Examinable assignment 3 - part 1

- Use the IndeptGamma.R code to use MCMC to produce samples from a Gamma(2.3,2.7) random variable
- Show how the performance deteriorates when you run the algorithm to produce samples from a Gamma(0.1,0.01).
  - Use the Gelman plot to illustrate the deterioration in performance
- Find a proposal kernel (i.e. q(x->x'), the way of producing new candidate values) that performs more efficiently.
  - Again use the Gelman plots to show how performance has changed.
  - Discuss how the performance has improved

- Due On March 25th, at 11.59pm.

Code on GitHub in Assignment3 as 'IndeptGamma.R'

Examinable Assignment 3 part 2: Code-breaking (due April 5th, 1pm)

- Gzo uclfg gcpo C qhcs okof te Gollk Qoeetb zo rhf slvem ce h LtqqfLtkio Fcqxol Rlhcgz tvgfcso gzo gollhio tu Gzo Sheiolf. Gzo ahlmced qtg hggoesheg zhs yltvdzg gzo ihl tvg hes zo rhf fgcqq ztqsced gzo sttl taoe yoihvfo Gollk Qoeetbf qoug uttg rhf fgcqq shedqced tvgfcso, hf cu zo zhs utldtggoe zo zhs teo. Zo zhs h ktvedqttmced uhio yvg zcf zhcl rhf yteo rzcgo. Ktv itvqs goqq yk zcf okof gzhg zo rhf aqhfgolos gt gzo zhclqceo, yvg tgzolrcfo zo qttmos qcmo hek tgzol ecio ktveddvk ce h sceeol jhimog rzt zhs yooe faoesced gtt pviz pteok ce h jtceg gzhg obcfgf utl gzhg avlatfo hes utl et tgzol. Gzolo rhf h dclq yofcso zcp. Zol zhcl rhf h qtxoqk fzhso tu shlm los hes fzo zhs h scfgheg fpcqo te zol qcaf hes txol zol fztvqsolf fzo zhs h yqvo pcem gzhg hqptfg phso gzo LtqqfLtkio qttm qcmo jvfg hetgzol hvgtptycqo. Cg scseg wvcgo. Etgzced ihe. Gzo hggoesheg rhf gzo vfvhq zhqugtvdz izhlhigol ce h rzcgo ithg rcgz gzo ehpo tu gzo lofghvlheg fgcgizos hiltff gzo ulteg tu cg ce los. Zo rhf doggced uos va. "Qttm, pcfgol," zo fhcs rcgz he osdo gt zcf xtcio, "rtvqs ktv pces h rztqo qtg avqqced ktvl qod cegt gzo ihl ft C ihe mces tu fzvg gzo sttl Tl fztvqs C taoe cg hqq gzo rhk ft ktv ihe uhqq tvg" Gzo dclq dhxo zcp h qttm rzciz tvdzg gt zhxo fgvim hg qohfg utvl ceizof tvg tu zcf yhim. Cg scseg ytgzol zcp oetvdz gt dcxo zcp gzo fzhmof. Hg Gzo Sheiolf gzok dog gzo ftlg tu aotaqo gzhg scfcqqvfcte ktv hytvg rzhg h qtg tu dtquced pteok ihe st utl gzo aolftehqcgk.

[CodedMessage_Short.txt in Assignment3 repo on Github]

Assignment 3 - Part 2

- Use one of the methods you have met in the course to try to break the code.
- I may not be easy to break the code completely without resorting to manual 'tweaks' at the end, but you should be able to get to the point at which you can work out what the text is saying.
- Write it up as an Rmarkdown file and include a description of the methods you are using (for both parts of the assignment) and a discussion of your results. Upload it to your version of the Assignment 3 repo on Github please.
- Due On March 25th, at 11.59pm.

Another (Simpler!) MH-MCMC Example - Bivariate normals

- Assume we have some data from a bivariate normal distribution, for which we wish to estimate the means.
- For convenience, we will assume the variance/covariance structure is known.

$$\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \sim N\left[\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_2\sigma_1 & \sigma_2^2 \end{pmatrix}\right]$$

Properties:

$$X_1 \sim N(\mu_1, \sigma_1^2)$$

$$X_2 \sim N(\mu_2, \sigma_2^2)$$

$$Correlation(X_1, X_2) = \rho. \text{ [Recall } \rho_{X_1X_2} = \frac{Cov(X_1X_2)}{\sigma_X\sigma_Y}]$$

In-class exercise

- Try running the algorithm for each of the three test datasets:

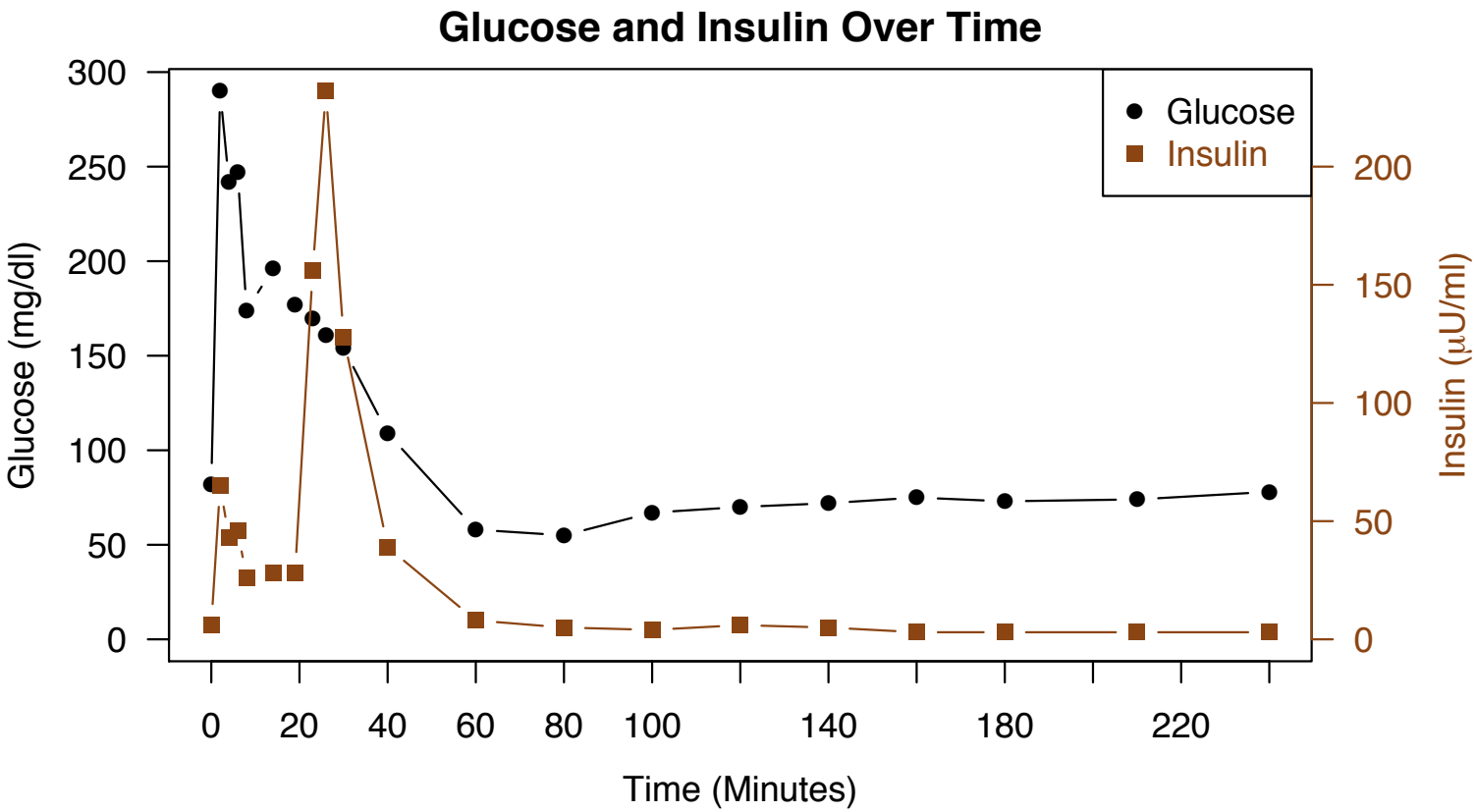
```
mu.vector <- c(3, 1)    # the vector of means for the multi-variate normal
variance.matrix <- cbind(c(1, 0), c(0, 4))   # the variance-covariance matrix for the multi-variate normal
# variance.matrix <- cbind(c(1, 1.5), c(1.5, 4))
# variance.matrix <- cbind(c(1, 1.98), c(1.98, 4))
```

- Explore behavior (use Gelman diagnostics: acfs, posterior densities) for each case.
- Fix it, for cases in which it performs badly.

Adaptive MCMC

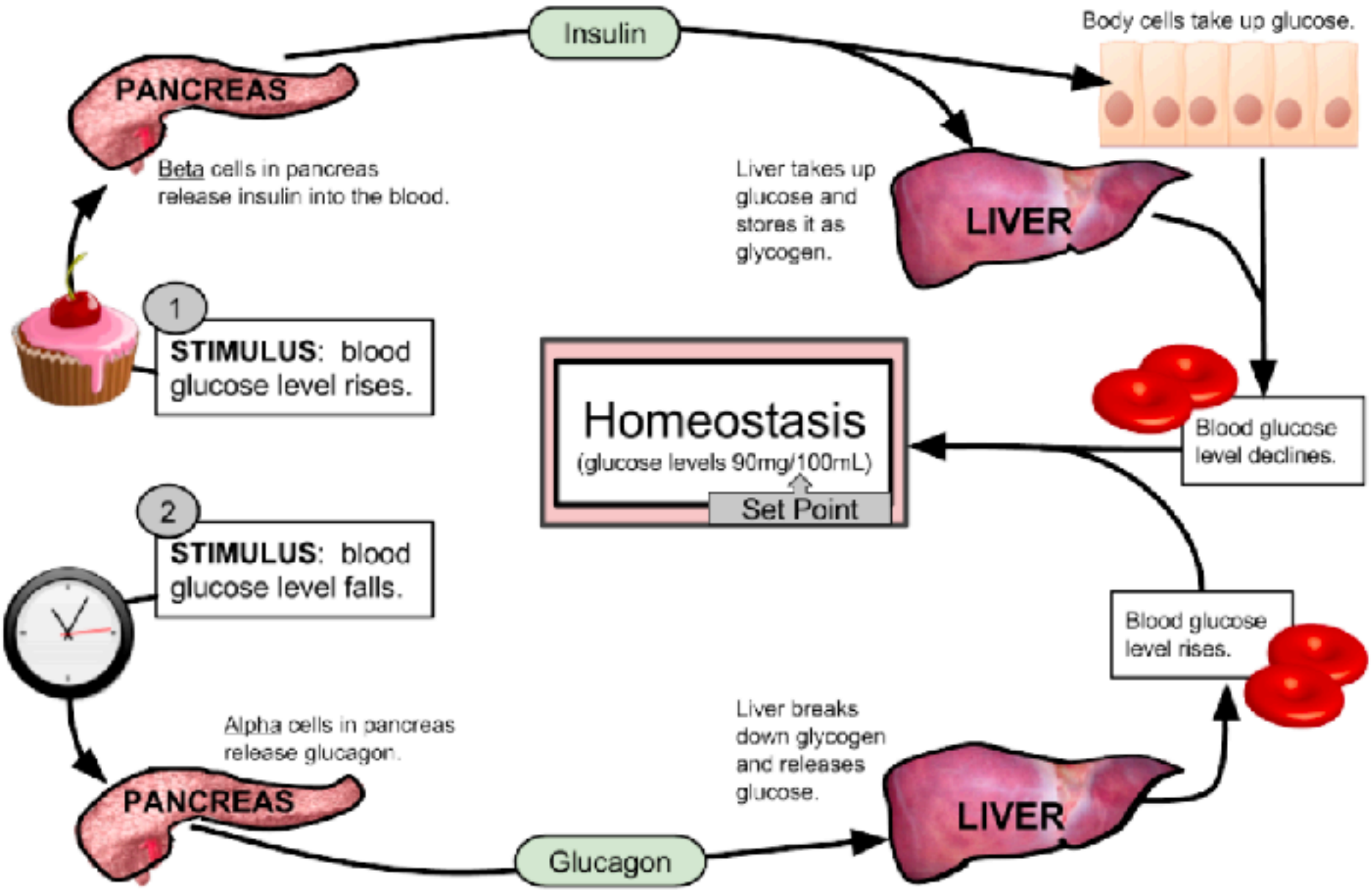
- with thanks to Patrick Muchmore (PhD) for original version of some of the slides

The Data



- FSIGT: Frequently Sampled Intravenous Glucose Test.
- At time 0, glucose is administered intravenously.
- Glucose and insulin levels measured 20 times over next 4 hours.

