CS-E4895 Gaussian Processes
Lecture 4: Integration and model selection

Aki Vehtari

Aalto University

7.3.2023

Outline

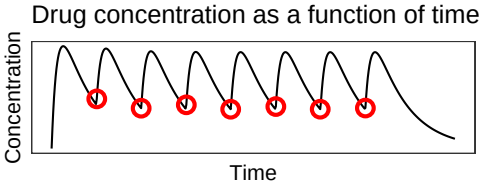
Gaussian processes – integration and model selection

- Background
- Relation to other lectures
- Point estimate vs. integration
  - motorcycle crash g-forces
- Using GPs as components
  - motorcycle crash g-forces
  - birthdays
- Model selection

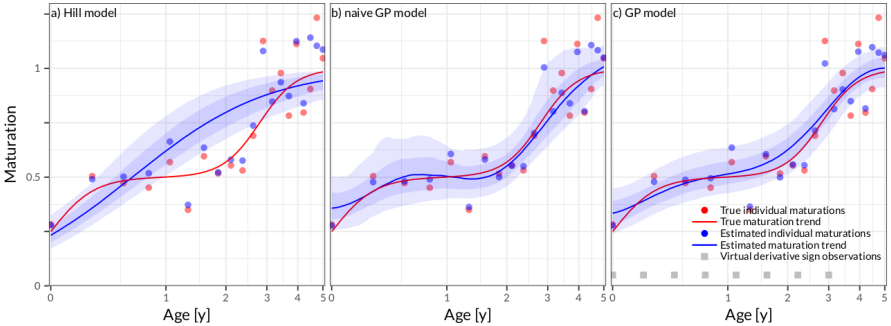
How I started working on GPs

$\rightarrow$

GPs as priors for model components

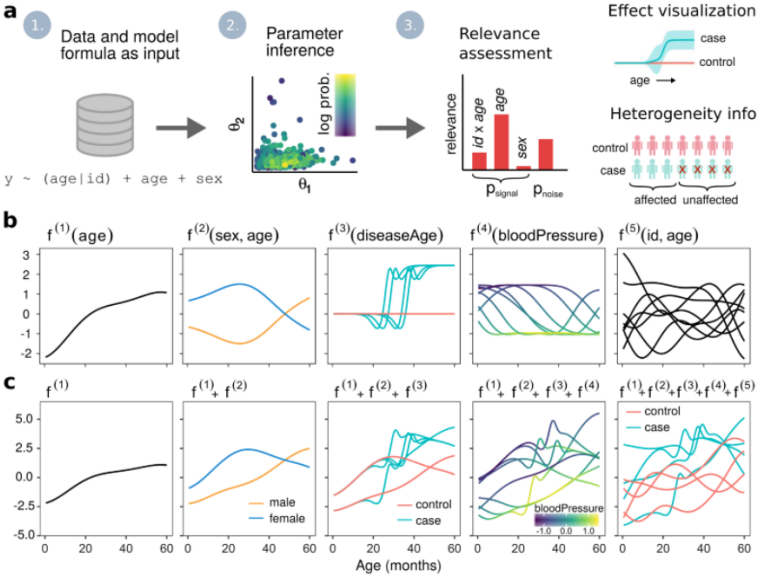


Monotonic maturation effect



lgpr – longitudinal Gaussian process regression

R package for **L**ongitudinal **G**aussian **P**rocess **R**egression.



GPs and Bayesian inference

- Lecture 3 & GPML Chapter 2:
  - Posterior $p(f_* \mid \boldsymbol{y})$
  - Posterior predictive $p(y_* \mid \boldsymbol{y}) = \int p(y_* \mid f_*) p(f_* \mid \boldsymbol{y}) \mathrm{d}f_*$
- Lecture 3 & GPML Chapter 5:
  - Marginal likelihood given covariance function parameters (hyperparameters) $p(\boldsymbol{y} \mid \boldsymbol{\theta}) = \int p(\boldsymbol{y} \mid \boldsymbol{f}) p(\boldsymbol{f} \mid \boldsymbol{\theta}) \mathrm{d}\boldsymbol{f}$
  - Uniform prior & optimize $\hat{\theta} = \arg\max_{\boldsymbol{\theta}} \ln p(\boldsymbol{y} \mid \boldsymbol{\theta})$
  - Predictive distribution $p(y_* \mid \boldsymbol{y}, \hat{\theta})$
- Lecture 6 & GPML Chapter 3:
  - Approximate marginal likelihood when $p(\boldsymbol{y} \mid \boldsymbol{f})$ is non-Gaussian

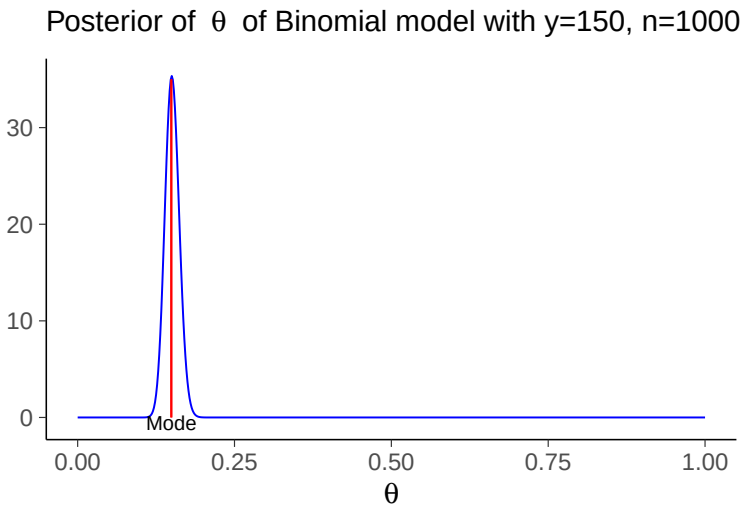
This lecture

- Point estimates work sometimes, but integration is better
- Laplace and VI are useful, but can fail catastrophically
- Markov Chain Monte Carlo (MCMC) is often very accurate, but slower
- Examples
  - Motorcycle crash g-forces
  - Birthdays

Sometimes point estimate is fine

$p(y_* \mid \boldsymbol{y}, \hat{\theta})$ vs. $p(y_* \mid \boldsymbol{y}) = \int p(y_* \mid \theta) p(\theta \mid y) d\theta$

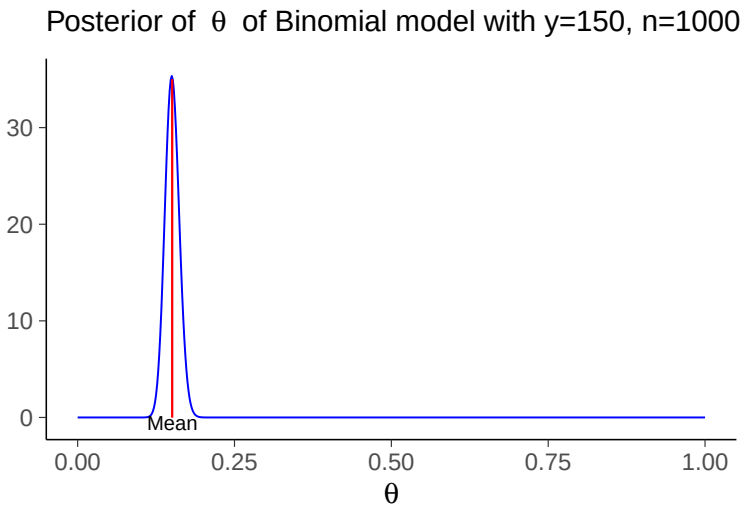
Binomial example: $n = 150, y = 1000$



Sometimes point estimate is fine

$p(y_* \mid \boldsymbol{y}, \hat{\theta})$ vs. $p(y_* \mid \boldsymbol{y}) = \int p(y_* \mid \theta) p(\theta \mid y) d\theta$

