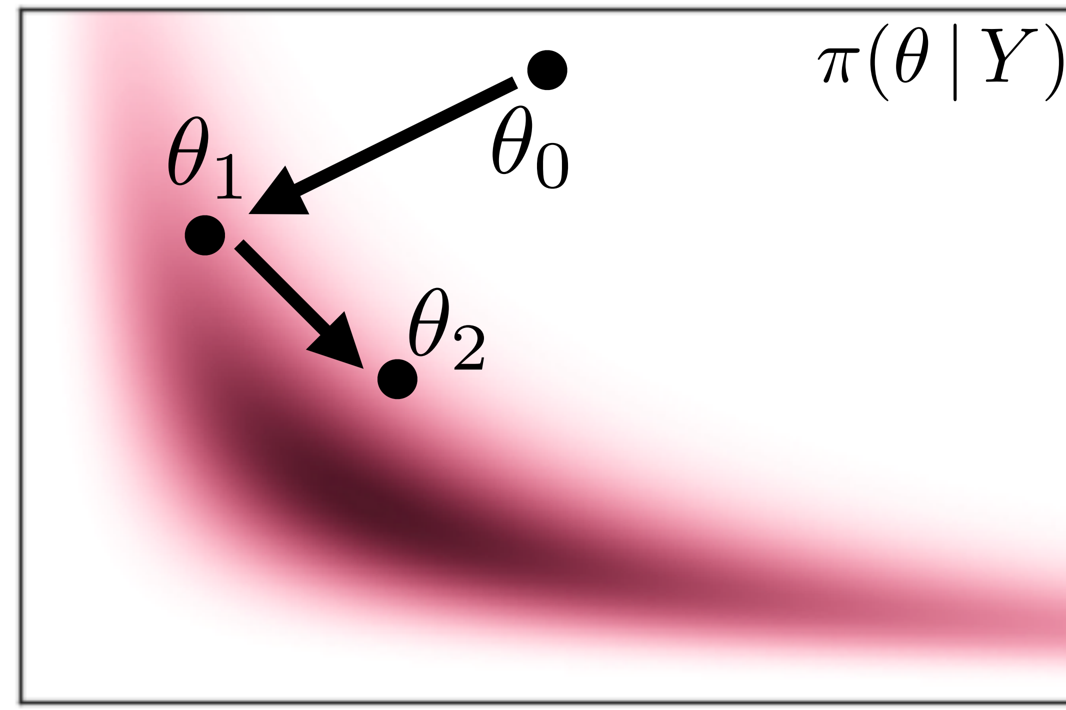
  ➡ A top 10 algorithm of the 20th century
  [Dongarra & Sullivan, *Computing in Science & Engineering*]



A scalable inference framework

- **Canonical, reliable approximate inference:** Markov chain Monte Carlo (MCMC)

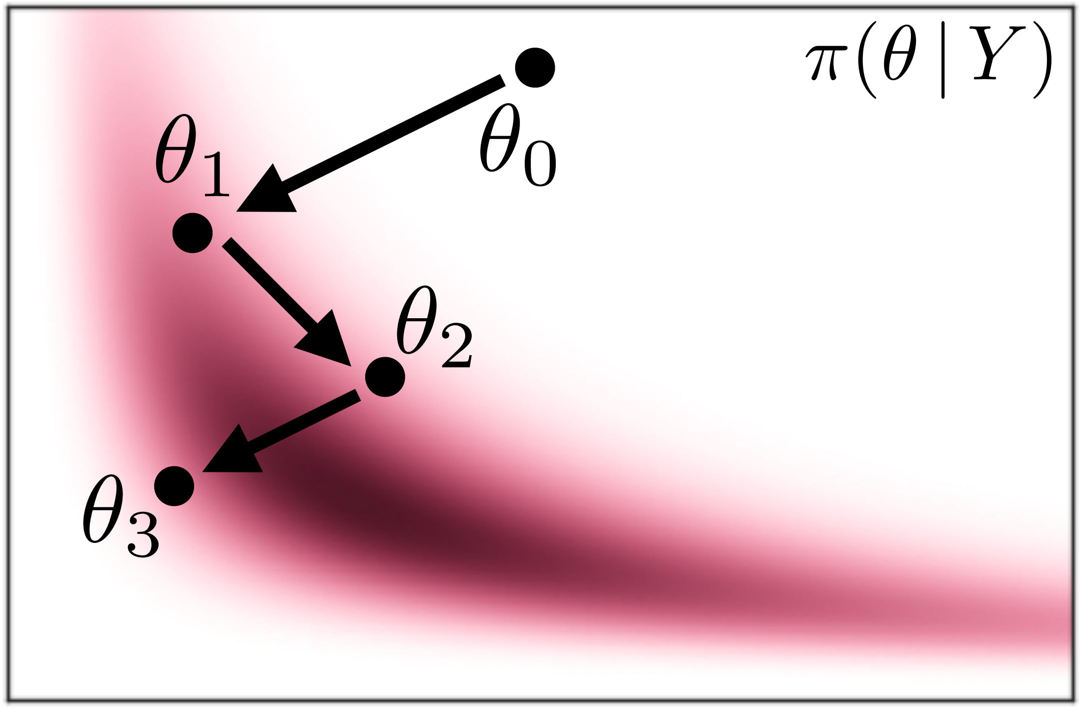
➡ A top 10 algorithm of the 20th century
[Dongarra & Sullivan, *Computing in Science & Engineering*]



A scalable inference framework

- **Canonical, reliable approximate inference:** Markov chain Monte Carlo (MCMC)

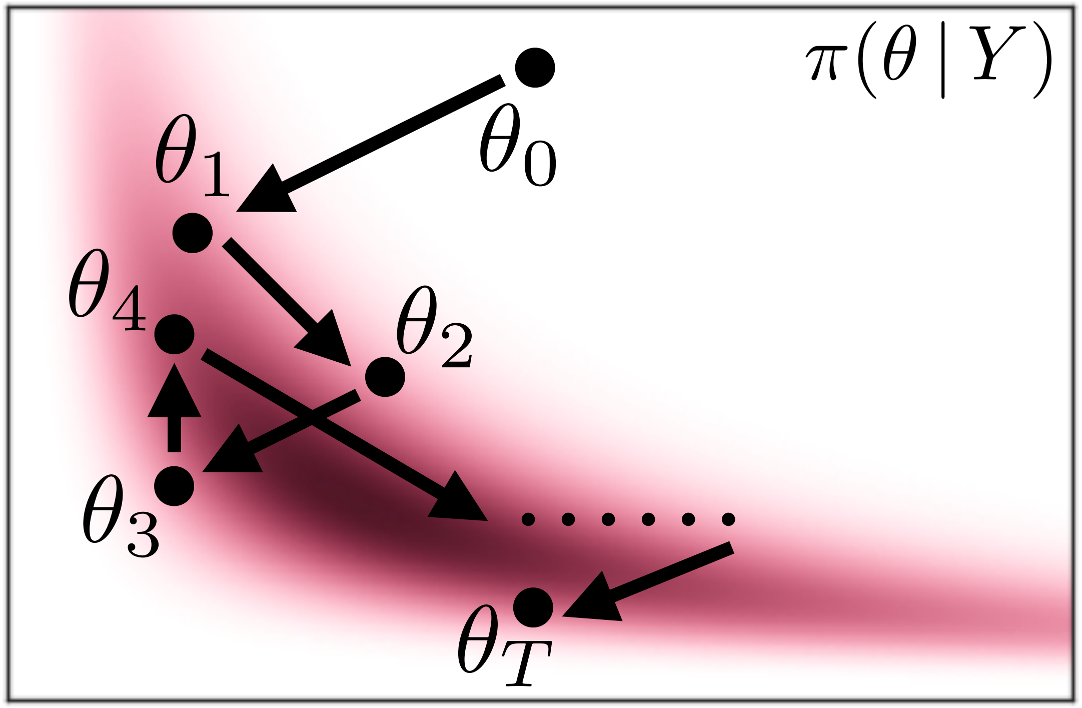
  ➡ A top 10 algorithm of the 20th century
  [Dongarra & Sullivan, *Computing in Science & Engineering*]



A scalable inference framework

- **Canonical, reliable approximate inference:** Markov chain Monte Carlo (MCMC)

  ➡ A top 10 algorithm of the 20th century
  [Dongarra & Sullivan, *Computing in Science & Engineering*]



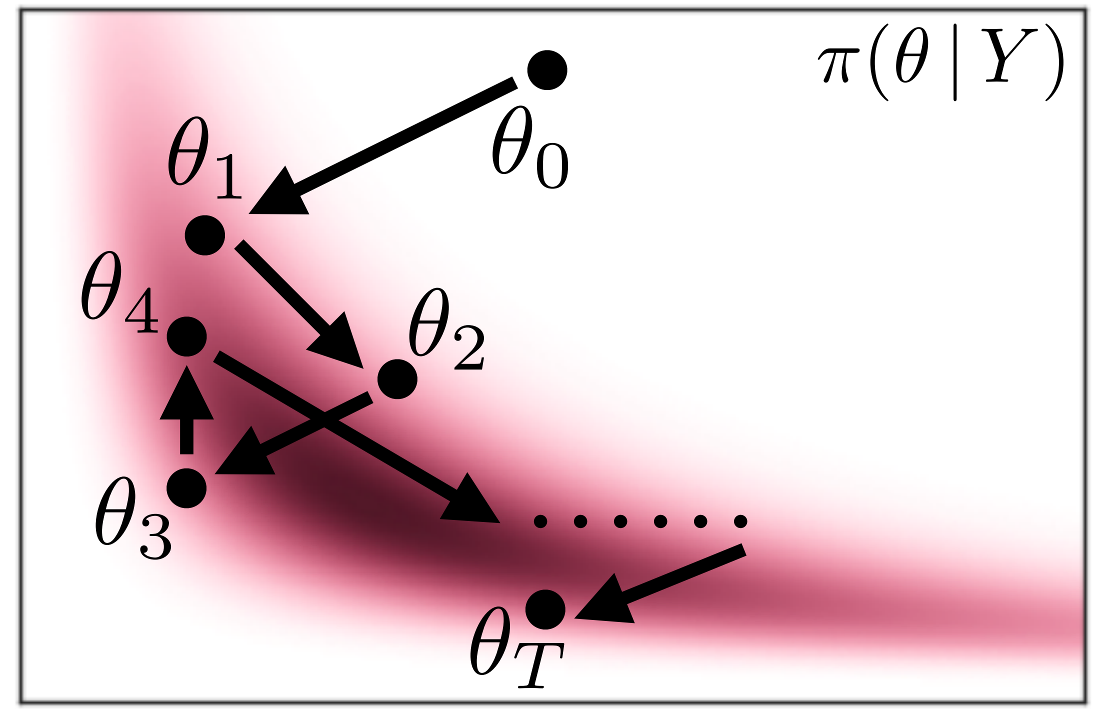
A scalable inference framework

- **Canonical, reliable approximate inference:** Markov chain Monte Carlo (MCMC)

  ➡ A top 10 algorithm of the 20th century
  [Dongarra & Sullivan, *Computing in Science & Engineering*]

- Approximate expectations:

  $$\mathbb{E}[f(\theta) \mid Y] \approx T^{-1} \sum_{t=1}^{T} f(\theta_t)$$



A scalable inference framework

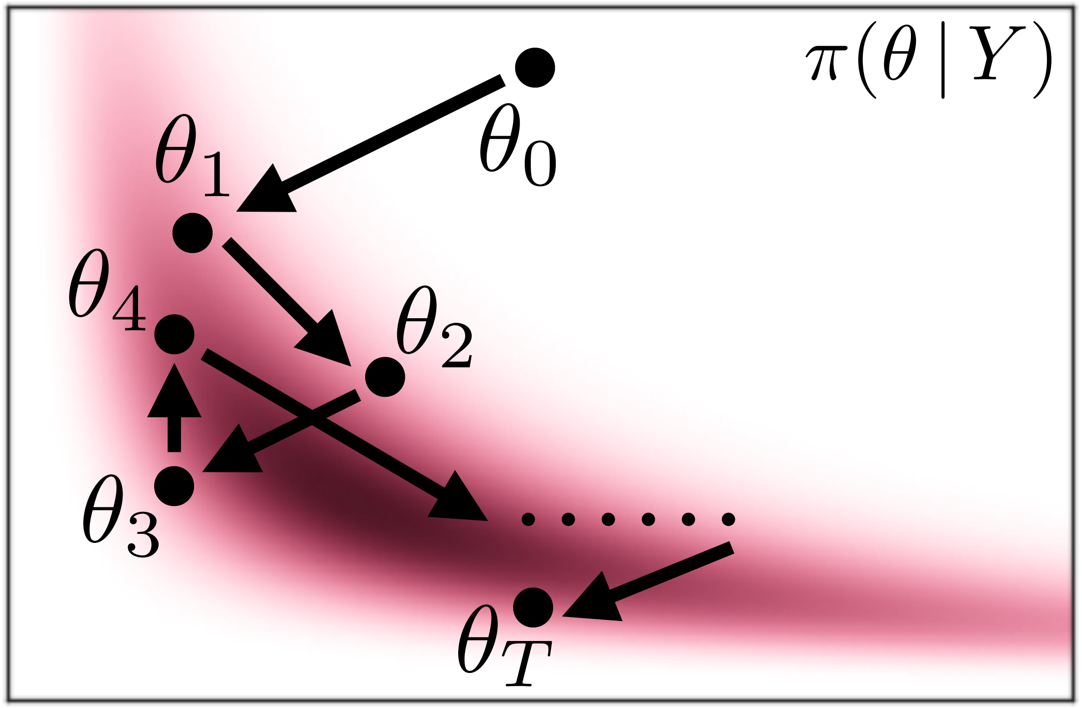
- **Canonical, reliable approximate inference:** Markov chain Monte Carlo (MCMC)

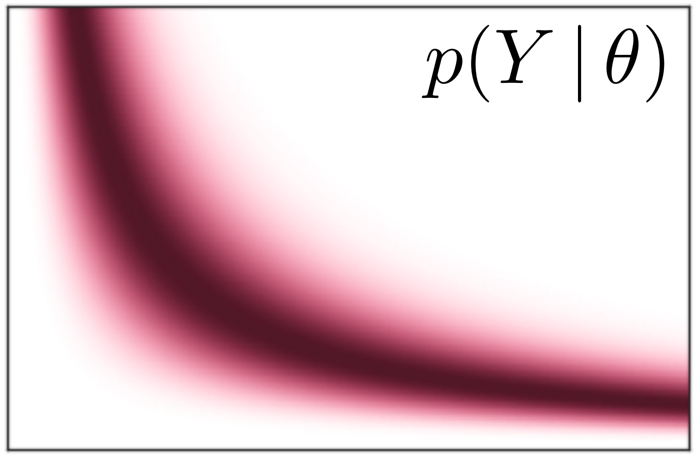
  ➡ A top 10 algorithm of the 20th century
  [Dongarra & Sullivan, *Computing in Science & Engineering*]

- Approximate expectations:

$$\mathbb{E}[f(\theta) \mid Y] \approx T^{-1} \sum_{t=1}^{T} f(\theta_t)$$

- **But MCMC is too slow:** need to perform expensive evaluation of $p(Y \mid \theta_t)$ at iteration t





A scalable inference framework

- **Canonical, reliable approximate inference:** Markov chain Monte Carlo (MCMC)

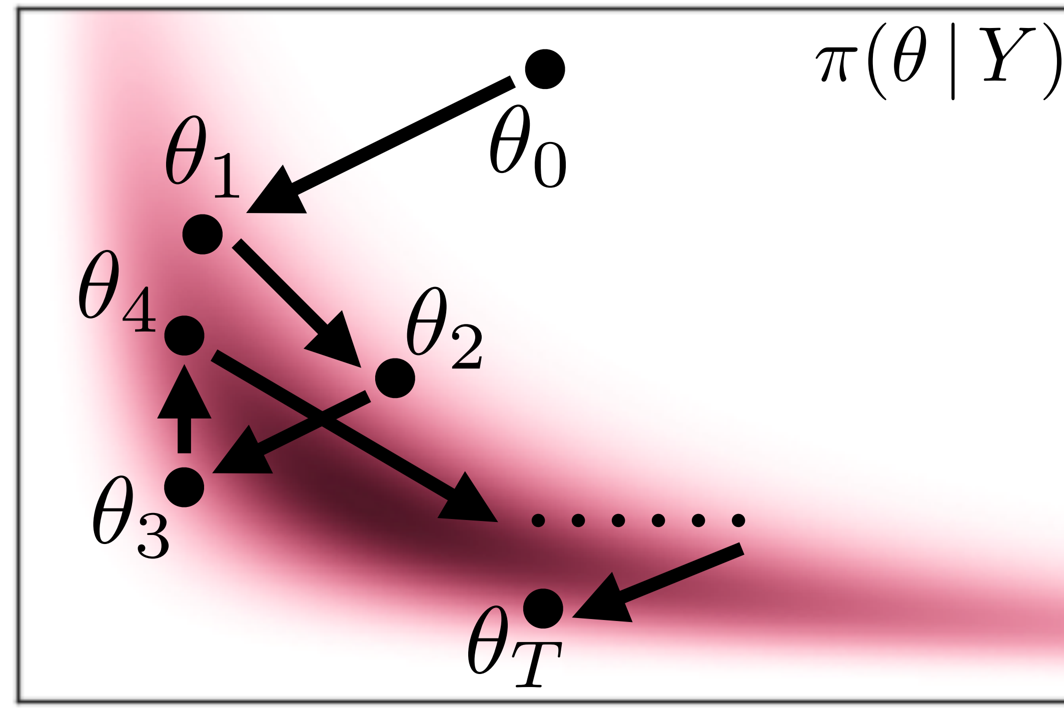
  ➡ A top 10 algorithm of the 20th century
  [Dongarra & Sullivan, *Computing in Science & Engineering*]

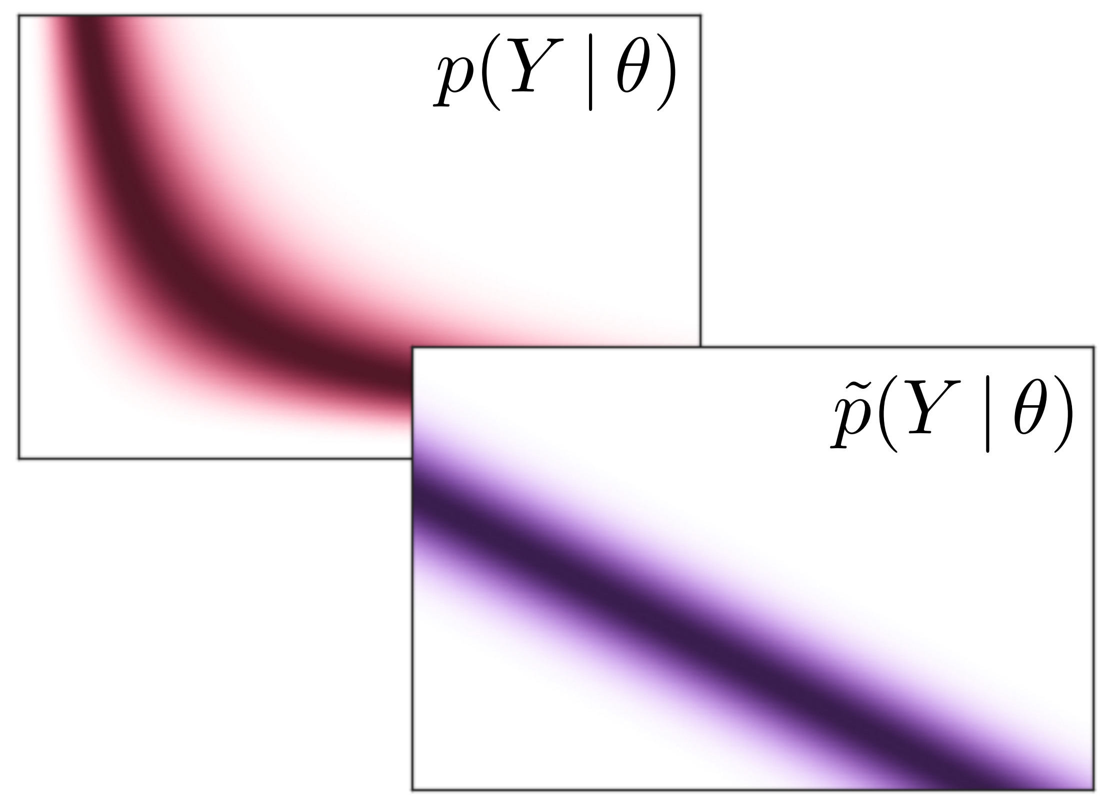
- Approximate expectations:

$$\mathbb{E}[f(\theta) \mid Y] \approx T^{-1} \sum_{t=1}^{T} f(\theta_t)$$

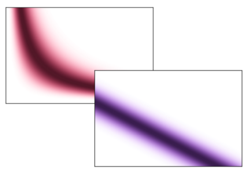
- **But MCMC is too slow:** need to perform expensive evaluation of $p(Y \mid \theta_t)$ at iteration t

- **Our scalable solution:** use likelihood approximations that…
  1. Are accurate
  2. Are fast to compute
  3. Can be rigorously analyzed





Agenda

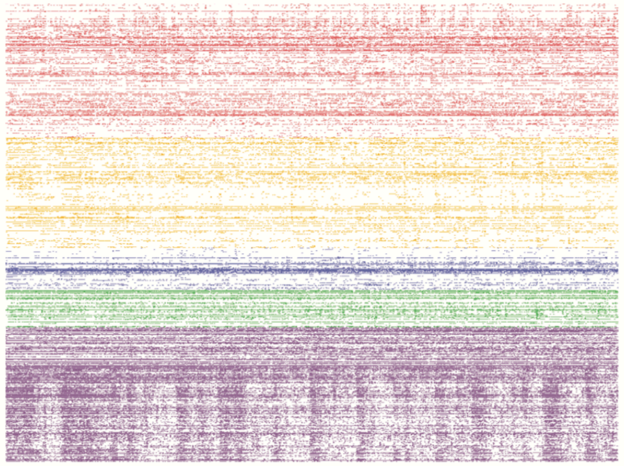
 A framework for scalable Bayesian inference

- ➡ **Algorithm design**
- Meaningful accuracy guarantees

- Validating results from heuristic algorithms

Likelihoods we will approximate

Types of observations



counts

[e.g. neural spikes]

continuous

[e.g. profit]

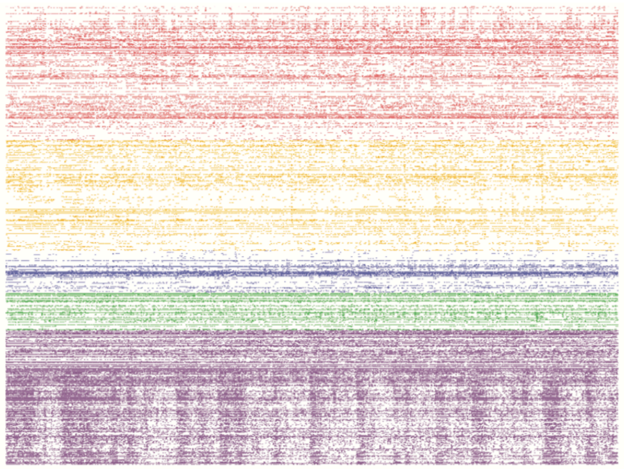
binary

[e.g. has disease?]

[Meager 2017, Park et al. 2014, Jackson, Best & Richardson 2008]

Likelihoods we will approximate

Types of observations



counts

[e.g. neural spikes]

continuous

[e.g. profit]

binary

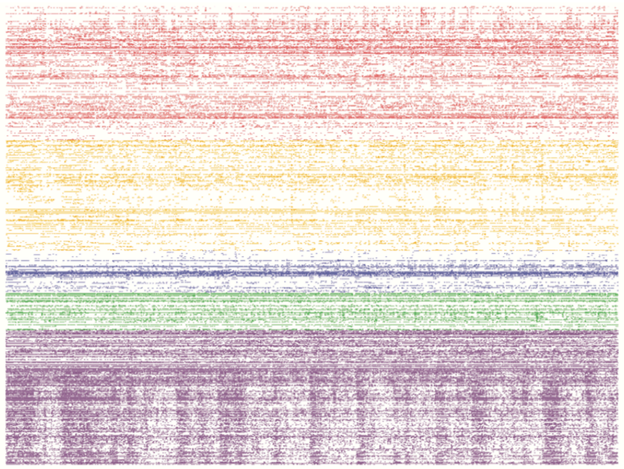
[e.g. has disease?]

Widely-adopted likelihood family: **generalized linear models**

[Meager 2017, Park et al. 2014, Jackson, Best & Richardson 2008]

Likelihoods we will approximate

Types of observations



counts

[e.g. neural spikes]

continuous

[e.g. profit]

binary

[e.g. has disease?]

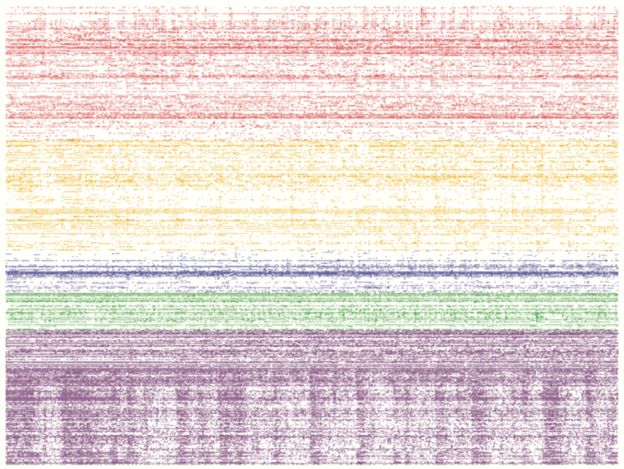
Widely-adopted likelihood family: **generalized linear models**

- Generalization of linear regression

[Meager 2017, Park et al. 2014, Jackson, Best & Richardson 2008]

Likelihoods we will approximate

Types of observations



counts

[e.g. neural spikes]

continuous

[e.g. profit]

binary

[e.g. has disease?]