The Glucose/Insulin Metabolic Pathway



https://bio.libretexts.org/Learning_Objects/Worksheets/Book%3A_The_Biology_Corner_(Worksheets)/Anatomy_Worksheets/Feedback_Loops%3A_Glucose_and_Glucagon

The Model of Pacini and Bergman (1986)

- $G(t)$ and $I(t)$ are the concentrations of glucose and insulin in the bloodstream at time t.
- $X(t)$ models the effect of insulin on net glucose disappearance at time t.
- Related by the system of differential equations:

$$\frac{dG}{dt} = -(p_1 + X(t))G(t) + p_1 G_b,\ G(0) = G_\emptyset$$

$$\frac{dX}{dt} = -p_2 X(t) + p_3(I(t) - I_b),\ X(0) = 0$$

$$\frac{dI}{dt} = -nI(t) + \gamma(G(t) - h)t,\ I(0) = I_\emptyset$$

- Treat $I(t)$ as a known input, need to estimate 4 parameters p_1, p_2, p_3, and $G_\emptyset$.

Parameter Estimation

- Model is fit under the assumption that glucose measurements are subject to mean zero normal errors. The error variance is assumed to be proportional to the true value.

- Optimization can be used to calculate parameter point estimates, but the nonlinear relationship between data and parameters makes standard testing procedures impossible.

- Bayesian techniques enable further inferences about parameters (e.g. credible intervals). With four parameters use Markov chain Monte Carlo (MCMC) to deal with "curse of dimensionality".

Markov Chain Monte Carlo for Bayesian Inference

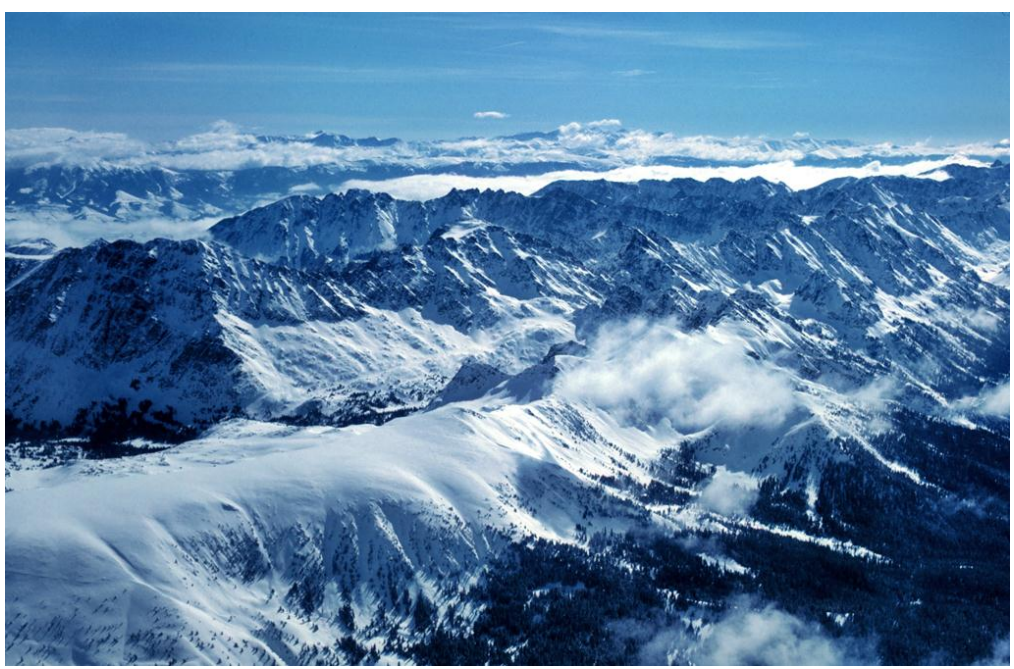
- Fundamental relationship of Bayesian inference: $f(\theta|x) \propto f(x|\theta)f(\theta)$. In words: the posterior is proportional to the likelihood times the prior. In this example $x =$ glucose measurements, $\theta = (p_1, p_2, p_3, G_\emptyset)$.
- MCMC: using a symmetric proposal kernel and uniform priors

1. Starting at some point θ, pick another "nearby" point θ'.
2. If $f(\theta'|x) > f(\theta|x)$ definitely move to θ', i.e. make θ' the new starting point θ. Otherwise, move with probability[2] $\frac{f(\theta'|x)}{f(\theta|x)}$.
3. Repeat infinitely many times.

Problem: Choosing Σ

i.e., the distribution of θ' given θ

- "Nearby" points are determined by a probability distribution. A common choice is MVN(θ, Σ).
- Efficiency of MCMC depends crucially on Σ, and what works well in one case may be a very poor choice in another.
- How to avoid manual calibration of Σ?



Problem: Choosing Σ

Solution: Adaptive MCMC

- Adaptive MCMC updates Σ while the chain is run. This eliminates the need to manually calibrate Σ as the chain will "learn" the appropriate transition variance.

- In the present model, much of the difficulty in estimation arises from the varying nature of the correlation among the parameters across individuals. By using a multivariate transition distribution, adaptive MCMC handles this complication automatically.

- Note: Because Σ depends on multiple prior steps, the Markov property does not hold. However, recent papers such as Roberts and Rosenthal (2009) show the desired stationary distribution is still achieved in a relatively wide range of cases.

Adaptive MCMC

1. If at θ_n, propose move to θ_{n+1} according to transition kernel $q(\theta_n \rightarrow \theta_{n+1} | \theta_1, \theta_2, \ldots \theta_n)$.
2. Calculate

$$h = \min \left\{ 1, \frac{f(\theta_{n+1}) q(\theta_{n+1} \rightarrow \theta_n \mid \theta_1, \ldots, [\theta_n], \theta_{n+1})}{f(\theta_n) q(\theta_n \rightarrow \theta_{n+1} \mid \theta_1, \ldots, \theta_n)} \right\}$$

3. Move to θ_{n+1} with prob. h, else remain at θ_n, (so set $\theta_{n+1} = \theta_n$).
4. Return to 1.

Does the Markov chain have a stationary distribution? Is it even Markov?

Examples of adaption

- **Global adaption**: 'train' your transition kernel q() using the entire history of the algorithm. For example, calculate the empirical covariance matrix for all parameter values output so far and use that as your transition(proposal) kernel.
  - **Extremely** computationally challenging.
  - This process does not obviously c converge to anything.

- **Local adaptation**:
  - Use just the last m iterations
  - Use the nearest m iteration.
    - Computationally easier.

Adaptive MCMC - Ensuring convergence

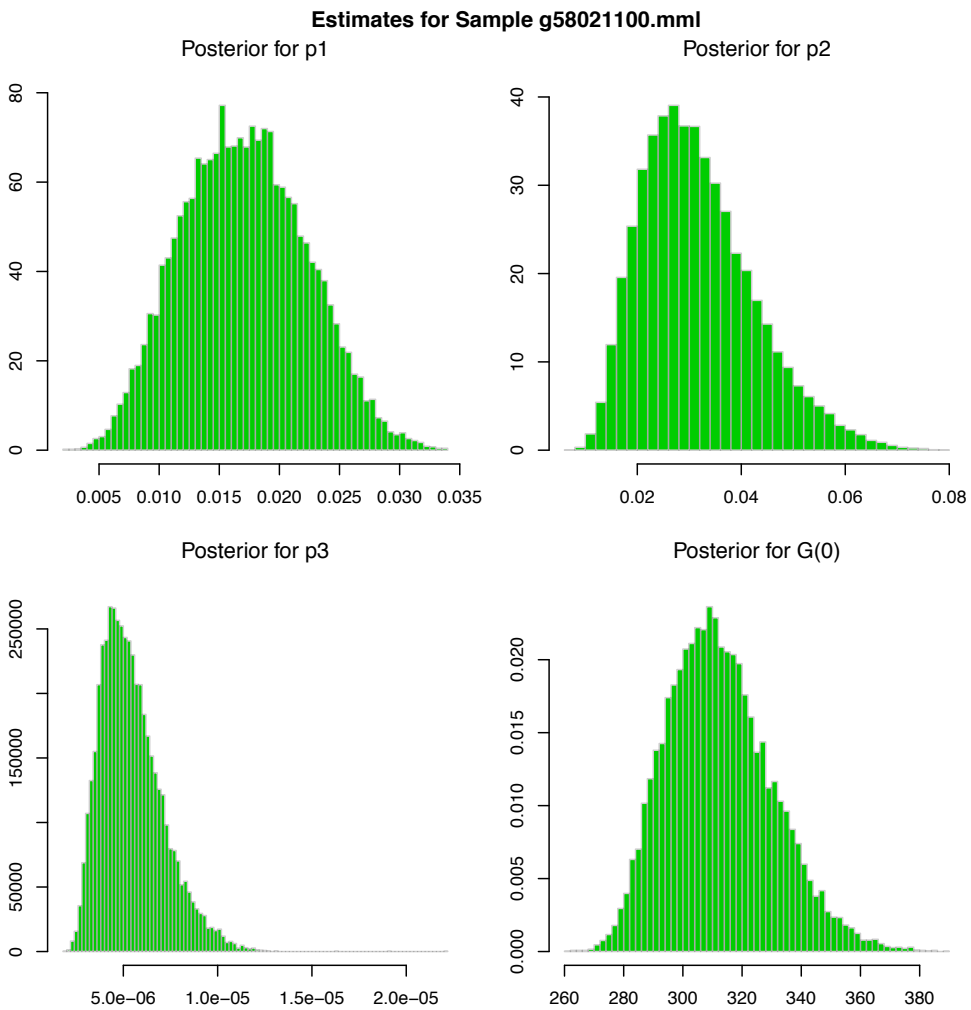
Fix 1: Stop adapting after *N* steps, for some fixed *N*. Then the Markov chain has stationary distribution f. Can even adapt by 'less and less' over time (so that the total ambient of adaptation is finite).

Fix 2: Use a proposal kernel that is a mixture of some initial (non-adaptive) kernel and the adaptive kernel. Then, provided the adaptive kernel uses ALL of $\theta_1,\ldots,\theta_n$ you avoid degenerate behavior.

