Nonparametric Bayesian Models for Sparse Matrices and Covariances

Zoubin Ghahramani

Department of Engineering
University of Cambridge, UK

zoubin@eng.cam.ac.uk
http://learning.eng.cam.ac.uk/zoubin/

Bayes 250

Edinburgh 2011

Bayesian Machine Learning

Everything follows from two simple rules:

Sum rule: $P(x) = \sum_y P(x,y)$

Product rule: $P(x,y) = P(x)P(y|x)$

$$P(\theta|\mathcal{D}) = \frac{P(\mathcal{D}|\theta)P(\theta)}{P(\mathcal{D})}$$

$P(\mathcal{D}\|\theta)$	likelihood of θ
$P(\theta)$	prior probability of θ
$P(\theta\|\mathcal{D})$	posterior of θ given $\mathcal{D}$

Prediction:

$$P(x|\mathcal{D},m) \quad = \quad \int P(x|\theta,\mathcal{D},m)P(\theta|\mathcal{D},m)d\theta$$

Model Comparison:

$$\begin{aligned} P(m|\mathcal{D}) &= \frac{P(\mathcal{D}|m)P(m)}{P(\mathcal{D})} \\ P(\mathcal{D}|m) &= \int P(\mathcal{D}|\theta,m)P(\theta|m)\,d\theta \end{aligned}$$

Myths and misconceptions about Bayesian methods

- **Bayesian methods make assumptions where other methods don't**
  All methods make assumptions! Otherwise it's impossible to predict. Bayesian methods are transparent in their assumptions whereas other methods are often opaque.

- **If you don't have the right prior you won't do well**
  Certainly a poor model will predict poorly but there is no such thing as the right prior! Your model (both prior and likelihood) should capture a reasonable range of possibilities. When in doubt you can choose vague priors (cf nonparametrics).

- **Maximum A Posteriori (MAP) is a Bayesian method**
  MAP is similar to regularization and offers no particular Bayesian advantages. The key ingredient in Bayesian methods is to average over your uncertain variables and parameters, rather than to optimize.

Myths and misconceptions about Bayesian methods

- **Bayesian methods don't have theoretical guarantees**
  One can often apply frequentist style generalization error bounds to Bayesian methods (e.g. PAC-Bayes). Moreover, it is often possible to prove convergence, consistency and rates for Bayesian methods.

- **Bayesian methods are generative**
  You can use Bayesian approaches for both generative and discriminative learning (e.g. Gaussian process classification).

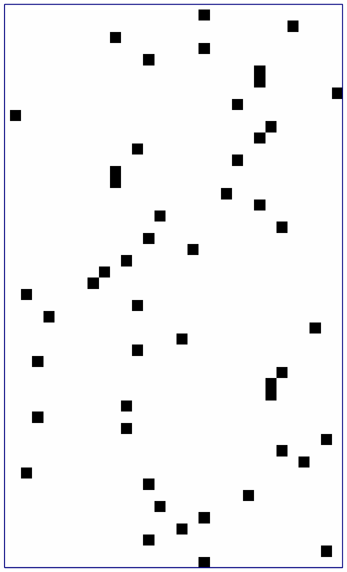
- **Bayesian methods don't scale well**
  With the right inference methods (variational, MCMC) it is possible to scale to very large datasets (e.g. excellent results for Bayesian Probabilistic Matrix Factorization on the Netflix dataset using MCMC), but it's true that averaging/integration is often more expensive than optimization.

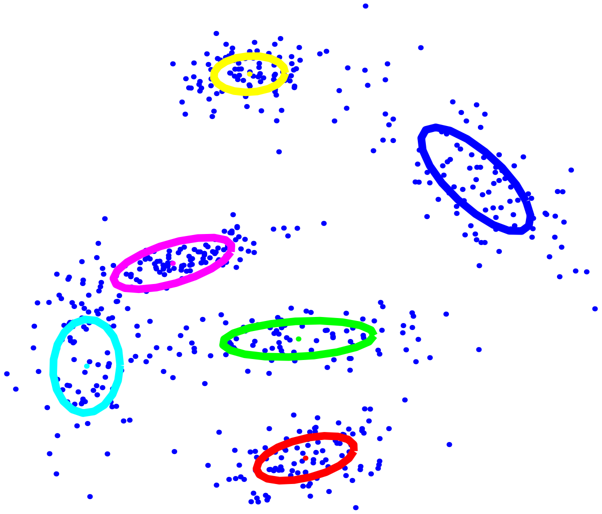
Non-parametric Bayesian Models

- Real-world phenomena are **complicated** and we don't really believe simple and inflexible models (e.g. a low-order polynomial or small mixture of Gaussians) can adequately model them.
- Non-parametric models are designed to be very **flexible**; many can be derived by taking the limit as the number of parameters goes to infinity of simpler parametric models.
- Bayesian inference makes it possible to reason with nonparametric models without overfitting.
- The effective complexity of the nonparametric model grows with more data.
- Nonparametric Bayesian models are often faster and conceptually easier to implement since one doesn't have to compare multiple nested models.

Sparse Matrices

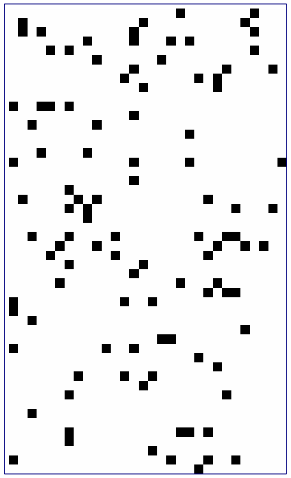
A binary matrix representation for clustering





- Rows are data points
- Columns are clusters
- Since each data point is assigned to one and only one cluster...
- ...the rows sum to one.

More general latent binary matrices



- Rows are data points
- Columns are latent features
- We can think of **infinite** binary matrices...
  ...where each data point can now have *multiple* features, so...
  ...the rows can sum to more than one.

Another way of thinking about this:

- there are multiple overlapping clusters
- each data point can belong to several clusters simultaneously.

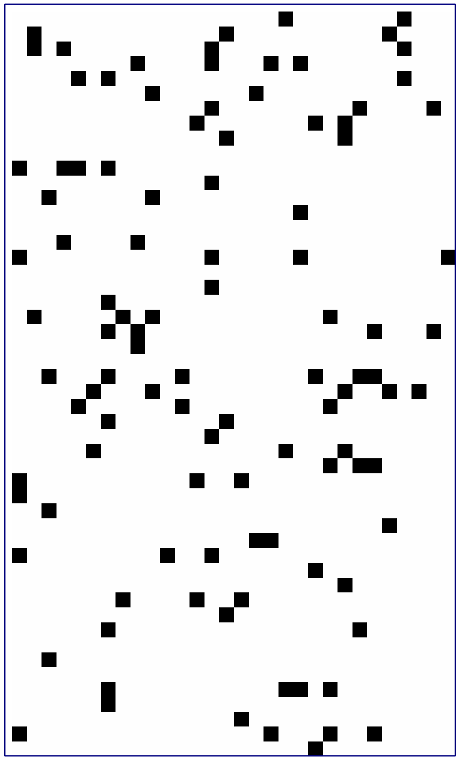
Why?

- Clustering models are restrictive; they do not have distributed or factorial representations.

- Consider modelling people's movie preferences (the "Netflix" problem). A movie might be described using features such as "is science fiction", "has Charlton Heston", "was made in the US", "was made in 1970s", "has apes in it"... Similarly a person may be described as "male", "teenager", "British", "urban". These features may be unobserved (latent).

- The number of potential latent features for describing a movie (or person, news story, image, gene, speech waveform, etc) is unlimited.

From finite to infinite binary matrices



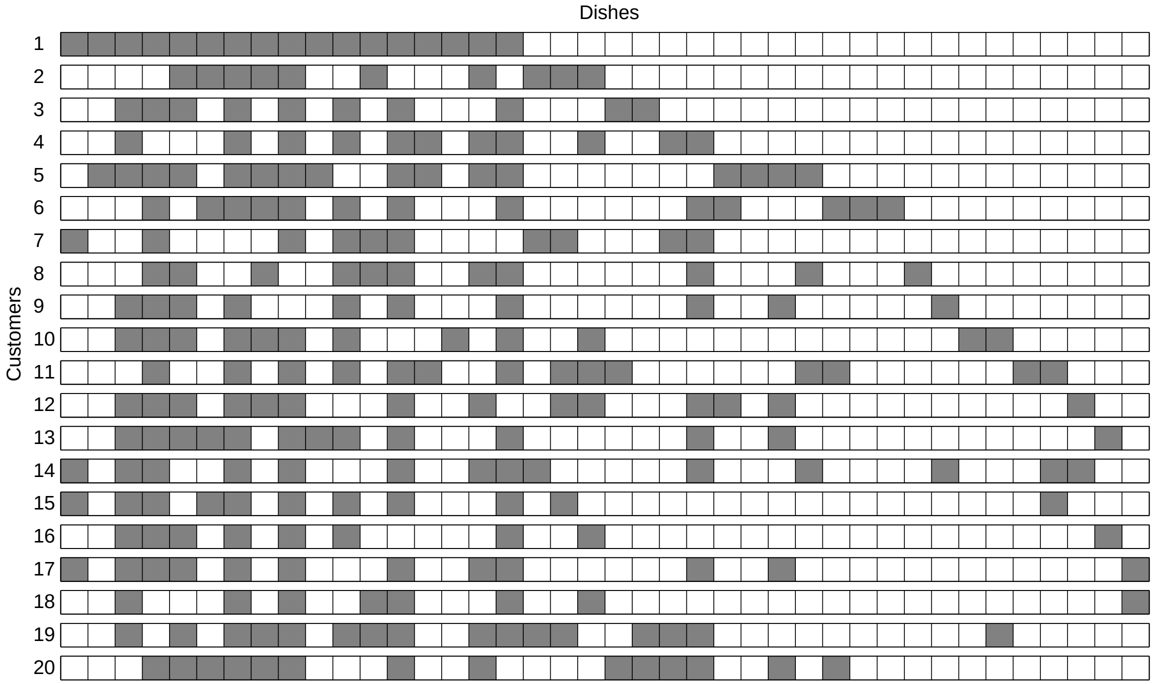
$z_{nk} = 1$ means object n has feature k:

