The Expectation-Propagation Algorithm: a tutorial

Day II: Recent Advances

Simon Barthelmé, Gipsa-lab, CNRS

2nd March 2016

Outline

- ABC-EP: EP for likelihood-free problems
- MCMC-EP: speeding up MCMC for large datasets
- Average-EP and stochastic EP: simpler EP

Likelihood-free Bayesian inference

- A class of techniques that can be applied when
  - Likelihood evaluation is impossible or very, very slow
  - Sampling from the model is comparatively easy

Likelihood-free Bayesian inference

- A class of techniques that can be applied when
  - Likelihood evaluation is impossible or very, very slow
  - Sampling from the model is comparatively easy
- Most famous incarnation: the Approximate Bayesian Computation alg. of Pritchard et al. (1999)

Approximate Bayesian Computation

- There are many intractable-likelihood models in Population Genetics, where researchers are interested for example in reconstructing evolutionary trees from molecular data.

Approximate Bayesian Computation

- There are many intractable-likelihood models in Population Genetics, where researchers are interested for example in reconstructing evolutionary trees from molecular data.
- Enter [?], with an algorithm now know as *ABC*, for Approximate Bayesian Computation, now perhaps the hottest topic in computational and applied statistics.

Approximate Bayesian Computation

- There are many intractable-likelihood models in Population Genetics, where researchers are interested for example in reconstructing evolutionary trees from molecular data.
- Enter [?], with an algorithm now know as *ABC*, for Approximate Bayesian Computation, now perhaps the hottest topic in computational and applied statistics.
- Idea brilliantly simple if one thinks of Bayesian modelling as defining a *joint* distribution $p(\mathbf{y}, \boldsymbol{\theta}) = p(\boldsymbol{\theta})p(\mathbf{y}|\boldsymbol{\theta})$ over data and parameters.

ABC^2 (The ABC of ABC)

1. Sample $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$

ABC^2 (The ABC of ABC)

1. Sample $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$
2. Sample $\mathbf{y} \sim p(\mathbf{y}|\boldsymbol{\theta})$

ABC^2 (The ABC of ABC)

1. Sample $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$
2. Sample $\mathbf{y} \sim p(\mathbf{y}|\boldsymbol{\theta})$
3. Accept $\boldsymbol{\theta}$ iff $||\mathbf{y} - \mathbf{y}^{\star}|| < \epsilon$

ABC^2 (The ABC of ABC)

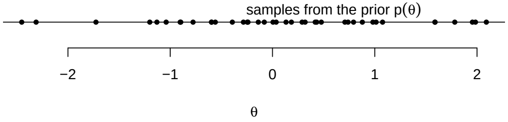
1. Sample $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$
2. Sample $\mathbf{y} \sim p(\mathbf{y}|\boldsymbol{\theta})$
3. Accept $\boldsymbol{\theta}$ iff $||\mathbf{y} - \mathbf{y}^{\star}|| < \epsilon$

This algorithm produces samples from

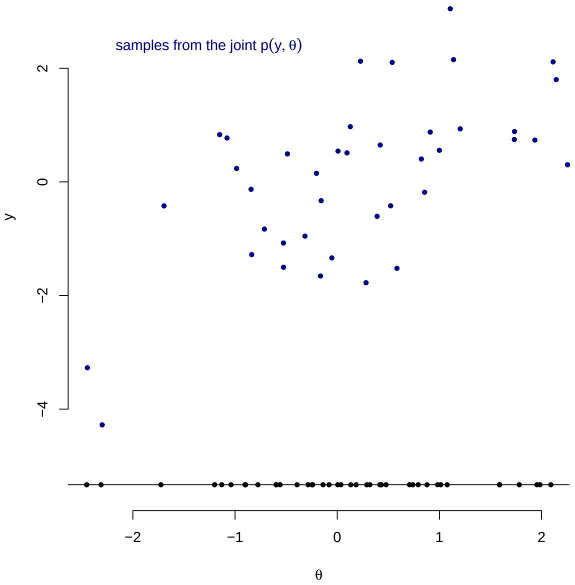
$$p_{\epsilon}\left(\boldsymbol{\theta}|\mathbf{y}^{\star}\right) \propto p\left(\boldsymbol{\theta}\right) \int p(\mathbf{y}|\boldsymbol{\theta}) \mathbb{I}\left(||\mathbf{y} - \mathbf{y}^{\star}|| < \epsilon\right) d\mathbf{y}$$

which tends to $p(\boldsymbol{\theta}|\mathbf{y})$ with $\epsilon \to 0$.