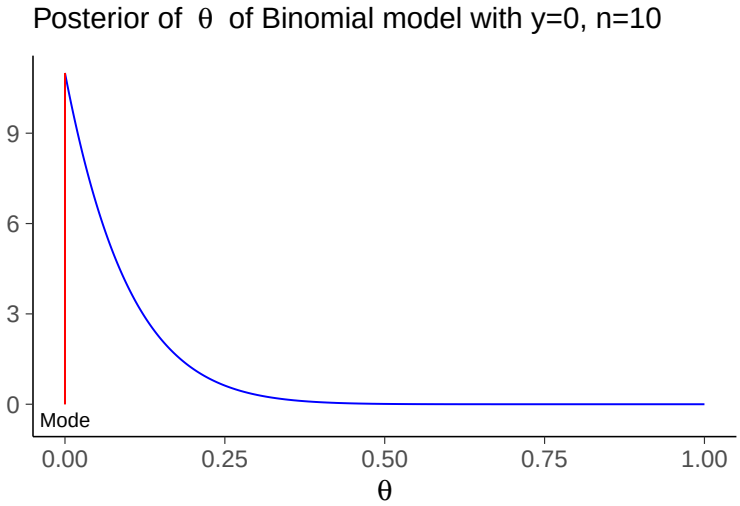
Binomial example: $n = 150, y = 1000$



Benefits of integration

$p(y_* \mid \boldsymbol{y}, \hat{\theta})$ vs. $p(y_* \mid \boldsymbol{y}) = \int p(y_* \mid \theta) p(\theta \mid y) d\theta$

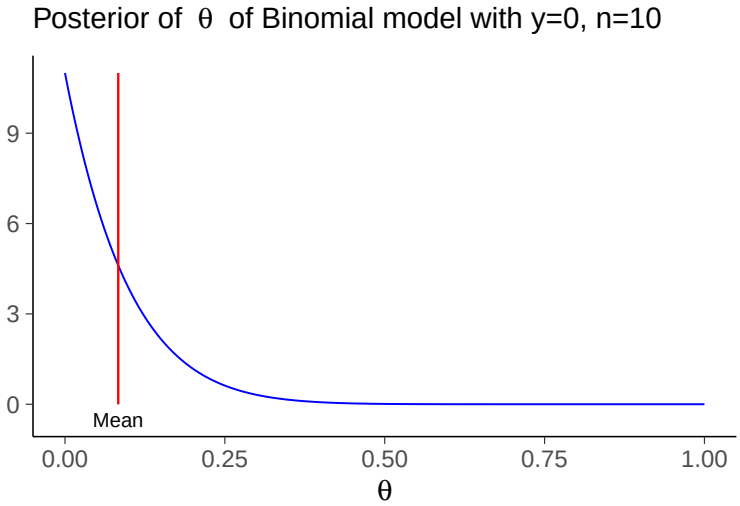
Binomial example: $n = 0, y = 10$



Benefits of integration

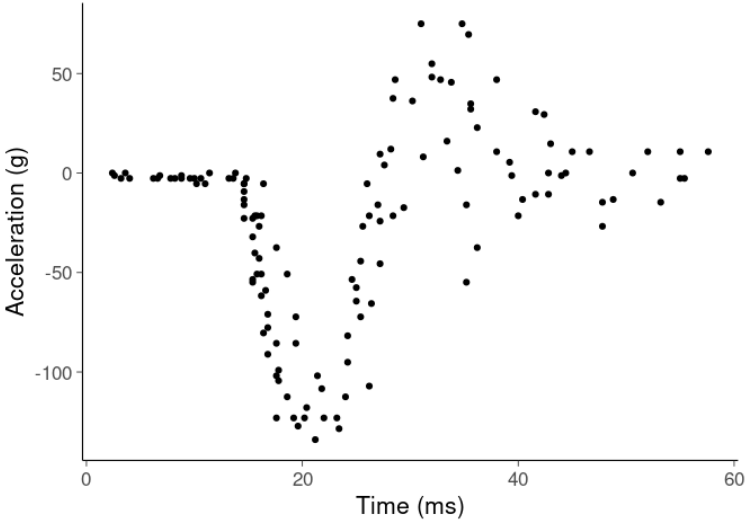
$p(y_* \mid \boldsymbol{y}, \hat{\theta})$ vs. $p(y_* \mid \boldsymbol{y}) = \int p(y_* \mid \theta) p(\theta \mid y) d\theta$

Binomial example: $n = 0, y = 10$



Motorcycle crash g-forces

Data are measurements of head acceleration in a simulated motorcycle accident, used to test crash helmets. https://avehtari.github.io/casestudies/Motorcycle/motorcycle_gpcourse.html



Motorcycle crash g-forces

1) normal distribution having Gaussian process prior on mean:

$$
\begin{aligned}
y &\sim \mathsf{normal}(f(x), \sigma) \\
f &\sim \mathsf{GP}(0, K_1) \\
\sigma &\sim \mathsf{normal}^+(0, 1)
\end{aligned}
$$

2) normal distribution having Gaussian process prior on mean and log standard deviation:

$$
\begin{aligned}
y &\sim \mathsf{normal}(f(x), \exp(g(x))) \\
f &\sim \mathsf{GP}(0, K_1) \\
g &\sim \mathsf{GP}(0, K_2)
\end{aligned}
$$

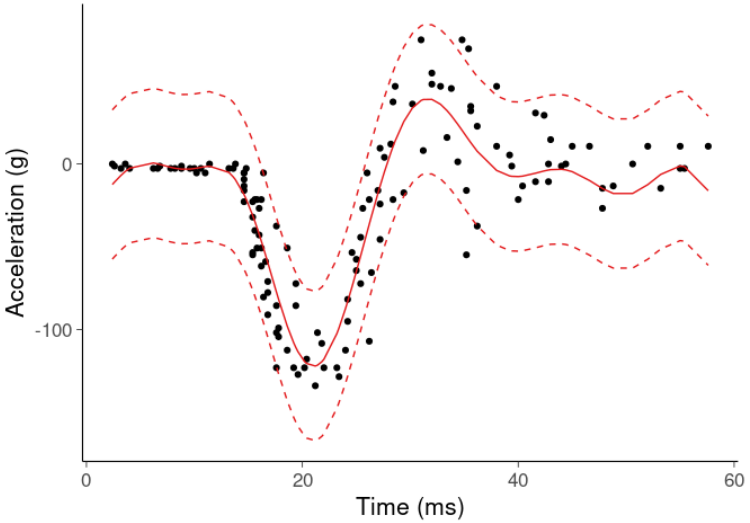
Homoskedastic GP

$$
\begin{aligned}
y &\sim \text{normal}(f(x), \sigma) \\
f &\sim \text{GP}(0, K_1(l_f, \sigma_f)) \\
\sigma &\sim \text{normal}^+(0, 1)
\end{aligned}
$$

possible to integrate analytically over f to obtain marginal likelihood for the covariance function parameters (l_f, σ_f) and residual scale σ (Lecture 3)

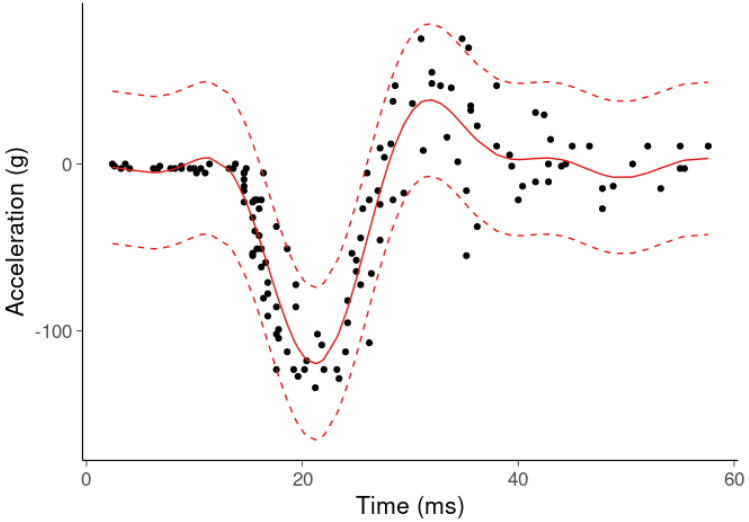
Homoskedastic GP – MAP

MAP for (l_f, σ_f, σ)