$$z_{nk} \sim \text{Bernoulli}(\theta_k)$$

$$\theta_k \sim \text{Beta}(\alpha/K, 1)$$

- Note that $P(z_{nk} = 1|\alpha) = E(\theta_k) = \frac{\alpha/K}{\alpha/K+1}$, so as K grows larger the matrix gets sparser.
- So if $\mathbf{Z}$ is $N \times K$, the expected number of nonzero entries is $N\alpha/(1 + \alpha/K) < N\alpha$.
- Even in the $K \to \infty$ limit, the matrix is expected to have a finite number of non-zero entries.

Indian buffet process



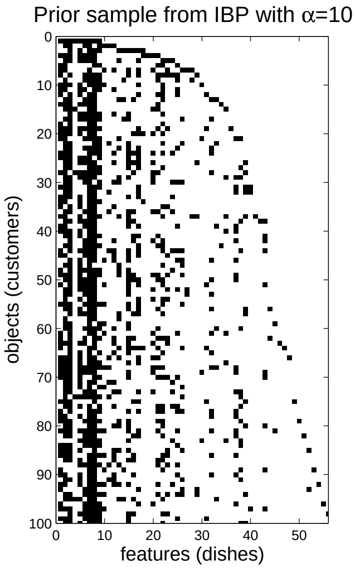
“Many Indian restaurants in London offer lunchtime buffets with an apparently infinite number of dishes”

- First customer starts at the left of the buffet, and takes a serving from each dish, stopping after a Poisson(α) number of dishes as her plate becomes overburdened.
- The nth customer moves along the buffet, sampling dishes in proportion to their popularity, serving himself with probability m_k/n, and trying a Poisson(α/n) number of new dishes.
- The customer-dish matrix is our feature matrix, $\mathbf{Z}$.

(Griffiths and Ghahramani, 2006; 2011)

Properties of the Indian buffet process

$$P([\mathbf{Z}]|\alpha) = \exp\left\{ -\alpha H_N \right\} \frac{\alpha^{K_+}}{\prod_{h>0} K_h!} \prod_{k \leq K_+} \frac{(N-m_k)!(m_k-1)!}{N!}$$



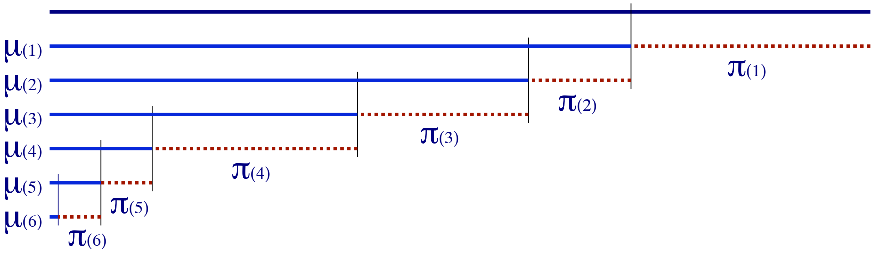


Figure 1: Stick-breaking construction for the DP and IBP. The black stick at top has length 1. At each iteration the vertical black line represents the break point. The brown dotted stick on the right is the weight obtained for the DP, while the blue stick on the left is the weight obtained for the IBP.

Shown in (Griffiths and Ghahramani, 2006):

- It is infinitely exchangeable.
- The number of ones in each row is Poisson(α)
- The expected total number of ones is αN.
- The number of nonzero columns grows as $O(\alpha \log N)$.

Additional properties:

- Has a stick-breaking representation (Teh, Görür, Ghahramani, 2007)
- Has as its de Finetti mixing distribution the Beta process (Thibaux and Jordan, 2007)

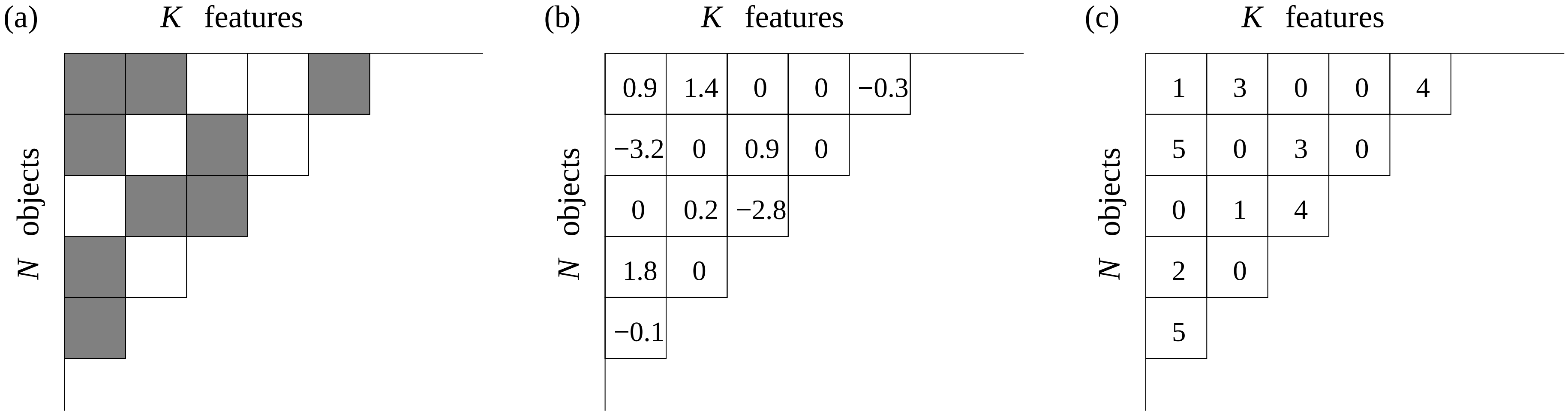
From binary to non-binary latent features

In many models we might want non-binary latent features.

A simple way to generate non-binary latent feature matrices from $\mathbf{Z}$:

$$\mathbf{F} = \mathbf{Z} \otimes \mathbf{V}$$

where $\otimes$ is the elementwise (Hadamard) product of two matrices, and $\mathbf{V}$ is a matrix of independent random variables (e.g. Gaussian, Poisson, Discrete, ...).



0.9	1.4	0	0	−0.3
−3.2	0	0.9	0	
0	0.2	−2.8		
1.8	0			
−0.1				

1	3	0	0	4
5	0	3	0	
0	1	4		
2	0			
5				

A two-parameter generalization of the IBP

$z_{nk} = 1$ means object n has feature k

One-parameter IBP

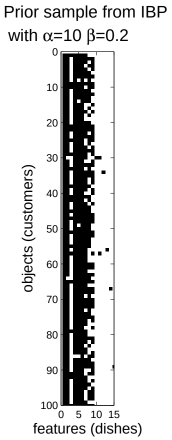
$$z_{nk} \sim \mathrm{Bernoulli}(\theta_k)$$
$$\theta_k \sim \mathrm{Beta}(\alpha/K, 1)$$

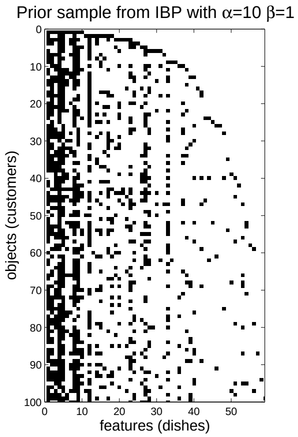
Two-parameter IBP

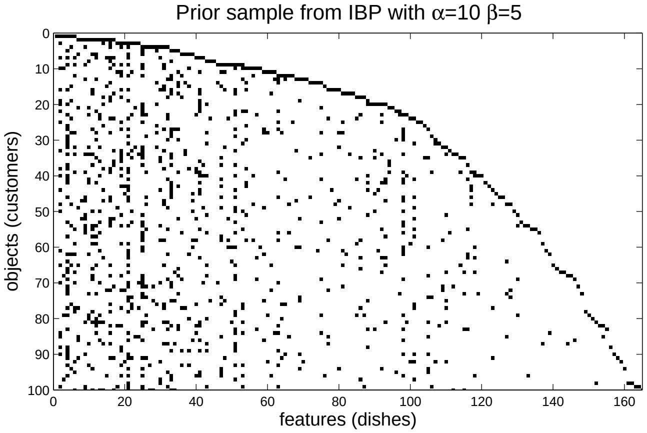
$$z_{nk} \sim \mathrm{Bernoulli}(\theta_k)$$
$$\theta_k \sim \mathrm{Beta}(\alpha\beta/K, \beta)$$

