REGLERTEKNIK

AUTOMATIC CONTROL

Linköpings universitet

# Gaussian process optimisation for approximate Bayesian inference

IDA ML seminar series, Linköping University.

Johan Dahlin
johan.dahlin@liu.se

Division of Automatic Control,
Department of Electrical Engineering,
Linköping University

April 1, 2015

- This is ongoing/current collaborative work together with
  - Dr. Fredrik Lindsten (University of Cambridge, United Kingdom)
  - Prof. Thomas Schön (Uppsala University, Sweden)
  - Prof. Mattias Villani (Linköping University, Sweden)

# Summary

- Aim
  - Efficient approximate Bayesian parameter inference in nonlinear SSMs.
  - Extending this to SSMs with intractable likelihoods.
  - Inference in copula models with $\alpha$-stable marginals.
- Methods
  - Gaussian process optimisation.
  - Sequential Monte Carlo methods.
  - Approximate Bayesian computations.
- Contributions
  - Decreased computational cost compared with popular methods.
  - Interesting method for solving other costly optimisation problems.

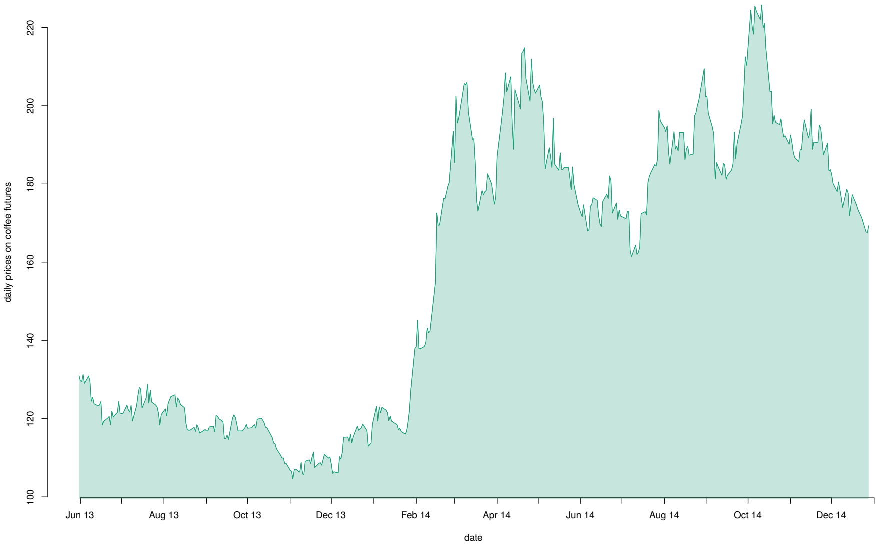
# Motivating example: Risk management

- Problem: modelling the risk in a portfolio of financial assets.
- Usually models the log-returns

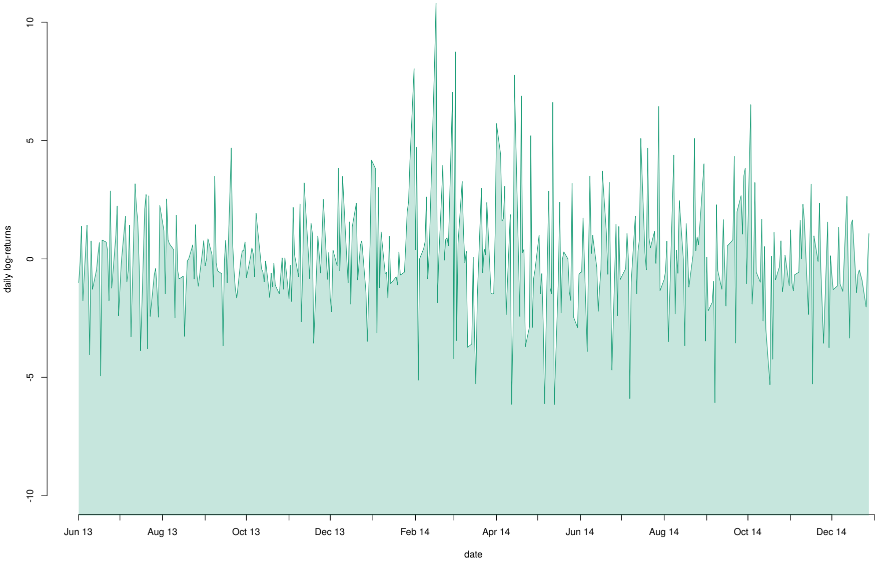
$$y_t = \log(s_t - s_{t-1}),$$

  of some asset price $s_t$.
- Copula model (dependency structure and models of margins).
- Margins modelled using stochastic volatility.

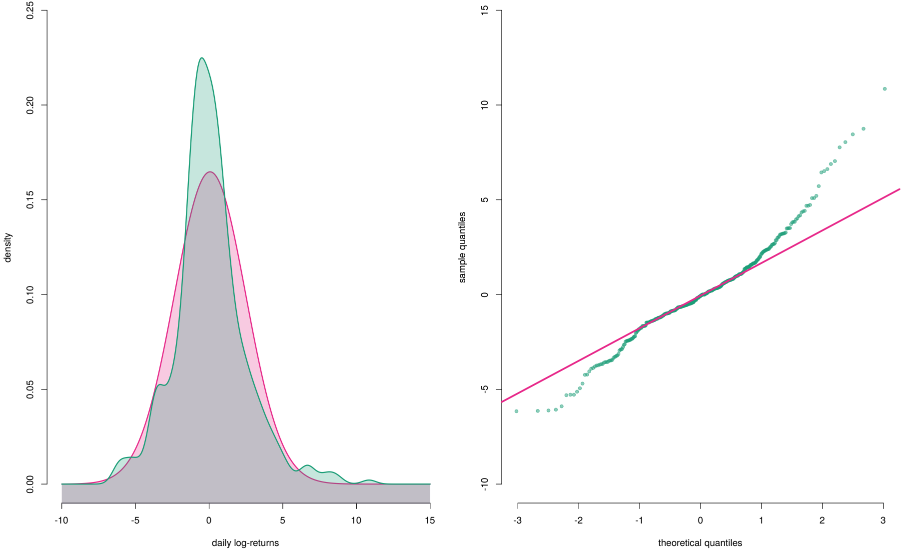
# Motivating example: coffee futures



# Motivating example: coffee futures

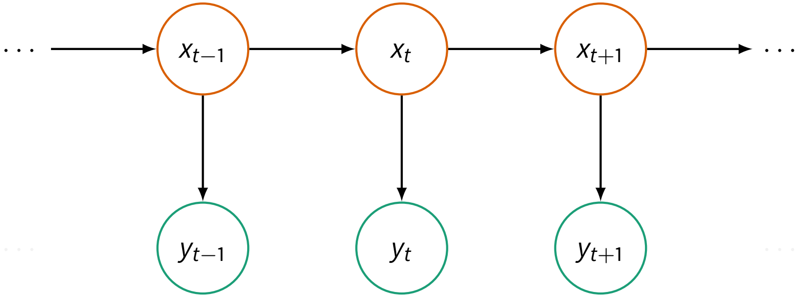


# Motivating example: coffee futures



# State space models: structure

Markov chain $\{X_{0:T}, Y_{1:T}\}$ with $X_t \in \mathcal{X} = \mathbb{R}$, $Y_t \in \mathcal{Y} = \mathbb{R}$ and $t \in \mathbb{N}$.



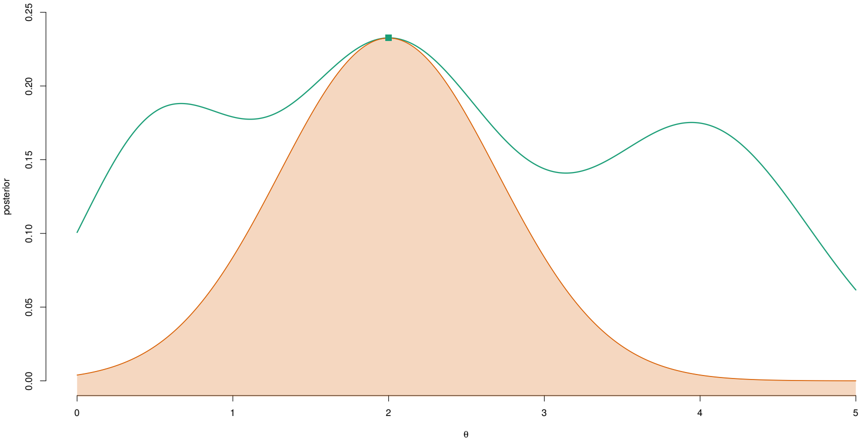

$$x_0 \sim \mu_\theta(x_0) \qquad x_{t+1}|x_t \sim f_\theta(x_{t+1}|x_t), \qquad y_t|x_t \sim g_\theta(y_t|x_t).$$

---

Example: stochastic volatility ($\theta = \{\mu, \phi, \sigma_v\}$):