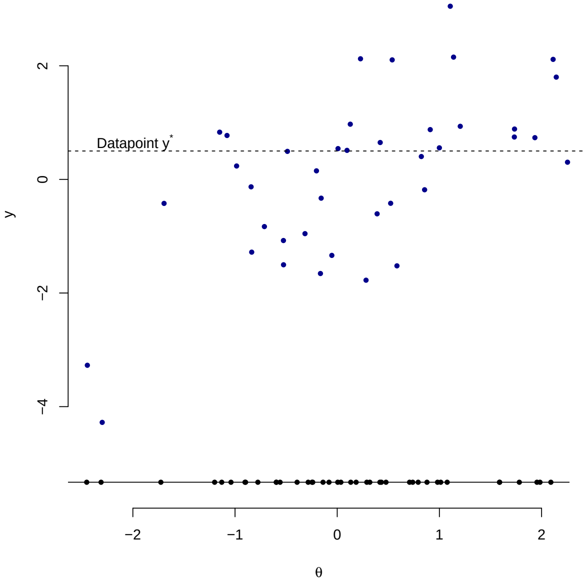
ABC in pictures



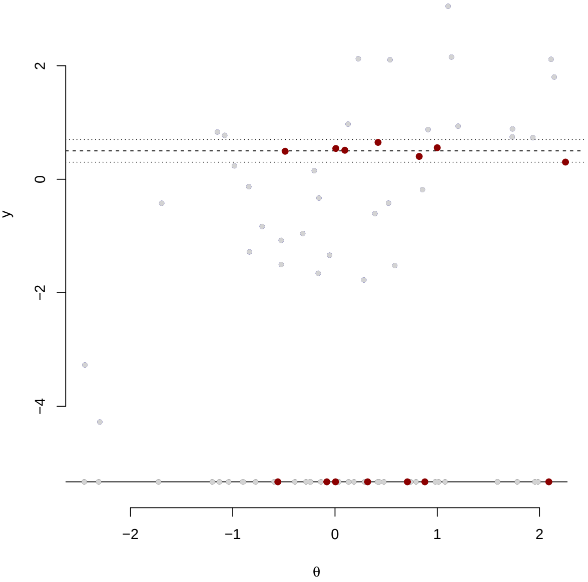
ABC in pictures



ABC in pictures



ABC in pictures



A problem with basic ABC

1. Sample $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$
2. Sample $\mathbf{y} \sim p(\mathbf{y}|\boldsymbol{\theta})$
3. Accept $\boldsymbol{\theta}$ iff $||\mathbf{y} - \mathbf{y}^{\star}|| < \epsilon$.

If there are many datapoints, then either ϵ is enormous or the probability of acceptance is going to be impractically small (the model will never reproduce *exactly* a large dataset).

Introducing summary statistics

- Solution found by [?]: reduce the dimension of $\mathbf{y}$ by computing some *summary statistics* $\mathbf{s}(\mathbf{y})$, and modify the algorithm:

1. Sample $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$
2. Sample $\mathbf{y} \sim p(\mathbf{y}|\boldsymbol{\theta})$
3. Accept $\boldsymbol{\theta}$ iff $||\mathbf{s}(\mathbf{y}) - \mathbf{s}(\mathbf{y}^{\star})|| < \epsilon$.

Introducing summary statistics

- Solution found by [?]: reduce the dimension of $\mathbf{y}$ by computing some *summary statistics* $\mathbf{s}(\mathbf{y})$, and modify the algorithm:

1. Sample $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$
2. Sample $\mathbf{y} \sim p(\mathbf{y}|\boldsymbol{\theta})$
3. Accept $\boldsymbol{\theta}$ iff $||\mathbf{s}(\mathbf{y}) - \mathbf{s}(\mathbf{y}^\star)|| < \epsilon$.

