ETS

Listening. Learning. Leading.®

Bayesian Networks in Educational Assessment Tutorial, 2nd Edition

April 13, 2018

Russell Almond, FSU
Roy Levy, ASU
Duanli Yan, ETS
Diego Zapata, ETS

Bayesian Networks in Educational Assessment Tutorial, 2nd Edition

Bob Mislevy, Duanli Yan, David Williamson and I first gave this tutorial at the 2002 NCME. At the time it was called “Graphical Models in Educational Assessment” and it was based on the book we were working on (for over a decade). We retitled it “Bayesian Networks in Educational Assessment” and gave it annually from 2004 through 2015. In 2015, our book *Bayesian Networks in Educational Assessment* finally came out. In 2016 Bob and I both took a year off from AERA/NCME.

At the end of the 2015 tutorial, Bob, Duanli, David and I discussed the future of the tutorial. When the tutorial started, Bayes nets were entirely unknown in educational circles, and the content slowly drifted to emphasize the introductory content. At the 2015 session, it was clear that there were now two groups of people who were registering for the tutorial: people who wanted an introduction to Bayes nets, and people who already new lots about Bayes nets but wanted advanced topics like parameter estimation and dynamic Bayesian networks. The decision was to split the tutorial into two pieces: a morning session which would be aimed at newcomers to Bayes nets and an afternoon session aimed at the more advanced topics

We had another problem as well. Bob (who never liked travel much) decided he would rather stay at home and play with his new grandchildren then come teach the tutorial (go figure), and David has been increasingly overwhelmed with managerial duties at ETS so no longer has time to come play with us. We solved this problem by recruiting two new people: Diego Zapata from ETS and Roy Levy from Arizona State University. Diego is a computer scientist who did his dissertation work on visualizing Bayesian networks for education. He has been working for over a decade on various applications of Bayesian networks in simulation and game-based assessments. Roy Levy is one of Bob Mislevy’s students who was involved with the NetPASS project with Cisco, and has since been involved in a large number of Bayes net projects. Diego and Duanli worked on putting together the morning session and Roy and I the afternoon session.

Since the book came out, I have been working on a project called *RNetica* (http://pluto.coe.fsu.edu/RNetica), which provides an R API for the Netica Bayes net engine (http://norsys.com/). This is actually a bundle of four packages: RNetica which provides the interface between R and Netica, CPTtools, which provides engine independent utilities for building Bayes nets and visualizing the output, and Peanut and PNetica, which provide object-oriented support for building parameterized networks. The second half of the revised tutorial will introduce some of the RNetica and CPTtools which can be used for practical applications.

We have a lot of people to thank in the making of this tutorial. Obviously Bob and David played big roles in the development of the original tutorials (and we are still using some of their slides). They also helped with the planning for this 2nd edition. Linda Steinberg was

extremely important in the original development of both evidence-centered assessment design and many of the original applications using Bayesian networks, particularly Biomass. Here project management skills drove us to find practical problems for many issues. Val Shute has generously offered us the Bayesian networks and data from here ACED project to use as examples. Brent Boerlage @ Norsys has generously provided us with time-limited keys for Netica (although many of the class exercises can be done with the student version). More information about both the Netica GUI and API (needed for RNetica to work) can be found at http://norsys.com/. ETS has generously covered the cost of printing and shipping the paper copies of the slides and Springer has been helpful in arranging for copies of the book *Bayesian Networks in Educational Assessment* to be available at the tutorial. Last but not least, we would like to thank the NCME staff and volunteers for arranging a host of important details without which this would be a much less pleasant version.

Two important web sites for the project:

- Tutorial Web Site – http://pluto.coe.fsu.edu/BNinEA/NCMETutorial/. This contains links for all of the sample networks and R scripts used in the tutorial, as well as instructions for running the Netica GUI under Mac and Linux systems.
- ECD Wiki – http://ecd.ralmond.net/ecdwiki/ . This has material about ECD, Bayes nets, examples from the book, and the complete ACED data and networks. Contributors welcome, contact me for an editing password (if its asking you for a password, it probably means that page isn't written yet and it is inviting you to write it).

Finally, we want to thank all of you who have come to the tutorial over the past decade. Your questions and feedback have helped us mold the tutorial to better meet the needs of the NCME audience. We hope that you will continue to provide us with questions and feedback.

Russell Almond
http://ralmond.net/
ralmond@fsu.edu, almond@acm.org
Tallahassee, FL April 17, 2017

Legal Stuff

Sessions III to Russell Almond (ralmond@fsu.edu) and Session IV to Roy Levy (roy.levy@asu.edu).

ACED development and data collection was sponsored by National Science Foundation Grant No. 0313202. Thanks to Val Shute for permission to use ACED data in this tutorial.

Tutorial: Bayesian Networks in Educational Assessment

Russell Almond, Florida State University
Roy Levy, Arizona State University
Duanli Yan, ETS
Diego Zapta, ETS

This is a collection of material related to our 2017 NCME Tutorial. This will also be available via memory stick at the tutorial. See you in New York!

Instructions for Attendees. There is now a "live computing" exercise included in the seminar. To do this we are recommending everybody who can bring a laptop.

If you don't have a laptop, hopefully you will be able to share with somebody who does. We are also recommending you do the following steps:

1. Download student/demonstration version of the software Netica from Norsys. (Other possible software packages are listed below, but we will be preparing the exercises in Netica.) You can try this out using the student/demonstration version, which is sufficient for the exercises in the workshop.
2. Download and install the appropriate version of R from CRAN. You do not need the absolute latest version, if you already have R version 3.X installed, you should be fine.
3. [Optional] Many people prefer to run R from R Studio. You can download the free community edition of R Studio from RStudio.com.
4. Install the package `rjags` from the CRAN library. You can do this by issuing the command `install.packages("rjags")` in R after starting it. This also requires you to load install the program JAGS from http://mcmc-jags.sourceforge.net/.
5. Download the code for CPTtools, RNetica, Peanut and PNetica packages. These packages are not yet on CRAN, but can be found on the RNetica homepage. The following table has the latest versions. [Note that compiling RNetica from source (required for Unix versions) requires downloading the Netica C API from Norsys, see the INSTALL file in the tarball or the RNetica homepage for details.] [Note: Manual build is behind for Peanut and PNetica, may be more recent version next week.]

Package	Source (Unix)	Windows	MacOS	Manual
CPTtools	CPTtools_0.4-2.tar.gz	CPTtools_0.4-2.zip	CPTtools_0.4-4.tgz	CPTtools-manual_0.4-2.pdf
RNetica	RNetica_0.5-2.tar.gz	RNetica_0.5-2.zip	RNetica_0.5-2.tgz	RNetica-manual_0.5-1.pdf
Peanut	Peanut_0.3-4.tar.gz	Peanut_0.3-4.zip	Peanut_0.3-4.tgz	Peanut-manual_0.2-2.pdf
PNetica	PNetica_0.3-4.tar.gz	PNetica_0.3-4.zip	PNetica_0.3-4.tgz	PNetica-manual_0.2-2.pdf

6. Download the example networks to be used (See Under Each session).

Mac and Linux usesrs Netica should run without problems in a variety of Windows emulators. In particular, it should run under WINE. I (Russell) have had success using WINE under both Mac OS X (version 10.6.8 up) and Ubuntu Linux (version 12.04 up). There are several options:

- Linux users: WINE is available through many major Linux distributions (including Macports).
- Mac users: Wineskin is a Mac app for installing Windows programs inside of a special wrapper providing the Windows (i.e., WINE) services. The first time you use Wineskin it downloads extra material (wrappers and engines) from the internet, so do this in a place where you have a good connection.
- Mac only: I have created a Wineskin wrapped version of Netica MacNetica1. This is an unlicensed version of Netica, you still need to purchase a license from Norsys (although the unlicensed student version is adequate for the tutorial).
- Both Linux and Mac: Codeweavers has a commercially supported version of Wine called Crossover Mac or Linux. They give excellent support.

We will have this material on a CD-ROM and Memory stick at the tutorial, so don't worry if you only have a slow internet connection.

Abstract

This tutorial follows the book *Bayesian Networks in Educational Assessment* (Almond, Mislevy, Steinberg, Yan and Williamson, 2015). The first part (Sessions I and II) contain an overview of Bayesian networks (Part I of the book) giving some examples of how they can be used. The second part (Sessions III and IV) look at software and techniques for building networks from expert opinion and data.

Bayesian networks are a technique for managing multidimensional models. By representing the variables of the model as nodes in the graph and using edges in the graph to represent patterns of dependence and independence among the variables, the model graph serves as a bridge between educational and psychometric experts, and further helps the computer derive efficient computational strategies.

This tutorial is based on the book *Bayesian Networks in Educational Assessment* now out from Springer.

Slides and Handouts

I. Evidence Centered Design and Bayesian Networks
    Covers basic models of ECD and their application to Bayes nets. Slides (PDF), Handout (PDF), Session I networks (Netica).

II. Bayes Net Applations including ACED
    This part looks at a number of simple applications of Bayes nets to provide more intution about how they work. Slides (PDF), Handout (PDF). Simple Example Networks (Netica), ACED Subset

(Netica),

III. RNetica and CPTtools

This looks at Tools for using and building Bayesian networks in R, particularly, the CPTtools and RNetica packages. It includes examples in scoring and using the built-in EM algorithm to fit models to data. The talk is split into two sets of slides. RNetica Slides (PDF) RNetica Handout (PDF). mini-ACED (Netica), Learning CPTs Slides (PDF) Learning CPTs Handout (PDF). A simple Learning Example.

IV. Advanced Topics

Covers two topics. Learning with Markov chain Monte Carlo (MCMC) and dynamic Bayesian networks (networks which unfold across time). MCMC Slides (PDF) MCMC Handout (PDF). DBN Slides (PDF) DBN Handout (PDF). All Session IV networks (Netica)

Bibliography

Bayes net and ECD Bibliography (Note: this is an out of date version of the book bibliography).

The handout version is also available as one big file containing all sessions and the bibliography. Honkin' big handout (PDF).

On-line Resources

For quick reference, here are the on-line resources referenced in the bibliography.

Computer programs and documentation available on the Web:

This is a partial list of software packages we have used or think are worth paying attention to. The list of Bayes net software found at the bottom of the Bayesian network Wikipedia entry http://en.wikipedia.org/wiki/Bayesian_network is a reasonably complete and up to date list of both free and commercial software.

Netica (Norsys Software Crop)

http://www.norsys.com/ Netica is another very complete commercial grade Bayes net engine, includes some learning tools.

RNetica (Netica API for R)

http://pluto.coe.fsu.edu/RNetica This is a work in progress binding for the Netica API into the R language. Currently only source verison is available. (Windows and Mac binaries will be available at the conference).

Genie/Smile (Decision Systems Lab, Univ. of Pittsburgh)

http://genie.sis.pitt.edu/ Open source project, free under Gnu Public License. Also contains a ``translator'' which translates between network formats.

Useful (Bayesian) Statistical Software

BUGS (Bayesian inference Using Gibbs Sampling).

http://www.mrc-bsu.cam.ac.uk/bugs/welcome.shtml Downloadable version for Windows. BUGS is no longer actively maintained. For serious work, I recommend OpenBUGS http://mathstat.helsinki.fi/openbugs/

JAGS (Just Another Gibbs Sampler)

https://sourceforge.net/projects/mcmc-jags/ A rewrite of Classic BUGS (command line only, no GUI support) that runs under Linux, MacOS X, and Windows.

FBM: Flexible Bayesian Modeling
: http://www.cs.utoronto.ca/~radford/fbm.software.html Radford Neal's Flexible Bayesian Modeling and Markov Chain Sampler.

R
: http://www.r-project.org/ General purpose statistical computing environment based on S language.

Stan
: http://mc-stan.org/Stan is a package for obtaining Bayesian inference using the No-U-Turn sampler, a variant of Hamiltonian Monte Carlo.

Other On-Line Resources:

ECD Wiki
: http://ecd.ralmond.net/ecdwiki/ Email Russell to get a password to contribute to the discussion.

Book page on the Wiki
: http://ecd.ralmond.net/ecdwiki/BN/BN. We are slowly working at getting sample networks, errata and other resources for working through the book up at this site.

ACED Page on ECD Wiki
: http://ecd.ralmond.net/ecdwiki/ACED/ACED Complete data from ACED field trial and ACED Bayes net are available at this site. This is a Wiki using the same user name and password as the ECD wiki.

Heckerman tutorial on learning (Heckerman, D. [1995])
: ftp://ftp.research.microsoft.com/pub/tr/tr-95-06.pdf Note: Other Microsoft Research technical reports are available on-line from http://www.research.microsoft.com/

Association for Uncertainty in Artificial Intelligence home page
: http://www.auai.org/ UAI conference proceedings is the most important publication in this area.

CRESST Technical Report Archive
: http://www.cse.ucla.edu/products/reports.asp Early versions of many of the Mislevy references (including in press references) are available here. (Hint: search for ``Mislevy''). *The CRESST web site changes frequently, so this link may be out of date. If the link is broken, google "CRESST Reports".*

CiteSeer Cross-Reference Database
: http://citeseer.ist.psu.edu/cis On-line cross reference database with lots of articles on Bayes nets. Many of the bibliography entries are available through CiteSeer.

Copyright

ACED development and data collection was sponsored by National Science Foundation Grant No.

0313202. Thanks to Val Shute for permission to use ACED data in this tutorial.

---

almond (at) acm.org
ralmond (at) fsu.edu
Last modified: Thursday, Apr 5, 2018

Bayesian Networks in Educational Assessment Tutorial

Session I: Evidence Centered Design Bayesian Networks

Duanli Yan, Diego Zapata, ETS
Russell Almond, FSU
Roy Levy, ASU

Agenda

SESSION	TOPIC	PRESENTERS
Session 1:	Evidence Centered Design Bayesian Networks	Duanli Yan & Diego Zapata
Session 2:	Bayes Net Applications ACED: ECD in Action	Duanli Yan & Diego Zapata
Session 3:	Refining Bayes Nets with Data	Russell Almond & Roy Levy
Session 4:	Refining Bayes Nets with Data	Russell Almond & Roy Levy

The Interplay of Design and Statistical Modeling

- Statistical models must be selected/tailored according to the needs of the assessment
- Such selection and adaptation is only meaningful in the larger context of the assessment design
- Understanding the discipline of assessment design is a necessary prerequisite for statistical modeling
- Evidence Centered Design is an assessment design framework with general applicability and utility

Test Design

- Stakeholders
- Requirements
  - Purpose of the test
  - Intended population
- Prospective Score Report
- Evidence-Centered Design
  - Claims
  - Validity
- Specifications

Evidence Centered Design

- Evidence Centered Design (ECD) provides a mechanism for
  - **Capturing and documenting information** about the structure and strength of evidentiary relationships.
  - **Coordinating the work** of test developers in authoring tasks and psychometricians in calibrating the measurement model.
  - **Documenting the scientific information** that provides the foundation for the assessment and its validity.

Evidence Centered Design

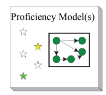
- The Evidence Centered Design process is a series of procedures which center around the questions:
  - "What can we observe about an examinee's performance which will provide evidence that the examinee has or does not have the knowledge, skills and abilities we wish to make claims about?"
  - "How can we structure situations to be able to make those observations?"
- This process results in a formal design for an assessment we call the **Conceptual Assessment Framework (CAF)**

The Initial Frame

- *Why* are we measuring?
  - What are the goals and the desires for use of this assessment?
  - Prospective Score Report
- *Who* are we measuring?
  - Who would take the assessment?
  - Who would view results and for what purpose?
- Goals of the assessment that represent the targets around which the rest of the design process is oriented

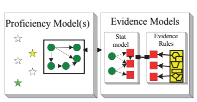
Conceptual Assessment Framework (CAF)

What we measure = Student **Proficiency** Model



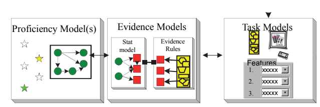
Conceptual Assessment Framework (CAF)

What we measure = Student **Proficiency** Model
How we measure = **Evidence** Model



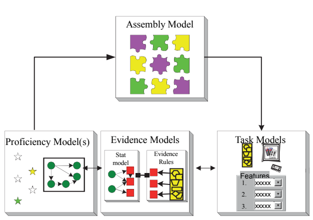
Conceptual Assessment Framework (CAF)

What we measure = Student **Proficiency** Model
How we measure = **Evidence** Model
Where we measure = **Task** Model



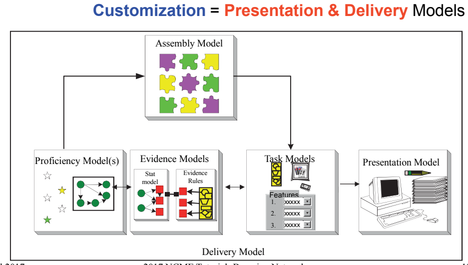
Conceptual Assessment Framework (CAF)

What we measure = Student **Proficiency** Model
How we measure = **Evidence** Model
Where we measure = **Task** Model
How Much we measure = **Assembly** Model



Conceptual Assessment Framework (CAF)

What we measure = Student **Proficiency** Model
How we measure = **Evidence** Model
Where we measure = **Task** Model
How Much we measure = **Assembly** Model
Customization = **Presentation & Delivery** Models



Activity 1: Driver's License Exam

- Redesign the driver's licensure exam
- Write down several claims you would like to make about people who receive a driver's license
- Group your claims into several proficiency variables related to the driver's test
- Do the claims hold for high, medium or low values of those variables?
- Use Netica as a drawing tool and add your variables

Activity 1 (cont)

- List a bunch of activities that you may want prospective drivers to do in their exam
- What is environment of the task
- What are manipulable features of the task?
- Pick one of the tasks you created and build an evidence model for it.
- What are some observable outcomes? their possible values?
- Which proficiencies do they measure?