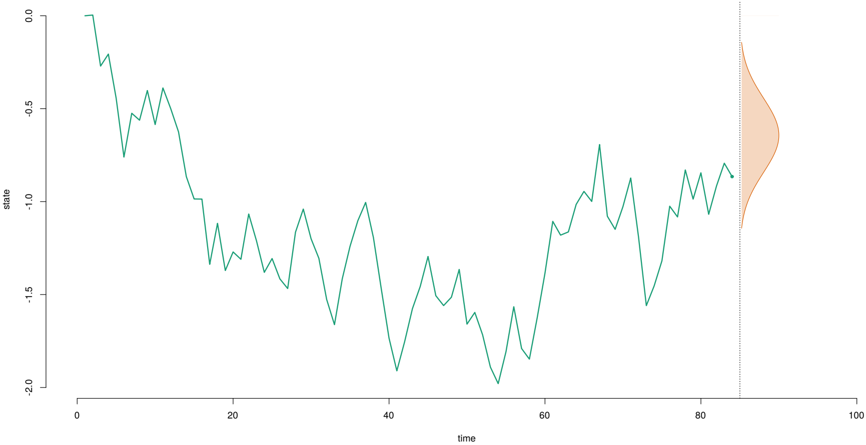
$$x_{t+1}|x_t \sim \mathcal{N}\Big(x_{t+1}; \mu + \phi(x_t - \mu), \sigma_v\Big), \qquad y_t|x_t \sim \mathcal{N}\Big(y_t; 0, \exp(x_t)\Big).$$

# State space models: parameter inference




$$\widehat{\theta}_{\text{MAP}} = \operatorname*{argmax}_{\theta} \Big[ \underbrace{\log p_\theta(y_{1:T}) + \log p(\theta)}_{\triangleq \log \pi(\theta)} \Big]. \quad \widehat{\pi}_{\text{Laplace}}(\theta) = \mathcal{N}\left( \widehat{\theta}_{\text{MAP}}, \Big[ -\nabla^2 \log \pi(\theta) \Big|_{\theta = \widehat{\theta}_{\text{MAP}}} \Big]^{-1} \right).$$

# State space models: state inference




filering: $p_\theta(x_t|y_{1:t})$, marginal smoothing: $p_\theta(x_t|y_{1:T})$, joint smoothing: $p_\theta(x_{1:T}|y_{1:T})$,

# Outline

- Particle-based Gaussian process optimisation
  - Sequential Monte Carlo / particle filtering.
  - Bayesian optimisation / Gaussian process optimisation.
- Extension to models with intractable likelihoods
  - $\alpha$-stable processes.
  - Approximate Bayesian computations.

**Based on the work presented in**:

J. Dahlin and F. Lindsten, **Particle filter-based Gaussian process optimisation for parameter inference**. Proceedings of the 19th World Congress of the International Federation of Automatic Control (IFAC), Cape Town, South Africa, August 2014.

# Gaussian process optimisation

- An instance of Bayesian optimisation.
- *Global* optimisation method.
- Few function evaluations.
- Gradient-free method.
- Popular in machine learning for estimating hyperparameters.


**References**:

J. Mockus, V. Tiesis and A. Zilinskas, **The application of Bayesian methods for seeking the extremum**. In L.C.W. Dixon and G.P. Szegos (editors), Toward Global optimisation, North-Holland, 1978.

D.J. Lizotte, **Practical Bayesian optimisation**. PhD thesis, University of Alberta, 2008.

J. Snoek, H. Larochelle and R.P. Adams, **Practical Bayesian optimisation of Machine learning algorithms**. In Advances in Neural Information Processing Systems (NIPS) 25, Curran Associates Inc, 2012.

# Overview of Gaussian process optimisation

(i) Given iterate $\theta_k$, estimate the log-posterior $\widehat{\pi}_k \approx \pi(\theta_k)$.

Particle filtering.

(ii) Given $\{\theta_j, \widehat{\pi}_j\}_{j=0}^{k}$, create a surrogate cost function of $\pi(\theta)$.

Gaussian process predictive distribution.

(iii) Select a new $\theta_{k+1}$ using the surrogate cost function.

Acquisition function based on the Gaussian process posterior.

# Particle filtering [I/II]

- An instance of sequential Monte Carlo (SMC) samplers.
- Estimates $\mathbb{E}\big[\varphi(x_t)|y_{1:t}\big]$ and $p_\theta(y_{1:T})$.

- Computational cost is in practice of order $\mathcal{O}(NT)$ with $N \sim T$.
- Well-understood statistical properties.

  (unbiasedness, large deviation inequalities, CLTs)

**References**:

A. Doucet and A. Johansen, **A tutorial on particle filtering and smoothing**. In D. Crisan and B. Rozovsky (editors), The Oxford Handbook of Nonlinear Filtering. Oxford University Press, 2011.

O. Cappé, S.J. Godsill and E. Moulines, **An overview of existing methods and recent advances in sequential Monte Carlo**. In Proceedings of the IEEE 95(5), 2007.

# Particle filtering [II/II]

$$\longrightarrow \boxed{\text{Resampling}} \longrightarrow \boxed{\text{Propagation}} \longrightarrow \boxed{\text{Weighting}} \longrightarrow$$

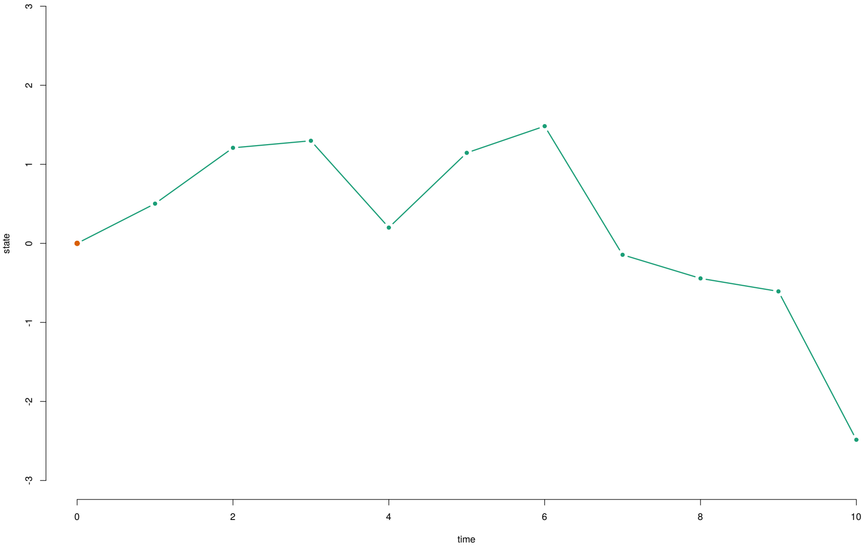
By iterating:

- Resampling: $\mathbb{P}(a_t^{(i)} = j) = \widetilde{w}_{t-1}^{(j)},$ for $i, j = 1, \ldots, N.$
- Propagation: $x_t^{(i)} \sim f_\theta\left(x_t \middle| x_{t-1}^{a_t^{(i)}}\right),$ for $i = 1, \ldots, N.$
- Weighting: $w_t^{(i)} = g_\theta\left(y_t \middle| x_t^{(i)}\right),$ for $i = 1, \ldots, N.$