Properties of the two-parameter IBP

- Number of features per object is $\mathrm{Poisson}(\alpha)$. Setting $\beta = 1$ reduces to IBP. Parameter β is feature repulsion, $1/\beta$ is feature stickiness.
- Total expected number of features is $\bar{K}_+ = \alpha \sum_{n=1}^{N} \frac{\beta}{\beta + n - 1} \longrightarrow \alpha\beta \log N$
- $\lim_{\beta \to 0} \bar{K}_+ = \alpha$ and $\lim_{\beta \to \infty} \bar{K}_+ = N\alpha$







Posterior Inference in IBPs

$$P(\mathbf{Z}, \alpha|\mathbf{X}) \propto P(\mathbf{X}|\mathbf{Z})P(\mathbf{Z}|\alpha)P(\alpha)$$

Gibbs sampling: $P(z_{nk} = 1|\mathbf{Z}_{-(nk)}, \mathbf{X}, \alpha) \propto P(z_{nk} = 1|\mathbf{Z}_{-(nk)}, \alpha)P(\mathbf{X}|\mathbf{Z})$

- If $m_{-n,k} > 0$, $P(z_{nk} = 1|\mathbf{z}_{-n,k}) = \frac{m_{-n,k}}{N}$
- For infinitely many k such that $m_{-n,k} = 0$: Metropolis steps with truncation* to sample from the number of new features for each object.
- If α has a Gamma prior then the posterior is also Gamma → Gibbs sample.

Conjugate sampler: assumes that $P(\mathbf{X}|\mathbf{Z})$ can be computed.
Non-conjugate sampler: $P(\mathbf{X}|\mathbf{Z}) = \int P(\mathbf{X}|\mathbf{Z}, \theta)P(\theta)d\theta$ cannot be computed, requires sampling latent θ as well (e.g. approximate samplers based on (Neal 2000) non-conjugate DPM samplers).
***Slice sampler:** works for non-conjugate case, is not approximate, and has an adaptive truncation level using an IBP stick-breaking construction (Teh, et al 2007) see also (Adams et al 2010).
Deterministic Inference: variational inference (Doshi et al 2009a) parallel inference (Doshi et al 2009b), beam-search MAP (Rai and Daume 2011), power-EP (Ding et al 2010)

Modelling Data with Indian Buffet Processes

Latent variable model: let $\mathbf{X}$ be the $N \times D$ matrix of observed data, and $\mathbf{Z}$ be the $N \times K$ matrix of binary latent features

$$P(\mathbf{X}, \mathbf{Z}|\alpha) = P(\mathbf{X}|\mathbf{Z})P(\mathbf{Z}|\alpha)$$

By combining the IBP with different likelihood functions we can get different kinds of models:

- Models for graph structures (w/ Wood, Griffiths, 2006; w/ Adams and Wallach, 2010)
- Models for protein complexes (w/ Chu, Wild, 2006)
- Models for choice behaviour (Görür & Rasmussen, 2006)
- Models for users in collaborative filtering (w/ Meeds, Roweis, Neal, 2007)
- Sparse latent trait, pPCA and ICA models (w/ Knowles, 2007, 2011)
- Models for overlapping clusters (w/ Heller, 2007)

Nonparametric Binary Matrix Factorization

genes × patients
users × movies

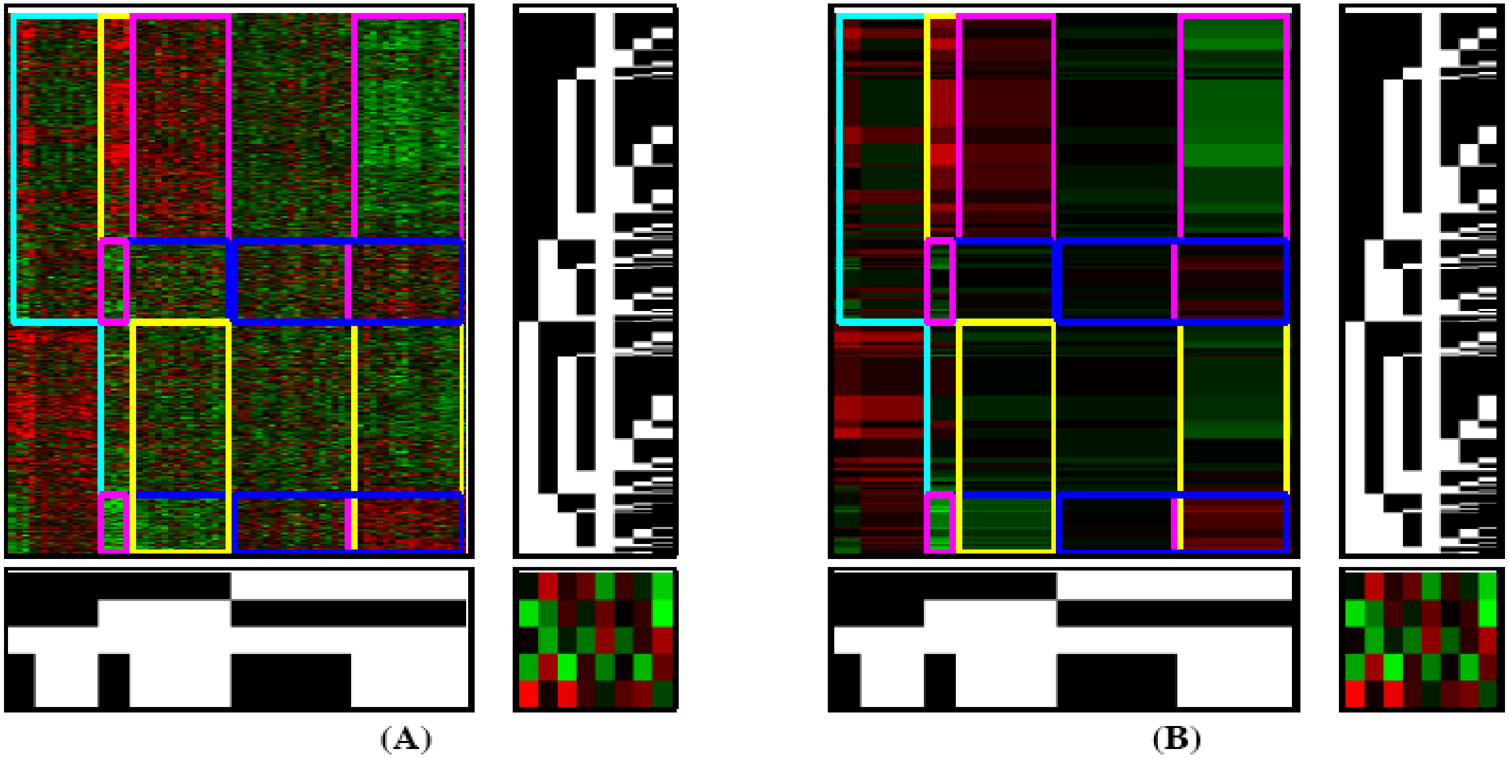
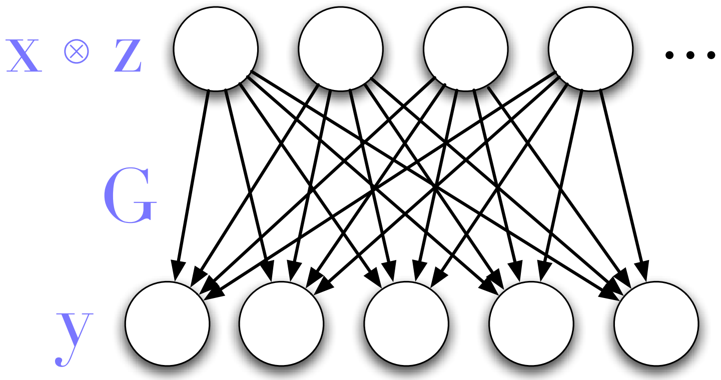


Figure 5: Gene expression results. **(A)** The top-left is $\mathbf{X}$ sorted according to contiguous features in the final $\mathbf{U}$ and $\mathbf{V}$ in the Markov chain. The bottom-left is $\mathbf{V}^\top$ and the top-right is $\mathbf{U}$. The bottom-right is $\mathbf{W}$. **(B)** The same as **(A)**, but the expected value of $\mathbf{X}$, $\hat{\mathbf{X}} = \mathbf{UWV}^\top$. We have hilighted regions that have both u_{ik} and v_{jl} on. For clarity, we have only shown the (at most) two largest contiguous regions for each feature pair.

