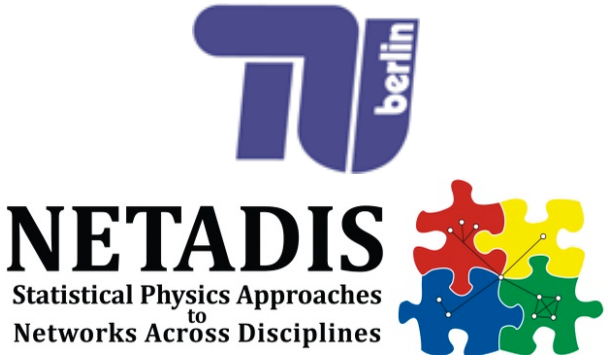
Expectation Propagation

Manfred Opper



TAP Gibbs free energy

- Set $\phi(x) = (x, -x^2)$, $\boldsymbol{\mu} = (\langle\phi(x_1)\rangle, \ldots, \langle\phi(x_1)\rangle)$

$$G_{\mathsf{TAP}}(\boldsymbol{\mu}) = G^{\mathsf{Gauss}}(\boldsymbol{\mu}) + G_0(\boldsymbol{\mu}) - G_0^{\mathsf{Gauss}}(\boldsymbol{\mu})$$

- Dual representation with $\lambda_i = (\gamma_i, \Lambda_i)$

$$\begin{aligned} Z^{\mathsf{Gauss}}(\lambda) &= \int d\mathbf{x} f_0(\mathbf{x})\, e^{\sum_{i=1}^N \lambda_i^\top \boldsymbol{\phi}(x_i)} \\ Z_0(\lambda) &= \int d\mathbf{x} \prod_i f_i(\mathbf{x})\, e^{\sum_{i=1}^N \lambda_i^\top \boldsymbol{\phi}(x_i)} \\ Z_0^{\mathsf{Gauss}}(\lambda) &= \int d\mathbf{x}\, e^{\sum_{i=1}^N \lambda_i^\top \boldsymbol{\phi}(x_i)} \end{aligned}$$

- Using $G(\boldsymbol{\mu}) = \max_\lambda \left\{ -\ln Z(\lambda) + \lambda^T \boldsymbol{\mu} \right\}$ and setting $\nabla_{\boldsymbol{\mu}} G_{\mathsf{TAP}}(\boldsymbol{\mu}) = 0$ we get

  $-\ln Z = -\ln Z_{\mathsf{TAP}} = -\ln Z^{\mathsf{Gauss}}(\lambda_1) - \ln Z_0(\lambda_2) + \ln Z_0^{\mathsf{Gauss}}(\lambda_1 + \lambda_2)$

  where the right hand side must be made stationary wrt the λ_i.

Double loop algorithms

Approximate Gibbs Free Energies are often not convex.

In many cases, free energies have the form $G_{\text{approx}}(\boldsymbol{\mu}) = G_A(\boldsymbol{\mu}) - G_B(\boldsymbol{\mu})$ with $G_{A,B}$ convex.

The following type of algorithm is guaranteed not to increase G_{approx}.

Repeat:

- Upper bound concave function $-G_B$ by linear function
  $-G_B(\boldsymbol{\mu}) \leq L(\boldsymbol{\mu}) = -G_B(\boldsymbol{\mu}_{\text{old}}) - (\boldsymbol{\mu} - \boldsymbol{\mu}_{\text{old}})\nabla G_B(\boldsymbol{\mu}_{\text{old}})$

- Minimise convex function $G_A(\boldsymbol{\mu}) + L(\boldsymbol{\mu})$ and get $\rightarrow \boldsymbol{\mu}_{\text{new}}$

Improving the accuracy of EP

- Choosing more structured families $q(\mathbf{x})$:

  1. For discrete models on graphs, replace factorizing q by tree (Minka & Qi, 2004).

  2. Use Gaussian models with tree consistency.

- Corrections after EP converges

EP with tree consistency & corrections

Let $\mathcal{T}$ be a tree. Rewrite Gaussian latent variable (e.g. Ising) as

$$p(\mathbf{x}) = \frac{1}{Z} \exp[-\frac{1}{2}\mathbf{x}^\top \mathbf{K}^{-1}\mathbf{x}] \frac{\prod_{(m,n)\in\mathcal{T}} f_m(x_m) f_n(x_n)}{\prod_n f_n(x_n)^{d_n-1}}$$

approximated by Gaussian

$$q(\mathbf{x}) = \exp[-\frac{1}{2}\mathbf{x}^\top \mathbf{K}^{-1}\mathbf{x}] \frac{\prod_{(m,n)\in\mathcal{T}} g_{m,n}(x_m, x_n)}{\prod_n g_n(x_n)^{d_n-1}}$$

Consistency on $\langle x_n \rangle$, $\langle x_n^2 \rangle$ and $\langle x_m x_n \rangle$ for $(m,n) \in \mathcal{T}$ is required.

Relating EP and exact model

- Exact distribution

$$p(\mathbf{x}) = \frac{1}{Z} \prod_n f_n(\mathbf{x})$$

- EP approximation

$$q(\mathbf{x}) = \frac{1}{Z_q} \prod_n g_n(\mathbf{x})$$

- tilted distribution

$$q_n(\mathbf{x}) = \frac{1}{Z_n} \left(\frac{q(\mathbf{x}) f_n(\mathbf{x})}{g_n(\mathbf{x})} \right) .$$

- Solving for f_n yields

$$\prod_n f_n(\mathbf{x}) = \prod_n \left(\frac{Z_n q_n(\mathbf{x}) g_n(\mathbf{x})}{q(\mathbf{x})} \right) = Z_{EP}\, q(\mathbf{x}) \prod_n \left(\frac{q_n(\mathbf{x})}{q(\mathbf{x})} \right)$$

- with the definition of the EP free energy

$$Z_{EP} = Z_q \prod_n Z_n \ .$$

- Define

$$F(\mathbf{x}) \equiv \prod_n \left(\frac{q_n(\mathbf{x})}{q(\mathbf{x})} \right)$$

then

$$R = Z/Z_{EP} = \int q(\mathbf{x}) F(\mathbf{x})\, \mathrm{d}\mathbf{x} \ ,$$

- Similarly we can write:

$$p(\mathbf{x}) = \frac{1}{Z} \prod_n f_n(\mathbf{x}) = \frac{Z_{EP}}{Z}\, q(\mathbf{x}) F(\mathbf{x}) = \frac{1}{R}\, q(\mathbf{x}) F(\mathbf{x}) \ .$$

Expansion I: Clusters

- Assume $\varepsilon_n(\mathbf{x}) = \frac{q_n(\mathbf{x})}{q(\mathbf{x})} - 1$ to be **typically small**. Expand products

$$p(\mathbf{x}) = \frac{q(\mathbf{x})\left(1 + \sum_n \varepsilon_n(\mathbf{x}) + \sum_{n_1<n_2} \varepsilon_{n_1}(\mathbf{x})\varepsilon_{n_2}(\mathbf{x}) + \ldots\right)}{1 + \sum_{n_1<n_2} \langle \varepsilon_{n_1}(\mathbf{x})\varepsilon_{n_2}(\mathbf{x})\rangle_q + \ldots},$$

  in terms of growing clusters of "interacting" variables $\varepsilon_n(\mathbf{x})$.