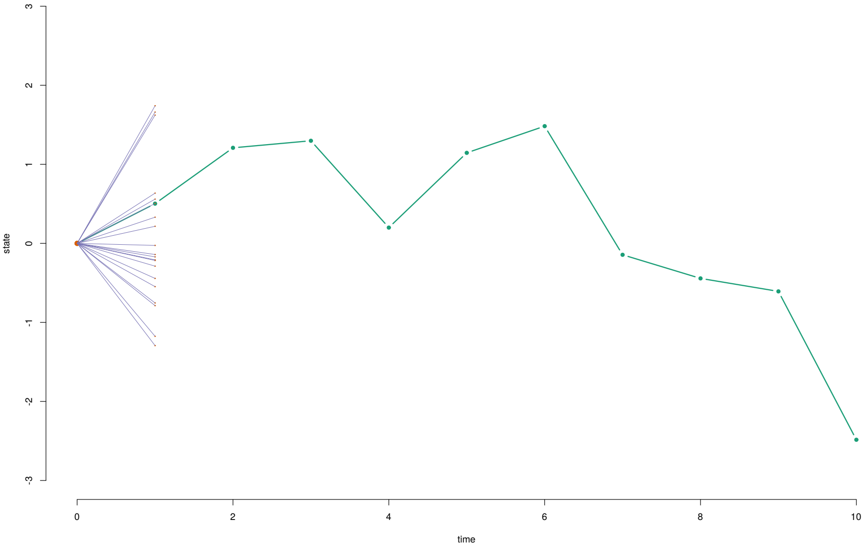
We obtain the particle system

$$\left\{ x_{0:T}^{(i)}, w_{0:T}^{(i)} \right\}_{i=1}^{N}.$$

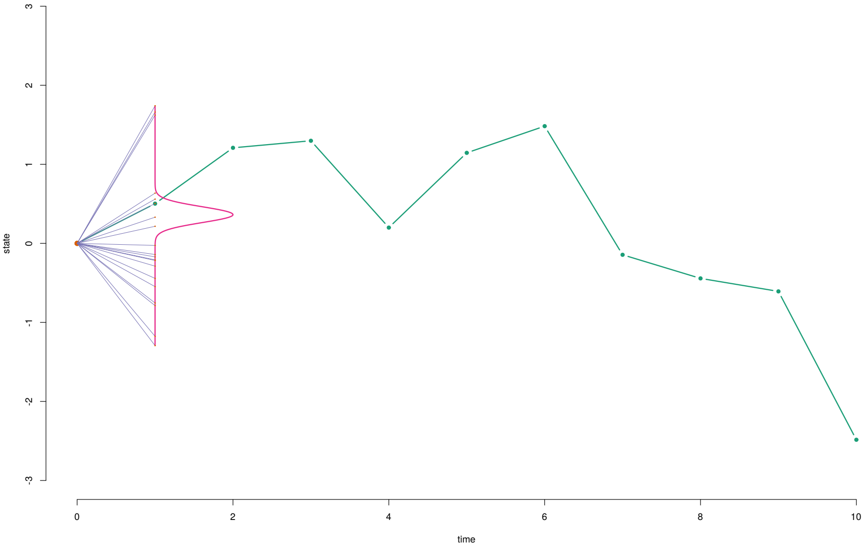
# Particle filtering: animation [I/II]



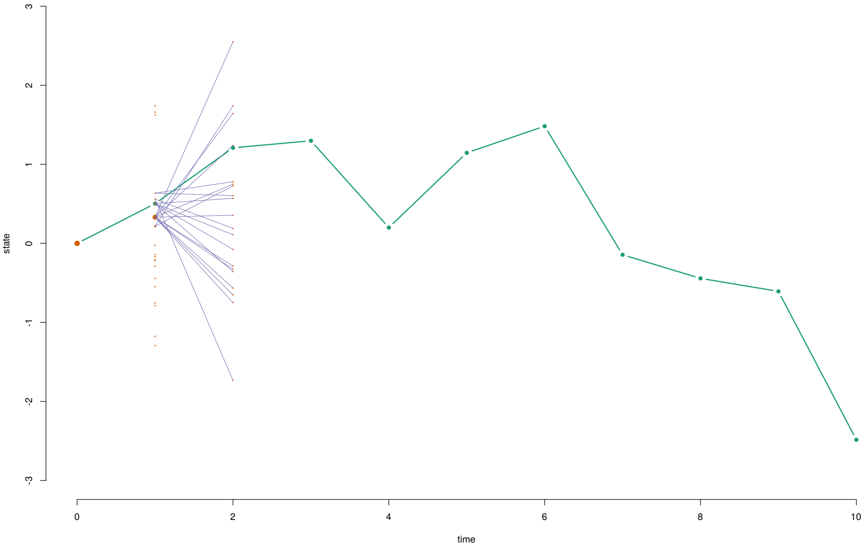
# Particle filtering: animation [I/II]



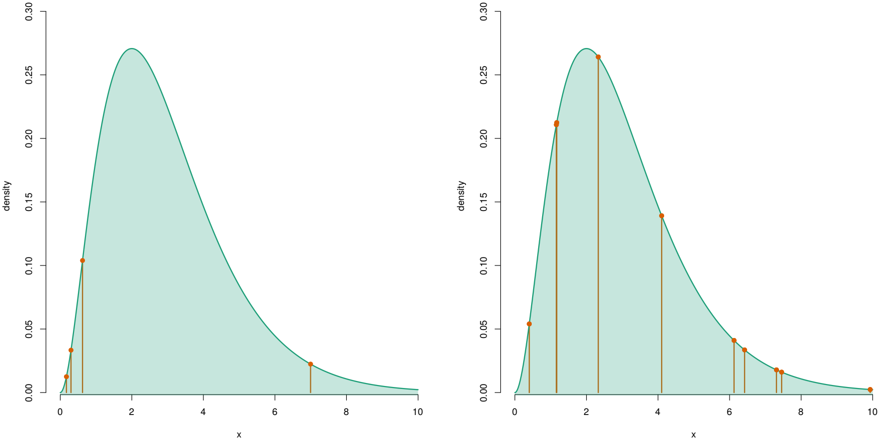
# Particle filtering: animation [I/II]



# Particle filtering: animation [II/II]

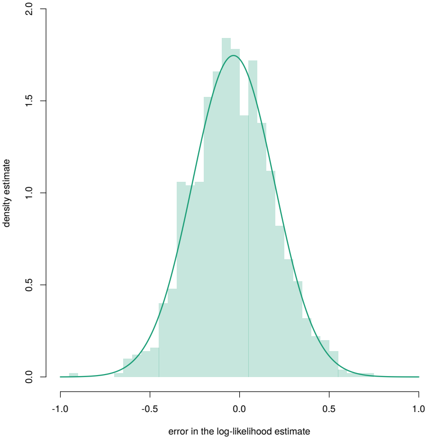


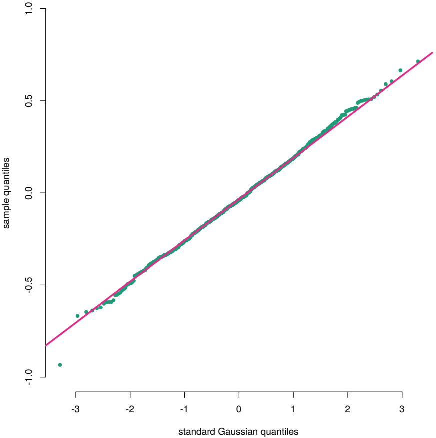
# Particle filtering: state estimation




$$\widehat{\varphi}_t^N \triangleq \widehat{\mathbb{E}}\left[\varphi(x_t)|y_{1:t}\right] = \sum_{i=1}^{N} w_t^{(i)} \varphi\left(x_t^{(i)}\right), \qquad \sqrt{N}\left(\varphi_t - \widehat{\varphi}_t^N\right) \xrightarrow{d} \mathcal{N}\left(0, \sigma_t^2(\varphi)\right).$$

# Particle filtering: likelihood estimation




$$\log \widehat{p}_\theta(y_{1:T}) = \sum_{t=1}^{T} \log\left(\sum_{i=1}^{N} w_t^{(i)}\right) - T\log N, \quad \sqrt{N}\left(\log p_\theta(y_{1:T}) - \log \widehat{p}_\theta(y_{1:T}) + \frac{\sigma^2_{\widehat{\pi}}}{2N}\right) \xrightarrow{d} \mathcal{N}\left(0, \sigma^2_{\widehat{\pi}}\right).$$

# Gaussian process regression [I/III]

- Estimator for $\pi(\theta)$ available with Gaussian error.

- A Gaussian process is an infinite dimensional Gaussian distribution.
- Specified by a mean function *m* and covariance function (kernel) $\kappa$.
- Hyperparameters estimated using empirical Bayes.

- Nonparametric Bayesian method.
- Can be used for regression using a standard prior-posterior update.

**References**:

C.E. Rasmussen and C.K.I Williams, **Gaussian Processes for Machine Learning**. MIT Press, 2006

# Gaussian process regression [II/III]

We assume a priori

$$\pi(\theta) \sim \mathcal{GP}\Big(m(\theta), \kappa(\theta, \theta')\Big),$$

which together with the data

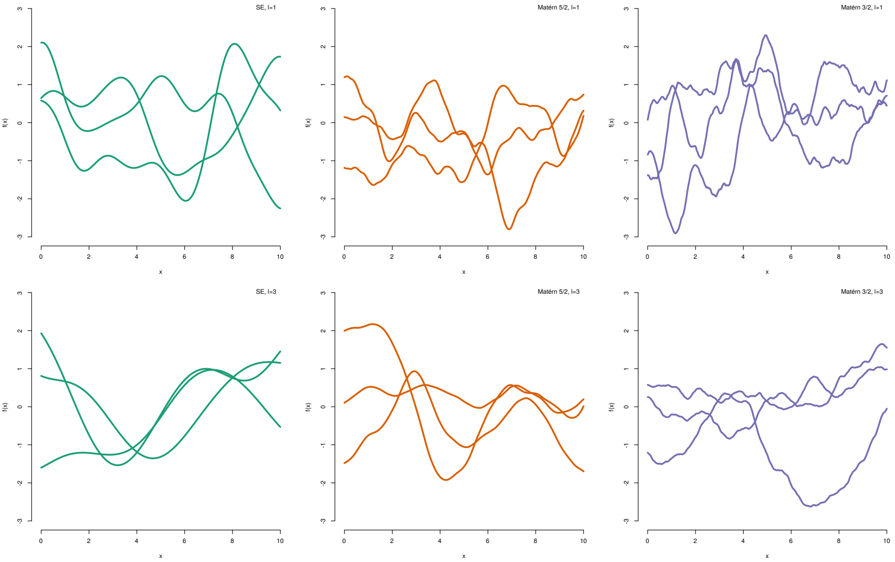
$$\widehat{\pi}(\theta) = \pi(\theta) + \sigma_{\widehat{\pi}} z_t, \qquad z_t \sim \mathcal{N}(0, 1),$$