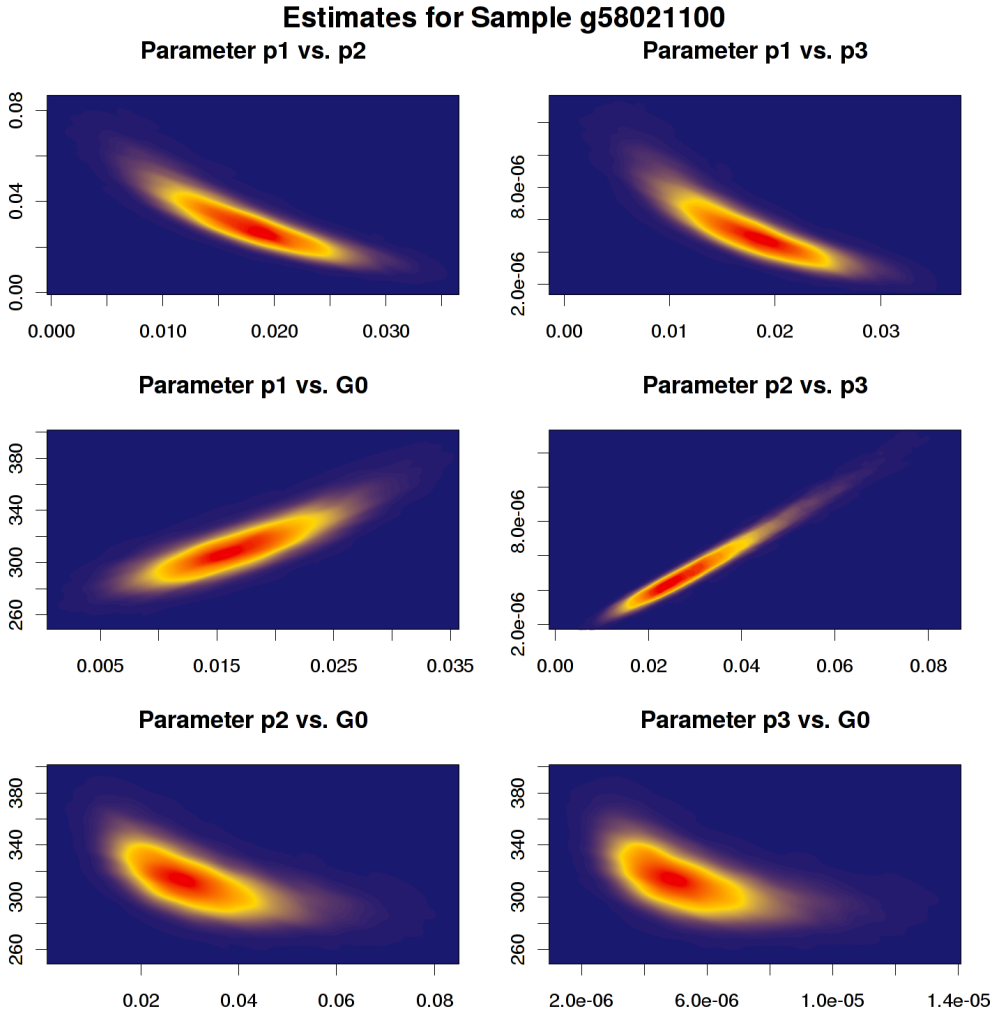
Roberts, Gareth O., and Jeffrey S. Rosenthal. "Examples of adaptive MCMC." Journal of Computational and Graphical Statistics 18.2 (2009): 349-367.

"Handbook of MCMC" (Brooks, Gelman, Jones, Meng) - Chapman and Hall.

Example of Parameter Estimates



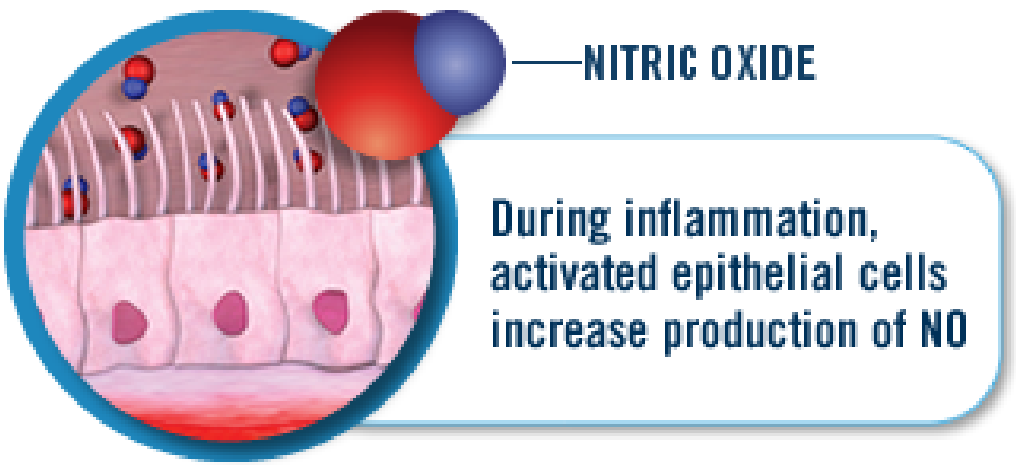
Univariate marginals

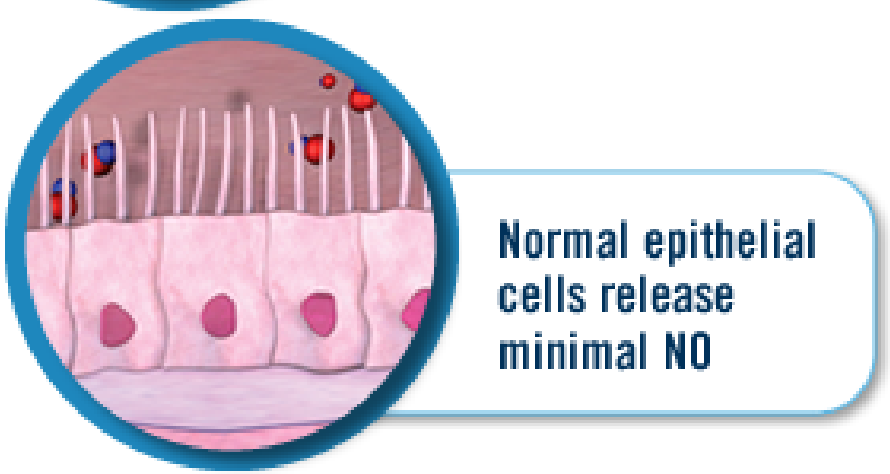


Bivariate marginals

Next Application: Fractional Exhaled Nitric Oxide (FeNO)

- NO produced in the alveoli is a biomarker of inflammation.
- Measured at the mouth; what can be inferred about the lungs?





FeNO: Airway Model

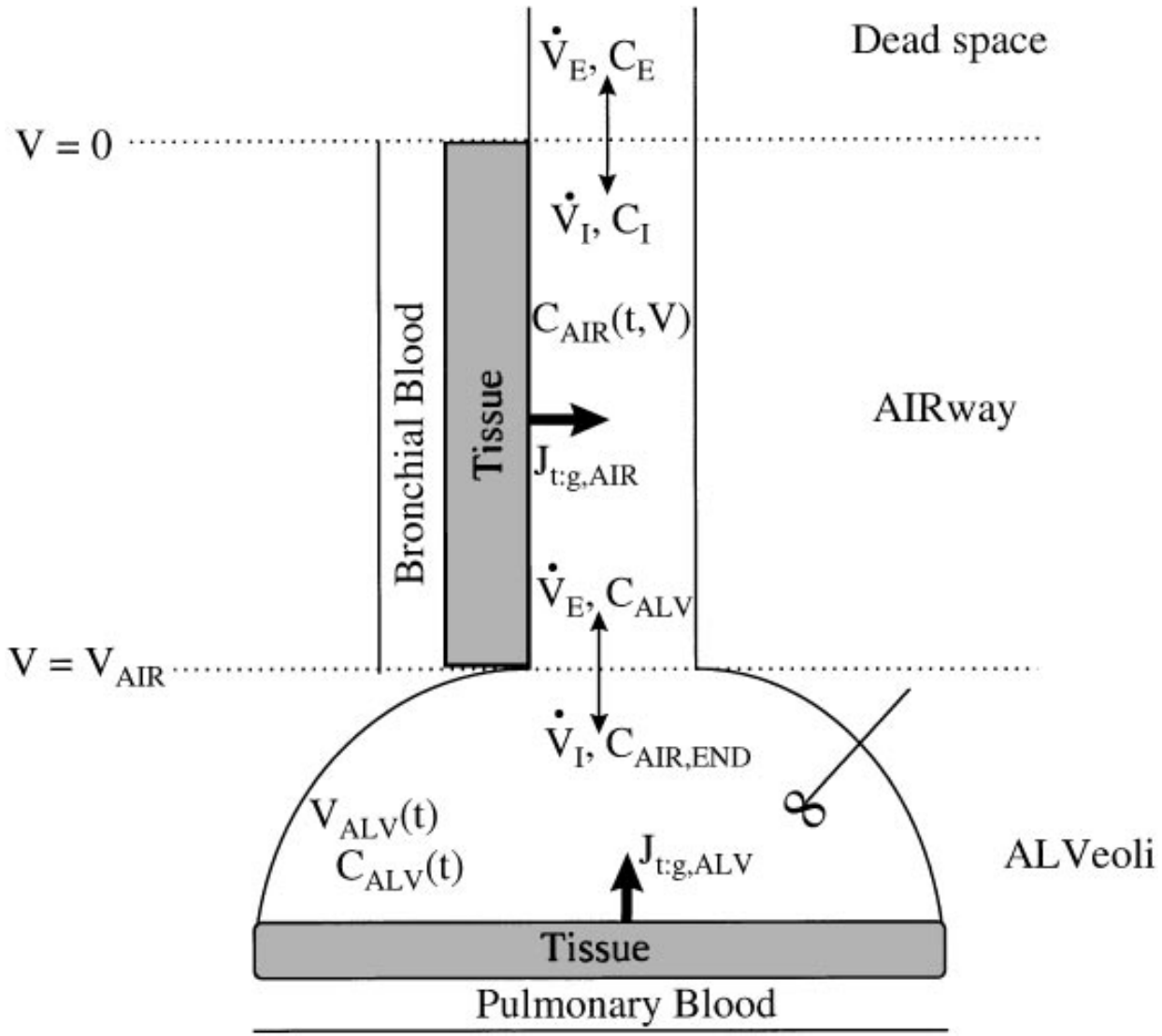
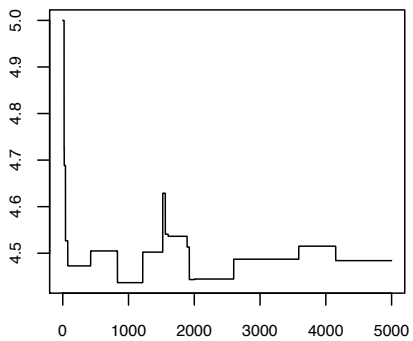
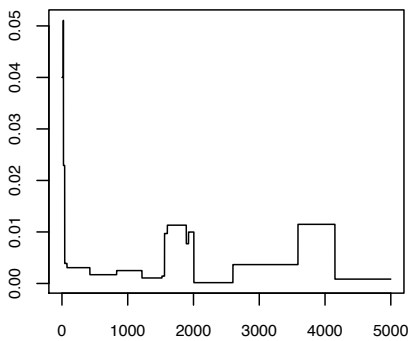
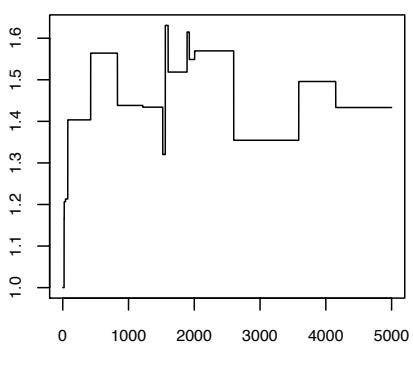


Figure from Tsoukias and George (1998).