- In a similar way

$$\frac{Z}{Z_{EP}} = 1 + \sum_{n_1<n_2} \left\langle \varepsilon_{n_1}(\mathbf{x})\varepsilon_{n_2}(\mathbf{x})\right\rangle_q + \sum_{n_1<n_2<n_3} \left\langle \varepsilon_{n_1}(\mathbf{x})\varepsilon_{n_2}(\mathbf{x})\varepsilon_{n_3}(\mathbf{x})\right\rangle_q + \ldots$$

  First order $\sum_n \langle \varepsilon_n(\mathbf{x})\rangle_q = 0$.

First order correction to posterior

Correction to posterior in first order $\varepsilon_n(\mathbf{x})$ is simple:

$$p(\mathbf{x}) \approx \sum_n q_n(\mathbf{x}) - (N-1)q(\mathbf{x}) \ .$$

Does not require computation of expectations !

In an similar spirit (Czeke & Heskes) one gets for the marginal

$$p(x_i) \approx q_i(x_i) \prod_{j \neq i} \int dx_j q(x_j|x_i) \frac{f_j(x_j)}{g_j(x_j)}$$

Example I: Bayesian mixture of Gaussians

- **Likelihood term** for data point ζ_n:

$$f_n(\mathbf{x}) = \sum_\kappa \pi_\kappa \mathcal{N}(\zeta_n; \boldsymbol{\mu}_\kappa, \Gamma_\kappa^{-1})$$

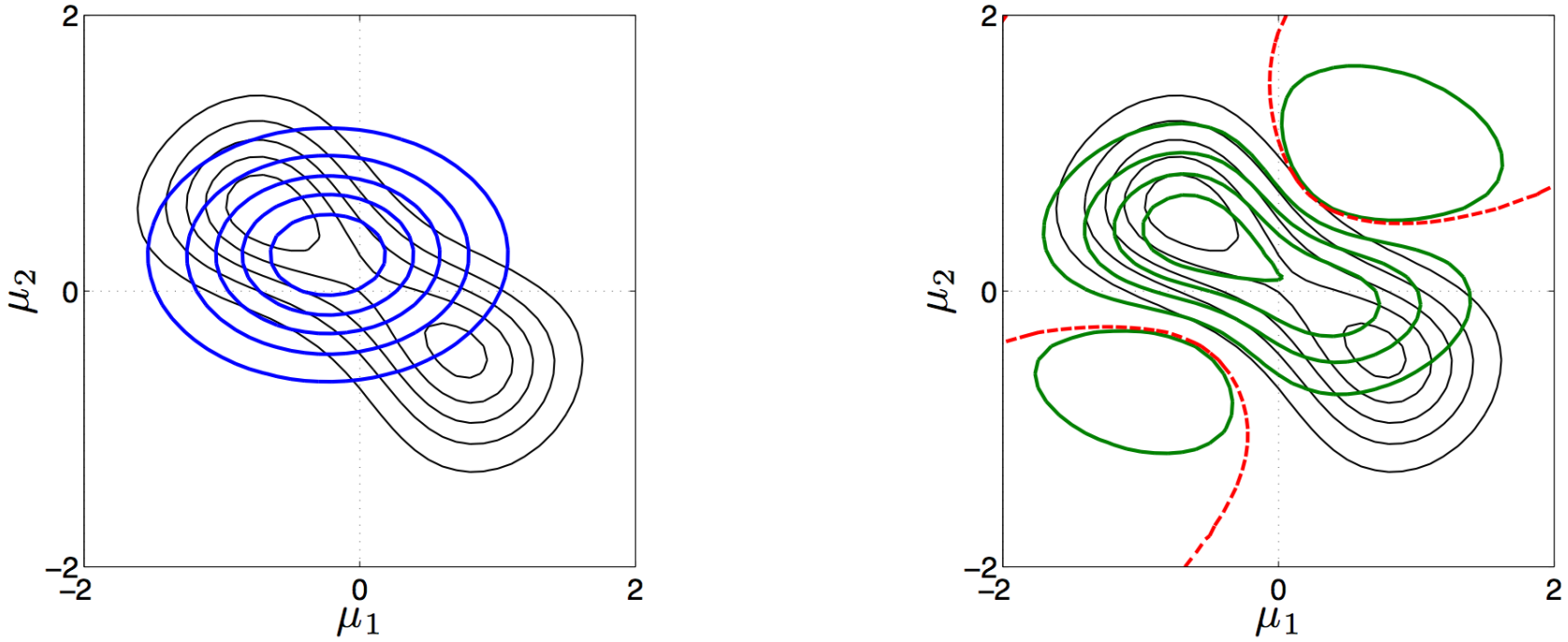
- **Latent variables:** $\mathbf{x} = \{\pi_\kappa, \boldsymbol{\mu}_\kappa, \Gamma_\kappa\}_{\kappa=1}^K$ (weights, means, precision matrix)

- **Prior** $f_0(\mathbf{x}) = \mathcal{D}(\pi) \prod_\kappa \mathcal{NW}(\boldsymbol{\mu}_\kappa, \Gamma_\kappa)$.

- **Posterior** $p(\mathbf{x}|\zeta_1, \ldots, \zeta_N) = \frac{1}{Z} \prod_{n \geq 0} f_n(\mathbf{x})$

- $q(\mathbf{x}) = \mathcal{D}(\pi) \prod_\kappa \mathcal{NW}(\boldsymbol{\mu}_\kappa, \Gamma_\kappa)$ follows prior.

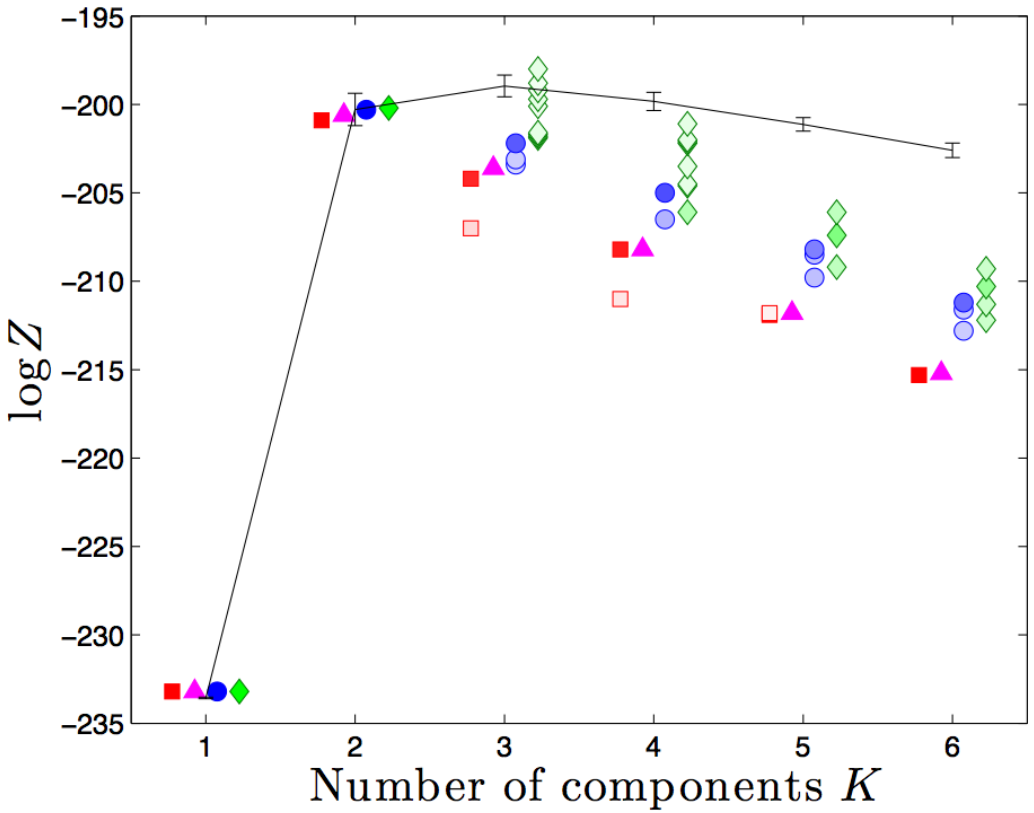
Results I: 1st order cluster corrections to posteriors



Posterior for **toy mixture model** (grey lines: exact).