gives the posterior predictive distribution

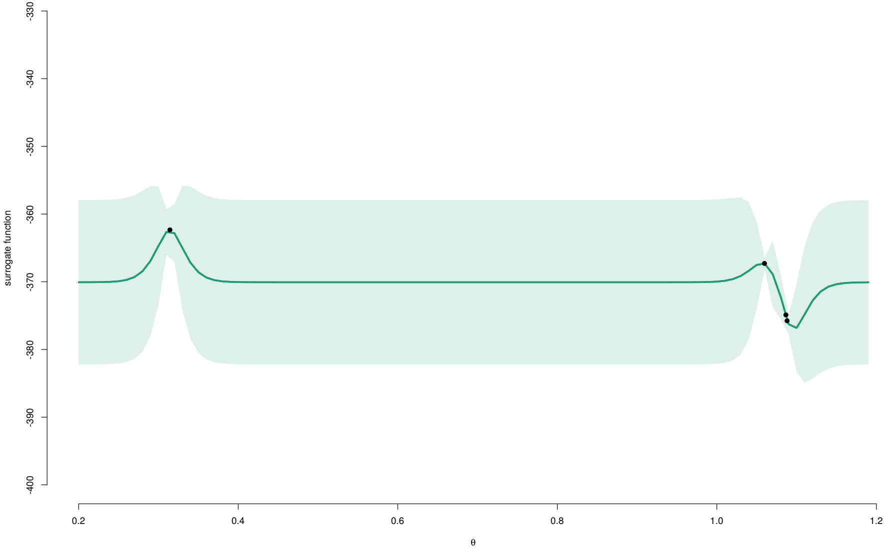
$$\pi(\theta_\star) | \mathcal{D}_k \sim \mathcal{N}\Big(\mu(\theta_\star | \mathcal{D}_k), \sigma^2(\theta_\star | \mathcal{D}_k) + \sigma^2_{\widehat{\pi}}\Big),$$

where the current data is denoted $\mathcal{D}_k = \{\theta_j, \widehat{\pi}_i\}_{j=1}^{k}$.

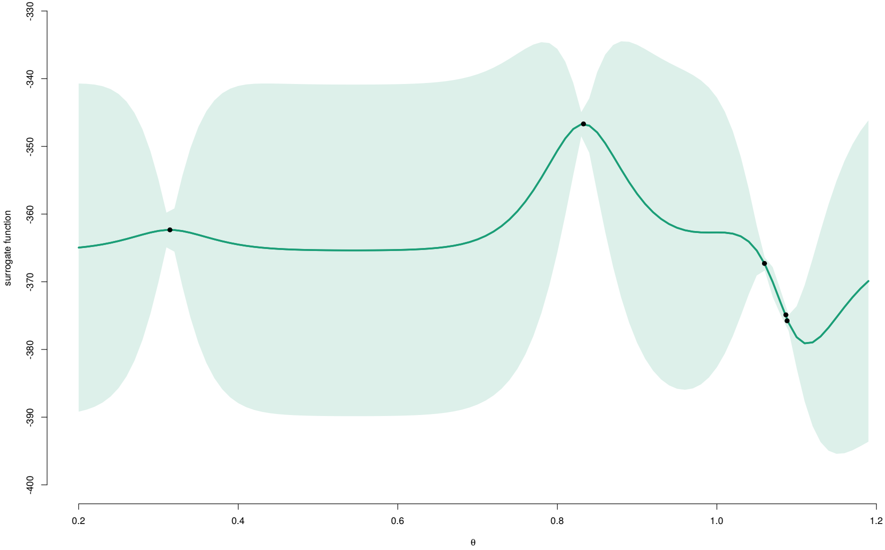
# Gaussian process regression [III/III]



# Gaussian process regression: toy example [I/II]



# Gaussian process regression: toy example [II/II]



# Acquisition rule for selecting sampling points

- Estimator for $\pi(\theta)$ available with Gaussian error.
- Surrogate of the log-posterior available as a Gaussian process.

- Idea: use the Gaussian process model to select $\theta_{k+1}$.
- Balance exploration and exploitation using an acquisition rule by

$$\theta_{k+1} = \underset{\theta_\star \in \Theta}{\operatorname{argmax}} \, \mathrm{AQ}(\theta_\star | \mathcal{D}_k).$$

- Cheap to evaluate $\mathrm{AQ}(\theta_\star | \mathcal{D}_k)$.

**References**:

J. Mockus, V. Tiesis and A. Zilinskas, **The application of Bayesian methods for seeking the extremum**. In L.C.W. Dixon and G.P. Szegos (editors), Toward Global optimisation, North-Holland, 1978.

D.J. Lizotte, **Practical Bayesian optimisation**. PhD thesis, University of Alberta, 2008.

J. Snoek, H. Larochelle and R.P. Adams, **Practical Bayesian optimisation of Machine learning algorithms**. In Advances in Neural Information Processing Systems (NIPS) 25, Curran Associates Inc, 2012.

# Acquisition rule: Upper confidence bound

Consider, the 95% upper confidence bound as the acquisition rule

$$\theta_{k+1} = \underset{\theta_\star \in \Theta}{\text{argmax}} \left[ \mu(\theta_\star | \mathcal{D}_k) + 1.96 \sqrt{\sigma^2(\theta_\star | \mathcal{D}_k)} \right],$$

to determine the next iterate $\theta_{k+1}$.

# Overview of Gaussian process optimisation

(i) Given iterate $\theta_k$, estimate the log-posterior $\widehat{\pi}_k \approx \pi(\theta_k)$.

Particle filtering.

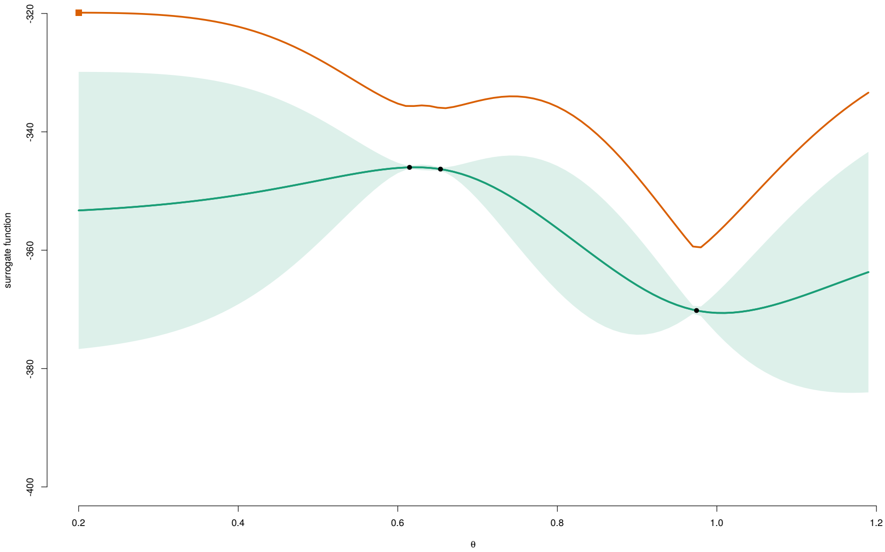
(ii) Given $\{\theta_j, \widehat{\pi}_j\}_{j=0}^{k}$, create a surrogate cost function of $\pi(\theta)$.

Gaussian process predictive distribution.

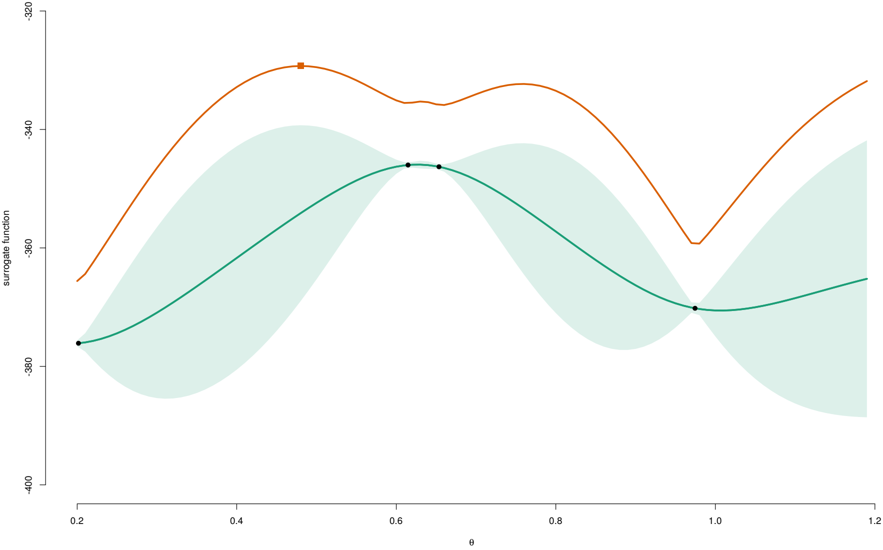
(iii) Select a new $\theta_{k+1}$ using the surrogate cost function.