- Provided that the choice of summary statistics is appropriate, this behaves reasonably and is actually computationally feasible.

More advanced variants

- There are by now many, more efficient, variants on the original algorithm, based on MCMC, Sequential Monte Carlo, etc. [ref.]
- All of them require the definition of summary statistics, and are rather slow and difficult to tune.
- Using EP you can get rid of summary statistics, and obtain substantial speed-ups (10-100x, Barthelmé & Chopin, 2011, Barthelmé & Chopin, 2014, Barthelmé, Chopin, Cottet, 2015).
  - Caveat I: you can't get rid of summary statistics in all models
  - Caveat II: implementation is a bit of work
  - Caveat III: you get a Gaussian approximation (it's still EP)

How to get rid of summary statistics

- In the ABC-reject algorithm, we needed summary statistics because we had more than one datapoint.
- In EP we only integrate one datapoint at a time, so
  - No need for summary statistics!
  - We can just compute all hybrid moments using ABC-reject
  - Our objective is
    $p_\epsilon(\boldsymbol{\theta}|\mathbf{y}^\star) \propto p(\boldsymbol{\theta}) \prod_{i=1}^n \left\{ \int f_i(y_i|\boldsymbol{\theta}) \mathbb{I}_{\{\|y_i - y_i^\star\| \leq \epsilon\}}\, dy_i \right\}$

ABC-EP in one slide

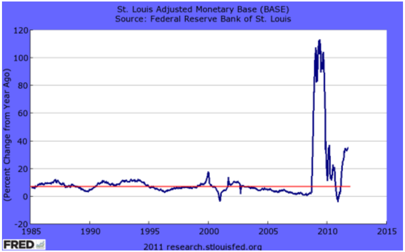
1. Initialise site parameters $\boldsymbol{\lambda}_1 \ldots \boldsymbol{\lambda}_n$. Global parameter: $\boldsymbol{\lambda} = \sum \boldsymbol{\lambda}_i$.
2. While not converged, loop over i:
   2.1 Form cavity: $\boldsymbol{\lambda}_{-i} = \boldsymbol{\lambda} - \boldsymbol{\lambda}_i$, hybrid $h_i(\boldsymbol{\theta}) \propto l_i(\boldsymbol{\theta}) \exp\left(s(\boldsymbol{\theta})^t \boldsymbol{\lambda}_{-i}\right)$
   2.2 Compute moments: $\boldsymbol{\eta}_i = E_{h_i}(s(\boldsymbol{\theta}))$ USING REJECTION ABC, transform back to natural parameters $\boldsymbol{\lambda}_i = \nu(\boldsymbol{\eta}_i) - \boldsymbol{\lambda}_{-i}$
   2.3 Update global approximation: $\boldsymbol{\lambda} = \boldsymbol{\lambda}_{-i} + \boldsymbol{\lambda}_i$

First example: alpha-stable distributions

- Alpha-stable densities are a class of univariate densities with potentially very heavy tails, that are popular in economics, because...

First example: alpha-stable distributions

- Alpha-stable densities are a class of univariate densities with potentially very heavy tails, that are popular in economics, because...

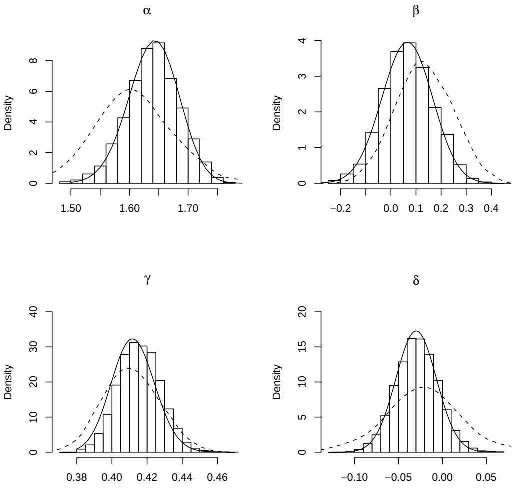


(fig. from Brad DeLong's blog).

