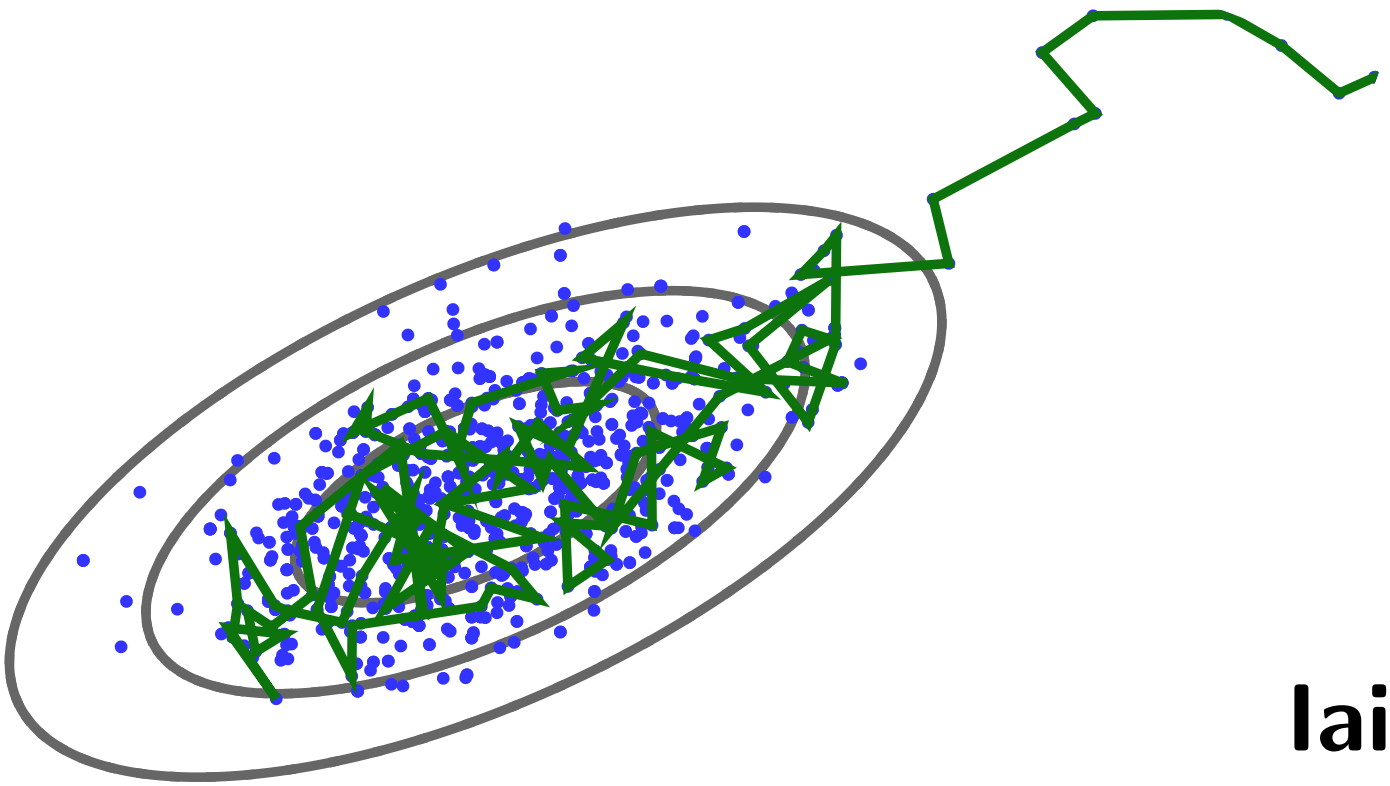
Monte Carlo and Machine Learning



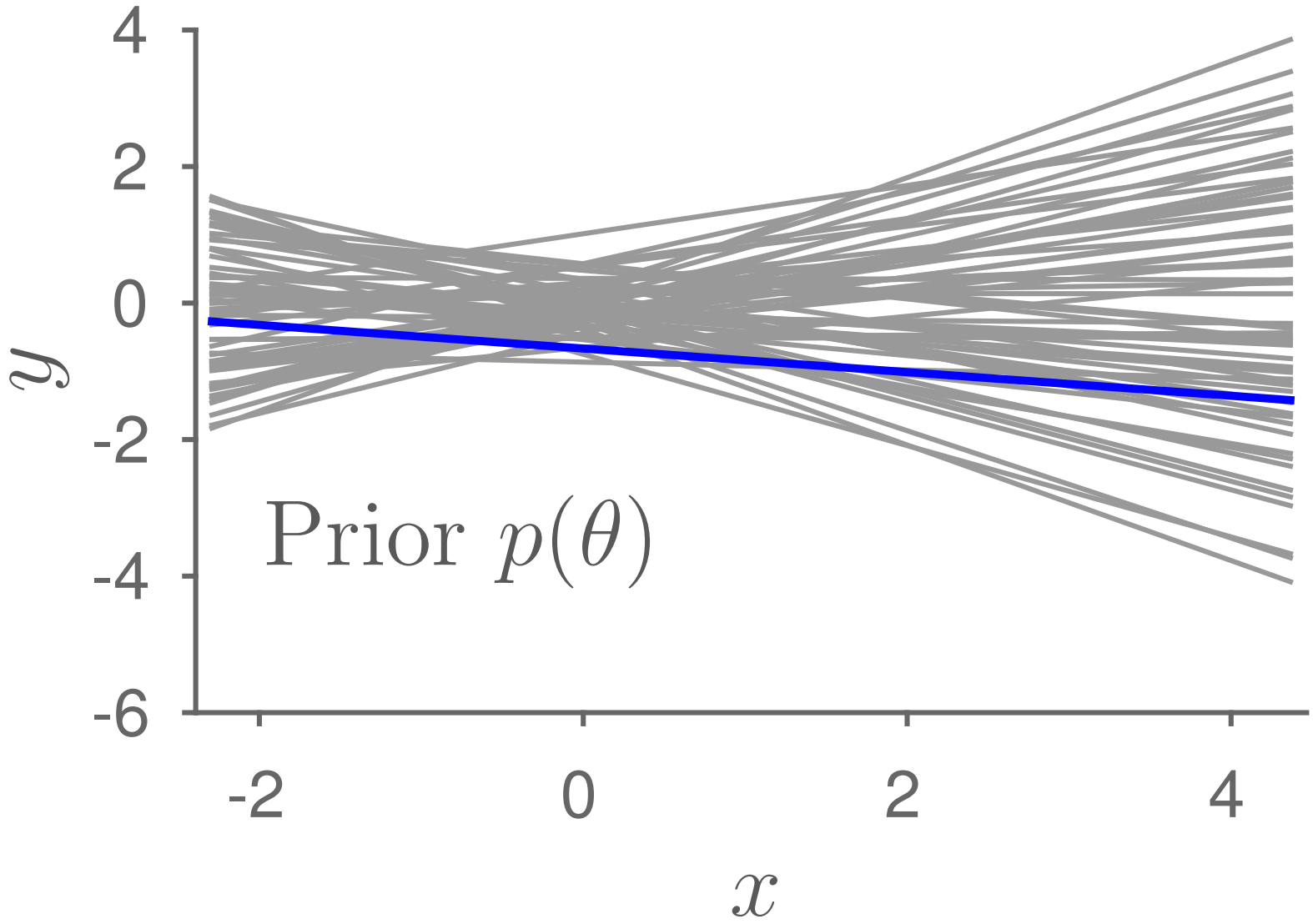
Iain Murray

University of Edinburgh