Left: EP, **right:** 1st order correction

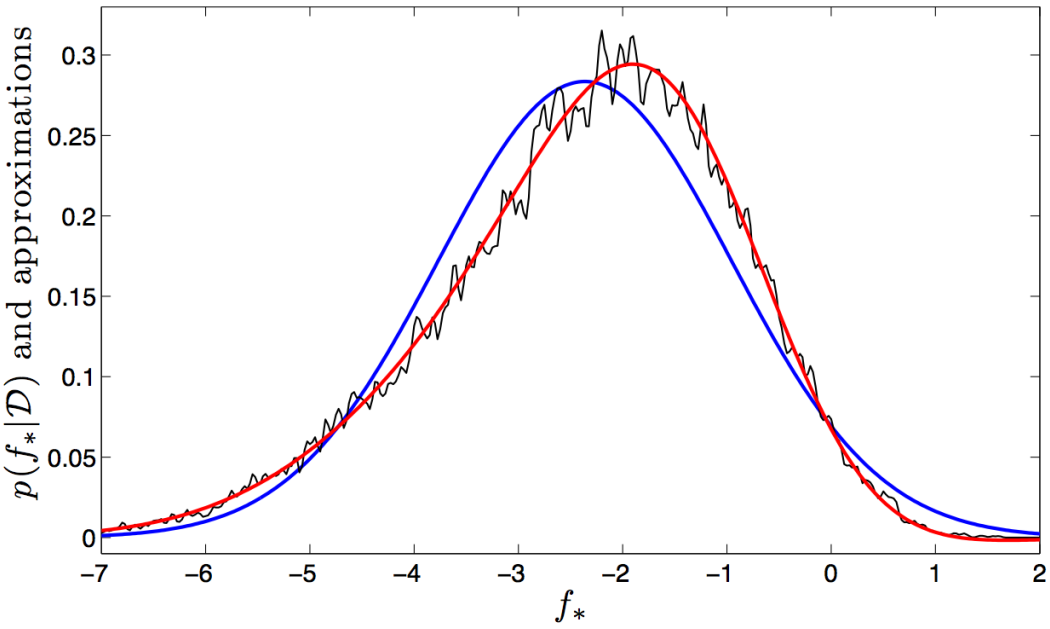
Results II: 2nd order cluster corrections to $\ln Z$



Gaussian mixture model on *acidity data set*

Variational Bayes (red squares), Minka's $\alpha = (\frac{1}{2})$-divergence message passing (magenta triangles); EP (blue circles); EP with the 2^{nd} order correction (green diamonds).

Corrections may lead to changes in estimation of model order !



Marginal posterior at test point for toy **Gaussian process classification**,

blue: EP and **red**: 1. order correction compared to MCMC estimate (grey).

Expansion II: Cumulants for latent Gaussian models

- Latent Gaussian models

$$p(\mathbf{x}) = \frac{1}{Z} \exp[-\frac{1}{2}\mathbf{x}^\top \mathbf{K}^{-1}\mathbf{x}] \prod_n f_n(x_n) \approx$$
$$q(\mathbf{x}) \propto \exp[-\frac{1}{2}\mathbf{x}^\top \mathbf{K}^{-1}\mathbf{x}] \prod_n e^{\gamma_n x - \frac{1}{2}\lambda_n x^2}$$

- The tilted density is defined as

$$q_n(\mathbf{x}) = \frac{1}{Z_n} \left(\frac{q(\mathbf{x}) f_n(x_n)}{g_n(x_n)} \right)$$

- Simplify correction to partition function

$$R = \int d\mathbf{x}\, q(\mathbf{x}) \prod_n \left(\frac{q_n(\mathbf{x})}{q(\mathbf{x})} \right) = \int d\mathbf{x}\, q(\mathbf{x}) \prod_n \left(\frac{q(\mathbf{x}_{\setminus n}|x_n) q_n(x_n)}{q(\mathbf{x}_{\setminus n}|x_n) q(x_n)} \right)$$
$$= \int d\mathbf{x}\, q(\mathbf{x}) \prod_n \left(\frac{q_n(x_n)}{q(x_n)} \right)$$

- $q_n(x)$ and $q(x)$ agree in 1. and 2. cumulant.

- Possible assumption: **Higher cumulants are small**. Try an expansion in these higher cumulants !

Cumulants

- Characteristic function

$$q_n(x_n) = \int_{-\infty}^{\infty} \frac{dk}{(2\pi)}\, e^{-ikx_n} \chi_n(k)$$

- Cumulants are defined by

$$\ln \chi_n(k) = \sum_l (i)^l \frac{c_{nl}}{l!}\, k^l = i m_n k - \frac{1}{2} S_n k^2 + r_n(k)$$

The term $r_n(k) = \sum_{l \geq 3} (i)^l \frac{c_{ln}}{l!}\, k^l$ contains the contributions of all **higher cumulants**.

Cumulant expansion for partition function

- We have

$$\frac{q_n(x_n)}{q(x_n)} = \sqrt{\frac{S_{nn}}{2\pi}}\, e^{\frac{(x_n-m_n)^2}{2S_{nn}}} \int_{-\infty}^{\infty} \frac{dk}{(2\pi)}\, e^{-ikx_n} \chi_n(k) =$$
$$\int_{-\infty}^{\infty} d\eta_n \sqrt{\frac{S_{nn}}{2\pi}}\, \exp\left[-\sum_n \frac{S_{nn}\eta_n^2}{2}\right]\, \exp\left[r_n\left(\eta_n - i\frac{(x_n-m_n)}{S_{nn}}\right)\right]$$

- To get

$$\frac{Z}{Z_{EP}} = E_q\left[\prod_n \left(\frac{q_n(x_n)}{q(x_n)}\right)\right]$$

we must take expectation over the multivariate Gaussian $q(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \mathbf{m}, \mathbf{S})$.