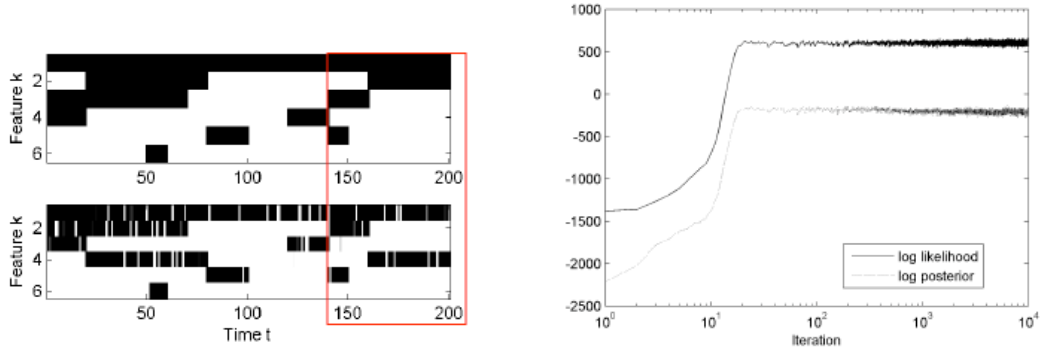
Meeds et al (2007) Modeling Dyadic Data with Binary Latent Factors.

Nonparametric Sparse Latent Factor Models and Infinite Independent Components Analysis



Model: $\mathbf{Y} = \mathbf{G}(\mathbf{Z} \otimes \mathbf{X}) + \mathbf{E}$

where $\mathbf{Y}$ is the data matrix, $\mathbf{G}$ is the mixing matrix $\mathbf{Z} \sim \mathrm{IBP}(\alpha, \beta)$ is a mask matrix, $\mathbf{X}$ is heavy tailed sources and $\mathbf{E}$ is Gaussian noise.



(a) *Top:* True **Z**. *Bottom:* Inferred **Z**. Red box denotes test data.

(b) Plot of the log likelihood and posterior for the duration of the iICA$_2$ run.

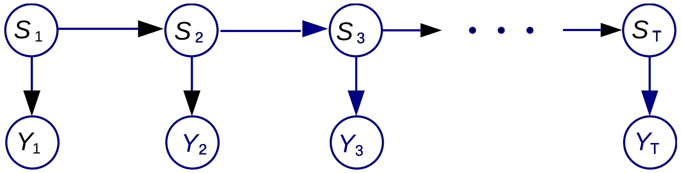
Fig. 1. True and inferred **Z** and algorithm convergence.

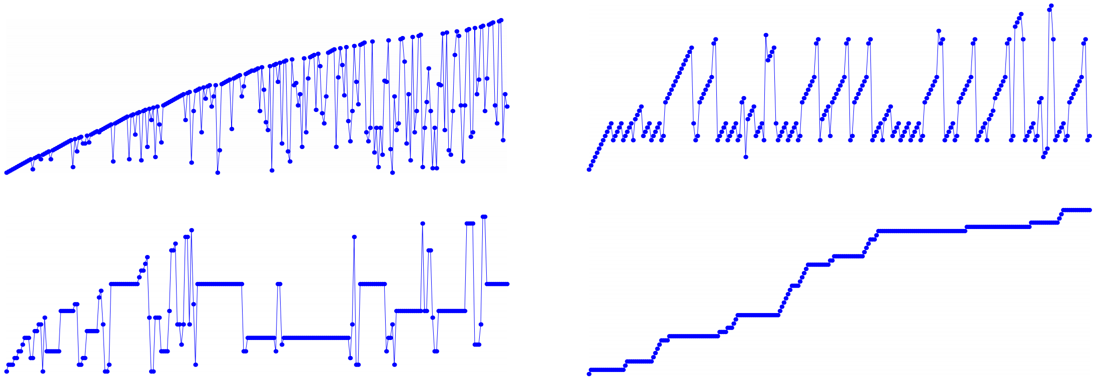
(w/ David Knowles, 2007, 2011)

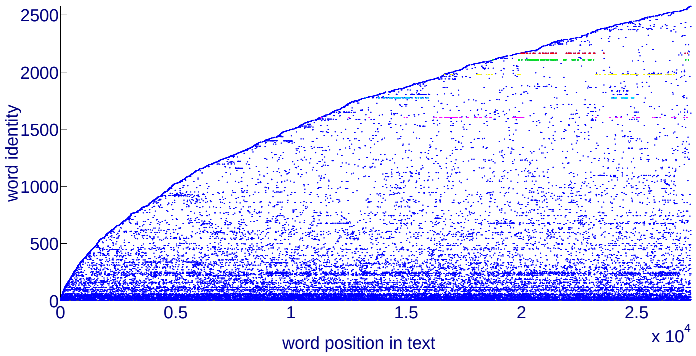
Time Series

Infinite hidden Markov models (iHMMs)

In an HMM with K states, the transition matrix has $K \times K$ elements. Let $K \to \infty$.

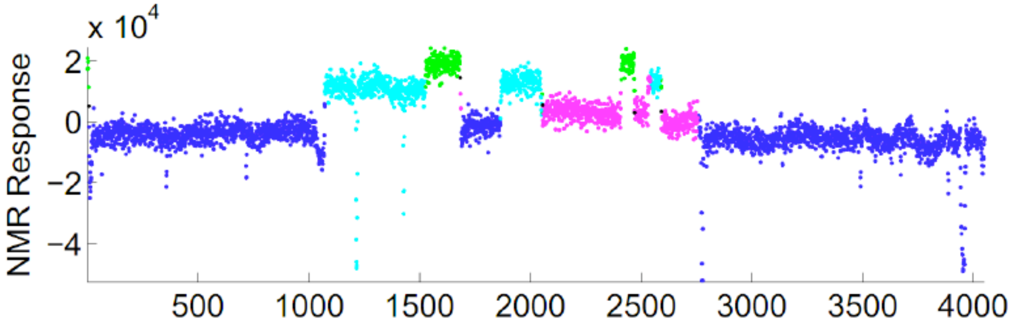


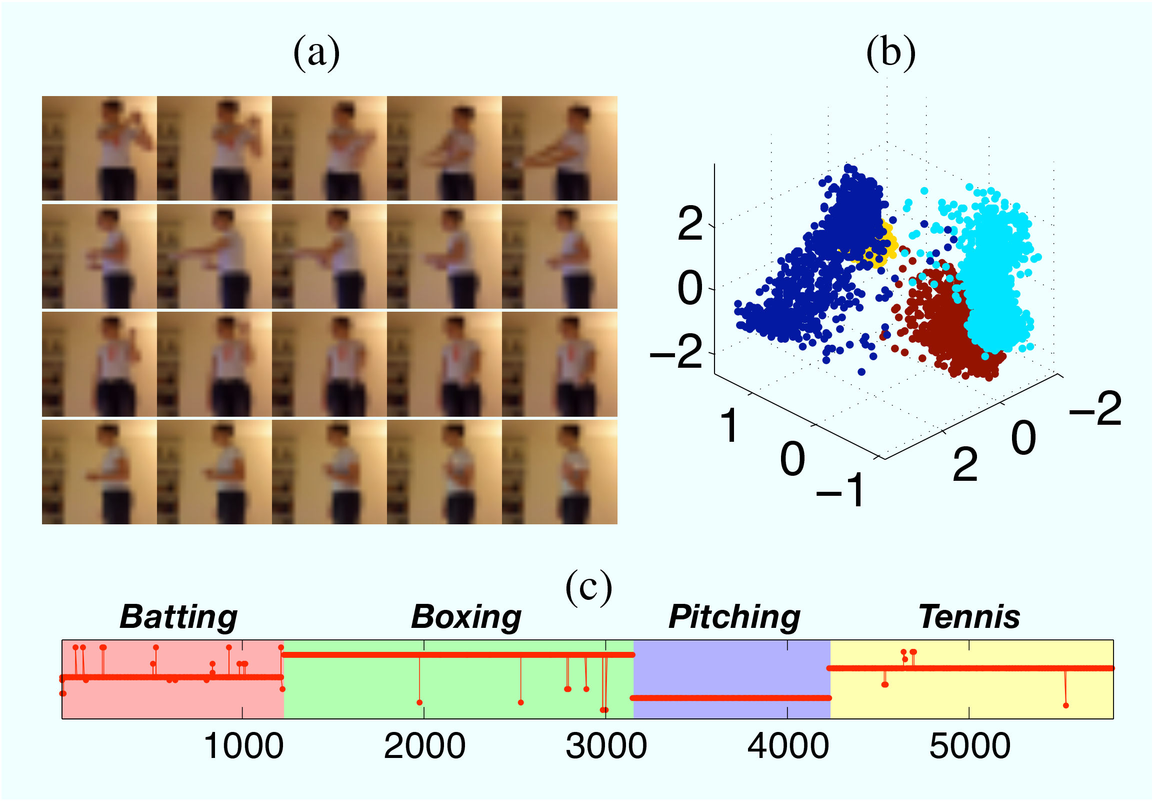




- iHMMs introduced in (Beal, Ghahramani and Rasmussen, 2002).
- Teh, Jordan, Beal and Blei (2005) showed that iHMMs can be derived from hierarchical Dirichlet processes, and provided a more efficient Gibbs sampler (note: HDP-HMM $\equiv$ iHMM).
- We have recently derived a much more efficient sampler based on Dynamic Programming (Van Gael, Saatci, Teh, and Ghahramani, 2008). `http://mloss.org/software/view/205/`
- And we have parallel (.NET) and distributed (Hadoop) implementations (Bratieres, Van Gael, Vlachos and Ghahramani, 2010).

