

Portland State University

PDXScholar

Systems Science Faculty Publications and
Presentations

Systems Science

7-29-2020

Reconstructability Analysis & Its Occam Implementation

Martin Zwick

Portland State University, zwick@pdx.edu

Follow this and additional works at: https://pdxscholar.library.pdx.edu/sysc_fac



Part of the [Logic and Foundations Commons](#), and the [Systems Architecture Commons](#)

Let us know how access to this document benefits you.

Citation Details

Zwick, Martin (2020). "Reconstructability Analysis and Its Occam Implementation." ICCS 2020, July 29.

This Presentation is brought to you for free and open access. It has been accepted for inclusion in Systems Science Faculty Publications and Presentations by an authorized administrator of PDXScholar. For more information, please contact pdxscholar@pdx.edu.

Reconstructability Analysis & Its Occam Implementation

Martin Zwick

Professor of Systems Science
Portland State University

zwick@pdx.edu

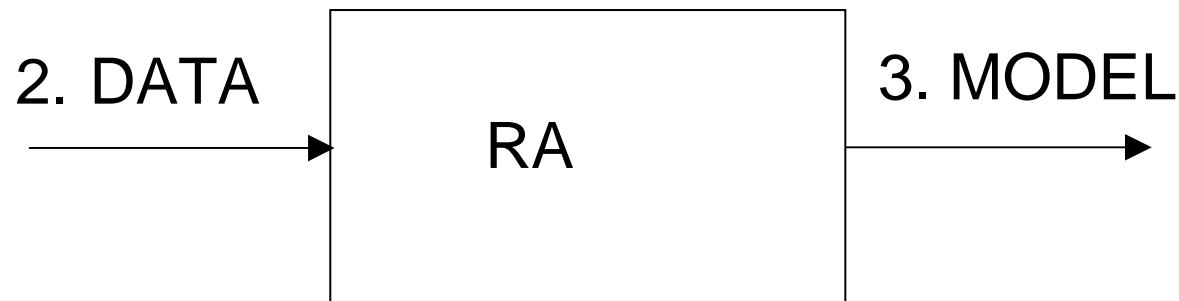
http://www.pdx.edu/sysc/research_dmm.html

ICCS 2020, July 29

1. Introduction: what is RA

2. Input data to RA

3. Output model from RA



INTRODUCTION: WHAT IS RA?

- **Reconstructability Analysis** (RA) = a probabilistic graphical modeling methodology
- RA = Information theory + Graph theory
- Graphs, applied to data, are **models**:
- node = variable; link = relationship
- RA uses not only graphs (a link joins 2 nodes), but hypergraphs (a link can join **>2** nodes)

WHY RA MIGHT BE OF INTEREST ^{1/2}

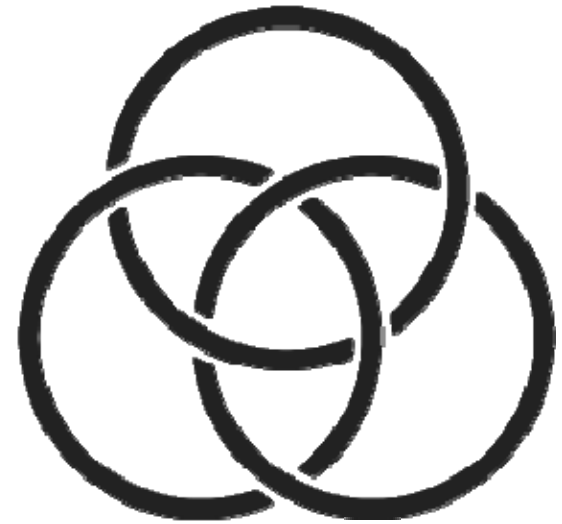
- Can detect **many-variable** or **non-linear** interactions not hypothesized in advance, i.e., it is explicitly designed for **exploratory** search
- **Transparent** -- not a black box like deep learning NNs
- Easily **interpretable & communicable**
- Designed for **nominal** variables
- Can also analyze **continuous** variables via **binning**
- **Prediction/classification**, **clustering/network** models
- **Time series, spatial** analyses
- Overlaps common **statistical & machine-learning** methods, but has unique features

