Learning a large pedigree from some nice data

The Taming of the Shrew

Robert Cowell

Cass Business School

Graphical Models and Genetic Applications Workshop,
Easter 2009

Outline

- "Standard" Bayesian network structure learning
- Data used for pedigree reconstruction
- Modelling pedigrees using Bayesian networks
- Pedigree reconstruction algorithm
- Enumeration
- Simulation
- Data on shrews
- Conclusions.

“Standard” Bayesian network learning: simplest case

“Standard” Bayesian network learning: simplest case

- A set of discrete random variables $\{X_i\}$

“Standard” Bayesian network learning: simplest case

- A set of discrete random variables $\{X_i\}$
- A complete dataset of independent “cases” on these variables

“Standard” Bayesian network learning: simplest case

- A set of discrete random variables $\{X_i\}$
- A complete dataset of independent “cases” on these variables
- A node ordering $(X_1, X_2, \cdots, X_n)$

“Standard” Bayesian network learning: simplest case

- A set of discrete random variables $\{X_i\}$
- A complete dataset of independent “cases” on these variables
- A node ordering $(X_1, X_2, \cdots, X_n)$

If G denotes the set of DAGs consistent with node ordering, then for $g \in G$ the log-likelihood decomposes and is readily maximized using marginal counts:

$$\log \hat{L}_g = \sum_i \sum_{x_i, x_{pa(i:g)}} n_{x_i, x_{pa(i:g)}} \log \frac{n_{x_i, x_{pa(i:g)}}}{n_{x_{pa(i:g)}}}$$

“Standard” Bayesian network learning: simplest case

“Standard” Bayesian network learning: simplest case

- Choose g to maximize likelihood by selecting node-parent sets independently for each node.

“Standard” Bayesian network learning: simplest case

- Choose g to maximize likelihood by selecting node-parent sets independently for each node.
- Usually carry out a stepwise search, adding potential parents greedily for maximum increase likelihood. (Recent developments have allowed full enumerative search for n up to around 30 variables.)

"Standard" Bayesian network learning: simplest case

- Choose g to maximize likelihood by selecting node-parent sets independently for each node.
- Usually carry out a stepwise search, adding potential parents greedily for maximum increase likelihood. (Recent developments have allowed full enumerative search for n up to around 30 variables.)
- Usually fast because of ordering and complete data.

“Standard” Bayesian network learning: simplest case

- Choose g to maximize likelihood by selecting node-parent sets independently for each node.
- Usually carry out a stepwise search, adding potential parents greedily for maximum increase likelihood. (Recent developments have allowed full enumerative search for n up to around 30 variables.)
- Usually fast because of ordering and complete data.
- Usually apply some cut-off when testing to add parents, to prevent always obtaining the complete graph.
  - Using marginal likelihood with decomposable Dirichlet prior on parameters avoids need for cut-off.

Data used for pedigree reconstruction

- Assumed population frequencies of STR (short tandem repeat) alleles of marker system.
- Genetic profile information on individuals, consisting of genotypes.
- Sex of individuals.

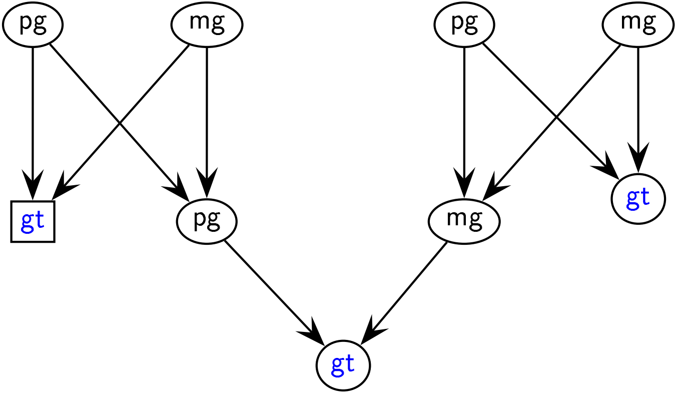
Data used for pedigree reconstruction

- Assumed population frequencies of STR (short tandem repeat) alleles of marker system.
- Genetic profile information on individuals, consisting of genotypes.
- Sex of individuals.
- Age information, if available.