Activity 1 (cont)

- Think a bit about putting this driver's test together
- How many tasks do we need of what types?
- How much time will be spent in written tests? On the road? In simulators?
- How do we verify the identity of applicants?

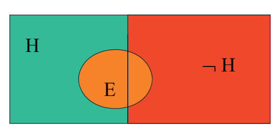
ECD → Bayes Nets

- Represent Qualitative ECD argument with a graph (Domain Modeling) (Session I)
- Turn graphical structure into probability distribution over proficiency variables and observable outcomes (Bayes net; Session I)
- Perform inference (scoring) using that Bayes net (Session II)
- Express probabilities in terms of unknown parameters -- learn parameters (Session III)
- Refine model based on how well it fits data (Session IV)

Cup and Cap notation

- In probability theory, events are sets (sets of balls in the urn).
- Let *A* and *B* be two events
- Either *A or B* occurs
  - Corresponds to *union* of sets
  - $A \cup B$
- Both *A and B* occur
  - Corresponds to *intersection* of sets
  - $A \cap B$
  - Sometimes $Pr(A,B)$
- *Not A* – the balls in the urn that are not in event *A*
  - $\neg A$
  - $Pr(\neg A) = 1\text{-}Pr(A)$

Conditional Probability



- Definition

$$\Pr(E|H) = \frac{\Pr(E \cap H)}{P(H)}$$

- Law of Total Probability

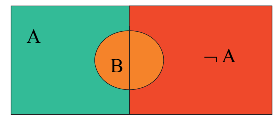
$$\Pr(E) = \Pr(E|H)\Pr(H) + \Pr(E|\neg H)\Pr(\neg H)$$

Bayes Theorem

$$\Pr(H|E) = \frac{\Pr(E|H)\Pr(H)}{\Pr(E)} = \frac{\Pr(E|H)\Pr(H)}{\Pr(E|H)\Pr(H) + \Pr(E|\neg H)\Pr(\neg H)}$$

- Prior $\Pr(H)$
- Likelihood $\Pr(E|H)$
- Posterior $\Pr(H|E)$

Independence



$$\begin{aligned} \Pr(B) &= \Pr(B|A) &= \Pr(B|\neg A) \\ \Pr(A) &= \Pr(A|B) &= \Pr(A|\neg B) \\ \Pr(A \cap B) &= \Pr(B|A)\Pr(B) &= \Pr(A)\Pr(B) \end{aligned}$$

- A provides no information about B

Accident Proneness (Feller, 1968)

- Driving Skill: 5/6 Normal, 1/6 Accident Prone
- Probability of an accident in a given year
  - 1/100 for Normal drivers
  - 1/10 for Accident prone drivers
- Accidents happen independently in each year
- What is the probability a randomly chosen driver will have an accident in Year 1?
- Given a driver had an accident in Year 1, what is probability of accident in Year 2?

Accident Proneness (cont)

- What is the probability a randomly chosen driver will have an accident in Year 1? Year 2?

$$\begin{aligned} P(A_i) &= P(A_i|N)P(N) + P(A_i|\overline{N})P(\overline{N}) \\ &= \frac{.05}{6} + \frac{.1}{6} = \frac{.15}{6} = .025\ . \end{aligned}$$

Accident Proneness (cont)

- Given a driver had an accident in Year 1, what is probability of accident in Year 2?

$$\begin{aligned} P(A_1 \cap A_2) &= P(A_1 \cap A_2|N)P(N) + P(A_1 \cap A_2|\overline{N})P(\overline{N}) \\ &= P(A_1|N)P(A_2|N)P(N) + P(A_1|\overline{N})P(A_2|\overline{N})P(\overline{N}) \\ &= .01 \times .01 \times \frac{5}{6} + .1 \times .1 \times \frac{1}{6} \\ &= \frac{.0005}{6} + \frac{.01}{6} = \frac{.0105}{6} = .00175\ . \end{aligned}$$

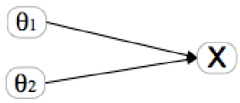
Note that

$$P(A_2|A_1) = \frac{P(A_1 \cap A_2)}{P(A_1)} = \frac{.00175}{.025} = .07\ .$$

Conditional Independence

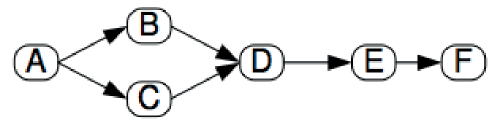
- Years are *conditionally independent* given driving skill
  $$p(Y_1, Y_2|S) = p(Y_1|S)p(Y_2|S)$$
- Years are *marginally dependent*
- Separation in graph tells the story

Competing Explanations



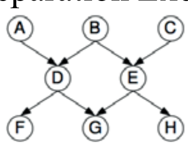
- Skill 1 and Skill 2 are (a priori) independent in population
- Task X requires both skills (conjunctive model)
- Answer the following questions:
  - What is posterior after learning X=False, and θ_1=High?
  - What is posterior after learning X=False, and θ_2=High?
  - What is true of joint posterior of θ_1 and θ_2˜˜af ter learning X=False?

D-Separation



- For ←→, →→, and ←← edges conditioning on middle variables renders outer variables independent
- For →← (collider) edges, if middle variable (or descendent is known) then variables are dependent
- A path is *active* if collider with middle node observed, or non-collider with middle node unobserved

D-Separation Exercise



- Are *A* and *C* independent if
  1. We have observed no other variables?
     - What could we condition on to make *A* and *C* independent?
  2. We have observed *F* and *H?*
     - What else could we condition on to make *A* and *C* independent?
  3. We have observed *G*?
     - What else could we condition on to make *A* and *C* independent?

Building Up Complex Networks

- Recursive representation of probability distributions:

$$p(x_1,\ldots,x_n) = p(x_n \mid x_{n-1},\ldots,x_1)p(x_{n-1} \mid x_{n-2},\ldots,x_1)\cdots p(x_2 \mid x_1)p(x_1)$$

$$= \prod_{j=1}^{n} p(x_j \mid x_{j-1},\ldots,x_1) = \prod_{j=1}^{n} p(x_j \mid Pa(x_j)),$$

- All orderings are equally correct, but some are more beneficial because they capitalize on causal, dependence, time-order, or theoretical relationships that we posit:

  Terms simplify when there is conditional independence – in ed measurement, due to unobservable student variables.

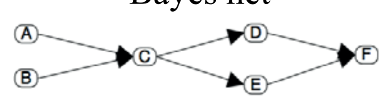
Building Up Complex Networks, cont.

- For example, in IRT, item responses are conditionally independent given θ:

$$p(x_1,\ldots,x_n,\theta)$$
$$= p(x_n \mid x_{n-1},\ldots,x_1,\theta)p(x_{n-1} \mid x_{n-2},\ldots,x_1,\theta)\cdots p(x_2 \mid x_1,\theta)p(x_1 \mid \theta)p(\theta)$$
$$= p(x_n \mid \theta) \qquad p(x_{n-1} \mid \theta) \qquad \cdots p(x_2 \mid \theta) \qquad p(x_1 \mid \theta)p(\theta)$$

$$= \prod_{j=1}^{n} p(x_j \mid \theta)p(\theta).$$

Bayes net



- One factor for each node in graph in recursive representation
- This factor is conditioned on parents in graph
- "Prior" nodes have no parents
- *p(A)p(B)p(C|A,B)p(D|C)p(E|C)p(F|D,E) = p(A,B,C,D,E,F)*
- Digraph must be acyclic

Activity 2: Build a Bayes Net

- Pick one of the tasks you created and build an a Bayes net in Netica:
- Proficiency variables, their possible values
- Observable variables, their possible values
- Conditional probabilities between Proficiency variables and Observable variables
- Add your observables to the proficiency model you made in Netica

Bayesian Networks in Educational Assessment Tutorial

Session II: Bayes Net Applications
ACED: ECD in Action

Duanli Yan, Diego Zapata, ETS
Russell Almond, FSU
Roy Levy, ASU

Agenda

SESSION	TOPIC	PRESENTERS
Session 1:	Evidence Centered Design Bayesian Networks	Duanli Yan & Diego Zapata
Session 2:	Bayes Net Applications ACED: ECD in Action	Duanli Yan & Diego Zapata
Session 3:	Refining Bayes Nets with Data	Russell Almond & Roy Levy
Session 4:	Refining Bayes Nets with Data	Russell Almond & Roy Levy

1. Discrete Item Response Theory (IRT)

- Proficiency Model
- Task/Evidence Models
- Assembly Model
- Some Numbers

IRT Proficiency Model

- There is one proficiency varaible, θ. (Sometimes called an "ability parameter", but we reserve the term *parameter* for quantites which are not person specific.)
- θ takes on values {-2, -1, 0, 1, 2} with prior probabilities of (0.1, 0.2, 0.4, 0.2, 0.1) (Triangular distribution).
- Observable outcome variables are all independent given θ
- Goal is to draw inferences about θ
  - Rank order students by θ
  - Classify students according to θ above or below a cut point

IRT Task/Evidence Model

- Tasks yield an work product which can be unambiguously scored right/wrong.
- Each task has a *single* observable outcome variable.
- *Tasks* are often called *items,* although the common usage often blurs the distinction between the presentation of the item and the outcome variable.

IRT (Rasch) Evidence Model

- Let X_j be observable outcome variable from Task j
- $P(X_j = right \mid \theta, \beta_j) = \frac{1}{1+e^{-(\theta-\beta_j)}}$
  β_j is the *difficulty* of the item.
- Can crank through the formula for each of the five values of θ to get values for Conditional Probability Tables (CPT)

IRT Assembly Model

- 5 items
- Increasing difficulty:
  $\beta \in \{-1.5, -0.75, 0, 0.75, 1.5\}$.
- Adaptive presentation of items

Conditional Probability Tables

θ	Prior	Item 1	Item 2	Item 3	Item 4	Item 5
-2	0.1	0.3775	0.2227	0.1192	0.0601	0.0293
-1	0.2	0.6225	0.4378	0.2689	0.1480	0.0759
0	0.4	0.8176	0.6792	0.5000	0.3208	0.1824
1	0.2	0.9241	0.8520	0.7311	0.5622	0.3775
2	0.1	0.9707	0.9399	0.8088	0.7773	0.6225

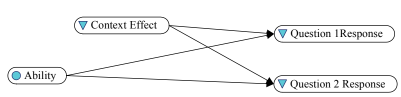
Problems Set 1

1. Assume θ=1, what is expected score (sum X_j)
2. Calculate $P(\theta|X_1$=right), $E(\theta|X_1$=right)
3. Calculate $P(\theta|X_5$=right), $E(\theta|X_5$=right)
4. Score three students who have the following observable patterns (Tasks 1--5):
   1,1,1,0,0
   1,0,0,1,1
   1,1,1,0,1
5. Suppose we have observed for a given student X_2=right and X_3=right, what is the next best item to present (hint, look for expected probabilities closest to .5,.5
6. Same thing, with X_2=right and X_3=wrong
7. Same thing, with X_2=wrong and X_3=wrong

2. "Context" effect --Testlets

- Standard assumption of conditional independence of observable variables given Proficiency Variables
- Violation
  - Shared stimulus
  - Context
  - Special knowledge
  - Shared Work Product
  - Sequential dependencies
  - Scoring Dependencies (Multi-step problem)
- Testlets (Wainer & Kiely, 1987)
- Violation results in overestimating the evidential value of observables for Proficiency Variables

"Context" effect -- Variables



- Context variable – A parent variable introduced to handle conditional dependence among observables (testlet)
  - Consistent with Stout's (1987) 'essential n-dimensionality'
  - Wang, Bradlow & Wainer (2001) SCORIGHT program for IRT
  - Patz & Junker (1999) model for multiple ratings

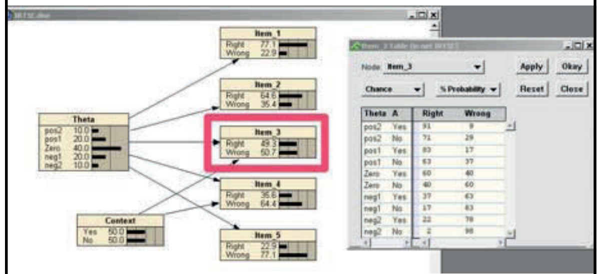
"Context" effect -- example

- Suppose that Items 3 and 4 share common presentation material
- Example: a word problem about "Yacht racing" might use nautical jargon like "leeward" and "tacking"
- People familiar with the content area would have an advantage over people unfamiliar with the content area.
- Would never us this example in practice because of DIF (Differential Item Functioning)

Adding a context variable

- Group Items 3 and 4 into a single task with two observed outcome variables
- Add a person-specific, task-specific latent variable called "context" with values familiar and unfamiliar
- Estimates of θ will "integrate out" the context effect
- Can use as a mathematical trick to force dependencies between observables.

IRT Model with Context Variable



Problem Set 2

- Compare the following quantities in the context and no context models:
  1. $P(X_2)$, $P(X_3)$, $P(X_4)$
  2. $P(\theta|X_2=\underline{right})$, $P(\theta|X_3=\underline{right})$
  3. $P(X_4|X_2=\underline{right})$, $P(X_4|X_3=\underline{right})$
  4. $P(\theta|X_3=\underline{wrong}, X_4=\underline{wrong})$, $P(\theta|X_3=\underline{right}, X_4=\underline{wrong})$,
  5. $P(\theta|X_3=\underline{wrong}, X_4=\underline{right})$, $P(\theta|X_3=\underline{right}, X_4=\underline{right})$

Context Effect Postscript

- If Context effect is generally construct-irrelevant variance, if correlated with group membership this is bad (DIF)
- When calibrating using 2PL IRT model, can get similar joint distribution for θ, X_3, and X_4 by decreasing the discrimination parameter

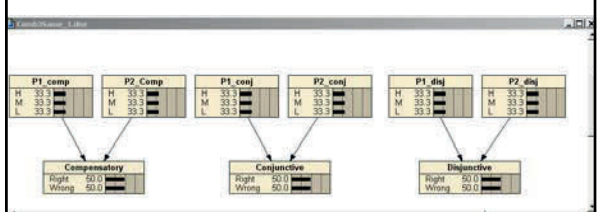
3. Combination Models

Consider a task which requires two Proficiencies:

Three different ways to combine those proficiencies:

- **Compensatory**: More of Proficiency 1 compensates for less of Proficiency 2. Combination rule is *sum*.
- **Conjunctive**: Both proficiencies are needed to solve the problem. Combination rule is *minimum*.
- **Disjunctive**: Two proficiencies represent alternative solution paths to the problem. Combination rule is *maximum*.

Combination Model Graphs



Common Setup for All Three Models

- There are two parent nodes, and both parents are conditionally independent of each other. The difference among the three models lies in the third term below:

$$P(P_1, P_2, X) = P(P_1) \cdot P(P_2) \cdot P(X \mid P_1, P_2)$$

- The priors for the parent nodes are the same for the three models with 0.3333 of probability at each of the H, M, and L states.
- The initial marginal probability for X is the same for the three models (50/50).

Conditional Probability Tables

This table contains the conditional probabilities for the parent nodes (P1 and P2) and the combination model for the three models.