Acquisition function based on the Gaussian process posterior.

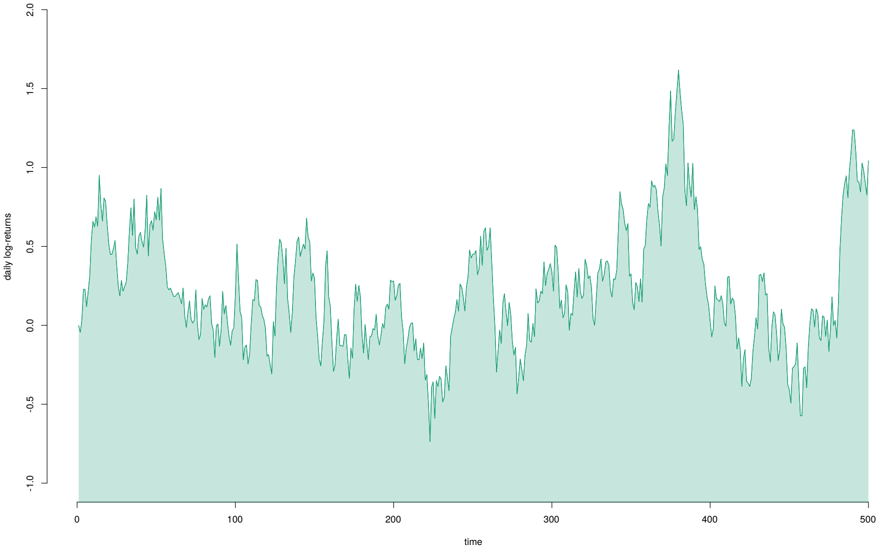
# Example: Gaussian process optimisation [I/II]



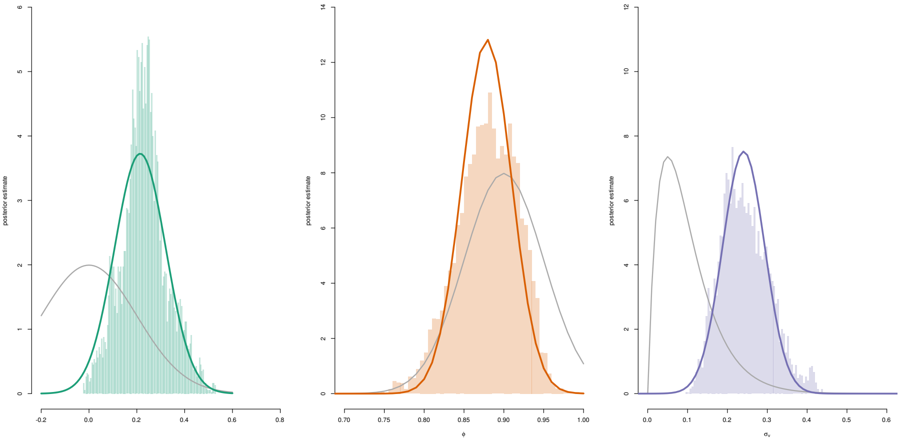
# Example: Gaussian process optimisation [II/II]



# Stochastic volatility: synthetic data [I/IV]



# Stochastic volatility: synthetic data [II/IV]

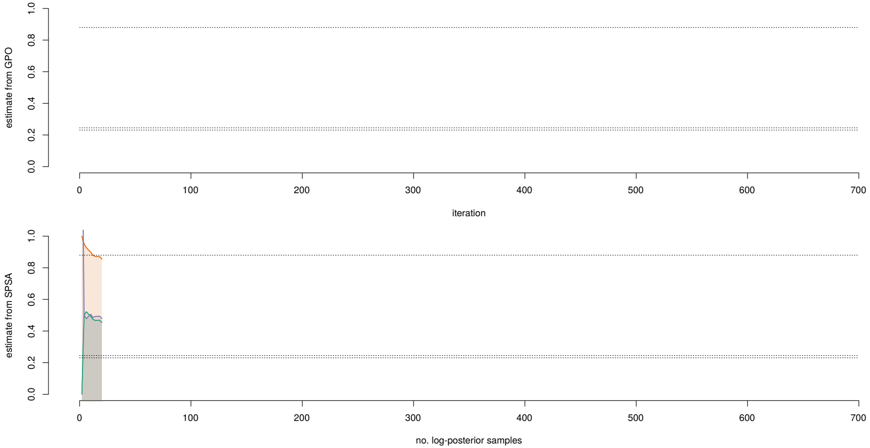



Stochastic volatility model with Gaussian returns ($\theta = \{\mu, \phi, \sigma_v\} = \{0.20, 0.96, 0.15\}$)

$$x_{t+1}|x_t \sim \mathcal{N}\Big(x_{t+1}; \mu + \phi(x_t - \mu), \sigma_v\Big), \qquad y_t|x_t \sim \mathcal{N}\Big(y_t; 0, \exp(x_t)\Big).$$

Log-posterior estimates for PMH: 10 000 (histogram) and GPO: 350 (solid lines).

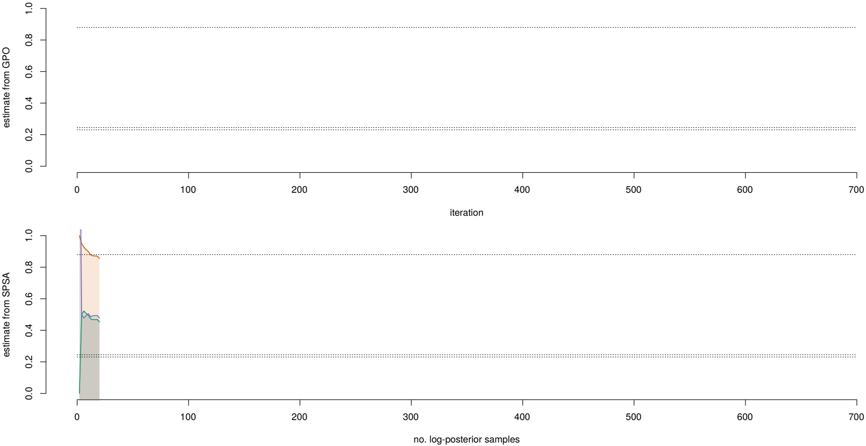
# Stochastic volatility: synthetic data [III/IV]




Stochastic volatility model with Gaussian returns ($\theta = (\mu, \phi, \sigma_v) = (0.20, 0.96, 0.15)$)

$$x_{t+1}|x_t \sim \mathcal{N}\Big(x_{t+1}; \mu + \phi(x_t - \mu), \sigma_v\Big), \qquad y_t|x_t \sim \mathcal{N}\Big(y_t; 0, \exp(x_t)\Big),$$

# Stochastic volatility: synthetic data [IV/IV]




Stochastic volatility model with Gaussian returns ($\theta = (\mu, \phi, \sigma_v) = (0.20, 0.96, 0.15)$)

$$x_{t+1}|x_t \sim \mathcal{N}\Big(x_{t+1}; \mu + \phi(x_t - \mu), \sigma_v\Big), \qquad y_t|x_t \sim \mathcal{N}\Big(y_t; 0, \exp(x_t)\Big),$$

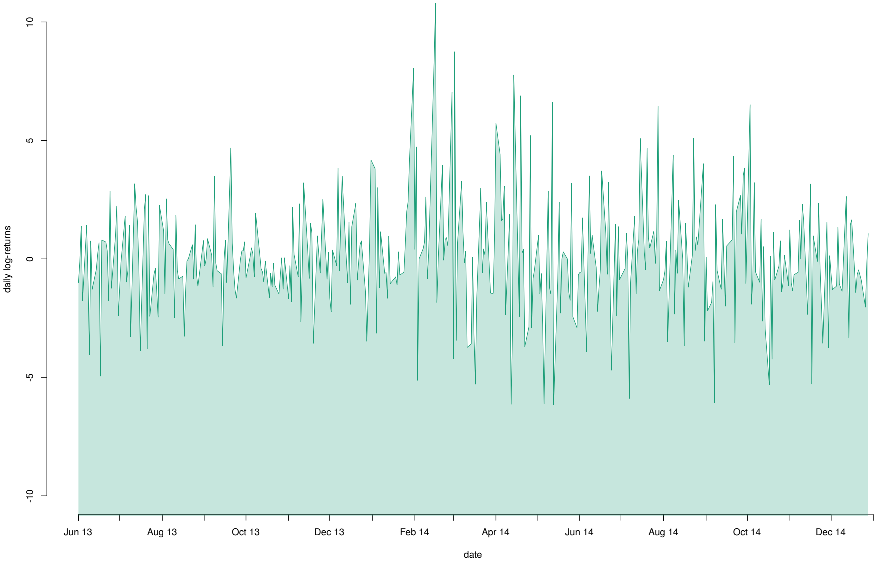
# Outline

- Particle-based Gaussian process optimisation
  - Sequential Monte Carlo / particle filtering.
  - Bayesian optimisation / Gaussian process optimisation.
- Extension to models with intractable likelihoods
  - $\alpha$-stable processes.
  - Approximate Bayesian computations.

