an introduction to bayesian inference

with an application to network analysis

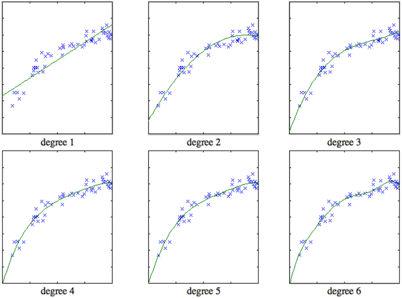
jake hofman

http://jakehofman.com

january 13, 2010

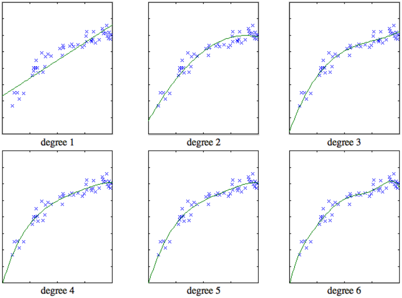
motivation

- would like models that:
    - provide predictive and explanatory power
    - are complex enough to describe observed phenomena
    - are simple enough to generalize to future observations



motivation

- would like models that:
  - provide predictive and explanatory power
  - are complex enough to describe observed phenomena
  - are simple enough to generalize to future observations



- claim: bayesian inference provides a systematic framework to infer such models from observed data

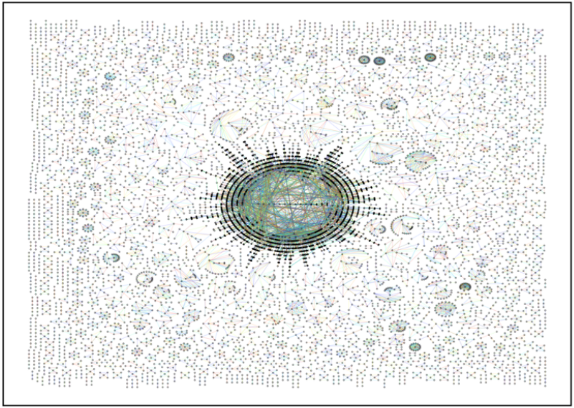
motivation

- principles behind bayesian interpretation of probability and bayesian inference are well established (bayes, laplace, etc., 18th century)

+

- recent advances in mathematical techniques and computational resources have enabled successful applications of these principles to real-world problems

motivation: a bayesian approach to network modularity



outline

1. principles (what we'd like to do)
   - background: joint, marginal, and conditional probabilities
   - bayes' theorem: inverting conditional probabilities
   - bayesian probability: unknowns as random variables
   - bayesian inference: bayesian probability + bayes' theorem
2. practice (what we're able to do)
   - monte carlo methods: representative samples
   - variational methods: bound optimization
   - references
3. application: bayesian inference for network data

joint, marginal, and conditional probabilities

joint distribution

$p_{XY}(X=x, Y=y)$: probability $X=x$ *and* $Y=y$

conditional distribution

$p_{X|Y}(X=x|Y=y)$: probability $X=x$ *given* $Y=y$

marginal distribution

$p_X(X)$: probability $X=x$ (*regardless of* Y)

sum and product rules

sum rule

sum out settings of irrelevant variables:

$$p(x) = \sum_{y \in \Omega_Y} p(x, y) \tag{1}$$

product rule

the joint as the product of the conditional and marginal:

$$\begin{aligned} p(x, y) &= p(x|y)\, p(y) \qquad (2) \\ &= p(y|x)\, p(x) \qquad (3) \end{aligned}$$

outline

1. principles (what we'd like to do)
   - background: joint, marginal, and conditional probabilities
   - bayes' theorem: inverting conditional probabilities
   - bayesian probability: unknowns as random variables
   - bayesian inference: bayesian probability + bayes' theorem
2. practice (what we're able to do)
   - monte carlo methods: representative samples
   - variational methods: bound optimization
   - references
3. application: bayesian inference for network data

inverting conditional probabilities

equate far right- and left-hand sides of product rule

$$p(y|x)\,p(x) = p(x,y) = p(x|y)\,p(y) \tag{4}$$

and divide:

bayes' theorem (bayes and price 1763)

the probability of Y given X from the probability of X given Y:

$$p(y|x) = \frac{p(x|y)\,p(y)}{p(x)} \tag{5}$$

where $p(x) = \sum_{y \in \Omega_Y} p(x|y)\,p(y)$ is the normalization constant

example: diagnoses a la bayes

- population 10,000
- 1% has (rare) disease[1]
- test is 99% (relatively) effective, i.e.
  - given a patient is sick, 99% test positive
  - given a patient is healthy, 99% test negative