Table 3 – Part 2

Conditional Problems for Compensatory, Conjunctive, and Disjunctive

P1	P2	Compensatory "Right"	Conjunctive "Right"	Disjunctive "Right"
H	H	0.9	0.9	0.7
H	M	0.7	0.7	0.7
H	L	0.5	0.3	0.7
M	H	0.7	0.7	0.7
M	M	0.5	0.7	0.3
M	L	0.3	0.3	0.3
L	H	0.5	0.3	0.7
L	M	0.3	0.3	0.3
L	L	0.1	0.3	0.1

Problem Set 3

1. Verify that $P(P_1)$, $P(P_2)$, and $P(Obs)$ are the same for all three models. (*Obs* represents either the node *Compensatory*, *Conjunctive*, or *Disjunctive*)
2. Assume *Obs*=right, Calculate $P(P_1)$ and $P(P_2)$ for all three models.
3. Assume *Obs*=wrong, Calculate $P(P_1)$ and $P(P_2)$ for all three models.
4. Assume *Obs*=right, and P_1 = H. Calculate $P(P_2)$ for all three models.
5. Assume *Obs*=right, and P_1 = M. Calculate $P(P_2)$ for all three models.
6. Assume *Obs*=right, and P_1 = L. Calculate $P(P_2)$ for all three models.
7. Explain the differences

Activity 3

- Go back to the Driver's License Exam you built in Session I and add some numbers
- Now put in some observed outcomes
  - How did the probabilities change?
  - Is that about what you expected?

ACED Background

- ACED (Adaptive Content with Evidence-based Diagnosis)
- Val Shute (PD), Aurora Graf, Jody Underwood, Eric Hansen, Peggy Redman, Russell Almond, Larry Casey, Waverly Hester, Steve Landau, Diego Zapata
- Domain: Middle School Math, Sequences
- Project Goals:
  - Adaptive Task Selection
  - Diagnostic Feedback
  - Accessibility

ACED Features

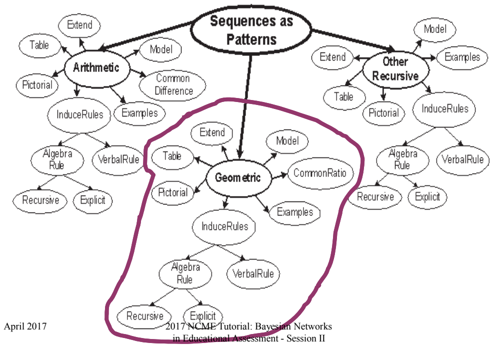
Valid Assessment. Based on evidence-centered design (ECD).

Adaptive Sequencing. Tasks presented in line with an adaptive algorithm.

Diagnostic Feedback. Feedback is immediate and addresses common errors and misconceptions.

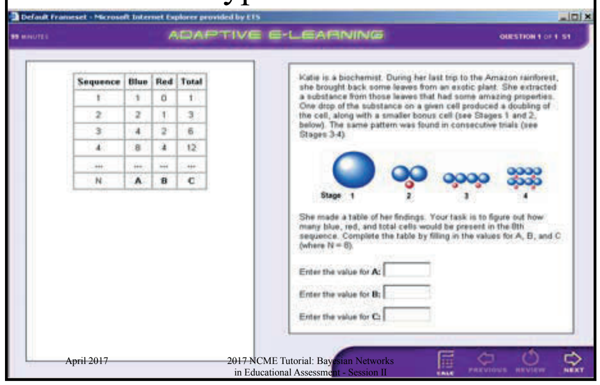
Aligned. Assessments aligned with (a) state and national standards and (b) curricula in current textbooks.

ACED Proficiency Model



April 2017 2017 NCME Tutorial: Bayesian Networks in Educational Assessment - Session II

Typical Task



April 2017 2017 NCME Tutorial: Bayesian Networks in Educational Assessment - Session II

ACED Design/Build Process

- Identify Proficiency variables
- Structure Proficiency Model
- Elicit Proficiency Model Parameters
- Construct Tasks to target proficiencies at Low/Medium/High difficulty
- Build Evidence Models based on difficulty/Q-Matrix

April 2017 2017 NCME Tutorial: Bayesian Networks in Educational Assessment - Session II 27

Parameterization of Network

- Proficiency Model:
  - Based on Regression model of child given parent
  - SME provided correlation and intercept
  - SME has low confidence in numeric values
- Evidence Model Fragment
  - Tasks Scored Right/Wrong
  - Based on IRT model
  - High/Medium/Low corresponds to $\theta = +1/0/-1$
  - Easy/Medium/Hard corresponds to difficulty -1/0/+1
  - Discrimination of 1
  - Used Q-Matrix to determine which node is parent

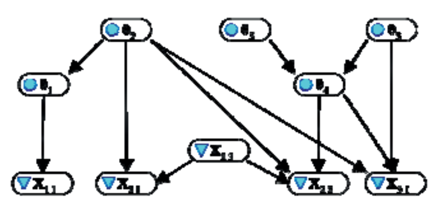
April 2017 2017 NCME Tutorial: Bayesian Networks in Educational Assessment - Session II 28

PM-EM Algorithm for Scoring

- Master Bayes net with just proficiency model(PM)
- Database of Bayes net fragments corresponding to evidence models (EMs), indexed by task ID
- To score a task:
  - Find EM fragment corresponding to task
  - Join EM fragment to PM
  - Enter Evidence
  - Absorb evidence from EM fragment into network
  - Detach EM fragment

April 2017 2017 NCME Tutorial: Bayesian Networks in Educational Assessment - Session II 29

An Example



- Five proficiency variables
- Three tasks, with observables $\{X_{11}\}$, $\{X_{21}, X_{22}, X_{23}\}$, $\{X_{31}\}$.

April 2017 2017 NCME Tutorial: Bayesian Networks in Educational Assessment - Session II 30

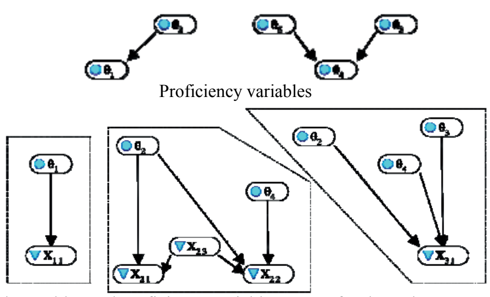
Q: Which observables depend on which proficiency variables?
A: See the Q-matrix (Fischer, Tatsuoka).

	θ_1	θ_2	θ_3	θ_4	θ_5	X_{23}
X_{11}	1	0	0	0	0	--
X_{21}	0	1	0	0	0	1
X_{22}	0	1	0	1	0	1
X_{23}	0	0	0	0	0	N/A
X_{31}	0	1	1	1	0	--

Proficiency Model / Evidence Model Split

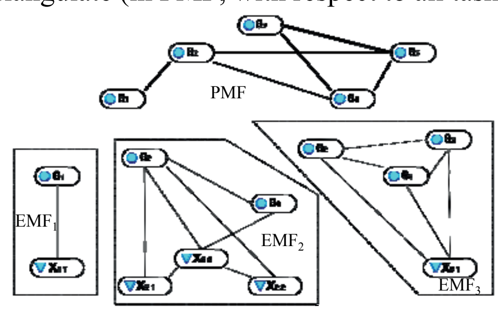
- Full Bayes net for proficiency model and observables for all tasks can be decomposed into fragments.
  - Proficiency model fragment(s) (PMFs) contain proficiency variables.
  - An evidence model fragment (EMF) for each task.
  - EMF contains observables for that task and all proficiency variables that are parents of any of them.
- Presumes observables are conditionally independent between tasks, but can be dependent within tasks.
- Allows for adaptively selecting tasks, docking EMF to PMF, and updating PMF on the fly.

On the way to PMF and EMFs…

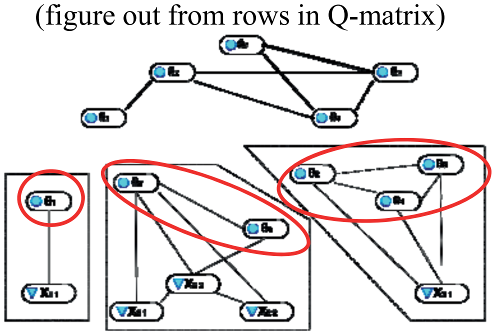


Observables and proficiency variable parents for the tasks

Marry parents, drop directions, and triangulate (in PMF, with respect to all tasks)



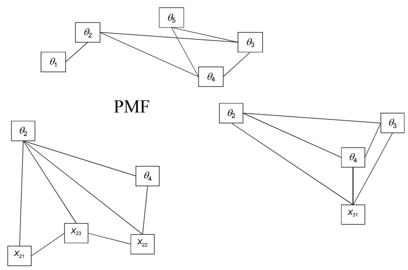
Footprints of tasks in proficiency model (figure out from rows in Q-matrix)



Result:

- Each EMF implies a join tree for Bayes net propagation.
  - Initial distributions for proficiency variables are uniform.
- The footprint of the PM in the EMF is a clique intersection between that EMF and the PMF.
- Can "dock" EMFs with PMF one-at-a-time, to …
  - absorb evidence from values of observables to that task as updated probabilities for proficiency variables, and
  - predict responses in new tasks, to evaluate potential evidentiary value of administering it.

Docking evidence model fragments



Scoring Exercise

Outcome	Task Name	Proficiency Variable	Difficulty
Wrong	tCommonRatio1a.xml	CommonRatio	Easy
Right	tCommonRatio2b.xml	CommonRatio	Medium
Wrong	tCommonRatio3b.xml	CommonRatio	Hard
Wrong	tExplicitGeometric1a.xml	ExplicitGoemetric	Easy
Right	tExplicitGeometric2a.xml	ExplicitGoemetric	Medium
Wrong	tExplicitGeometric3b.xml	ExplicitGoemetric	Hard
Wrong	tRecursiveRuleGeometric1a.xml	RecursiveRuleGeometric	Easy
Wrong	tRecursiveRuleGeometric2b.xml	RecursiveRuleGeometric	Medium
Wrong	tRecursiveRuleGeometric3a.xml	RecursiveRuleGeometric	Hard
Right	tTableExtendGeometric1a.xml	TableGeometric	Easy
Right	tTableExtendGeometric2b.xml	TableGeometric	Medium
Right	tTableExtendGeometric3a.xml	TableGeometric	Hard
Wrong	tVerbalRuleExtendModelGeometric1a.xml	VerbalRuleGeometric	Easy
Wrong	tVerbalRuleExtendModelGeometric1b.xml	VerbalRuleGeometric	Easy
Right	tVerbalRuleExtendModelGeometric2a.xml	VerbalRuleGeometric	Medium
Wrong	tVisualExtendGeometric1a.xml	VisualGeometric	Easy
Wrong	tVisualExtendGeometric2a.xml	VisualGeometric	Medium
Wrong	tVisualExtendGeometric3a.xml	VisualGeometric	Hard

Weight of Evidence

- Good (1985)
- *H* is binary hypothesis, e.g., *Proficiency* > <u>Medium</u>
- *E* is evidence for hypothesis
- Weight of Evidence (WOE) is

$$W(H:E) = \log \frac{\Pr(E|H)}{\Pr(E|\overline{H})} = \log \frac{\Pr(H|E)}{\Pr(\overline{H}|E)} - \log \frac{\Pr(H)}{\Pr(\overline{H})}$$

Properties of WOE

- "Centibans" (log base 10, multiply by 100)
- Positive for evidence supporting hypothesis, negative for evidence refuting hypothesis
- Movement in tails of distribution as important as movement near center
- Bayes theorem using log odds

Conditional Weight of Evidence

- Can define Conditional Weight of Evidence

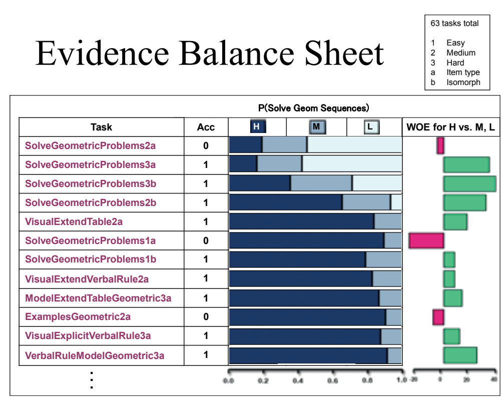
$$W(H:E_2|E_1) = \log \frac{\Pr(E_2|H,E_1)}{\Pr(E_2|\overline{H},E_1)}$$

- Nice Additive properties

$$W(H:E_1,E_2) = W(H:E_1) + W(H:E_2|E_1)$$

- Order sensitive
- WOE Balance Sheet (Madigan, Mosurski & Almond, 1997)

Evidence Balance Sheet



Expected Weight of Evidence

When choosing next "test" (task/item) look at expected value of WOE where expectation is taken wrt *P(E|H)*.

$$EW(H:E) = \sum_{j=1}^{n} W(H:e_j)\Pr(e_j \mid H)$$

where $\{e_j, j = 1, ..., n\}$ represent the possible results.

Calculating EWOE

Madigan and Almond (1996)

- Enter any observed evidence into net

1. Instantiate Hypothesis = True (may need to use virtual evidence if hypothesis is compound)
2. Calculate $\mathrm{P}(E_i|H)$ for each candidate item
3. Instantiate Hypothesis = False
4. Calculate $\mathrm{P}(E_i|\overline{H})$ for each candidate item

Related Measures

- Value of Information

$$\mathrm{VoI}(T) = \mathrm{E}_T\left[\max_d \mathrm{E}_{\mathbf{S}} u(d,\mathbf{S}) - \max_d \mathrm{E}_{\mathbf{S}|T} u(d,\mathbf{S})\right]$$

- **S** is proficiency state
- *d* is decision
- *u* is utility

Related Measures (2)

- Mutual Information
- Extends to non-binary hypothesis nodes

$$\sum_{x,y} \mathrm{P}(x,y) \log \frac{\mathrm{P}(x,y)}{\mathrm{P}(x)\mathrm{P}(y)}$$

- Kullback-Liebler distance between joint distribution and independence

$$\sum_{x} \mathrm{P}(x) \sum_{y} \mathrm{P}(y|x) \log \frac{\mathrm{P}(y|x)}{\mathrm{P}(y)}$$

Task Selection Exercise 1

- Use ACEDMotif1.dne
  - Easy, Medium, and Hard tasks for Common Ratio and Visual Geometric
- Use Hypothesis *SolveGeometricProblems* > <u>Medium</u>
- Calculate EWOE for six observables
- Assume candidate gets first item right and repeat
- Next assume candidate gets first item wrong and repeat
- Repeat exercise using hypothesis *SolveGeometricProblems* > <u>Low</u>

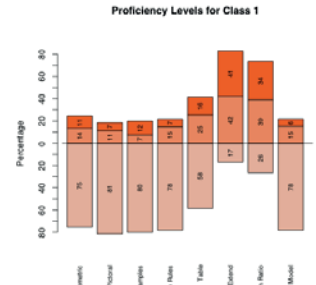
Task Selection Exercise 2

- Use Network ACEDMotif2.dne
- Select the *SolveGeometricProblems* node
- Run the program Network>Sensitivity to Findings
- This will list the Mutual information for all nodes
- Select the observable with the highest mutual information as the first task
- Use this to process a person who gets every task right
- Use this to process a person who gets every task wrong

ACED Evaluation

- Middle School Students
- Did not normally study geometric series
- Four conditions:
  - Elaborated Feedback/Adaptive (E/A; n=71)
  - Simple Feedback/Adaptive (S/A; n=75)
  - Elaborated Feedback/Linear (E/L; n=67)
  - Control (no instruction; n=55)
- Students given all 61 geometric items
- Also given pretest/posttest (25 items each)

ACED Scores



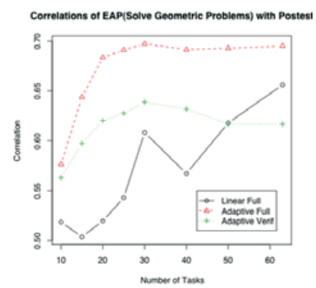
- For Each Proficiency Variable
  - Marginal Distribution
  - Modal Classification
  - EAP Score (High=1, Low=-1)

ACED Reliability

Proficiency (EAP)	Reliability
Solve Geometric Sequences (SGS)	*0.88*
Find Common Ratio	0.90
Generate Examples	0.92
Extend Sequence	0.86
Model Sequence	0.80
Use Table	0.82
Use Pictures	0.82
Induce Rules	0.78
Number Right	0.88

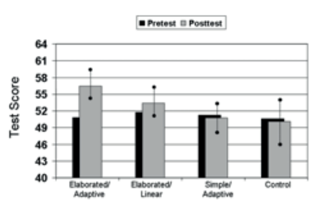
- Calculated with Split Halves (ECD design)
- Correlation of EAP score with posttest is 0.65 (close to reliability of posttest)
- Even with pretest forced into the equation, EAP score accounted for 17% unique variance
- Reliability of modal classifications was worse

Effect of Adaptivity



- For adaptive conditions, correlation with posttest seems to hit upper limit by 20 items
- Standard Error of Correlations is large
- Jump in linear case related to sequence of items

Effect of feedback



- E/A showed significant gains
- Others did not
- Learning and assessment reliability!!!!!