Widely-adopted likelihood family: **generalized linear models**

- Generalization of linear regression
- Flexible, but still interpretable

[Meager 2017, Park et al. 2014, Jackson, Best & Richardson 2008]

Likelihood approximation strategy

Given to us

Data $Y = \{y_1, y_2, \ldots, y_N\}$, $y_n \in \mathbb{R}^d$, and parameter $\theta \in \mathbb{R}^p$

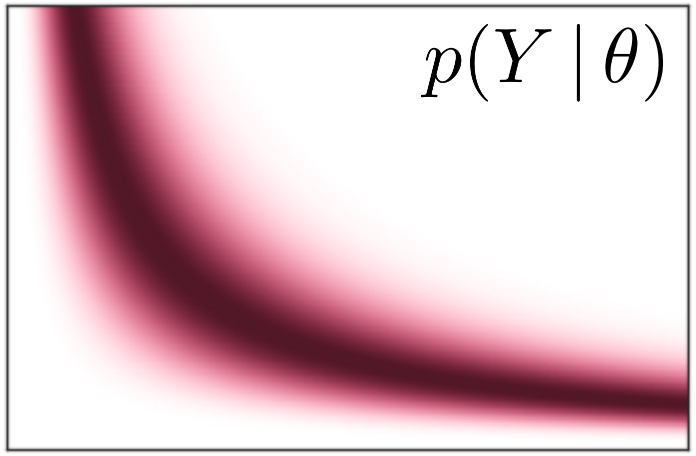
[**H**, Adams & Broderick 2017]

Likelihood approximation strategy

Given to us

Data $Y = \{y_1, y_2, \ldots, y_N\}$, $y_n \in \mathbb{R}^d$, and parameter $\theta \in \mathbb{R}^p$

Log-likelihood: $\log p(Y \mid \theta) = \sum_{n=1}^{N} \log p(y_n \mid \theta)$

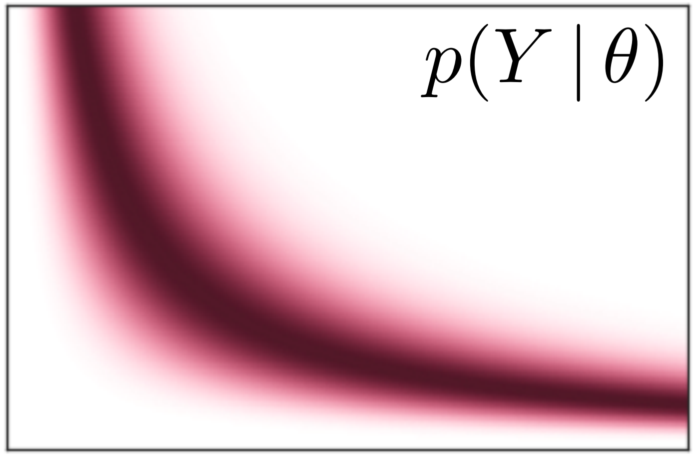


Likelihood approximation strategy

Given to us

Data $Y = \{y_1, y_2, \ldots, y_N\}$, $y_n \in \mathbb{R}^d$, and parameter $\theta \in \mathbb{R}^p$

Log-likelihood: $\log p(Y \mid \theta) = \sum_{n=1}^{N} \log p(y_n \mid \theta)$



We construct *approximate sufficient statistics*

Reparameterization function $\eta(\theta) \in \mathbb{R}^L$

Sufficient statistic function $\tau(y_n) \in \mathbb{R}^L$

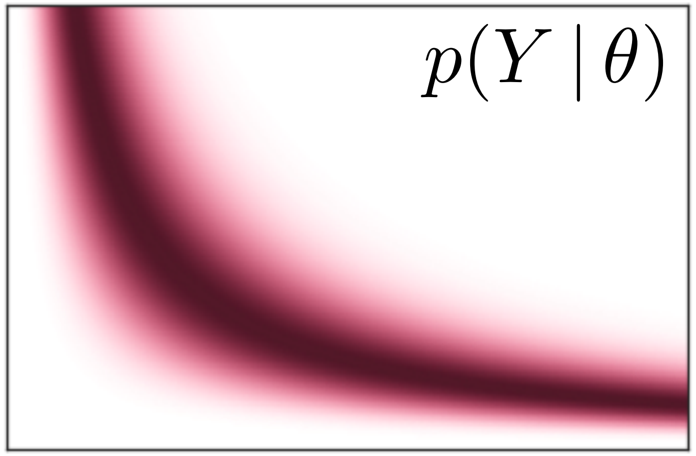
[**H**, Adams & Broderick 2017]

Likelihood approximation strategy

Given to us

Data $Y = \{y_1, y_2, \ldots, y_N\}$, $y_n \in \mathbb{R}^d$, and parameter $\theta \in \mathbb{R}^p$

Log-likelihood: $\log p(Y \mid \theta) = \sum_{n=1}^{N} \log p(y_n \mid \theta)$



We construct *approximate sufficient statistics*

Reparameterization function $\eta(\theta) \in \mathbb{R}^L$

Sufficient statistic function $\tau(y_n) \in \mathbb{R}^L$

Log-likelihood approximation $\log p(y_n \mid \theta) \approx \eta(\theta) \cdot \tau(y_n)$

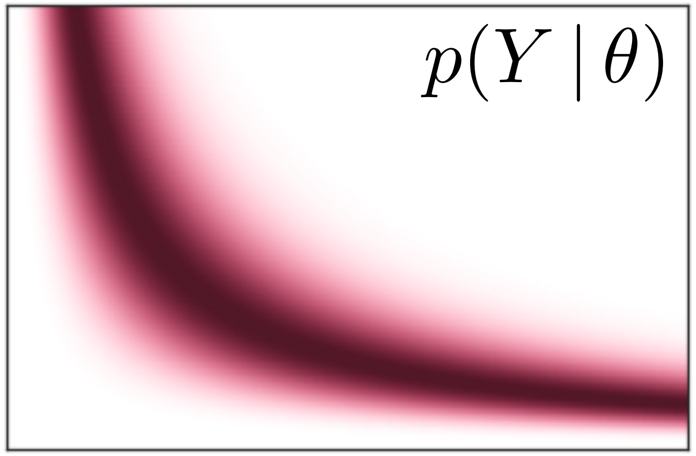
[H, Adams & Broderick 2017]

Likelihood approximation strategy

Given to us

Data $Y = \{y_1, y_2, \ldots, y_N\}$, $y_n \in \mathbb{R}^d$, and parameter $\theta \in \mathbb{R}^p$

Log-likelihood: $\log p(Y \mid \theta) = \sum_{n=1}^{N} \log p(y_n \mid \theta)$



We construct *approximate sufficient statistics*

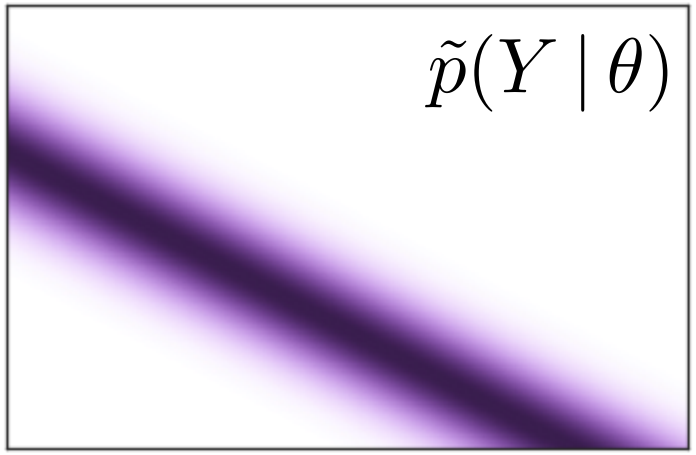
Reparameterization function $\eta(\theta) \in \mathbb{R}^L$

Sufficient statistic function $\tau(y_n) \in \mathbb{R}^L$

Log-likelihood approximation $\log p(y_n \mid \theta) \approx \eta(\theta) \cdot \tau(y_n)$

Resulting approximation

$$\log p(Y \mid \theta) \approx \log \tilde{p}(Y \mid \theta) := \eta(\theta) \cdot \underbrace{\sum_{n=1}^{N} \tau(y_n)}_{\tau(Y)}$$



[**H**, Adams & Broderick 2017]

Why our strategy works

Original

$$\log p(Y \mid \theta) = \sum_{n=1}^{N} \log p(y_n \mid \theta)$$

Our approximation

$$\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \underbrace{\sum_{n=1}^{N} \tau(y_n)}_{\tau(Y)}$$

Run MCMC for T iterations

Why our strategy works

Original

$$\log p(Y \mid \theta) = \sum_{n=1}^{N} \log p(y_n \mid \theta)$$

Our approximation

$$\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \underbrace{\sum_{n=1}^{N} \tau(y_n)}_{\tau(Y)}$$

Run MCMC for T iterations

- $\log p(Y \mid \theta_t)$ takes $\Theta(N)$ time to evaluate
- Overall: $\Theta(N \times T)$ time

[H, Adams & Broderick 2017]

Why our strategy works

Original

$$\log p(Y \mid \theta) = \sum_{n=1}^{N} \log p(y_n \mid \theta)$$

Our approximation

$$\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \underbrace{\sum_{n=1}^{N} \tau(y_n)}_{\tau(Y)}$$

Run MCMC for T iterations

- $\log p(Y \mid \theta_t)$ takes $\Theta(N)$ time to evaluate
- Overall: $\Theta(N \times T)$ time

- $\tau(Y)$ takes $\Theta(N)$ time to evaluate
- approximation to $\log \tilde{p}(Y \mid \theta_t)$ takes $\Theta(1)$ time to evaluate
- Overall: $\Theta(N + T)$ time

[**H**, Adams & Broderick 2017]

Why our strategy works

Original

$$\log p(Y \mid \theta) = \sum_{n=1}^{N} \log p(y_n \mid \theta)$$

Our approximation

$$\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \underbrace{\sum_{n=1}^{N} \tau(y_n)}_{\tau(Y)}$$

Run MCMC for T iterations

- $\log p(Y \mid \theta_t)$ takes $\Theta(N)$ time to evaluate
- Overall: $\Theta(N \times T)$ time

- $\tau(Y)$ takes $\Theta(N)$ time to evaluate
- approximation to $\log \tilde{p}(Y \mid \theta_t)$ takes $\Theta(1)$ time to evaluate
- Overall: $\Theta(N + T)$ time

[H, Adams & Broderick 2017]

Why our strategy works

Original

$$\log p(Y \mid \theta) = \sum_{n=1}^{N} \log p(y_n \mid \theta)$$

Our approximation

$$\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \underbrace{\sum_{n=1}^{N} \tau(y_n)}_{\tau(Y)}$$

Run MCMC for T iterations

- $\log p(Y \mid \theta_t)$ takes $\Theta(N)$ time to evaluate
- Overall: $\Theta(N \times T)$ time

- $\tau(Y)$ takes $\Theta(N)$ time to evaluate
- approximation to $\log \tilde{p}(Y \mid \theta_t)$ takes $\Theta(1)$ time to evaluate
- Overall: $\Theta(N + T)$ time

[H, Adams & Broderick 2017]

Why our strategy works

Original

$$\log p(Y \mid \theta) = \sum_{n=1}^{N} \log p(y_n \mid \theta)$$

Our approximation

$$\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \underbrace{\sum_{n=1}^{N} \tau(y_n)}_{\tau(Y)}$$

Run MCMC for T iterations

Streaming and distributed too!

$\tau(Y)$ takes $\Theta(N)$ time to evaluate

- $\log p(Y \mid \theta_t)$ takes $\Theta(N)$ time to evaluate
- Overall: $\Theta(N \times T)$ time

- approximation to $\log \tilde{p}(Y \mid \theta_t)$ takes $\Theta(1)$ time to evaluate
- Overall: $\Theta(N + T)$ time

[H, Adams & Broderick 2017]

Constructing polynomial approximate sufficient statistics (PASS)

Likelihood Approximation

Reparameterization function $\eta(\theta) \in \mathbb{R}^L$

Sufficient statistic function $\tau(y_n) \in \mathbb{R}^L$

Log-likelihood approximation $\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \sum_{n=1}^{N} \tau(y_n)$

[**H**, Adams & Broderick 2017]

Constructing polynomial approximate sufficient statistics (PASS)

Likelihood Approximation

Reparameterization function $\eta(\theta) \in \mathbb{R}^L$

Sufficient statistic function $\tau(y_n) \in \mathbb{R}^L$

Log-likelihood approximation $\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \sum_{n=1}^{N} \tau(y_n)$

How do we choose η and τ?

Each component a polynomial: $\eta(\theta)_\ell \in \mathrm{poly}(\theta), \tau(y_n)_\ell \in \mathrm{poly}(y_n)$

[**H**, Adams & Broderick 2017]

Constructing polynomial approximate sufficient statistics (PASS)

Likelihood Approximation

Reparameterization function $\eta(\theta) \in \mathbb{R}^L$

Sufficient statistic function $\tau(y_n) \in \mathbb{R}^L$

Log-likelihood approximation $\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \sum_{n=1}^{N} \tau(y_n)$

How do we choose η and τ?

Each component a polynomial: $\eta(\theta)_\ell \in \mathrm{poly}(\theta), \tau(y_n)_\ell \in \mathrm{poly}(y_n)$

Why polynomials?

[**H**, Adams & Broderick 2017]

Constructing polynomial approximate sufficient statistics (PASS)

Likelihood Approximation

Reparameterization function $\eta(\theta) \in \mathbb{R}^L$

Sufficient statistic function $\tau(y_n) \in \mathbb{R}^L$

Log-likelihood approximation $\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \sum_{n=1}^{N} \tau(y_n)$

How do we choose η and τ?

Each component a polynomial: $\eta(\theta)_\ell \in \mathrm{poly}(\theta), \tau(y_n)_\ell \in \mathrm{poly}(y_n)$

Why polynomials?

1. Computationally convenient

[**H**, Adams & Broderick 2017]

Constructing polynomial approximate sufficient statistics (PASS)

Likelihood Approximation

Reparameterization function $\eta(\theta) \in \mathbb{R}^L$

Sufficient statistic function $\tau(y_n) \in \mathbb{R}^L$

Log-likelihood approximation $\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \sum_{n=1}^{N} \tau(y_n)$

How do we choose η and τ?

Each component a polynomial: $\eta(\theta)_\ell \in \mathrm{poly}(\theta), \tau(y_n)_\ell \in \mathrm{poly}(y_n)$

Why polynomials?

1. Computationally convenient
2. Can approximate any smooth function

[H, Adams & Broderick 2017]

Constructing polynomial approximate sufficient statistics (PASS)

Likelihood Approximation

Reparameterization function $\eta(\theta) \in \mathbb{R}^L$

Sufficient statistic function $\tau(y_n) \in \mathbb{R}^L$

Log-likelihood approximation $\log \tilde{p}(Y \mid \theta) = \eta(\theta) \cdot \sum_{n=1}^{N} \tau(y_n)$

How do we choose η and τ?

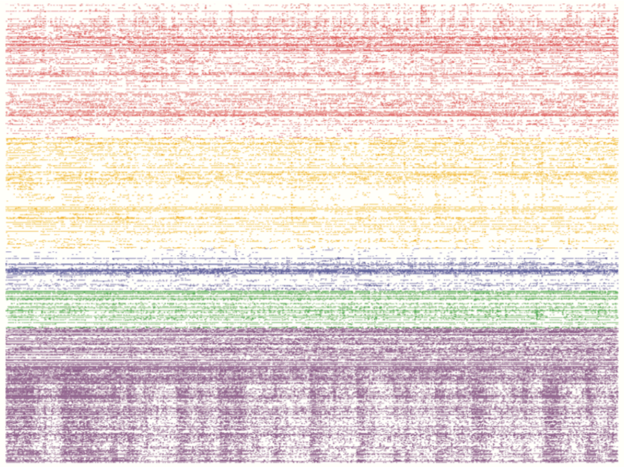
Each component a polynomial: $\eta(\theta)_\ell \in \mathrm{poly}(\theta), \tau(y_n)_\ell \in \mathrm{poly}(y_n)$

Why polynomials?

1. Computationally convenient
2. Can approximate any smooth function
3. Approximation properties are well-understood

[**H**, Adams & Broderick 2017]

Polynomial approximations very accurate



counts

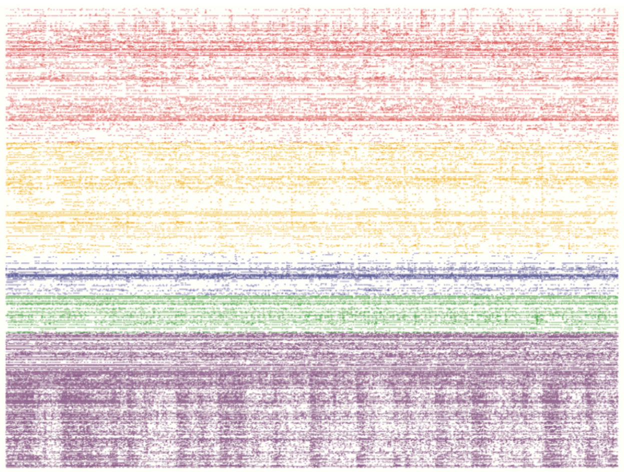
[e.g. neural spikes]

binary

[e.g. has disease?]

[**H**, Adams & Broderick 2017, Zoltowski & Pillow 2018]

Polynomial approximations very accurate

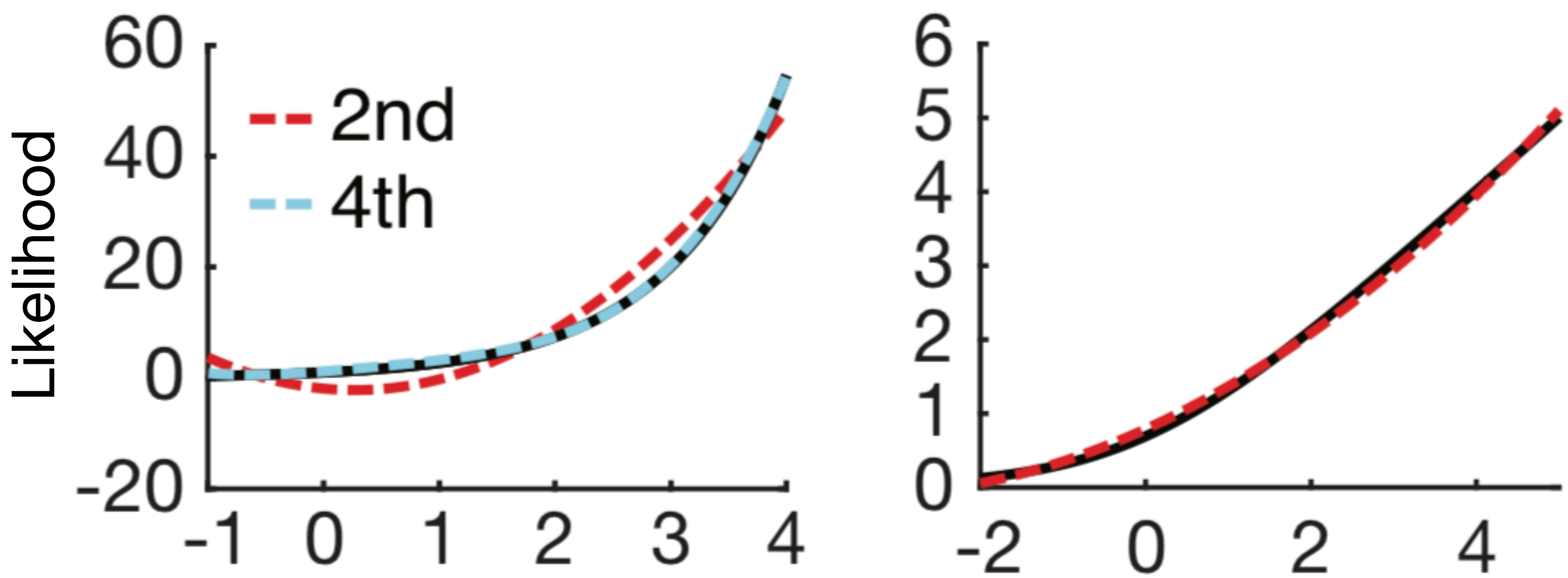


counts

[e.g. neural spikes]

binary

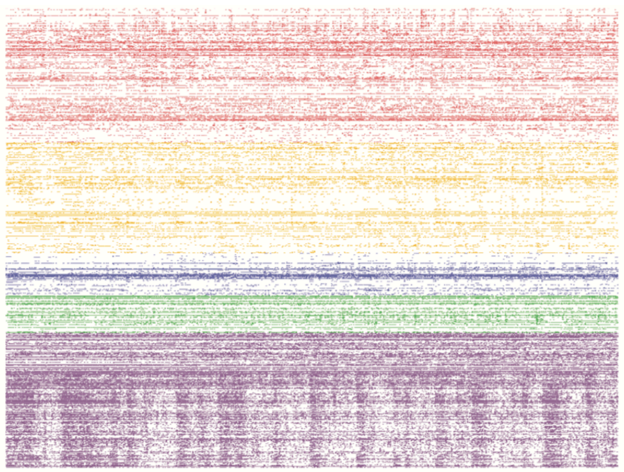
[e.g. has disease?]



Poisson regression

[**H**, Adams & Broderick 2017, Zoltowski & Pillow 2018]

Polynomial approximations very accurate

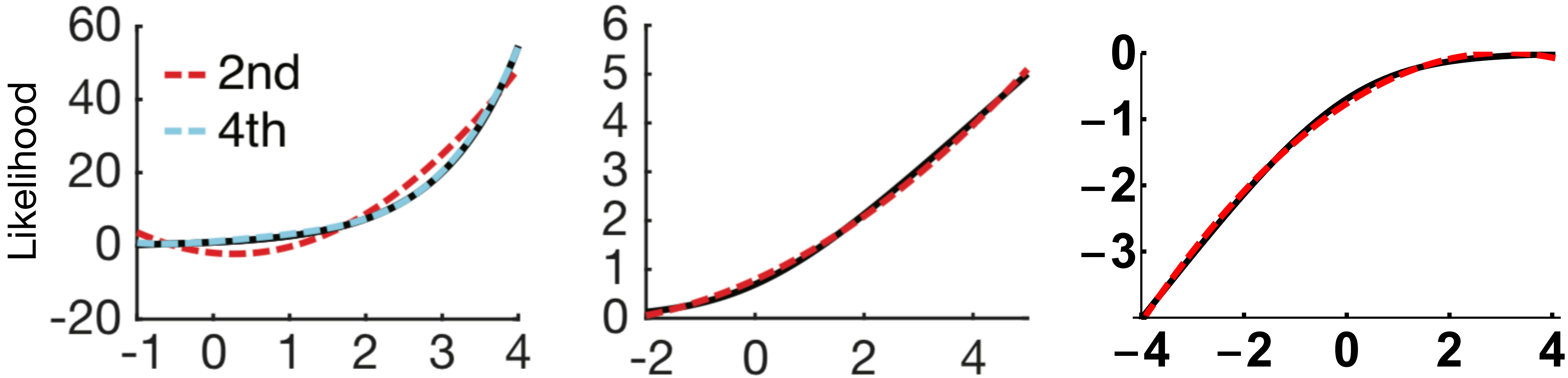


counts

[e.g. neural spikes]

binary

[e.g. has disease?]