http://iainmurray.net

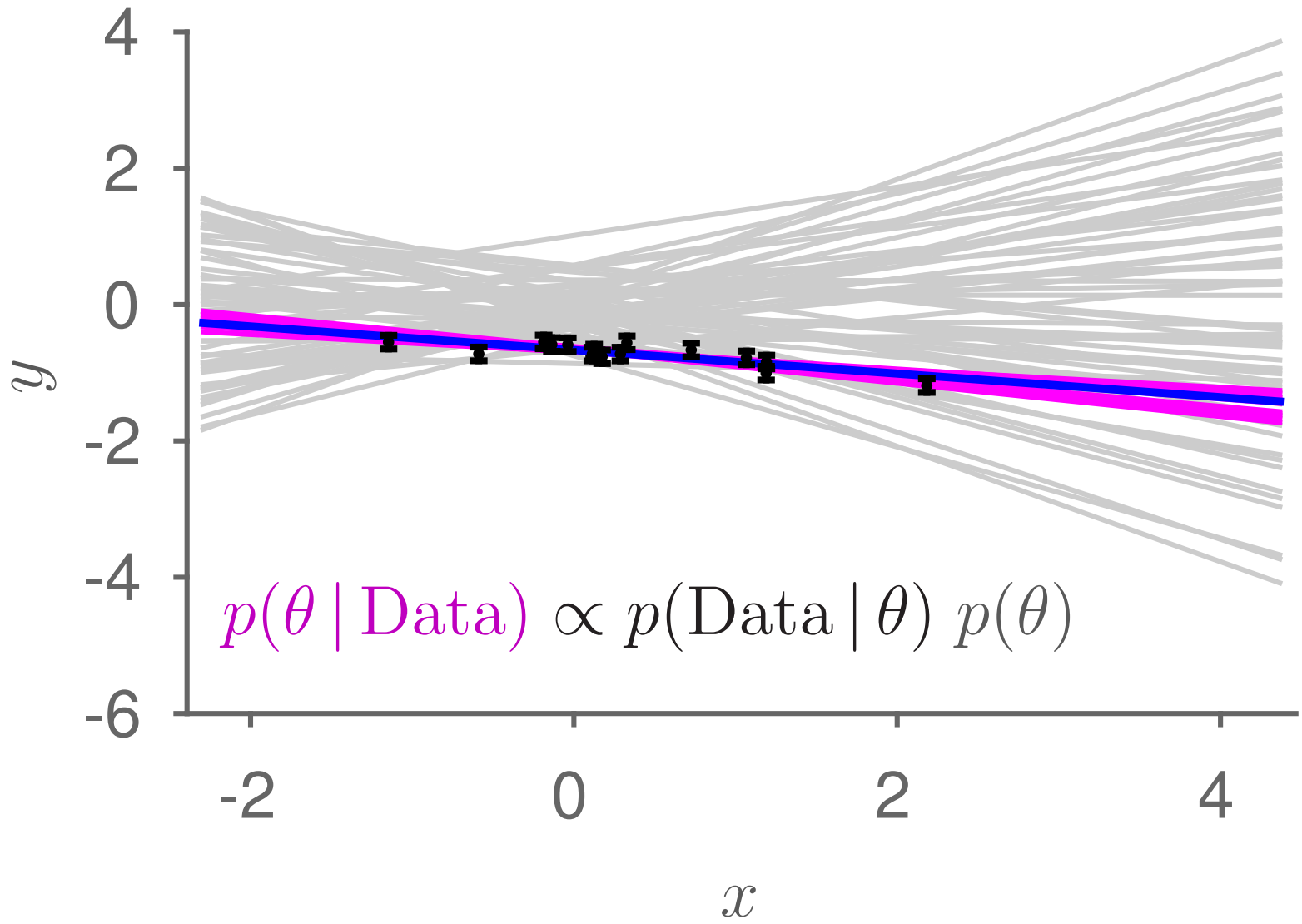
Linear regression

$y = \theta_1 x + \theta_2, \qquad p(\theta) = \mathcal{N}(\theta;\, 0, 0.4^2 I)$