Alpha-stable densities

- No closed-form for the density function.
- We take data: $n = 1200$ AUD/GBP log-returns computed from daily exchange rates.
- Data assumed IID from alpha-stable distribution with parameters $\boldsymbol{\theta}$
- $\boldsymbol{\theta}$ is α (tail heaviness), β (skewness),δ (location),γ (scale)

Results from alpha-stable example



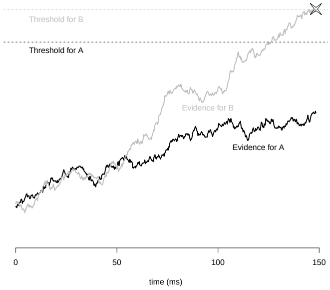
MCMC-ABC: 50 times more alpha-stable simulations than EP.
"Exact" MCMC takes 60 hours.

Example II: race model for reaction times

- Hierarchical model with independent data (but not IID)
- Subject must choose between k alternatives. Evidence $e_j(t)$ in favour of choice j follows a Brownian motion with drift:

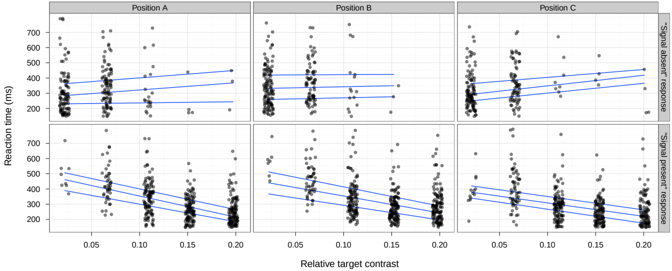
$$\tau de_j(t) = m_j dt + dW_t^j.$$

Decision is taken when one evidence "wins the race"; see plot.



Data

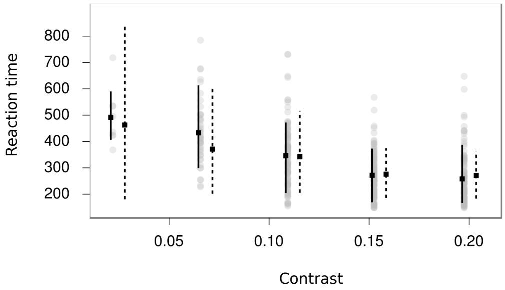
1860 Observations (courtesy of M Maertens, TU Berlin), from a single human being, who must choose between “signal absent”, and “signal present”.

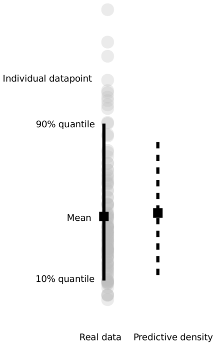


The hierarchical model

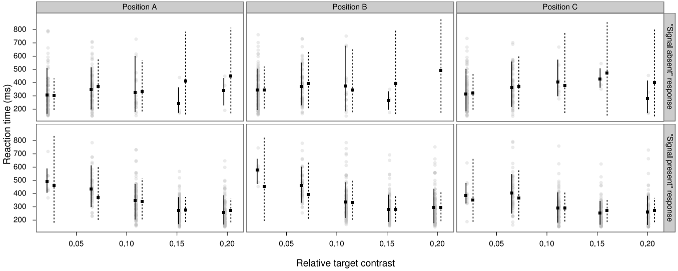
- The relative speed of the two racing diffusions changes according to experimental condition ($\approx$random effect).
- Global parameters: boundaries, noise variance.
- 33 parameters in total (3 shared, 30 condition-specific).
- 1860 observations, 30 subgroups.
- CPU time ~ 40 min