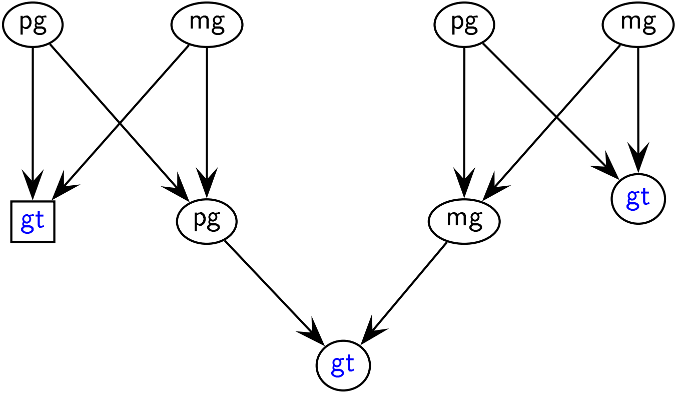
Example: single STR marker, no mutation

Individual	gt	sex	age	possible parent of c?	possible parents of c?
c	(5,8)	M	3	no	no
p1	(6,4)	M	2	no	no
p2	(5,9)	F	8	y	y (with p4)
p3	(5,12)	M	12	y	no
p4	(7,8)	M	7	y	y (with p2 or p5)
p5	(5,7)	F	12	y	y (with p4)

Representation by *pg/mg/gt* triples

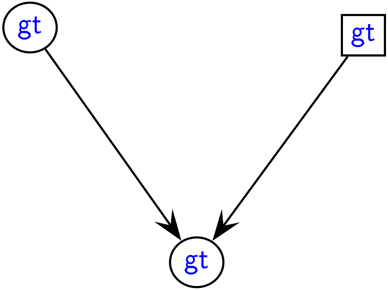


Representation by *pg/mg/gt* triples

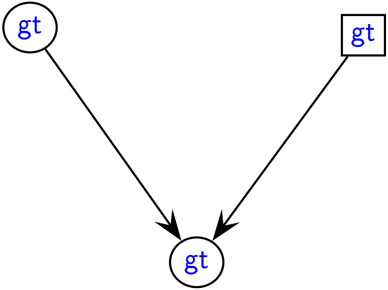


Learning a pedigree network in this representation is an incomplete-data/ latent variable problem, because the *pg* and *mg* values are not observed.

Representation by *gt* triples

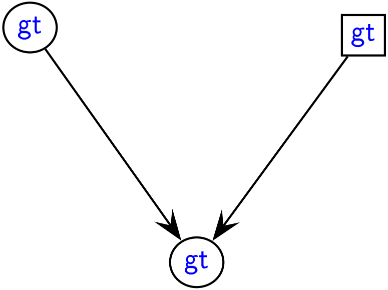


Representation by *gt* triples



No hidden/latent variable nodes: complete data problem.

Representation by *gt* triples



No hidden/latent variable nodes: complete data problem.

Simplify problem further by not including explicitly unmeasured parents or ancestors.

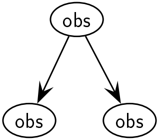
Pedigree reconstruction algorithm

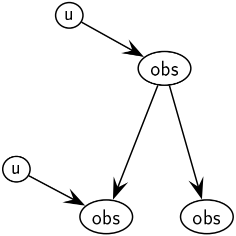
Say an individual is *observed* if their genotype is known.
Restrict pedigree search with the following constraints:

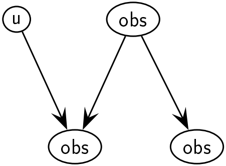
- Any child of an observed individual is observed.
- An unobserved parent has only one child, and that child is observed.