**Based on the work presented in**:

J. Dahlin, T. B. Schön, and M. Villani. **Approximate inference in state space models with intractable likelihoods using Gaussian process optimisation**. Technical Report LiTH-ISY-R-3075, Department of Electrical Engineering, Linköping University, Linköping, Sweden, April 2014.

# Motivating example: coffee futures



# Stochastic volatility model with $\alpha$-stable returns [I/II]

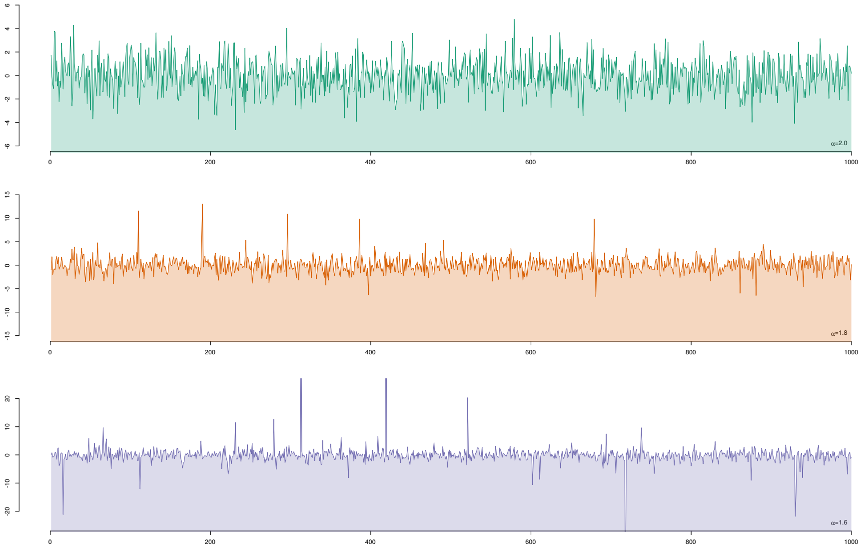
- Consider the model

$$x_{t+1}|x_t \sim \mathcal{N}\Big(x_{t+1}; \mu + \phi(x_t - \mu), \sigma_v\Big),$$
$$y_t|x_t \sim \mathcal{A}\Big(y_t; \alpha, \exp(x_t)\Big),$$

  with $\theta = \{\mu, \phi, \sigma_v, \alpha\}$.

- $\mathcal{A}(\alpha, \gamma)$ denotes a symmetric $\alpha$-stable distribution with zero mean and scale $\gamma$.

# Stochastic volatility model with $\alpha$-stable returns [II/II]



# Approximate Bayesian computations (ABC)

- The likelihood for this model is intractable.
- Often easy to simulate from the data distribution (likelihood).

- Idea: Simulate data from model and compare with observations.
- Augment the posterior with an auxiliary variable

$$\pi_\epsilon(\theta, \tilde{y}_{1:T}) = \frac{p_\theta(\tilde{y}_{1:T}) p(\theta) \kappa_\epsilon(\tilde{y}_{1:T} - y_{1:T})}{\int p_{\theta'}(\tilde{y}_{1:T}) p(\theta') \kappa_\epsilon(\tilde{y}_{1:T} - y_{1:T}) \, \mathrm{d}\theta'}.$$

and for a small enough $\epsilon$, we have

$$\pi_\epsilon(\theta) = \int \pi_\epsilon(\theta, \tilde{y}_{1:T}) \, \mathrm{d}\tilde{y}_{1:T}.$$

**References**:

- S. Yildirim, S.S. Singh, T. Dean and A. Jasra. **Parameter Estimation in Hidden Markov Models with Intractable Likelihoods Using Sequential Monte Carlo**. Journal of Computational and Graphical Statistics, 2014.
- J.M., Marin, P. Pudlo, C.P. Robert, and R.J. Ryder. **Approximate Bayesian computational methods**. Statistics and Computing, 22(6), 1167-1180, 2012.

# Particle filtering with ABC [I/II]

- We can simulate an $\alpha$-stable r.v. by the transformation $\tilde{y}_t = \tau_\theta(v_t, x_t)$.
- Consider adding noise to the measurements

$$y_t^\star = y_t + \epsilon z_t, \qquad z_t \sim \mathcal{N}(0,1).$$

- The model can then be rewritten as

$$\begin{aligned} x_{t+1}|x_t &\sim \mathcal{N}\Big(x_{t+1}; \mu + \phi(x_t - \mu), \sigma^2\Big), \\ v_t|x_t &\sim \nu_\theta(v_t; x_t), \\ y_t^\star|v_t &\sim h_{\theta,\epsilon}(y_t|v_t) = \mathcal{N}(y_t^\star; \tau_\theta(v_t, x_t), \epsilon^2), \end{aligned}$$

where $(x_t, v_t)$ is the new state vector.

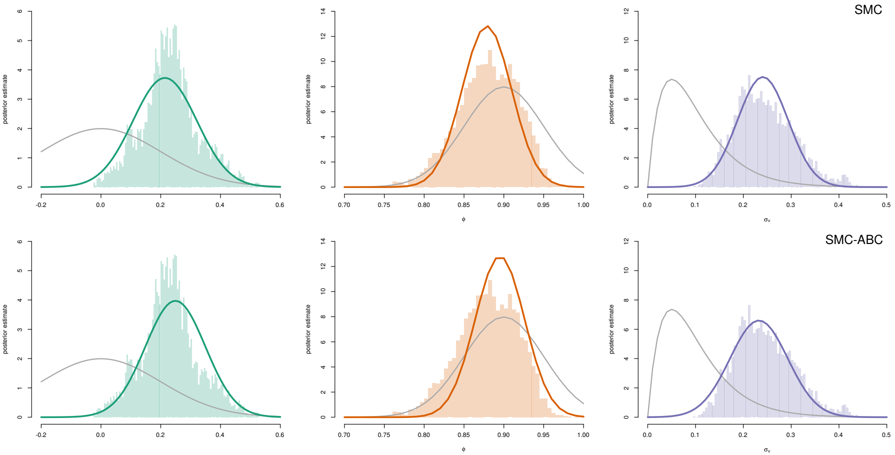
# Particle filtering with ABC [II/II]

$\rightarrow$ Resampling $\rightarrow$ Propagation $\rightarrow$ Weighting $\rightarrow$

- Resampling: $\mathbb{P}(a_t^{(i)} = j) = \widetilde{w}_{t-1}^{(j)}$, for $i, j = 1, \ldots, N$.
- Propagation [1]: $x_t^{(i)} \sim f_\theta\left(x_t | x_{t-1}^{a_t^{(i)}}\right)$, for $i = 1, \ldots, N$.
- Propagation [2] : $v_t^{(i)} \sim \nu_\theta\left(v_t | x_t^{(i)}\right)$, for $i = 1, \ldots, N$.
- Weighting: $w_t^{(i)} = h_{\theta,\epsilon}\left(y_t^\star | v_t^{(i)}\right)$, for $i = 1, \ldots, N$.

Results in an unbiased likelihood estimator (under some assumptions).

# Stochastic volatility: synthetic data [I/III]

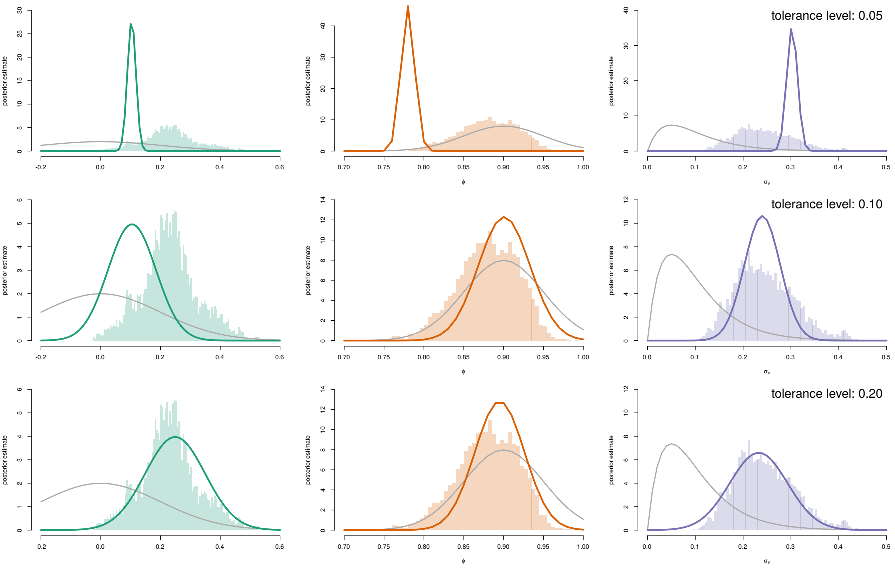



Stochastic volatility model with Gaussian returns ($\theta = \{\mu, \phi, \sigma_v\} = \{0.20, 0.96, 0.15\}$)