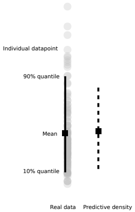
Reaction times: results





Reaction times: results





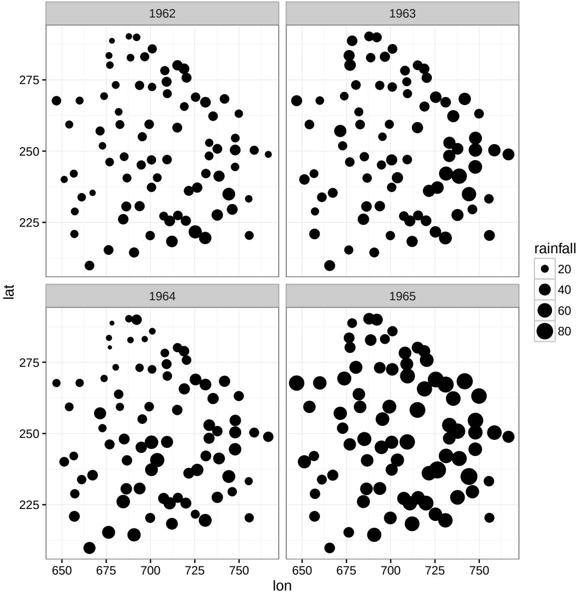
Reaction times: results

- Hierarchical model with ~ 30 parameters would be very challenging for standard ABC
- ABC-EP makes it do-able.
- Has actually been used again in an actual neuroscience paper (Park et al. 2016)

Example III: ABC-EP with summary statistics

- Sometimes we can't get rid of summary statistics entirely:
  - Spatial extremes: each observation is a set of extreme rainfall values over different weather stations
  - IID over years but not over stations (because of spatial dependencies)
  - We want to infer spatial dependencies

Swiss rainfall



ABC-EP in spatial extremes

- In recent work (Barthelmé, Chopin, Cottet, 2015) we suggest using ABC-EP with "local" summary statistics: summarise the observations over stations, but keep the successive years as IID sites
- Summary statistics is a robust estimate of spatial dependence: estimated value of a, b in following regression

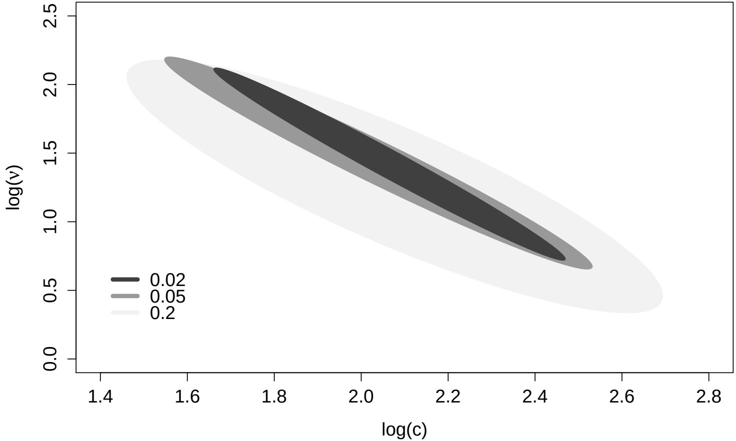
$$\log |F(y_i(x_j)) - F(y_i(x_k))| = a + b \log \|x_j - x_k\| + \epsilon_{jk}, \quad 1 \leq j < k \leq d$$

(F is the Fréchet CDF).

ABC-EP in spatial extremes

- We used the Swiss rainfall dataset (79 sites, 1962-2008).
- MCMC-ABC approach by Ehrardt & Smith (2012) essentially returns the prior after running for a week
- ABC-EP gives you something in about 3 hours
- Posterior is over the parameters of the covariance function (length-scale, height)

ABC-EP in spatial extremes



ABC-EP with summary statistics

- Even if you can't get rid of summary statistics entirely, you can get still choose summary statistics that are "local" and low-dimensional.
- Because you're still integrating the data bit-by-bit, the acceptance rate is high and you get considerable speed-ups over MCMC
- Caveat: in this example, it took us a while to find the right set of local summary statistics

ABC-EP in practice

- We are taking a deterministic algorithm and making it stochastic: have to be careful about Monte Carlo variance.

ABC-EP in practice

- We are taking a deterministic algorithm and making it stochastic: have to be careful about Monte Carlo variance.
- In the paper we highlight a set of tricks, among which:
  - Ways to "recycle" previous model simulations or exploit Markov structure

ABC-EP in practice

- We are taking a deterministic algorithm and making it stochastic: have to be careful about Monte Carlo variance.
- In the paper we highlight a set of tricks, among which:
  - Ways to "recycle" previous model simulations or exploit Markov structure
  - Quasi-Monte Carlo