WHY RA MIGHT BE OF INTEREST 2/2

- Analyses at **3 levels of refinement**:
 - coarse (very fast, in principle *many* variables)
 - fine (slower, 100s of variables) (~500 is max so far)
 - ultra-fine (slow, < 10 variables)
- **Standard application**: frequency data $f(A_i, B_j, C_k, Z_l)$
- Variety of **non-standard capabilities**
 - Data: set-theoretic relations & mappings
 - Predict continuous dependent variables
 - Integrate multiple inconsistent data sets (not yet in Occam)
 - Regression-like Fourier version (not yet in Occam)

OCCAM, SOFTWARE FOR RA

- OCCAM, developed by Systems Science Program, Portland State University, is now **open source**
- <https://www.occam-ra.io/>
- github.com/occam-ra/occam
- Contact me if you want to become involved:
- zwick@pdx.edu



RA (DMM) WEB PAGE

<http://pdx.edu/sysc/research-discrete-multivariate-modeling>

Portland State Systems Science Graduate Program | Research: Discrete Multivariate Modeling - Mozilla Firefox

File Edit View History Bookmarks Tools Help Google

Portland State Systems Science Graduate Pr... +

www.pdx.edu/sysc/research-discrete-multivariate-modeling

Portland State UNIVERSITY Systems Science Graduate Program myPSU Contact SYSC Quick Menu

Courses Program Faculty Students Research Resources

PSU » System Science » Research » Research: Discrete Multivariate Modeling

Research: Discrete Multivariate Modeling

The methods used are also known in the systems literature as "reconstructability analysis" (RA). RA overlaps significantly with the fields of logic design and machine learning and with log-linear statistical modeling. The papers "Wholes and Parts in General Systems Methodology" and "An Overview of Reconstructability Analysis" listed below offer a concise review of RA methodology.

Projects

Theory/Methodology

OCCAM: RA software for data analysis & data mining

- [Occam3 \(web accessible; try it out\)](#)
- [User manual \(PDF\)](#)

EDA: Extended Dependency Analysis

Heuristic RA search for loopless models.

- [Download executable, sample files, and documentation \(for Windows\)](#)

RA utility programs

Below is the lattice of structures for a 4-variable *directed* system with 1 dependent variable (output).
Boxes = relations; lines = variables;
bold lines = the dependent variable.

PAST RA APPLICATIONS

- ***BIOMEDICAL***

Gene-disease association, disease risk factors, gene expression, health care use & outcomes, **dementia**, diabetes, heart disease, prostate cancer, brain injury, primate health, surgery

- ***FINANCE-ECONOMICS-BUSINESS***

Stock market, bank loans, credit decisions, apparel analyses, market segmentation

- ***SOCIAL-POLITICAL-ENVIRONMENTAL***

Socio-ecological interactions, wars, urban water use, rainfall, forest attributes

- ***MATH-ENGINEERING***

Logic circuits, automata dynamics, genetic algorithm & neural network preprocessing, chip manufacturing, pattern recognition, decision analysis