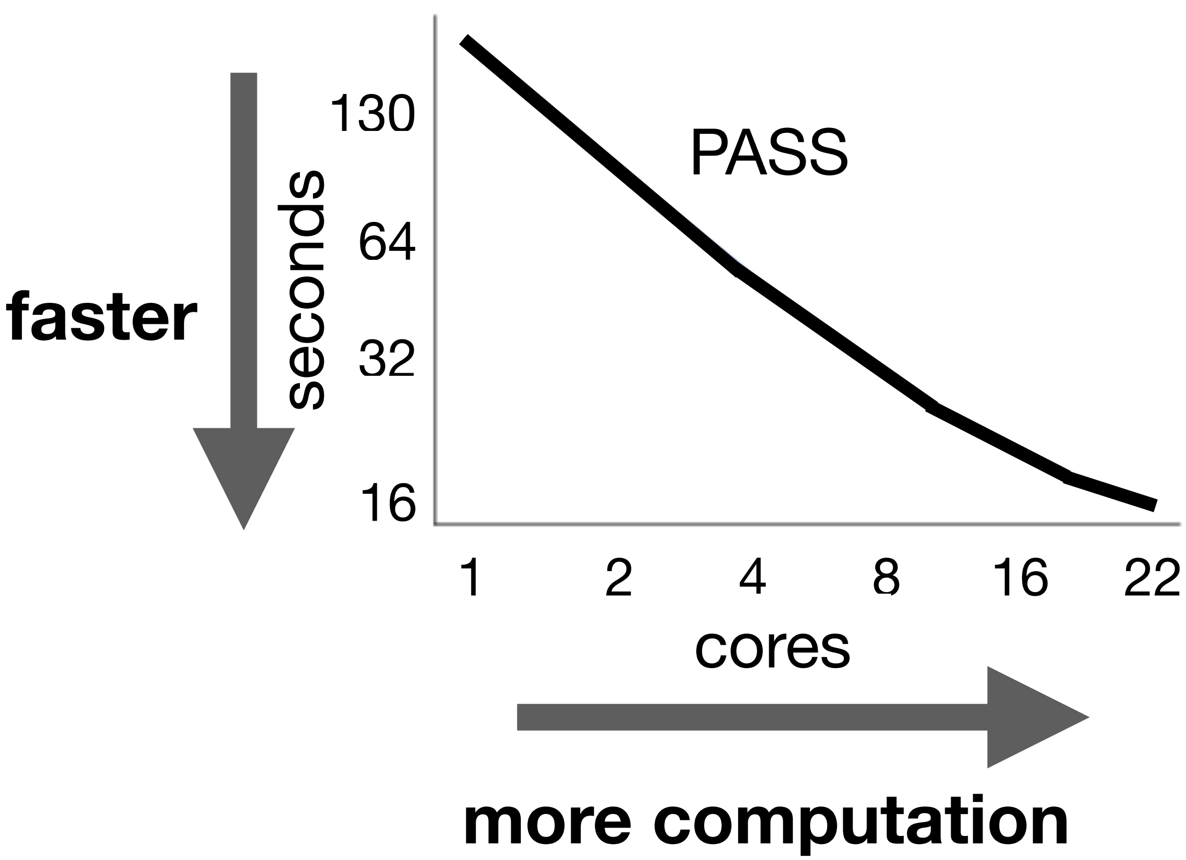
Poisson regression

Logistic regression

[**H**, Adams & Broderick 2017, Zoltowski & Pillow 2018]

Fast, accurate empirical performance

Fast distributed computation

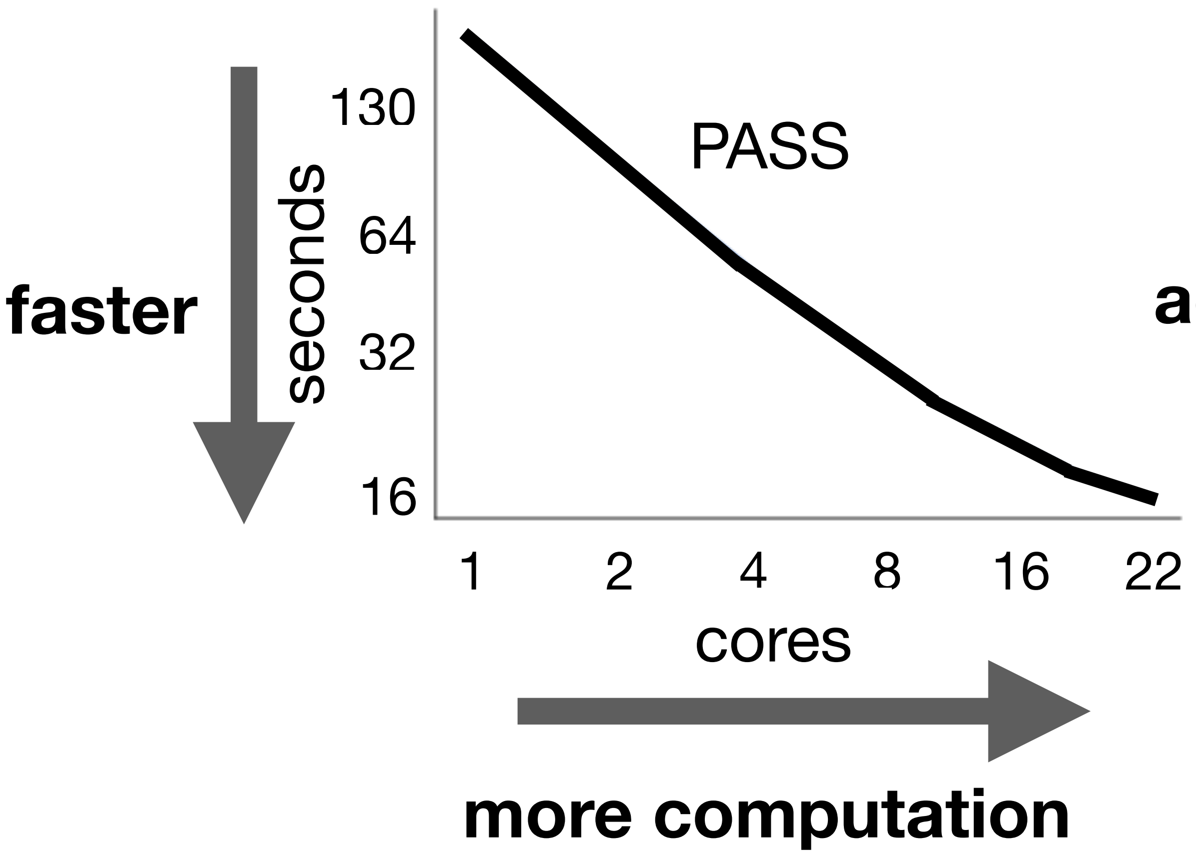


- Logistic regression
- 6 million observations with 1,000 covariates
- MCMC: 1+ days

[**H**, Adams & Broderick 2017]

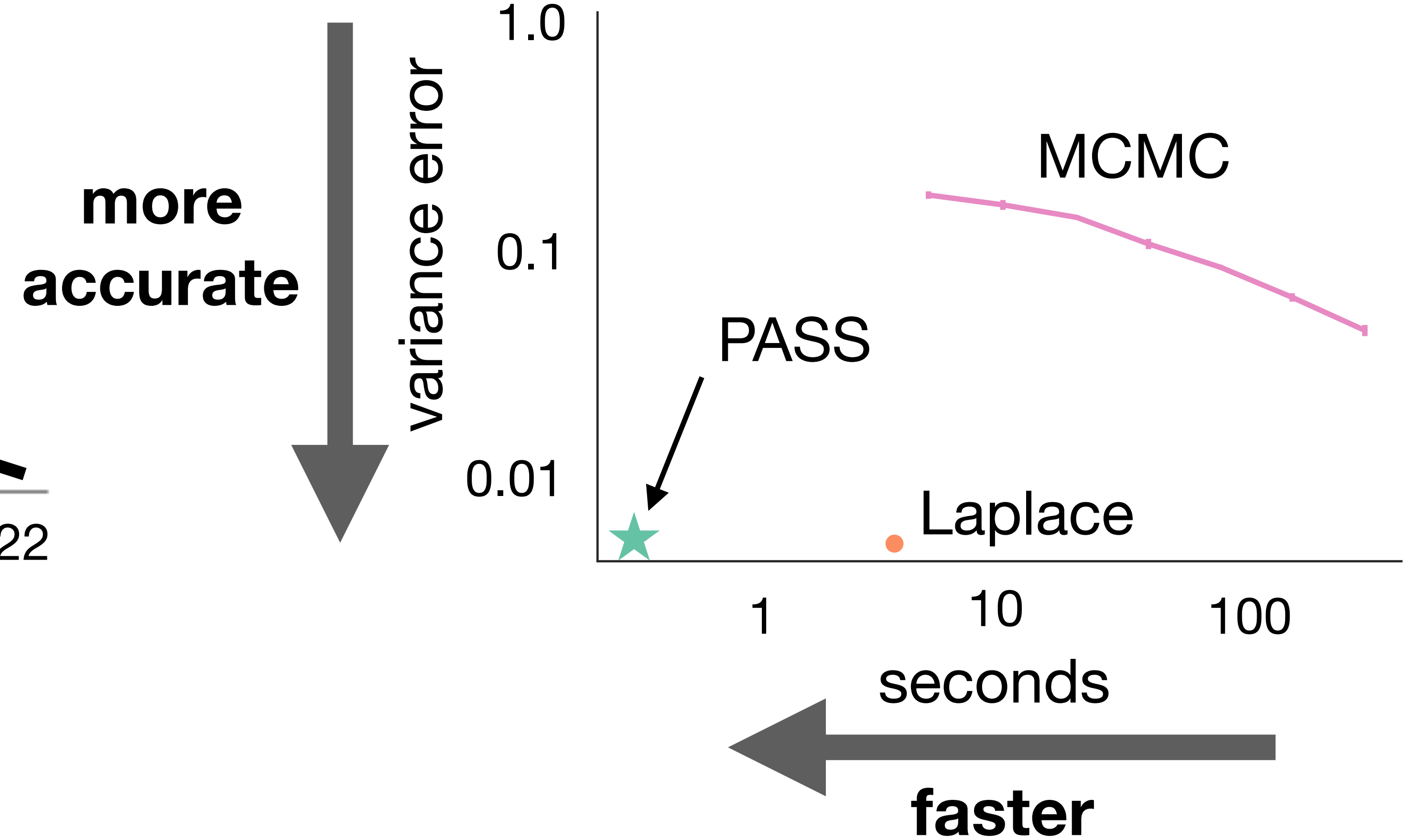
Fast, accurate empirical performance

Fast distributed computation



- Logistic regression
- 6 million observations with 1,000 covariates
- MCMC: 1+ days

Fast and accurate

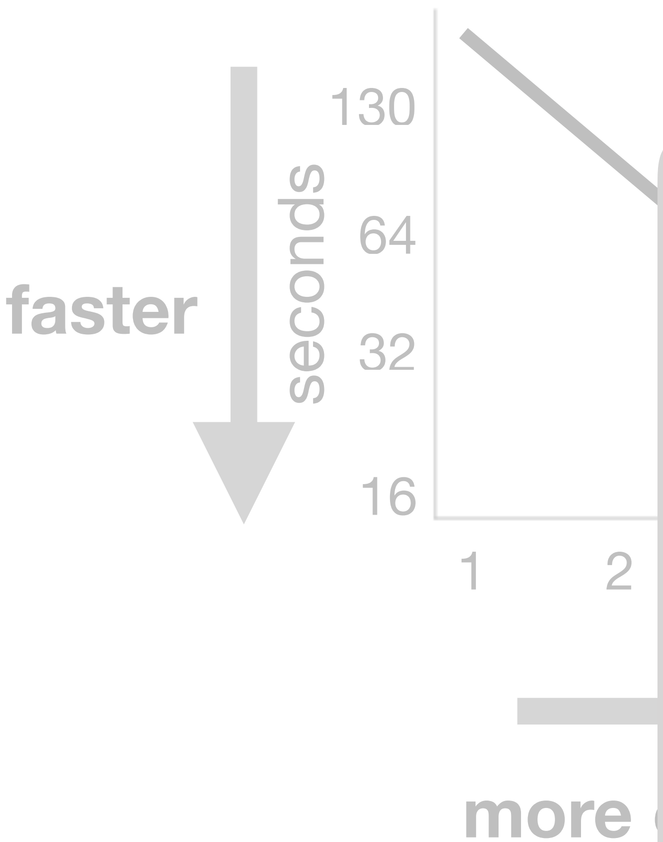


- Logistic regression
- 350,000 observations with 127 covariates
- Good mean estimation and predictive performance too

[H, Adams & Broderick 2017]

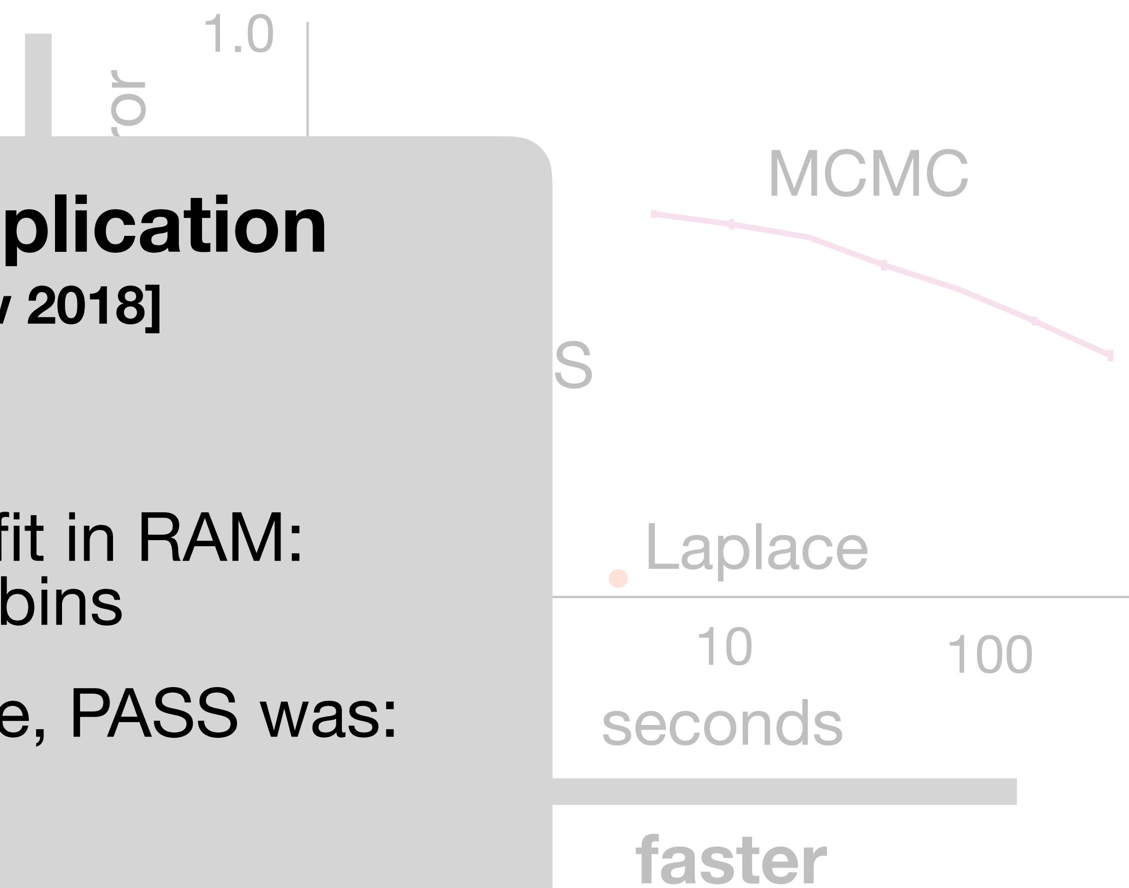
Fast, accurate empirical performance

Fast distributed computation



- Logistic regr
- 6 million obs
1,000 covariates
- MCMC: 1+ days

Fast and accurate



ons with
127 covariates

- Good mean estimation and predictive performance too

Neuroscience application

[Zoltowski & Pillow 2018]

- Poisson regression
- Full dataset doesn't fit in RAM:
2 billion spike count bins
- Compared to Laplace, PASS was:
  - 60x faster
  - 1000x memory reduction
  - Essentially no loss of accuracy

[H, Adams & Broderick 2017]

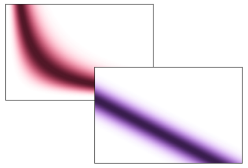
References

Huggins, Campbell, Kasprzak & Broderick. *Scalable Gaussian process inference with finite-data mean and variance guarantees*. AISTATS, 2019.

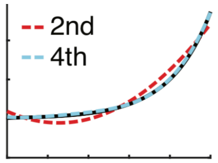
Huggins, Adams & Broderick. *PASS-GLM: polynomial approximate sufficient statistics for scalable Bayesian GLM inference*. Neural Information Processing Systems, 2017.

Huggins, Campbell & Broderick. *Coresets for scalable Bayesian logistic regression*. Neural Information Processing Systems, 2016.

Agenda



A framework for scalable Bayesian inference



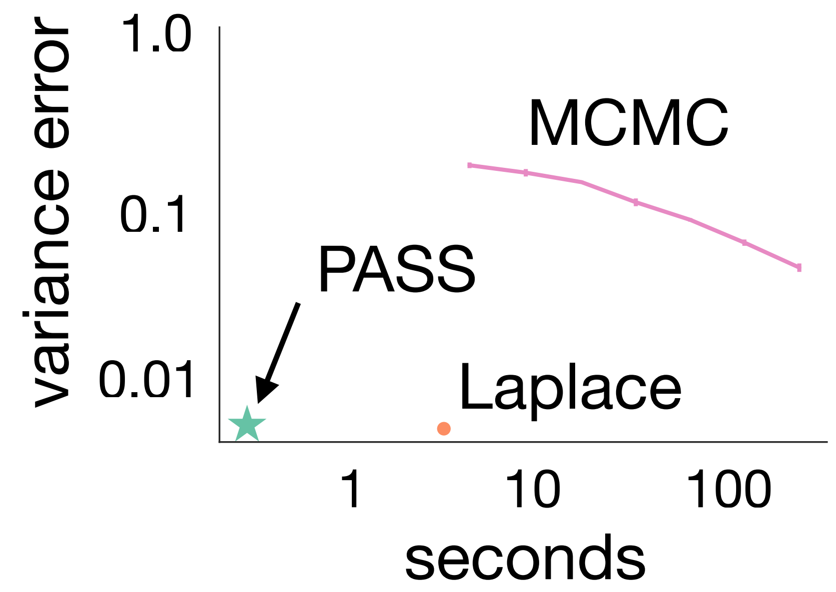
Algorithm design

➡ **Meaningful accuracy guarantees**

- Validating results from heuristic algorithms

What about the next dataset?

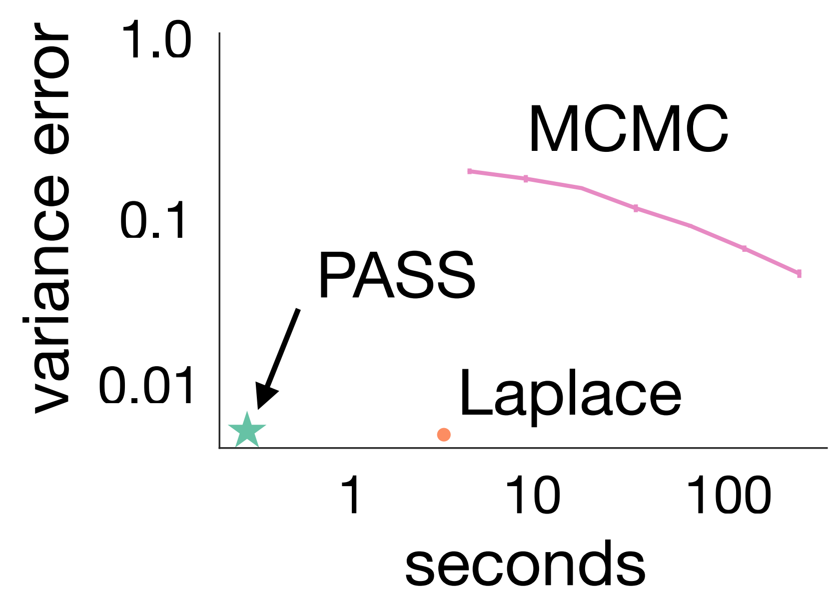
- **Goal:** Can we *prove* that PASS (or another likelihood approximation) will be accurate?
- If not, unsure if method is reliable

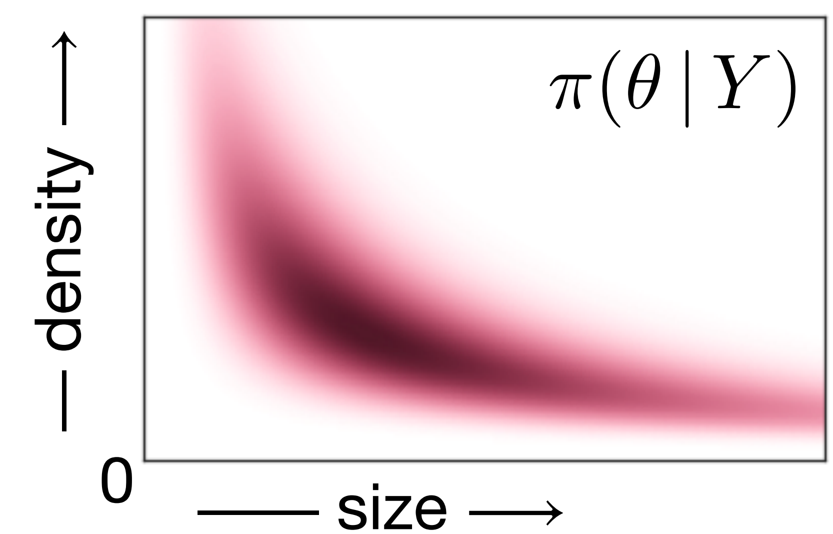


[**H**, Kasprzak, Campbell & Broderick 2018]

What about the next dataset?

- **Goal:** Can we *prove* that PASS (or another likelihood approximation) will be accurate?
- If not, unsure if method is reliable
- What's useful notion of accuracy?
- **What do we want from the approximation?**
  - Point estimate: mean
  - Uncertainty: standard deviation





Convenient…but meaningful?

- **Goal:** good mean and standard deviation estimates

[**H**, Kasprzak, Campbell & Broderick 2018]

Convenient…but meaningful?

- **Goal:** good mean and standard deviation estimates
- **Computationally convenient:** Kullback–Leibler divergence

$$\mathsf{KL}(q||\pi) = \mathbb{E}_q\left[\log\frac{q(\theta)}{\pi(\theta \mid Y)}\right] = \mathbb{E}_q\left[\log\frac{q(\theta)}{p(Y \mid \theta)\pi_0(\theta)}\right] + \text{constant}$$

[**H**, Kasprzak, Campbell & Broderick 2018]

Convenient…but meaningful?

- **Goal:** good mean and standard deviation estimates
- **Computationally convenient:** Kullback–Leibler divergence

$$\mathrm{KL}(q||\pi) = \mathbb{E}_q\left[\log \frac{q(\theta)}{\pi(\theta \mid Y)}\right] = \mathbb{E}_q\left[\log \frac{q(\theta)}{p(Y \mid \theta)\pi_0(\theta)}\right] + \text{constant}$$

Proposition [HKCB18]

There exist q and π such that $\mathrm{stdev}(q) = 1$ and $\mathrm{stdev}(\pi) = \infty$ but

$$\mathrm{KL}(q||\pi) < 1$$

[H, Kasprzak, Campbell & Broderick 2018]

Convenient…but meaningful?

- **Goal:** good mean and standard deviation estimates
- **Computationally convenient:** Kullback–Leibler divergence

$$\mathsf{KL}(q||\pi) = \mathbb{E}_q\left[\log \frac{q(\theta)}{\pi(\theta \mid Y)}\right] = \mathbb{E}_q\left[\log \frac{q(\theta)}{p(Y \mid \theta)\pi_0(\theta)}\right] + \text{constant}$$

Proposition [HKCB18]

There exist q and π such that $\mathrm{stdev}(q) = 1$ and $\mathrm{stdev}(\pi) = \infty$ but

$$\mathsf{KL}(q||\pi) < 1$$

Proposition [HKCB18]

For Gaussians q and π such that $\mathrm{stdev}(q) = 1$, it is possible that

$$|\,\mathrm{mean}(q) - \mathrm{mean}(\pi)| = e^{\mathsf{KL}(q||\pi)}$$

[H, Kasprzak, Campbell & Broderick 2018]

Meaningful…but convenient?

Better approximation properties: Wasserstein distance

$$\mathrm{W}(\pi, q)^2 = \inf_{\gamma \in \Gamma(\pi, q)} \int \|\theta - \theta'\|_2^2 \gamma(\mathrm{d}\theta, \mathrm{d}\theta')$$

[**H**, Kasprzak, Campbell & Broderick 2018]

Meaningful…but convenient?

Better approximation properties: Wasserstein distance

$$\mathrm{W}(\pi, q)^2 = \inf_{\gamma \in \Gamma(\pi, q)} \int \|\theta - \theta'\|_2^2 \gamma(\mathrm{d}\theta, \mathrm{d}\theta')$$

Theorem [HKCB18]

$$|\,\mathrm{mean}(\pi) - \mathrm{mean}(q)| \leq \mathrm{W}(\pi, q)$$
$$|\,\mathrm{stdev}(\pi) - \mathrm{stdev}(q)| \leq 2\mathrm{W}(\pi, q)$$

[H, Kasprzak, Campbell & Broderick 2018]

Meaningful…but convenient?

Better approximation properties: Wasserstein distance

$$\mathrm{W}(\pi, q)^2 = \inf_{\gamma \in \Gamma(\pi, q)} \int \|\theta - \theta'\|_2^2 \gamma(\mathrm{d}\theta, \mathrm{d}\theta')$$

Theorem [HKCB18]

$$|\operatorname{mean}(\pi) - \operatorname{mean}(q)| \leq \mathrm{W}(\pi, q)$$
$$|\operatorname{stdev}(\pi) - \operatorname{stdev}(q)| \leq 2\mathrm{W}(\pi, q)$$

- **But**, cannot compute Wasserstein distance efficiently

[H, Kasprzak, Campbell & Broderick 2018]

Meaningful…but convenient?