ABC-EP in practice

- We are taking a deterministic algorithm and making it stochastic: have to be careful about Monte Carlo variance.
- In the paper we highlight a set of tricks, among which:
  - Ways to "recycle" previous model simulations or exploit Markov structure
  - Quasi-Monte Carlo
  - Rao-Blackwellisation A.K.A. Conditional Monte Carlo

ABC-EP in practice

- We are taking a deterministic algorithm and making it stochastic: have to be careful about Monte Carlo variance.
- In the paper we highlight a set of tricks, among which:
  - Ways to "recycle" previous model simulations or exploit Markov structure
  - Quasi-Monte Carlo
  - Rao-Blackwellisation A.K.A. Conditional Monte Carlo
- A lot of these tricks are model-specific and require a bit of work.

Conclusion on ABC-EP

- ABC-EP can bring tremendous speed improvements, but:
  - It is not trivial to implement
  - If your likelihood leads to a multi-modal posterior, the best you can do is recover one mode (hard to know in advance in ABC settings)
- When you switch from deterministic moment computations to Monte Carlo ones, stability becomes a problem
  - We'll see that matters a lot for the algorithms in the next part of this tutorial

MCMC in large datasets

- A lot of attention currently on how to scale MCMC to large datasets (Angelino, Johnson, Adams, 2016)
- As everybody knows, we need to parallelise
- In 2014 two groups came up with the same idea
  - Split datasets, run independent MCMC chains
  - Use EP to synchronise
- Xu, M., Lakshminarayanan, B., Teh, Y. W., Zhu, J., and Zhang, B. (2014). Distributed Bayesian posterior sampling via moment sharing. In Advances in Neural Information Processing Systems
- Gelman, A., Vehtari, A., Jylanki, P., Robert, C., Chopin, N., and Cunningham, J. P. (2014). Expectation propagation as a way of life. ArXiv:1412.4869

Parallel EP

- EP parallelises trivially
- All we need to do is compute all site updates in parallel rather than sequentially

Parallel EP in one slide

1. Initialise site parameters $\boldsymbol{\lambda}_1 \dots \boldsymbol{\lambda}_n$. Global parameter: $\boldsymbol{\lambda} = \sum \boldsymbol{\lambda}_i$.
2. While not converged:
   2.1 **Split:** For all i's , do:
      2.1.1 Form cavity: $\boldsymbol{\lambda}_{-i} = \boldsymbol{\lambda} - \boldsymbol{\lambda}_i$, hybrid $h_i(\boldsymbol{\theta}) \propto l_i(\boldsymbol{\theta}) \exp\left(s(\boldsymbol{\theta})^t \boldsymbol{\lambda}_{-i}\right)$
      2.1.2 Compute moments: $\boldsymbol{\eta}_i = E_{h_i}(s(\boldsymbol{\theta}))$, transform back to natural parameters $\boldsymbol{\lambda}_i \leftarrow \nu(\boldsymbol{\eta}_i) - \boldsymbol{\lambda}_{-i}$

   2.2 **Combine:** Update global approximation: $\boldsymbol{\lambda} \leftarrow \sum \boldsymbol{\lambda}_i$

MCMC-EP

- In our logistic regression application, a site was just a single datapoint and we could compute moments almost exactly using 1d quadrature
- What if we had sites with k datapoints?
  - We could maybe still do $k = 3$ using quadrature but it gets expensive
  - There's no hope of doing $k = 500$
  - Use MCMC!

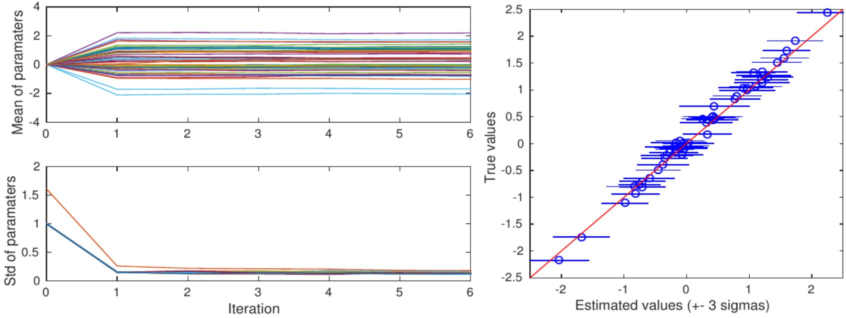
MCMC-EP

- There's an additional insight: in hierarchical models, you only need to synchronise *global, shared* parameters using EP.
- The parameters that are private to each batch can just be integrated over