- ***OTHER***

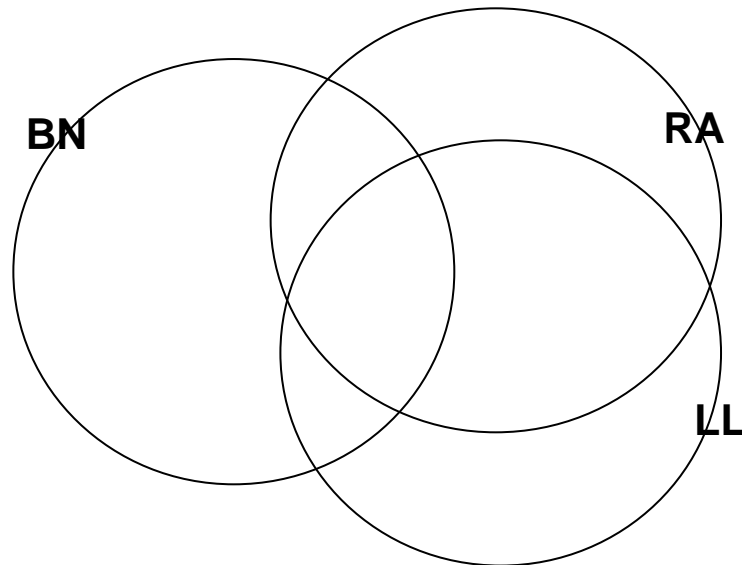
Textual analysis, language analysis

OVERLAP WITH STATISTICAL, ML METHODS

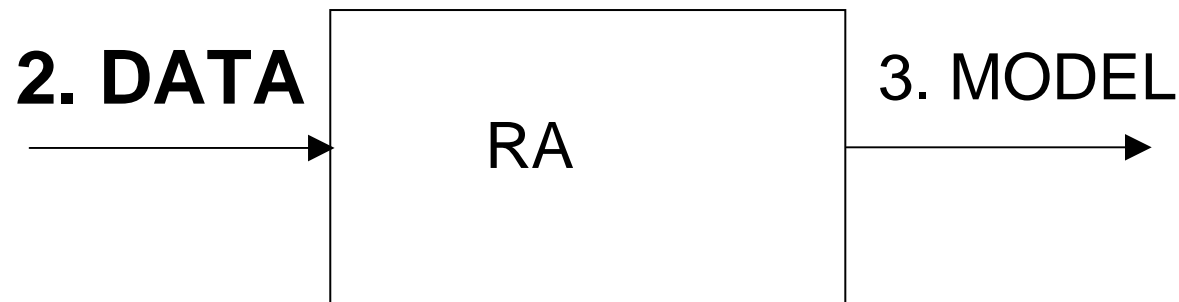
Closely related to other PGM methods, e.g., **log linear** (LL) (& logistic regression) models & **Bayesian networks** (BN)

Where methods overlap, they're **equivalent**

These PGM methods totally **different** from **neural nets**



1. Introduction: what is RA
- 2. Input data to RA**
3. Output model from RA



FORM OF DATA

Variables

- Type: **nominal**; **bin** if continuous (continuous DV needn't be binned)
- Number: few variables to 100s (in principle >1000s coarse analysis)

Data analysis

directed system

- IV-DV distinction: *predict/classify* a DV from IVs

neutral system

- No IV-DV distinction: model association, **clustering**

FORM OF DATA

- frequency(A_i, B_j, C_k, Z_l) or individual cases

				frequency
A_0	B_0	C_0	Z_0	13
A_0	B_0	C_0	Z_1	2
A_0	B_0	C_1	Z_0	9
A_0	B_0	C_1	Z_1	11
...	—
				N

N = sample size

	A	B	C	Z
case ₁	A_0	B_0	C_0	Z_0
case ₂	A_1	B_2	C_3	Z_1
...				
case _N	A_0	B_0	C_0	Z_0

Cases are indexed by
 individual (in a population),
 time, or
 space

$$\text{frequency}(ABCZ) / N = p_{\text{data}}(ABCZ)$$

OCCAM input file, **DATA** CASES INDEXED BY **INDIVIDUAL**

ID ,413,0,ID #Index specifying individual
 APOE ,2,1,Ap
 Gender ,2,1,Sx
 Education ,3,1,Ed
 AgeLastExam ,3,1,Ag
 rs1801133 ,3,1,A
 rs3818361 ,4,1,B
 rs7561528 ,3,1,C
 rs744373 ,3,1,D
 rs6943822 ,3,1,E
 rs4298437 ,3,1,F
 rs7012010 ,3,1,G
 rs11136000 ,3,1,H
 rs10786998 ,4,1,J
 rs11193130 ,4,1,K
 rs610932 ,3,1,L
 rs3851179 ,3,1,M
 rs3764650 ,4,1,N
 rs3865444 ,4,1,P
 Dementia ,2,2,Z