Acknowledgements

- Special thanks to Val Shute for letting us used ACED data and models in this tutorial.
- ACED development and data collection was sponsored by National Science Foundation Grant No. 0313202.
- Complete data available at: http://ecd.ralmond.net/ecdwiki/ACED/ACED

RNetica

Quick Start Guide

Scoring A Student

Copyright 2017-8 Russell G. Almond

Downloading

- http://pluto.coe.fsu.edu/RNetica/
- Four Packages:
  - RNetica – R to Netica link
  - CPTtools – Design patterns for CPTs
  - Peanut/PNetica -- Object-Oriented Parameterized Network
- Source & binary version (Win 64, Mac OS X)
  - Binary versions include Netica.dll/libNetica.so
    - In RStudio select "Package Archive" rather than CRAN
  - Source version need to download from http://www.norsys.com/ first
    - See INSTALLATION

License

- R – GPL-3 (Free and open source)
- RNetica – Artistic (Free and open source)
- Netica.dll/libNetica.so – Commercial (open API, but not open source)
  - Free Student/Demo version
    - Limited number of nodes
    - Limited usage (education, evaluation of Netica)
  - Paid version (see http://www.norsys.com/ for price information)
    - Need to purchase API not GUI version of Netica
    - May want both (use GUI to visualize networks build in RNetica)
- CPTtools – Artistic (Free and open source), does not depend on Netica

Installing the License Key

- When you purchase a license, Norsys will send you a license key. Something that looks like: "+Course/FloridaSU/Ex15-05-30,120,310/XXXXX" (Where I've obscured the last 5 security digits)
- To install the license key, start R in your project directory and type:

```
> DefaultNeticaSession <-
NeticaSession(LicenseKey="+Course/FloridaSU/
Ex15-05-30,120,310/XXXXX")
> q("yes")
```

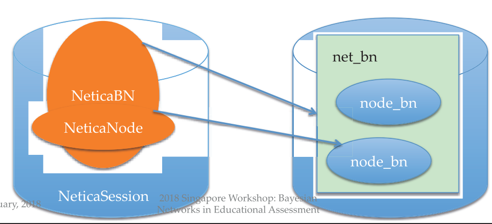
- Restart R and type

```
> library(RNetica)
>  startSession(DefaultNeticaSession)
```

- If license key is not installed, then you will get the limited/student mode. Most of these examples will run

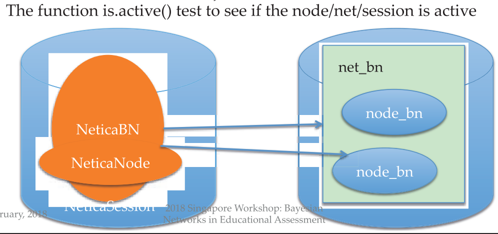
The R heap and the Netica heap

- R and Netica have two different workspaces (memory heaps)
- R workspace is saved and restored automatically when you quick and restart R.
- Netica heap must be reconnected manually.



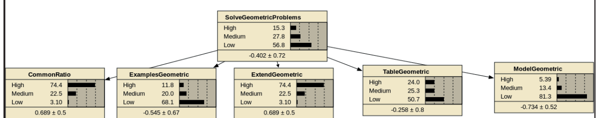
Active and Inactive pointers

- When RNetica creates/finds a Netica object it creates a corresponding R object
- R NeticaBN objects live in the NeticaSession object. R NeticaNode objects live in the NeticaBN.
- If the pointer gets broken (saving & restarting R, deleting the network/node) then the R object becomes inactive.
- The function is.active() test to see if the node/net/session is active



Mini-ACED Proficiency model

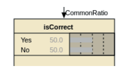
- Subset of ACED network (Shute, Hansen & Almond (2008); http://ecd.ralmond.net/ecdwiki/ACED)
- Proficiency Model subset:



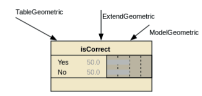
Session IIIa -- RNetica Quick Start 7

Mini-ACED EM Fragments

- All ACED tasks were scored correct/ incorrect
- Each evidence model is represented by a fragment consisting of observables with *stub* edges indicating where it should be *adjoined* with the network.



Common Ratio Easy



Model Extend Table Hard

Session IIIa -- RNetica Quick Start 8

Task to EM map

- Need a table to tell us which EM to use with which task

Task ID	EM Filename	X	Y
tCommonRatio1b	CommonRatioEasyEM	108	414
tCommonRatio2a	CommonRatioMedEM	108	534
tCommonRatio2b	CommonRatioMedEM	108	654
tCommonRatio3a	CommonRatioHardEM	108	774
tCommonRatio3b	CommonRatioHardEM	108	894
tExamplesGeometric1a	ExamplesEasyEM	342	294
tExamplesGeometric1b	ExamplesEasyEM	342	414

Session IIIa -- RNetica Quick Start 9

Scoring Script

- Follow along using the script found in `ScoringScript.R` in the miniACED folder.
- Don't forget to `setwd()` to the miniACED folder (as it needs to find its networks).
- Don't forget to start the Netica session using the license key (if you have one).

Session IIIa -- RNetica Quick Start 10

Loading and starting the Session

```
## Scoring Script
## Preliminaries
library(RNetica)
library(CPTtools)

## start the session
sess <- NeticaSession(<key>)
startSession(sess)
```

Session IIIa -- RNetica Quick Start 11

Reloading Nets and Nodes

```
## Read in network - Do this every time R is restarted
profModel <- ReadNetworks("miniACEDPnet.dne")
## If profModels already exists could also use

## Reconnect nodes - Do this every time R is restarted
allNodes <- NetworkAllNodes(profModel)
sgp <- allNodes$SolveGeometricProblems
profNodes <- NetworkNodesInSet(profModel,"Proficiencies")
```

Session IIIa -- RNetica Quick Start 12

Aside 1: Node Sets

- Netica defines a node set functionality which
  - Adds a collection of labels (sets) to each node
  - Defines a collection of nodes with that label
- Netica GUI really only offers the opportunity to color nodes by set
- RNetica can loop over node sets (lists of nodes)

```
## Node Sets
NetworkNodeSets(profModel)
NetworkNodesInSet(profModel,"pnodes")
NodeSets(sgp)

## These are all settable
NodeSets(sgp) <- c(NodeSets(sgp),"HighLevel")
NodeSets(sgp)
```

Session IIIa -- RNetica Quick Start 13

Aside 2: RNetica Functions

```
## Querying Nodes
NodeStates(sgp)   #List states
NodeParents(sgp)  #List parents
NodeLevels(sgp)   #List numeric values associated with
states
NodeProbs(sgp) # Conditional Probability Table (as array)
sgp[] # Conditional Probability Table (as data frame)
## These are all settable (can be used on RHS of <-) for
## model construction

## Inference
CompileNetwork(profModel) #Lightning bolt on GUI
## Must do this before inference
## Recompiling an already compiled network is harmless
```

Session IIIa -- RNetica Quick Start 14

Aside 2: Inference

```
## Enter Evidence by setting values for these
functions
NodeValue(sgp) #View or set the value
NodeLikelihood(sgp) #Virtual evidence

## Query beliefs
NodeBeliefs(sgp) #Current probability (given entered
evidence)
NodeExpectedValue(sgp) #If node has values, EAP
## These aren't settable

## Retract Evidence
RetractNodeFinding(profNodes$ExamplesGeometric)
RetractNetFindings(profModel)
```

Session IIIa -- RNetica Quick Start 15

Aside 2: Example

```
## Enter Evidence
NodeValue(profNodes$CommonRatio) <- "Medium"
## Enter Evidence "Not Low" ("High or Medium")
NodeLikelihood(profNodes$ExamplesGeometric) <-
c(1,1,0)

NodeBeliefs(sgp) #Current probability (given entered
evidence)
NodeExpectedValue(sgp) #If node has values, EAP

## Retract Evidence
RetractNetFindings(profModel)

## Many more examples
help(RNetica)
```

Session IIIa -- RNetica Quick Start 16

Back to work

- Load the evidence model table
- Row names are task IDs
- EM column contains evidence model name
- EM filename has suffix ".dne" attached.

```
## Read in task->evidence model mapping
EMtable <-
read.csv("MiniACEDEMTable.csv",row.names=1,
                        as.is=2) #Keep EM names
as strings
head(EMtable)
```

Session IIIa -- RNetica Quick Start 17

A student walks into the test center …

- Student gives the name "Fred"
- Student is the right grade/age for ACED (8^{th} or 9^{th} grader, pre-algebra)
- Bayes net has three states
  - Fred logs into ACED
  - Fred attempts the task `tCommonRatio1a` and gets it right
  - Fred attempts the task `tCommonRatio2a` and gets it wrong

Session IIIa -- RNetica Quick Start 18

Start a new student

```
## Copy the master proficiency model
## to make student model
Fred.SM <- CopyNetworks(profModel,"Fred")
Fred.SMvars <- NetworkAllNodes(Fred.SM)
CompileNetwork(Fred.SM)

## Setup score history
prior <-
NodeBeliefs(Fred.SMvars$SolveGeometricProblems)
Fred.History <- matrix(prior,1,3)
row.names(Fred.History) <- "*Baseline*"
colnames(Fred.History) <- names(prior)
Fred.History
```


Score 1st Task

```
### Fred does a task
t.name <- "tCommonRatio1a"
t.isCorrect <- "Yes"

## Adjoin SM and EM
EMnet <-
ReadNetworks(paste(EMtable[t.name,"EM"],"dne",sep=".
"))
obs <- AdjoinNetwork(Fred.SM,EMnet)
NetworkAllNodes(Fred.SM)
## Fred.SM is now the Motif for the current task.
CompileNetwork(Fred.SM)

## Enter finding
NodeFinding(obs$isCorrect) <- t.isCorrect
```


Stats and Cleanup for 1st task

```
## Calculate statistics of interest
post <-
NodeBeliefs(Fred.SMvars$SolveGeometricProblems)
Fred.History <- rbind(Fred.History,new=post)
rownames(Fred.History)[nrow(Fred.History)] <-
paste(t.name,t.isCorrect,sep="=")
Fred.History

## Cleanup and Observable no longer needed, so
absorb it:
DeleteNetwork(EMnet) ## Delete EM
##AbsorbNodes(obs)
## Currently, there is a Netica bug with Absorb
Nodes, we will leave
## this node in place as that is mostly harmless.
```


2nd Task

```
### Fred does another task
t.name <- "tCommonRatio2a"
t.isCorrect <- "No"

EMnet <- ReadNetworks(paste(EMtable[t.name,"EM"],"dne",sep="."))
obs <- AdjoinNetwork(Fred.SM,EMnet)
NetworkAllNodes(Fred.SM)
## Fred.SM is now the Motif for the current task.
CompileNetwork(Fred.SM)

NodeFinding(obs[[1]]) <- t.isCorrect
post <- NodeBeliefs(Fred.SMvars$SolveGeometricProblems)
Fred.History <- rbind(Fred.History,new=post)
rownames(Fred.History)[nrow(Fred.History)] <-
      paste(t.name,t.isCorrect,sep="=")
Fred.History

## Cleanup:  Delete EM and Absorb observables
DeleteNetwork(EMnet) ## Delete EM
## AbsorbNodes(obs)
```


Save and Restore

```
## Fred logs out
WriteNetworks(Fred.SM,"FredSM.dne")
DeleteNetwork(Fred.SM)
is.active(Fred.SM)
## No longer active in Netica space

## Fred logs back in
Fred.SM <- ReadNetworks("FredSM.dne")
is.active(Fred.SM)
```


Getting Serious

- ACED field test has 230 students attempt all 63 tasks.
- File miniACED-Geometric contains 30 task subset
  - There may be data registration issues here, don't publish using these data before checking with me for an update
- Each row is one student Record
- Lets score the first student
  - And build a score history

Setup for mini-ACED

```
miniACED.data <- read.csv("miniACED-Geometric.csv",row.names=1)
head(miniACED.data)
names(miniACED.data)
## Mark columns of table corresponding to tasks
first.task <- 9
last.task <- ncol(miniACED.data)
## Code key for numeric values
t.vals <- c("No","Yes")
```


Setup new Student

```
## Pick a student, we might normally iterate over this.
Student.row <- 1

## Setup for student in sample
## Create Student Model from Proficiency Model
Student.SM <- CopyNetworks(profModel,"Student")
Student.SMvars <- NetworkAllNodes(Student.SM)
CompileNetwork(Student.SM)

## Initialize history list
prior <-
NodeBeliefs(Student.SMvars$SolveGeometricProblems)
Student.History <- matrix(prior,1,3)
row.names(Student.History) <- "*Baseline*"
colnames(Student.History) <- names(prior)
```


Loop Part 1: Add Evidence

```
## Now loop over tasks
for (itask in first.task:last.task) {

  ## Look up the EM for the task, and adjoin it.
  tid <- names(miniACED.data)[itask]
  EMnet <-
ReadNetworks(paste(EMtable[tid,"EM"],"dne",sep="."))
  obs <- AdjoinNetwork(Student.SM,EMnet)
  CompileNetwork(Student.SM)

  ## Add the evidence
  t.val <- t.vals[miniACED.data[Student.row,itask]]
#Decode integer
  NodeFinding(obs[[1]]) <- t.val
```


Loop Part 2: Capture Statistics

```
## Update the history
  post <-
NodeBeliefs(Student.SMvars$SolveGeometricProblem
s)
  Student.History <-
rbind(Student.History,new=post)
  rownames(Student.History)
[nrow(Student.History)] <-
paste(tid,t.val,sep="=")

  ## Cleanup, Delete EM and Absob Observables
  DeleteNetwork(EMnet)
  ## AbsorbNodes(obs) # Still broken
}
```


Weight of Evidence

- Good (1985)
- H is binary hypothesis, e.g., *Proficiency* > Medium
- E is evidence for hypothesis
- Weight of Evidence (WOE) is

$$W(H:E) = \log \frac{\Pr(E|H)}{\Pr(E|\overline{H})} = \log \frac{\Pr(H|E)}{\Pr(\overline{H}|E)} - \log \frac{\Pr(H)}{\Pr(\overline{H})}$$

Conditional Weight of Evidence

- Can define Conditional Weight of Evidence

$$W(H:E_2|E_1) = \log \frac{\Pr(E_2|H,E_1)}{\Pr(E_2|\overline{H},E_1)}$$

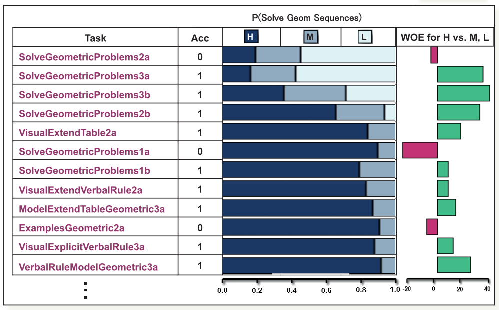
- Nice Additive properties

$$W(H:E_1,E_2) = W(H:E_1) + W(H:E_2|E_1)$$

- Order sensitive
- WOE Balance Sheet (Madigan, Mosurski & Almond, 1997)

Evidence Balance Sheet

63 tasks total
1 Easy
2 Medium
3 Hard
a Item type
b Isomorph



Weight of Evidence Balance Sheet

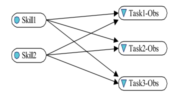
```
## Now examine scoring history
head(Student.History)
woeBal(Student.History,c("High","Medium"),"Low",
       title=paste("Evidence Balance Sheet for ",
       rownames(miniACED.data)[Student.row]))

## More ways to display scores
help(CPTtools)
```

Learning CPTs

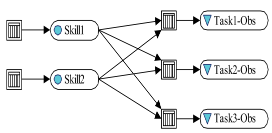
Thanks to Bob Mislevy for letting me use some of the slides from his class.