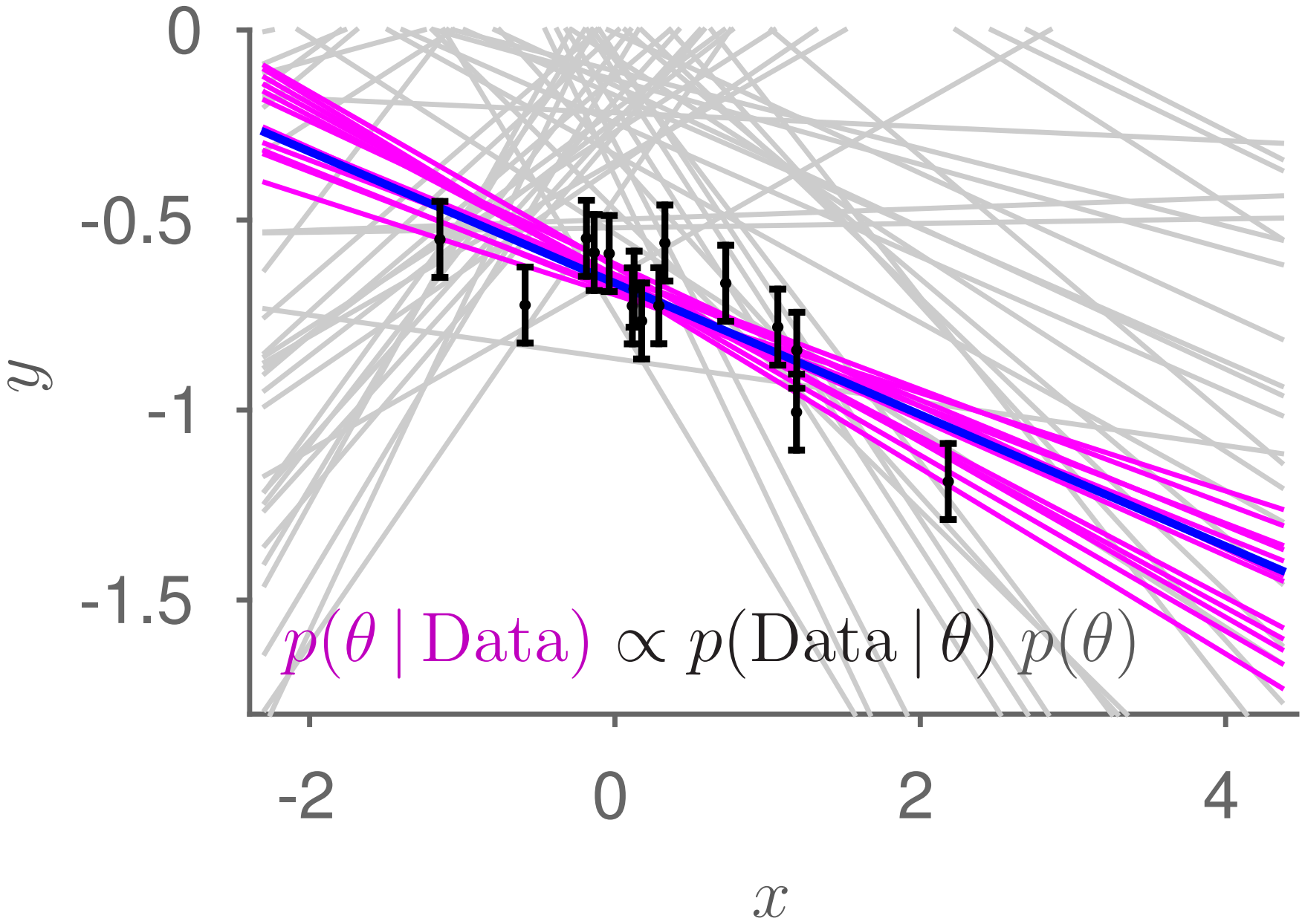


Linear regression

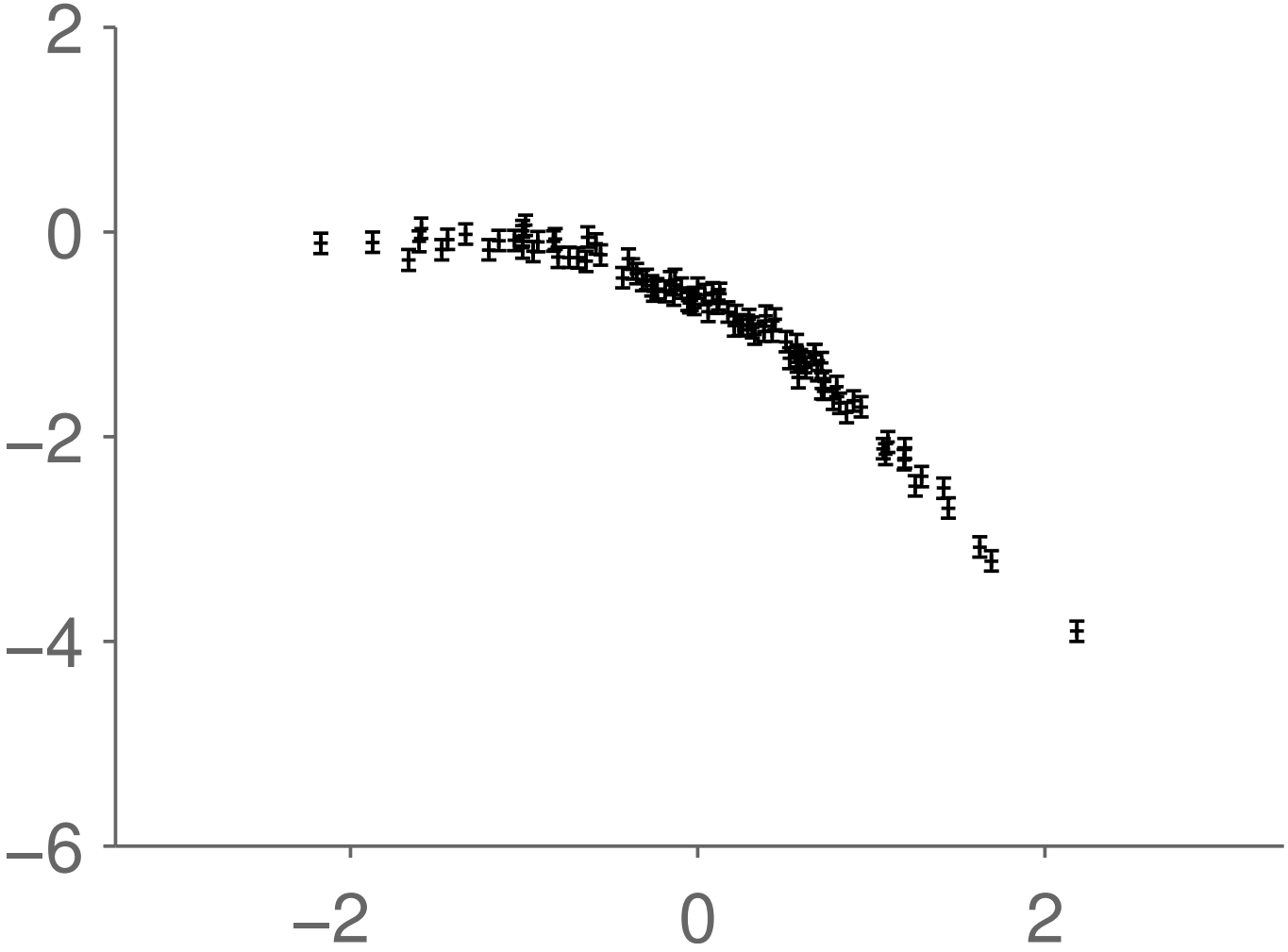
$$y^{(n)} = \theta_1 x^{(n)} + \theta_2 + \epsilon^{(n)}, \qquad \epsilon^{(n)} \sim \mathcal{N}(0, 0.1^2)$$