Examples

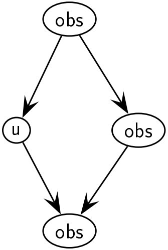
Allowed

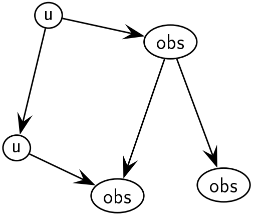


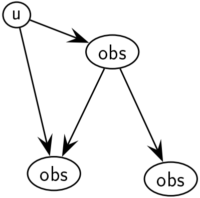




Not allowed







The pedigree likelihood

$$L(gt(X); g) = \prod_{x} P(gt(x) | \, gt(pa(x : g)))$$

Under the restrictions, the pedigree likelihood factorizes into three types of terms.

The pedigree likelihood

$$L(gt(X);g) = \prod_{x} P(gt(x)|\, gt(pa(x:g)))$$

Under the restrictions, the pedigree likelihood factorizes into three types of terms.

1. Terms in which $pa(x:g)$ has two (observed) individuals of opposite sex.

The pedigree likelihood

$$L(gt(X); g) = \prod_{x} P(gt(x) | gt(pa(x : g)))$$

Under the restrictions, the pedigree likelihood factorizes into three types of terms.

1. Terms in which $pa(x : g)$ has two (observed) individuals of opposite sex.
2. Terms in which $pa(x : g)$ has one individual, (and thus x has one observed and one unobserved founder).

The pedigree likelihood

$$L(gt(X); g) = \prod_{x} P(gt(x) | gt(pa(x : g)))$$

Under the restrictions, the pedigree likelihood factorizes into three types of terms.

1. Terms in which $pa(x : g)$ has two (observed) individuals of opposite sex.
2. Terms in which $pa(x : g)$ has one individual, (and thus x has one observed and one unobserved founder).
3. Terms in which $pa(x : g) = \emptyset$, (and thus x is an observed *founder*).

Both parents are observed

Mendelian inheritance: (a, b, c, d distinct alleles). Non-zero values:

$$P(gt_j(x_i) = (a,a) | gt_j(m) = (a,a), gt_j(f) = (a,a)) = 1$$

$$P(gt_j(x_i) = (a,a) | gt_j(m) = (a,a), gt_j(f) = (a,b)) = 0.5$$

$$P(gt_j(x_i) = (a,a) | gt_j(m) = (a,b), gt_j(f) = (a,b)) = 0.25$$

$$P(gt_j(x_i) = (a,a) | gt_j(m) = (a,b), gt_j(f) = (a,c)) = 0.25$$

$$P(gt_j(x_i) = (a,b) | gt_j(m) = (a,a), gt_j(f) = (b,b)) = 1$$

$$P(gt_j(x_i) = (a,b) | gt_j(m) = (a,a), gt_j(f) = (a,b)) = 0.5$$

$$P(gt_j(x_i) = (a,b) | gt_j(m) = (a,b), gt_j(f) = (a,b)) = 0.5$$

$$P(gt_j(x_i) = (a,b) | gt_j(m) = (a,b), gt_j(f) = (b,c)) = 0.25$$

$$P(gt_j(x_i) = (a,b) | gt_j(m) = (a,c), gt_j(f) = (b,c)) = 0.25$$

$$P(gt_j(x_i) = (a,b) | gt_j(m) = (a,c), gt_j(f) = (b,d)) = 0.25$$

One or other of m or f is unobserved, but not both.

Taking $f = \emptyset$, there are several distinct cases to consider:

$$\begin{aligned}
P(gt_j(x_i) = (a,a) | \, gt_j(m) = (a,a)) &= p(a) \\
P(gt_j(x_i) = (a,a) | \, gt_j(m) = (a,b)) &= p(a)/2 \\
P(gt_j(x_i) = (a,a) | \, gt_j(m) = (b,c)) &= 0 \\
P(gt_j(x_i) = (a,b) | \, gt_j(m) = (a,a)) &= p(b) \\
P(gt_j(x_i) = (a,b) | \, gt_j(m) = (a,b)) &= (p(a) + p(b))/2 \\
P(gt_j(x_i) = (a,b) | \, gt_j(m) = (a,c)) &= p(b)/2 \\
P(gt_j(x_i) = (a,b) | \, gt_j(m) = (c,d)) &= 0
\end{aligned}$$

where $p(a)$ is the frequency of the allele a in the population, etc.
Ditto for $m = \emptyset$.