[1] subtlety: assuming this fraction is *known*

example: diagnoses a la bayes

- population 10,000
- 1% has (rare) disease
- test is 99% (relatively) effective, i.e.
  - given a patient is sick, 99% test positive
  - given a patient is healthy, 99% test negative
- given positive test, what is probability the patient is sick?[1]

[1] follows wiggins (2006)

example: diagnoses a la bayes

- 99 sick patients test positive, 99 healthy patients test positive

example: diagnoses a la bayes

- 99 sick patients test positive, 99 healthy patients test positive
- given positive test, 50% probability that patient is sick

example: diagnoses a la bayes

- know probability of testing positive/negative given sick/healthy
- use bayes' theorem to "invert" to probability of sick/healthy given positive/negative test

$$p\left(sick|test\ +\right) = \frac{\overbrace{p\left(test\ +|sick\right)}^{99/100}\overbrace{p\left(sick\right)}^{1/100}}{\underbrace{p\left(test\ +\right)}_{99/100^2+99/100^2=198/100^2}} = \frac{99}{198} = \frac{1}{2} \quad (6)$$

example: diagnoses a la bayes

- know probability of testing positive/negative given sick/healthy
- use bayes' theorem to "invert" to probability of sick/healthy given positive/negative test

$$p\left(sick|test\ +\right) = \frac{\overbrace{p\left(test\ +|sick\right)}^{99/100}\overbrace{p\left(sick\right)}^{1/100}}{\underbrace{p\left(test\ +\right)}_{99/100^2+99/100^2=198/100^2}} = \frac{99}{198} = \frac{1}{2} \tag{6}$$

- most "work" in calculating denominator (normalization)

outline

1. principles (what we'd like to do)
   - background: joint, marginal, and conditional probabilities
   - bayes' theorem: inverting conditional probabilities
   - bayesian probability: unknowns as random variables
   - bayesian inference: bayesian probability + bayes' theorem
2. practice (what we're able to do)
   - monte carlo methods: representative samples
   - variational methods: bound optimization
   - references
3. application: bayesian inference for network data

interpretations of probabilities

(just enough philosophy)

- frequentists: limit of relative frequency of events for large number of trials
- bayesians: measure of a state of knowledge, quantifying degrees of belief (jaynes 2003)

interpretations of probabilities
(just enough philosophy)

- frequentists: limit of relative frequency of events for large number of trials
- bayesians: measure of a state of knowledge, quantifying degrees of belief (jaynes 2003)
- key difference: bayesians permit assignment of probabilities to unknown/unobservable hypotheses (frequentists do not)

interpretations of probabilities

(just enough philosophy)

- e.g., inferring model parameters Θ from observed data $\mathcal{D}$:
  - frequentist approach: calculate parameter setting that maximizes likelihood of observed data (point estimate),

$$\widehat{\Theta} = \underset{\Theta}{\operatorname{argmax}}\, p(\mathcal{D}|\Theta) \tag{7}$$

  - bayesian approach: calculate distribution over parameter settings given data,

$$p(\Theta|\mathcal{D}) = ? \tag{8}$$

interpretations of probabilities
(just enough philosophy)

- using bayes' rule $\neq$ "being bayesian"

interpretations of probabilities
(just enough philosophy)

- using bayes' rule $\neq$ "being bayesian"
- s/bayesian/subjective probabilist/g

outline

1. principles (what we'd like to do)
   - background: joint, marginal, and conditional probabilities
   - bayes' theorem: inverting conditional probabilities
   - bayesian probability: unknowns as random variables
   - bayesian inference: bayesian probability + bayes' theorem
2. practice (what we're able to do)
   - monte carlo methods: representative samples
   - variational methods: bound optimization
   - references
3. application: bayesian inference for network data

bayesian probability + bayes' theorem

- bayesian inference:
  - treat unknown quantities as random variables
  - use bayes' theorem to systematically update prior knowledge in the presence of observed data