Linear regression (zoomed in)



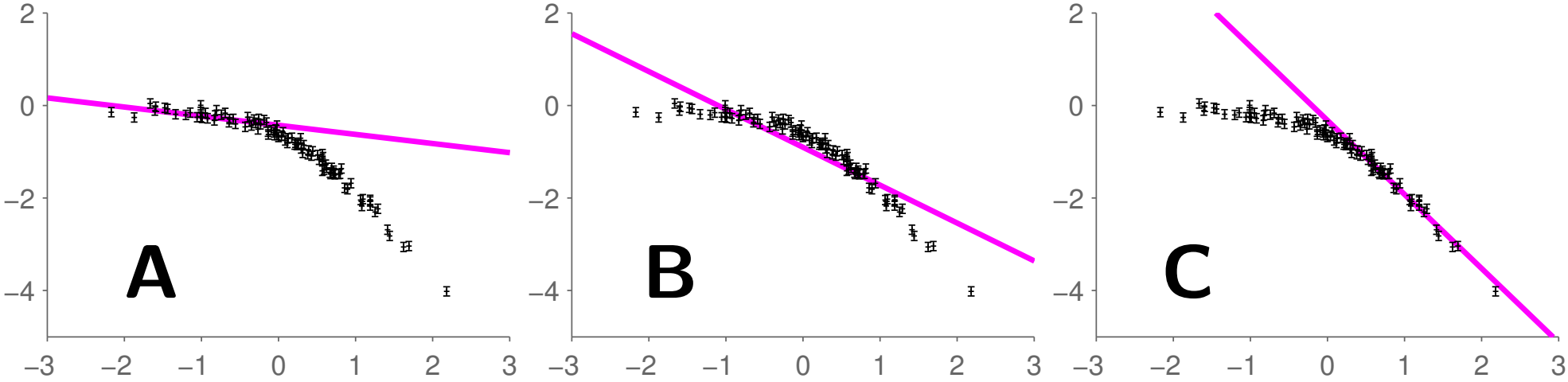
Model mismatch



What will Bayesian linear regression do?

Quiz

Given a (wrong) linear assumption, which explanations are typical of the posterior distribution?

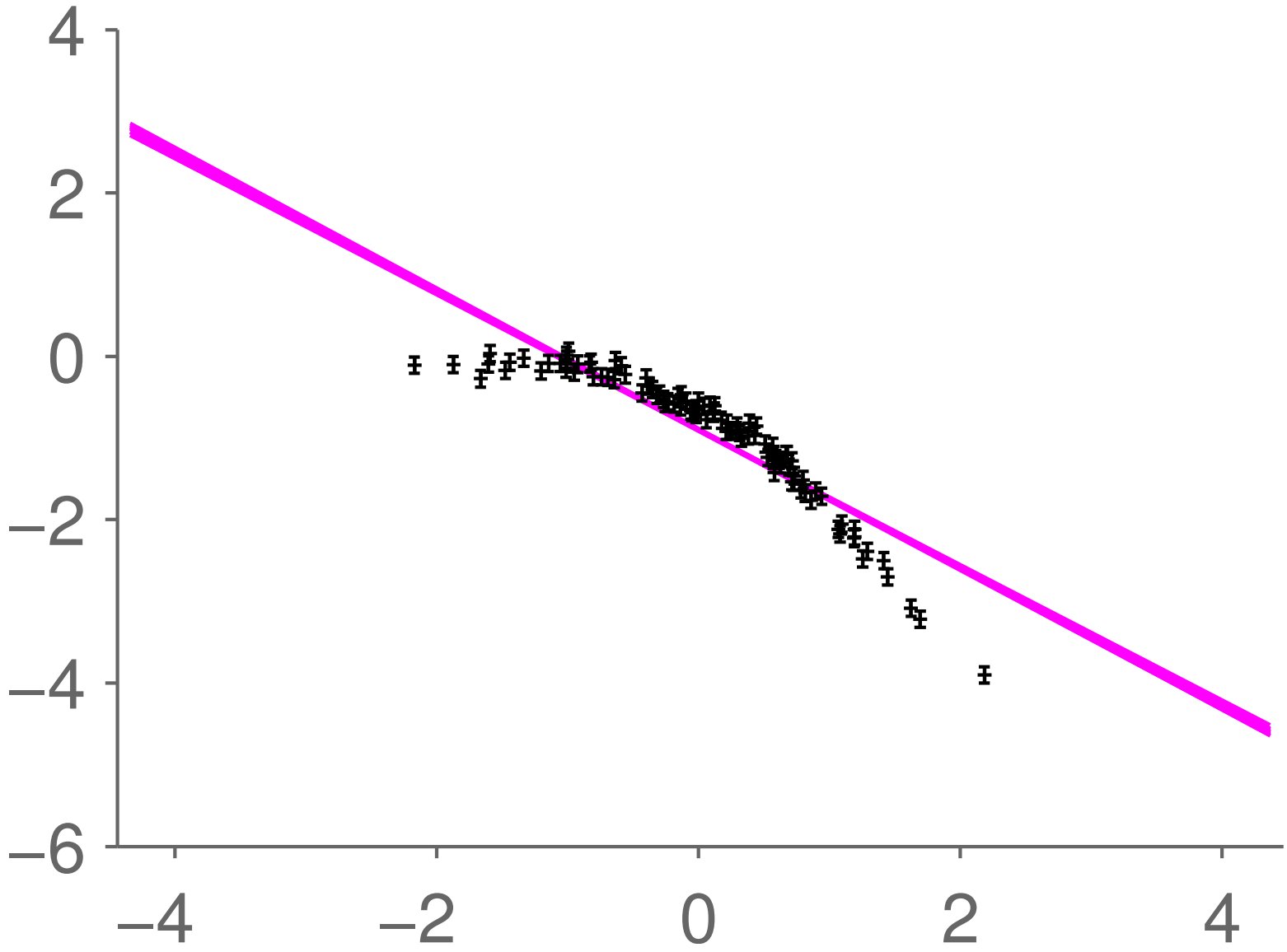


D All of the above

E None of the above

Z Not sure

‘Underfitting’



Posterior *very* certain despite blatant misfit. Peaked around least bad option.