First Layer



- A simple model with two skills and 3 observables

Distributions and Variables



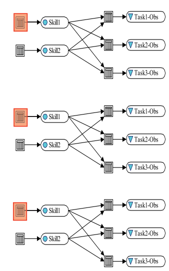
- Variables (values are person specific)
- *Distributions* provide probabilities for variables

Different People, Same Distributions

Student 1

Student 2

Student 3

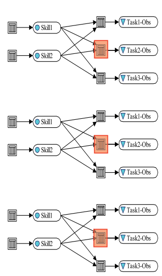


Different People, Same Distributions

Student 1

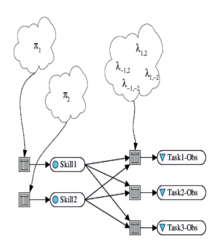
Student 2

Student 3



Second Layer

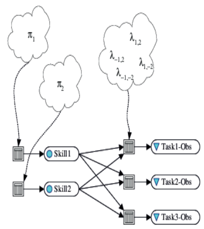
- Distributions have Parameters
- Parameters are the same across all people
- Parameters drop down into first layer to do person specific computations (e.g., scoring)



Probability distributions of parameters are called *Laws*

Second Layer

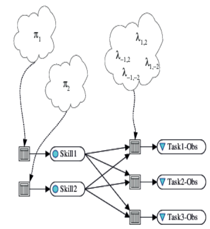
- Distributions have Parameters
- Parameters are the same across all people
- Parameters drop down into first layer to do person specific computations (e.g., scoring)



Probability distributions of parameters are called *Laws*

Second Layer

$$\pi_1 = \Pr(\bigcirc \text{Skill1})$$
$$\pi_1 = \Pr(\bigcirc \text{Skill2})$$
$$\lambda_{1,2} = \Pr(\bigtriangledown \text{Task1} - \text{obs} | \bigcirc \text{Skill1}, \bigcirc \text{Skill2})$$
$$\lambda_{-1,2} = \Pr(\bigtriangledown \text{Task1} - \text{obs} | \neg\bigcirc \text{Skill1}, \bigcirc \text{Skill2})$$
$$\lambda_{1,-2} = \Pr(\bigtriangledown \text{Task1} - \text{obs} | \bigcirc \text{Skill1}, \neg\bigcirc \text{Skill2})$$
$$\lambda_{-1,-2} = \Pr(\bigtriangledown \text{Task1} - \text{obs} | \neg\bigcirc \text{Skill1}, \neg\bigcirc \text{Skill2})$$



Speigelhalter And Lauritzen (1990)

- *Global Parameter Independence* – parameters of laws for different CPTs are independent
- *Local Parameter Independence* – parameters for laws for different rows of CPTs are independent

Under these two assumptions, the natural conjugate law of a Bayesian network is a *hyper-Dirichlet law*, a law where the probabilities on each row of each CPT follow a Dirichlet law.

Abusing the definition, we say that a CPT for which each rows is given an independent Dirichlet law follows a *hyper-Dirichlet distribution* (really should be law).

A closer look at the binomial distribution

- **Binomial.** For counts of successes in binary trials, each with probability p, in n independent trials. E.g., n coin flips, with p the common probability of heads.

The variable

Count of successes

$$p(r|\pi,n) \propto \pi^{\boxed{r}}\underbrace{(1-\pi)}^{\boxed{n-r}}$$

Count of failures

The "success probability" parameter

The failure probability

The success probability

We will be using this as a likelihood in an example of the use of conjugate distributior

A closer look at the Beta distribution

- **Beta.** Defined on [0,1]. Conjugate prior for the probability parameter in Bernoulli & binomial models.

```
p ~ dbeta(a,b)
```

Mean(p): $\dfrac{a}{a+b}$

Variance(p): $\dfrac{ab}{(a+b)^2(a+b+1)}$

Mode(p): $\dfrac{a-1}{a+b-2}$

PseudoCount of successes

PseudoCount of failures

The variable: "success probability"

$$beta(\pi|\underbrace{a,b}) \propto \pi^{\boxed{a-1}}\underbrace{(1-\pi)}^{\boxed{b-1}}$$

The failure probability

Shape, or "prior sample info"

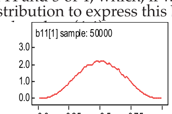
The success probability

An example with a continuous variable:

A beta-binomial example--the Prior Distribution

- The prior distribution:

  Let's suppose we think it is more likely that the coin is close to fair, so π is probably nearer to .5 than it is to either 0 or 1. We don't have any reason to think it is biased toward either heads or tails, so we'll want a prior distribution that is symmetric around .5. We're not real sure about what π might be--say about as sure as only 6 observations. This corresponds to 3 pseudo-counts of H and 3 of T, which, if we want to use a beta distribution to express this belief, corres



An example with a continuous variable:
A beta-binomial example--the Prior Distribution

- **Beta.** Defined on [0,1]. Conjugate prior for the probability parameter in Bernoulli & binomial models.

```
π ~ dbeta(4,4)
```

Mean(π): $\frac{4}{4+4} = .5$

Variance(π): $\frac{4 \cdot 4}{(4+4)^2(4+4+1)} = .028$

Mode(π): $\frac{4-1}{4+4-2} = .5$

PseudoCount of successes

PseudoCount of failures

The variable: "success probability"

$$p(\pi|\underbrace{4,4}_{}) \propto \pi^{\boxed{4-1}} \underbrace{(1-\pi)}_{}^{\boxed{4-1}}$$

The failure probability

Shape, or "prior sample info"

The success probability

An example with a continuous variable:
A beta-binomial example--the Likelihood

- The likelihood:

  Next we will flip the coin ten times. Assuming the same true (but unknown to us) value of π is in effect for each of ten independent trials, we can use the binomial distribution to model the probability of getting any number of heads: i.e.,

The variable

Count of *observed* successes

$$p(r|\pi,10) \propto \pi^{\boxed{r}} \underbrace{(1-\pi)}_{}^{\boxed{10-r}}$$

Count of *observed* failures

The "success probability" parameter

The failure probability

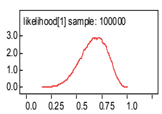
The success probability

An example with a continuous variable:
A beta-binomial example--the Likelihood

- The likelihood:

  We flip the coin ten times, and observe 7 heads; i.e., r=7. The likelihood is obtained now using the same form as in the preceding slide, except now r is fixed at 7 and we are interested in the relative value of this function at different possible values of π:

$$p(7|\pi,10) \propto \pi^7 (1-\pi)^3$$



An example with a continuous variable:
Obtaining the posterior by Bayes Theorem

General form:

$$\underset{\text{posterior}}{p(y|x^*)} \propto \underset{\text{likelihood}}{p(x^*|y)} \quad \underset{\text{prior}}{p(y)}$$

In our example, 7 plays the role of x*, and p plays the role of y. Before normalizing:

$$p(\pi|r=7) \propto \left[\pi^7 (1-\pi)^3\right]\left[\pi^{4-1}(1-\pi)^{4-1}\right]$$

$$= \left[\pi^{10}(1-\pi)^6\right]$$

This function is proportional to a beta(11,7) distribution.

$$= \left[\pi^{11-1}(1-\pi)^{7-1}\right]$$

An example with a continuous variable:
Obtaining the posterior by Bayes Theorem

posterior

$$p(y|x^*) = \frac{p(x^*|y)p(y)}{\int_y p(x^*|y)p(y)\partial y}$$

After normalizing:

$$p(\pi|r=7) = \frac{\pi^{11-1}(1-\pi)^{7-1}}{\int_z \left[z^{11-1}(1-z)^{7-1}\right]\partial z}$$

Now, how can we get an idea of what this means we believe about π after combining our prior belief and our observations?

An example with a continuous variable: In pictures

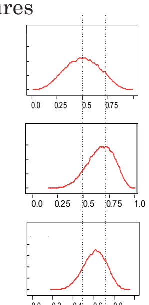
Prior

x

Likelihood

∝

Posterior



Dirichlet—Categorical conjugate distribution

- Assume a variable X takes on category $1,\ldots,K$ with probabilities $\pi_1,\ldots\pi_K$
- Take N draws from this distribution and observe counts $N=X_1+\ldots+X_K$
- Likelihood is $p(X_1,,\ldots,X_K) \propto \pi_1^{X_1}\cdots\pi_K^{X_K}$
- Dirichlet Prior: $f(\pi_1,,\ldots,\pi_K) \propto \pi_1^{\alpha_1-1}\cdots\pi_K^{\alpha_K-1}$
- Posterior:

$$f(\pi_1,,\ldots,\pi_K|X_1,\ldots,X_K) \propto \pi_1^{X_1+\alpha_1-1}\cdots\pi_K^{X_K+\alpha_K-1}$$

Updating an unconditional probability table (no parent variables)

- Prior is a table of alphas:

α_1	…	α_K

- Sum of alphas is pseudo-sample size for prior: Netica calls this Node Experience $A=\alpha_1+\cdots+\alpha_K$
- Sufficient statistic is a table of counts in each category

X_1	…	X_K

- Posterior is an updated table

α_1+X_1	…	α_K+X_K

- With updated Node Experience A'=A+N

Details

- Equivalent to beta-binomial when variable only takes two values
- Alphas must be positive, but don't need to be integers
- Alpha = ½ is non-informative prior
- A (sum of alphas) acts like a pseudo-sample size for the prior
- Can also write as $\alpha_k = A\pi_k^*$

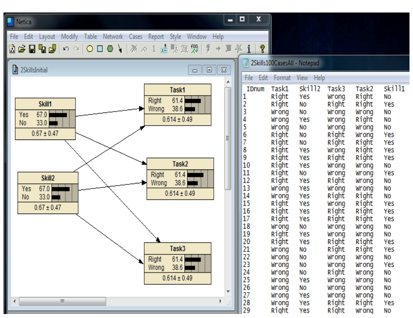
CPT updating when parents are fully observed

- Data are contingency table of child variable given parents
- Prior is a table of pseudo-counts
- Get posterior by adding them together

$$\begin{pmatrix}\alpha_{11} & \cdots & \alpha_{1K}\\ \vdots & \ddots & \vdots\\ \alpha_{J1} & \cdots & \alpha_{JK}\end{pmatrix}+\begin{pmatrix}X_{11} & \cdots & X_{1K}\\ \vdots & \ddots & \vdots\\ X_{J1} & \cdots & X_{JK}\end{pmatrix}=\begin{pmatrix}\alpha_{11}+X_{11} & \cdots & \alpha_{1K}+X_{1K}\\ \vdots & \ddots & \vdots\\ \alpha_{J1}+X_{J1} & \cdots & \alpha_{JK}+X_{JK}\end{pmatrix}$$

Note: Both prior and posterior effective sample size (Node Experience) can be different for each row.

Netica example – fully observed



RNetica example (Ex 8.3)

- File Hyperdirchlet
- Set up network
- Two parents, one child

```
sess <- NeticaSession()
startSession(sess)
hdnet <- CreateNetwork("hyperDirchlet",sess)
skills <- NewDiscreteNode(hdnet,c("Skill1","Skill2"),
      c("High","Medium","Low"))
obs <-
NewDiscreteNode(hdnet,"Observable",c("Right","Wrong"))
NodeParents(obs) <- skills
```

Set up prior for Observation

- Do this by setting CPT and NodeExperience (row pseudo-sample sizes)

```
ptab <- data.frame(
   Skill1=rep(c("High","Medium","Low"),3),
   Skill2=rep(c("High","Medium","Low"), each=3),
  Right=c(.975,.875,.5,.875,.5,.125,.5,.125,.025),
Wrong=1-c(.975,.875,.5,.875,.5,.125,.5,.125,.025))

obs[] <- ptab
NodeExperience(obs) <- 10 #All rows equally weighted
```


Aside: Using CPTtools

- The function calcDPCframe will (among other things) calculate tables according to the DiBello—Samejima models described in the morning session.

```
## Using CPTtools
ptab1 <- calcDPCFrame(
   ParentStates(obs),
   NodeStates(obs),
   log(c(Skill1=1.2,Skill2=.8)),0,
   rules="Compensatory")
```

- Note uses log of discrimination as parameter

Prior CPT

ptab

	Skill1	Skill2	Right	Wrong
1	High	High	0.975	0.025
2	Medium	High	0.875	0.125
3	Low	High	0.500	0.500
4	High	Medium	0.875	0.125
5	Medium	Medium	0.500	0.500
6	Low	Medium	0.125	0.875
7	High	Low	0.500	0.500
8	Medium	Low	0.125	0.875
9	Low	Low	0.025	0.975

rescaleTable(ptab,10)

	Skill1	Skill2	Right	Wrong
1	High	High	9.75	0.25
2	Medium	High	8.75	1.25
3	Low	High	5.00	5.00
4	High	Medium	8.75	1.25
5	Medium	Medium	5.00	5.00
6	Low	Medium	1.25	8.75
7	High	Low	5.00	5.00
8	Medium	Low	1.25	8.75
9	Low	Low	0.25	9.75

Netica Case files

- Text file, column separated by tabs (same as .xls files, but have .cas extension)
- One column for each observed variable (need both parents and child in this case)
- Optional IDnum column
- Optional NumCases column gives replication count
- So can either repeat out cases, or use summary counts.
- write.CaseFile() writes out a case file for use with Netica

Case Table for Ex 8.3

```
dtab <- data.frame(Skill1=rep(c("High","Medium","Low"),3,each=2),
                   Skill2=rep(c("High","Medium","Low"),each=6),
                   Observable=rep(c("Right","Wrong"),9),
                   NumCases=c(293,3,
                     112,16,
                     0,1,
                     14,1,
                     92,55,
                     4,5,
                     5,1,
                     62,156,
                     8,172))

write.CaseFile(dtab,"Ex8.3.cas")
```


Example Case File

	Skill1	Skill2	Observable	NumCases
1	High	High	Right	293
2	High	High	Wrong	3
3	Medium	High	Right	112
4	Medium	High	Wrong	16
5	Low	High	Right	0
6	Low	High	Wrong	1
7	High	Medium	Right	14
8	High	Medium	Wrong	1
9	Medium	Medium	Right	92
10	Medium	Medium	Wrong	55
11	Low	Medium	Right	4
12	Low	Medium	Wrong	5
13	High	Low	Right	5
14	High	Low	Wrong	1
15	Medium	Low	Right	62
16	Medium	Low	Wrong	156
17	Low	Low	Right	8
18	Low	Low	Wrong	172

Learn CPTs

- LearnCases does complete data hyper-Dirichlet updating

```
LearnCases("Ex8.3.cas",obs)
NodeExperience(obs)
                Skill2
Skill1   High Medium Low
  High    306     25  16
  Medium  138    157 228
  Low      11     19 190
```


Prior and Posterior CPTs

Prior

	Skill1	Skill2	Right	Wrong
1	High	High	0.975	0.025
2	Medium	High	0.875	0.125
3	Low	High	0.500	0.500
4	High	Medium	0.875	0.125
5	Medium	Medium	0.500	0.500
6	Low	Medium	0.125	0.875
7	High	Low	0.500	0.500
8	Medium	Low	0.125	0.875
9	Low	Low	0.025	0.975

Posterior

	Skill1	Skill2	Right	Wrong
1	High	High	0.989	0.011
2	Medium	High	0.848	0.152
3	Low	High	0.795	0.205
4	High	Medium	0.760	0.240
5	Medium	Medium	0.588	0.412
6	Low	Medium	0.276	0.724
7	High	Low	0.859	0.141
8	Medium	Low	0.277	0.723
9	Low	Low	0.068	0.932

Prior and Posterior Alphas

Prior

	Skill1	Skill2	Right	Wrong
1	High	High	9.75	0.25
2	Medium	High	8.75	1.25
3	Low	High	5.00	5.00
4	High	Medium	8.75	1.25
5	Medium	Medium	5.00	5.00
6	Low	Medium	1.25	8.75
7	High	Low	5.00	5.00
8	Medium	Low	1.25	8.75
9	Low	Low	0.25	9.75

Posterior

	Skill1	Skill2	Right	Wrong
1	High	High	302.75	3.25
2	Medium	High	117.00	21.00
3	Low	High	8.75	2.25
4	High	Medium	19.00	6.00
5	Medium	Medium	92.25	64.75
6	Low	Medium	5.25	13.75
7	High	Low	13.75	2.25
8	Medium	Low	63.25	164.75
9	Low	Low	13.00	177.00