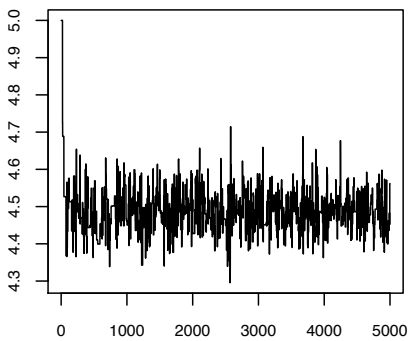
Adaptive Proposal Chains for FeNO Parameters

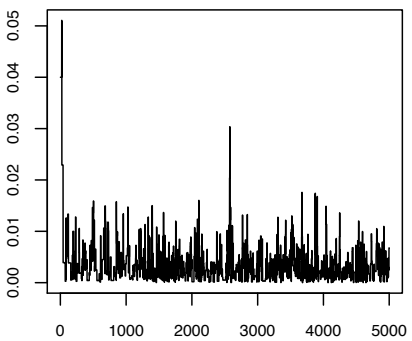


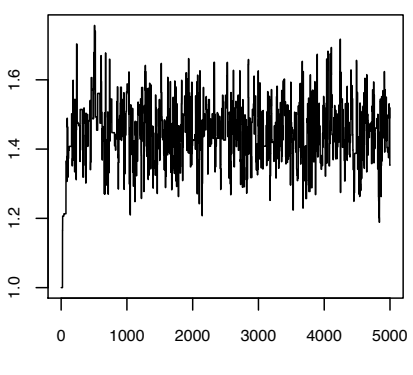




Fixed proposal distribution







Adaptive proposal distribution

References

Bergman, R. N. (1989): "Toward physiological understanding of glucose tolerance: minimal-model approach," *Diabetes*, 38, 1512–1527.

Pacini, G. and R. N. Bergman (1986): "MINMOD: a computer program to calculate insulin sensitivity and pancreatic responsivity from the frequently sampled intravenous glucose tolerance test," *Comput Methods Programs Biomed*, 23, 113–122.

Roberts, G. O. and J. S. Rosenthal (2009): "Examples of adaptive mcmc," *Journal of Computational and Graphical Statistics*, 18, 349–367.

Tsoukias, N. M. and S. C. George (1998): "A two-compartment model of pulmonary nitric oxide exchange dynamics," *Journal of Applied Physiology*, 85, 653–666.

Impact of different fixed flow sampling protocols on flow-independent exhaled nitric oxide parameter estimates using the Bayesian dynamic two-compartment model. P.Muchmore et al. Physiological Reports. 2020;8:e14336.

AdaptMCMC

- There is an R-package for adaptive MCMC: AdaptMCMC.
- It aims to 'tune' the adaption so that a target acceptance rate is reached (typically ~23%).
- The intuition is that we scale the step-size (i.e., var-covar matrix) up if the acceptance rate is too high, or down if the acceptance rate is too low.
- See example in AdaptationMCMCExample.Rmd on Github ("Week9_supporting_materials")

MCMC: Tempering

- Multiple, coupled samplers (Geyer, C. (1992). Practical Markov chain Monte Carlo. *Statistical Science*, *7*, 473–483.)

- Annealing/Tempering (Geyer and Thompson, Journal of the American Statistical Association, 90:909-920, 1995).

Tempering (Annealing)

- Suppose we are inferring a posterior distribution f(θ|D), for example
- Define a 'temperature' t ≥ 1 (from a range of possible temperatures **T**)
- Augment state-space to be (θ,t) ∈ (Θ,**T**)
- Work with $f(\theta|D)^{1/t}$ rather than f(θ|D)
- As t increases, it smoothes out the posterior ⇒ mixing is improved
- Must only sample final answer from iterations with t=1

$$h = \min\left\{1, \frac{[f(\Theta'|D)^{1/t'} q(\Theta' \to \Theta)}{[f(\Theta|D)^{1/t} q(\Theta \to \Theta')}\right\}$$

Parallel Tempering (Population MCMC)

- Run several chains in parallel, each at a different temperature.
- Propose that the chains swap stats from time to time.
- For a simple example, see the file ‘Tempering.Rmd’ on Github (Week8_supporting_materials)

Parallel Tempering - Population MCMC

5. Nonlinear ODE models of biochemical pathways

The inferential framework presented and illustrated is completely general and we now demonstrate it on dynamic models defined by systems of nonlinear ordinary Differential Equations (ODE). The example we now consider is one which is directly related to systems biol
been developed to demonstrate t
models describe a process of enz
At the same time enzyme degrad
The first model (see Fig. 4(a))
differential equations (15) that de
parameters, the statistical model
The systematic component of t
stochastic component of the mod
at each of N distinct time points c
which the ODE's define will gene
inducing a complex covariance st
by a multivariate Gaussian Proces
with mean μ and covariance $\Sigma \otimes$
values for S, D, R, Rpp at each of t
the covariance function values at
covariance.

$$\mathcal{M}_1 = \begin{cases} \frac{dS}{dt} = -k_1 \cdot S \\ \frac{dD}{dt} = k_1 \cdot S \\ \frac{dR}{dt} = -\frac{V_1 \cdot R \cdot S}{Km_1 + R} + \frac{V_2 \cdot Rpp}{Km_2 + Rpp} \\ \frac{dRpp}{dt} = \frac{V_1 \cdot R \cdot S}{Km_1 + R} - \frac{V_2 \cdot Rpp}{Km_2 + Rpp}. \end{cases} \quad (15)$$

Theoretical Computer Science 408 (2008) 4–16

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Bayesian inference for differential equations

Mark Girolami *

University of Glasgow, Department of Computing Science, Sir Alwyn Williams Building Room 302, G12 8QQ Scotland, United Kingdom

Population MCMC

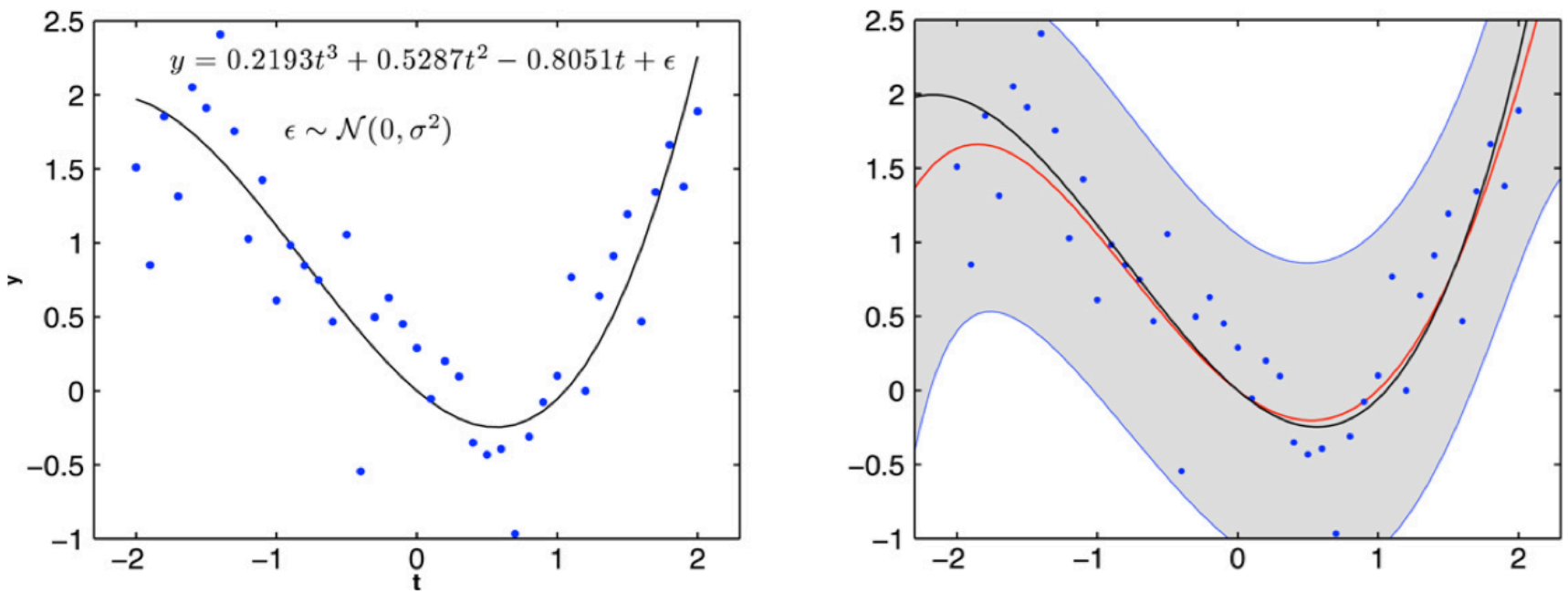


Fig. 1. The left hand figure shows the observed data points and the solid line is the noise free function which generated the finite number of noisy observations. The right hand figure shows the observed data and the noise free function. In addition the Bayesian model averaged predictive mean value (Eq. (13)), is shown in solid red and the shaded region corresponds to ± one standard deviation of the model averaged predictive distribution (Eq. (14)).

They generate time-series data using 3rd order polynomial plus Gaussian noise

Population MCMC

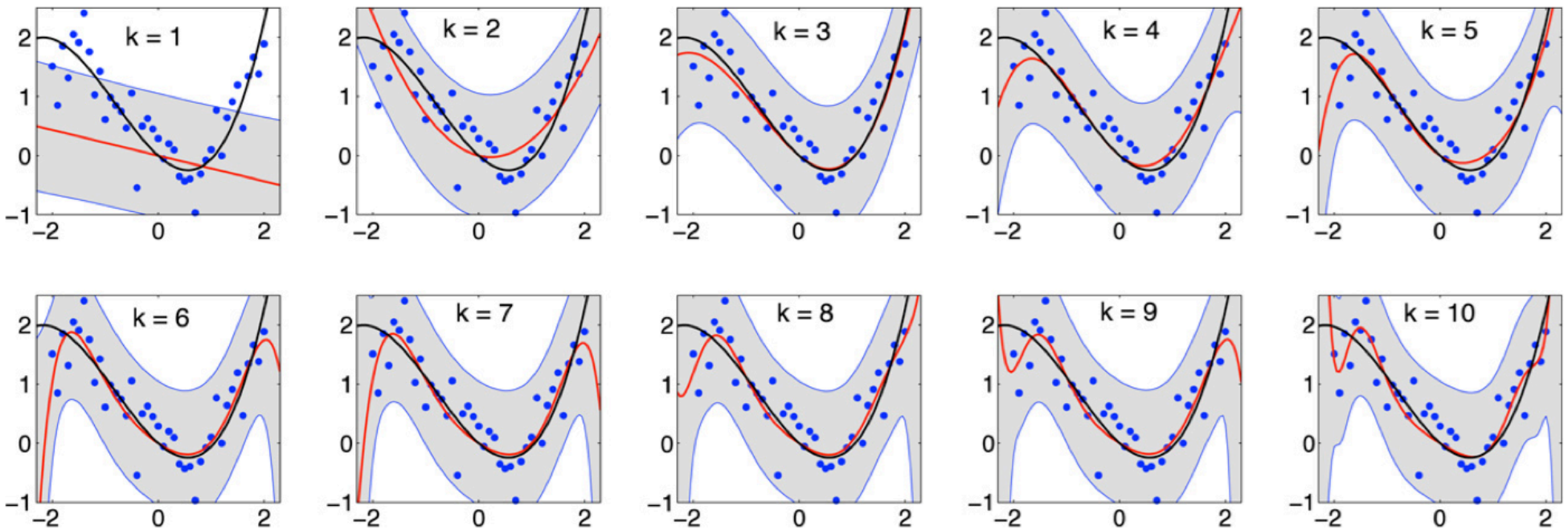


Fig. 3. The figures show the original data, the true underlying function, as well as the mean and variance of the predictive distribution for each of the ten models considered.