Model checking

arXiv:1705.07057 — Masked Autoregressive Flow

Introduction: Gelman et al., *Bayesian Data Analysis*

Roadmap

— Looking at samples

— **Monte Carlo computations**

— Scaling to large datasets

Simple Monte Carlo Integration

$$\int f(\theta)\,\pi(\theta)\,\mathrm{d}\theta = \text{“average over } \pi \text{ of } f\text{”}$$

$$\approx \frac{1}{S}\sum_{s=1}^{S} f(\theta^{(s)}), \quad \theta^{(s)} \sim \pi$$

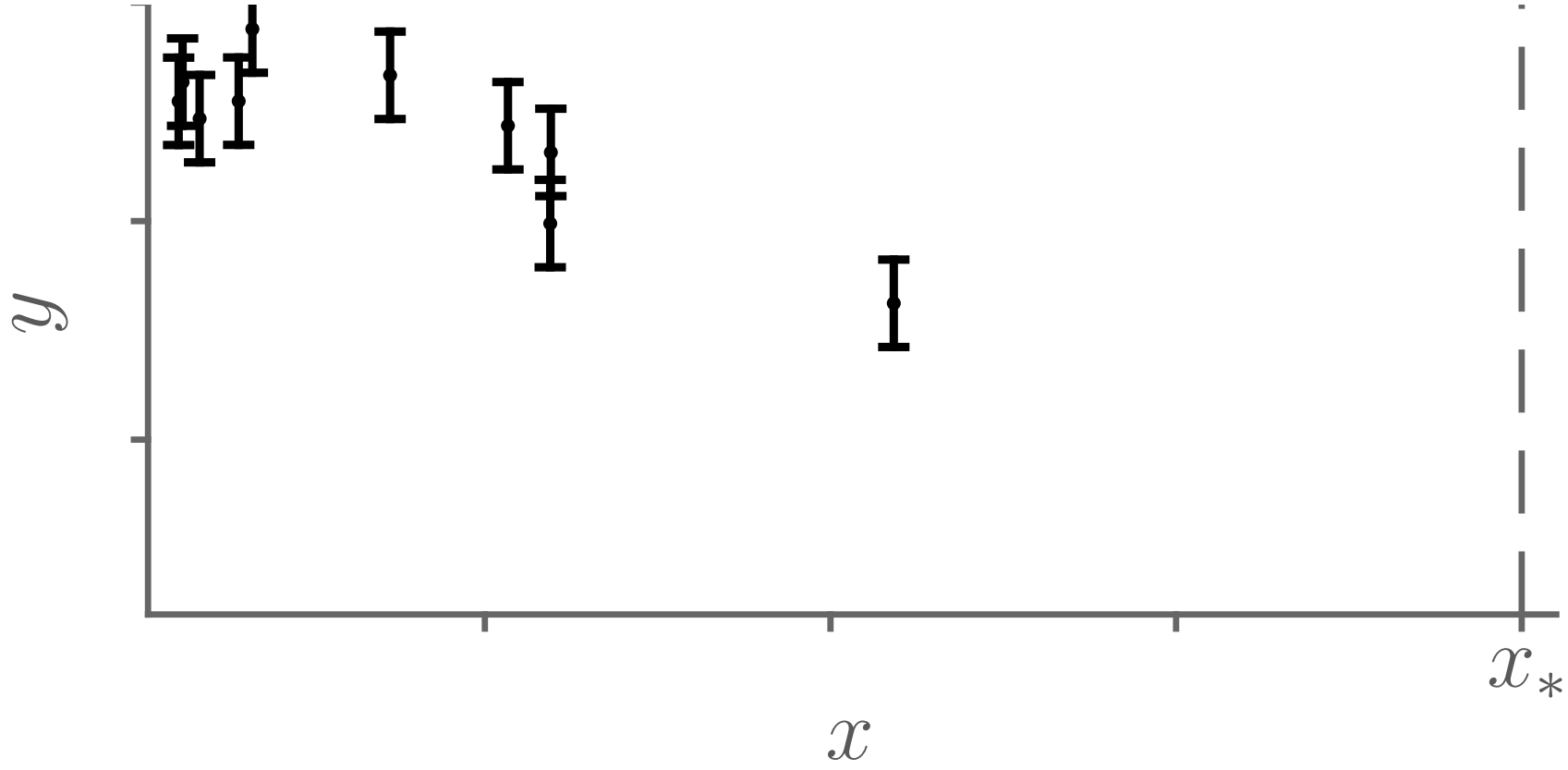
Unbiased

Variance $\sim 1/S$

Prediction

$$p(y_* \mid x_*, \mathcal{D}) = \int p(y_* \mid x_*, \theta)\, p(\theta \mid \mathcal{D})\, \mathrm{d}\theta$$