Better approximation properties: Wasserstein distance

$$\mathrm{W}(\pi, q)^2 = \inf_{\gamma \in \Gamma(\pi, q)} \int \|\theta - \theta'\|_2^2 \gamma(\mathrm{d}\theta, \mathrm{d}\theta')$$

Theorem [HKCB18]

$$|\,\mathrm{mean}(\pi) - \mathrm{mean}(q)| \le \mathrm{W}(\pi, q)$$
$$|\,\mathrm{stdev}(\pi) - \mathrm{stdev}(q)| \le 2\mathrm{W}(\pi, q)$$

- **But**, cannot compute Wasserstein distance efficiently
- **Goal:** computational efficiency of Kullback–Leibler divergence *and* guarantees of Wasserstein distance

[**H**, Kasprzak, Campbell & Broderick 2018]

A meaningful and convenient accuracy measure

Many Wasserstein guarantees for MCMC…

[Johnson & Barron 2004, Bolley et al. 2012,
Ley & Swan 2013, **H** & Zou 2017, **H** et al. 2018]

A meaningful and convenient accuracy measure

Many Wasserstein guarantees for MCMC…

…but not for likelihood approximations

[Johnson & Barron 2004, Bolley et al. 2012,
Ley & Swan 2013, **H** & Zou 2017, **H** et al. 2018]

A meaningful and convenient accuracy measure

Many Wasserstein guarantees for MCMC…

…but not for likelihood approximations

$$\pi(\theta \mid Y) \propto p(Y \mid \theta)\pi_0(\theta)$$

$$q(\theta)$$

[Johnson & Barron 2004, Bolley et al. 2012,
Ley & Swan 2013, **H** & Zou 2017, **H** et al. 2018]

A meaningful and convenient accuracy measure

Many Wasserstein guarantees for MCMC…

…but not for likelihood approximations

$$\pi(\theta \mid Y) \propto p(Y \mid \theta)\pi_0(\theta)$$
$$q(\theta)$$

$$\nabla \log \pi(\theta \mid Y) = \nabla \log p(Y \mid \theta)\pi_0(\theta)$$
$$\nabla \log q(\theta)$$

[Johnson & Barron 2004, Bolley et al. 2012, Ley & Swan 2013, **H** & Zou 2017, **H** et al. 2018]

A meaningful and convenient accuracy measure

Many Wasserstein guarantees for MCMC…

…but not for likelihood approximations

$$\pi(\theta \mid Y) \propto p(Y \mid \theta)\pi_0(\theta) \qquad \nabla \log \pi(\theta \mid Y) = \nabla \log p(Y \mid \theta)\pi_0(\theta)$$

$$q(\theta) \qquad \nabla \log q(\theta)$$

η-Fisher distance: $\mathsf{F}_\eta(\pi, q) = \mathbb{E}_\eta\left[\|\nabla \log \pi - \nabla \log q\|_2^2\right]^{1/2}$

[Johnson & Barron 2004, Bolley et al. 2012, Ley & Swan 2013, **H** & Zou 2017, **H** et al. 2018]

A meaningful and convenient accuracy measure

Many Wasserstein guarantees for MCMC…

…but not for likelihood approximations

$$\pi(\theta \mid Y) \propto p(Y \mid \theta)\pi_0(\theta) \qquad \nabla \log \pi(\theta \mid Y) = \nabla \log p(Y \mid \theta)\pi_0(\theta)$$

$$q(\theta) \qquad \nabla \log q(\theta)$$

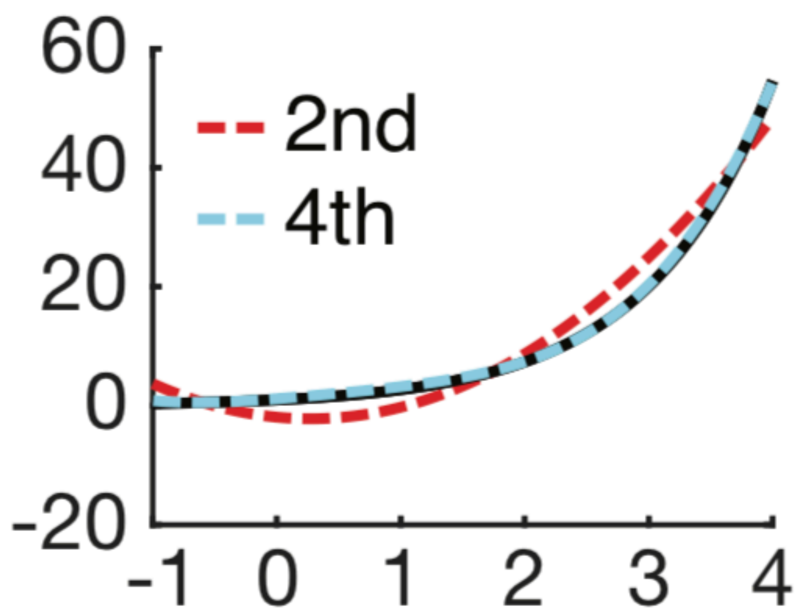
η-Fisher distance: $\mathsf{F}_\eta(\pi, q) = \mathbb{E}_\eta \left[\|\nabla \log \pi - \nabla \log q\|_2^2 \right]^{1/2}$

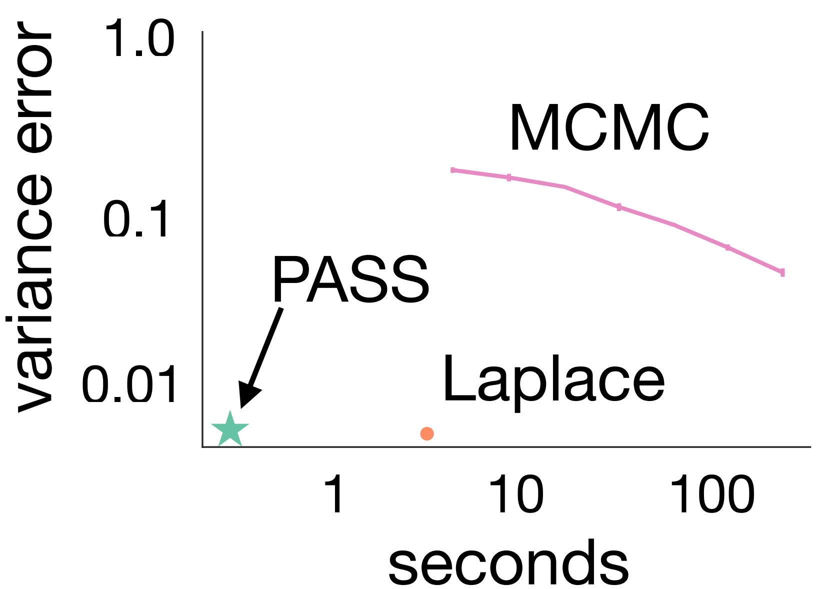
Theorem [HZ17, HKCB18]

$$\mathsf{W}(\pi, q) \leq C(q)C'(\eta, \pi)\mathsf{F}_\eta(\pi, q)$$

[Johnson & Barron 2004, Bolley et al. 2012, Ley & Swan 2013, **H** & Zou 2017, **H** et al. 2018]

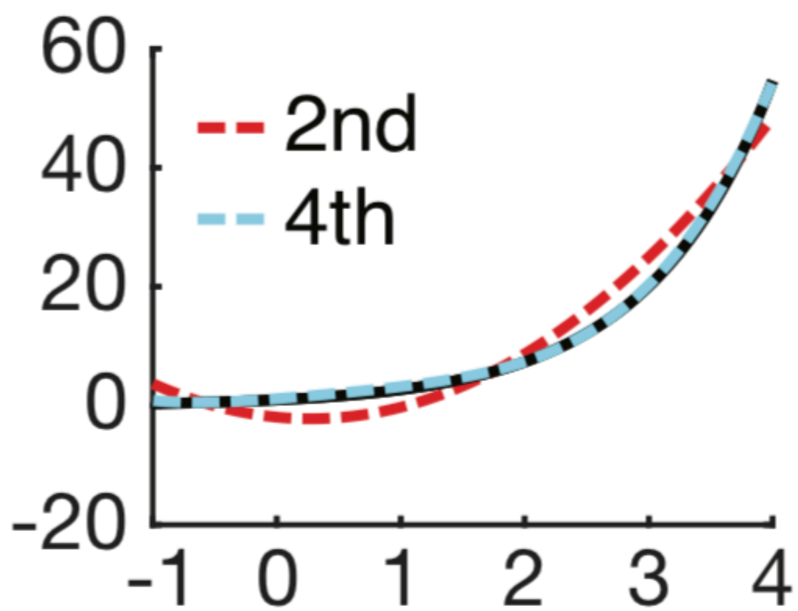
Application: PASS reliably provides a high-quality approximation

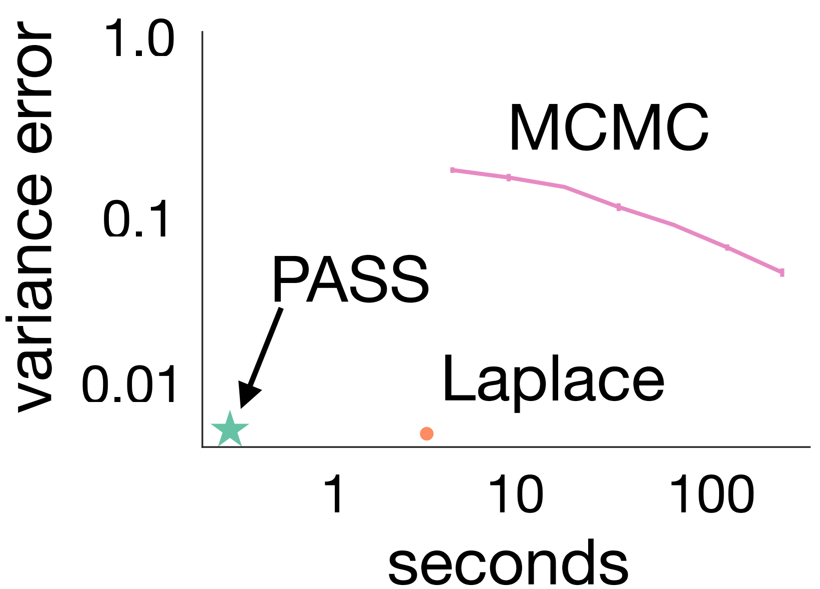




[H, Adams & Broderick 2017, H & Zou 2017, H et al. 2018]

Application: PASS reliably provides a high-quality approximation





Theorem [HAB17]

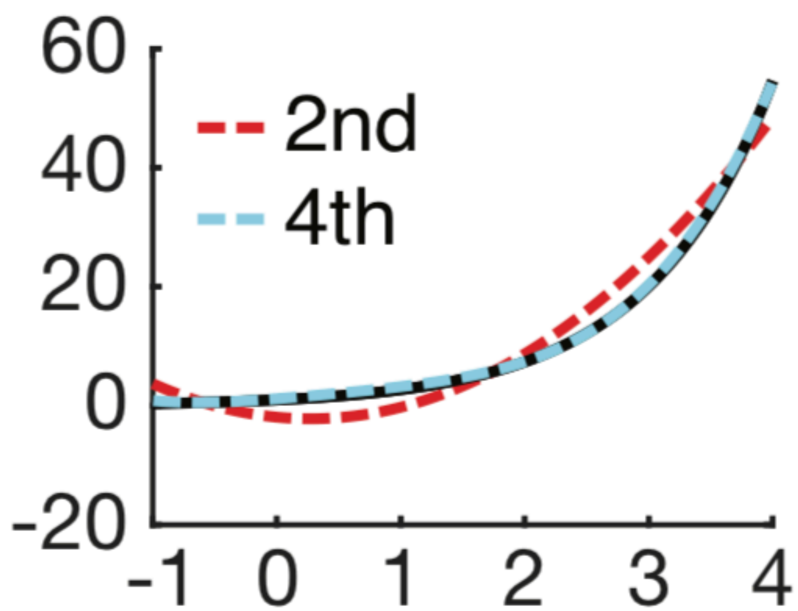
Let $q_M =$ the PASS approximate posterior using degree M polynomials.

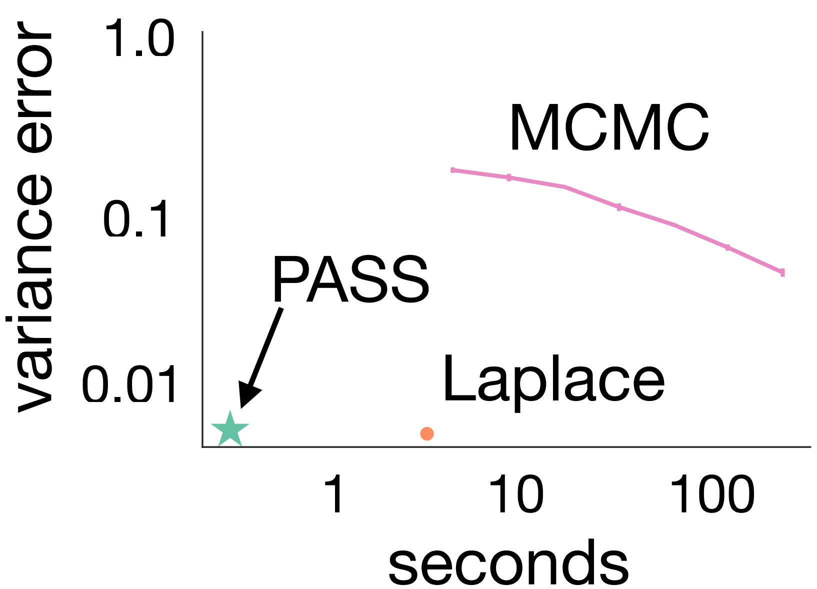
Then the Wasserstein distance decreases exponentially in M:

$$\mathrm{W}(\pi, q_M) \leq c r^M$$

[H, Adams & Broderick 2017, H & Zou 2017, H et al. 2018]

Application: PASS reliably provides a high-quality approximation





Theorem [H AB17]

Let $q_M =$ the PASS approximate posterior using degree M polynomials.

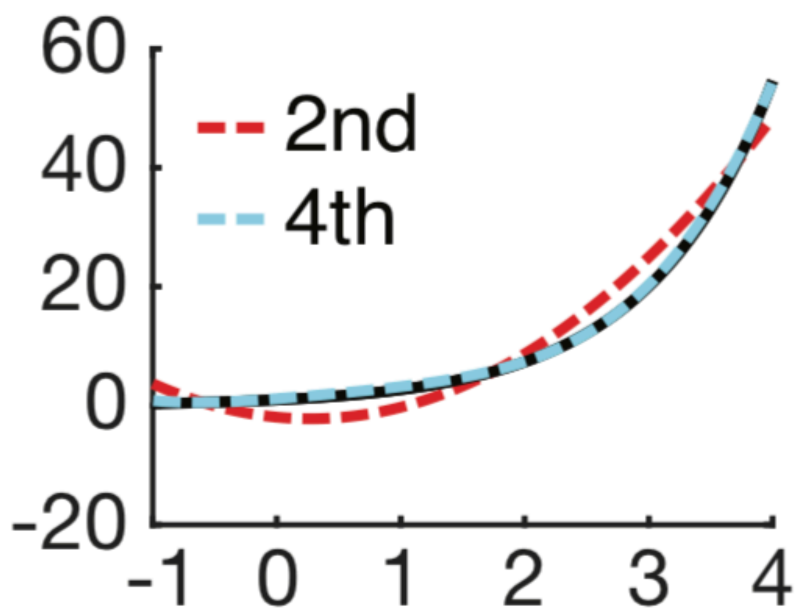
Then the Wasserstein distance decreases exponentially in M:

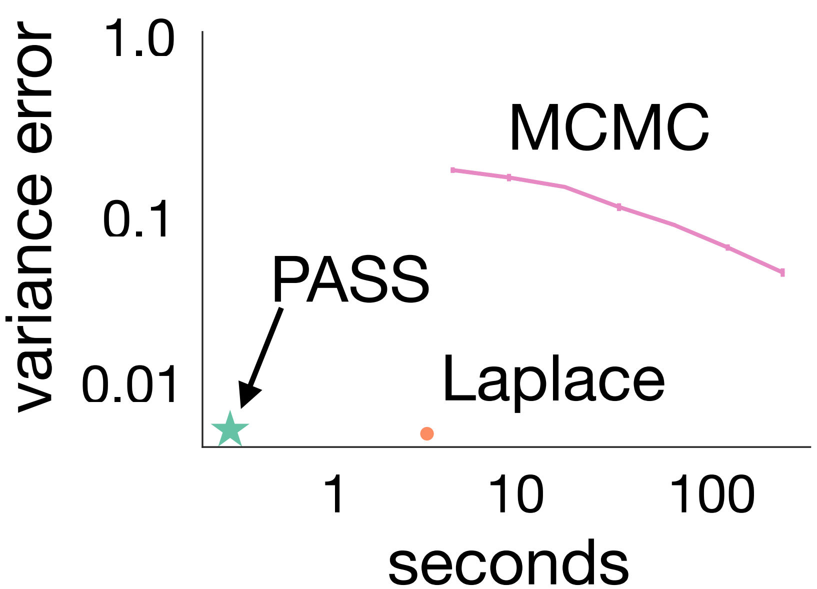
$$\mathrm{W}(\pi, q_M) \leq cr^M$$

- **Benefit:** confidence to use PASS with a new dataset

[H, Adams & Broderick 2017, H & Zou 2017, H et al. 2018]

Application: PASS reliably provides a high-quality approximation





Theorem [HAB17]

Let $q_M =$ the PASS approximate posterior using degree M polynomials.

Then the Wasserstein distance decreases exponentially in M:

$$\mathrm{W}(\pi, q_M) \leq cr^M$$

- **Benefit:** confidence to use PASS with a new dataset
- Can also use η**-Fisher distance** to prove accuracy bounds for other likelihood approximations (e.g. Laplace approximation and coresets).

[H, Adams & Broderick 2017, H & Zou 2017, H et al. 2018]

References

Theory

Huggins, Kasprzak, Campbell & Broderick. *Practical bounds on the error of Bayesian posterior approximations: A nonasymptotic approach.* arXiv:1809.09505 [stat.TH], 2018.

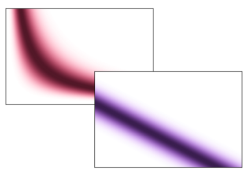
Huggins* & Zou*. *Quantifying the accuracy of approximate diffusions and Markov chains.* AISTATS, 2017.

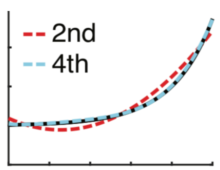
Applications

Huggins, Campbell, Kasprzak & Broderick. *Scalable Gaussian process inference with finite-data mean and variance guarantees*. AISTATS, 2019.

Huggins, Adams & Broderick. *PASS-GLM: polynomial approximate sufficient statistics for scalable Bayesian GLM inference*. Neural Information Processing Systems, 2017.

Agenda

 A framework for scalable Bayesian inference

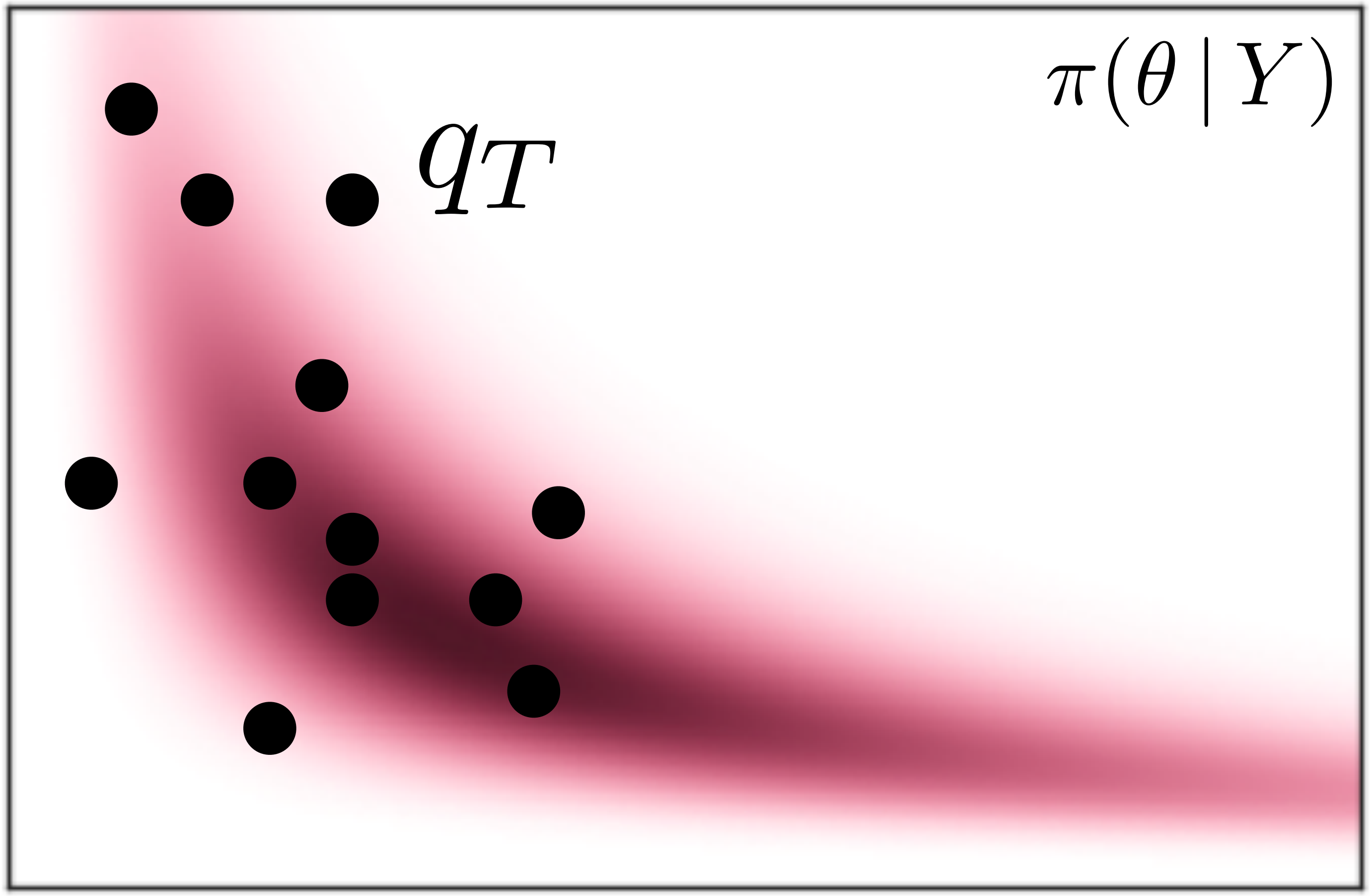


Algorithm design

F_η Meaningful accuracy guarantees

➡ **Validating results from heuristic algorithms**

Is that heuristic approximation any good?



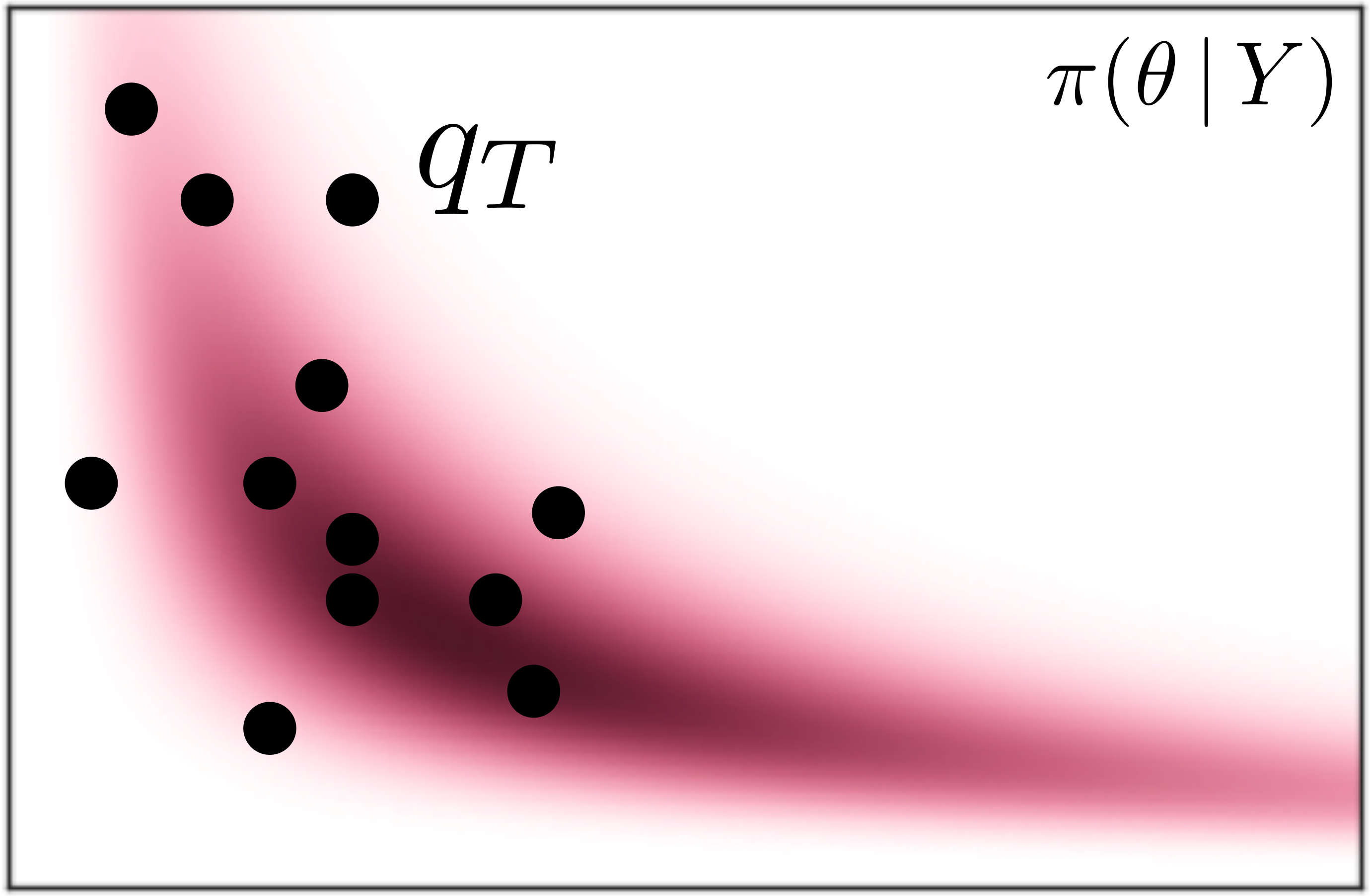
[Gorham & Mackey 2015, **H** & Mackey 2018]

Is that heuristic approximation any good?