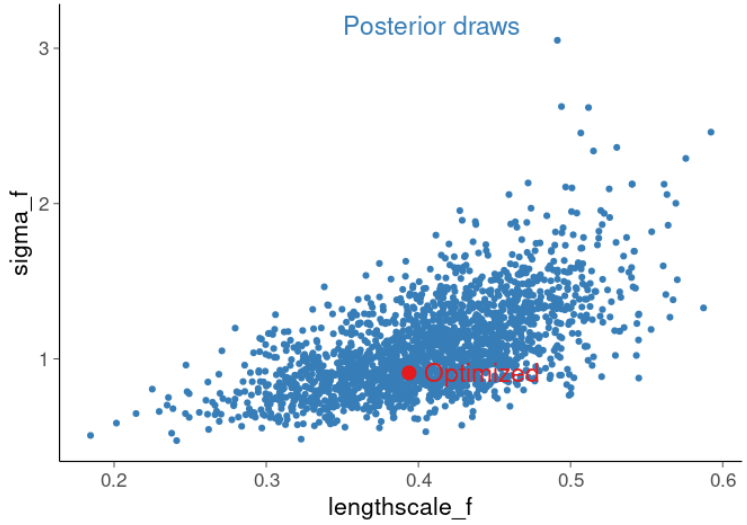
Homoskedastic GP – MCMC

MCMC integration over posterior of (l_f, σ_f, σ)



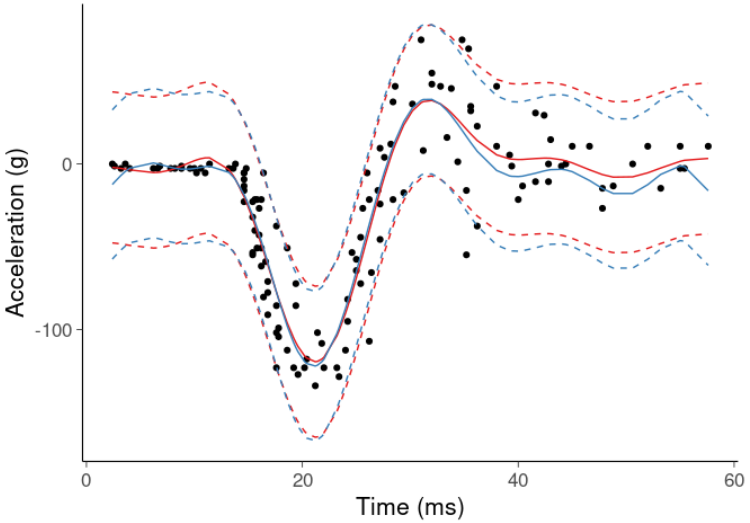
Homoskedastic GP – MAP vs. MCMC

MAP vs MCMC for (l_f, σ_f, σ)



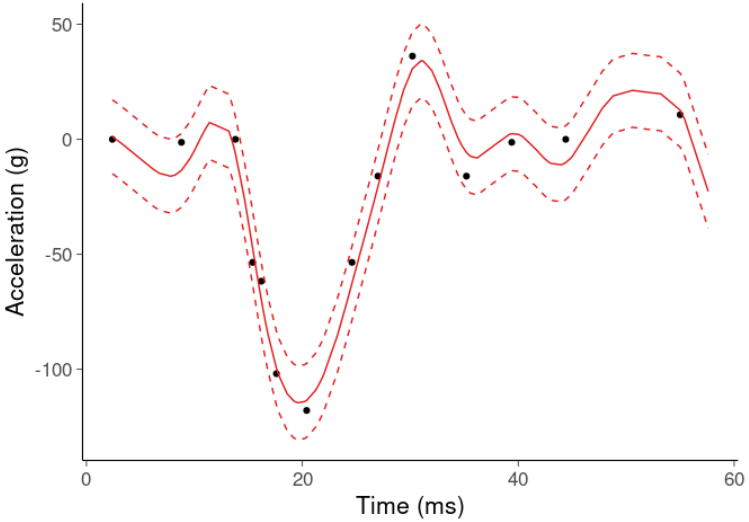
Homoskedastic GP – MAP vs. MCMC

MAP vs MCMC for (l_f, σ_f, σ)



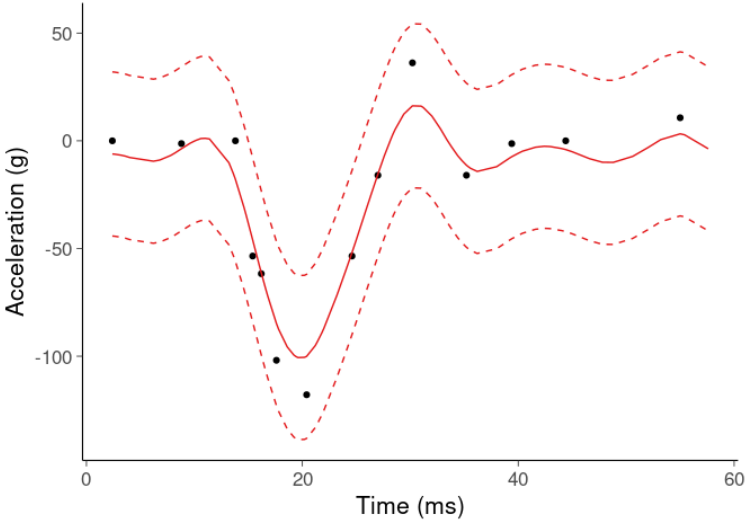
Homoskedastic GP – MAP with small data

MAP for (l_f, σ_f, σ)



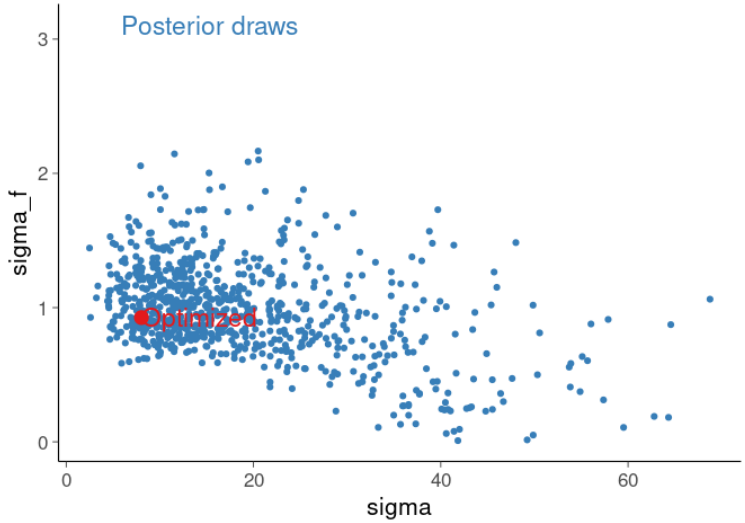
Homoskedastic GP – MCMC with small data

MCMC integration over posterior of (l_f, σ_f, σ)



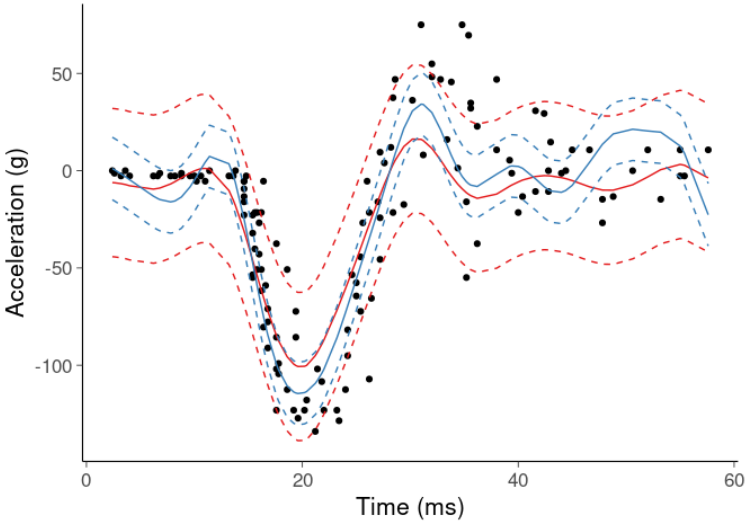
Homoskedastic GP – MAP vs. MCMC with small data