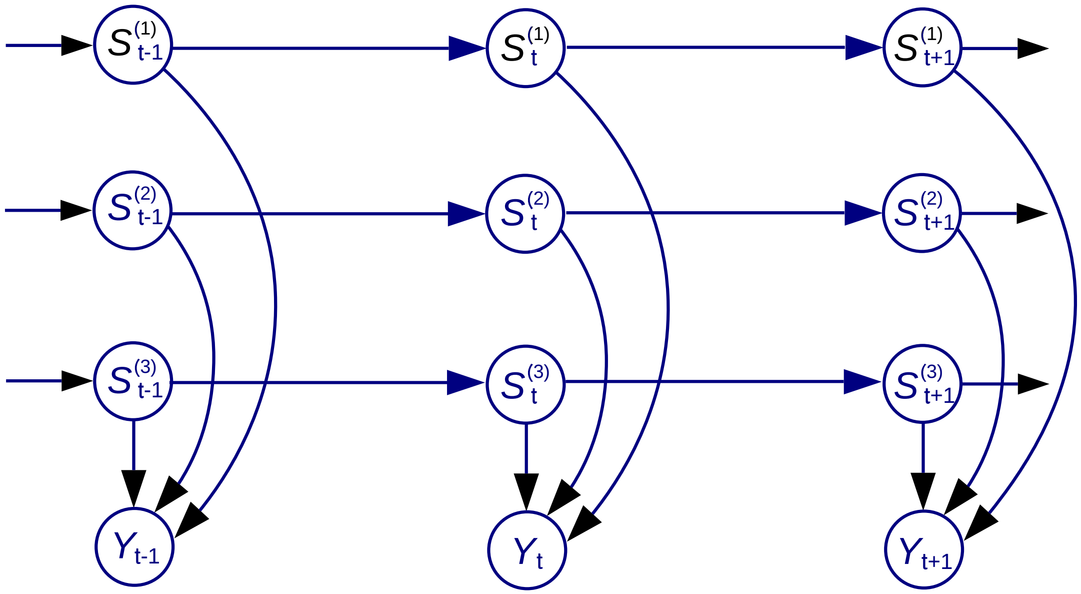
Infinite HMM: Changepoint detection and video segmentation





(Stepleton, et al 2009)

Markov Indian Buffet Process and Infinite Factorial Hidden Markov Models

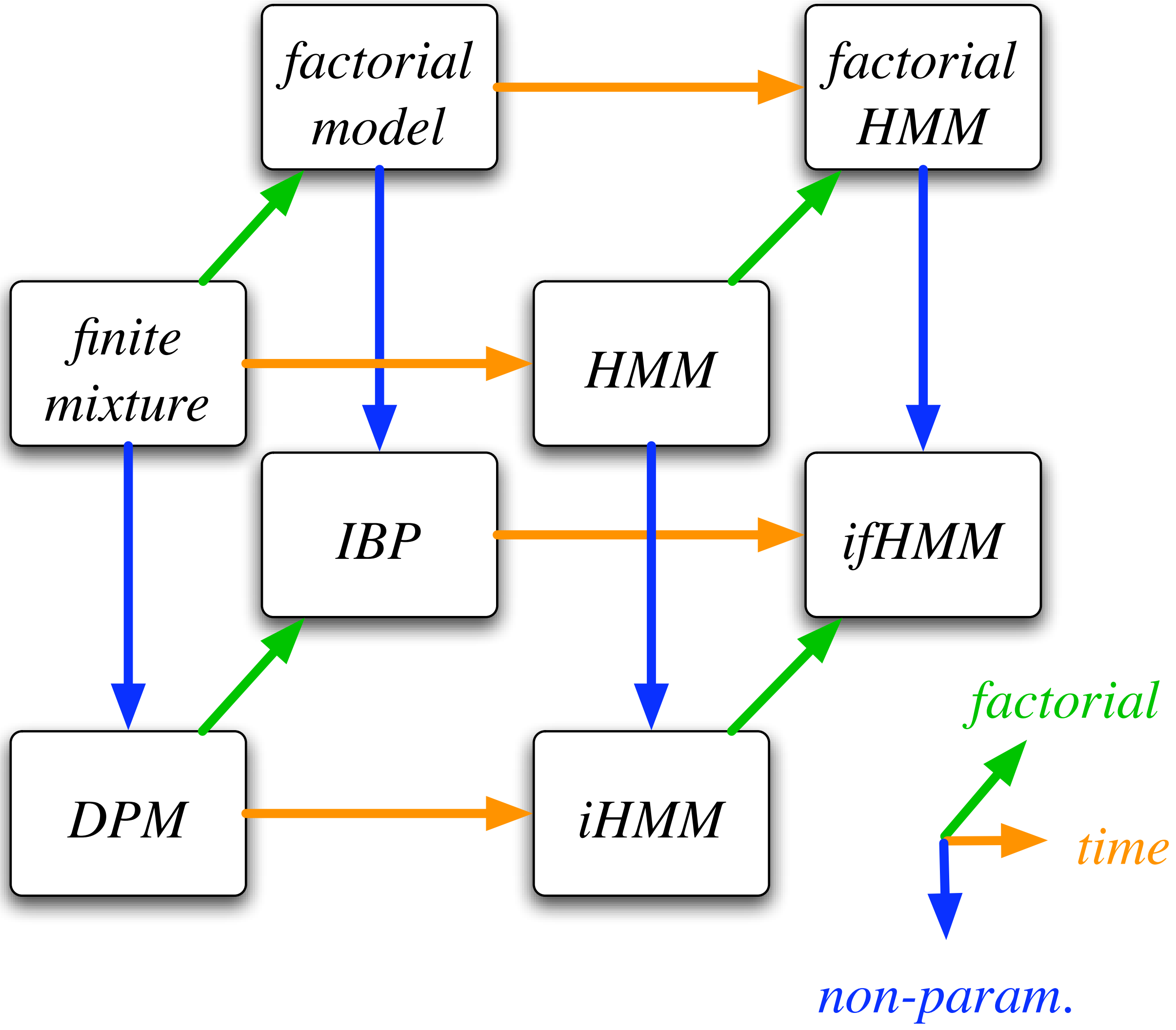


- Hidden Markov models (HMMs) represent the history of a time series using a **single** discrete state variable

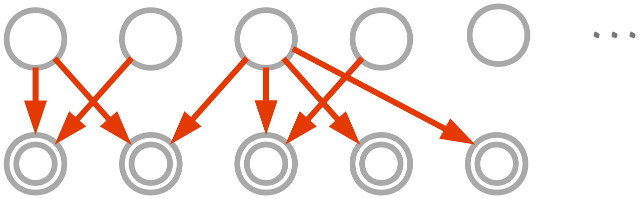
- Factorial HMMs (fHMM) are a kind of HMM with a factored state representation (w/ Jordan, 1997)

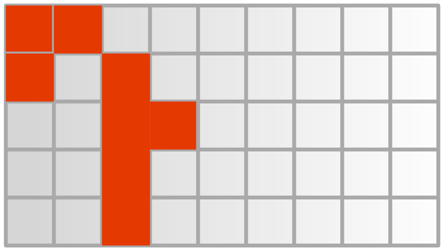
- We can extend the Indian Buffet Process to time series and use it to define a non-parametric version of the fHMM (w/ van Gael, Teh, 2008)

A Picture:
Relations between some models

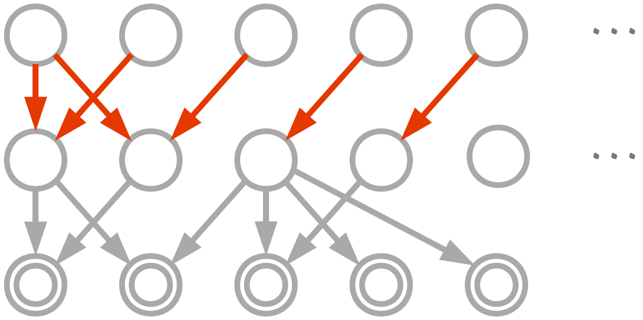


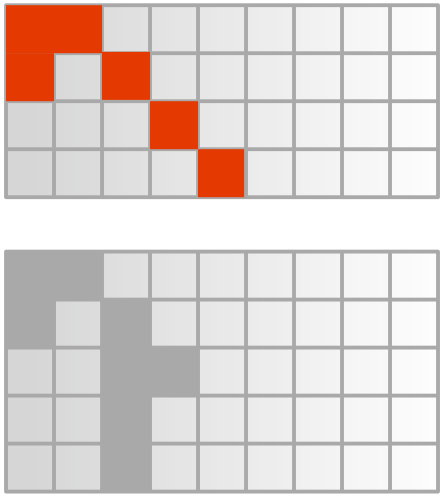
Learning Structure of Deep Sparse Graphical Models