Goals

1. approximation quality

$$q_T \approx \pi?$$



[Gorham & Mackey 2015, **H** & Mackey 2018]

Is that heuristic approximation any good?

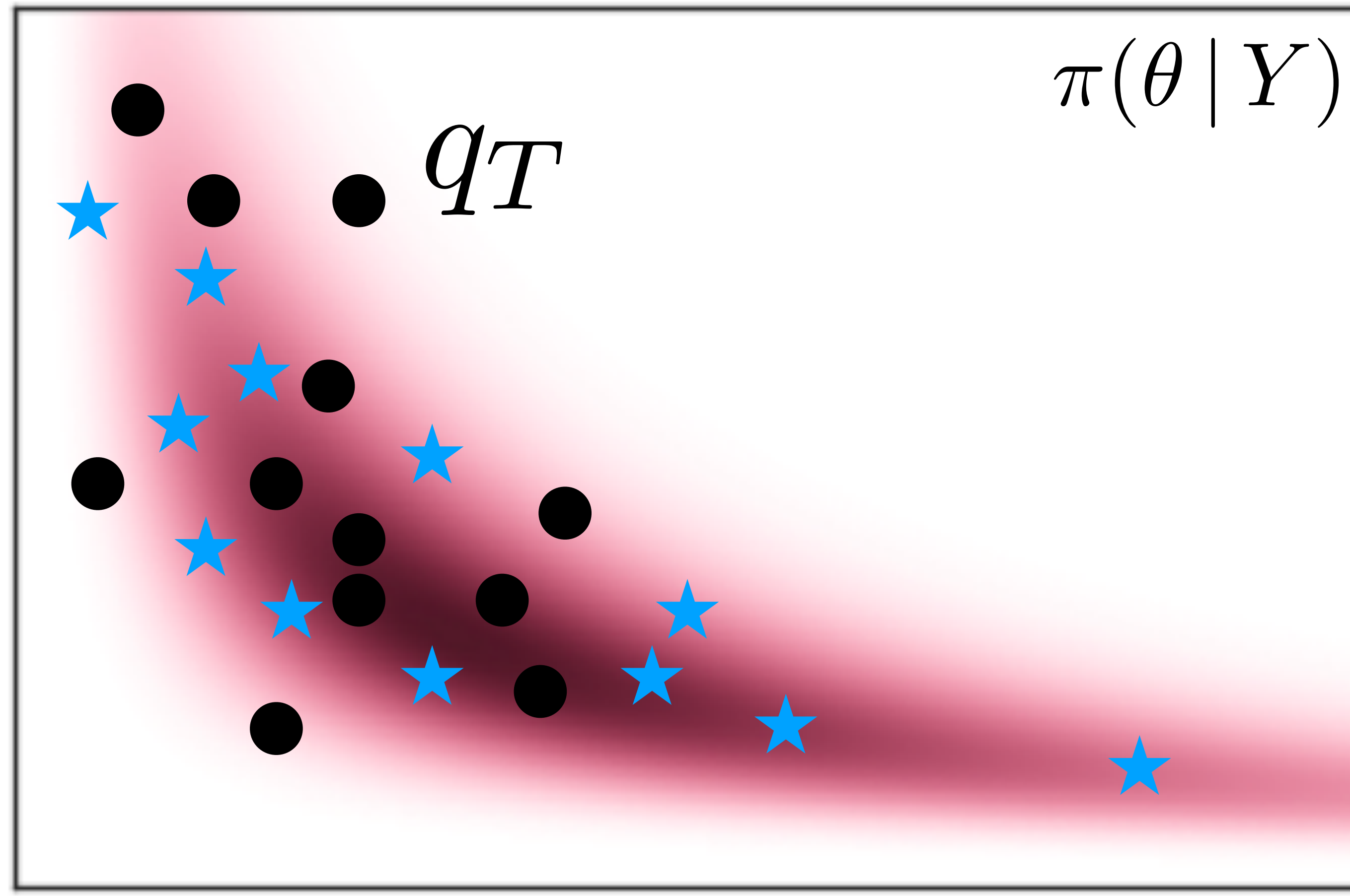
Goals

1. approximation quality

$$q_T \approx \pi?$$

2. algorithm selection

★ **versus** ●



[Gorham & Mackey 2015, **H** & Mackey 2018]

Is that heuristic approximation any good?

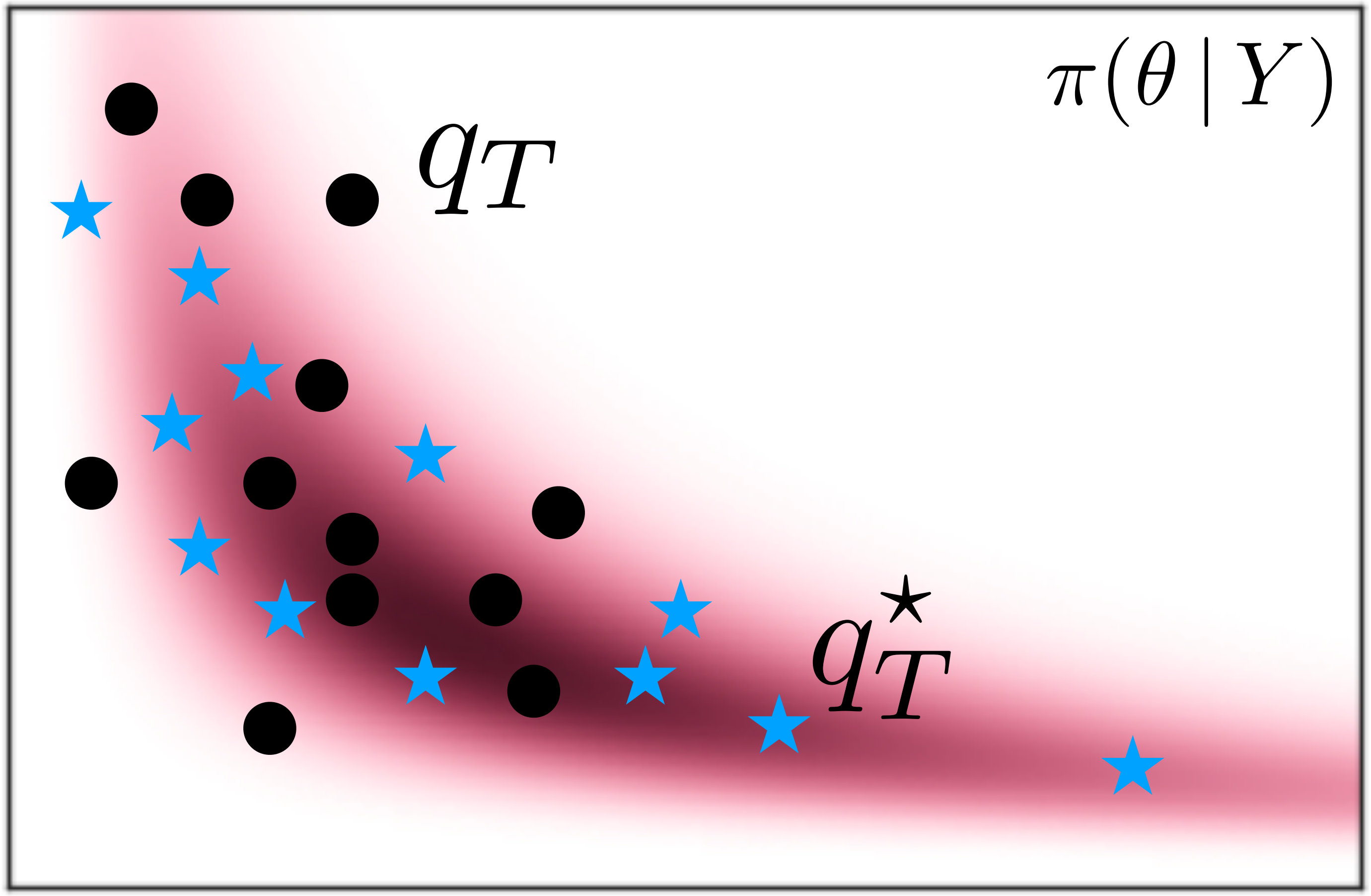
Goals

1. approximation quality

$$q_T \approx \pi?$$

2. algorithm selection

★ **versus** ●

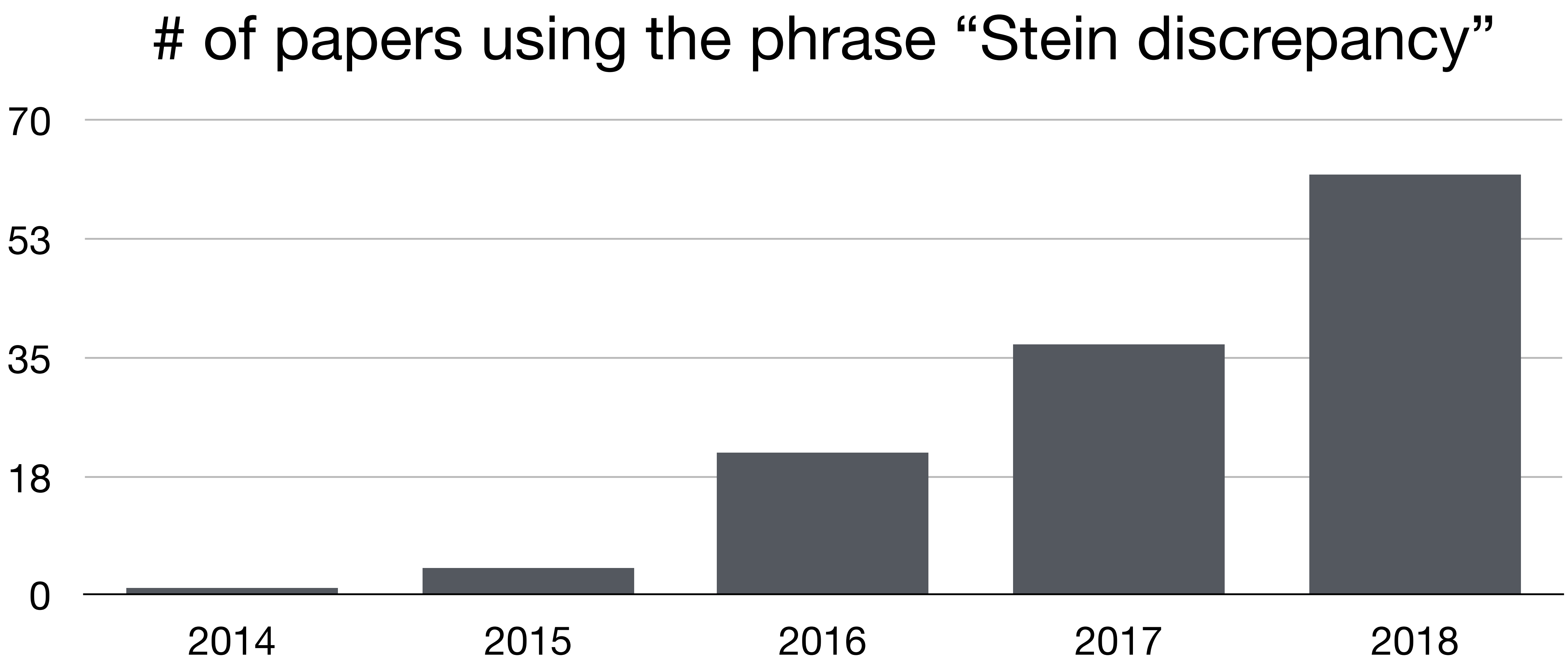


Approach: use a discrepancy measure $d(\pi, q_T)$

- **Goal 1:** is $d(\pi, q_T) \approx 0$?
- **Goal 2:** is $d(\pi, q_T^\star)$ or $d(\pi, q_T)$ smaller?

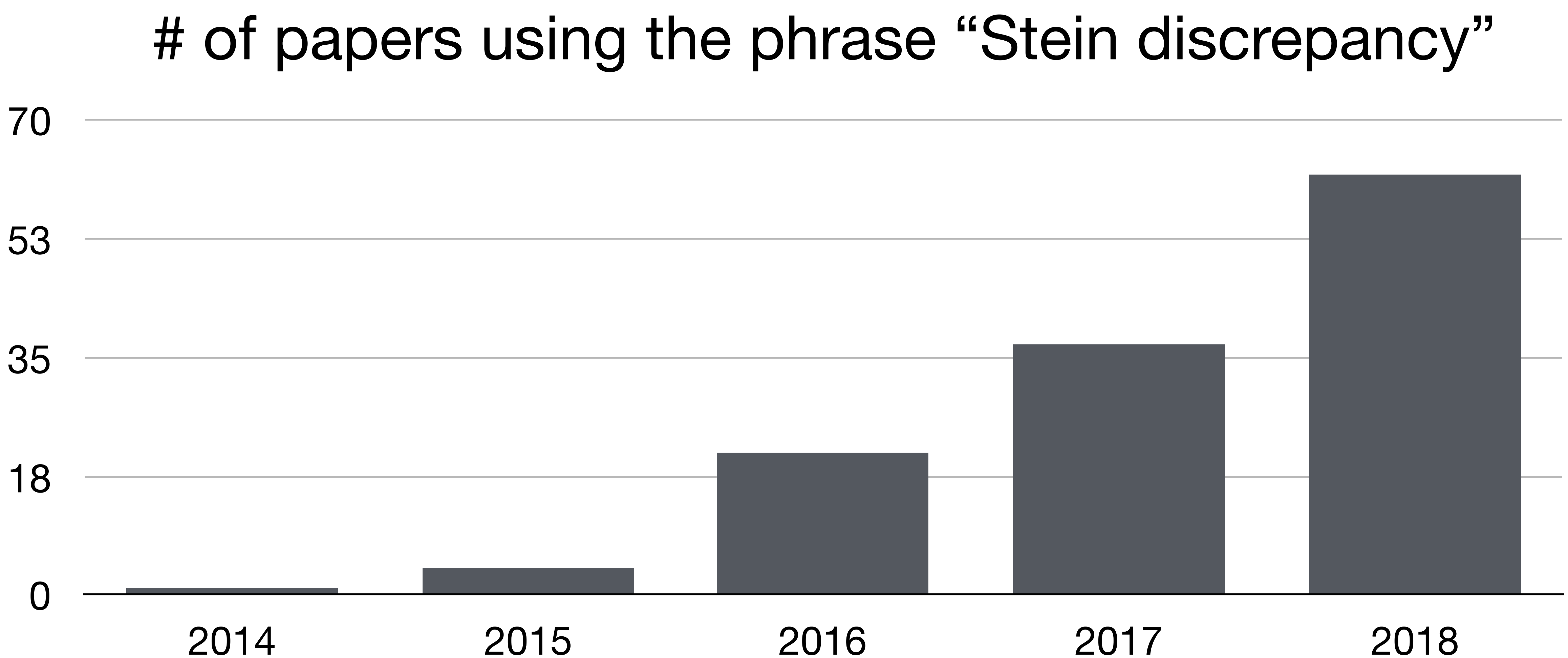
[Gorham & Mackey 2015, H & Mackey 2018]

Approach: Stein discrepancies



[Gorham & Mackey 2015, **H** & Mackey 2018]

Approach: Stein discrepancies

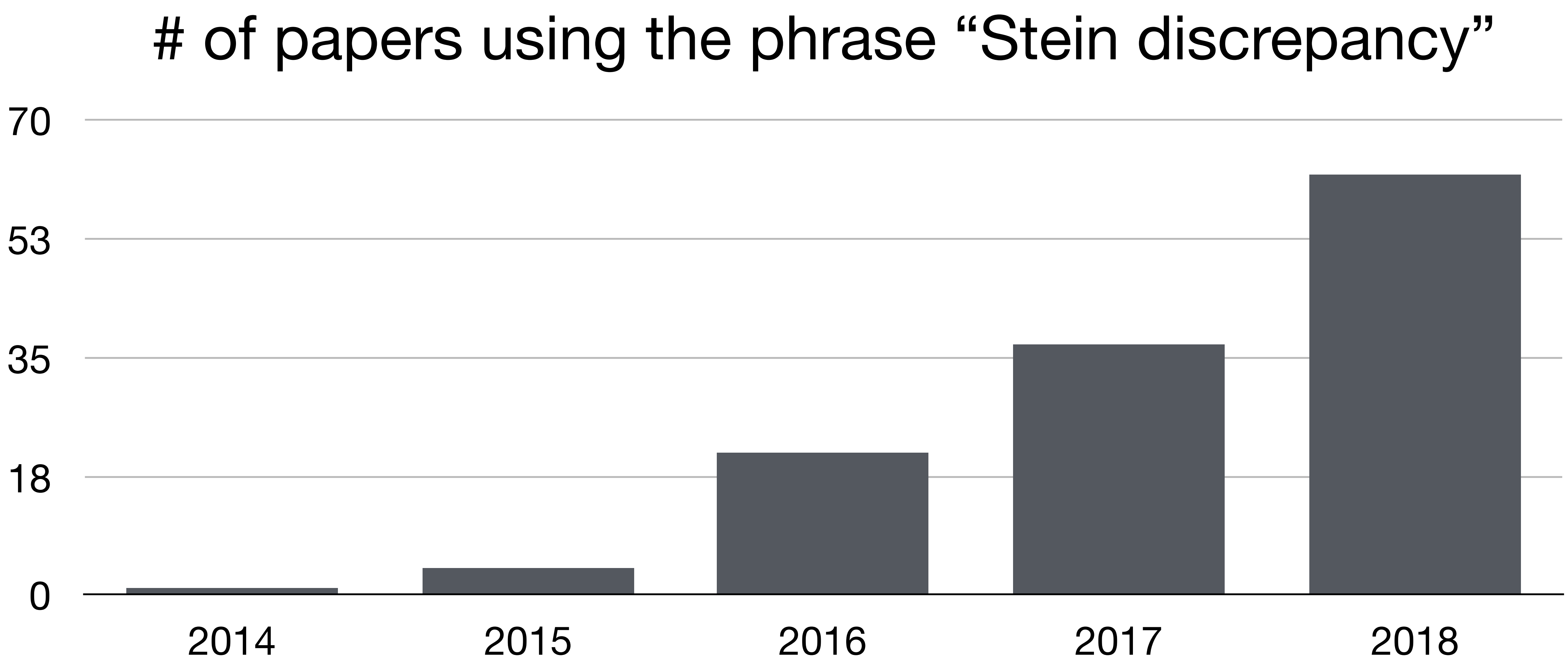


Definition

$d(\pi, q_T)$ is **theoretically sound** if it detects (non-)convergence of $q_T \to \pi$ as $T \to \infty$

[Gorham & Mackey 2015, **H** & Mackey 2018]

Approach: Stein discrepancies



Definition

$d(\pi, q_T)$ is **theoretically sound** if it detects (non-)convergence of $q_T \to \pi$ as $T \to \infty$

We provide the first discrepancy measure that is

✓ fast

✓ theoretically sound

[Gorham & Mackey 2015, **H** & Mackey 2018]

Dilemma: soundness or speed

$q_T = T^{-1} \sum_{t=1}^{T} \delta_{\theta_t}$ kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$

[Jitkrittum et al. 2017, Chwialkowski et al. 2015, Gorham & Mackey 2017]

Dilemma: soundness or speed

$q_T = T^{-1} \sum_{t=1}^{T} \delta_{\theta_t}$ kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$

Theoretically sound approach: kernel Stein discrepancies (KSDs)

$$S_k(\pi, q_T) = T^{-2} \sum_{t=1}^{T} \sum_{t'=1}^{T} (\mathcal{T}_\pi \otimes \mathcal{T}_\pi) k(\theta_t, \theta_{t'})$$

[Jitkrittum et al. 2017, Chwialkowski et al. 2015, Gorham & Mackey 2017]

Dilemma: soundness or speed

$q_T = T^{-1} \sum_{t=1}^{T} \delta_{\theta_t}$ kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$

Theoretically sound approach: kernel Stein discrepancies (KSDs)

$$S_k(\pi, q_T) = T^{-2} \sum_{t=1}^{T} \sum_{t'=1}^{T} (\mathcal{T}_\pi \otimes \mathcal{T}_\pi) k(\theta_t, \theta_{t'})$$

- Stein operator: $\mathcal{T}_\pi(g)(\theta) = \frac{\mathrm{d}g}{\mathrm{d}\theta}(\theta) + g(\theta) \frac{\mathrm{d} \log \pi}{\mathrm{d}\theta}(\theta)$

[Jitkrittum et al. 2017, Chwialkowski et al. 2015, Gorham & Mackey 2017]

Dilemma: soundness or speed

$q_T = T^{-1} \sum_{t=1}^{T} \delta_{\theta_t}$ kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$

Theoretically sound approach: kernel Stein discrepancies (KSDs)

$$S_k(\pi, q_T) = T^{-2} \sum_{t=1}^{T} \sum_{t'=1}^{T} (\mathcal{T}_\pi \otimes \mathcal{T}_\pi) k(\theta_t, \theta_{t'})$$

- Stein operator: $\mathcal{T}_\pi(g)(\theta) = \frac{\mathrm{d}g}{\mathrm{d}\theta}(\theta) + g(\theta) \frac{\mathrm{d}\log\pi}{\mathrm{d}\theta}(\theta)$

[Jitkrittum et al. 2017, Chwialkowski et al. 2015, Gorham & Mackey 2017]

Dilemma: soundness or speed

$q_T = T^{-1} \sum_{t=1}^{T} \delta_{\theta_t}$ kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$

Theoretically sound approach: kernel Stein discrepancies (KSDs)

$$S_k(\pi, q_T) = T^{-2} \sum_{t=1}^{T} \sum_{t'=1}^{T} (\mathcal{T}_\pi \otimes \mathcal{T}_\pi) k(\theta_t, \theta_{t'})$$

- Stein operator: $\mathcal{T}_\pi(g)(\theta) = \frac{\mathrm{d}g}{\mathrm{d}\theta}(\theta) + g(\theta)\frac{\mathrm{d}\log\pi}{\mathrm{d}\theta}(\theta)$
- **But too slow:** $\Theta(T^2)$ time

[Jitkrittum et al. 2017, Chwialkowski et al. 2015, Gorham & Mackey 2017]

Dilemma: soundness or speed

$q_T = T^{-1} \sum_{t=1}^{T} \delta_{\theta_t}$ kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$

Theoretically sound approach: kernel Stein discrepancies (KSDs)

$$S_k(\pi, q_T) = T^{-2} \sum_{t=1}^{T} \sum_{t'=1}^{T} (\mathcal{T}_\pi \otimes \mathcal{T}_\pi) k(\theta_t, \theta_{t'})$$

- Stein operator: $\mathcal{T}_\pi(g)(\theta) = \frac{\mathrm{d}g}{\mathrm{d}\theta}(\theta) + g(\theta)\frac{\mathrm{d}\log\pi}{\mathrm{d}\theta}(\theta)$
- **But too slow:** $\Theta(T^2)$ time

Fast approach: importance sampling approximation to KSD

[Jitkrittum et al. 2017, Chwialkowski et al. 2015, Gorham & Mackey 2017]

Dilemma: soundness or speed

$q_T = T^{-1} \sum_{t=1}^{T} \delta_{\theta_t}$ kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$

Theoretically sound approach: kernel Stein discrepancies (KSDs)

$$S_k(\pi, q_T) = T^{-2} \sum_{t=1}^{T} \sum_{t'=1}^{T} (\mathcal{T}_\pi \otimes \mathcal{T}_\pi) k(\theta_t, \theta_{t'})$$

- Stein operator: $\mathcal{T}_\pi(g)(\theta) = \frac{\mathrm{d}g}{\mathrm{d}\theta}(\theta) + g(\theta)\frac{\mathrm{d}\log\pi}{\mathrm{d}\theta}(\theta)$
- **But too slow:** $\Theta(T^2)$ time

Fast approach: importance sampling approximation to KSD

- since $k(\theta, \theta') = \int \psi(\theta, z)\psi(\theta', z)\mathrm{d}z$, for $Z_m \overset{\text{i.i.d.}}{\sim} \rho$

$$S_k(\pi, q_T) \approx M^{-1} \sum_{m=1}^{M} \rho(Z_m)^{-1} \left\{ T^{-1} \sum_{t=1}^{T} \mathcal{T}_\pi \psi(\theta_t, Z_m) \right\}^2$$

[Jitkrittum et al. 2017, Chwialkowski et al. 2015, Gorham & Mackey 2017]

Dilemma: soundness or speed

$$q_T = T^{-1} \sum_{t=1}^{T} \delta_{\theta_t} \qquad \text{kernel } k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$$

Theoretically sound approach: kernel Stein discrepancies (KSDs)

$$S_k(\pi, q_T) = T^{-2} \sum_{t=1}^{T} \sum_{t'=1}^{T} (\mathcal{T}_\pi \otimes \mathcal{T}_\pi) k(\theta_t, \theta_{t'})$$