- Introduce **complex** zero mean Gaussian random vector

$$z_n = \eta_n - i\frac{x_n - m_n}{S_{nn}}$$

with

$$\begin{aligned} \langle z_i z_j \rangle_{\mathbf{z}} &= -\frac{S_{ij}}{S_{ii}S_{jj}} \qquad i \neq j \\ \langle z_i^2 \rangle_{\mathbf{z}} &= 0 \end{aligned}$$

- Then

$$\frac{Z}{Z_{EP}} = \left\langle \exp\left[\sum_n r_n(z_n)\right] \right\rangle_{\mathbf{z}}$$

Power series expansion

for small c_{ln} ($l > 2$):

$$\ln\left(\frac{Z}{Z_{EP}}\right) = \ln\frac{Z}{Z_{EP}} = \ln\left\langle \exp\left[\sum_n r_n\left(z_n\right)\right]\right\rangle_{\mathbf{z}}$$

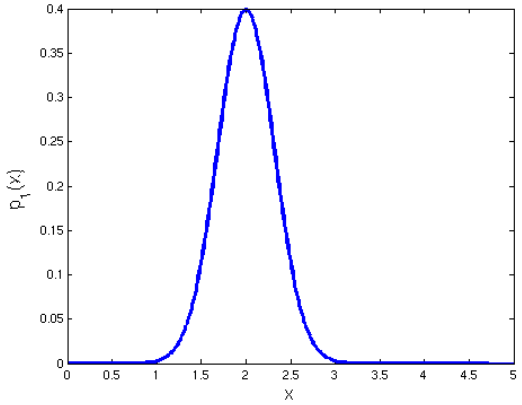
$$= \frac{1}{2}\sum_{m\neq n}\langle r_m r_n\rangle_{\mathbf{z}} \pm \ldots \qquad = \sum_{m\neq n}\sum_{l\geq 3}\frac{c_{ln}c_{lm}}{l!}\left(\frac{S_{nm}}{S_{nn}S_{mm}}\right)^l \pm \ldots$$

No "self interactions" (loops) ! This indicates that corrections may not scale with N.

Conjecture: EP is fairly accurate if:

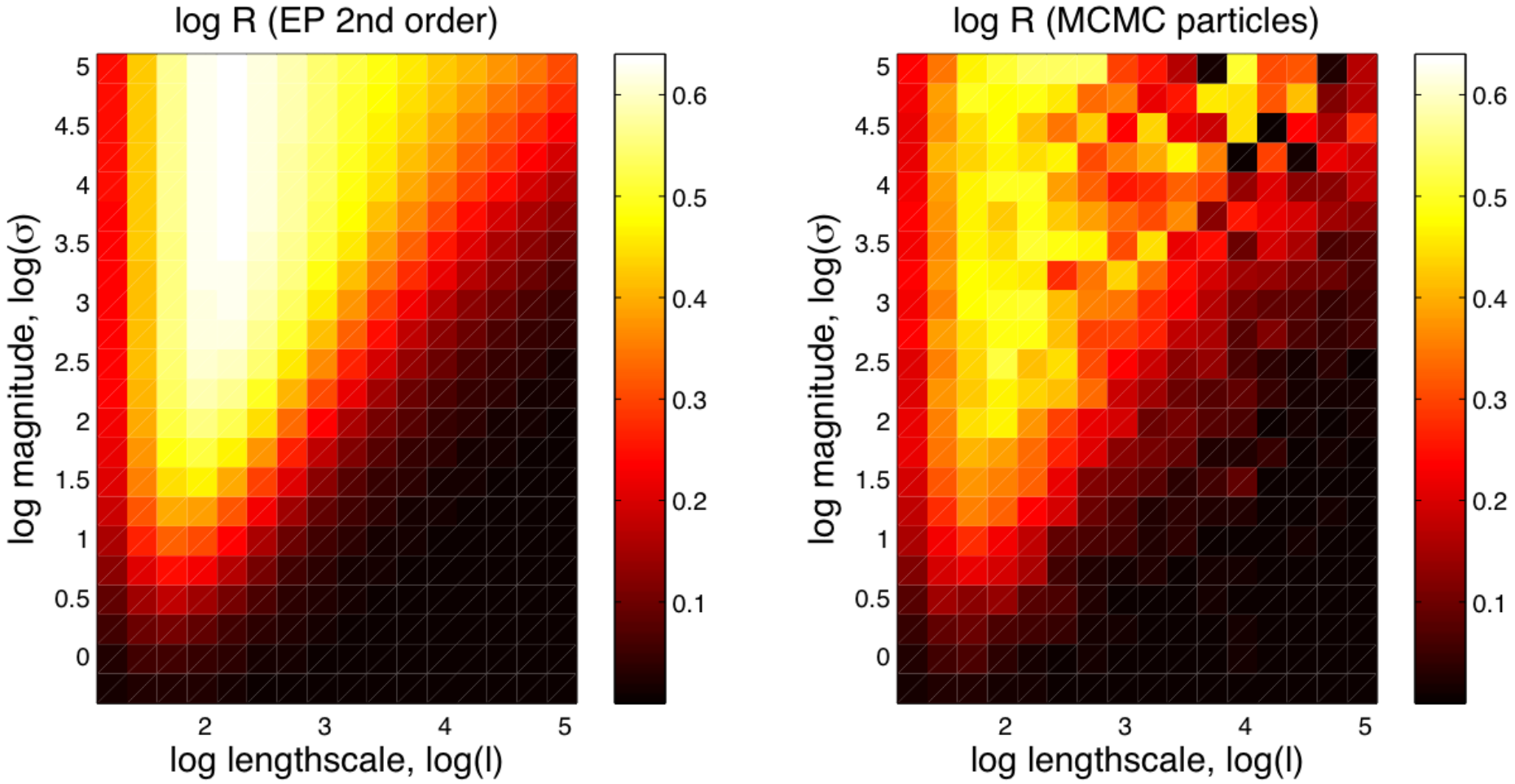
- the cumulants c_{ln} are small. This holds possibly for classification likelihoods $f_i(x_i) = \Theta(y_i x_i)$, when posterior variance small compared to the mean.