DEMENTIA EXAMPLE
 Z = 0 no disease; Z = 1 disease

#ID	Ap	Sx	Ed	Ag	A	B	C	D	E	F	G	H	J	K	L	M	N	P	Z
101	0	0	2	2	1	1	0	1	2	2	1	1	2	0	1	1	2	2	1
103	0	0	2	1	0	2	2	0	1	1	1	2	2	0	1	1	0	1	0
111	0	1	2	1	2	2	1	1	0	1	1	2	1	1	2	2	0	1	0
112	0	0	2	2	2	2	1	1	1	2	1	1	0	2	2	0	0	2	0
118	0	1	0	2	2	2	2	0	0	1	1	1	.	.	1	1	0	2	0
120	0	1	2	2	1	2	1	1	0	1	1	2	1	1	1	2	0	.	1
121	0	0	2	2	2	2	1	1	2	0	0	0	2	0	1	1	1	.	1
122	0	0	1	2	1	2	1	1	2	0	0	2	2	0	1	1	1	1	0
123	0	0	2	2	2	2	2	0	1	1	0	0	2	0	2	1	0	1	1

...

DATA CASES INDEXED BY TIME

	X	Y	Z
t-4	--	--	--
t-3	0	1	2
t-2	3	4	5
t-1	6	7	8
t	9	10	11

original data

A	B	C	X	Y	Z
--	--	--	--	--	--
--	--	--	--	--	--
0	1	2	3	4	5
3	4	5	6	7	8
6	7	8	9	10	11

transformed data

Values are labels for variable states at particular times

XYZ = **generating variables**

Apply **mask** (here # lags = 1) to data

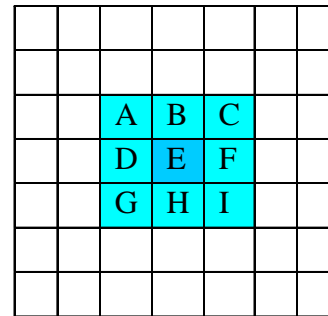
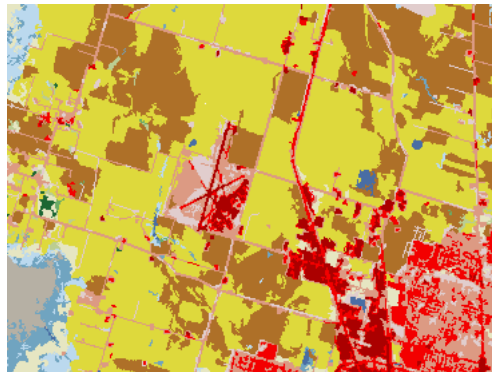
Mask adds lagged variables, $ABC(t) = XYZ(t-1)$