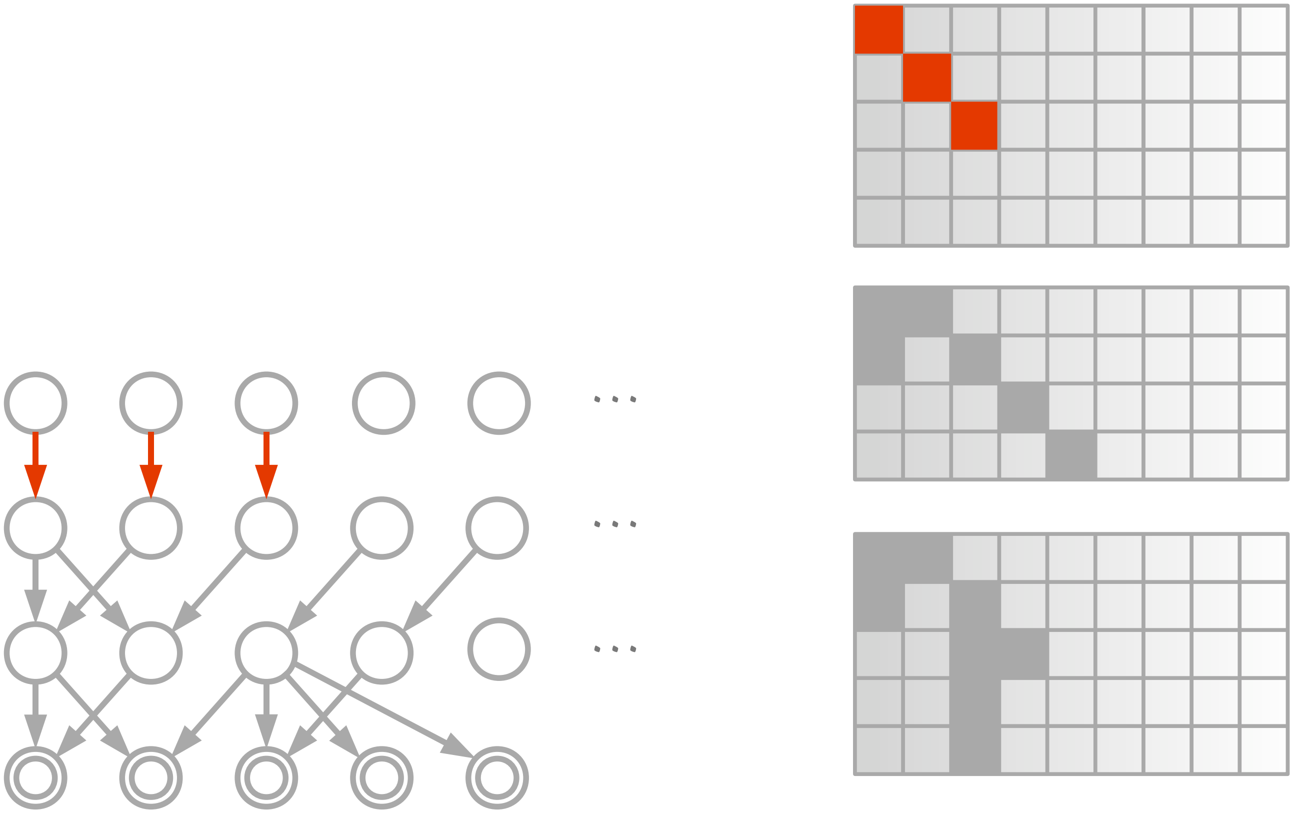


Learning Structure of Deep Sparse Graphical Models

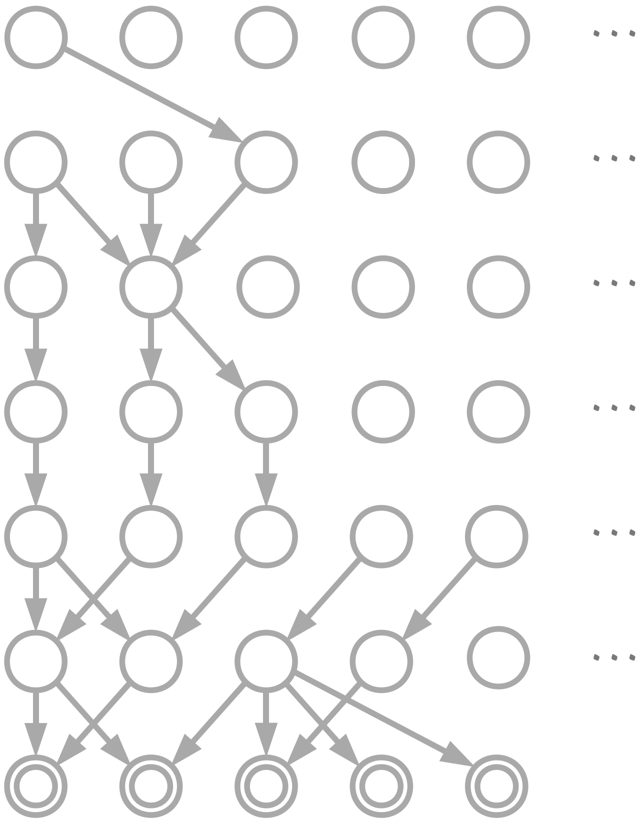


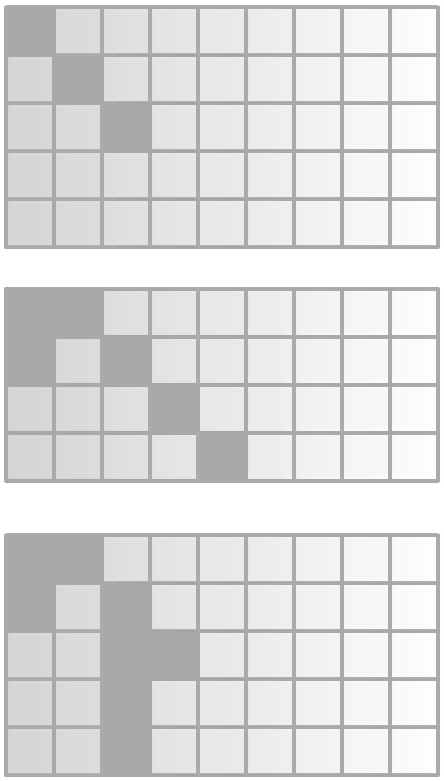


Learning Structure of Deep Sparse Graphical Models



Learning Structure of Deep Sparse Graphical Models



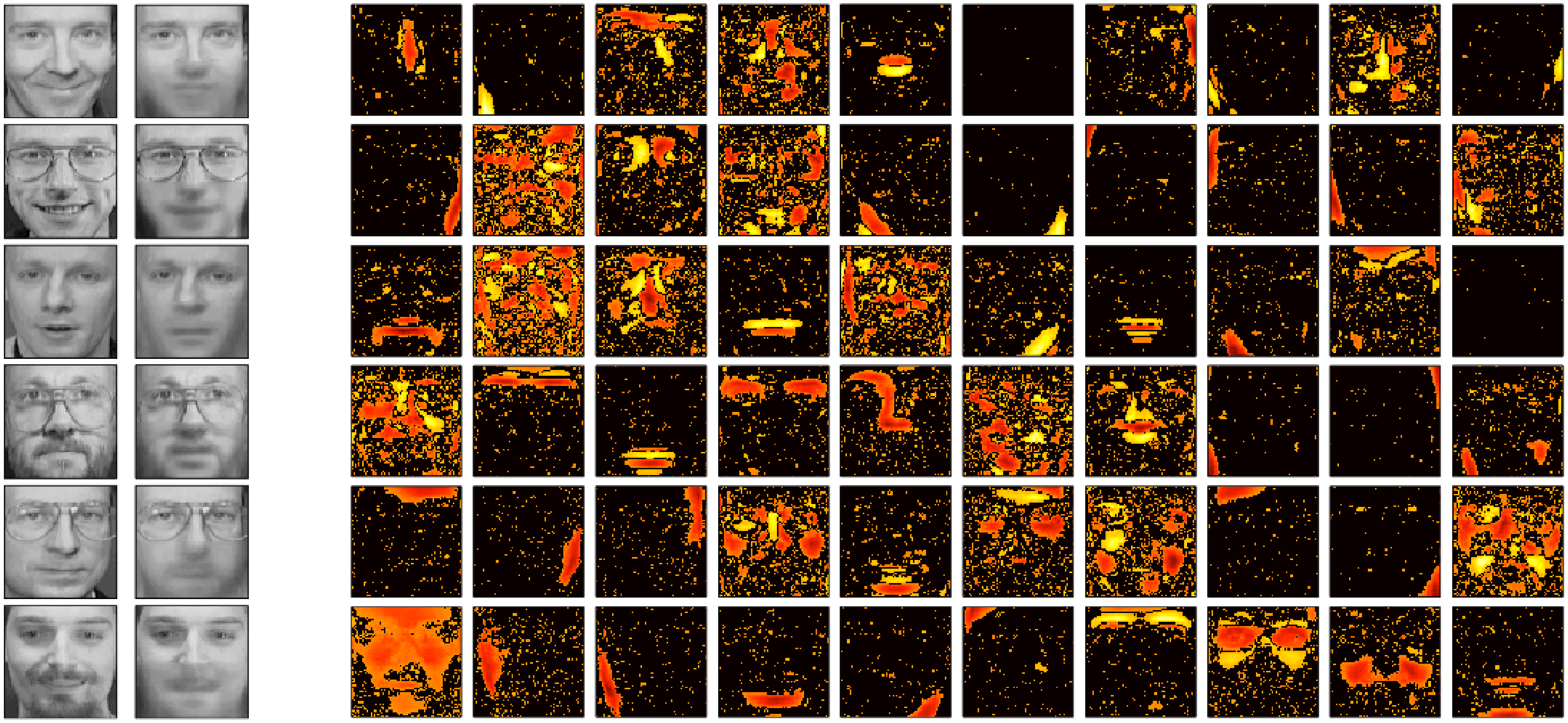


(w/ Ryan P. Adams, Hanna Wallach, 2010)

Learning Structure of Deep Sparse Graphical Models

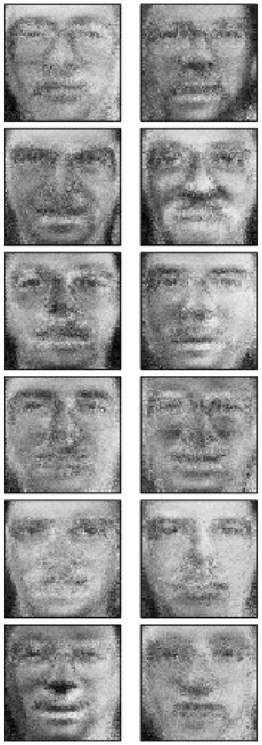
Olivetti Faces: 350 + 50 images of 40 faces (64×64)
Inferred: 3 hidden layers, 70 units per layer.