Problems with hyper-Dirichlet approach

- Learn more about some rows than others
- Local parameter independence assumption is unrealistic – often want CPT to be monotonic (increasing skill means increasing chance of success)
  - $\lambda_{2,2} > \lambda_{2,1} > \lambda_{1,1}$ and $\lambda_{2,2} > \lambda_{1,2} > \lambda_{1,1}$
- Solution is to use parametric models for CPT:
  - Noisy-min & Noisy-max
  - DiBello-Samejima families
  - Discrete Partial Credit families

Learning CPTs for a parametric family

- Contingency table is sufficient statistic for law for any CPT!
- Pick value of law parameters that maximize the posterior probability (or likelihood) of the observed contingency table.
- Fully Bayesian method
  - Put hyper-laws over law hyperparameters
  - Calculate observed contingency table
  - MAP estimates maximize posterior probability of contingency table
- Semi-Bayesian method
  - Use prior hyperparameters to calculate prior table.
  - Establish a pseudo-sample size for each row and calculate prior alphas
  - Do hyper-Dirichlet updating to get posterior alphas
  - MAP estimates maximize posterior probability of posterior alphas (treating them as if they were data)
  - CPTtools function `mapCPT` does this

Latent and Missing Values

- These are okay as long as they are *missing at random*
- MAR means missingness indicator is conditionally independent of the value of the missing variable given the fully observed variables
- Latent variables are always MCAR
- With other missing variables, it depends on the study design
- Can use the EM or MCMC algorithms in the presence of MAR data

EM Algorithm (Dempster, Laird & Rubin, 1977)

Key idea:

1. Pick a set of value for parameters
2. *E-step (a):* Calculate distribution for missing variables given observed variables & current parameter values.
3. *E-step (b):* Calculate expected value of sufficient statistics
4. *M-step:* Use Gradient Decent to produce MAP/MLE estimates for parameters given sufficient statistics
5. Loop 2—4 until convergence

EM algorithm details

- Only need to take a few steps of the gradient algorithm in Step 4 (Generalized EM)
- Can exploit conditional independence conditions, particularly global parameter independence (Structual EM, Meng and van Dyke)
  - Once CPT at a time
- Can be slow
  - But not as slow as MCMC
- Netica provides built-in support for special case of hyper-Dirichlet law

Expected value of missing (latent) node

- Can calculate this using ordinary Netica operations (instantiate all observed variables and read off joint beliefs)
- Instead of adding count to the table, add fractional count to the table
- Similarly use joint beliefs when more than one parent is missing

Example

- Observable X in $\{0, 1\}$; Latent θ in $\{H,M,L\}$
- Observations:
  1. $X=1; p(\theta) = H{:}.33, M{:}.33, L{:}.33$
  2. $X=1; p(\theta) = H{:}.5, M{:}.33, L{:}.2$
  3. $X=0; p(\theta) = H{:}.2, M{:}.3, L{:}.5$
- Expected table:

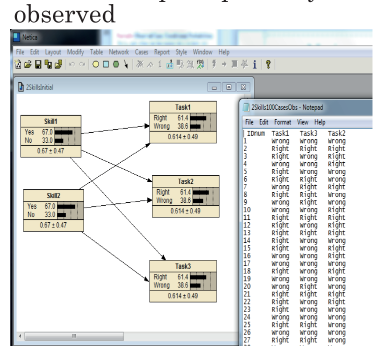
	H	M	L
1	.83	.67	.53
0	.2	.3	.5

EM for hyper-Dirichlet (RNetica LearnCPTs function)

1. Use current CPTs to calculate expected tables for all of the CPTs we are learning
2. Use the hyper-Dirichlet conjugate updating to update the CPTs
3. Loop 1 and 2 until convergence

Note: RNetica LearnCPT function currently does not reveal whether or not convergence was reached.

Netica example – partially observed



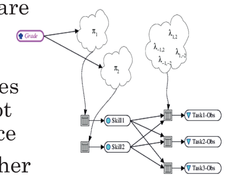
Parameterized tables

1. Use current parameters to set initial CPTs
2. Use Netica's LearnCPTs to calculate posterior tables
3. Multiple posterior tables by node experience to get pseudo-table for each CPT
4. Use gradient decent to optimize CPT parameters
5. Loop 1—4 until convergence

I'm currently working on an implementation in R (Peanut package function `GEMfit`; available from RNetica site).

Breakdown of global parameter independence

- Even if parameters are *a priori* independent, when there is missing (or latent) data then parameters are not independent *a posteriori*.
- EM algorithm only gives point estimate, does not capture this dependence
- There might also be other information which makes parameters dependent.



Markov Chain Monte Carlo (MCMC)

- In place of E-step, randomly sample values for unknown (latent & missing) variables
- In place of M-step, randomly sample values for parameters
- Takes longer than EM, but gives you an impression of the whole distribution rather than just a part.

Bayesian Networks in Educational Assessment

Estimating Parameters with MCMC

Roy Levy
Arizona State University
Roy.Levy@asu.edu

Bayesian Inference: Expanding Our Context

Posterior Distribution

Posterior distribution for ***unknowns*** given ***knowns*** is

$$p(unknowns \mid knowns) \propto p(knowns \mid unknowns)\, p(unknowns)$$

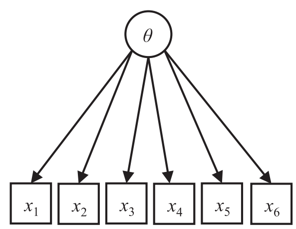
Inference about examinee latent variables (θ) given observables ($\mathbf{x}$)

$$p(\theta \mid \mathbf{x}) \propto p(\mathbf{x} \mid \theta)\, p(\theta)$$

Example: ACED Bayes Net Fragment for *Common Ratio*

- θ = *Common Ratio*
- $\mathbf{x}$ = Observables from tasks that measure *Common Ratio*

Bayes Net Fragment



θ = *Common Ratio*

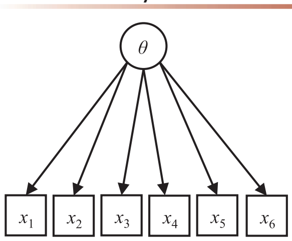
$p(\theta)$

xs = Observables from tasks that measure *Common Ratio*

$$p(\theta \mid x) \propto p(x \mid \theta)\, p(\theta)$$

$$p(\mathbf{x} \mid \theta) = \prod_{j=1}^{J} p(x_j \mid \theta)$$

Probability Distribution for the Latent Variable



θ = *Common Ratio*

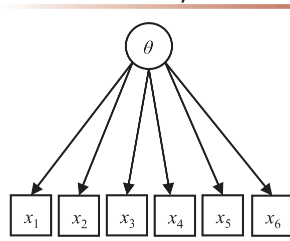
$\theta \sim \text{Categorical}(\boldsymbol{\lambda})$

ACED Example

- 2 Levels of θ (Low, High)
- $\boldsymbol{\lambda} = (\lambda_1, \lambda_2)$ contains probabilities for Low and High

	θ (Common Ratio)	
	1	2
Prob.	λ_1	λ_2

Probability Distribution for the Observables



xs = Observables from tasks that measure *Common Ratio*

$(x_j \mid \theta = c) \sim \text{Bernoulli}(\pi_{cj})$

ACED Example

- π_{cj} is the probability of correct response on task j given $\theta = c$

	$p(x_j \mid \theta)$	
θ	0	1
1	$1 - \pi_{1j}$	π_{1j}
2	$1 - \pi_{2j}$	π_{2j}

Bayesian Inference

$$p(\theta \mid \mathbf{x}) \propto p(\mathbf{x} \mid \theta)\, p(\theta)$$

	θ (Common Ratio)	
	1	2
Prob.	λ_1	λ_2

If the λs and πs are unknown, they become subject to posterior inference too

	$p(x_j \mid \theta)$	
θ	0	1
1	$1 - \pi_{1j}$	π_{1j}
2	$1 - \pi_{2j}$	π_{2j}

Bayesian Inference

$$p(\theta \mid \mathbf{x}) \propto p(\mathbf{x} \mid \theta)\, p(\theta)$$

	$p(x_j \mid \theta)$	
θ	0	1
1	$1 - \pi_{1j}$	π_{1j}
2	$1 - \pi_{2j}$	π_{2j}

A convenient choice for prior distribution is the beta distribution

$$\pi_{cj} \sim \text{Beta}(\alpha_{\pi_c}, \beta_{\pi_c})$$

ACED Example: $\pi_{1j} \sim \text{Beta}(1, 1)$ $\quad \pi_{2j} \sim \text{Beta}(1, 1)$

For first task, constrain $(\pi_{21} > \pi_{11})$ to resolve indeterminacy in the latent variable and avoid label switching

Bayesian Inference

$$p(\theta \mid \mathbf{x}) \propto p(\mathbf{x} \mid \theta)\, p(\theta)$$

	θ (Common Ratio)	
	1 (Low)	2 (High)
Prob.	λ_1	λ_2

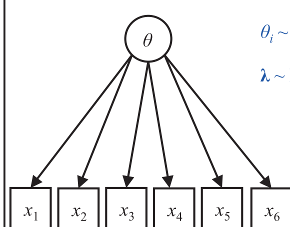
A convenient choice for the prior distribution is the Dirichlet distribution

$$\boldsymbol{\lambda} \sim \text{Dirichlet}(\boldsymbol{\alpha}_\lambda) \qquad \boldsymbol{\alpha}_\lambda = (\alpha_{\lambda_1}, \alpha_{\lambda_2})$$

which generalizes the Beta distribution to the case of multiple categories

ACED Example: $\boldsymbol{\lambda} = (\lambda_1, \lambda_2) \sim \text{Dirichlet}(1, 1)$

Model Summary



$\theta_i \sim \text{Categorical}(\boldsymbol{\lambda})$

$\boldsymbol{\lambda} \sim \text{Dirichlet}(1, 1)$

$(x_{ij} \mid \theta_i = c) \sim \text{Bernoulli}(\pi_{cj})$

$\pi_{11} \sim \text{Beta}(1, 1)$

$\pi_{21} \sim \text{Beta}(1, 1)\ \ I(\pi_{21} > \pi_{11})$

$\pi_{cj} \sim \text{Beta}(1, 1)$ for others obs.

JAGS Code

JAGS Code

```
for (i in 1:n){
  for(j in 1:J){
    x[i,j] ~ dbern(pi[theta[i],j])
  }
}
```

$(x_{ij} \mid \theta_i = c) \sim \text{Bernoulli}(\pi_{cj})$

Referencing the table for π_js in terms of $\theta = 1$ or 2

	$p(x_j \mid \theta)$	
θ	0	1
1	$1 - \pi_{1j}$	π_{1j}
2	$1 - \pi_{2j}$	π_{2j}

JAGS Code

```
pi[1,1] ~ dbeta(1,1)
pi[2,1] ~ dbeta(1,1) T(pi[1,1], )

for(c in 1:C){
  for(j in 2:J){
    pi[c,j] ~ dbeta(1,1)
  }
}
```

$\pi_{11} \sim \text{Beta}(1, 1)$

$\pi_{21} \sim \text{Beta}(1, 1)\ \ I(\pi_{21} > \pi_{11})$

$\pi_{cj} \sim \text{Beta}(1, 1)$ for remaining observables

MCMC 13

JAGS Code

```
for (i in 1:n){
  theta[i] ~ dcat(lambda[])
}

lambda[1:C] ~ ddirch(alpha_lambda[])
for(c in 1:C){
  alpha_lambda[c] <- 1
}
```

$\theta_i \sim \text{Categorical}(\boldsymbol{\lambda})$

$\boldsymbol{\lambda} \sim \text{Dirichlet}(1, 1)$

MCMC 14

Markov Chain Monte Carlo

MCMC 15

Estimation in Bayesian Modeling

- Our "answer" is a posterior distribution
  - All parameters treated as random, not fixed
- Contrasts with frequentist approaches to inference, estimation
  - Parameters are fixed, so estimation comes to finding the single best value
  - "Best" here in terms of a criterion (ML, LS, etc.)
- Peak of a mountain vs. mapping the entire terrain of peaks, valleys, and plateaus (of a landscape)

MCMC 16

What's In a Name?

Markov chain ***Monte Carlo***

- Construct a sampling algorithm to ***simulate*** or ***draw from*** the posterior.
- Collect many such draws, which serve to empirically approximate the posterior distribution, and can be used to empirical approximate summary statistics.

Monte Carlo Principle:

> Anything we want to know about a random variable θ can be learned by sampling many times from $f(\theta)$, the density of θ.
>
> -- Jackman (2009)

MCMC 17

What's In a Name?

Markov ***chain*** Monte Carlo

- Values really generated as a sequence or chain
- t denotes the step in the chain
- $\theta^{(0)}, \theta^{(1)}, \theta^{(2)},\ldots, \theta^{(t)},\ldots, \theta^{(T)}$
- Also thought of as a time indicator

Markov chain Monte Carlo

- Follows the Markov property…

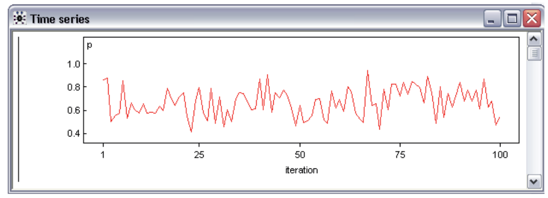
MCMC 18

The Markov Property

- Current state depends on previous position
  - Examples: weather, checkers, baseball counts & scoring
- Next state conditionally independent of past, given the present
  - Akin to a full mediation model
- $p(\theta^{(t+1)} \mid \theta^{(t)}, \theta^{(t-1)}, \theta^{(t-2)}, \ldots) = p(\theta^{(t+1)} \mid \theta^{(t)})$

$$\boxed{\theta^{(0)}} \rightarrow \boxed{\theta^{(1)}} \rightarrow \boxed{\theta^{(2)}} \rightarrow \boxed{\theta^{(3)}}$$

Visualizing the Chain: Trace Plot



Markov Chain Monte Carlo

- Markov chains are ***sequences of numbers*** that have the Markov property
  - Draws in cycle t+1 depend on values from cycle t, but given those not on previous cycles (Markov property)
- Under certain assumptions Markov chains reach ***stationarity***
- The collection of values converges to a distribution, referred to as a stationary distribution
  - Memoryless: It will "forget" where it starts
  - Start anywhere, will reach stationarity if regularity conditions hold
  - For Bayes, set it up so that this is the posterior distribution
- Upon convergence, samples from the chain approximate the stationary (posterior) distribution

Assessing Convergence

Diagnosing Convergence

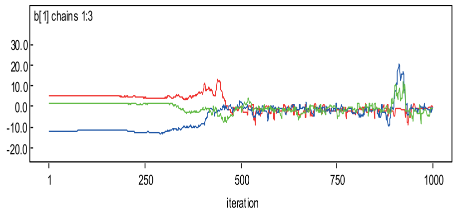
- With MCMC, convergence to a ***distribution***, not a point
- ML:
  - Convergence is when we've reached the highest point in the likelihood,
  - The highest peak of the mountain
- MCMC:
  - Convergence when we're sampling values from the correct distribution,
  - We are mapping the entire terrain accurately

Diagnosing Convergence

- A properly constructed Markov chain is guaranteed to converge to the stationary (posterior) distribution…eventually
- Upon convergence, it will sample over the full support of the stationary (posterior) distribution…over an ∞ number of draws
- In a finite chain, no guarantee that the chain has converged or is sampling through the full support of the stationary (posterior) distribution
- Many ways to diagnose convergence
- Whole software packages dedicated to just assessing convergence of chains (e.g., R packages 'coda' and 'boa')

Gelman & Rubin's (1992) Potential Scale Reduction Factor (PSRF)

- Run ***multiple*** chains from dispersed starting points
- Suggest convergence when the chains come together
- If they all go to the same place, it's probably the stationary distribution

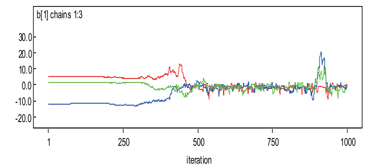


Gelman & Rubin's (1992) Potential Scale Reduction Factor (PSRF)

- An analysis of variance type argument
- *PSRF* or R =

$$\frac{\text{Total Variance}}{\text{Within Chain Variance}} = \frac{\text{Between Chain Variance} + \text{Within Chain Variance}}{\text{Within Chain Variance}}$$

- If there is substantial between-chain variance, will be >> 1



Gelman & Rubin's (1992) Potential Scale Reduction Factor (PSRF)

- Run ***multiple*** chains from dispersed starting points
- Suggest convergence when the chains come together
- Operationalized in terms of partitioning variability
- Run multiple chains for $2T$ iterations, discard first half
- Examine between and within chain variability
- Various versions, modifications suggested over time