Fitting 10 models: $M_1, M_2, \ldots, M_{10}$.
Model M_i is an i^{th} order polynomial

Population MCMC

Model posterior probabilities

Maximum likelihood estimates

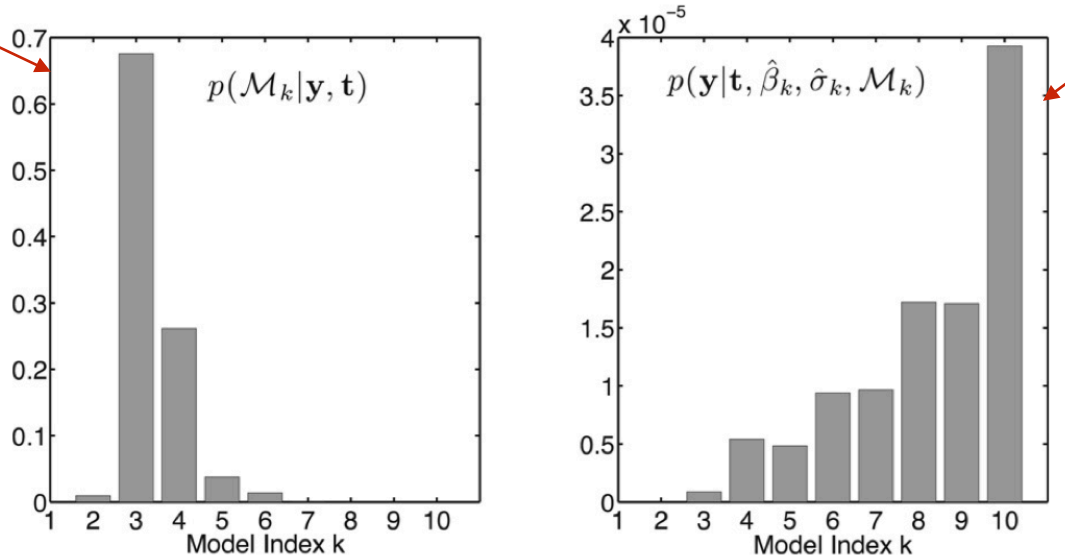


Fig. 2. The right hand bar chart displays the data likelihood for each model from $\mathcal{M}_1$ to $\mathcal{M}_{10}$. The data likelihood is obtained when the *maximum a posteriori* point estimates of the model parameters $\hat{\beta}_k, \hat{\sigma}$ are employed. The data likelihood is a raw measure of how well each model *fits* the observed data and it is clear that as the model complexity, that is the order of the polynomial expansion, increases then the misfit to the data decreases and so the data likelihood increases. The left hand bar chart shows the discrete probability distribution over the data models and here we observe a quite different pattern.

They fitted polynomial models of orders 1 through 10

$$p(\mathcal{M}_k \mid y,t) \propto p(y,t \mid \mathcal{M}_k)\pi(\mathcal{M}_k)$$
$$= \int_{\beta_k,\sigma_k} p(y,t \mid \mathcal{M}_k,\beta_k,\sigma_k)\pi(\beta_k,\sigma_k)d(\beta_k,\sigma_k)\ \pi(\mathcal{M}_k)$$

Population MCMC

Consider the following simple model of a Circadian Oscillator as originally described in [14].

$$\frac{dx}{dt} = \frac{k_1}{36 + k_2 y} - k_3, \qquad \frac{dy}{dt} = k_4 x - k_5 \tag{23}$$

where $k_1 = 72, k_2 = 1, k_3 = 2, k_4 = 1$ and $k_5 = 1$, and the initial values are set to $x(0) = 7$ and $y(0) = -10$. The data comprised of 120 data points, $\mathbf{y}$, which were simulated using these settings, between $t = 0$ and $t = 60$ in discrete steps of 0.5, $\mathbf{t}$, to which independent Gaussian noise was added with variance $\sigma = 0.5$. The posterior was then calculated conditionally over the parameters k_3 and k_4 and plotted from 0 to 5 on each axis.

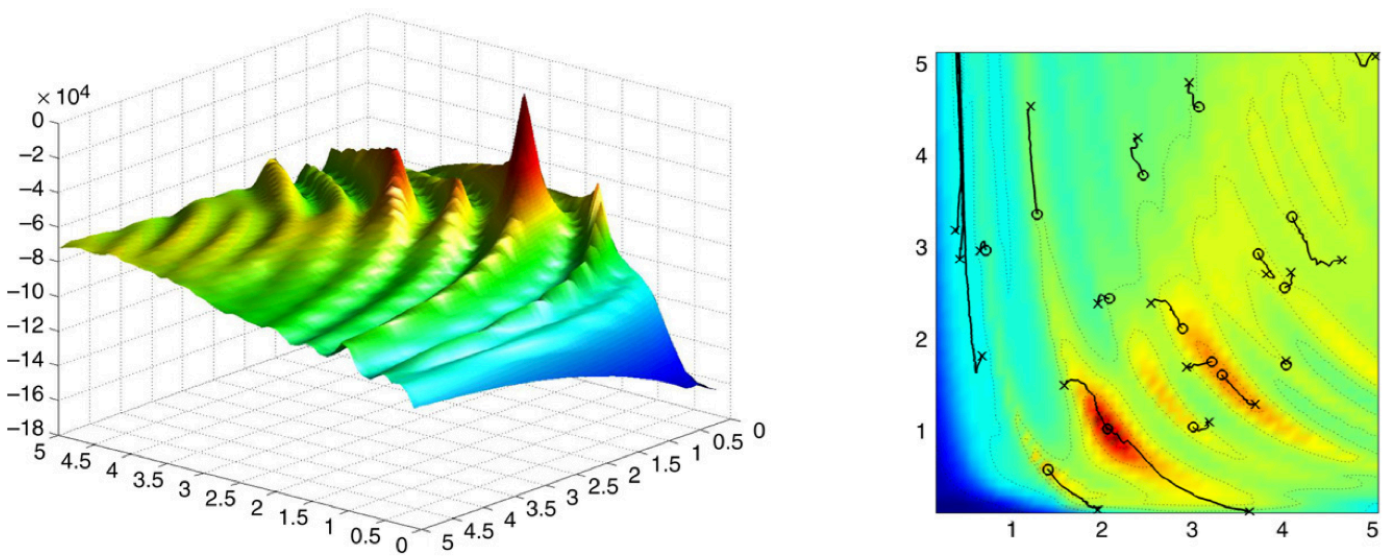


Fig. 5. The posterior $p(k_3, k_4|\mathbf{y}, \mathbf{t}, k_1 = 72, k_2 = 1, k_5 = 1, x(0) = 7, y(0) = -10, \sigma = 0.5)$. The right hand plot shows the path of twenty independent Markov chains derived from the Metropolis method, it is clear that they do not mix or cover the whole parameter space.

Population MCMC

- Girolami et al. 2008 do the following
  - Run a set of MCMC processes in parallel: $M_1,\ldots,M_n$.
  - Chain M_i runs at temperature T_i, where $0=T_1<T_2<\ldots<T_n=1$. (Note T_i corresponds to 1/t in the description of tempering on our slide 27.)
  - So:
    - chain M_1 explores the prior distribution
    - chain M_n explore the posterior distribution
    - other chains explore intermediate distributions.
  - At each iteration:
    - Apply local moves to each chain (M_i) via some proposal kernel q_i
    - Then, attempt to exchange the states of one or more pairs of chains (typically try to exchange M_i with M_{i+1} or M_{i-1})
  - Wait until stationarity is reached and then sample from chain M_n only.

$$h = \min\left\{1, \frac{f(D|\theta')^{T_i}\pi(\theta')q(\theta' \to \theta)}{f(D|\theta)^{T_i}\pi(\theta)q(\theta \to \theta')}\right\} \text{ (where } \theta = (\beta, \sigma)) \qquad (1)$$

Population MCMC

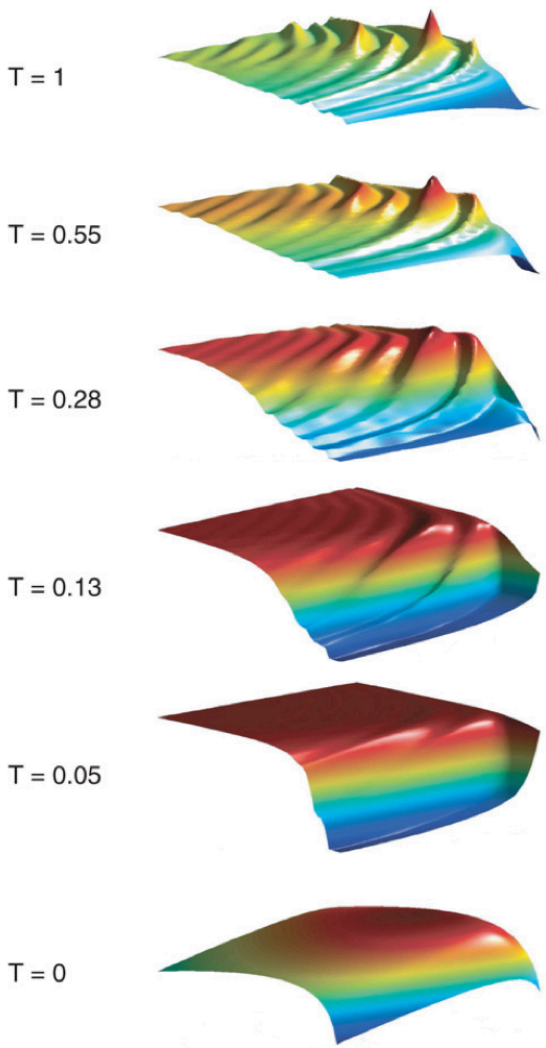


Fig. 6. Tempered posterior surfaces conditioned on two parameters of a 2-variable Goodwin oscillator model. The shapes of the power posteriors change most rapidly between between $t = 0$ and $t = 0.28$, and the overall transition from smooth prior to spiky posterior allows chains to globally explore the parameter space through exchanges between temperatures.

Hastings Ratio for 'swap' move

$$h = \min\left\{1, \frac{f(\theta_i|D)^{T_j} f(\theta_j|D)^{T_i} q(\Theta' \to \Theta)}{f(\theta_i|D)^{T_i} f(\theta_j|D)^{T_j} q(\Theta \to \Theta')}\right\}$$

where chain *i* was in state θ_i and chain *j* was in state θ_j before the swap.

h is likely to be largest when swapping chains with similar temperatures (T_i, T_j)

"cross-over" moves are also possible: take subset of parameters and attempt to swap their current values between chains i and j (for some i,j). [Hastings ratio similar to the above]