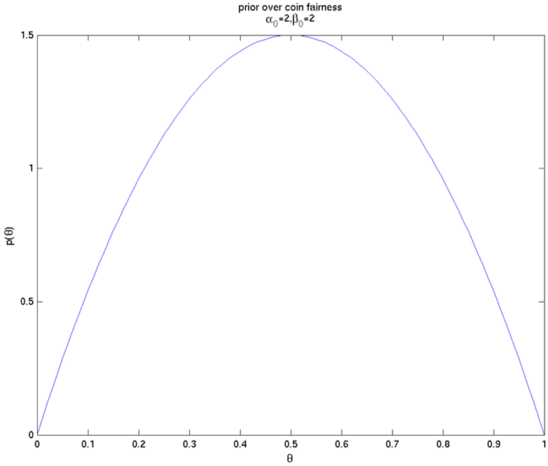
$$\overbrace{p(\Theta|\mathcal{D})}^{posterior} = \frac{\overbrace{p(\mathcal{D}|\Theta)}^{likelihood}\,\overbrace{p(\Theta)}^{prior}}{\underbrace{p(\mathcal{D})}_{evidence}} \tag{9}$$

example: coin flipping

- observe independent coin flips (bernoulli trials)
- infer distribution over coin bias

example: coin flipping

prior $p(\Theta)$ over coin bias before observing flips

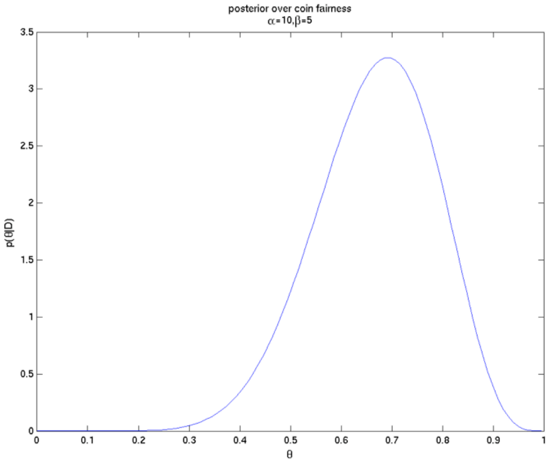


example: coin flipping

observe flips: HTHHHTTHHHH

example: coin flipping

update posterior $p(\Theta|\mathcal{D})$ using bayes' theorem

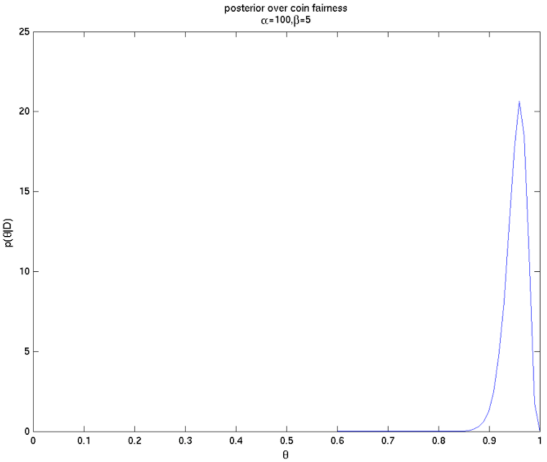


example: coin flipping

observe flips: HHHHHHHHHHHHTHHHHHHHHHH
HHHHHHHHHHHHHHHHHHHHHHH
HHHHHHTHHHHHHHHHHHHHHHH
HHHHHHHHHHHHHHHHHHHHHHH HHHHHHHHT

example: coin flipping

update posterior $p(\Theta|\mathcal{D})$ using bayes' theorem



"naive" "bayes" for document classifiction

- model presence/absence of each word as independent coin flip

$$
\begin{aligned}
p\,(\text{word}|\text{class}) &= \text{Bernoulli}(\theta_{wc}) && (10)\\
p\,(\text{words}|\text{class}) &= p\,(\text{word}_1|\text{class})\,p\,(\text{word}_2|\text{class})\ldots && (11)
\end{aligned}
$$