- Stein operator: $\mathcal{T}_\pi(g)(\theta) = \frac{\mathrm{d}g}{\mathrm{d}\theta}(\theta) + g(\theta)\frac{\mathrm{d}\log\pi}{\mathrm{d}\theta}(\theta)$
- **But too slow:** $\Theta(T^2)$ time

Fast approach: importance sampling approximation to KSD

- since $k(\theta, \theta') = \int \psi(\theta, z)\psi(\theta', z)\mathrm{d}z$, for $Z_m \overset{\text{i.i.d.}}{\sim} \rho$

$$S_k(\pi, q_T) \approx M^{-1} \sum_{m=1}^{M} \rho(Z_m)^{-1} \left\{ T^{-1} \sum_{t=1}^{T} \mathcal{T}_\pi \psi(\theta_t, Z_m) \right\}^2$$

- $\Theta(MT)$ time complexity **but not theoretically sound**

[Jitkrittum et al. 2017, Chwialkowski et al. 2015, Gorham & Mackey 2017]

Random feature Stein discrepancies: fast *and* theoretically sound

[H & Mackey 2018]

Random feature Stein discrepancies: fast *and* theoretically sound

Our solution: design Stein discrepancy from the start for use with importance sampling

Random feature Stein discrepancies: fast *and* theoretically sound

Our solution: design Stein discrepancy from the start for use with importance sampling

Feature Stein discrepancies are theoretically sound:

$$\Phi\mathrm{SD}_{\Phi,r}(\pi, q_T) = \left[\int \left\{T^{-1}\sum_{t=1}^{T} \mathcal{T}_\pi \Phi(\theta_t, z)\right\}^r \mathrm{d}z\right]^{2/r}$$

[H & Mackey 2018]

Random feature Stein discrepancies: fast *and* theoretically sound

Our solution: design Stein discrepancy from the start for use with importance sampling

Feature Stein discrepancies are theoretically sound:

$$\Phi\mathrm{SD}_{\Phi,r}(\pi, q_T) = \left[\int \left\{T^{-1}\textstyle\sum_{t=1}^{T} \mathcal{T}_\pi \Phi(\theta_t, z)\right\}^r \mathrm{d}z\right]^{2/r}$$

Random feature Stein discrepancies are importance sampled approximations:

$$\mathrm{R}\Phi\mathrm{SD}_{\Phi,r}(\pi, q_T) = \left[M^{-1}\textstyle\sum_{m=1}^{M} \rho(Z_m)^{-1} \left\{T^{-1}\textstyle\sum_{t=1}^{T} \mathcal{T}_\pi \Phi(\theta_t, Z_m)\right\}^r\right]^{2/r}$$

[**H** & Mackey 2018]

Random feature Stein discrepancies: fast *and* theoretically sound

Our solution: design Stein discrepancy from the start for use with importance sampling

Feature Stein discrepancies are theoretically sound:

$$\Phi\mathrm{SD}_{\Phi,r}(\pi, q_T) = \left[\int \left\{T^{-1}\sum_{t=1}^{T} \mathcal{T}_\pi \Phi(\theta_t, z)\right\}^r \mathrm{d}z\right]^{2/r}$$

Random feature Stein discrepancies are importance sampled approximations:

$$\mathrm{R}\Phi\mathrm{SD}_{\Phi,r}(\pi, q_T) = \left[M^{-1}\sum_{m=1}^{M} \rho(Z_m)^{-1} \left\{T^{-1}\sum_{t=1}^{T} \mathcal{T}_\pi \Phi(\theta_t, Z_m)\right\}^r\right]^{2/r}$$

Recall: $\Theta(MT)$ time complexity when using M importance samples

Random feature Stein discrepancies: fast *and* theoretically sound

Our solution: design Stein discrepancy from the start for use with importance sampling

Feature Stein discrepancies are theoretically sound:

$$\Phi\mathrm{SD}_{\Phi,r}(\pi, q_T) = \left[\int \left\{T^{-1}\textstyle\sum_{t=1}^{T} \mathcal{T}_\pi \Phi(\theta_t, z)\right\}^r \mathrm{d}z\right]^{2/r}$$

Random feature Stein discrepancies are importance sampled approximations:

$$\mathrm{R}\Phi\mathrm{SD}_{\Phi,r}(\pi, q_T) = \left[M^{-1}\textstyle\sum_{m=1}^{M} \rho(Z_m)^{-1} \left\{T^{-1}\textstyle\sum_{t=1}^{T} \mathcal{T}_\pi \Phi(\theta_t, Z_m)\right\}^r\right]^{2/r}$$

Recall: $\Theta(MT)$ time complexity when using M importance samples

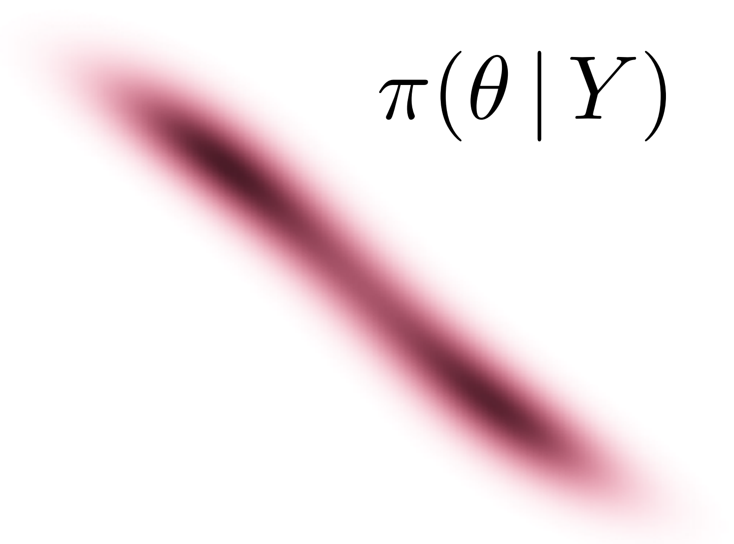
Theorem [HM18]

For any $\alpha > 0$, we can compute a theoretically sound random feature Stein discrepancy using $M = \Theta(T^\alpha)$ importance samples in near-linear $\Theta(T^{1+\alpha})$ time.

[H & Mackey 2018]

Application #1: selecting the best inference algorithm

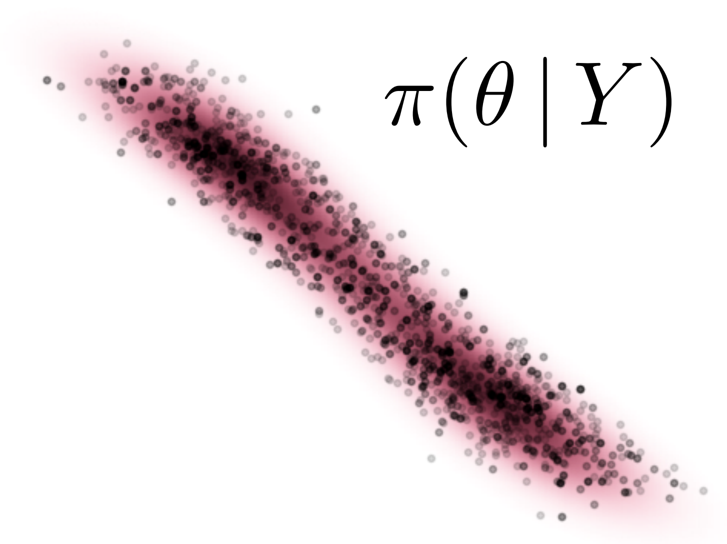
“exact” MCMC



$\pi(\theta \,|\, Y)$

Application #1: selecting the best inference algorithm

"exact" MCMC



$$\pi(\theta \mid Y)$$

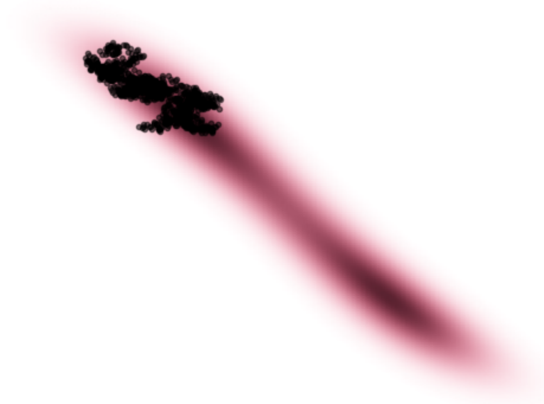
Application #1: selecting the best inference algorithm

“exact” MCMC

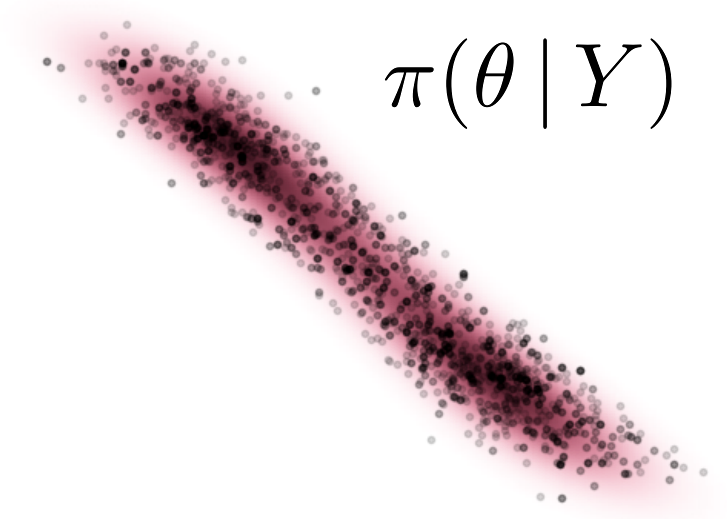
approximate MCMC(ε)

small $\varepsilon =$ less bias, slower exploration

ε = 0.0005

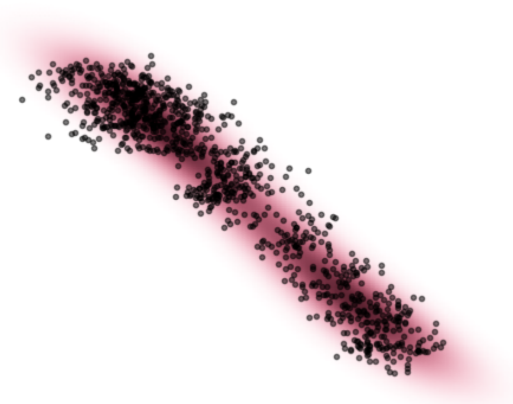


$$\pi(\theta \mid Y)$$

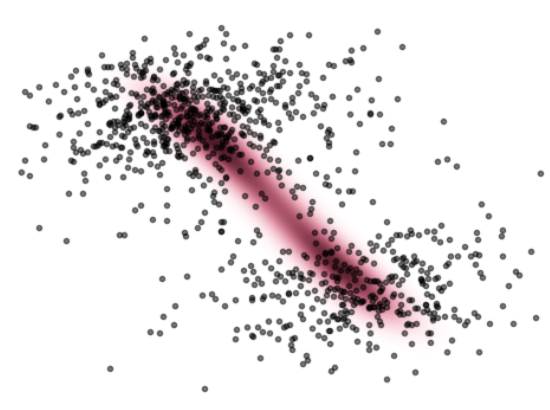


vs

ε = 0.005



ε = 0.05



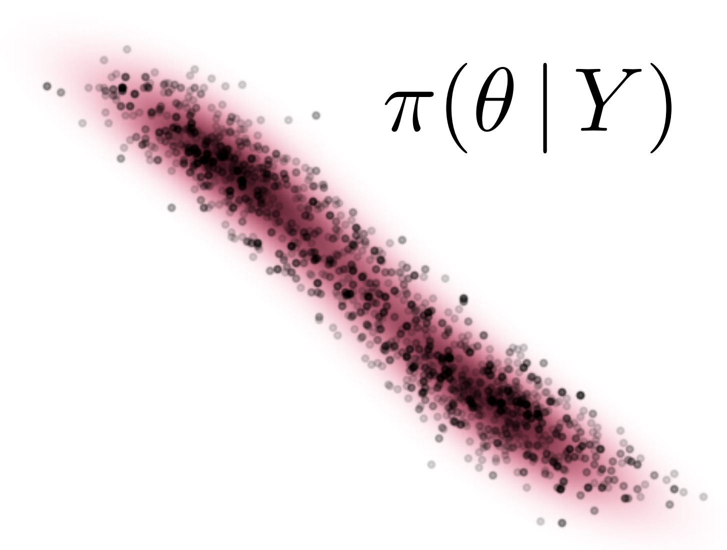
[**H** & Mackey 2018]

Application #1: selecting the best inference algorithm

“exact” MCMC

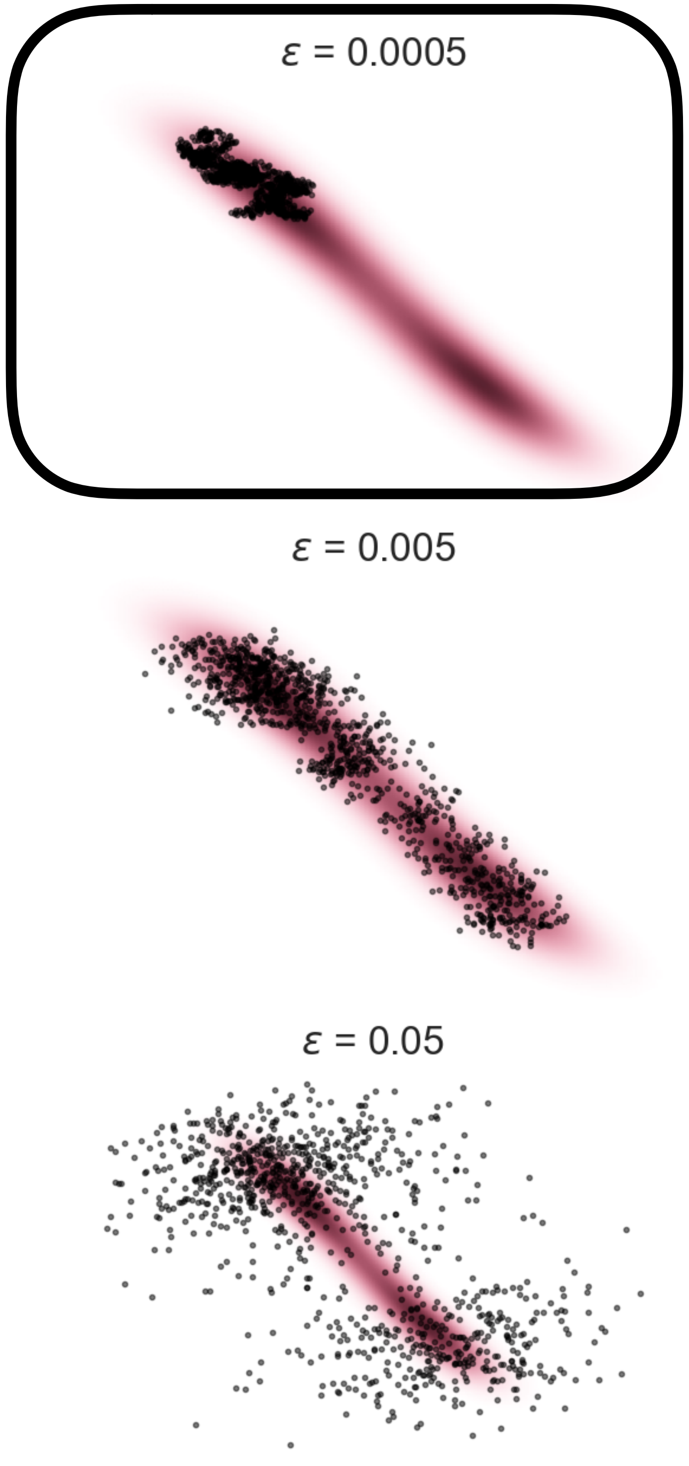
approximate MCMC(ε)

small $\varepsilon =$ less bias, slower exploration



$\pi(\theta \mid Y)$

vs



[**H** & Mackey 2018]

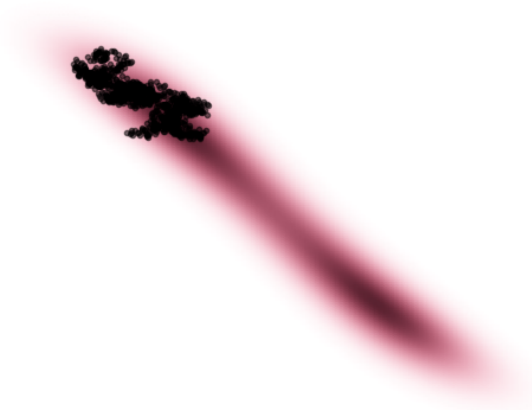
Application #1: selecting the best inference algorithm

"exact" MCMC

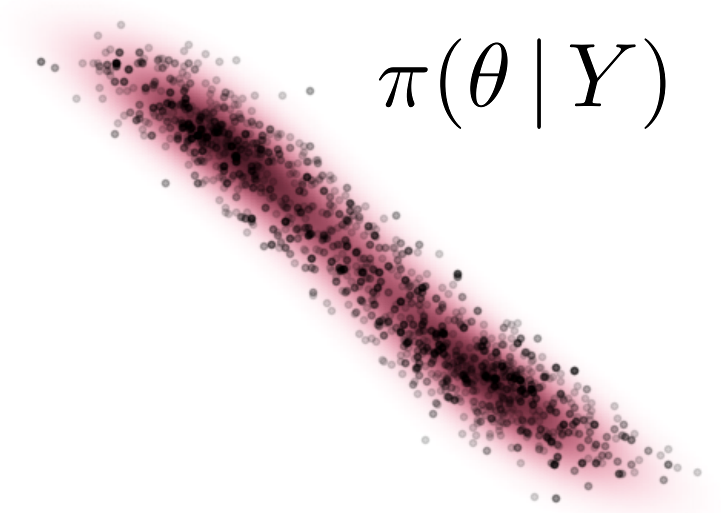
approximate MCMC(ε)

small ε = less bias, slower exploration

ε = 0.0005

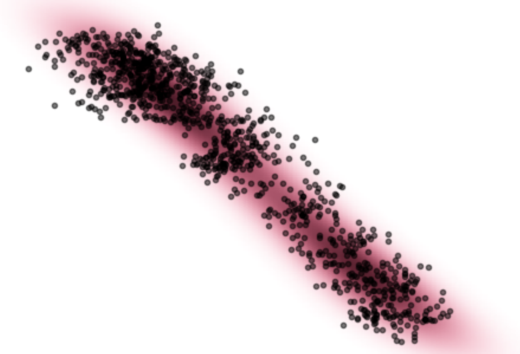


$\pi(\theta \mid Y)$

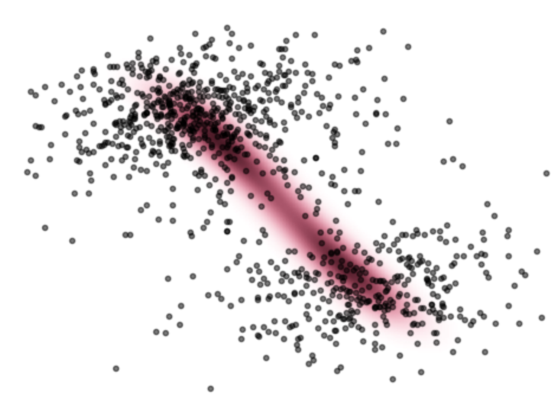


vs

ε = 0.005



ε = 0.05



[**H** & Mackey 2018]

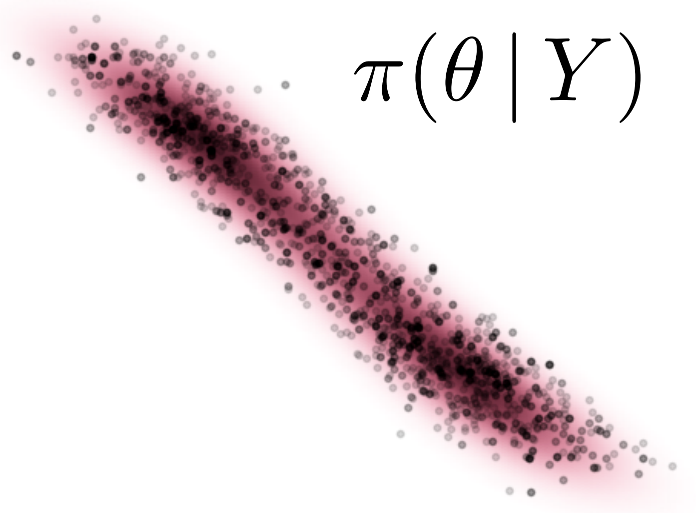
Application #1: selecting the best inference algorithm

"exact" MCMC

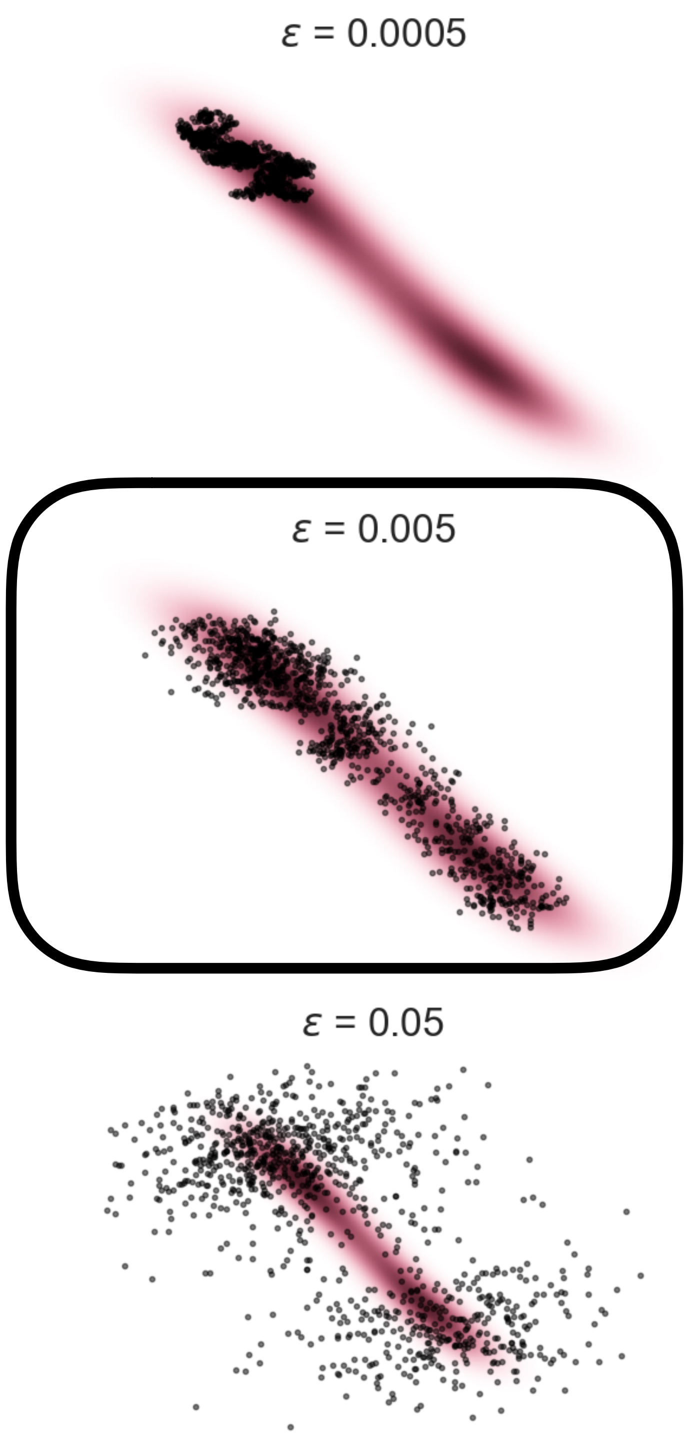
approximate MCMC(ε)

small $\varepsilon =$ less bias, slower exploration

$\pi(\theta \mid Y)$



vs



[H & Mackey 2018]

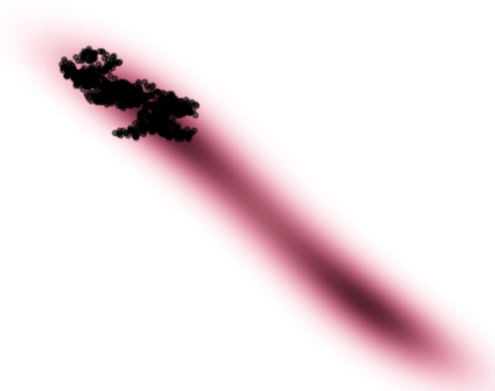
Application #1: selecting the best inference algorithm

"exact" MCMC

approximate MCMC(ε)