MAP vs MCMC for (l_f, σ_f, σ)



Homoskedastic GP – MAP vs. MCMC with small data

MAP vs MCMC for (l_f, σ_f, σ)



Heteroskedastic GP

$$y \sim \text{normal}(f(x), \exp(g(x))$$
$$f \sim \text{GP}(0, K_1(l_f, \sigma_f)$$
$$g \sim \text{GP}(0, K_2(l_g, \sigma_g))$$

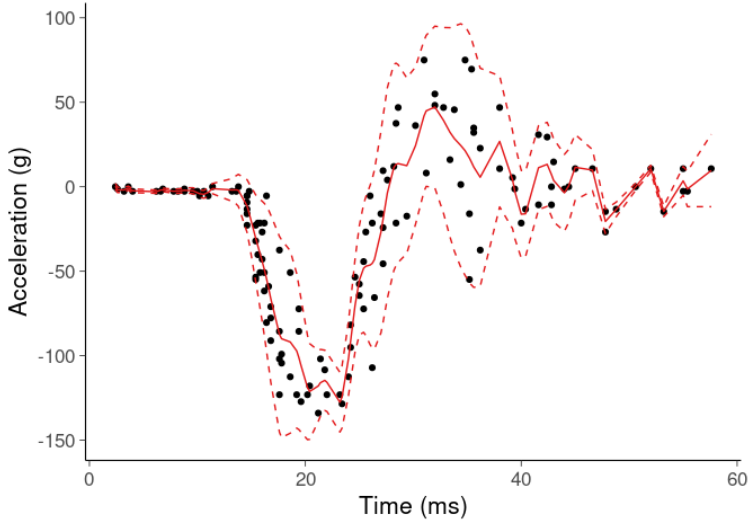
Not possible to integrate analytically over g
(cf. Lecture 2, not possible to present as a linear model wrt g)

Variants of Laplace, VI, and EP could be used to approximately integrate over f and g to get approximate marginal likelihood (a bit more complex than in case of classification, Lecture 5)

We can do the inference for all $(f, g, l_f, \sigma_f, l_g, \sigma_g, \sigma)$ jointly, but now the number of unknown parameters is bigger than the number of observations

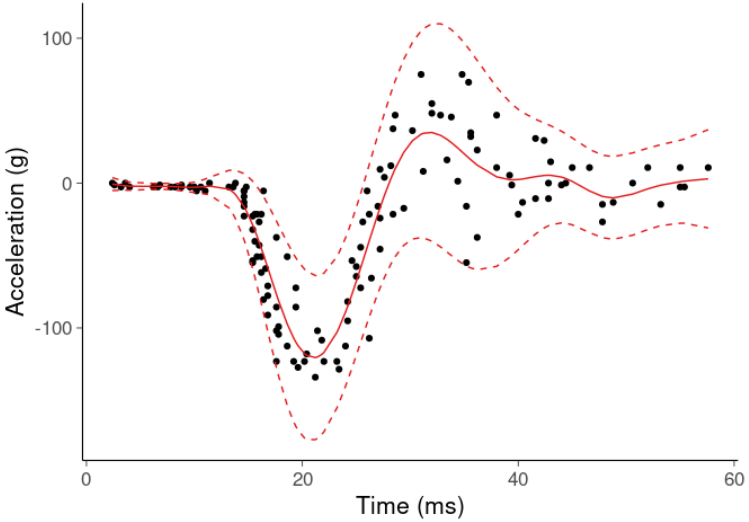
Heteroskedastic GP – MAP

MAP for $(f, g, l_f, \sigma_f, l_g, \sigma_g, \sigma)$



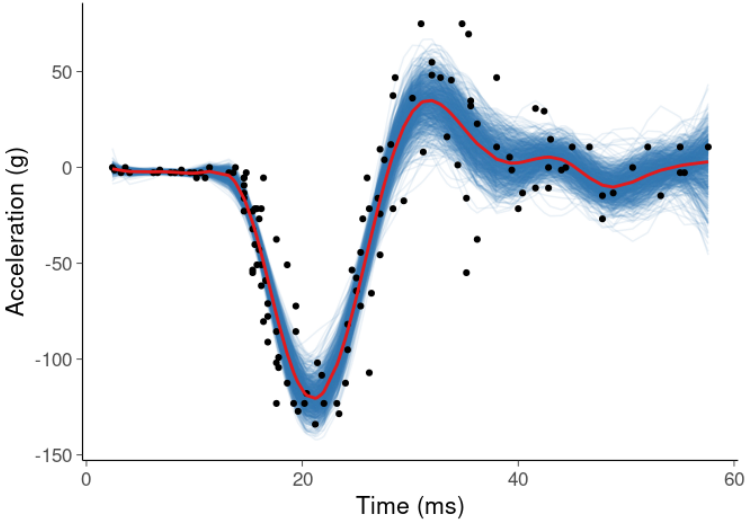
Heteroskedastic GP – MCMC

MCMC integration over posterior of $(f, g, l_f, \sigma_f, l_g, \sigma_g, \sigma)$



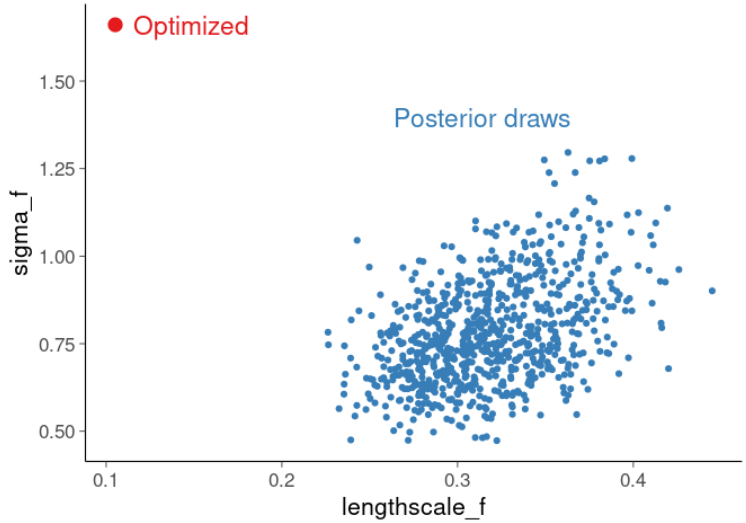
Heteroskedastic GP – MCMC

MCMC posterior posterior draws of f



Heteroskedastic GP – MAP vs. MCMC

MAP vs MCMC for (l_f, σ_f)

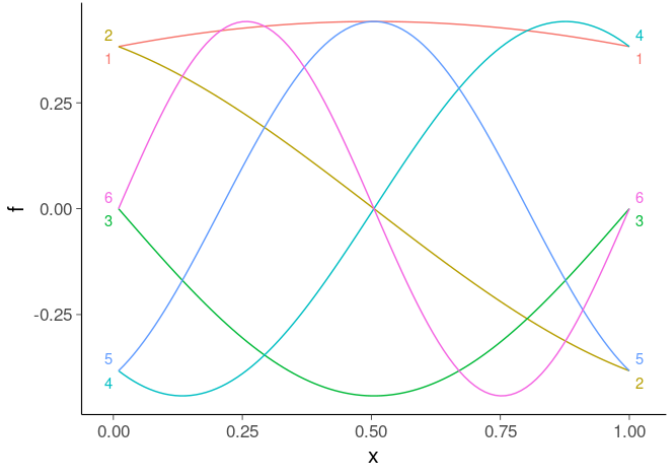


GP + MCMC

- MCMC usually requires many (log) posterior density evaluations
- When using MCMC for f and g, need to compute Cholesky of the covariance matrix, which scales as $O(n^3)$ for each density evaluation
  - use something else than MCMC or approximate GP (later lectures)