"naive" "bayes" for document classifiction

- model presence/absence of each word as independent coin flip

$$p(\mathrm{word}|\mathrm{class}) = \mathrm{Bernoulli}(\theta_{wc}) \tag{10}$$

$$p(\mathrm{words}|\mathrm{class}) = p(\mathrm{word}_1|\mathrm{class})\, p(\mathrm{word}_2|\mathrm{class}) \ldots \tag{11}$$

- maximum likelihood estimates of probabilities from word and class counts

$$\hat{\theta}_{wc} = \frac{N_{wc}}{N_c} \tag{12}$$

"naive" "bayes" for document classifiction

- model presence/absence of each word as independent coin flip

$$p\left(\text{word}|\text{class}\right) = \text{Bernoulli}(\theta_{wc}) \quad (10)$$
$$p\left(\text{words}|\text{class}\right) = p\left(\text{word}_1|\text{class}\right) p\left(\text{word}_2|\text{class}\right) \ldots (11)$$

- maximum likelihood estimates of probabilities from word and class counts

$$\hat{\theta}_{wc} = \frac{N_{wc}}{N_c} \quad (12)$$

- use bayes' rule to calculate distribution over classes given words

$$p\left(\text{class}|\text{words}, \Theta\right) = \frac{p\left(\text{words}|\text{class}, \Theta\right) p\left(\text{class}, \Theta\right)}{p\left(\text{words}, \Theta\right)} \quad (13)$$

"naive" "bayes" for document classification

- example: spam filtering for enron email using *one word*[2]

[2] code: `http://github.com/jhofman/ddm/blob/master/2009/lecture_03/enron_naive_bayes.sh`

"naive" "bayes" for document classification

- example: spam filtering for enron email using *one word*[2]

```
$ ./enron1.sh meeting
1500 spam examples
3672 ham examples
16 spam examples containing meeting
153 ham examples containing meeting

estimated P(spam) = .2900
estimated P(ham) = .7100
estimated P(meeting|spam) = .0106
estimated P(meeting|ham) = .0416

P(spam|meeting) = .0923
```

[2] code: `http://github.com/jhofman/ddm/blob/master/2009/lecture_03/enron_naive_bayes.sh`

"naive" "bayes" for document classification

- example: spam filtering for enron email using *one word*[2]

```
$ ./enron1.sh meeting
1500 spam examples
3672 ham examples
16 spam examples containing meeting
153 ham examples containing meeting

estimated P(spam) = .2900
estimated P(ham) = .7100
estimated P(meeting|spam) = .0106
estimated P(meeting|ham) = .0416

P(spam|meeting) = .0923
```

```
$ ./enron1.sh money
1500 spam examples
3672 ham examples
194 spam examples containing money
50 ham examples containing money

estimated P(spam) = .2900
estimated P(ham) = .7100
estimated P(money|spam) = .1293
estimated P(money|ham) = .0136

P(spam|money) = .7957
```

[2] code: `http://github.com/jhofman/ddm/blob/master/2009/lecture_03/enron_naive_bayes.sh`

"naive" "bayes" for document classification

- example: spam filtering for enron email using *one word*[2]

```
$ ./enron1.sh meeting
1500 spam examples
3672 ham examples
16 spam examples containing meeting
153 ham examples containing meeting

estimated P(spam) = .2900
estimated P(ham) = .7100
estimated P(meeting|spam) = .0106
estimated P(meeting|ham) = .0416

P(spam|meeting) = .0923
```

```
$ ./enron1.sh money
1500 spam examples
3672 ham examples
194 spam examples containing money
50 ham examples containing money

estimated P(spam) = .2900
estimated P(ham) = .7100
estimated P(money|spam) = .1293
estimated P(money|ham) = .0136

P(spam|money) = .7957
```

```
$ ./enron1.sh enron
1500 spam examples
3672 ham examples
0 spam examples containing enron
1478 ham examples containing enron

estimated P(spam) = .2900
estimated P(ham) = .7100
estimated P(enron|spam) = 0
estimated P(enron|ham) = .4025

P(spam|enron) = 0
```

[2] code: `http://github.com/jhofman/ddm/blob/master/2009/lecture_03/enron_naive_bayes.sh`