  The marginal $q(x_i)$ might look like this

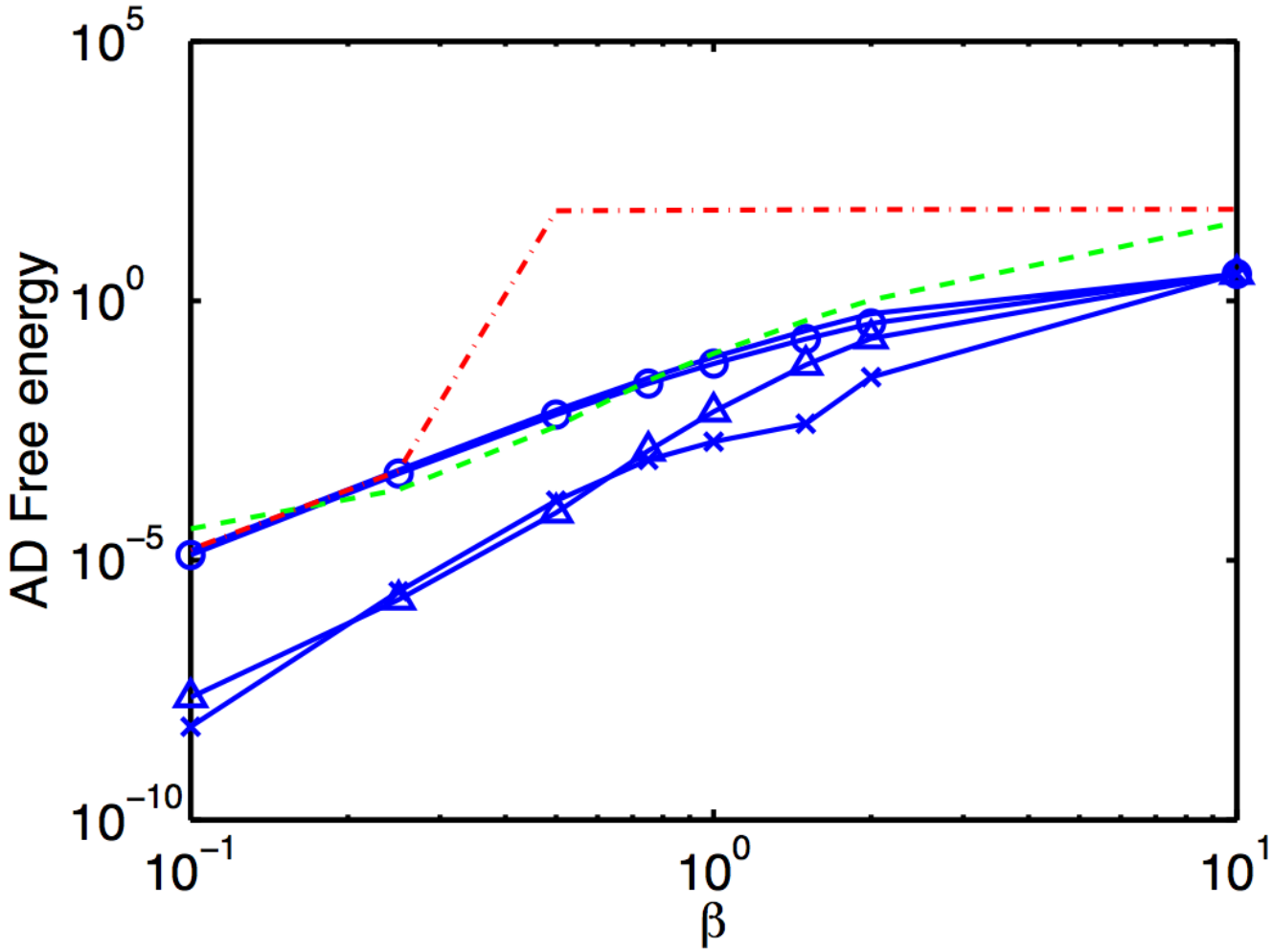


- if posterior covariances S_{ij} small for $i \neq j$.

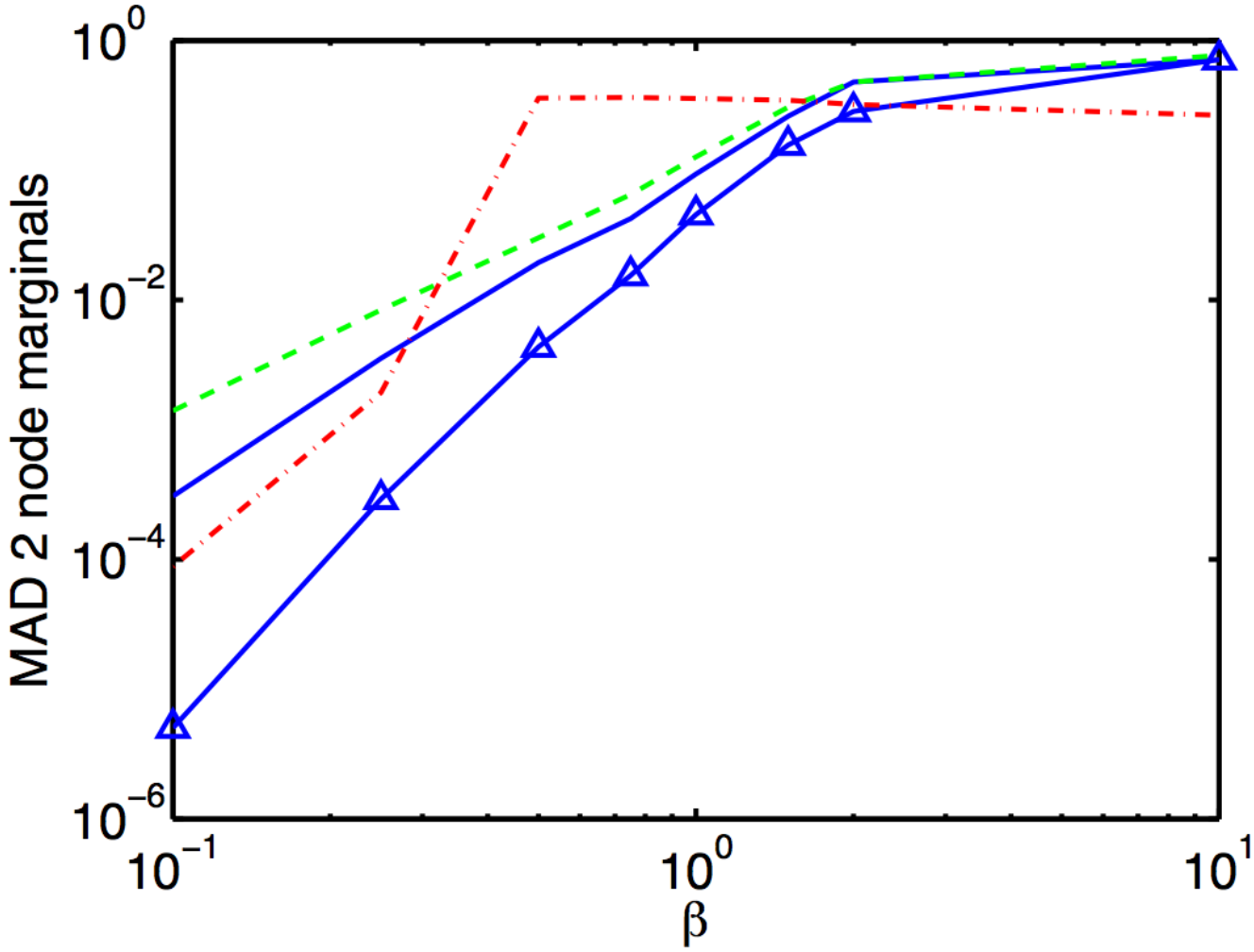
Cumulant corrections



$\ln\left(\frac{Z}{Z_{EP}}\right)$ for GP classification on USPS data
left: analytical, **right:** Monte Carlo



Ising with $N = 10$ (**J** random, variance β^2). Error on $-\ln Z$ (Ising, 2nd order, $l = 3, 4, 5$) for EP (blue), EP 2^{nd} order $l = 4$ corrections (blue with triangles), loopy BP (dashed green) and Kikuchi or generalized LBP (dash–dotted red).