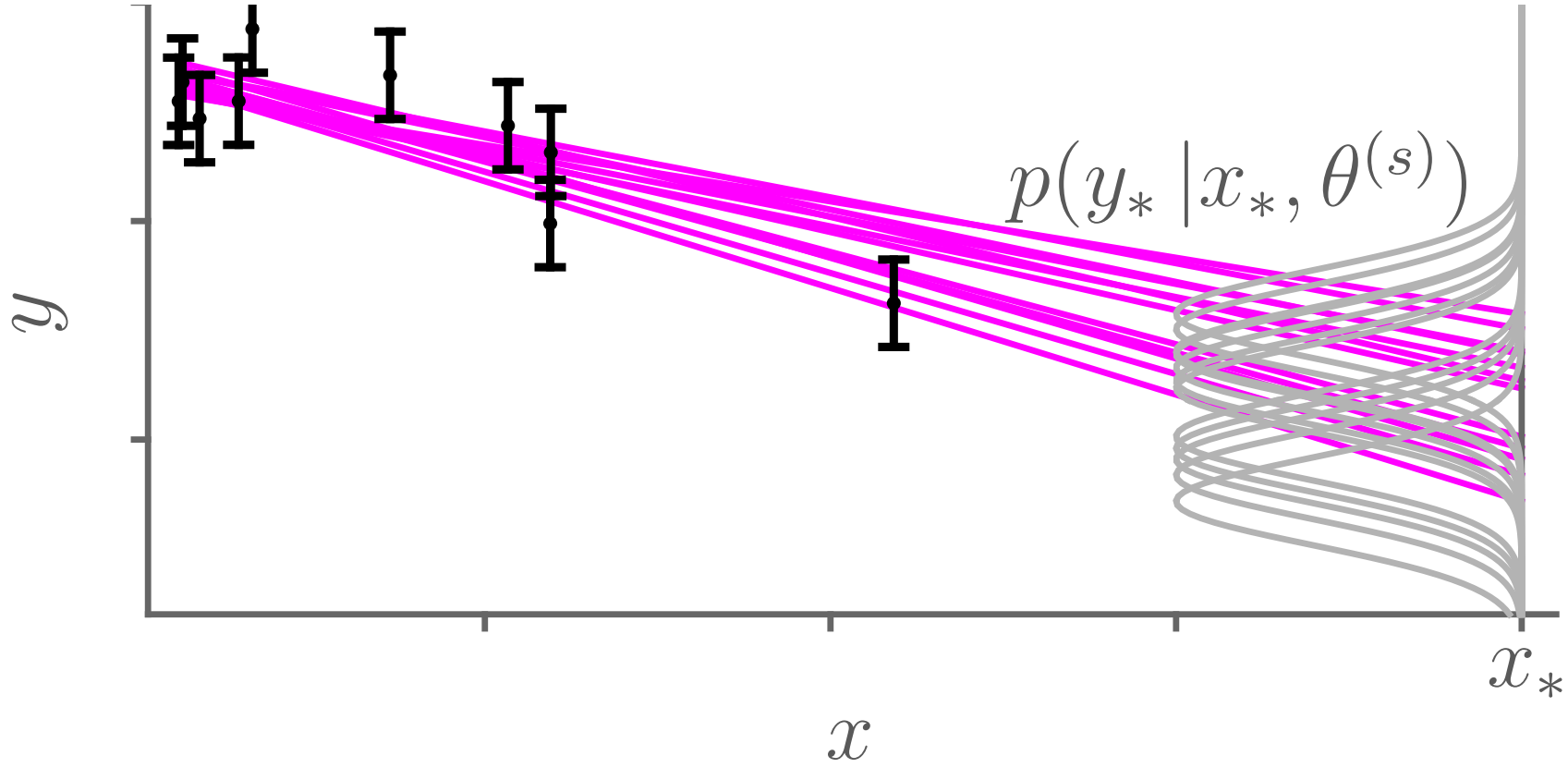
$$\approx \frac{1}{S} \sum_s p(y_* \mid x_*, \theta^{(s)}), \quad \theta^{(s)} \sim p(\theta \mid \mathcal{D})$$



Prediction

$$p(y_* \mid x_*, \mathcal{D}) = \int p(y_* \mid x_*, \theta)\, p(\theta \mid \mathcal{D})\, \mathrm{d}\theta$$

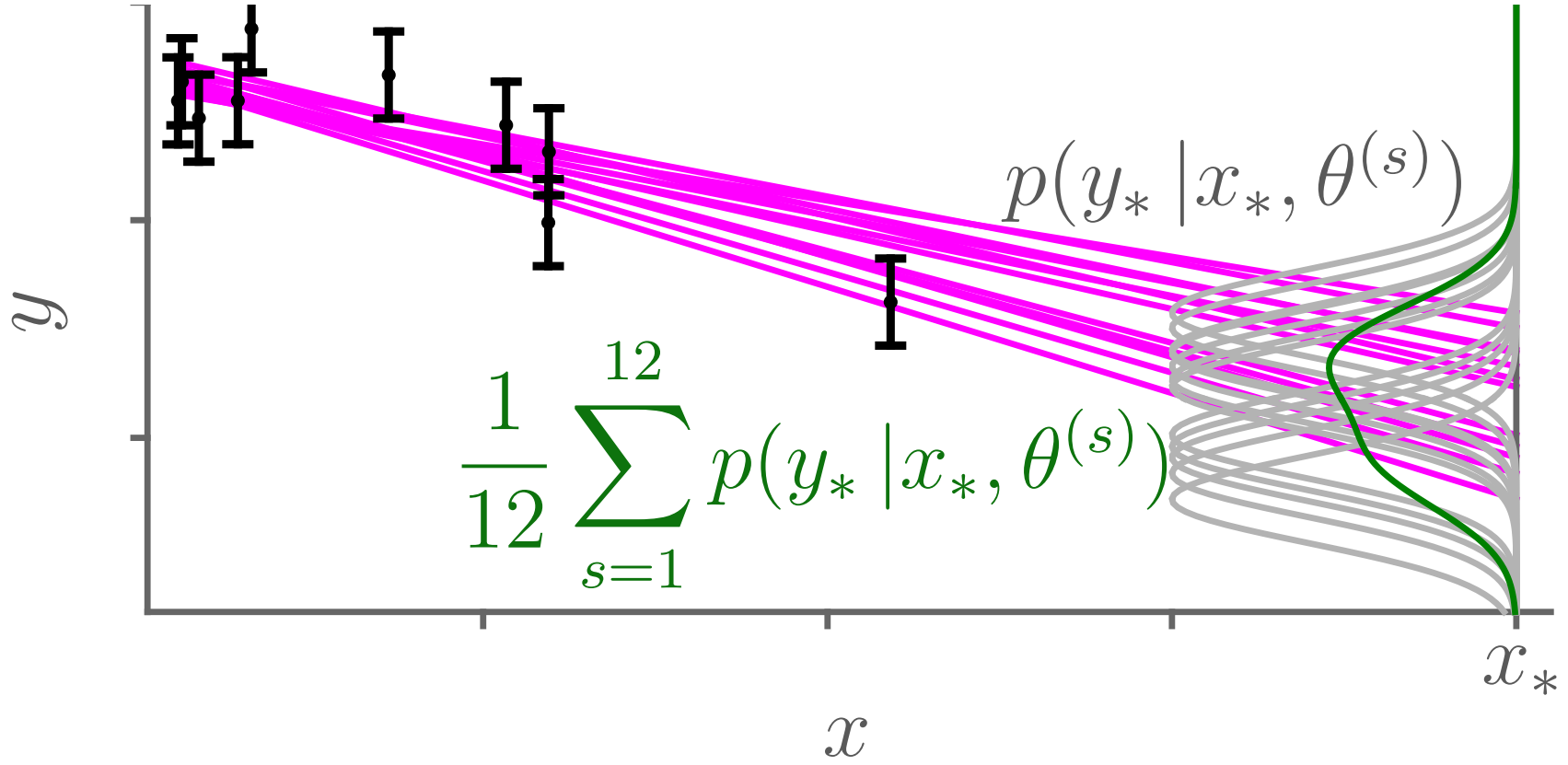
$$\approx \frac{1}{S} \sum_s p(y_* \mid x_*, \theta^{(s)}), \quad \theta^{(s)} \sim p(\theta \mid \mathcal{D})$$