Population MCMC

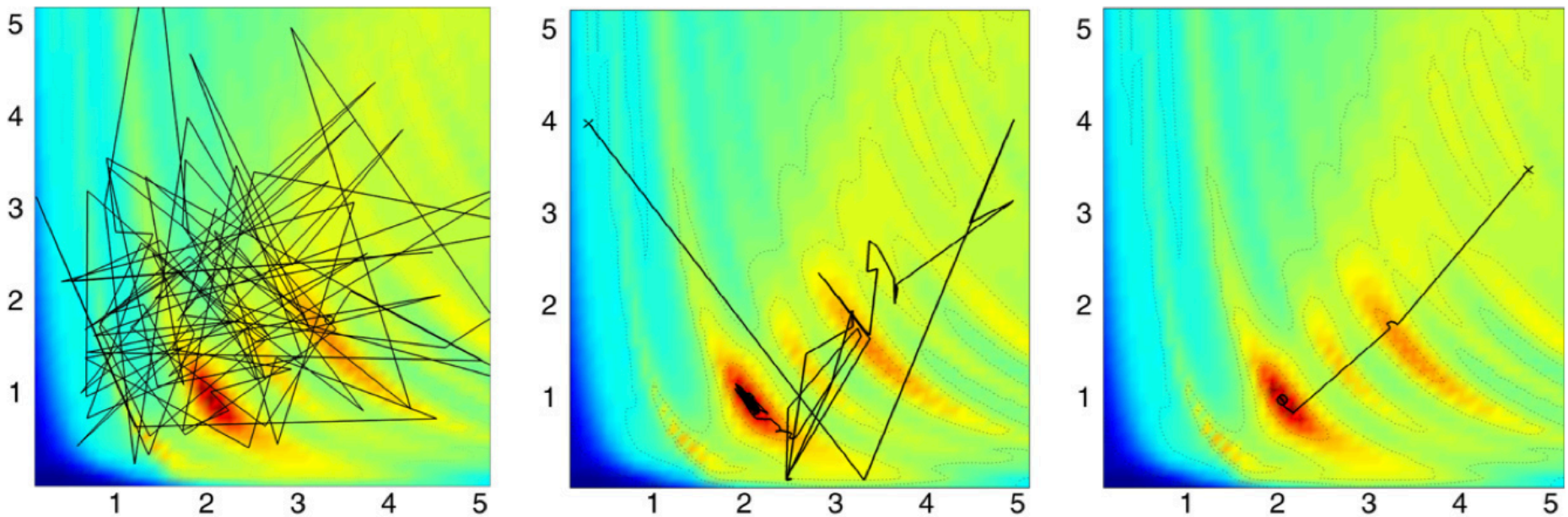


Fig. 7. Samples obtained from a chain at $t = 0$, which is effectively sampling from the prior. The free movement within the parameter space is clear to see. The iso-contours of the posterior are also plotted in this case. Progress of samples drawn from a chain at temperature $t = 0.5$ are shown against the iso-contours of the full posterior. The free movement across modes is most apparent and this is mainly due to the exchange proposals between temperatures. Samples drawn from the posterior, when $t = 1$. There are great differences between this and the highly localised *sticky* exploration in Fig. 5. The Population MCMC algorithm clearly has a much greater ability to move between modes in order to find the most likely one.

Tempered MCMC Example: Developmental pathways

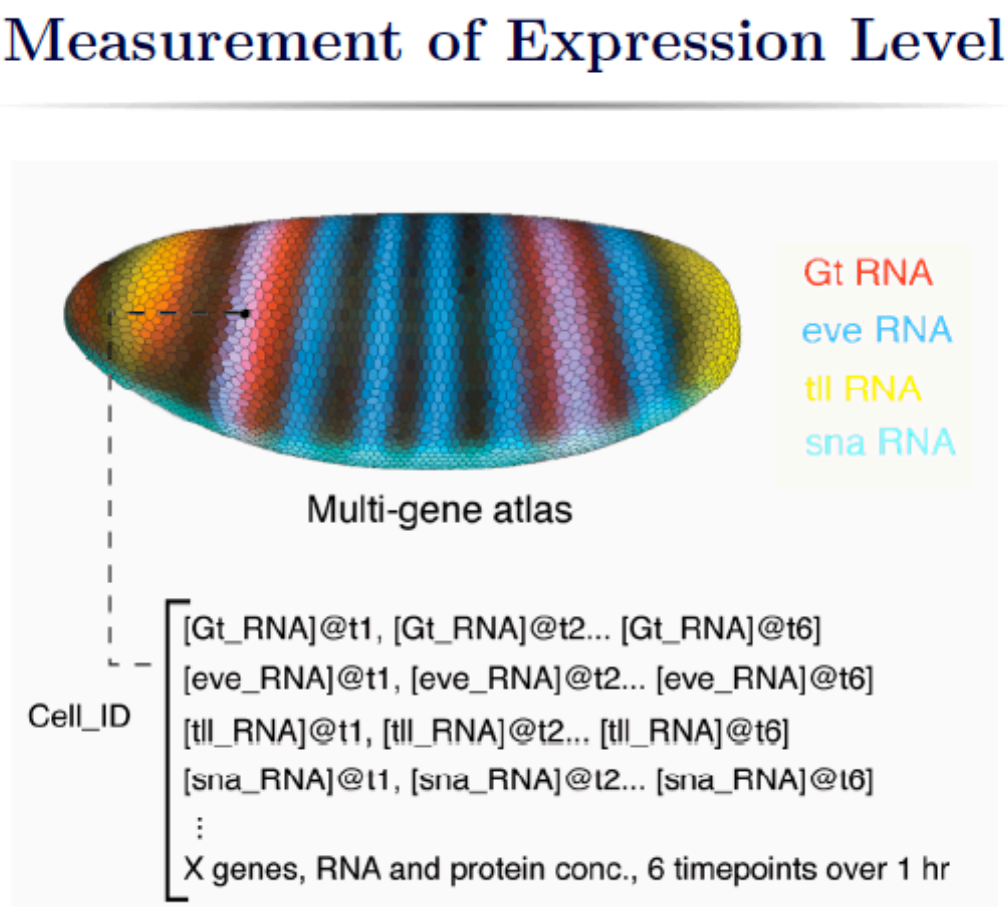
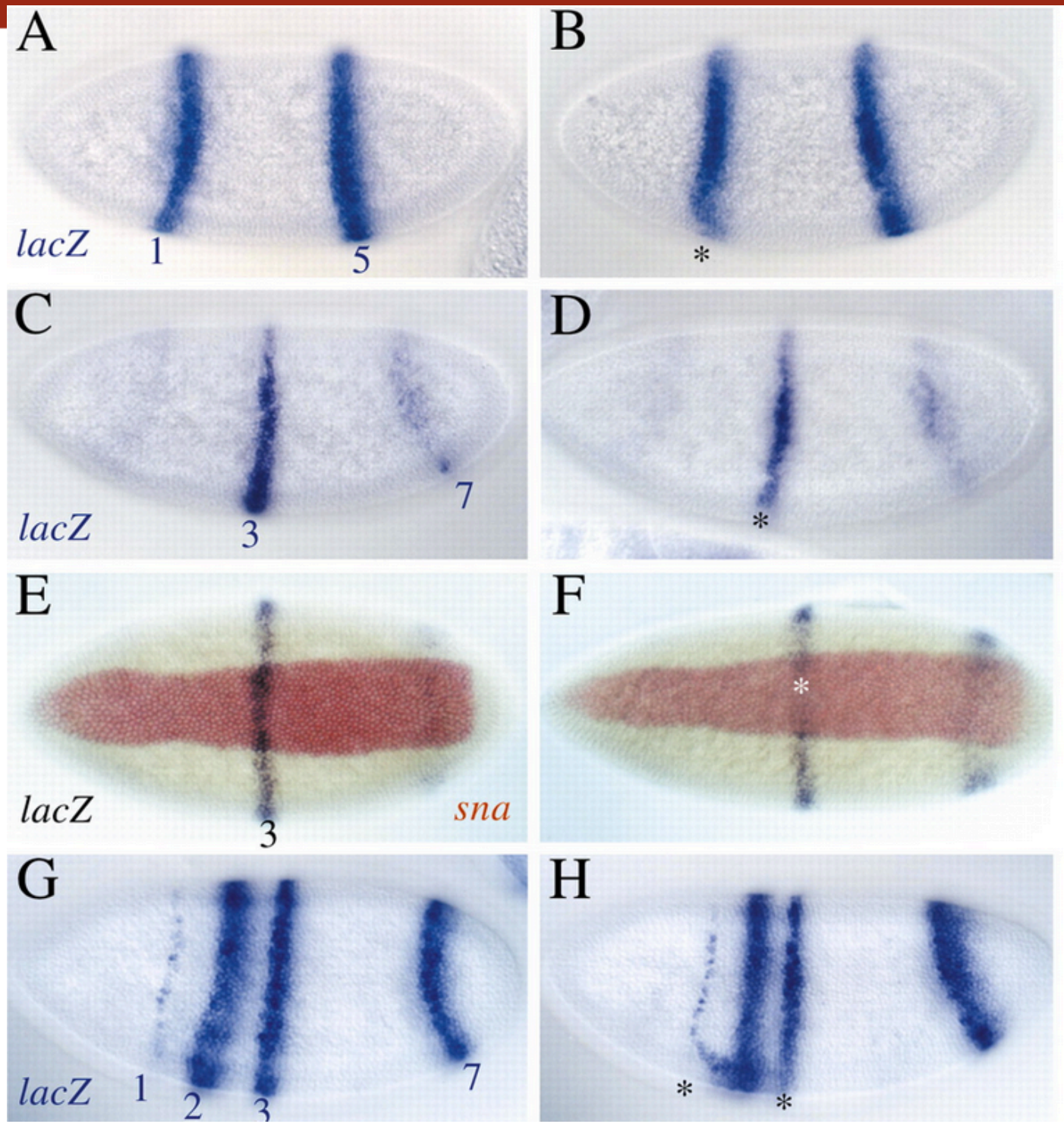


Figure 3: A single morphological framework, containing expression information for multiple genes per cell over time. Data is from *D.mel*, collected by BDTNP.


A
lacZ
1
5
B
*
C
lacZ
3
7
D
*
E
lacZ
3
sna
F
*
G
lacZ
1
2
3
7
H
*
*

Gene expression patterns

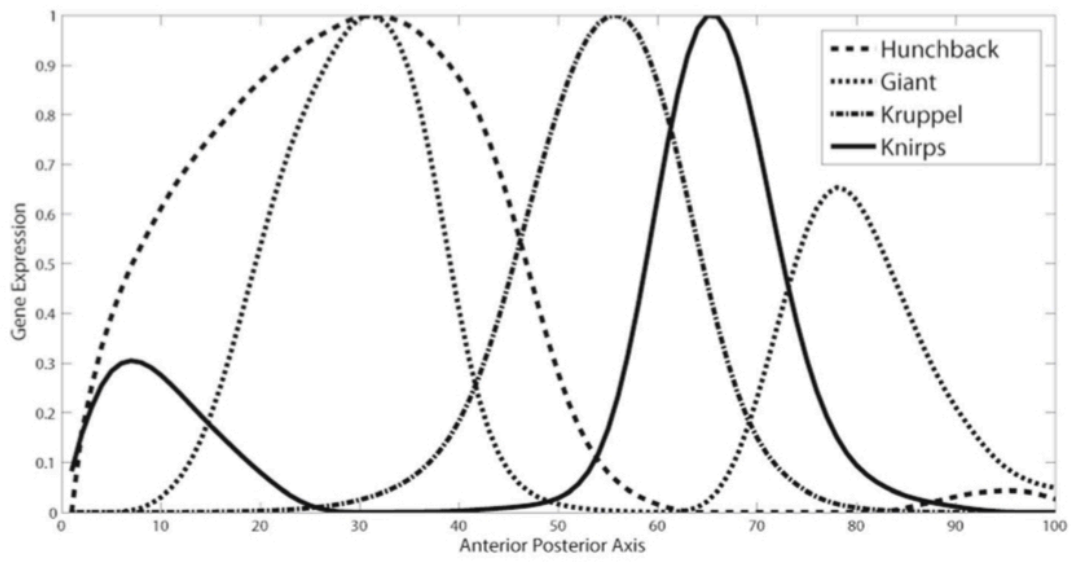


Fig. 1 Expression data. Gap gene expression values at cleavage cycle 14.4 along the anterior-posterior axis of developing embryo are used to fit the model

Zubair, A., Rosen, I. G., Nuzdhn, S. V., & Marjoram, P. (2019). Bayesian model selection for the Drosophila gap gene network. BMC Bioinformatics, 20(1), 327.