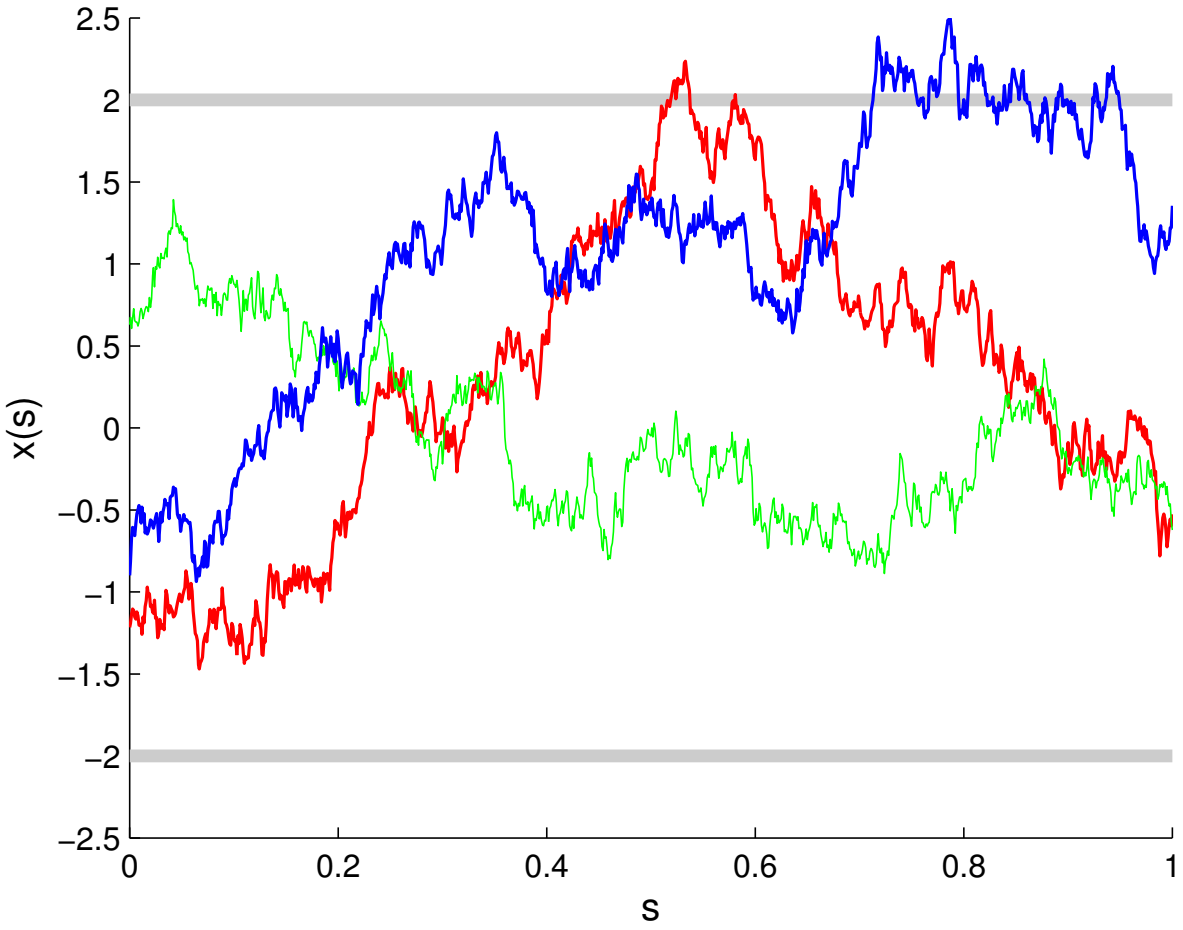
Ising with $N = 10$ ($\mathbf{J}$ random, variance β^2). Error on covariance matrix for EP (blue), EP 2^{nd} order $l = 4$ corrections (blue with triangles), loopy BP (dashed green) and Kikuchi or generalized LBP (dash–dotted red).

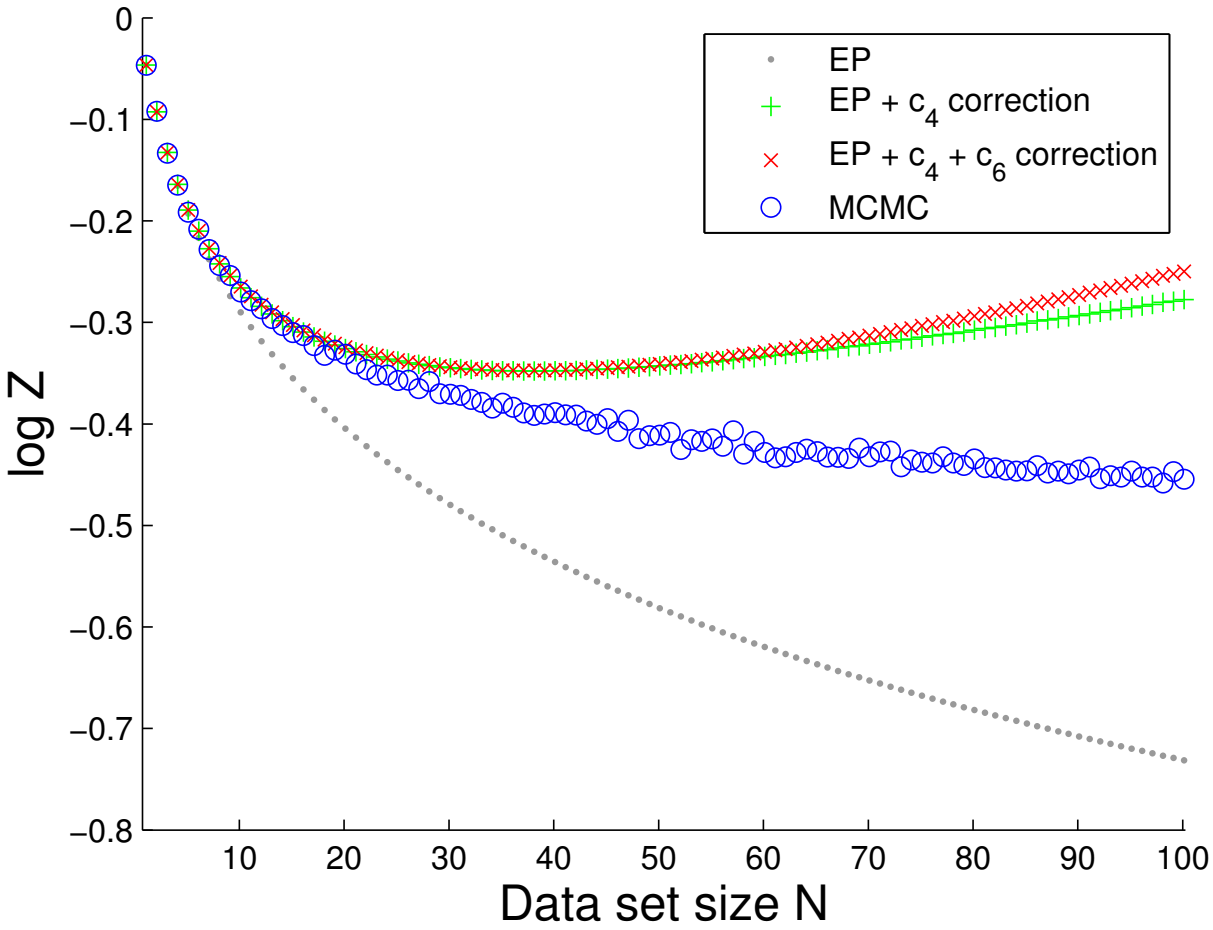
Graph	Coupling	d_{coup}	EP	EP c	EP t	EP tc
Full	Repulsive	0.25	0.0310	0.0018	0.0104	0.0010
	Repulsive	0.50	0.3358	0.0639	0.1412	0.0440
	Mixed	0.25	0.0235	0.0013	0.0129	0.0009
	Mixed	0.50	0.3362	0.0655	0.1798	0.0620
	Attractive	0.06	0.0236	0.0028	0.0166	0.0006
	Attractive	0.12	0.8297	0.1882	0.2672	0.2094
Grid	Repulsive	1.0	1.7776	0.8461	0.0279	0.0115
	Repulsive	2.0	4.3555	2.9239	0.0086	0.0077
	Mixed	1.0	0.3539	0.1443	0.0133	0.0039
	Mixed	2.0	1.2960	0.7057	0.0566	0.0179
	Attractive	1.0	1.6114	0.7916	0.0282	0.0111
	Attractive	2.0	4.2861	2.9350	0.0441	0.0433

Average absolute deviation of ln Z function in a Wainwright-Jordan set-up ($N = 16$), comparing EP, EP with $l = 4$ second order correction (EP c), EP tree (EP t) and EP tree with $l = 4$ second order correction (EP tc).