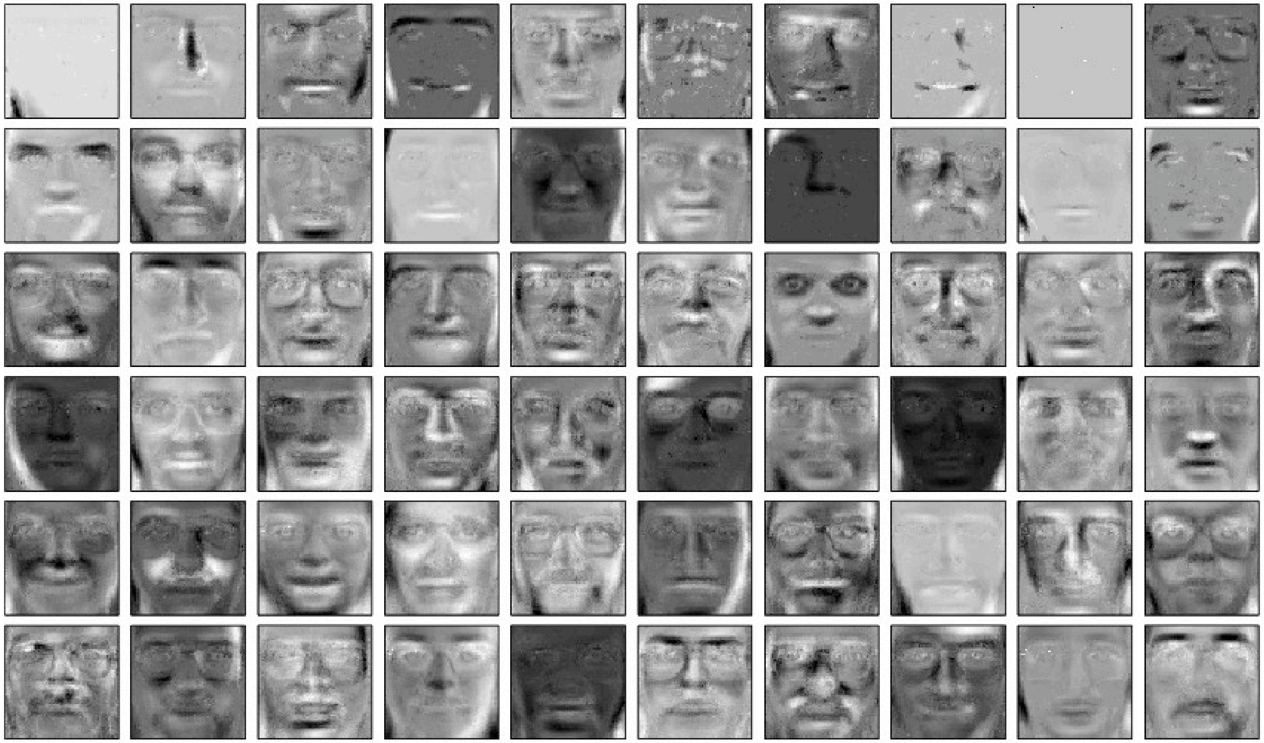
Reconstructions and Features:



Learning Structure of Deep Sparse Graphical Models

Fantasies and Activations:

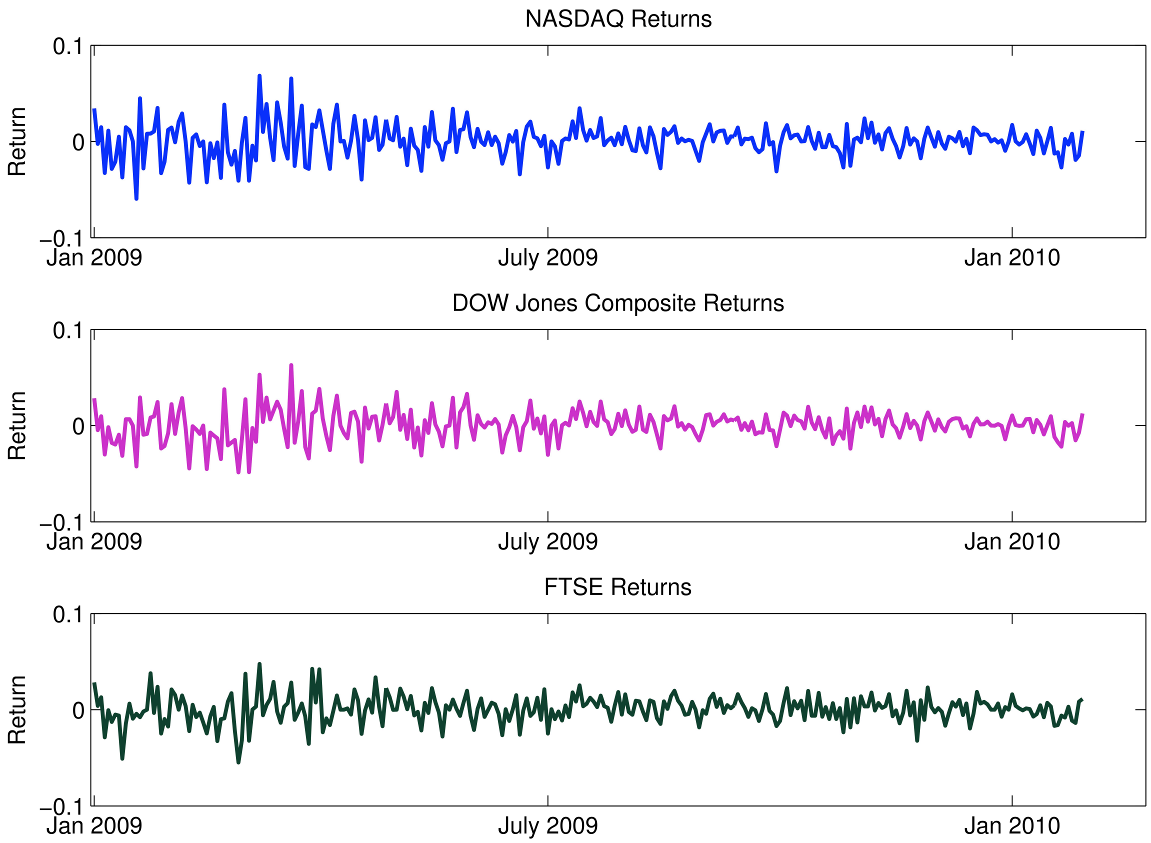




Covariances

Covariances

Consider the problem of modelling a covariance matrix Σ that can change as a function of time, $\Sigma(t)$, or other input variables $\Sigma(x)$. This is a widely studied problem in *Econometrics*.



Models commonly used are multivariate GARCH, and multivariate stochastic volatility models, but these only depend on t, and generally don't scale well.

Generalised Wishart Processes for Covariance modelling

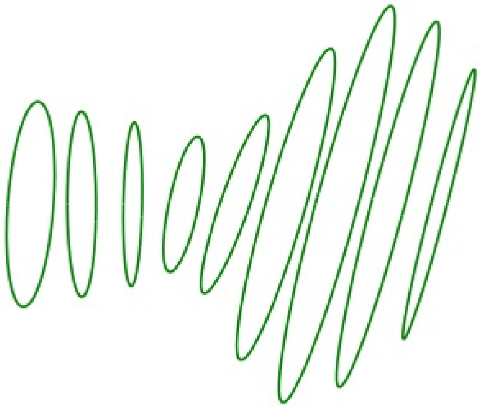
Modelling time- and spatially-varying covariance matrices. Note that covariance matrices have to be symmetric positive (semi-)definite.

If $\mathbf{u}_i \sim \mathcal{N}$, then $\Sigma = \sum_{i=1}^{\nu} \mathbf{u}_i \mathbf{u}_i^\top$ is s.p.d. and has a Wishart distribution.

We are going to generalise Wishart distributions to be dependent on time or other inputs, making a nonparametric Bayesian model based on Gaussian Processes (GPs).