Potential Scale Reduction Factor (PSRF)

- For any θ, for any chain c the within-chain variance is

$$W_c = \frac{1}{T-1}\sum_{t=1}^{T}(\theta_{(c)}^{(t)} - \bar{\theta}_{(c)})^2$$

- For all chains, the pooled within-chain variance is

$$W = \frac{1}{C}\sum_{c=1}^{C} W_c = \frac{1}{C(T-1)}\sum_{c=1}^{C}\sum_{t=1}^{T}(\theta_{(c)}^{(t)} - \bar{\theta}_{(c)})^2$$

Potential Scale Reduction Factor (PSRF)

- The between-chain variance is

$$B = \frac{T}{C-1}\sum_{c=1}^{C}(\bar{\theta}_{(c)} - \bar{\theta})^2$$

- The estimated variance is

$$\hat{V}ar(\theta) = (T - 1/T)W + (1/T)B$$

Potential Scale Reduction Factor (PSRF)

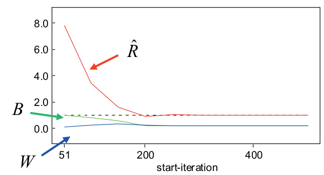
- The potential scale reduction factor is

$$\hat{R} = \sqrt{\frac{\hat{V}ar(\theta)}{W}}$$

- If close to 1 (e.g., < 1.1) for all parameters, can conclude convergence

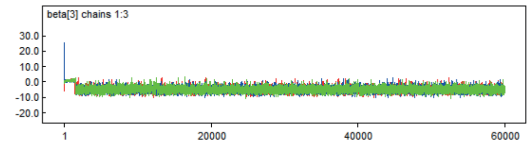
Potential Scale Reduction Factor (PSRF)

- Examine it over "time", look for $\hat{R} \rightarrow 1$, stability of B and W
- If close to 1 (e.g., < 1.2, or < 1.1) can conclude convergence



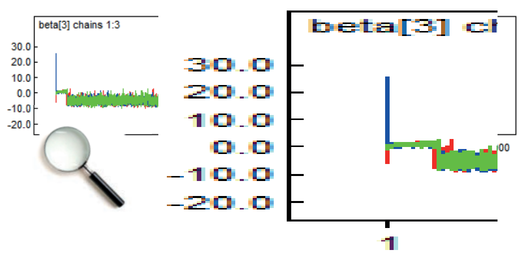
Assessing Convergence: No Guarantees

Multiple chains coming together does not guarantee they have converged



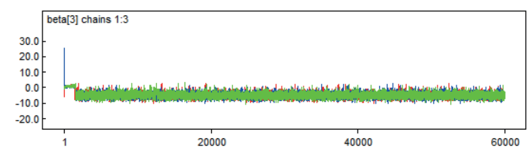
Assessing Convergence: No Guarantees

multiple chains come together does not guarantee they have converged



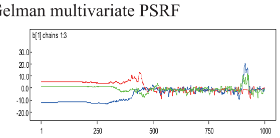
Assessing Convergence: No Guarantees

Multiple chains coming together does not guarantee they have converged



Assessing Convergence

- Recommend running multiple chains far apart and determine when they reach the same "place"
  - PSRF criterion an approximation to this
  - Akin to starting ML from different start values and seeing if they reach the same maximum
  - Here, convergence to a distribution, not a point
- A chain hasn't converged until ***all*** parameters converged
  - Brooks & Gelman multivariate PSRF



Serial Dependence

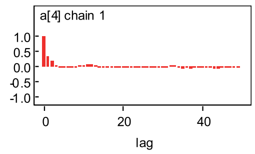
Serial Dependence

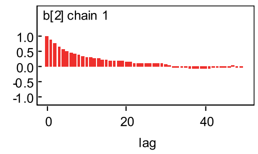
- Serial dependence between draws due to the dependent nature of the draws (i.e., the Markov structure)
- $p(\theta^{(t+1)} \mid \theta^{(t)}, \theta^{(t-1)}, \theta^{(t-2)}, \ldots) = p(\theta^{(t+1)} \mid \theta^{(t)})$

$$\boxed{\theta^{(0)}} \rightarrow \boxed{\theta^{(1)}} \rightarrow \boxed{\theta^{(2)}} \rightarrow \boxed{\theta^{(3)}}$$

- However there is a ***marginal*** dependence across multiple lags
- Can examine the autocorrelation across different lags

Autocorrelation



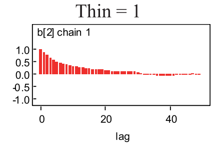


Thinning

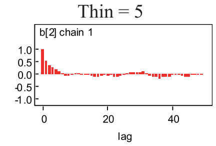
- Can "thin" the chain by dropping certain iterations
  Thin = 1 → keep every iteration
  Thin = 2 → keep every other iteration (1, 3, 5,…)
  Thin = 5 → keep every 5th iteration (1, 6, 11,…)
  Thin = 10 → keep every 10th iteration (1, 11, 21,…)
  Thin = 100 → keep every 100th iteration (1, 101, 201,…)

Thinning

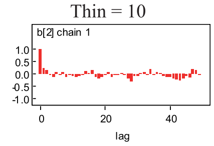
Thin = 1



Thin = 5



Thin = 10



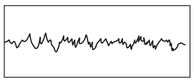
Thinning

- Can "thin" the chain by dropping certain iterations
  Thin = 1 → keep every iteration
  Thin = 2 → keep every other iteration (1, 3, 5,…)
  Thin = 5 → keep every 5th iteration (1, 6, 11,…)
  Thin = 10 → keep every 10th iteration (1, 11, 21,…)
  Thin = 100 → keep every 100th iteration (1, 101, 201,…)
- Thinning ***does not*** provide a better portrait of the posterior
  - A loss of information
- May want to keep, and account for time-series dependence
- Useful when data storage, other computations an issue
  - *I want 1000 iterations, rather have 1000 approximately independent iterations*
- Dependence ***within*** chains, but none ***between*** chains

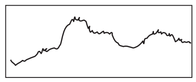
Mixing

Mixing

- We don’t want the sampler to get “stuck” in some region of the posterior , or ignore a certain area of the posterior
- Mixing refers to the chain “moving” throughout the support of the distribution in a reasonable way



relatively good mixing



relatively poor mixing

MCMC 43

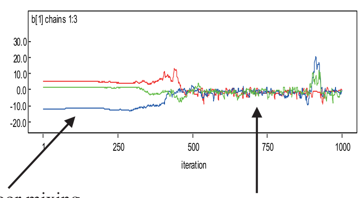
Mixing

- Mixing ≠ convergence, but better mixing usually leads to faster convergence
- Mixing ≠ autocorrelation, but better mixing usually goes with lower autocorrelation (and cross-correlations between parameters)
- With better mixing, then for a given number of MCMC iterations, get more information about the posterior
  - Ideal scenario is independent draws from the posterior
- With worse mixing, need more iterations to (a) achieve convergence and (b) achieve a desired level of precision for the summary statistics of the posterior

MCMC 44

Mixing

- Chains may mix differently at different times
- Often indicative of an adaptive MCMC algorithm



relatively poor mixing

relatively good mixing

MCMC 45

Mixing

- Slow mixing can also be caused by high dependence between parameters
  - Example: multicollinearity
- Reparameterizing the model can improve mixing
  - Example: centering predictors in regression

MCMC 46

Stopping the Chain(s)

MCMC 47

When to Stop The Chain(s)

- Discard the iterations prior to convergence as ***burn-in***
- How many more iterations to run?
  - As many as you want ☺
  - As many as time provides
- Autocorrelaion complicates things
- Software may provide the “MC error”
  - Estimate of the sampling variability of the sample mean
  - Sample here is the sample of iterations
  - Accounts for the dependence between iterations
  - Guideline is to go at least until MC error is less than 5% of the posterior standard deviation
- Effective sample size
  - Approximation of how many independent samples we have

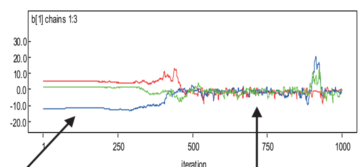
MCMC 48

Steps in MCMC in Practice

Steps in MCMC (1)

- Setup MCMC using any of a number of algorithms
  - Program yourself (have fun ☺)
  - Use existing software (BUGS, JAGS)
- Diagnose convergence
  - Monitor trace plots, PSRF criteria
- Discard iterations prior to convergence as ***burn-in***
  - Software may indicate a minimum number of iterations needed
  - A lower bound

Adapting MCMC → Automatic Discard



relatively poor mixing
during adaptive phase

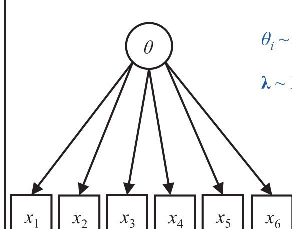
relatively good mixing
after adaptive phase

Steps in MCMC (2)

- Run the chain for a desired number of iterations
  - Understanding serial dependence/autocorrelation
  - Understanding mixing
- Summarize results
  - Monte Carlo principle
  - Densities
  - Summary statistics

ACED Example

Model Summary



$\theta_i \sim \text{Categorical}(\boldsymbol{\lambda})$

$\boldsymbol{\lambda} \sim \text{Dirichlet}(1, 1)$

$(x_{ij} \mid \theta_i = c) \sim \text{Bernoulli}(\pi_{cj})$

$\pi_{11} \sim \text{Beta}(1, 1)$

$\pi_{21} \sim \text{Beta}(1, 1)\;\; I(\pi_{21} > \pi_{11})$

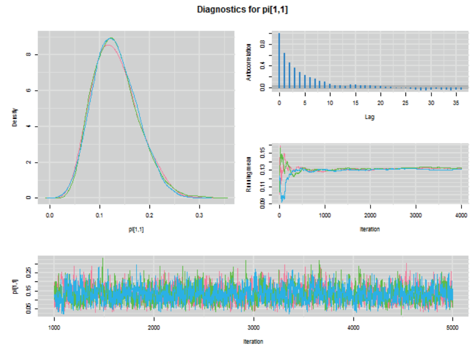
$\pi_{cj} \sim \text{Beta}(1, 1)$ for others obs.

ACED Example

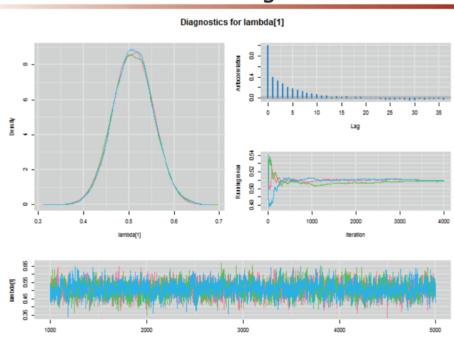
See 'ACED Analysis.R' for Running the analysis in R

See Following Slides for Select Results

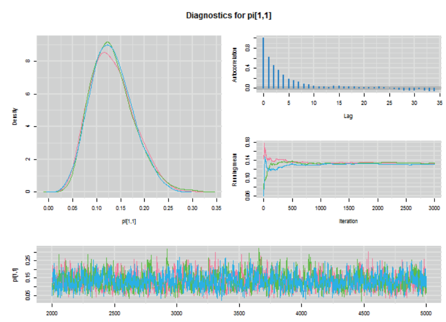
Convergence Assessment (1)



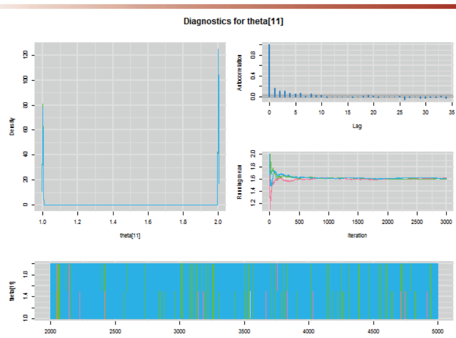
Convergence Assessment (2)



Posterior Summary (1)



Posterior Summary (2)



Posterior Summary (3)

	Mean	SD	Naive SE	Time-series SE	0.025	0.25	0.5	0.75	0.975	Median	95% HPD lower	95% HPD Upper
lambda[1]	0.51	0.04	0	0	0.42	0.48	0.51	0.54	0.6	0.51	0.43	0.6
lambda[2]	0.49	0.04	0	0	0.4	0.46	0.49	0.52	0.58	0.49	0.4	0.57
pi[1,1]	0.13	0.04	0	0	0.06	0.1	0.13	0.16	0.23	0.13	0.05	0.22
pi[2,1]	0.84	0.04	0	0	0.75	0.81	0.84	0.87	0.91	0.84	0.75	0.92
pi[1,2]	0.22	0.05	0	0	0.12	0.18	0.22	0.26	0.33	0.22	0.12	0.33
pi[2,2]	0.98	0.02	0	0	0.93	0.97	0.99	0.99	1	0.99	0.94	1
pi[1,3]	0.02	0.01	0	0	0	0.01	0.02	0.03	0.06	0.02	0	0.05
pi[2,3]	0.19	0.04	0	0	0.12	0.17	0.19	0.22	0.28	0.19	0.12	0.27
pi[1,4]	0.03	0.02	0	0	0.01	0.02	0.03	0.04	0.07	0.03	0	0.06
pi[2,4]	0.23	0.05	0	0	0.15	0.2	0.23	0.26	0.33	0.23	0.15	0.33
pi[1,5]	0.15	0.04	0	0	0.08	0.12	0.15	0.17	0.22	0.15	0.08	0.22
pi[2,5]	0.64	0.05	0	0	0.53	0.6	0.64	0.67	0.74	0.64	0.53	0.74
pi[1,6]	0.17	0.04	0	0	0.1	0.14	0.17	0.2	0.25	0.17	0.1	0.25
pi[2,6]	0.82	0.05	0	0	0.72	0.79	0.82	0.86	0.92	0.82	0.73	0.92
theta[1]	2	0.06	0	0	2	2	2	2	2	2	2	2
theta[2]	1	0.02	0	0	1	1	1	1	1	1	1	1
theta[3]	1	0.01	0	0	1	1	1	1	1	1	1	1
theta[4]	1.97	0.17	0	0	1	2	2	2	2	2	2	2
theta[5]	1.17	0.38	0	0.01	1	1	1	1	2	1	1	2
theta[6]	1	0.01	0	0	1	1	1	1	1	1	1	1
theta[7]	1.01	0.07	0	0	1	1	1	1	1	1	1	1

Summary and Conclusion

MCMC 61

Summary

- Dependence on initial values is "forgotten" after a sufficiently long run of the chain (memoryless)
- Convergence to a ***distribution***
  - Recommend monitoring multiple chains
  - PSRF as approximation
- Let the chain "burn-in"
  - Discard draws prior to convergence
  - Retain the remaining draws as draws from the posterior
- Dependence across draws induce autocorrelations
  - Can thin if desired
- Dependence across draws within and between parameters can slow mixing
  - Reparameterizing may help

MCMC 62

Wise Words of Caution

Beware: MCMC sampling can be dangerous!