Prediction

$$p(y_* \mid x_*, \mathcal{D}) = \int p(y_* \mid x_*, \theta)\, p(\theta \mid \mathcal{D})\, \mathrm{d}\theta$$

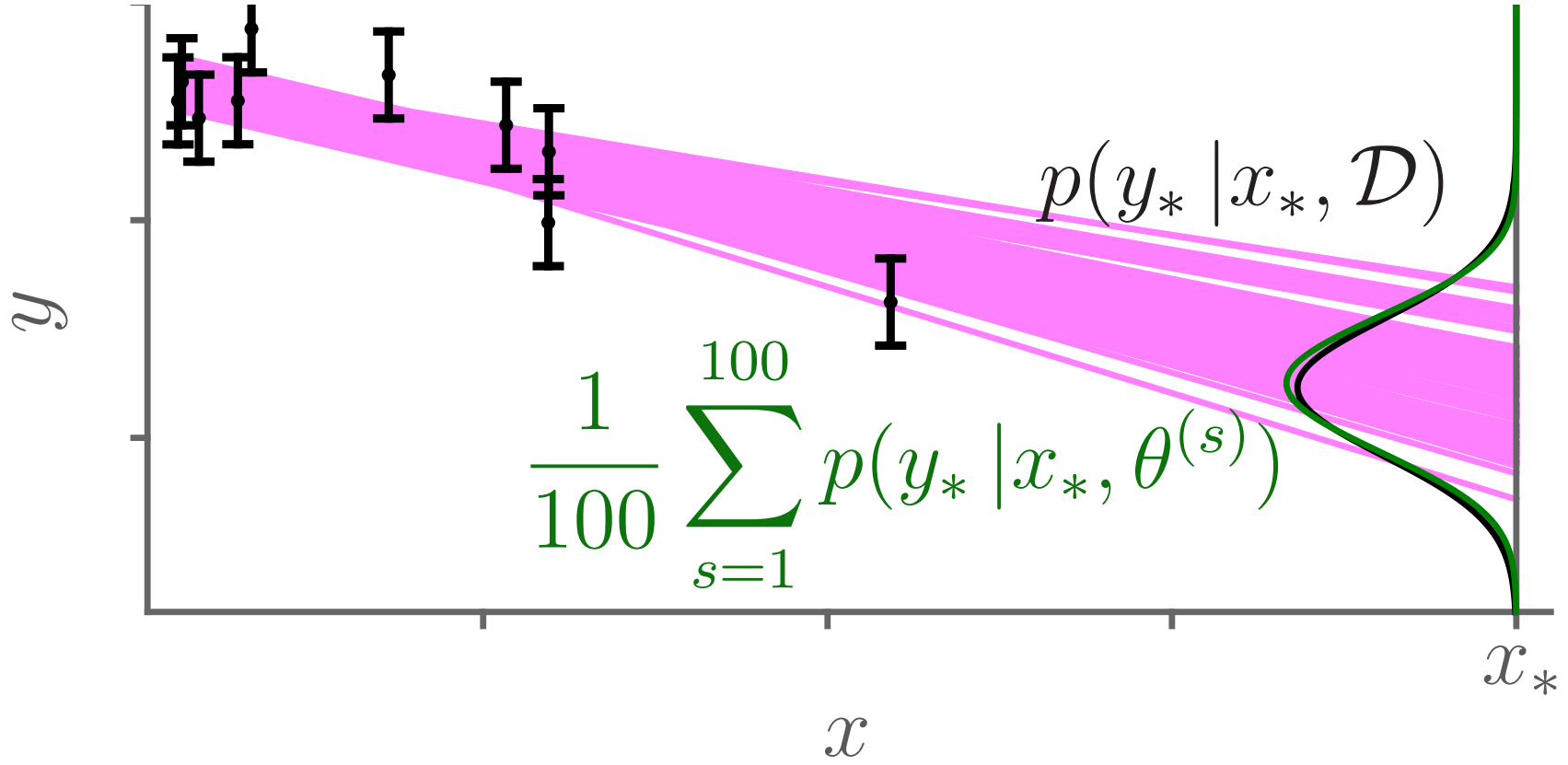
$$\approx \frac{1}{S} \sum_s p(y_* \mid x_*, \theta^{(s)}), \quad \theta^{(s)} \sim p(\theta \mid \mathcal{D})$$



Prediction

$$p(y_* \mid x_*, \mathcal{D}) = \int p(y_* \mid x_*, \theta)\, p(\theta \mid \mathcal{D})\, \mathrm{d}\theta$$

$$\approx \frac{1}{S} \sum_s p(y_* \mid x_*, \theta^{(s)}), \quad \theta^{(s)} \sim p(\theta \mid \mathcal{D})$$



How do we sample $p(\theta \mid \mathcal{D})$?

$$\pi(\theta) \;\propto\; \pi^*(\theta) \;=\; p(\mathcal{D} \mid \theta)\, p(\theta)$$

THE JOURNAL OF CHEMICAL PHYSICS VOLUME 21, NUMBER 6 JUNE, 1953

Equation of State Calculations by Fast Computing Machines

NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AND AUGUSTA H. TELLER,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

EDWARD TELLER,* *Department of Physics, University of Chicago, Chicago, Illinois*

(Received March 6, 1953)