GP in a box

$$p(\mathbf{x}) = \frac{1}{Z} \prod_n \mathbb{I}\Big[|x_n| < a\Big] \mathcal{N}(\mathbf{x}\,;\,\mathbf{0},\,\mathbf{K}) \ . \tag{1}$$




EP
EP + c4 correction
EP + c4 + c6 correction
MCMC
log Z
Data set size N
0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.6
−0.7
−0.8
10
20
30
40
50
60
70
80
90
100

TrueSkill™ Ranking System

The TrueSkill™ ranking system is a skill based ranking system for Xbox Live developed at Microsoft Research.

The TrueSkill ranking system is a skill based ranking system for Xbox Live developed at Microsoft Research. The purpose of a ranking system is to both identify and track the skills of gamers in a game (mode) in order to be able to match them into competitive matches. The TrueSkill ranking system only uses the final standings of all teams in a game in order to update the skill estimates (ranks) of all gamers playing in this game. Ranking systems have been proposed for many sports but possibly the most prominent ranking system in use today is ELO.

Ranking Players

So, what is so special about the TrueSkill ranking system? In short, the biggest difference to other ranking systems is that in the TrueSkill ranking system skill is characterised by **two** numbers:

- The average skill of the gamer (**μ** in the picture).
- The degree of uncertainty in the gamer's skill (**σ** in the picture).

List of further EP applications

`http://research.microsoft.com/en-us/um/people/minka/papers/ep/roadmap.htm`

Bootstrap estimators for Gaussian process regression models

- Goal: Estimate average case properties (test errors) of statistical estimator

  $E[x_i|D]$.

- Bootstrap: Generate pseudo data via resampling with replacement, replace true (unknwon) distribution by emperirical distribution.

  Problem: Each sample requires new running of algorithm.

- Try approximate analytical approach instead!

Approximate Analytical Bootstrap

- Bootstrap generalization error

$$\varepsilon(m) \doteq \frac{1}{N} \sum_{i=1}^{N} \frac{E_D\left[\delta_{m_i,0}\left(E[x_i|D] - y_i\right)^2\right]}{E_D\left[\delta_{m_i,0}\right]}$$

  with m_i random "occupation number" of datapoint i.

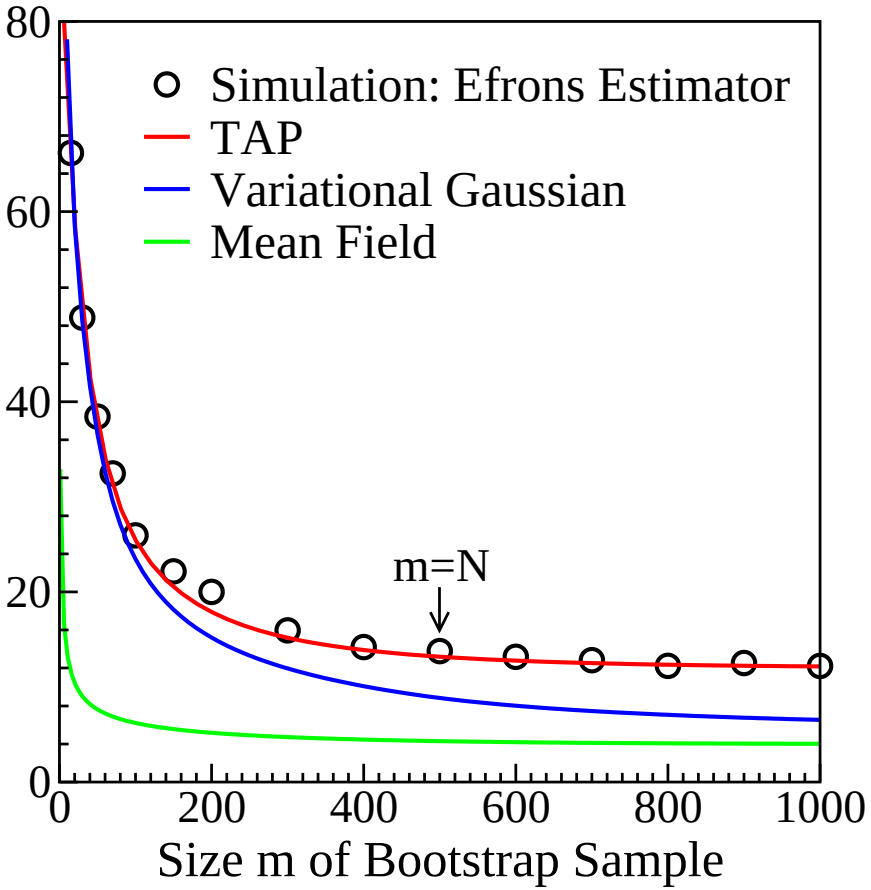
- Exact average with replica trick

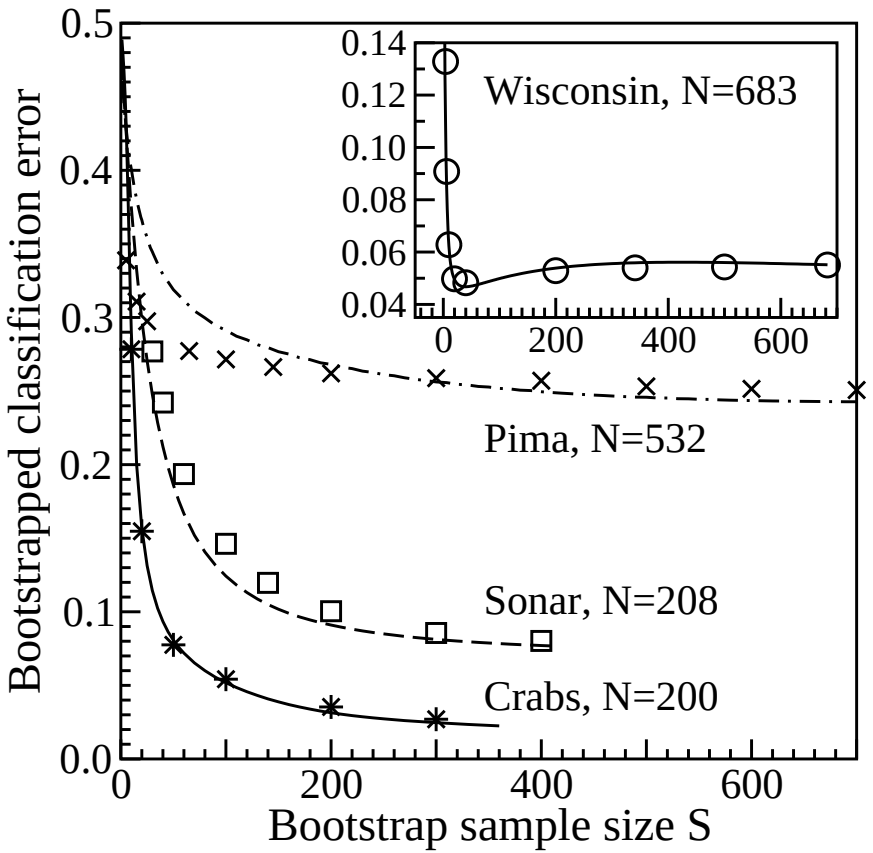
$$\varepsilon(m) = \lim_{n\to 0} \frac{1}{e^{-m/N} N} \sum_{i=1}^{N} E_D\Big[\delta_{m_i,0}\; Z^{n-2}$$