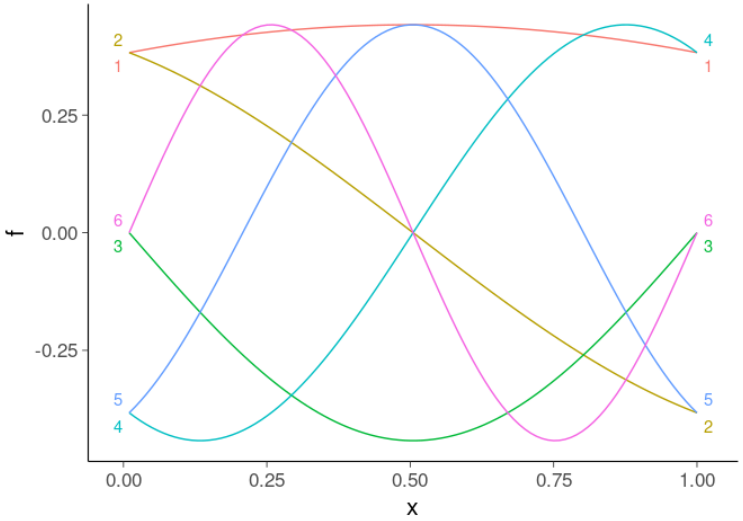
GP with Hilbert space basis functions

- Exact with infinite number of, but good enough with less than n basis functions
  - linear combination of basis functions (Lecture 2 Bayesian linear regression)
  - normal prior on coefficients
  - prior scales determined by the covariance function



GP with Hilbert space basis functions

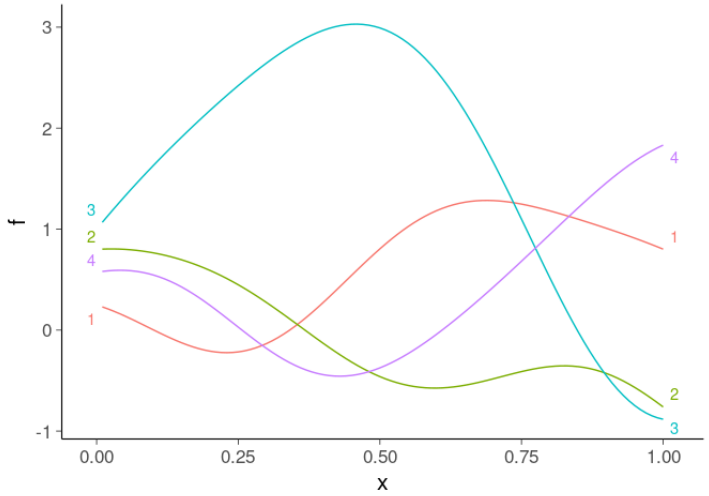
Basis functions



GP with Hilbert space basis functions

Prior draws from the GP with exponentiated quadratic cf, $l_f = 1$

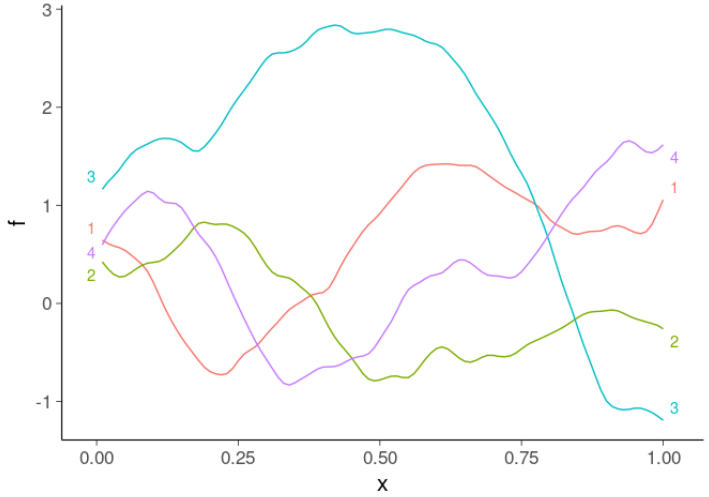
prior scales: 1.55 1.44 1.28 1.09 0.88 0.68 0.50 0.35 0.24 0.15 0.09 ...



GP with Hilbert space basis functions

Prior draws from the GP with Matérn-$3/2$ cf, $l_f = 1$

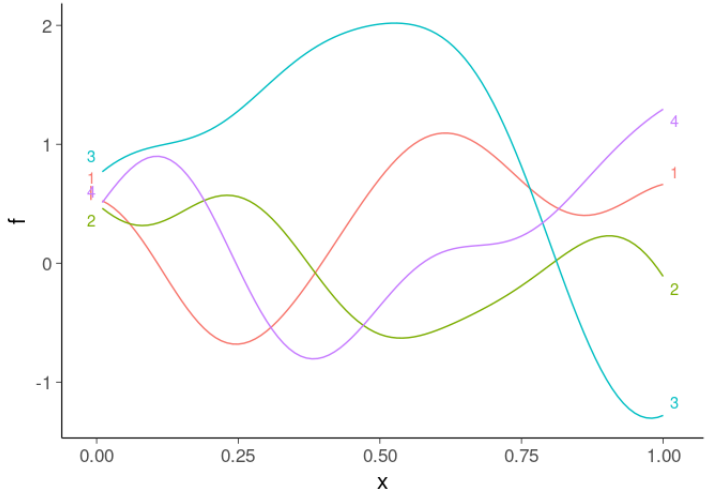
prior scales: 1.47 1.35 1.18 1.01 0.85 0.71 0.60 0.51 0.43 0.37 0.32 ...



GP with Hilbert space basis functions

Prior draws from the GP with exponentiated quadratic cf, $l_f = 0.3$

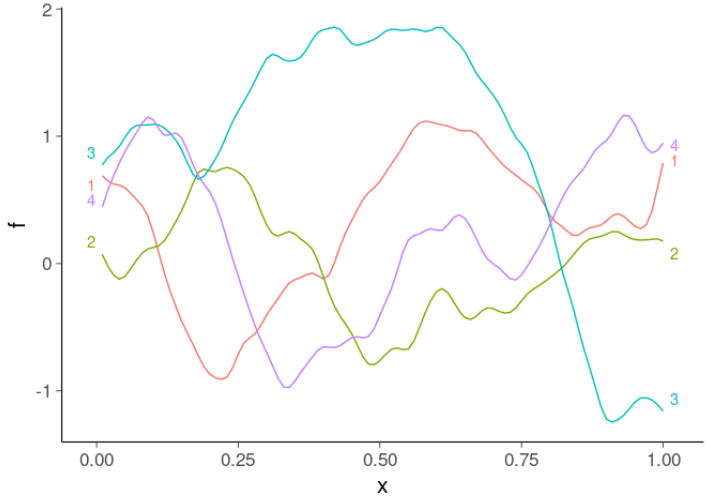
prior scales: 0.86 0.84 0.80 0.76 0.70 0.64 0.57 0.50 0.44 0.37 0.31 ...



GP with Hilbert space basis functions

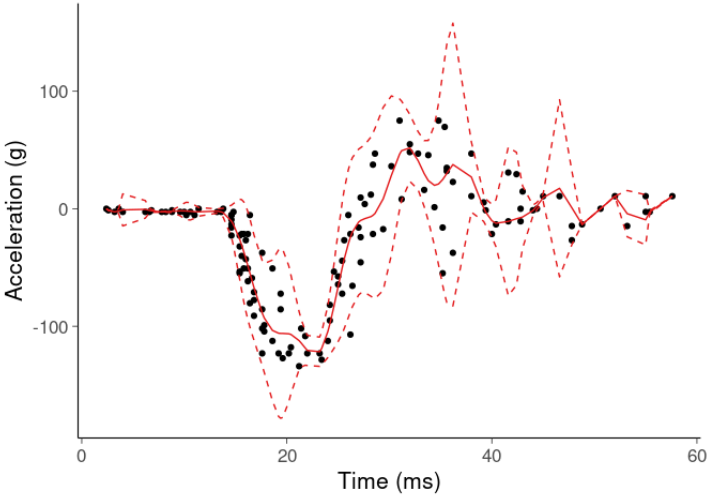
Prior draws from the GP with Matérn-$3/2$ cf, $l_f = 0.3$

prior scales: 0.82 0.80 0.76 0.70 0.65 0.59 0.54 0.48 0.43 0.39 0.35 ...