"naive" "bayes" for document classification

- example: spam filtering for enron email using *one word*[2]

```
$ ./enron1.sh meeting
1500 spam examples
3672 ham examples
16 spam examples containing meeting
153 ham examples containing meeting

estimated P(spam) = .2900
estimated P(ham) = .7100
estimated P(meeting|spam) = .0106
estimated P(meeting|ham) = .0416

P(spam|meeting) = .0923
```

```
$ ./enron1.sh money
1500 spam examples
3672 ham examples
194 spam examples containing money
50 ham examples containing money

estimated P(spam) = .2900
estimated P(ham) = .7100
estimated P(money|spam) = .1293
estimated P(money|ham) = .0136

P(spam|money) = .7957
```

```
$ ./enron1.sh enron
1500 spam examples
3672 ham examples
0 spam examples containing enron
1478 ham examples containing enron

estimated P(spam) = .2900
estimated P(ham) = .7100
estimated P(enron|spam) = 0
estimated P(enron|ham) = .4025

P(spam|enron) = 0
```

- guard against overfitting by "smoothing" counts, equivalent to maximum *a posteriori* (map) inference

$$\hat{\theta}_{wc} = \frac{N_{wc} + \alpha}{N_c + \alpha + \beta} \tag{14}$$

[2]code: `http://github.com/jhofman/ddm/blob/master/2009/lecture_03/enron_naive_bayes.sh`

"naive" "bayes" for document classification

- naive bayes for document classification is neither naive nor bayesian!
  - not so naive: works well in practice, not in theory
  - not bayesian: point estimates $\hat{\theta}_{wc}$ for parameters rather than distributions over parameters

quantities of interest

- bayesian inference maintains full posterior distributions over unknowns
- many quantities of interest require expectations under these posteriors, e.g. posterior mean and predictive distribution:

$$\bar{\Theta} = \mathbb{E}_{p(\Theta|\mathcal{D})}\left[\Theta\right] = \int d\Theta \; \Theta \; p(\Theta|\mathcal{D}) \tag{15}$$

$$p\left(x|\mathcal{D}\right) = \mathbb{E}_{p(\Theta|\mathcal{D})}\left[p\left(x|\Theta,\mathcal{D}\right)\right] = \int d\Theta \; p\left(x|\Theta,\mathcal{D}\right) p(\Theta|\mathcal{D}) \tag{16}$$

quantities of interest

- bayesian inference maintains full posterior distributions over unknowns
- many quantities of interest require expectations under these posteriors, e.g. posterior mean and predictive distribution:

$$\bar{\Theta} = \mathbb{E}_{p(\Theta|\mathcal{D})}\left[\Theta\right] = \int d\Theta \; \Theta \; p(\Theta|\mathcal{D}) \tag{15}$$

$$p\left(x|\mathcal{D}\right) = \mathbb{E}_{p(\Theta|\mathcal{D})}\left[p\left(x|\Theta,\mathcal{D}\right)\right] = \int d\Theta \; p\left(x|\Theta,\mathcal{D}\right) p(\Theta|\mathcal{D}) \tag{16}$$

- often can't compute posterior (normalization), let alone expectations with respect to it $\rightarrow$ approximation methods

outline

1. principles (what we'd like to do)
   - background: joint, marginal, and conditional probabilities
   - bayes' theorem: inverting conditional probabilities
   - bayesian probability: unknowns as random variables
   - bayesian inference: bayesian probability + bayes' theorem
2. practice (what we're able to do)
   - monte carlo methods: representative samples
   - variational methods: bound optimization
   - references
3. application: bayesian inference for network data

representative samples

- general approach: approximate intractable expectations via sum over representative samples[3]

$$\Phi = \mathbb{E}_{p(x)}[\phi(x)] = \int dx \underbrace{\phi(x)}_{arbitrary\ \ function} \underbrace{p(x)}_{target\ \ density} \tag{17}$$