Developmental pathway

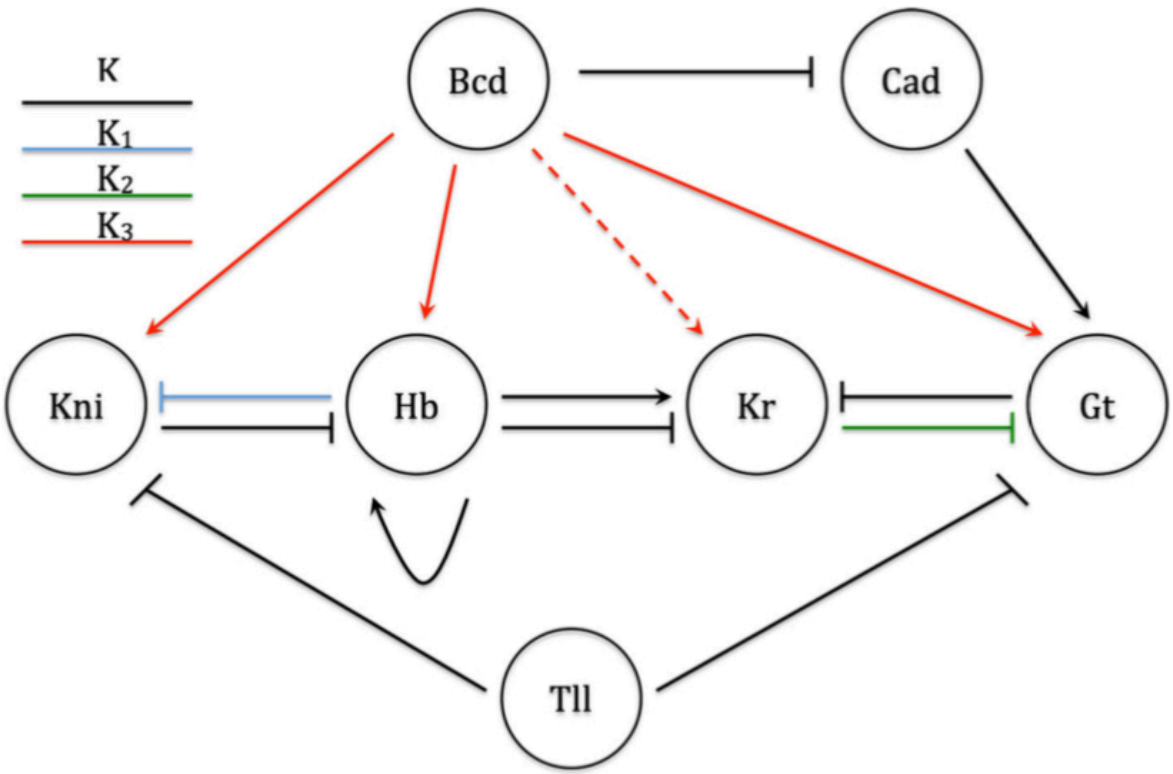


Fig. 2 Gap gene network. Gap gene network showing regulatory interactions between maternal genes, Bicoid (Bcd) & Caudal (Cad), and gap genes (Knirps (Kni), Hunchback (Hb), Kruppel (Kr), Giant (Gt)). Two types of binding affinity parameters are shown - global (K) and edge-specific (K_1, K_2, K_3). We also investigate evidence for Bicoid activation of Kruppel (shown as dashed arrow)

Represented by a set of partial differential equations

MCMC details

- Simplest analysis: Network topology is assumed to be fixed.
- Statespace: list of parameters. At any given iteration of the MCMC process, each parameter will have an assigned value.
- Transition kernel q will propose changes to parameters. e.g., pick a parameter at random, and add a Unif[-0.5,0.5] to it. (So the q-term will cancel in the HR if we use reflecting boundaries when needed.)
- Given a set of parameter values, the differential equations tell us the expected level of the measured value of **gene expression** at each **spatial** point. Suppose, the expected value at location t for that set of parameter values is y(t).
- Assume measurement error is normally deviated with mean 0. So, if the measured value is denoted by z(t), we assume $z(t) \sim \text{Normal}(y(t),\sigma^2)$.
- Assume measured values at each location are independent **conditional on the expected values**.
- So, the likelihood of z=(z(1),z(2),…,z(n)) is a product of Normal densities.
- Used Parallel Tempered MCMC

- More complex analysis: which network topology is best (should branches be removed/added?)

Example convergence diagnostics

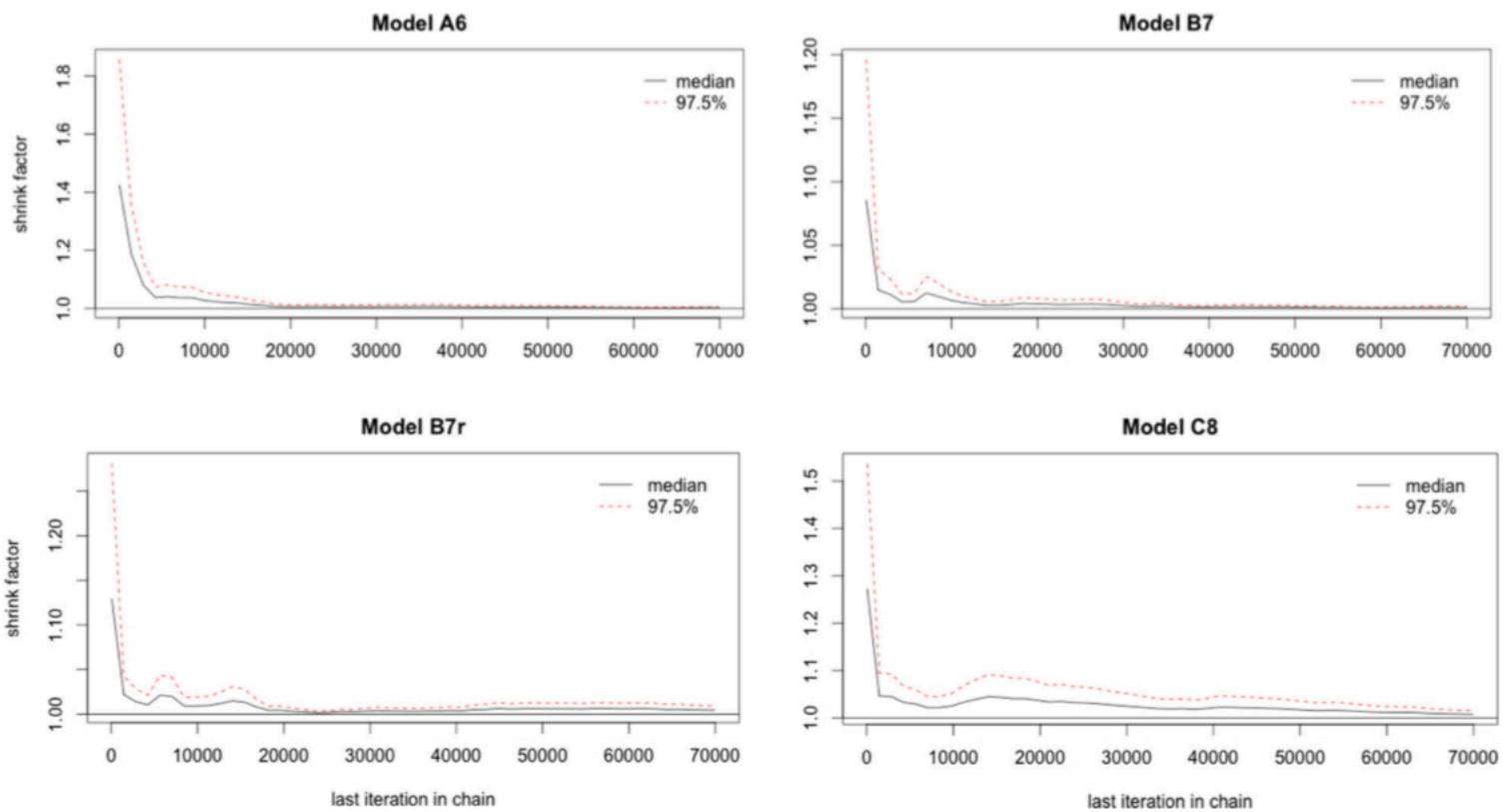


Fig. 3 MCMC convergence diagnostics. Gelman plot showing the evolution of the gelmna-rubin statistic for four models (A6, B7, B7r, C8) as a function of iterations. The diagnostic metric was evaluated for 10 independent chains with random start points for each model. Values less 1.2 imply good mixing of the chains. Diagnostic plots for other models can be found in Additional file 1

Zubair, A., Rosen, I. G., Nuzdhn, S. V., & Marjoram, P. (2019). Bayesian model selection for the Drosophila gap gene network. BMC Bioinformatics, 20(1), 327.

Model fit

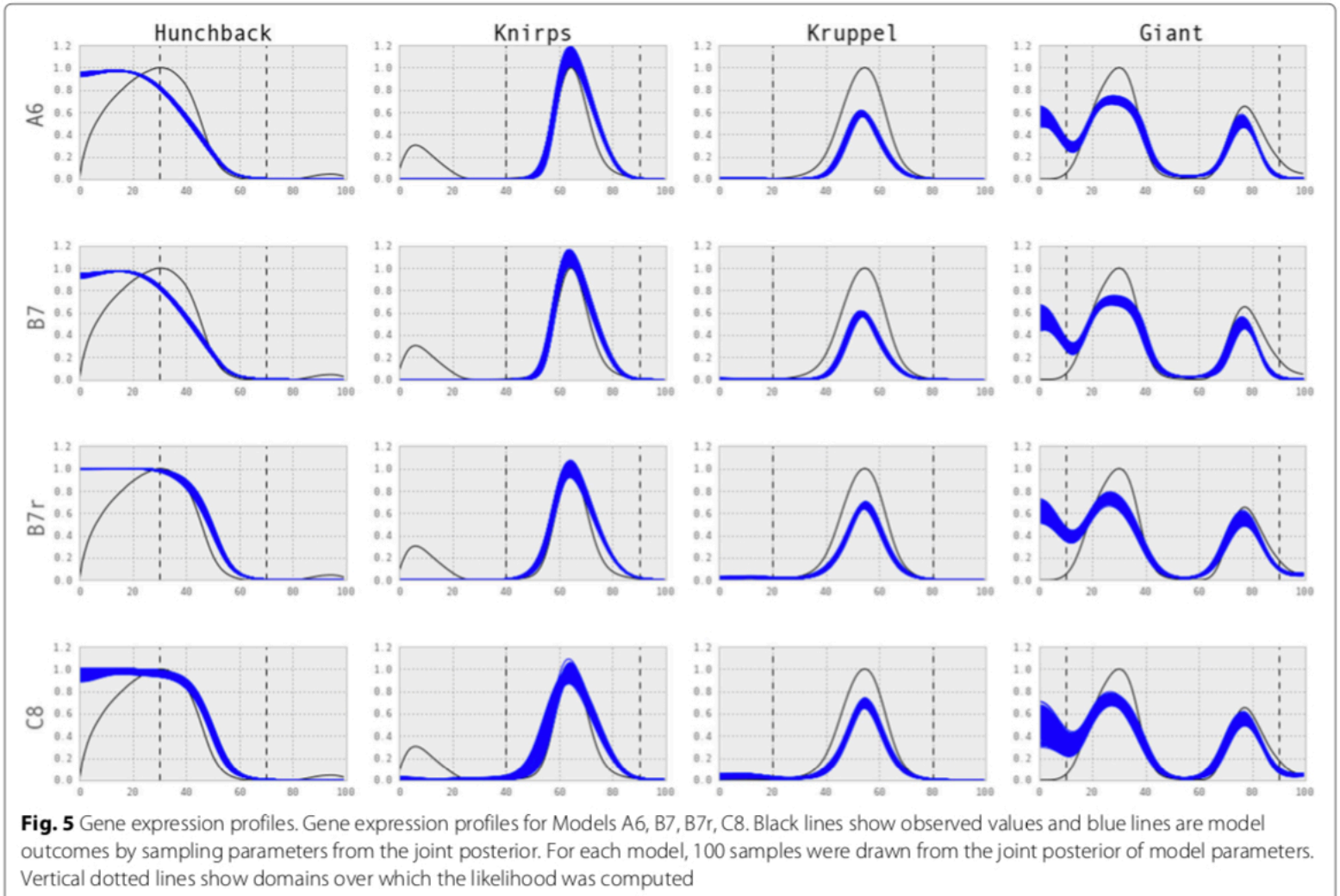


Fig. 5 Gene expression profiles. Gene expression profiles for Models A6, B7, B7r, C8. Black lines show observed values and blue lines are model outcomes by sampling parameters from the joint posterior. For each model, 100 samples were drawn from the joint posterior of model parameters. Vertical dotted lines show domains over which the likelihood was computed

Zubair, A., Rosen, I. G., Nuzdhn, S. V., & Marjoram, P. (2019). Bayesian model selection for the Drosophila gap gene network. BMC Bioinformatics, 20(1), 327.