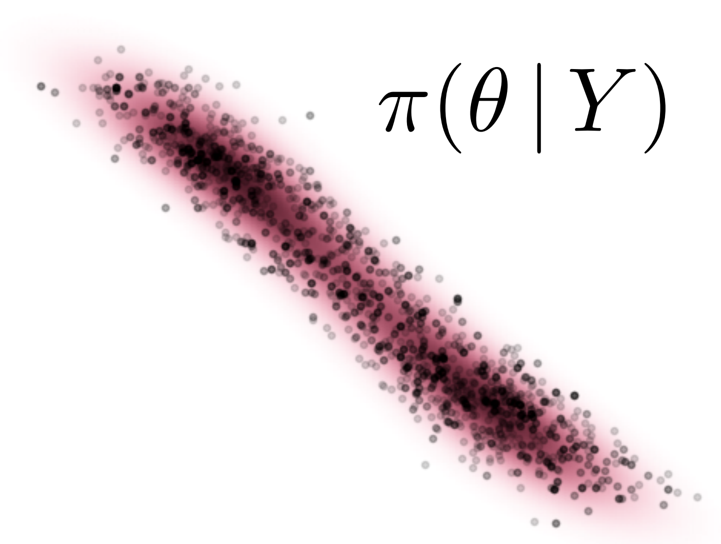
small ε = less bias, slower exploration

ε = 0.0005



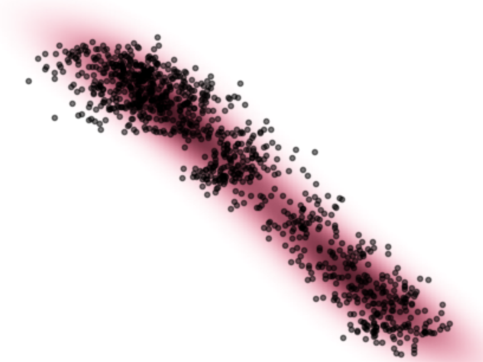
Our method and quadratic-time method select same ε value

$\pi(\theta \mid Y)$

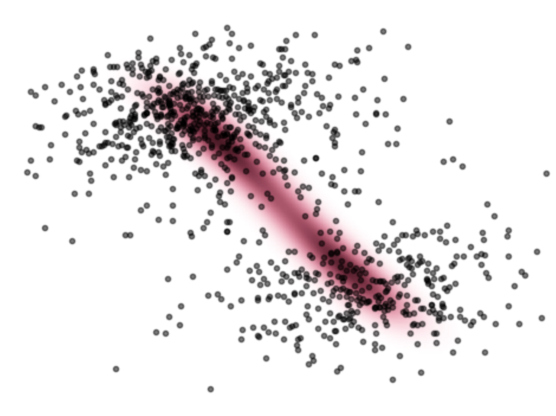


vs

ε = 0.005



ε = 0.05



[**H** & Mackey 2018]

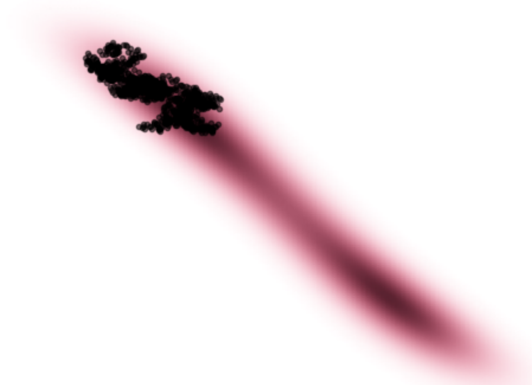
Application #1: selecting the best inference algorithm

"exact" MCMC

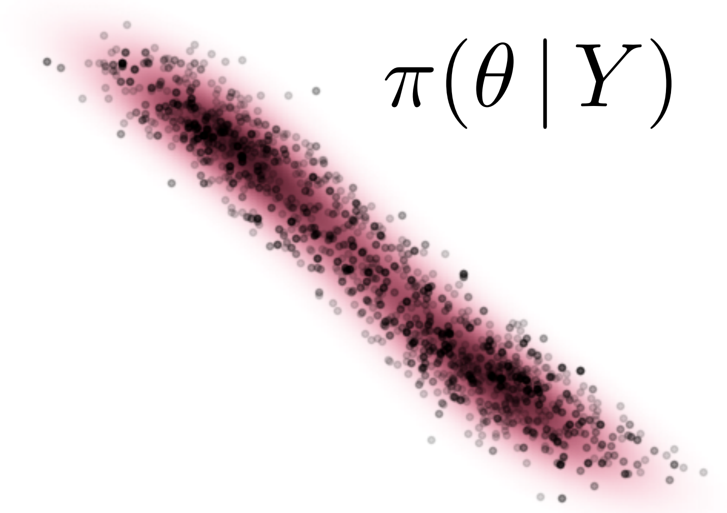
approximate MCMC(ε)

small ε = less bias, slower exploration

ε = 0.0005



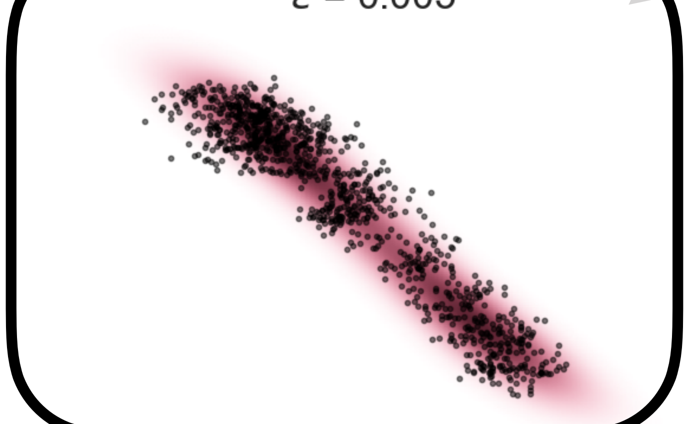
Our method and quadratic-time method select same ε value



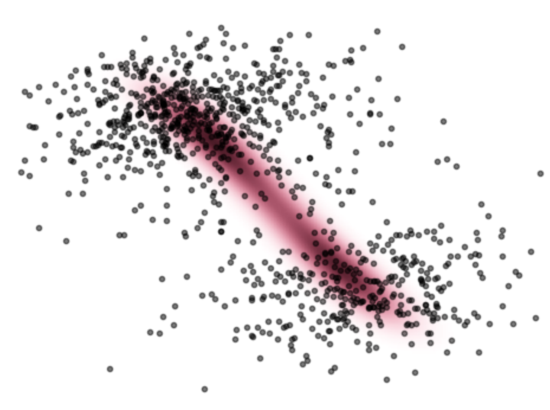
$\pi(\theta \mid Y)$

vs

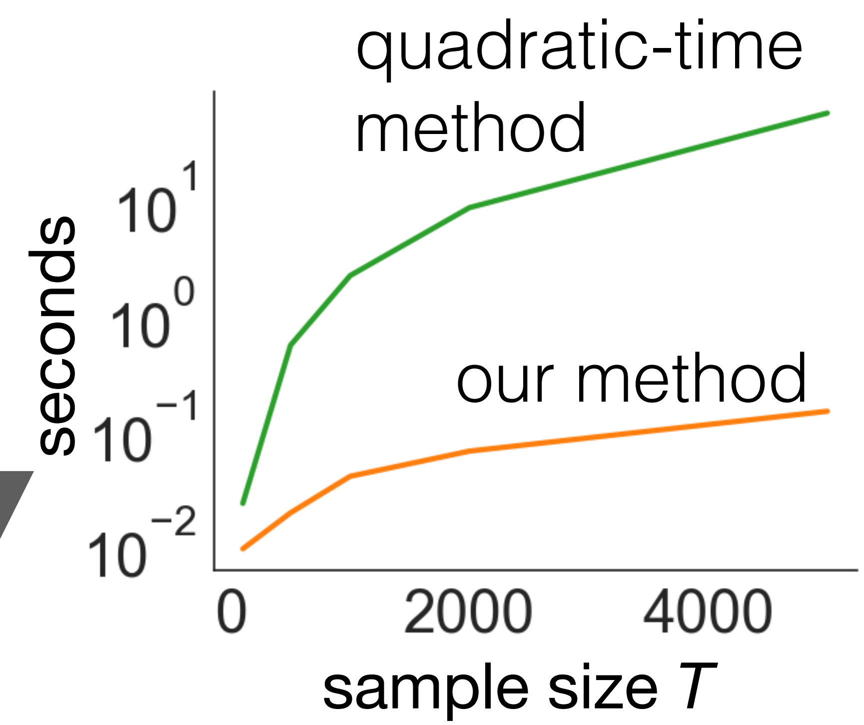
ε = 0.005



ε = 0.05

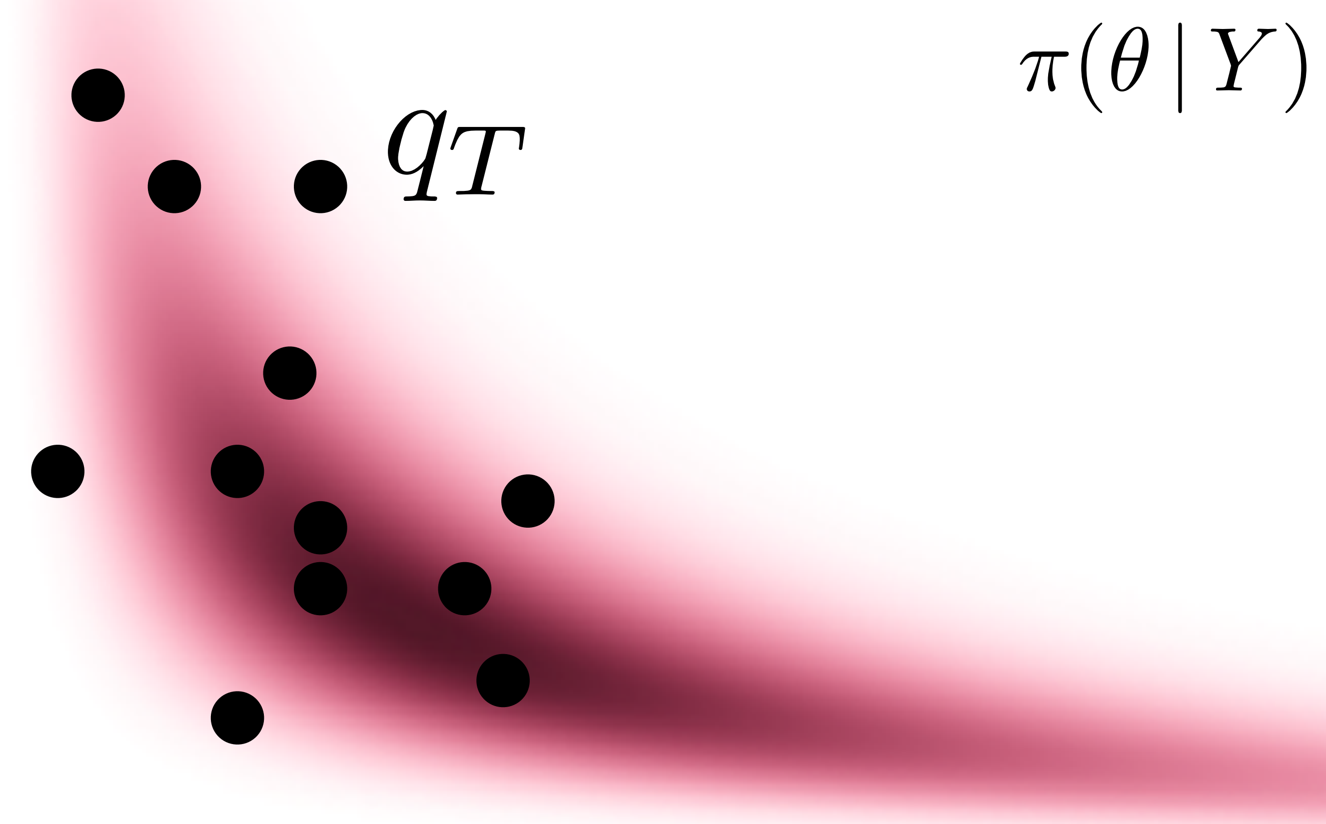


faster



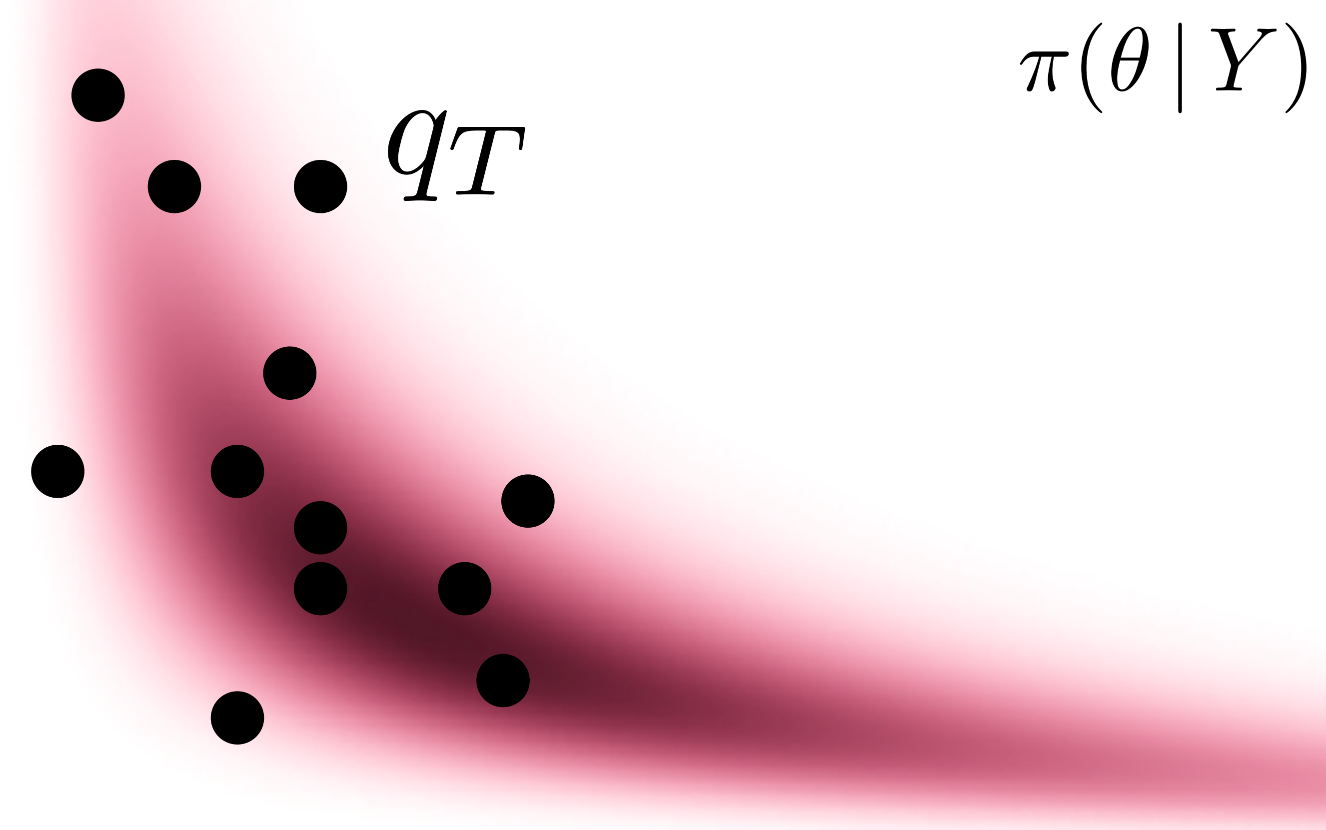
Application #2: goodness-of-fit testing

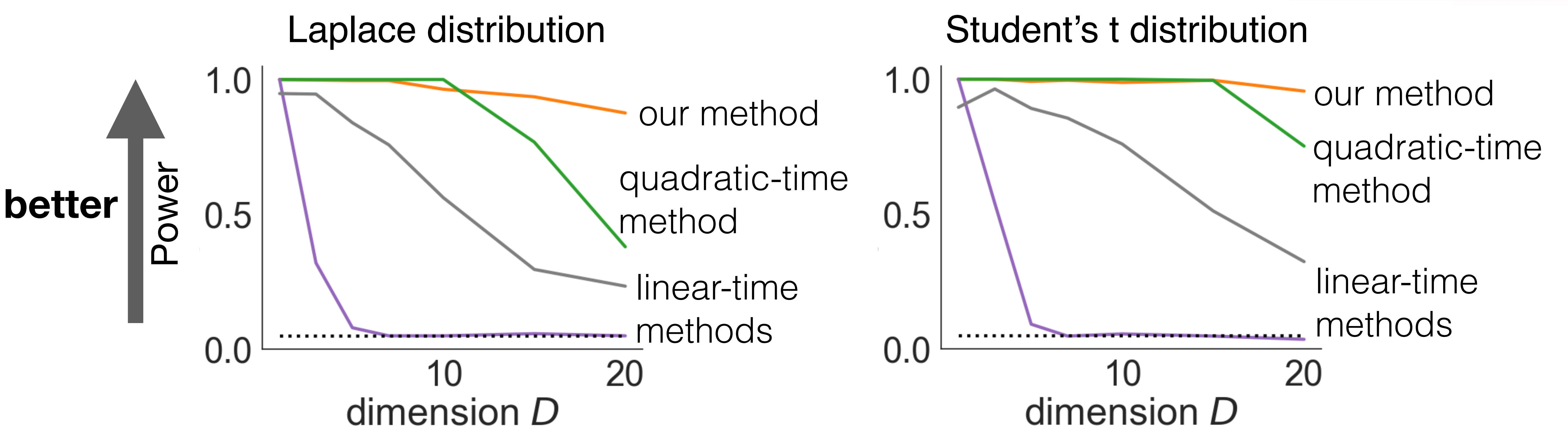
- **Question:** $q_T \approx \pi?$
- Power = probability of correctly rejecting null
- π = standard Gaussian



Application #2: goodness-of-fit testing

- **Question:** $q_T \approx \pi?$
- Power = probability of correctly rejecting null
- π = standard Gaussian





[H & Mackey 2018]

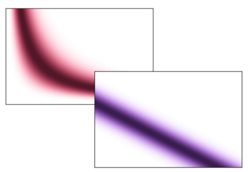
References

Agrawal, Campbell, **Huggins** & Broderick. *Data-dependent compression of random features for large-scale kernel approximation*. AISTATS, 2019.

Huggins* & Mackey*. *Random feature Stein discrepancies*. Neural Information Processing Systems, 2018.

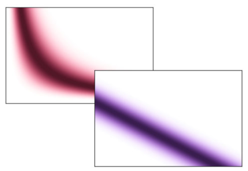
Huggins, Kasprzak, Campbell & Broderick. *Practical bounds on the error of Bayesian posterior approximations: A nonasymptotic approach.* arXiv:1809.09505 [stat.TH], 2018.

Review

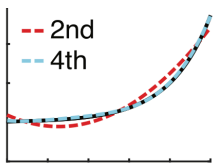


A framework for scalable Bayesian inference

Review



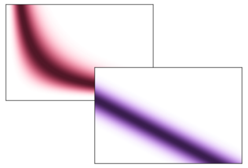
A framework for scalable Bayesian inference



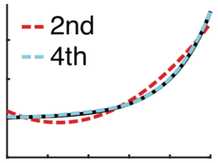
Algorithm design

➡ Polynomial approximate sufficient statistics (PASS) scales to 10 million observations with up to 1000x speed-up and memory reduction

Review



A framework for scalable Bayesian inference



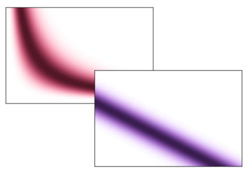
Algorithm design

➡ Polynomial approximate sufficient statistics (PASS) scales to 10 million observations with up to 1000x speed-up and memory reduction

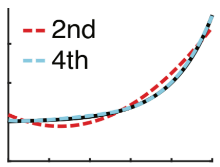
F_η Meaningful accuracy guarantees

➡ General theory for obtaining practical error bounds for likelihood approximations, including PASS

Review



A framework for scalable Bayesian inference

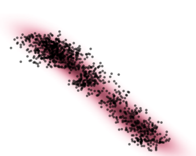


Algorithm design

➡ Polynomial approximate sufficient statistics (PASS) scales to 10 million observations with up to 1000x speed-up and memory reduction

F_η Meaningful accuracy guarantees

➡ General theory for obtaining practical error bounds for likelihood approximations, including PASS

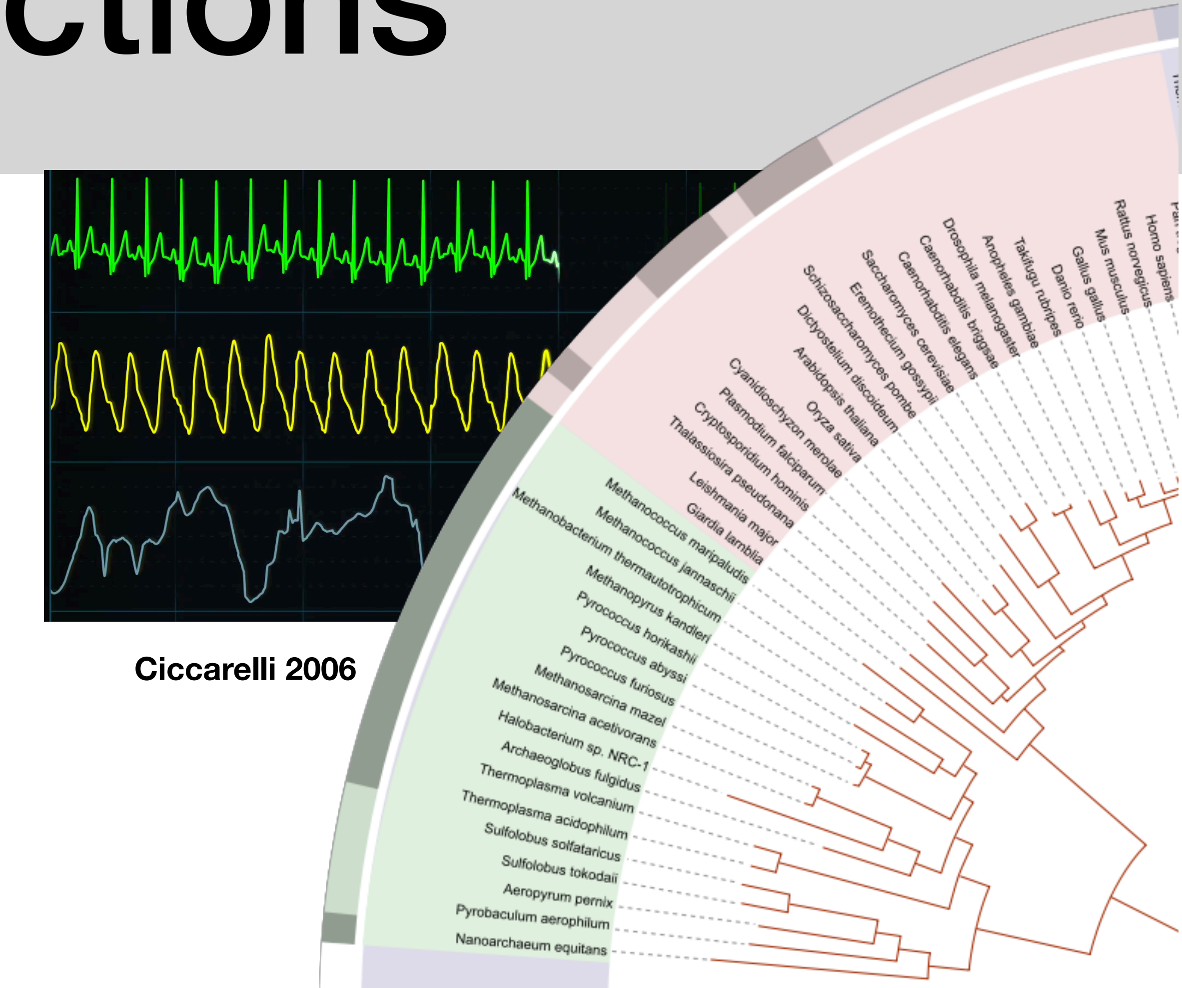


Validating results from heuristic algorithms

➡ Fast (near-linear time) and theoretically sound Stein discrepancy measure

Future Directions

- Scalable inference for time series and other structured data **[e.g. phylogenetic trees]**