[3] follows mackay (2003)

representative samples

- general approach: approximate intractable expectations via sum over representative samples[3]

$$\Phi = \mathbb{E}_{p(x)}\left[\phi(x)\right] = \int dx \underbrace{\phi(x)}_{arbitrary\ \ function} \underbrace{p(x)}_{target\ \ density} \tag{17}$$

$$\Downarrow$$

$$\widehat{\Phi} = \frac{1}{R}\sum_{r=1}^{R} \phi(x^{(r)}) \tag{18}$$

[3] follows mackay (2003)

representative samples

- general approach: approximate intractable expectations via sum over representative samples[3]

$$\Phi = \mathbb{E}_{p(x)}\left[\phi(x)\right] = \int dx \underbrace{\phi(x)}_{arbitrary\ function} \underbrace{p(x)}_{target\ density} \quad (17)$$

$$\Downarrow$$

$$\widehat{\Phi} = \frac{1}{R}\sum_{r=1}^{R} \phi(x^{(r)}) \quad (18)$$

- shifts problem to finding "good" samples

[3] follows mackay (2003)

representative samples

- further complication: in general we can only evaluate the target density to within a multiplicative (normalization) constant, i.e.

$$p(x) = \frac{p^*(x)}{Z} \tag{19}$$

and $p^*(x^{(r)})$ can be evaluated with Z unknown

sampling methods

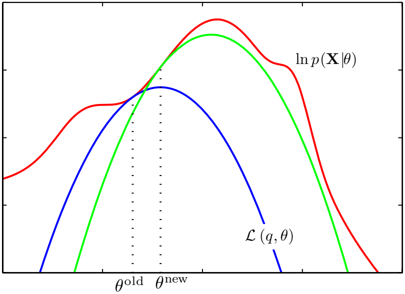
- monte carlo methods
  - uniform sampling
  - importance sampling
  - rejection sampling
  - ...
- markov chain monte carlo (mcmc) methods
  - metropolis-hastings
  - gibbs sampling
  - ...

outline

1. principles (what we'd like to do)
   - background: joint, marginal, and conditional probabilities
   - bayes' theorem: inverting conditional probabilities
   - bayesian probability: unknowns as random variables
   - bayesian inference: bayesian probability + bayes' theorem
2. practice (what we're able to do)
   - monte carlo methods: representative samples
   - variational methods: bound optimization
   - references
3. application: bayesian inference for network data

bound optimization

- general approach: replace integration with optimization
- construct auxiliary function upper-bounded by log-evidence, maximize auxiliary function



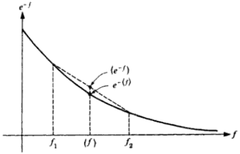
[4]

[4] image from bishop (2006)

variational bayes

- bound log of expected value by expected value of log using jensen's inequality[5]:

$$
\begin{aligned}
-\ln p(\mathcal{D}) &= -\ln \int d\Theta \; p(\mathcal{D}|\Theta)p(\Theta) \\
&= -\ln \int d\Theta \; \frac{p(\mathcal{D}|\Theta)p(\Theta)}{\color{red}{q(\Theta)}} \color{red}{q(\Theta)} \\
&\leq -\int d\Theta \; \ln \left[\frac{p(\mathcal{D}|\Theta)p(\Theta)}{q(\Theta)} \right] q(\Theta)
\end{aligned}
$$



- for sufficiently simple (i.e. factorized) approximating distribution $q(\Theta)$, right-hand side can be easily evaluated and optimized

[5] image from feynman (1972)

variational bayes

- iterative coordinate ascent algorithm provides controlled analytic approxmations to posterior and evidence
- approximate posterior $q(\Theta)$ minimizes kullback-leibler distance to true posterior
- resulting deterministic algorithm is often fast and scalable

variational bayes

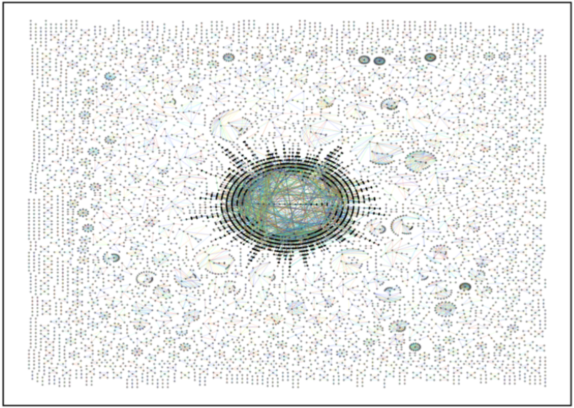
- iterative coordinate ascent algorithm provides controlled analytic approxmations to posterior and evidence
- approximate posterior $q(\Theta)$ minimizes kullback-leibler distance to true posterior
- resulting deterministic algorithm is often fast and scalable
- complexity of approximation often limited (to, e.g., mean-field theory, assuming weak interaction between unknowns)
- iterative algorithm requires restarts, no guarantees on quality of approximation