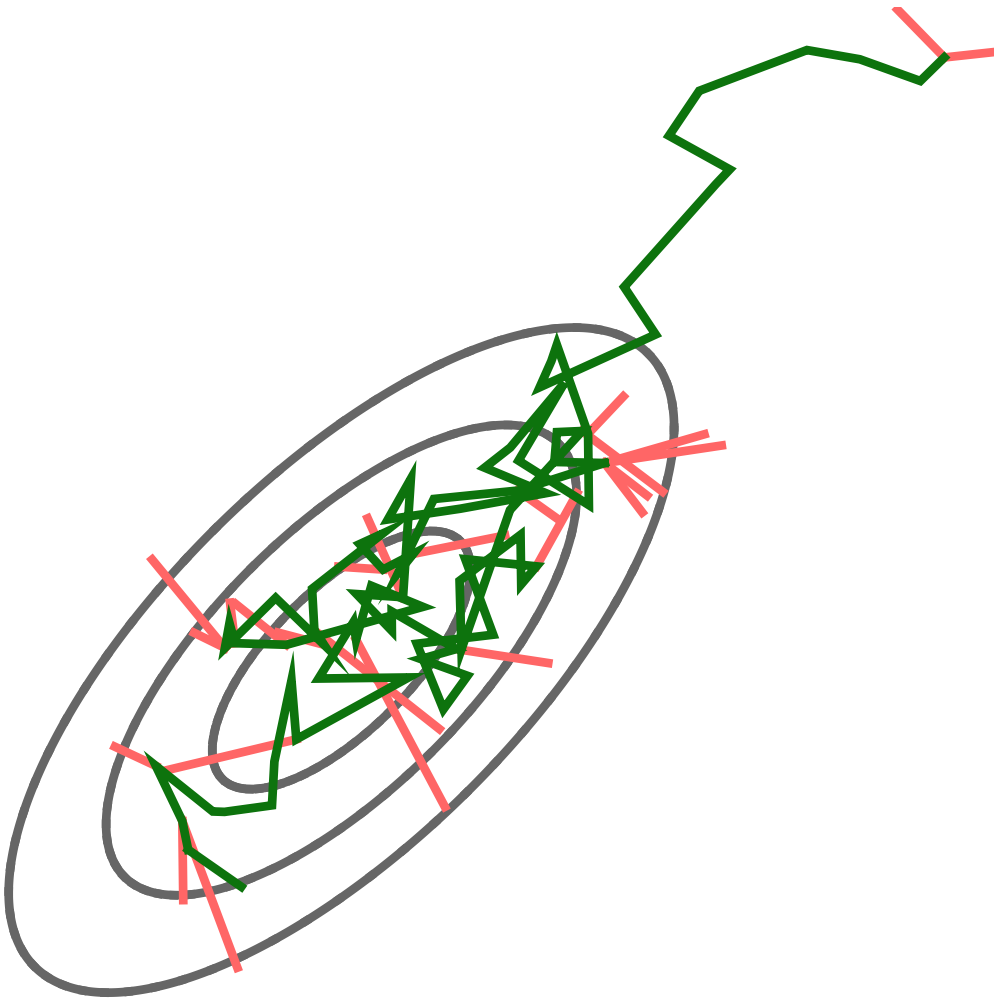
THE purpose of this paper is to describe a general method, suitable for fast electronic computing machines, of calculating the properties of any substance which may be considered as composed of interacting individual molecules. Classical statistics is assumed,

>35,000 citations

Marshall Rosenbluth's account:

http://dx.doi.org/10.1063/1.1887186

Metropolis–Hastings



$$\theta' \sim q(\theta'; \theta^{(s)})$$

```
if accept:
```

$$\theta^{(s+1)} \leftarrow \theta'$$

```
else:
```

$$\theta^{(s+1)} \leftarrow \theta^{(s)}$$

$$P(\text{accept}) = \min\left(1, \frac{\pi^*(\theta')\, q(\theta^{(s)}; \theta')}{\pi^*(\theta^{(s)})\, q(\theta'; \theta^{(s)})}\right)$$

What can we approximate?

Unbiased estimates, from subset of data?

A) $p(\mathcal{D} \mid \theta) = \prod_n p(\mathcal{D}_n \mid \theta)$

B) $\log p(\mathcal{D} \mid \theta) = \sum_n \log p(\mathcal{D}_n \mid \theta)$

C) $\nabla_\theta \, p(\mathcal{D} \mid \theta)$

D) $\nabla_\theta \log p(\mathcal{D} \mid \theta)$

E) None of the above

F) All of the above

Z) Don't know

Pseudo-marginal MCMC

MCMC with unbiased estimate of $\pi^*(\theta)$

Step towards big data: arXiv:1403.5693

Allowing negative estimators: arXiv:1306.4032

Easy-to-use algorithm: `http://iainmurray.net/pub/16pmss/`

Not yet a solution to MCMC for big data.

Dynamical methods

Simulation using $\nabla_\theta \log \pi^*(\theta)$ estimates

Langevin/Hamiltonian dynamics, and newer developments

An overview: https://papers.nips.cc/paper/5891-a-complete-recipe-for-stochastic-gradient-mcmc

A recent new direction: arXiv:1611.07873

Use for optimization?
http://proceedings.mlr.press/v51/chen16c.html

Sequential Monte Carlo

Popular in signal processing, probabilistic programming

One recent paper:

https://papers.nips.cc/paper/5450-asynchronous-anytime-sequential-monte-carlo

Variational Methods

Approximate $p(\theta \,|\, \mathcal{D}) = \pi(\theta)$ **with** $q(\theta)$

Classic approach: minimize $D_{\mathrm{KL}}(q||\pi)$

Monte Carlo approximation to gradients

"Black box": just need $\nabla_\theta \log p(\mathcal{D}_n \,|\, \theta)$

Refs: http://shakirm.com/papers/VITutorial.pdf
https://www.cs.toronto.edu/~duvenaud/papers/blackbox.pdf
http://www.inf.ed.ac.uk/teaching/courses/mlpr/2016/notes/

Will neural nets eat everything?

Can simulate from $p(\theta)$ and $p(\mathcal{D} \,|\, \theta)$

Training data!

$$\{\theta^{(s)}, \mathcal{D}^{(s)}\} \sim p(\theta, \mathcal{D})$$

Just fit $p(\mathcal{D} \,|\, \theta)$ and/or $p(\theta \,|\, \mathcal{D})$?

Example recognition network: arXiv:1605.06376
Example conditional density estimator: arXiv:1705.07057

Appendix slides

MCMC References

My first reading: MacKay's book and Neal's review:

www.inference.phy.cam.ac.uk/mackay/itila/

www.cs.toronto.edu/~radford/review.abstract.html

Handbook of Markov Chain Monte Carlo

(Brooks, Gelman, Jones, Meng eds, 2011)

http://www.mcmchandbook.net/HandbookSampleChapters.html

Practical MCMC examples

My exercise sheet:

http://iainmurray.net/teaching/09mlss/

BUGS examples and more in STAN:

https://github.com/stan-dev/example-models/

Kaggle entry:

http://iainmurray.net/pub/12kaggle_dark/