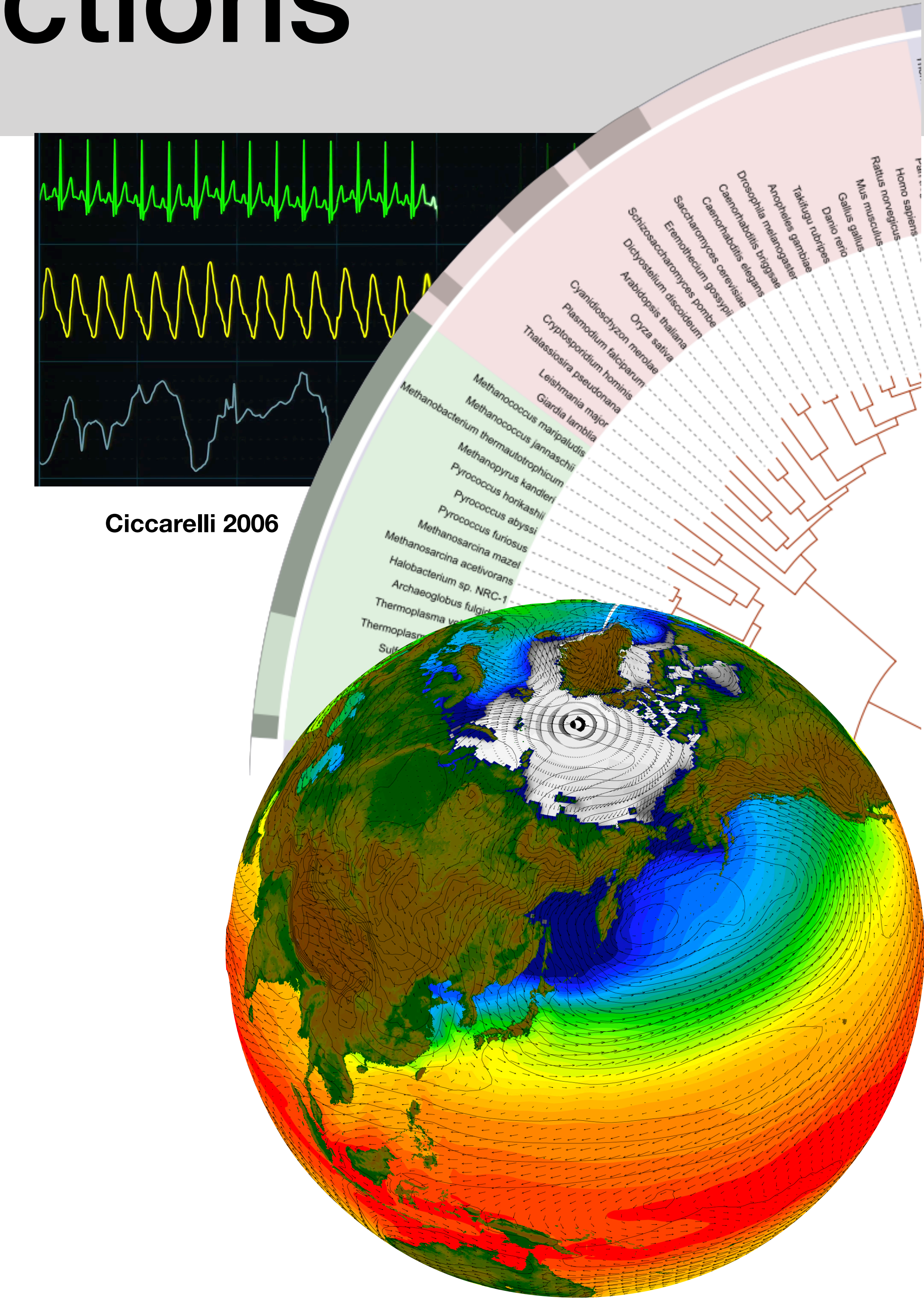


Ciccarelli 2006

Future Directions

- Scalable inference for time series and other structured data **[e.g. phylogenetic trees]**

- Likelihood approximations for PDE-based models **[e.g. climate and other physical systems]**

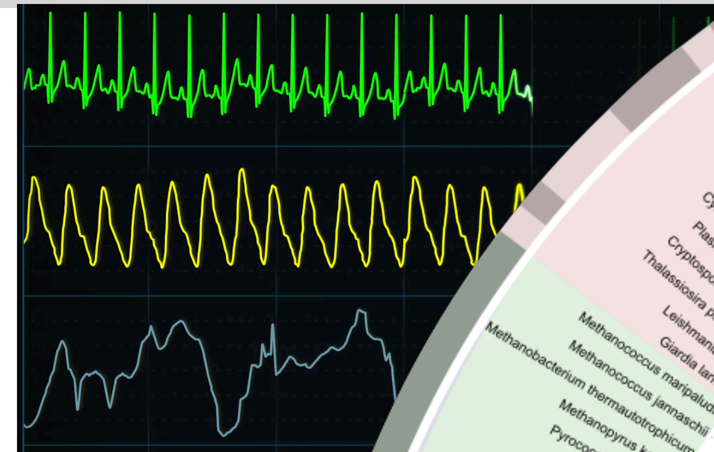


Ciccarelli 2006

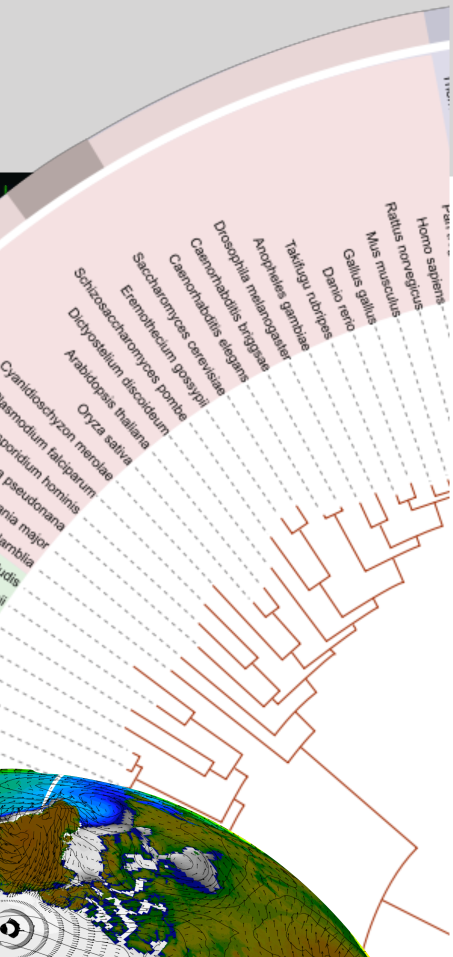
©UCAR, image courtesy Gary Strand, NCAR

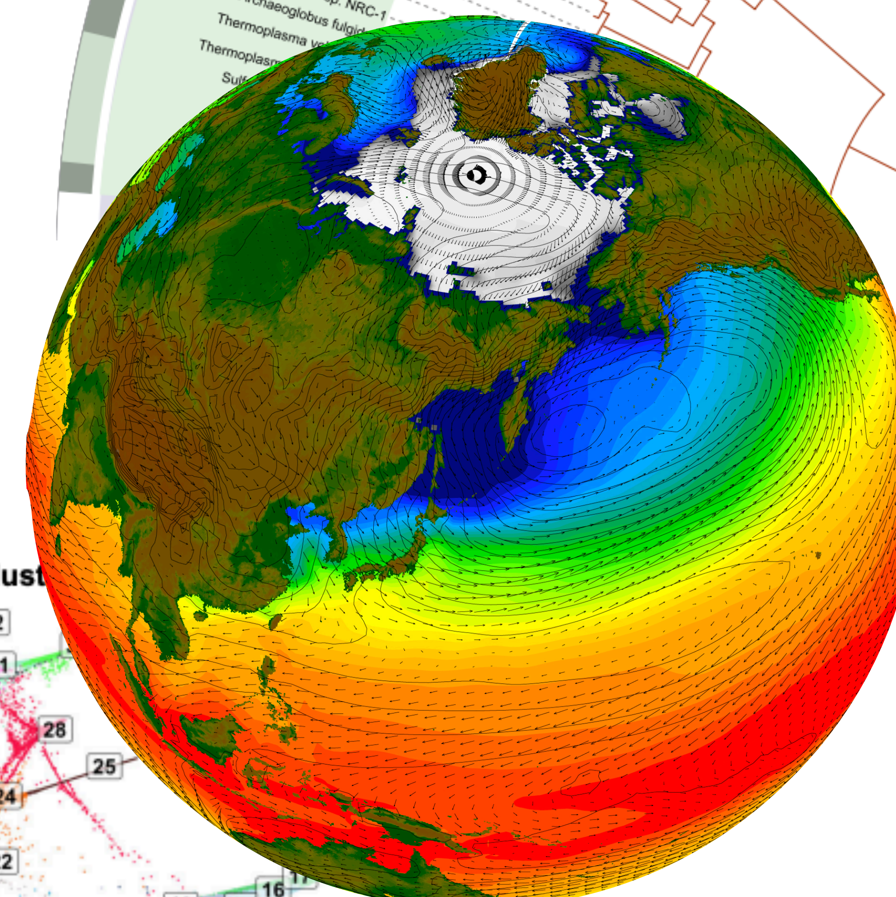
Future Directions

- Scalable inference for time series and other structured data **[e.g. phylogenetic trees]**
- Likelihood approximations for PDE-based models **[e.g. climate and other physical systems]**
- Statistically robust yet scalable inference **[e.g. in cancer genomics]**

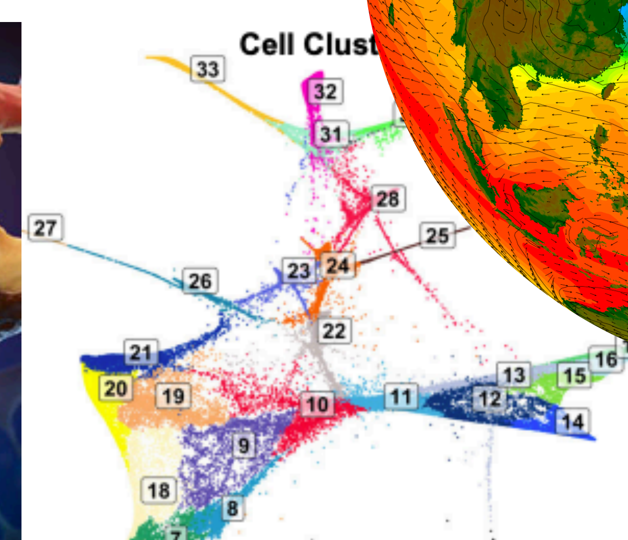


Ciccarelli 2006





©UCAR, image courtesy Gary Strand, NCAR



Schiebinger et al 2018

References

Huggins, Campbell, Kasprzak & Broderick. *Scalable Gaussian process inference with finite-data mean and variance guarantees*. AISTATS, 2019.

Agrawal, Campbell, **Huggins** & Broderick. *Data-dependent compression of random features for large-scale kernel approximation*. AISTATS, 2019.

Huggins* & Mackey*. *Random feature Stein discrepancies*. Neural Information Processing Systems, 2018.

Huggins, Kasprzak, Campbell & Broderick. *Practical bounds on the error of Bayesian posterior approximations: A nonasymptotic approach*. arXiv:1809.09505 [stat.TH], 2018.

Huggins, Adams & Broderick. *PASS-GLM: polynomial approximate sufficient statistics for scalable Bayesian GLM inference*. Neural Information Processing Systems, 2017.

Huggins* & Zou*. *Quantifying the accuracy of approximate diffusions and Markov chains*. AISTATS, 2017.

Huggins, Campbell & Broderick. *Coresets for scalable Bayesian logistic regression*. Neural Information Processing Systems, 2016.

Theory and practice for MCMC and numerical optimization

Theory and practice for MCMC and numerical optimization

	Markov chain Monte Carlo	Optimization

Theory and practice for MCMC and numerical optimization

	Markov chain Monte Carlo	Optimization
Theory applies to "simple" cases	e.g. strongly log-concave density	e.g. convex functions

Theory and practice for MCMC and numerical optimization

	Markov chain Monte Carlo	Optimization
Theory applies to “simple” cases	e.g. strongly log-concave density	e.g. convex functions
Theory provides loose quantitative rates	e.g. geometric ergodicity	e.g. linear or 1/T

Theory and practice for MCMC and numerical optimization

	Markov chain Monte Carlo	Optimization
Theory applies to “simple” cases	e.g. strongly log-concave density	e.g. convex functions
Theory provides loose quantitative rates	e.g. geometric ergodicity	e.g. linear or 1/T
Theory does not apply to many real-world cases	e.g. non-trivial hierarchical models	e.g. non-convex functions

Theory and practice for MCMC and numerical optimization

	Markov chain Monte Carlo	Optimization
Theory applies to “simple” cases	e.g. strongly log-concave density	e.g. convex functions
Theory provides loose quantitative rates	e.g. geometric ergodicity	e.g. linear or 1/T
Theory does not apply to many real-world cases	e.g. non-trivial hierarchical models	e.g. non-convex functions
Practical methods to evaluate success	e.g. Gelman–Rubin diagnostic	e.g. norm of gradient, duality gap

Theory and practice for MCMC and numerical optimization

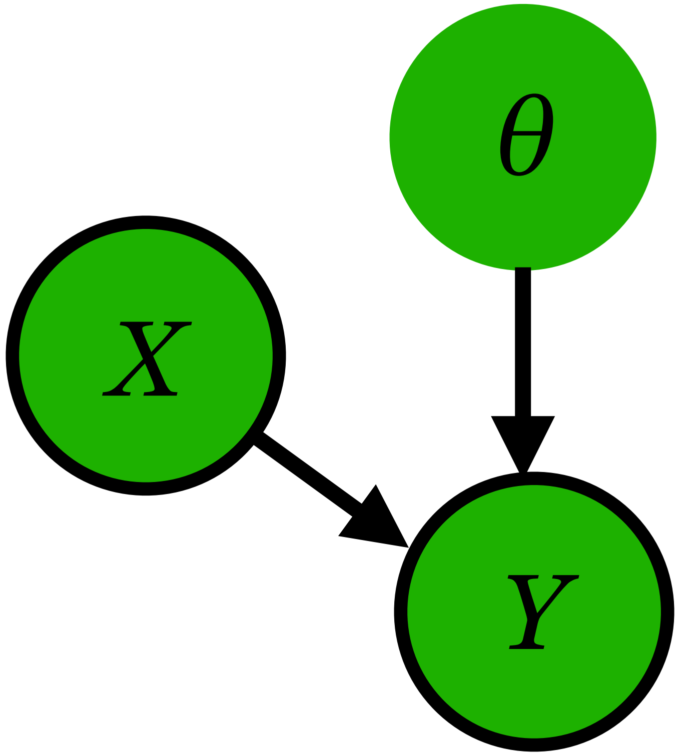
	Markov chain Monte Carlo	Optimization
Theory applies to "simple" cases	e.g. strongly log-concave density	e.g. convex functions
Theory provides loose quantitative rates	e.g. geometric ergodicity	e.g. linear or 1/T
Theory does not apply to many real-world cases	e.g. non-trivial hierarchical models	e.g. non-convex functions
Practical methods to evaluate success	e.g. Gelman–Rubin diagnostic	e.g. norm of gradient, duality gap
Comparison of algorithms	e.g. effective samples per second	e.g. time to convergence

Theory and practice for MCMC and numerical optimization

	Markov chain Monte Carlo	Optimization	Scalable Bayes
Theory applies to "simple" cases	e.g. strongly log-concave density	e.g. convex functions	e.g. strongly log-concave density
Theory provides loose quantitative rates	e.g. geometric ergodicity	e.g. linear or 1/T	e.g. exponentially small error
Theory does not apply to many real-world cases	e.g. non-trivial hierarchical models	e.g. non-convex functions	e.g. non-trivial hierarchical models
Practical methods to evaluate success	e.g. Gelman–Rubin diagnostic	e.g. norm of gradient, duality gap	e.g. Stein discrepancies
Comparison of algorithms	e.g. effective samples per second	e.g. time to convergence	e.g. Stein discrepancies

PASS for generalized linear models (PASS-GLM)

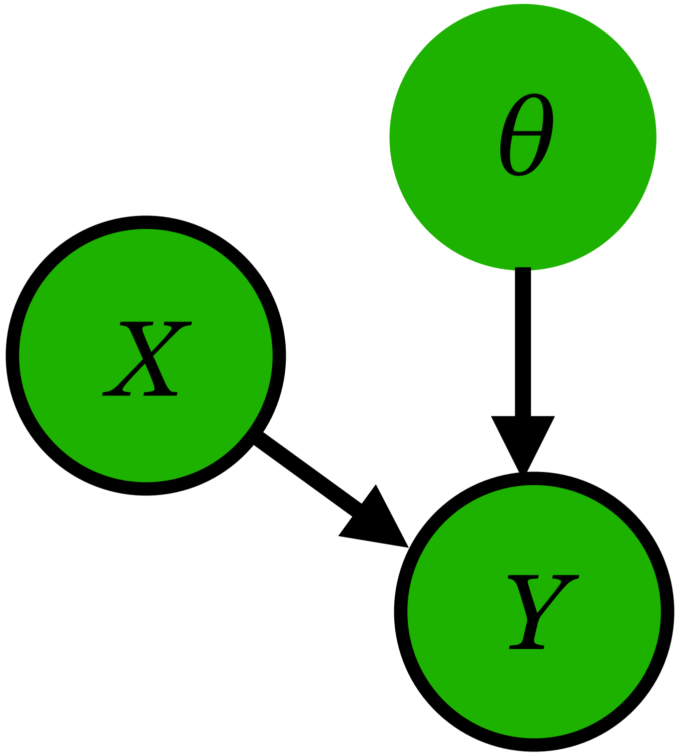
$Y = \{y_1, y_2, \ldots, y_N\}$, $y_n \in \mathbb{R}$, $X = \{x_1, x_2, \ldots, x_N\}$, $x_n \in \mathbb{R}^d$, $\theta \in \mathbb{R}^d$



PASS for generalized linear models (PASS-GLM)

$$Y = \{y_1, y_2, \ldots, y_N\},\ y_n \in \mathbb{R},\ X = \{x_1, x_2, \ldots, x_N\},\ x_n \in \mathbb{R}^d,\ \theta \in \mathbb{R}^d$$

GLMs: $\log p(y_n \,|\, x_n, \theta) = \phi(y_n, \theta \cdot x_n) \approx \eta(\theta) \cdot \tau(y_n, x_n)$

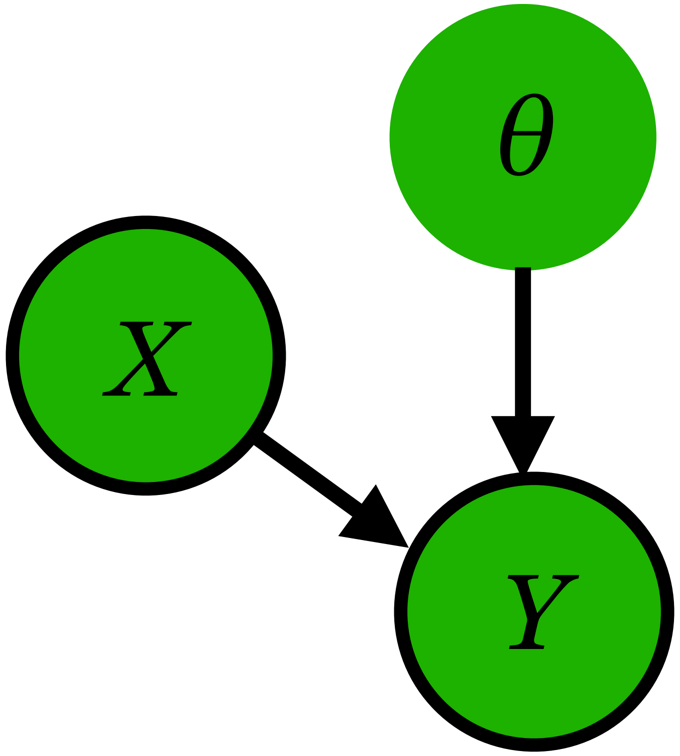


PASS for generalized linear models (PASS-GLM)

$$Y = \{y_1, y_2, \ldots, y_N\},\ y_n \in \mathbb{R},\ X = \{x_1, x_2, \ldots, x_N\},\ x_n \in \mathbb{R}^d,\ \theta \in \mathbb{R}^d$$

GLMs: $\log p(y_n \,|\, x_n, \theta) = \phi(y_n, \theta \cdot x_n) \approx \eta(\theta) \cdot \tau(y_n, x_n)$

$$\begin{aligned}\tau(y_n, x_n) = (&y_n, x_{n1}, x_{n2}, \ldots, x_{nd},\\ &y_n^2, x_{n1}^2, x_{n2}^2, \ldots, x_{nd}^2,\\ &y_n x_{n1}, \ldots, y_n x_{nd},\\ &x_{n1}x_{n2}, x_{n1}x_{n3}, \ldots,\\ &\ldots,\\ &y_n^M, x_{n1}^M, x_{n2}^M, \ldots, x_{nd}^M)\end{aligned}$$



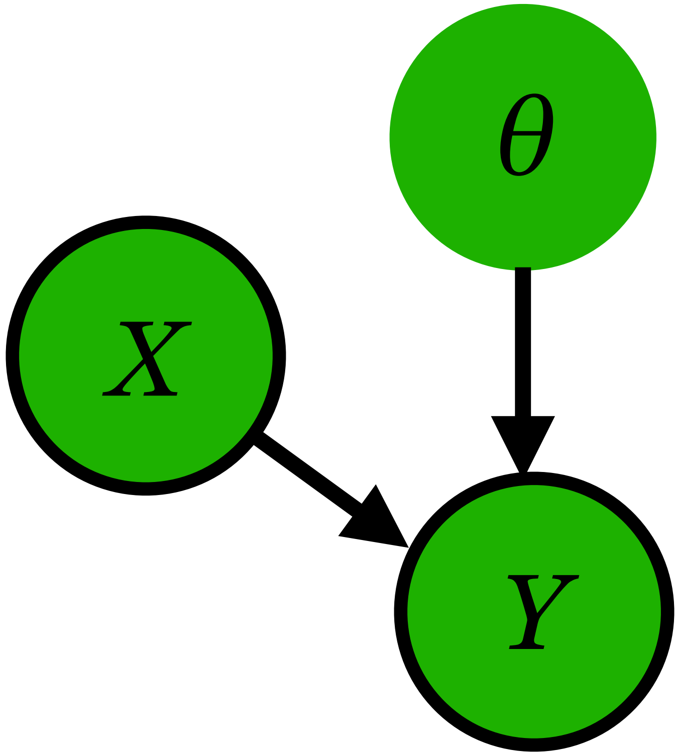
PASS for generalized linear models (PASS-GLM)

$$Y = \{y_1, y_2, \ldots, y_N\},\ y_n \in \mathbb{R},\ X = \{x_1, x_2, \ldots, x_N\},\ x_n \in \mathbb{R}^d,\ \theta \in \mathbb{R}^d$$

GLMs: $\log p(y_n \,|\, x_n, \theta) = \phi(y_n, \theta \cdot x_n) \approx \eta(\theta) \cdot \tau(y_n, x_n)$

$$\begin{aligned}\tau(y_n, x_n) = (&y_n, x_{n1}, x_{n2}, \ldots, x_{nd},\\ &y_n^2, x_{n1}^2, x_{n2}^2, \ldots, x_{nd}^2,\\ &y_n x_{n1}, \ldots, y_n x_{nd},\\ &x_{n1}x_{n2}, x_{n1}x_{n3}, \ldots,\\ &\ldots,\\ &y_n^M, x_{n1}^M, x_{n2}^M, \ldots, x_{nd}^M)\end{aligned}$$

$$\begin{aligned}L &= \dim(\tau)\\ &= \binom{m+d+1}{m}\\ &= O([d+1]^m)\end{aligned}$$



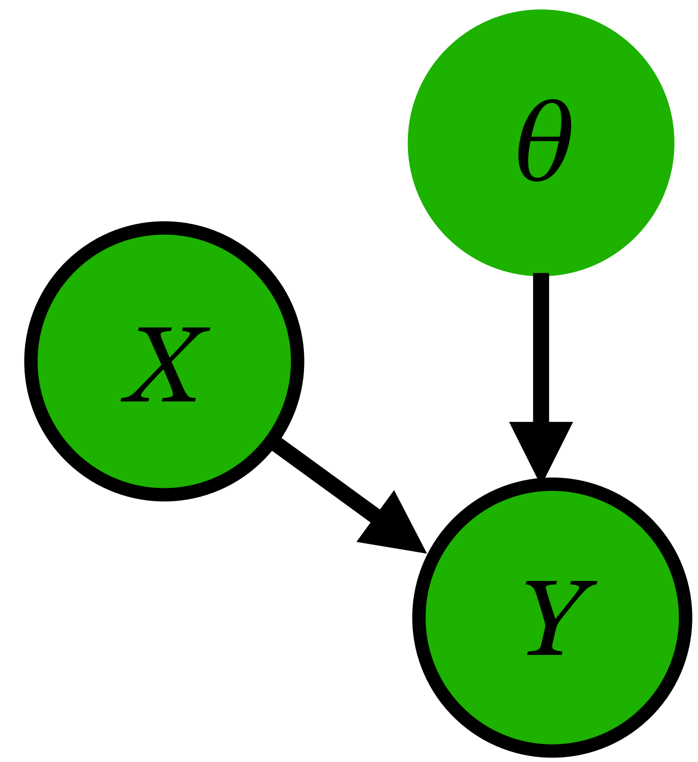
PASS for generalized linear models (PASS-GLM)

$$Y = \{y_1, y_2, \ldots, y_N\},\ y_n \in \mathbb{R},\ X = \{x_1, x_2, \ldots, x_N\},\ x_n \in \mathbb{R}^d,\ \theta \in \mathbb{R}^d$$

GLMs: $\log p(y_n \,|\, x_n, \theta) = \phi(y_n, \theta \cdot x_n) \approx \eta(\theta) \cdot \tau(y_n, x_n)$

$$\begin{aligned}\tau(y_n, x_n) = (&y_n, x_{n1}, x_{n2}, \ldots, x_{nd},\\ &y_n^2, x_{n1}^2, x_{n2}^2, \ldots, x_{nd}^2,\\ &y_n x_{n1}, \ldots, y_n x_{nd},\\ &x_{n1}x_{n2}, x_{n1}x_{n3}, \ldots,\\ &\ldots,\\ &y_n^M, x_{n1}^M, x_{n2}^M, \ldots, x_{nd}^M)\end{aligned}$$

$$\begin{aligned}L &= \dim(\tau)\\ &= \binom{m+d+1}{m}\\ &= O([d+1]^m)\end{aligned}$$



$$\tau(y_n, x_n) = \left(a(k, M) y_n^{k_0} \textstyle\prod_{i=1}^d x_{ni}^{k_i}\right)_{\substack{k \in \mathbb{N}^{d+1} \\ \sum_i k \leq M}} \qquad \eta(\theta) = \left(\textstyle\prod_{i=1}^d \theta_i^{k_i}\right)_{\substack{k \in \mathbb{N}^{d+1} \\ \sum_i k \leq M}}$$