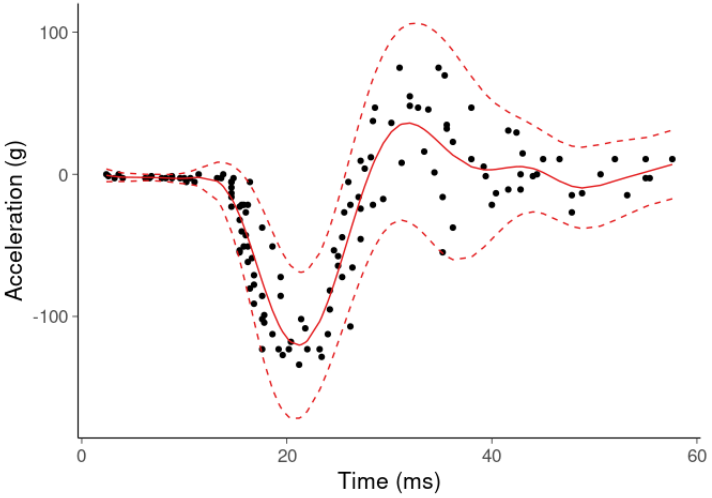
Heteroskedastic HS-GP – MAP

MAP for $(\beta_f, \beta_g, l_f, \sigma_f, l_g, \sigma_g, \sigma)$



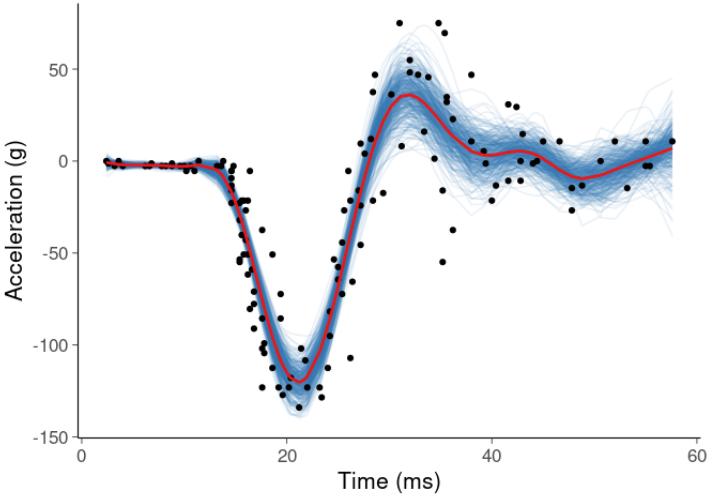
Heteroskedastic HS-GP – MCMC

MCMC integration over posterior of $(\beta_f, \beta_g, l_f, \sigma_f, l_g, \sigma_g, \sigma)$



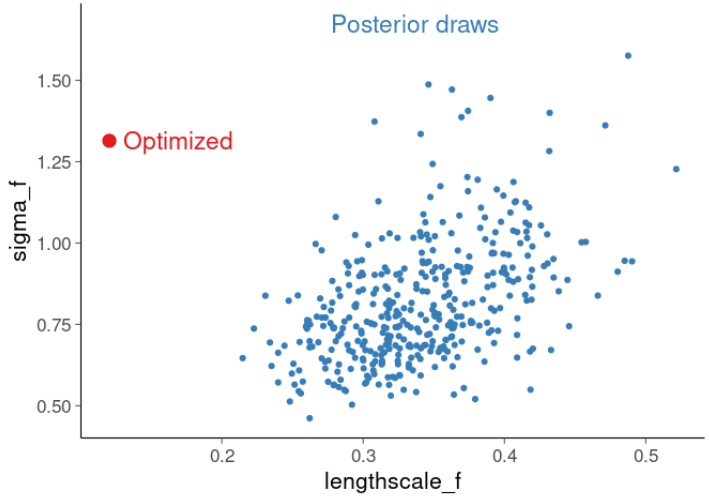
Heteroskedastic HS-GP – MCMC

MCMC posterior posterior draws of f



Heteroskedastic HS-GP – MAP vs. MCMC

MAP vs MCMC for (l_f, σ_f)



Heteroskedastic HS-GP – MCMC

- Hilbert space basis function approach is much faster
  - with my laptop in this example 160 times faster
  - no practical difference in modeling accuracy

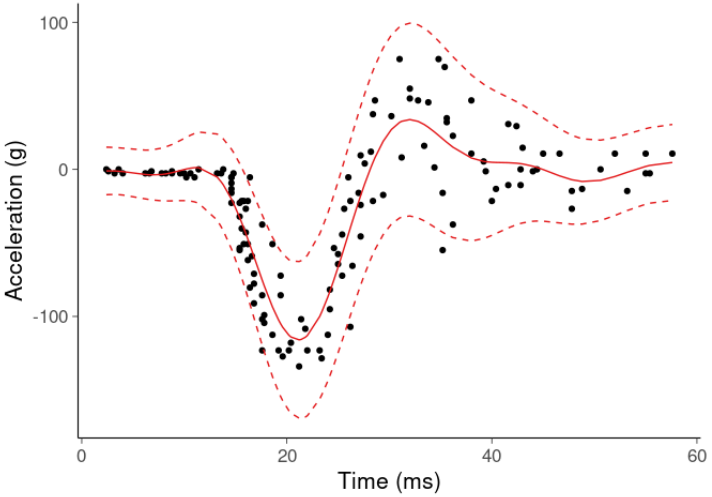
- Solin, and Särkkä (2020). Hilbert space methods for reduced-rank Gaussian process regression. Statistics and Computing 30(2):419–446. doi:10.1007/s11222-019-09886-w
- Riutort-Mayol, Bürkner, Andersen, Solin, and Vehtari (2023). Practical Hilbert space approximate Bayesian Gaussian processes for probabilistic programming. Statistics and Computing, 33(17):1-28. doi:10.1007/s11222-022-10167-2

Heteroskedastic HS-GP – Variational inference

- ML folklore says variational inference is fast, but...
  - you can only choose two from 1) fast, 2) black box, 3) accurate

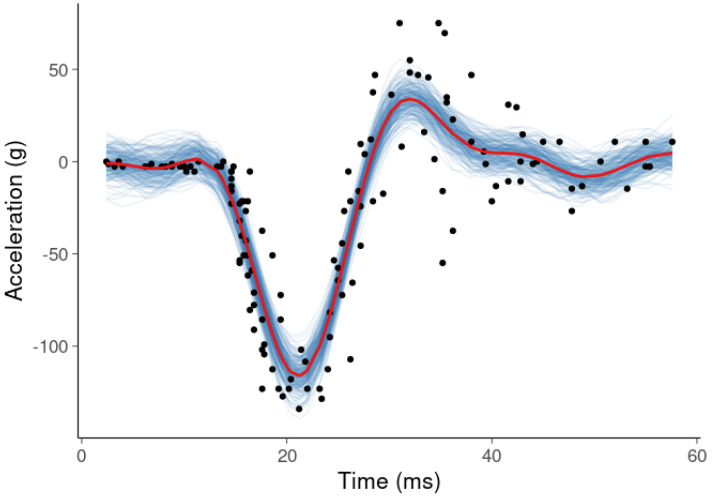
Heteroskedastic HS-GP – Variational inference

Black box autodiff variational inference (ADVI) with meanfield normal approximation



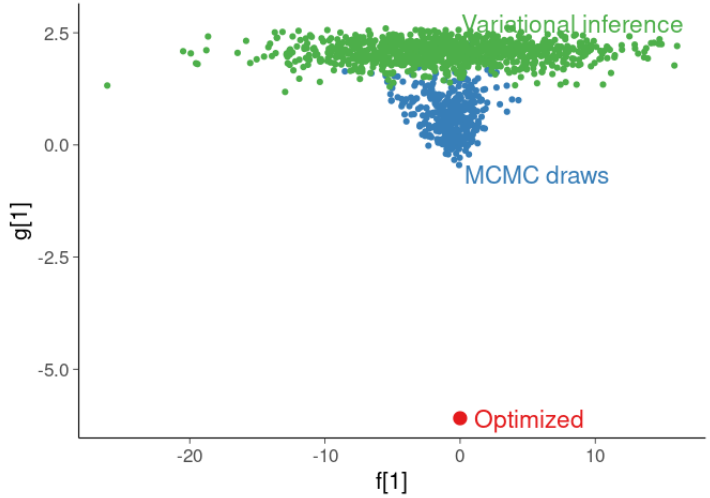
Heteroskedastic HS-GP – Variational inference