Both parents unobserved

Under Hardy-Weinberg equilibrium:

$$\begin{aligned} P(gt_j(x_i) = (a, a)) &= p(a)^2 \\ P(gt_j(x_i) = (a, b)) &= 2p(a)p(b) \end{aligned}$$

Reconstruction algorithm

Reconstruction algorithm

- Sort individuals by age

Reconstruction algorithm

- Sort individuals by age
- For each individual, find a list of possible mothers

Reconstruction algorithm

- Sort individuals by age
- For each individual, find a list of possible mothers
- For each individual, find a list of possible fathers

Reconstruction algorithm

- Sort individuals by age
- For each individual, find a list of possible mothers
- For each individual, find a list of possible fathers
- For preceding two lists, find a list of possible (mother,father)pairs for each individual.

Reconstruction algorithm

- Sort individuals by age
- For each individual, find a list of possible mothers
- For each individual, find a list of possible fathers
- For preceding two lists, find a list of possible (mother,father)pairs for each individual.
- For each individual, find combination of possible parents to maximize contribution to the likelihood.

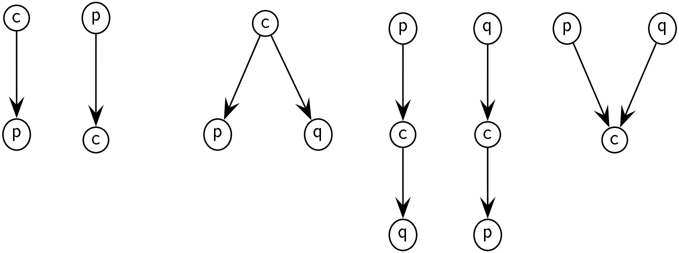
Comparison to “standard” structure learning

Both standard and pedigree DAG learning (can) use decomposable scoring functions.
In pedigree learning:

- Fewer DAGs to search through—number of (graphical) parents is limited to at most two nodes, and in that case, of opposite sex.
- Parent-child genetic constraints reduce the set quite drastically.
- Probability tables are known, they do not need estimation—so no need for ad-hod cut-off parameter: can search for the maximum likelihood DAG
- “Getting more data” means genotyping the individuals on further STR markers.

Orienting arcs (no age information)

- Without age information, cannot tell from a parent-child pair which is the parent using the genotype information.
- If both parents are available, can tell which is the child.



Enumeration: How big is the problem?

- Order individuals by age, oldest first: $s(1), s(2), \ldots, s(n)$
- Let $f(i)$ denote denote the number of females up to but not including $s(i)$ (ie, older than $s(i)$)
- Let $m(i)$ denote the number of males up to but not including $s(i)$.
- So $f(1) = m(1) = 0$.

1. $s(i)$ has no parents represented in the previous set of individuals. This can happen in only one way.
2. $s(i)$'s mother but not father is represented in the previous set of individuals. This can happen in $f(i)$ ways.
3. $s(i)$'s father but not mother is represented in the previous set of individuals. This can happen in $m(i)$ ways.
4. Both of $s(i)$'s parents are represented in the previous set of individuals. This can happen in $f(i)m(i)$ ways.

Number of pedigrees on m males and f females is

$$\prod_{i=1}^{m+f} (1 + f(i))(1 + m(i))$$

Example: $mmffm$

i	1	2	3	4	5
$s(i)$	m	m	f	f	m
$m(i)$	0	1	2	2	2
$f(i)$	0	0	0	1	2
$(1+f(i))(1+m(i))$	1	2	3	6	9

which leads to there being $1 \times 2 \times 3 \times 6 \times 9 = 324$ possible pedigrees.