Other MCMC variants/tricks

Transition kernel: q(.)

- We often augment the state-space in order to improve mixing
- e.g.
  - 1. The temperature in Parallel Tempered MCMC.
  - 2. The Wilson and Balding paper from Lecture 7.
- Modify Hastings Ratio a transition kernel to include all parts of the augmented statespace Θ.

$$h = \min\left\{1, \frac{[P(D|\Theta')\pi(\Theta')/f(D)]q(\Theta' \rightarrow \Theta)}{[P(D|\Theta)\pi(\Theta)/f(D)]q(\Theta \rightarrow \Theta')}\right\}$$

The independence sampler

1. If at Θ, propose move to Θ′ according to transition kernel q(Θ -> Θ′)=π(Θ′), (the prior for Θ - but we could use any distribution/function).
2. Calculate

$$
\begin{aligned}
h &= \min\left\{1, \frac{P(D|\Theta')\pi(\Theta')q(\Theta' \rightarrow \Theta)}{P(D|\Theta)\pi(\Theta)q(\Theta \rightarrow \Theta')}\right\} \\
&= \min\left\{1, \frac{P(D|\Theta')\pi(\Theta')\pi(\Theta)}{P(D|\Theta)\pi(\Theta)\pi(\Theta')}\right\} \\
&= \min\left\{1, \frac{P(D|\Theta')}{P(D|\Theta)}\right\}
\end{aligned}
$$

3. Move to Θ′ with prob. h, else remain at Θ.
4. Return to 1.

[Computationally nice, but may mix poorly]

Using two transition kernels

- Use two kernels, q_1 and q_2. Typically, one might have a high variance and the other will have a lower variance.
- e.g.,
  - $q_1(\theta)=N(\theta,\sigma_1^2)$
  - $q_2(\theta)=N(\theta,\sigma_2^2)$

  where $\sigma_2^2 >> \sigma_1^2$

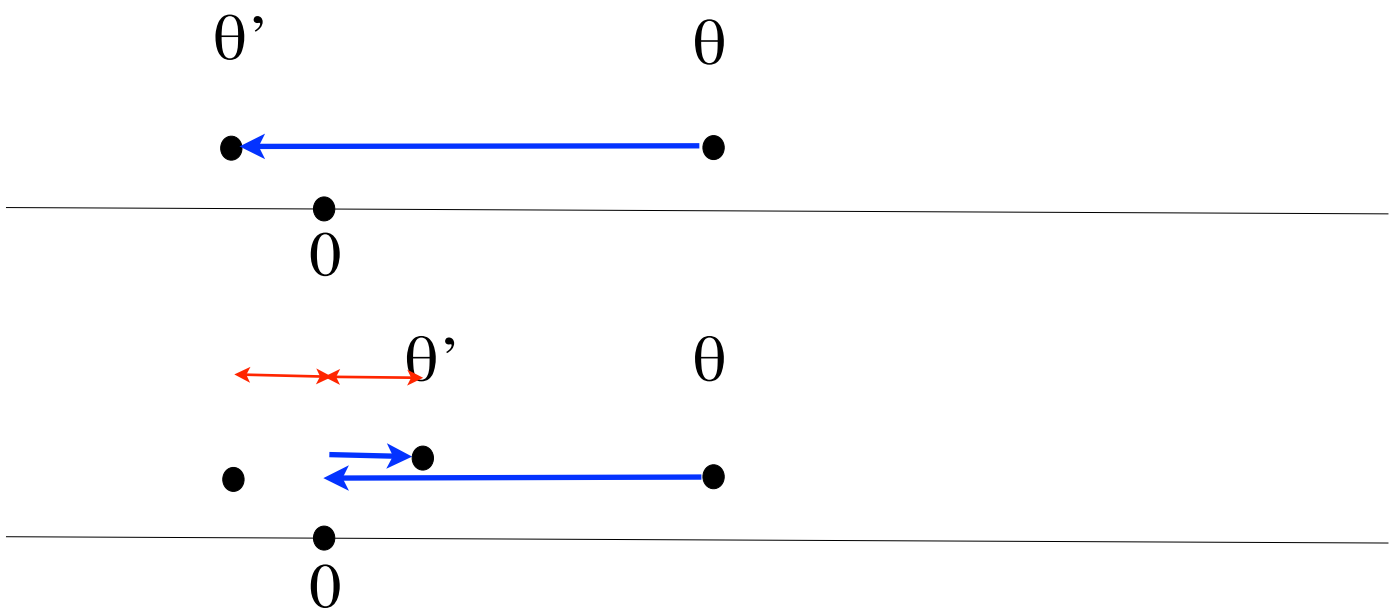
  At any step of the algorithm, choose q_1 with prob. p; choose q_1 otherwise

The rationale is that one kernel is good at making big jumps, while the other is good for making small jumps.

Note that when calculating $q(\theta->\theta')$ and $q(\theta'->\theta)$ for the Hastings ratio you must allow for the possibility of having used either q_1 or q_2.

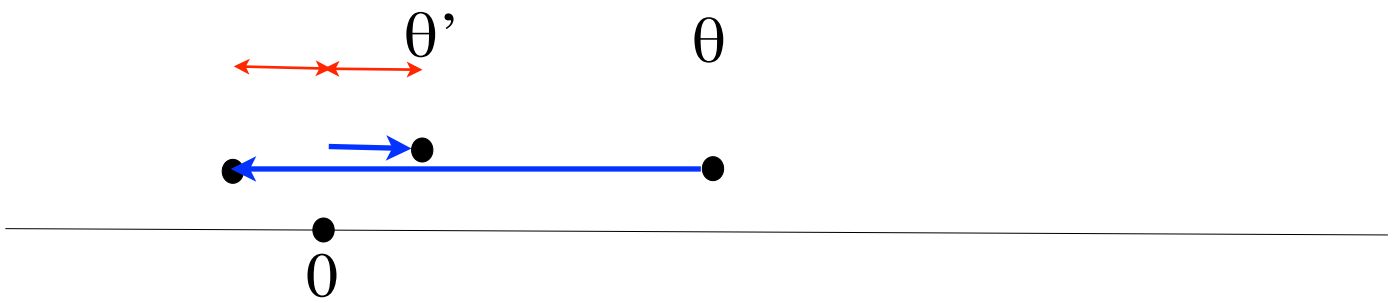
MCMC: Reflecting boundaries

- Suppose parameter θ must be positive
- Imagine we use θ' = θ + Normal(0,σ^2) as our proposal kernel q()
- If proposed θ' is < 0, reflect it at 0



Reflecting boundaries

- If proposed θ’ is < 0, reflect it at 0



- So, if θ’<0, redefine θ’ to be θ’’= -θ’.

- Result: if your proposal kernel is symmetric, as it is here, then it will still be the case that q(θ-> θ’)=q(θ’-> θ) for any θ,θ’.

Reversible jump MCMC

- Allows for situations in which the dimension of the parameter space changes (e.g. gene regulatory network analysis in which the size of the network can change, …).
- In order to deal with the change of dimensionality you have to add ugly Jacobian matrix calculations.
- Be very careful with these things!

MCMC: Metropolis-Hastings

Example: model selection

Suppose we are choosing between a model with one parameter θ_1, *or two parameters* θ_1, θ_2.

1. If at θ, propose move to θ' according to transition kernel q(θ -> θ'). When $\theta=\theta_1$ *and* $\theta'=\theta'_1,\theta_2$ we get
2. Calculate

$$h = \min\left\{1, \frac{P(D \mid \theta_1, \theta_2)\pi(\theta_1, \theta_2)q(\theta_1, \theta_2 \rightarrow \theta)}{P(D \mid \theta)\pi(\theta)q(\theta \rightarrow \theta_1, \theta_2)}\mathrm{Jac(q())}\right\}$$

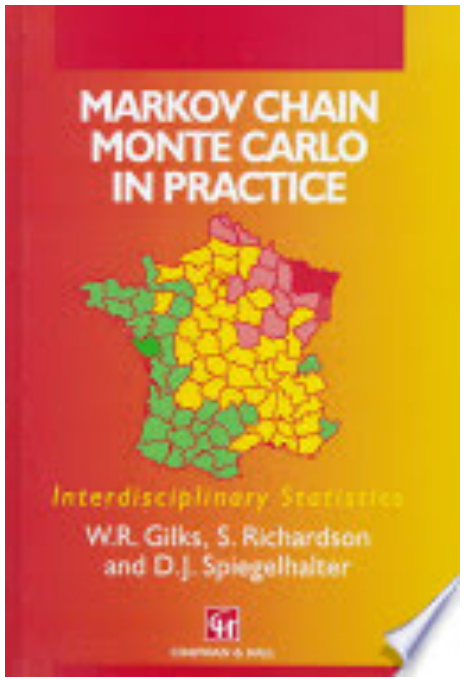
3. Move to θ' with prob. h, else remain at θ.
4. Return to 1.

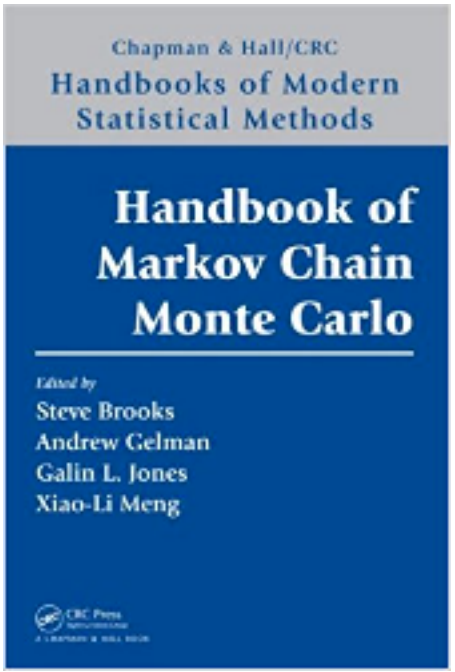
Similar thing when we go from $\theta'=\theta_1,\theta_2$ to $\theta=\theta_1$

Further reading:

Robert and Casella “Monte Carlo Statistical Methods” (Springer) for more on this and other MC methods. (A hard read!)

Andrieu, C., De Freitas, N., Doucet, A., & Jordan, M. I. (2002). An Introduction to MCMC for Machine Learning. *Machine Learning*, *50*, 5–43.





Suggested exercises: non-examinable

1. Adapt the example from the MCMC1.Rmd document to run several MCMC chains for that problem and check convergence using the 'coda' package.

2. Use the metrop function to implement any of the MCMC exercises we've seen so far in this course. For example, last week's multivariate normal example.

3. Try re-writing some of the exercises to use adaptive MCMC or tempering. It generally ends up working much better, but at the price of more work.

4. Would parallel tempering work for the gamma rv question in Assignment 3?

END