outline

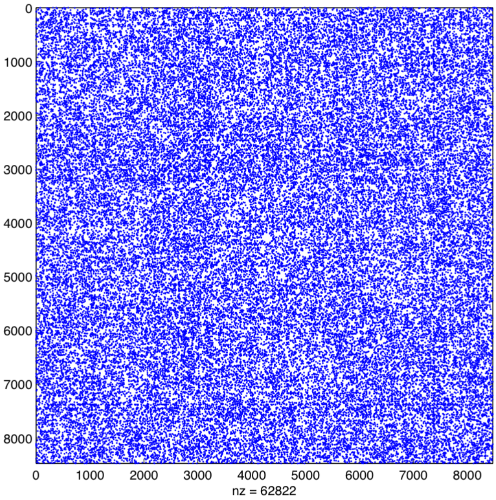
1. principles (what we'd like to do)
   - background: joint, marginal, and conditional probabilities
   - bayes' theorem: inverting conditional probabilities
   - bayesian probability: unknowns as random variables
   - bayesian inference: bayesian probability + bayes' theorem
2. practice (what we're able to do)
   - monte carlo methods: representative samples
   - variational methods: bound optimization
   - references
3. application: bayesian inference for network data

- "information theory, inference, and learning algorithms", mackay (2003)
- "pattern recognition and machine learning", bishop (2006)
- "bayesian data analysis", gelman, et. al. (2003)
- "probabilistic inference using markov chain monte carlo methods", neal (1993)
- "graphical models, exponential families, and variational inference", wainwright & jordan (2006)
- "probability theory: the logic of science", jaynes (2003)
- "what is bayes' theorem ...", wiggins (2006)
- bayesian inference view on cran
- variational-bayes.org
- variational bayesian inference for network modularity

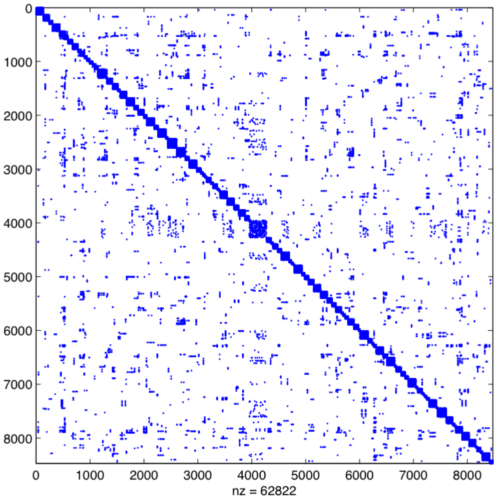
application: bayesian inference for network data



example: a bayesian approach to network modularity

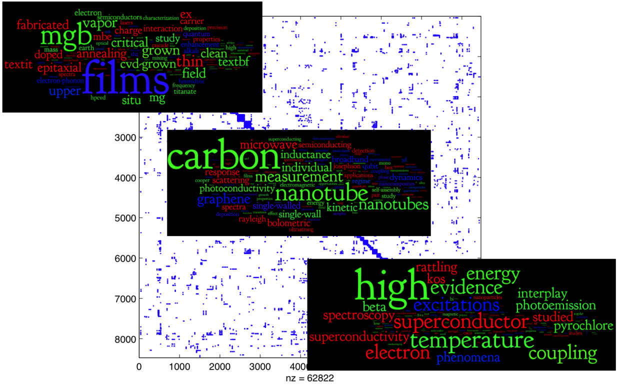


example: a bayesian approach to network modularity



example: a bayesian approach to network modularity

inferred topological communities correspond to sub-disciplines



Thanks.

Questions?[6]

[6] jake@jakehofman.com, @jakehofman