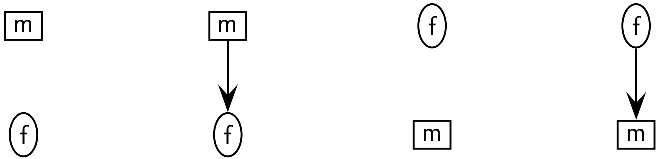
Recurrence relation

Let $A_{f,m}$ denote the number of pedigrees with f females and m males in which the individuals are totally ordered (in unspecified way) by age.

Set $A_{f,-1} = A_{-1,m} = 0$. Then $A_{0,0} = 1$, and

$$A_{f,m} = f(1+m)A_{f-1,m} + m(1+f)A_{f,m-1}$$

Special cases: $A_{0,m} = m!$, $A_{f,0} = f!$



(a) (b) (c) (d)

Total numbers of aged ordered pedigrees: $A(f, m)$

	m			
f	0	1	2	3
0	1	1	2	6
1	1	4	22	156
2	2	22	264	3624
3	6	156	3624	86976
4	24	1368	57168	2249136
5	120	14400	1030320	63528480
6	720	177840	21035520	1966429440
7	5040	2530080	482227200	66633477120
8	40320	40844160	12308647680	2464604755200
9	362880	738823680	347109960960	99139070016000
10	3628800	14816390400	10739259417600	4319958361420800

$$A_{n,n} = O\left(4^n (n!)^4\right) ???$$

Define $B_{f,m}$ by $A_{f,m} = f!m!B_{f,m}$, then

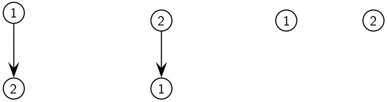
$$B_{f,m} = (1+m)B_{f-1,m} + (1+f)B_{f,m-1}$$

	m				
f	0	1	2	3	4
0	1	1	1	1	1
1	1	4	11	26	57
2	1	11	66	302	1191
3	1	26	302	2416	15619
4	1	57	1191	15619	156190
5	1	120	4293	88234	1310354
6	1	247	14608	455192	9738114
7	1	502	47840	2203488	66318474
8	1	1013	152637	10187685	423281535

Enumerating single sex pedigrees

- n males or n females
- each has at most one parent
- there are no loops
- $\implies$ pedigree is a tree or forest
- $\implies$ number of pedigree on n labelled males/females is the same as number of trees on $n+1$ labelled vertices: Cayley's formula
$$(n+1)^{n-1}$$

Eg $n = 2:$

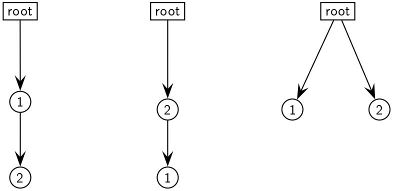


Enumerating single sex pedigrees

- n males or n females
- each has at most one parent
- there are no loops
- $\implies$ pedigree is a tree or forest
- $\implies$ number of pedigree on n labelled males/females is the same as number of trees on $n+1$ labelled vertices: Cayley's formula