$$\int d\mathbf{x}^1 d\mathbf{x}^2\; p_0(\mathbf{x}^{(1)})\; p_0(\mathbf{x}^{(2)})\; P(D|\mathbf{x}^{(1)})\; P(D|\mathbf{x}^{(2)})\;\; (x_i^{(1)} - y_i)\; (x_i^{(2)} - y_i)\Big]$$

- Perform EP inference and let $n \to 0$.

Results for Regression and SVM classification





Generalized Models

$$p(\mathbf{x}) \propto \prod_{i=1}^{N} f_i(x_i) \exp\left[\sum_{i<j}^{N} x_i J_{ij} x_j\right] \prod_{k=1}^{m} F\left(\sum_{i=1}^{N} \hat{J}_{ik} x_k\right)$$

can be cast into the form

$p(\sigma) \propto \prod_i \rho_i(\sigma_i)\ \exp\left[\sum_{i<j} \sigma_i A_{ij} \sigma_j\right]$.

with augmented set of "random variables"

$\sigma = (\mathbf{x}, \hat{\mathbf{x}})$

and terms $\rho_i(\sigma_i) = f_i(x_i)\hat{f}_i(\hat{x})$, where

$\hat{f}_i(\hat{x}) = \int \frac{dh}{2\pi i} e^{-\hat{x}h} F_i(h)$

The augmented coupling matrix is

$$\mathbf{A} = \begin{pmatrix} \mathbf{J} & \hat{\mathbf{J}} \\ \hat{\mathbf{J}}^T & 0 \end{pmatrix} .$$

Inference in continuous time stochastic dynamics

- Prior process (Ornstein–Uhlenbeck)

$$d\boldsymbol{x}_t = (\boldsymbol{A}_t \boldsymbol{x}_t + \boldsymbol{c}_t)dt + \boldsymbol{B}_t^{1/2} d\boldsymbol{W}_t,$$

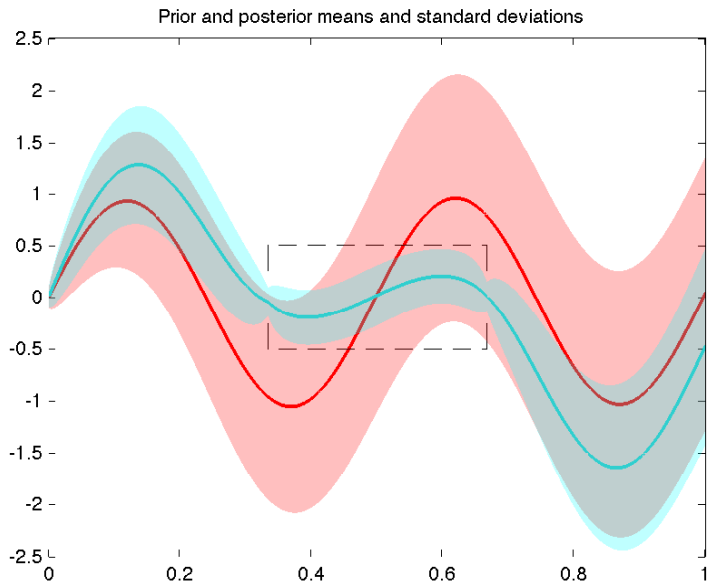
- Likelihood for continuous and discrete time observations

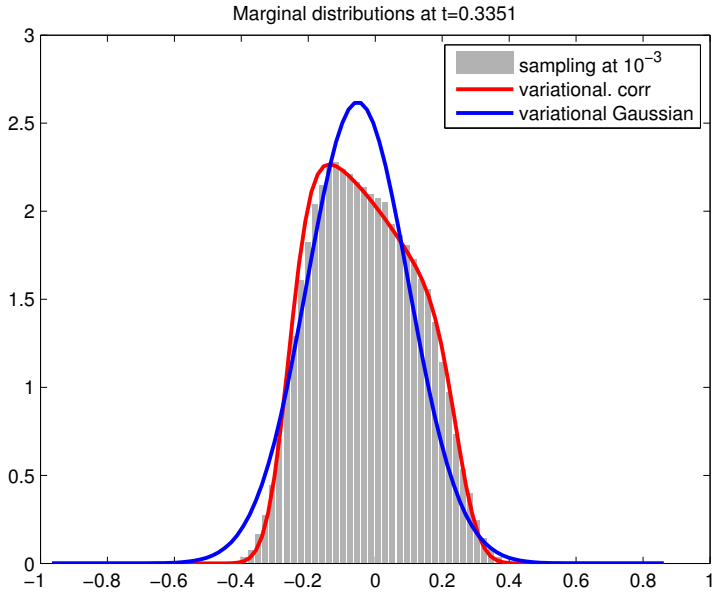
$$p(\{\boldsymbol{y}^d_{t_i}\}_i, \{\boldsymbol{y}^c_t\} | \{\boldsymbol{x}_t\}) \propto \prod_{t_i \in T_d} p(\boldsymbol{y}^d_{t_i} | \boldsymbol{x}_{t_i}) \times \exp\left\{ - \int_0^1 dt V(t, \boldsymbol{y}^c_t, \boldsymbol{x}_t) \right\}$$

- Time discretized version

$$p(\{\boldsymbol{y}^d_{t_i}\}_i, \boldsymbol{y}^c, \boldsymbol{x}) = p_0(\boldsymbol{x}) \times \prod_i p(\boldsymbol{y}^d_{t_i} | \boldsymbol{x}_{t_i}) \prod_k \exp\left\{ -\Delta t_k V(t_k, \boldsymbol{y}^c_{t_k}, \boldsymbol{x}_{t_k}) \right\}$$

- Does EP survive the $\Delta t \to 0$ limit ?





The continuous time potential is defined as $V(t, x_t) = (2x_t)^8 I_{[1/2,2/3]}(t)$ and we assume two hard box discrete likelihood terms $\mathrm{I}_{[-0.25,0.25]}(x_{t_1})$ and $\mathrm{I}_{[-0.25,0.25]}(x_{t_2})$ placed at $t_1 = 1/3$ and $t_2 = 2/3$. The prior is defined by the parameters $a_t = -1$, $c_t = 4\pi \cos(4\pi t)$ and $b_t = 4$.

Some open problems

- Scaling up to large systems when approximations are structured, parallelization.
- Convergence properties
- Bounds on free energies
- Performance bounds (PAC–Bayes ?)