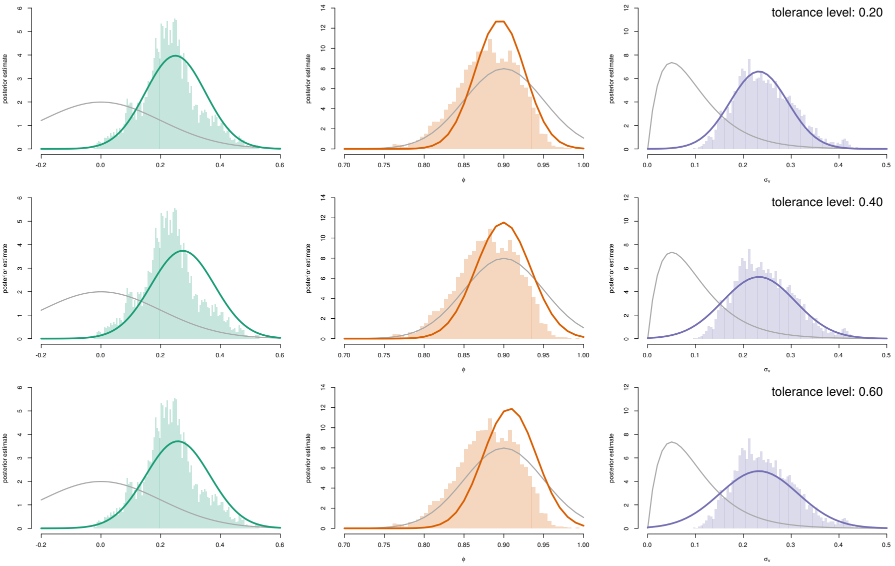
$$x_{t+1} | x_t \sim \mathcal{N}\Big(x_{t+1}; \mu + \phi(x_t - \mu), \sigma_v\Big), \qquad y_t | x_t \sim \mathcal{N}\Big(y_t; 0, \exp(x_t)\Big).$$

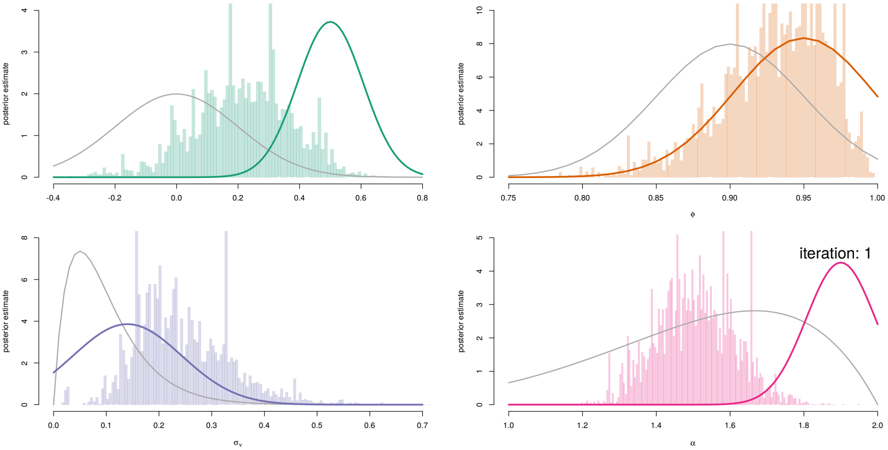
Log-posterior estimates for PMH: 10 000 (histogram) and GPO: 350 (solid lines).

# Stochastic volatility: synthetic data [II/III]



# Stochastic volatility: synthetic data [III/III]



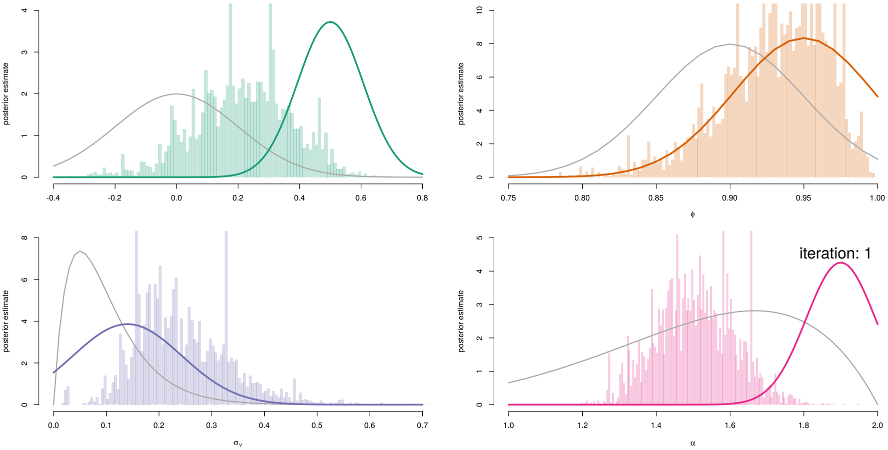



Stochastic volatility model with $\alpha$-stable returns

$$x_{t+1}|x_t \sim \mathcal{N}\Big(x_{t+1}; \mu + \phi(x_t - \mu), \sigma_v\Big), \qquad y_t|x_t \sim \mathcal{A}\Big(y_t; \alpha, \exp(x_t)\Big).$$

Log-posterior estimates for PMH-ABC: 30 000 (histogram) and GPO-ABC: 350 (solid lines).

# Motivating example: volatility of coffee futures [II/III]

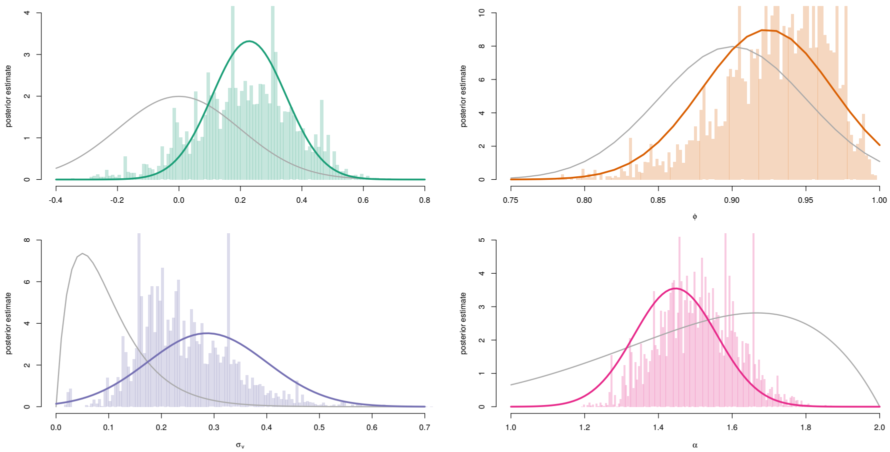



Stochastic volatility model with $\alpha$-stable returns

$$x_{t+1}|x_t \sim \mathcal{N}\Big(x_{t+1}; \mu + \phi(x_t - \mu), \sigma_v\Big), \qquad y_t|x_t \sim \mathcal{A}\Big(y_t; \alpha, \exp(x_t)\Big).$$

Log-posterior estimates for PMH-ABC: 30 000 (histogram) and GPO-ABC: 350 (solid lines).