Black box autodiff variational inference (ADVI) with meanfield normal approximation



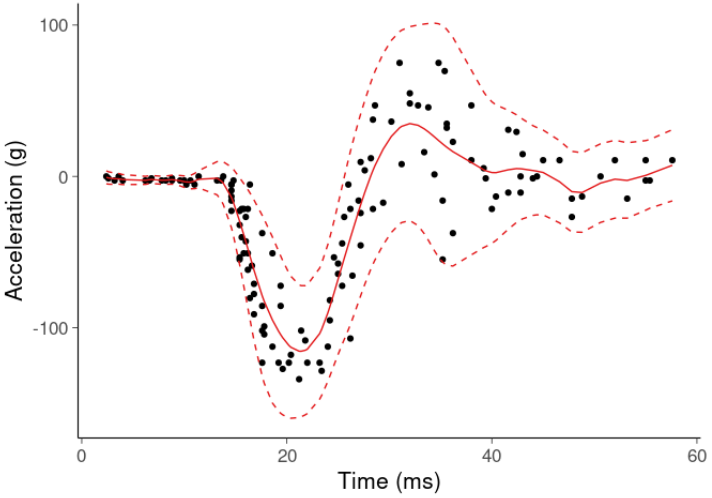
Heteroskedastic HS-GP – MAP vs MCMC vs ADVI

MAP vs MCMC vs ADVI



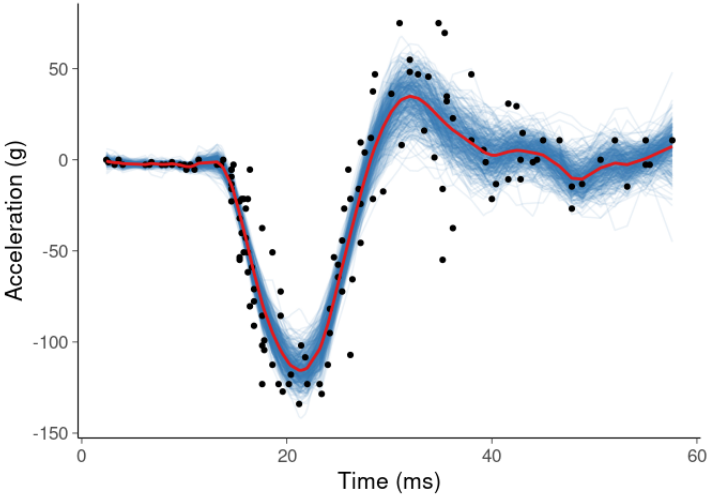
Heteroskedastic HS-GP Matérn-3/2 – MCMC

MCMC integration over posterior of $(\beta_f, \beta_g, l_f, \sigma_f, l_g, \sigma_g, \sigma)$



Heteroskedastic HS-GP Matérn-3/2 – MCMC

MCMC posterior posterior draws of f



Summary

- Optimization / MAP is fast and works sometimes
  - more likely to work with big data
- Laplace is sometimes good for integrating out f (and g)
- Variational inference is sometimes good for integrating out f (and g)
- MCMC often can integrate over everything, but can be slow
- Integration is useful
- Uncertainty quantification is useful

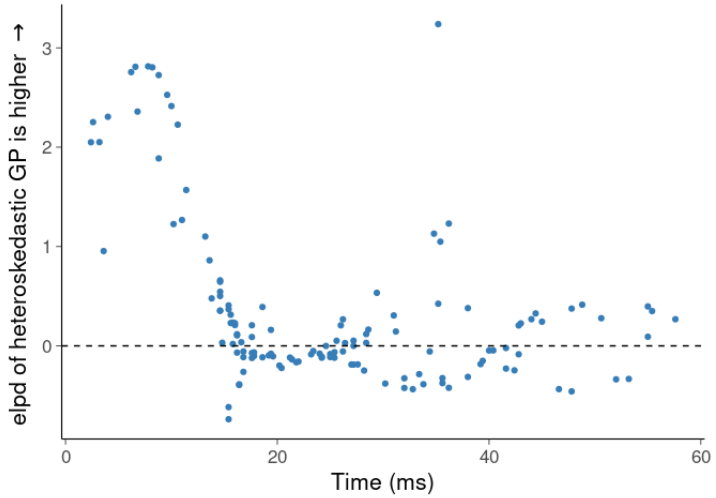
Model selection

- GPML chapter calls selecting $\hat{\theta}$ as model selection
- In statistics, model selection usually refers to selecting from discrete model space, e.g.
  - different observation models (e.g. homoskedastic vs heteroskedastic)
  - covariate selection
  - functional shapes (e.g. covariance function)
- When MAP or type II MAP (integrating out f) works, marginal likelihood or evidence may be useful for model selection
  - When more elaborate integration is needed, computation of marginal likelihood over θ is often difficult
- Cross-validation is easy (BDA Lecture 8)

- Vehtari, Gelman and Gabry (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. Statistics and Computing, 27(5):1413–1432. doi:10.1007/s11222-016-9696-4.
- Vehtari, Mononen, Tolvanen, Sivula and Winther (2016). Bayesian leave-one-out cross-validation approximations for Gaussian latent variable models. Journal of Machine Learning Research, 17(103):1–38. https://jmlr.org/papers/v17/14-540.html