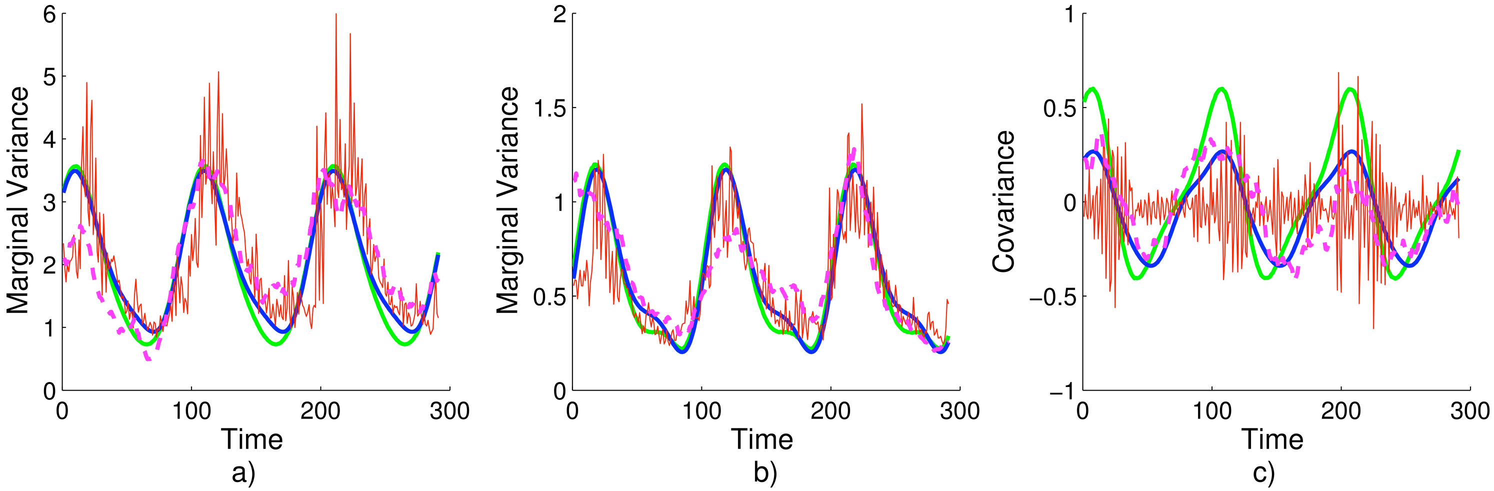
So if $\mathbf{u}_i(t) \sim \mathrm{GP}$, then $\Sigma(t) = \sum_{i=1}^{\nu} \mathbf{u}_i(t) \mathbf{u}_i(t)^\top$ defines a **Wishart process**.

This is the simplest form, many generalisations are possible.



(w/ Andrew Wilson, 2010)

Generalised Wishart Process Results



Legend: Truth, GWP, WP, MGARCH

The GWP significantly outperforms its competitors (in MSE and likelihood) on simulated and financial data, even in lower dimensions (<5) and on data that is especially suited to GARCH.

On 5D equity index data, using a GWP with a squared exponential covariance function, forecast log likelihoods are:
GWP: 2930, WP: 1710, BEKK MGARCH: 2760.

Generalised Wishart Process Results

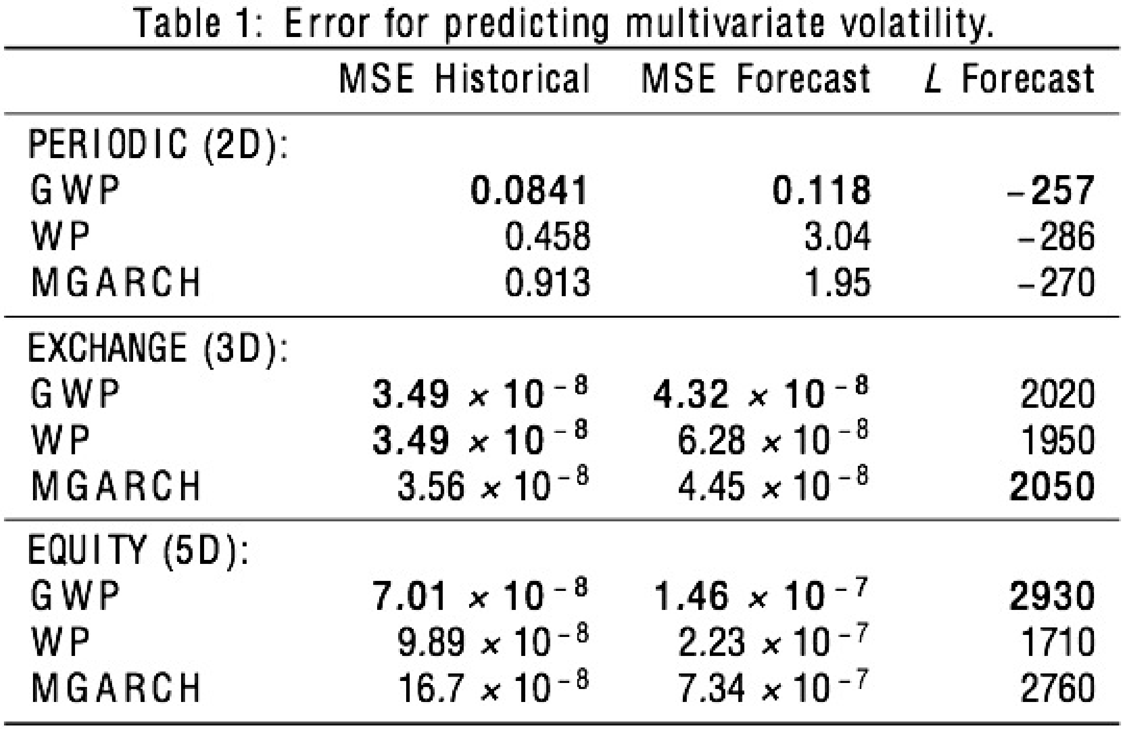


Table 1: Error for predicting multivariate volatility.

	MSE Historical	MSE Forecast	L Forecast
PERIODIC (2D):			
GWP	**0.0841**	**0.118**	**−257**
WP	0.458	3.04	−286
MGARCH	0.913	1.95	−270
EXCHANGE (3D):			
GWP	**3.49×10^{-8}**	**4.32×10^{-8}**	2020
WP	**3.49×10^{-8}**	6.28×10^{-8}	1950
MGARCH	3.56×10^{-8}	4.45×10^{-8}	**2050**
EQUITY (5D):			
GWP	**7.01×10^{-8}**	**1.46×10^{-7}**	**2930**
WP	9.89×10^{-8}	2.23×10^{-7}	1710
MGARCH	16.7×10^{-8}	7.34×10^{-7}	2760

- GWP can learn the GP kernel from data and accommodate dependence on time and other covariates.

- Scales well to high-dimensional data using MCMC inference based on elliptical slice sampling.

- Related work: Bru (1991), Gelfand et al (2004), Philipov and Glickman (2006), Gourieroux et al (2009).

Summary

- Probabilistic modelling and Bayesian inference are two sides of the same coin
- Bayesian machine learning treats learning as a probabilistic inference problem
- Bayesian methods work well when the models are flexible enough to capture relevant properties of the data
- This motivates non-parametric Bayesian methods, e.g.:
  - Indian buffet processes for sparse matrices and latent feature modelling
  - Infinite HMMs for time series modelling
  - Wishart processes for covariance modelling

`http://learning.eng.cam.ac.uk/zoubin`
`zoubin@eng.cam.ac.uk`

Some References

- Adams, R.P., Wallach, H., Ghahramani, Z. (2010) Learning the Structure of Deep Sparse Graphical Models. AISTATS 2010.
- Beal, M. J., Ghahramani, Z. and Rasmussen, C. E. (2002) The infinite hidden Markov model. NIPS **14**:577–585.
- Bratieres, S., van Gael, J., Vlachos, A., and Ghahramani, Z. (2010) Scaling the iHMM: Parallelization versus Hadoop. International Workshop on Scalable Machine Learning and Applications (SMLA-10), 1235–1240.
- Griffiths, T.L., and Ghahramani, Z. (2006) Infinite Latent Feature Models and the Indian Buffet Process. NIPS **18**:475–482.
- Griffiths, T. L., and Ghahramani, Z. (2011) The Indian buffet process: An introduction and review. *Journal of Machine Learning Research* **12**(Apr):1185–1224.
- Meeds, E., Ghahramani, Z., Neal, R. and Roweis, S.T. (2007) Modeling Dyadic Data with Binary Latent Factors. NIPS **19**:978–983.
- Stepleton, T., Ghahramani, Z., Gordon, G., Lee, T.-S. (2009) The Block Diagonal Infinite Hidden Markov Model. AISTATS 2009, 552–559.
- Wilson, A.G., and Ghahramani, Z. (2010, 2011) Generalised Wishart Processes. arXiv:1101.0240v1. and UAI 2011
- van Gael, J., Saatci, Y., Teh, Y.-W., and Ghahramani, Z. (2008) Beam sampling for the infinite Hidden Markov Model. ICML 2008, 1088-1095.
- van Gael, J and Ghahramani, Z. (2010) Nonparametric Hidden Markov Models. In Barber, D., Cemgil, A.T. and Chiappa, S. *Inference and Learning in Dynamic Models*. CUP.