# Motivating example: volatility of coffee futures [III/III]

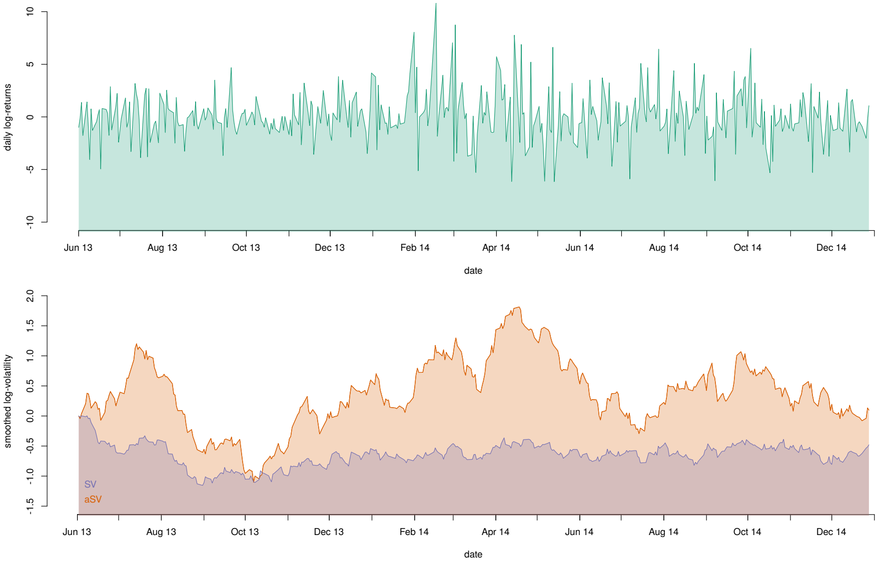



Stochastic volatility model with $\alpha$-stable returns

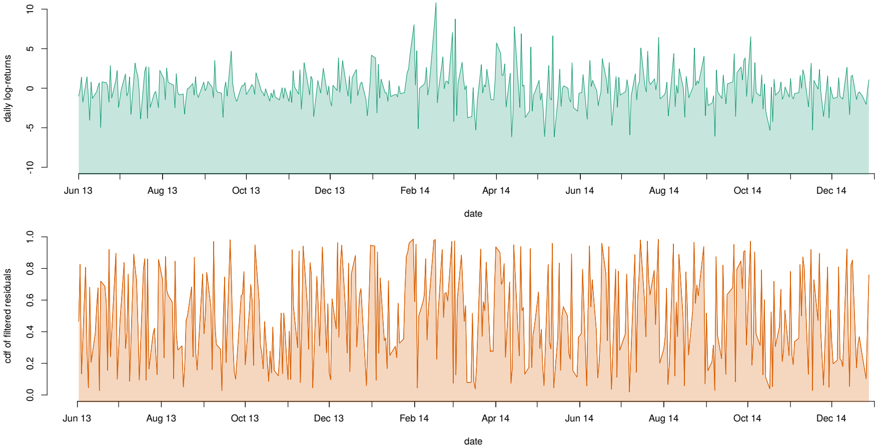
$$x_{t+1}|x_t \sim \mathcal{N}\Big(x_{t+1}; \mu + \phi(x_t - \mu), \sigma_v\Big), \qquad y_t|x_t \sim \mathcal{A}\Big(y_t; \alpha, \exp(x_t)\Big).$$

Log-posterior estimates for PMH-ABC: 30 000 (histogram) and GPO-ABC: 350 (solid lines).

# Motivating example: volatility of coffee futures

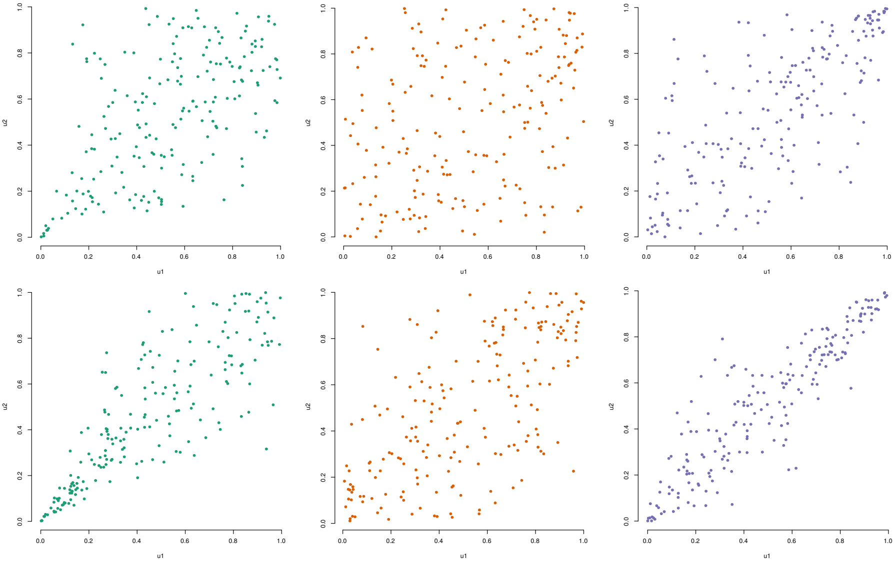


# Motivating example: residuals of coffee futures




$$\widehat{e}_t = y_t \exp(-\widehat{x}_t/2), \qquad \widehat{u}_t = G_{\widehat{\theta}}(\widehat{e}_t).$$

# Motivating example: residuals of coffee futures



# Conclusions

- Methods
  - Particle filtering for log-likelihood estimation.
  - Gaussian process surrogate model and acquisition rules.
  - Approximate Bayesian computations.
- Advantages
  - Decreased computational cost compared with popular methods.
  - Only makes use of *cheap* zero-order information.
- Future work
  - Bias compensation of log-likelihood estimate.
  - Sparse Gaussian processes.
  - Kernel design and new acquisition rules.
  - Improved estimates of the Hessian.

## Thank you for your attention!

- Questions, comments and suggestions are most welcome.
- The papers and some implementations are available for download from my homepage: http://work.johandahlin.com/.

**This presentation is based on the work presented in**:

J. Dahlin and F. Lindsten, **Particle filter-based Gaussian process optimisation for parameter inference**. Proceedings of the 19th World Congress of the International Federation of Automatic Control (IFAC), Cape Town, South Africa, August 2014.

J. Dahlin, T. B. Schön, and M. Villani. **Approximate inference in state space models with intractable likelihoods using Gaussian process optimisation**. Technical Report LiTH-ISY-R-3075, Department of Electrical Engineering, Linköping University, Linköping, Sweden, April 2014.

# Acquisition rule: Probability of improvement

Consider the probability

$$\mathbb{P}(\mu(\theta_\star|\mathcal{D}_k) \geq \mu_{\max} + \epsilon) = \mathbb{P}\left(\frac{\mu(\theta_\star|\mathcal{D}_k) - \mu_{\max} - \epsilon}{\sigma^2(\theta_\star|\mathcal{D}_k)} \geq 0\right)$$
$$\triangleq \Phi(z_k(\theta_\star))$$

where $\mu_{\max}$ denotes the current maximum of $\{\widehat{\pi}_1, \ldots, \widehat{\pi}_k\}$.

# Acquisition rule: Expected improvement

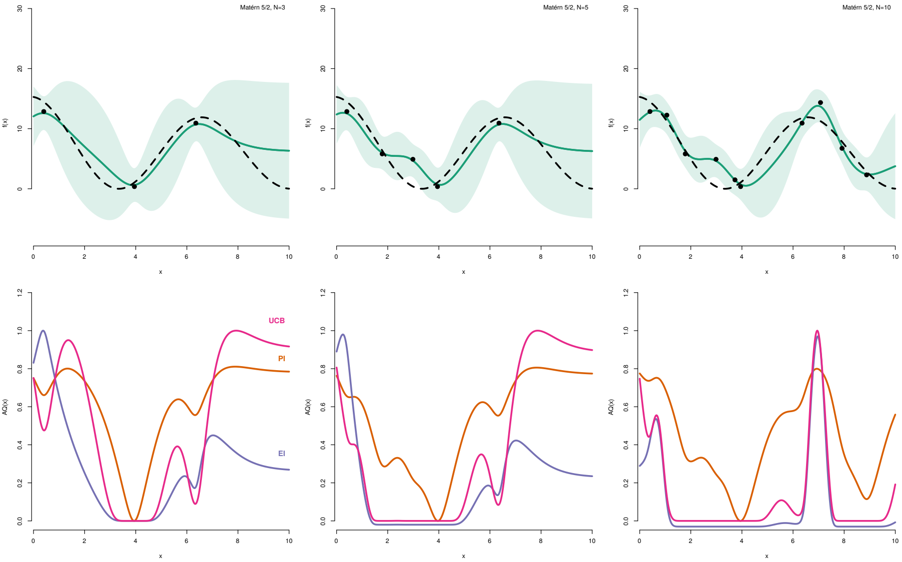
Consider the function

$$I(\theta_\star) = \max\left\{0, \mu(\theta_\star|\mathcal{D}_k) - \mu_{\text{max}} - \epsilon\right\},$$

and its expected value with respect to the GP, given by

$$\begin{aligned}\mathbb{E}\Big[I(\theta_\star)|\mathcal{D}_k\Big] &= \int I(\theta_\star)\,\mathrm{d}\Phi(z_k(\theta_\star)) \\ &= \sigma^2(\theta_\star|\mathcal{D}_k)\Big[z_k(\theta_\star)\Phi\big(z_k(\theta_\star)\big) + \phi\big(z_k(\theta_\star)\big)\Big].\end{aligned}$$

# Acquisition rule: comparison



# Importance sampling (IS) with ABC

(i) For $k = 1, \ldots, K$

  (a) Sample $\theta^{(k)} \sim p(\theta)$,

  (b) Compute the weight $w^{(k)} = \pi(\theta^{(k)})$,

(ii) Estimate the parameter posterior mean

$$\mathbb{E}[\theta | y_{1:T}] \approx \sum_{k=1}^{K} \frac{w^{(k)}}{\sum_{l=1}^{K} w^{(l)}} \theta^{(k)}.$$

# Importance sampling (IS) with ABC

(i) For $k = 1, \ldots, K$
   (a) Sample $\theta^{(k)} \sim p(\theta)$,
   (b) Generate $\tilde{y}_{1:T} \sim \nu_{\theta^{(k)}}$.
   (c) Compute the weight

$$w^{(k)} = p(\theta^{(k)}) \kappa_\epsilon \Big( y_{1:T} - \tilde{y}_{1:T} \Big),$$

(ii) Estimate the parameter posterior mean

$$\mathbb{E}[\theta | y_{1:T}] \approx \sum_{k=1}^{K} \frac{w^{(k)}}{\sum_{l=1}^{K} w^{(l)}} \theta^{(k)}.$$

LINKÖPINGS UNIVERSITET

LINKÖPINGS UNIVERSITET