MCMC-EP

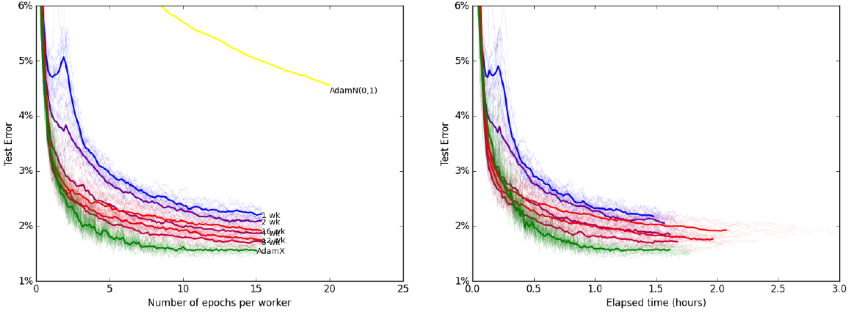
- Strategy is very simple: you have m workers and n datapoints.
  - Split dataset into m batches of $k \approx n/m$ datapoints.
  - Each hybrid is now a Gaussian pseudo-prior times k likelihood terms
  - Compute moments using m parallel MCMC chains over your workers
  - Update global approximation once everybody's done

Does it work?



From Gelman et al. Logistic regression example, $k = 50$, $n = 2500$

Does it work?



Teh et al. (2016). Fully connected neural net on MNIST dataset.

Conclusion on combining EP and MCMC

- The results so far are proof-of-concept
  - Either good results on toy models
  - Or so-so results on non-toy models
- Stability is a problem, just like in ABC-EP, which required model-specific work
  - Need better theory, right now we have a collection of hacks
- It's potentially very promising, but we are still quite far from effective black-box EP (The Stan team is working on it, though, pers. communication)
- Extension to sequential settings: De Freitas et al. (2015) (haven't had time to look at it yet)

aEP, sEP: Even more approximate EP

- aEP is a simpler version of EP we originally introduced to study asymptotic properties of EP (Dehaene & Barthelmé, 2015)
  - Forget about individual site parameters, use an average cavity parameter
  - Average cavity $\boldsymbol{\lambda}_c = \frac{n-1}{n}\boldsymbol{\lambda}$
- It has a nice interpretation as an approximate projection algorithm:
  - Form hybrids, approximate all hybrids as Gaussians
  - Average hybrids
- Same asymptotic properties as EP

Is aEP practical?

- We originally didn't make much of it, but noted
  - aEP is easier to implement
  - It cuts on the linear algebra by half
- Hernandez-Lobato et al. (2015) introduced stochastic EP (sEP)
  - essentially aEP with a random update schedule
  - obtained good results on neural networks
  - claim reduction in memory footprint (true but of limited consequence)

Is aEP practical?

- M. Beaumont (talk at NIPS) found aEP more stable in ABC setting
- We also found that generic fixed-point acceleration schemes (SQUAREM, Varadhan, 2014) worked well on aEP
- Easier in aEP than EP because there are far fewer parameters
- Potentially promising

General conclusions

- The original EP algorithm is extremely successful in GLMs, GAMs, etc., everybody should be using it
  - (still need quality implementation comparable to INLA or mgcv)
- EP is very promising as a generic black-box inference scheme
  - in ABC settings
  - for large hierarchical models
- Early days
  - Either a lot of model-specific work (ABC-EP)
  - Or proof-of-concept
  - aEP, sEP interesting direction

Things I wish I had time to mention

- Corrections to EP
  - find EP approximation and improve it using expansions
  - Reviewed in Opper's lecture notes
- Marginal likelihood approximation
- Double-loop algorithms
- EP for bilinear models
- EP for Gibbs distributions
  - e.g. Ising model, see Opper's lecture notes

Collaborators

Thanks to Nicolas Chopin, Vincent Cottet, Guillaume Dehaene, Gina Gruenhage, Manfred Opper