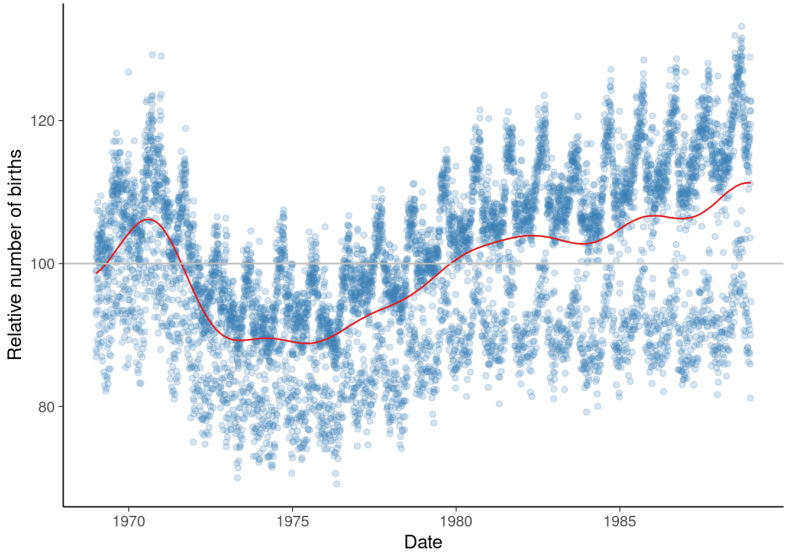
Model selection with LOO-CV

	elpd_diff	se_diff
heteroskedastic	0.0	0.0
homoskedastic	−49.1	9.9



Birthdays

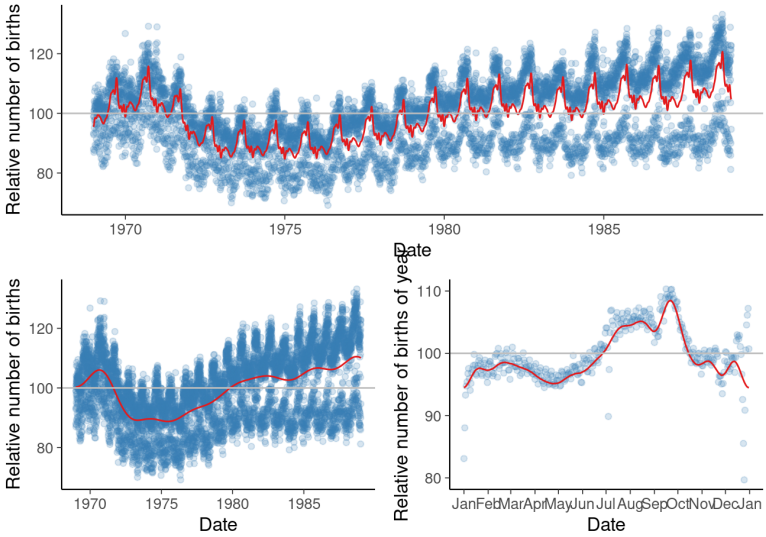
Model 1: Slow trend



https://avehtari.github.io/casestudies/Birthdays/birthdays.html

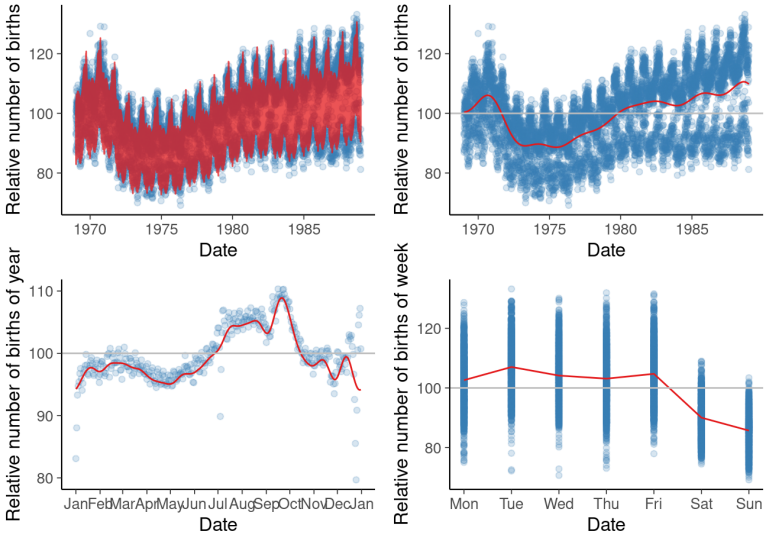
Birthdays

Model 2: Slow trend + yearly seasonal trend



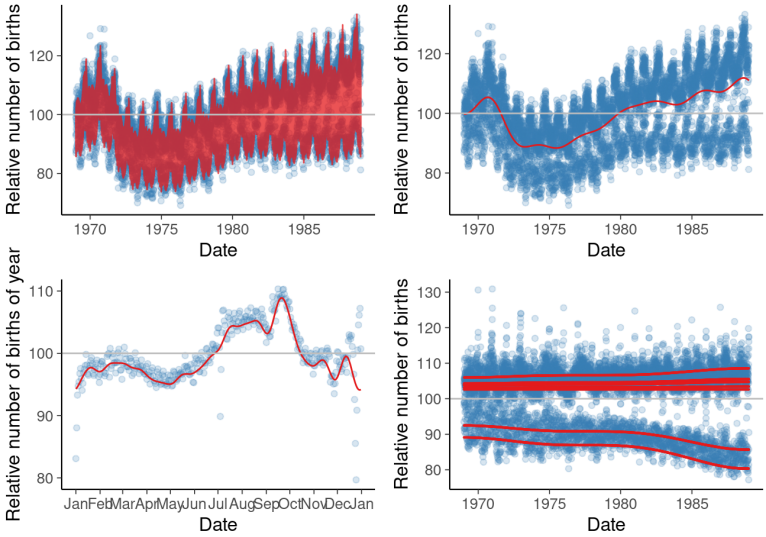
Birthdays

Model 3: Slow trend + yearly seasonal trend + day of week



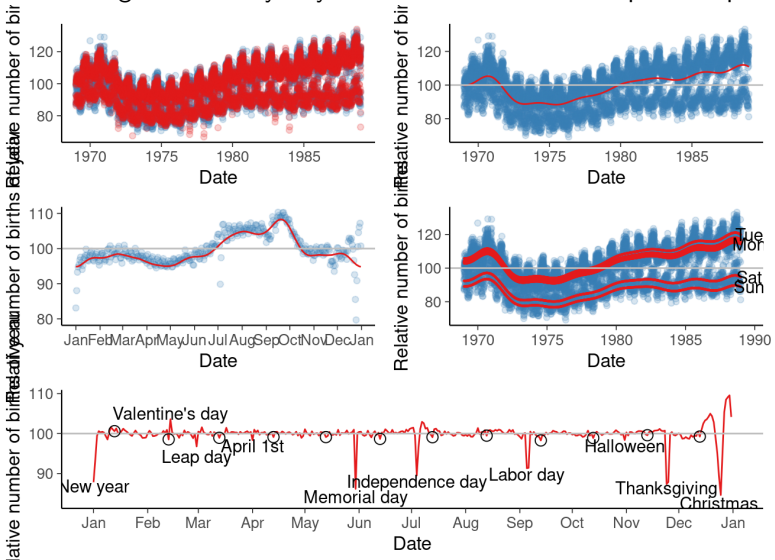
Birthdays

Model 4: long term smooth + seasonal + weekday with increasing magnitude



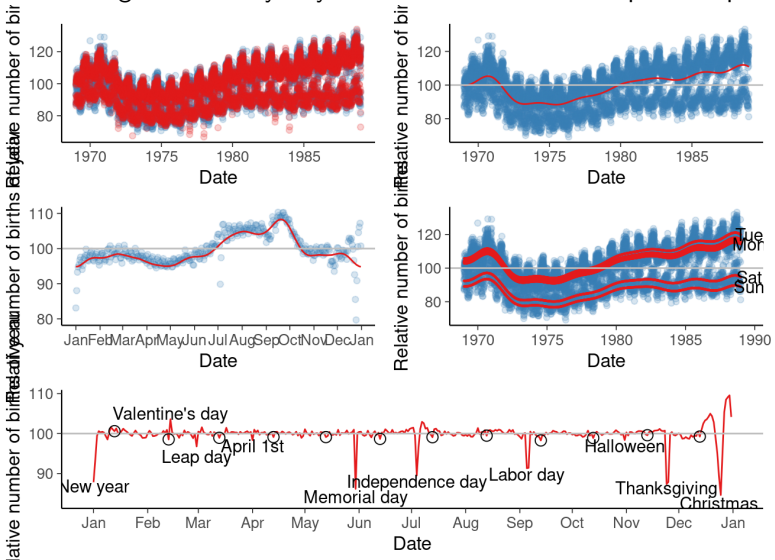
Birthdays

Model 8+t_nu: long term smooth + seasonal + weekday with time dependent magnitude + day of year effect with Student's t prior + special



Birthdays

Model 8+t_nu: long term smooth + seasonal + weekday with time dependent magnitude + day of year effect with Student's t prior + special



		elpd_diff	se_diff
Model	8t	0	0
Model	4	−1994	129
Model	3	−2479	115
Model	2	−8489	102
Model	1	−9033	103

Challenges in building widely applicable GP software

- The full joint posterior has difficult geometry
  - MCMC is likely to be slow
  - distributional approximations are likely to be bad
- The conditional distribution for latent values is easier
  - integrate out the latent variables using approximations
  - Laplace, EP, variational
  - if big data, maximizing marginal likelihood is OK

Flexibility

- Different observation models
  - exponential family easy
  - non-exponential family varyingly difficult
  - observation models depending on multiple latent values
  - observation models depending on multiple observations
  - censored data
  - multioutput
  - derivative observations

Flexibility vs complexity

- Combinatorial explosion if all features need to work together
  - approximate computation related to covariance matrix
  - approximate integration (latent or joint)
  - different observation models
  - different priors
  - combine with other models like ODEs

Flexibility vs speed

- How about Turing complete probabilistic programming language, autodiff and automatic inference?
- Speed in autodiff systems is not automatic!
  - what is a node in autodiff?
  - forward, reverse, mixed, adjoints, etc.
- Inference speed depends on
  - computational cost of single (marginal) log density
  - difficult posterior geometries require more (marginal) log density evaluations
  - integration vs maximizing marginal likelihood

Conclusion

- Very unlikely that one software would be best for everything
- Tradeoff between flexibility, speed, and additional implementation effort
- Prediction: There will be improvements in modularity and interoperability