-- Spiegelhalter, Thomas, Best, & Lunn (2007)
(WinBUGS User Manual)

MCMC 63

Bayesian Networks in Educational Assessment

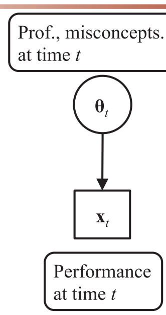
Dynamic Bayesian Networks

Roy Levy
Arizona State University
Roy.Levy@asu.edu

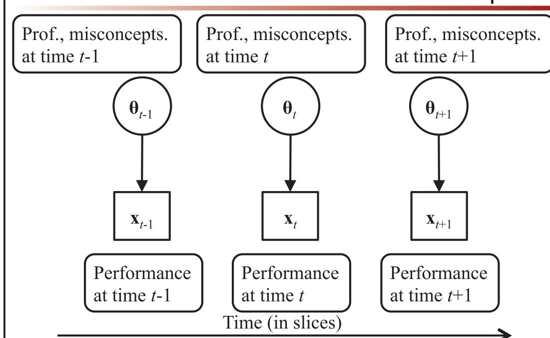
Dynamic Bayesian Networks (DBNs)

- ***Dynamic*** BNs (DBNs) for modeling longitudinal data
- Bayesian network where variables are repeated, usually over time or related index
- Assessment applications: monitor learning and growth
  - Proficiency and performance on first attempt, on second attempt, etc.
- Knowledge tracing, latent Markov models, latent transition models, growth models

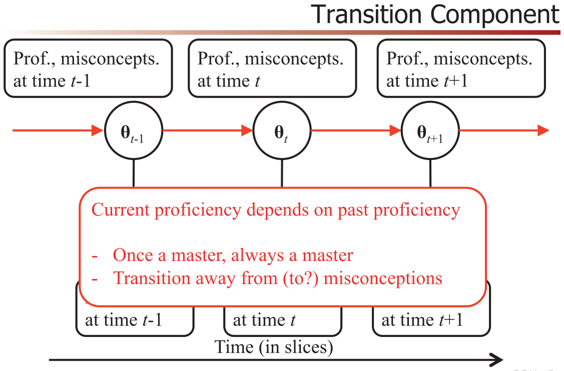
DBN Psychometric Models: Within-Time Component



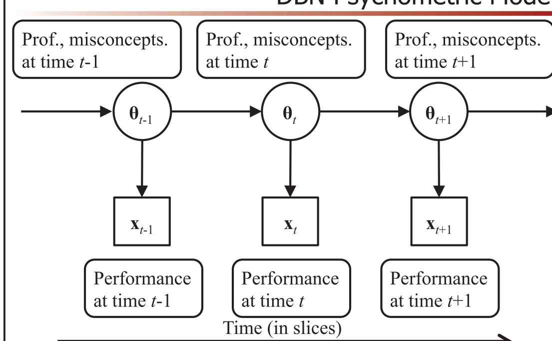
DBN Psychometric Models: Within-Time Component



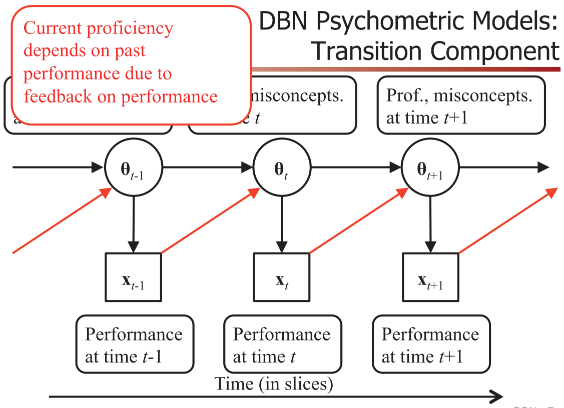
DBN Psychometric Models: Transition Component



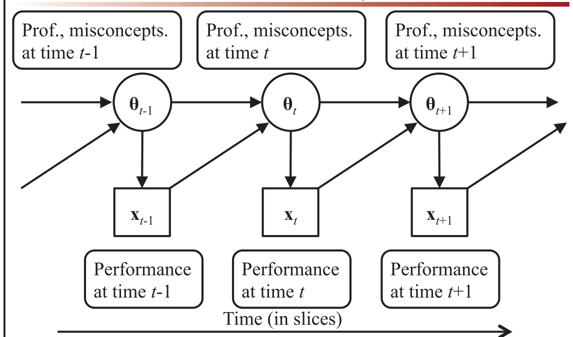
DBN Psychometric Models



DBN Psychometric Models: Transition Component



DBN Psychometric Models



Example

Example Context: Save Patch

- Educational video game targeting rational number equivalence
  - Adding whole numbers
  - Finding appropriate denominators, fractions < 1
  - Finding appropriate denominators, fractions > 1
  - Adding fractions given correct ingredients
  - Adding fractions greater than 1
- Student lays out ropes for character to navigate across to end
  - Success on a level leads to more complicated levels
  - Advanced levels involve converting ropes (fractions), more complicated layouts, and gaming features (picking up keys, coins)

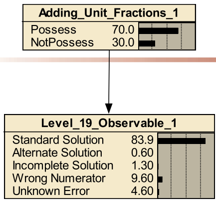
Example Context: Save Patch

- Complete a level, move on to the next level
- Don't complete a level, try again (and again, and again,…)
- Constructed as a learning tool
- Assesses proficiency of various skills (converting fractions, adding fractions, etc.) and
- Assesses various misconceptions/errors (inclusion, partitioning, etc.)
- Game-playing strategies relevant too (e.g., everything in order)
- Key departures from standard assessment paradigm
  - Feedback (student knows if correctly or incorrectly completed)
  - Learning during assessment (by design!)
  - Performances not conditionally independent (you know what you did, and how it turned out, for the most part)

Dynamic Bayesian Networks (DBNs)

- Characterization of performance
  - Standard solution
  - Alternate solution
  - Incomplete solution
  - Errors (many different kinds)
  - Skipped key
  - Wrong direction
  - Reset solution
- Example: performance on Level 19
  - Assuming the examinee does not have the misconception
  - 2-class latent variable for mastery of whole numbers
  - Probabilities estimated using MCMC, input to Netica
  - Analysis of first four types of performance, attempts resulting in others ignored

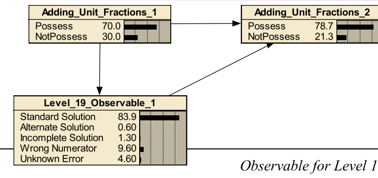
Conditional Probability of the Observable



p(*Observable for Level 19* | *Adding Unit Fractions*)

Adding Unit Fractions	Standard Solution	Alternate Solution	Incomplete Solution	Wrong Numerator Error	Unknown Error
Possess	0.95	0.00	0.01	0.03	0.01
Not Possess	0.58	0.02	0.01	0.25	0.13

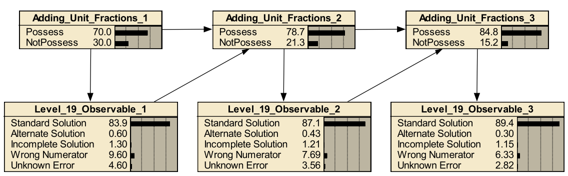
Transition Probability



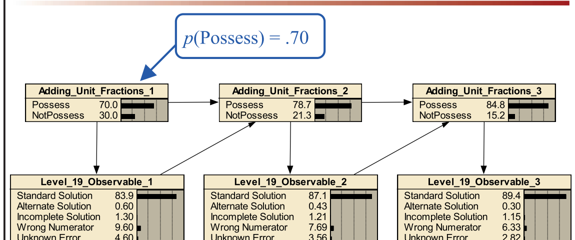
Observable for Level 19 at Time *t*

Adding Unit Fractions at Time *t*	Standard Solution	Alternate Solution	Incomplete Solution	Wrong Numerator Error	Unknown Error
Possess	1	1	1	1	1
Not Possess	.38	.17	.19	.20	.09

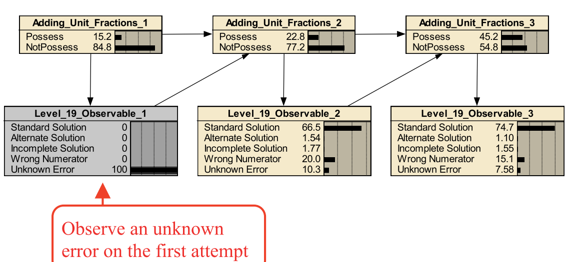
3 Time Slices



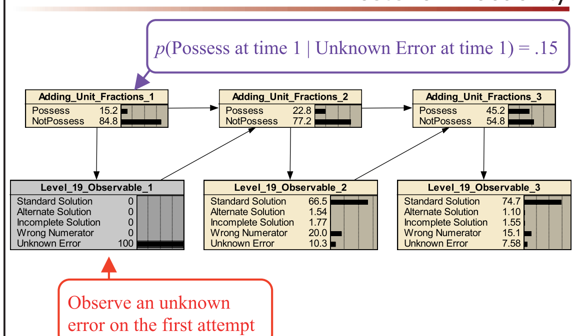
Prior



1st Observation

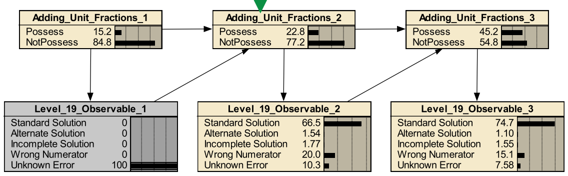


Posterior Probability



Posterior Predictive Probability

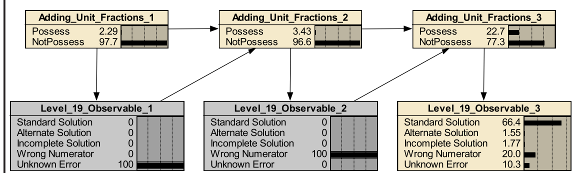
p(Possess at time 2 | Unknown Error at time 1) = .23



Observe an unknown error on the first attempt

DBN 19

2nd Observation

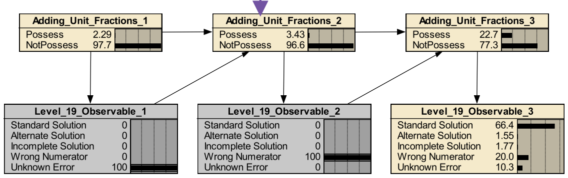


Observe a wrong numerator on the second attempt

DBN 20

Posterior Probability

p(Possess at time 2 | Unkn Err at t 1, Wrong Num Error at t 2) = .03

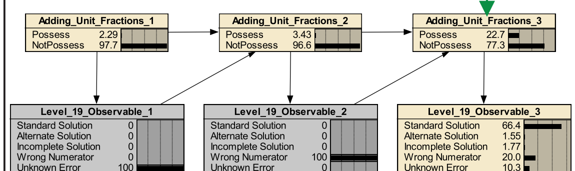


Observe a wrong numerator on the second attempt

DBN 21

Posterior Predictive Probability

p(Possess at time 3 | Unkn Err at t 1, Wrong Num Error at t 2) = .23

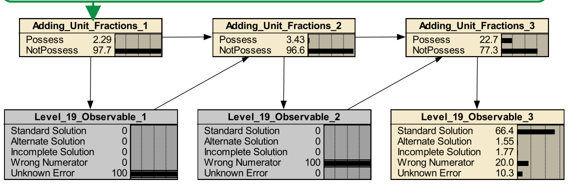


Observe a wrong numerator on the second attempt

DBN 22

Posterior Retrodictive Probability

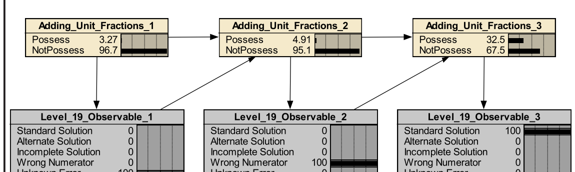
p(Possess at time 1 | Unkn Err at t 1, Wrong Num Error at t 2) = .02



Observe a wrong numerator on the second attempt

DBN 23

3rd Observation

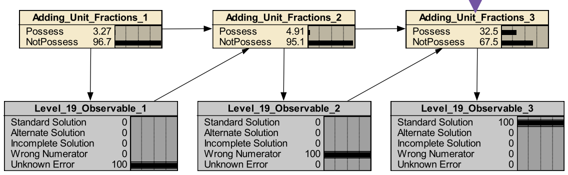


Observe a standard solution on the third attempt

DBN 24

Posterior Probability

p(Possess at time 3 | Unkn Err, Wrong Num Err, Stand Sol) = .33



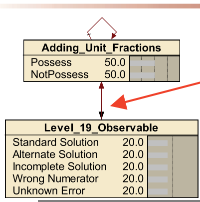
Observe a standard solution on the third attempt

Netica Files:

Save Patch Level 19 Simplified Dynamic.neta

Save Patch Level 19 Simplified Dynamic Expanded 3 Time Points.neta

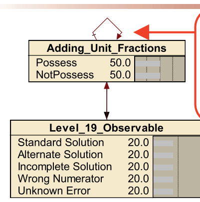
Save Patch Level 19 Simplified Dynamic.neta



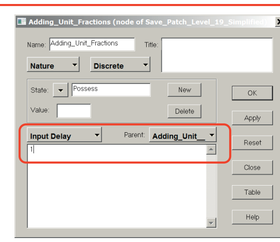
Link from Adding_Unit_Fractions to Level_19_Observable indicates observable at any time depends on proficiency at that time

Adding Unit Fractions	Standard Solution	Alternate Solution	Incomplete Solution	Wrong Numerator Error	Unknown Error
	p(Observable for Level 19 | Adding Unit Fractions)				
Possess	0.95	0.00	0.01	0.03	0.01
Not Possess	0.58	0.02	0.01	0.25	0.13

Save Patch Level 19 Simplified Dynamic.neta

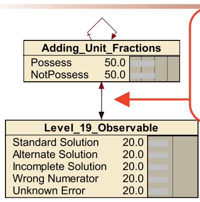


Link from Adding_Unit_Fractions to itself indicates that Adding_Unit_Fractions at any time depends on Adding_Unit_Fractions in the past

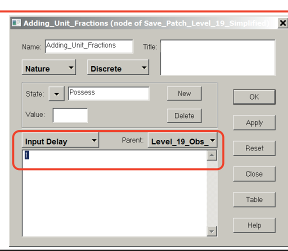


Input Delay = 1 means lag of 1

Save Patch Level 19 Simplified Dynamic.neta

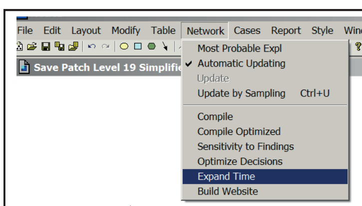


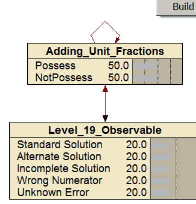
Link from Level_19_Observable to Adding_Unit_Fractions to indicates future proficiency at any time depends on current observable



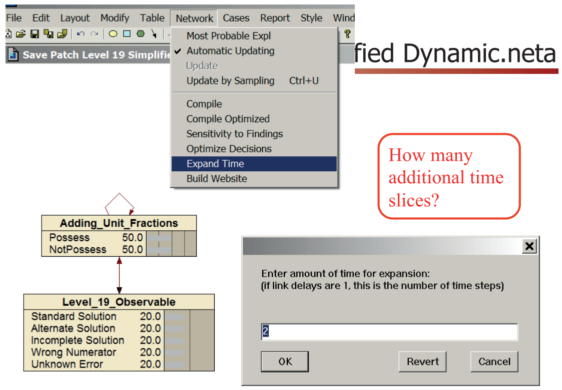
Input Delay = 1 means lag of 1

...fied Dynamic.neta



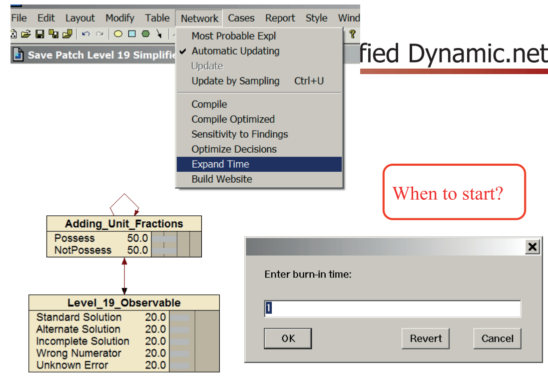


fied Dynamic.neta



How many additional time slices?

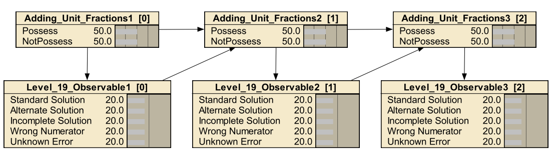
fied Dynamic.neta



When to start?

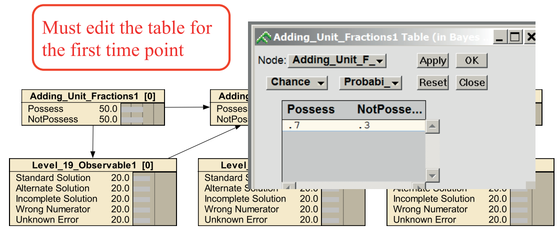
Save Patch Level 19 Simplified Dynamic.neta

Must edit the table for the first time point

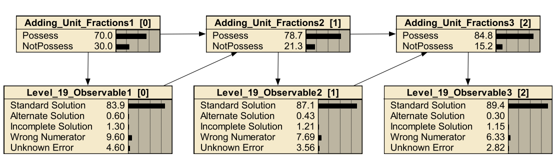


Save Patch Level 19 Simplified Dynamic.neta

Must edit the table for the first time point



Save Patch Level 19 Simplified Dynamic.neta



See
'Save Patch Level 19 Simplified Dynamic Expanded 3 Time Points.neta'