$$(n+1)^{n-1}$$

Eg $n = 2:$

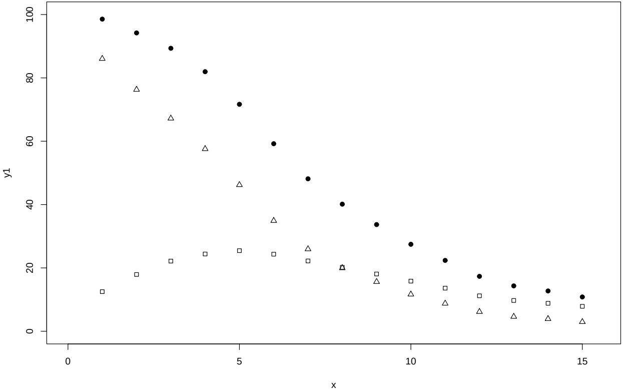


Simulation

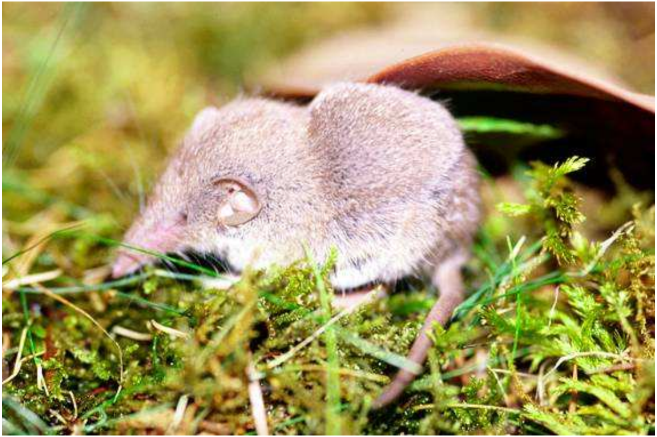
- Average from 1000 simulated networks
- Each network generated had 10 males and 10 females per generation
- 40 generations (making a pedigree of 800 individuals).
- For each network, data on individuals for $1, 2, 3, \ldots, 15$ markers were simulated.

Simulation

Averages of percentage of nodes having incorrect parents. Triangles/Squares/Circles represent individuals for which no parents/exactly one parent/at most one parent respectively were identified correctly. X-axis denotes number of markers used.



Crocidura russala: Greater white-toothed shrew



Background information

- Small mammal
- Monogamous mating cycle
- Can breed after an average of 75 days old gestate for 28 days.
- Live up to four years (in captivity)
- Average of 3.5 litters per year

Data kindly supplied by Caroline Reuter, Imperial College

- Data obtained in the field over the period 1997-2001.
- 890 individuals
- Sex on most, but not all
- Year, and for some day, of birth (for known parents)
- 227 individuals born same year as a parent
- 12 genetic markers (some incomplete)
- Two software systems used for verifying parentage analysis: Probmax and Cervus.
- Geographic and other non-genetic information additionally used to check parentage assignment.

After cleaning

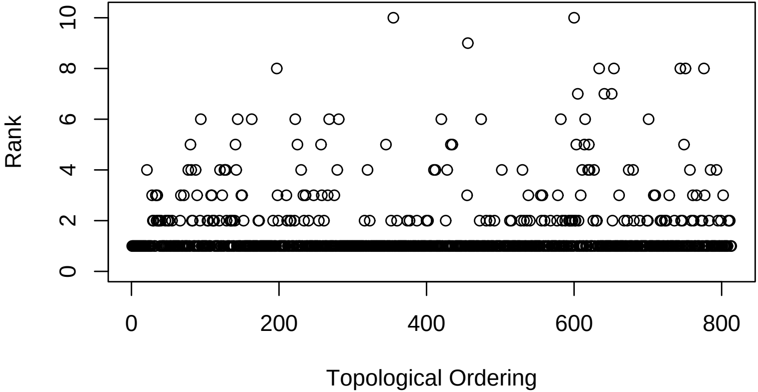
- Remove individuals with incomplete sex or genotype information
- Remove individuals whose parentage assignment was incompatible assuming no mutation.
- This left 813 individuals.

Summary of pedigree search

Rankings of true parentage scores among those found to be possible parents.

Ranking	Count	Ranking	Count
1	599	11	4
2	99	12	1
3	33	13	0
4	26	14	2
5	11	15	1
6	11	16	0
7	3	17	1
8	6	18	2
9	1	19	0
10	2	20	1
		21	10

Rankings of correct parentage scores



Summary

- Brief comparison of Bayesian network and pedigree network learning.
- A brief look at counting pedigrees.
- A simple pedigree reconstruction algorithm
    - Applied to simulated pedigrees of 800 individuals
    - Applied to a real dataset of over 800 wild shrews.

Possible future work

- Relax no-mutation.
- Relax or eliminate total ordering constraint
- Relax absence of unobserved individuals
- Introduce FST corrections.
- Priors over structural elements.

Thank you for listening