E.g., $A(t) = X(t-1)$, labeled 6

Masking: time series data → **atemporal** data

DATA CASES INDEXED BY SPACE : 1 generating variable

A,14,1,A
 B,14,1,B
 C,14,1,C
 D,14,1,D
E,14,2,E
 F,14,1,F
 G,14,1,G
 H,14,1,H
 I,14,1,I



Moore neighborhood

E = DV

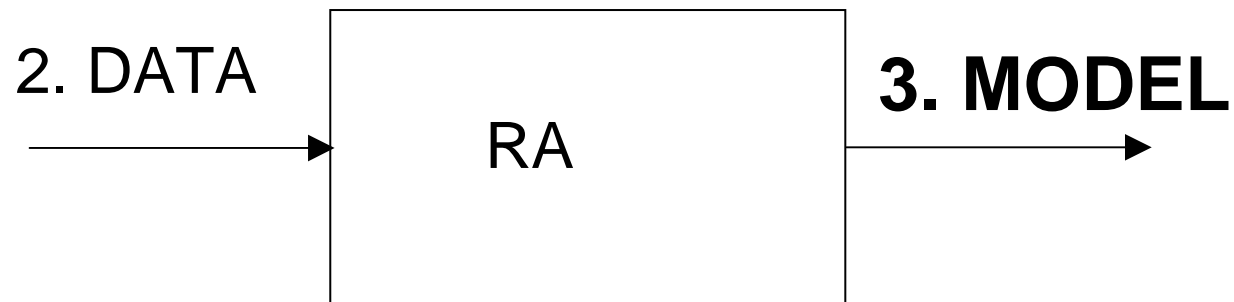
A,B,C,D,F,G,H,I = IVs

IVs & DV have 14 possible states

#A	B	C	D	E	F	G	H	I
71	71	71	71	71	71	71	71	71
71	71	71	71	71	71	71	71	71
71	71	71	71	71	71	71	71	71
71	71	71	71	71	71	71	71	71
71	71	71	71	71	71	71	71	71
71	71	71	71	71	71	71	71	71
71	71	71	71	71	71	71	71	71
71	71	71	71	71	71	71	71	71
71	71	71	95	71	95	71	71	71
95	71	95	95	71	95	71	71	71
95	95	95	95	95	71	71	71	95
71	95	95	90	95	95	71	95	95
95	95	90	90	71	95	95	95	95
95	90	90	90	95	90	95	95	90

...

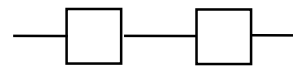
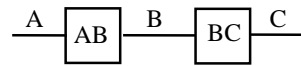
1. Introduction: what is RA
2. Input data to RA
- 3. Output model from RA**



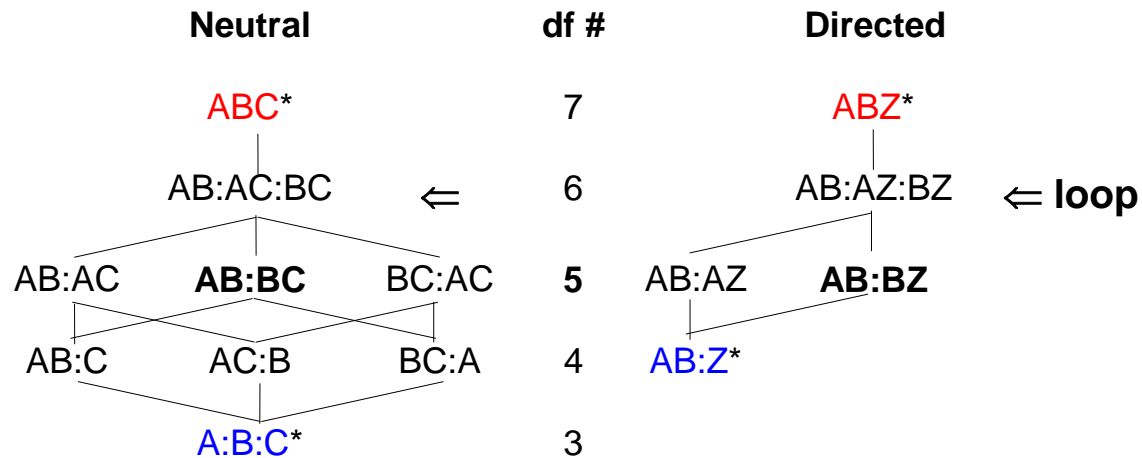
MODEL = STRUCTURE APPLIED TO DATA

A structure (graph or hypergraph) is a set of relationships (GT)

Specific structure **AB:BC** General structure



LATTICE OF SPECIFIC STRUCTURES (3 variables)



* **Reference model is data or independence**
 # df (degrees of freedom) values are for binary variables

***STRUCTURES* (GT)**

Combinatorial explosion

# variables	3	4	5	6
# general structures <small>neutral</small>	5	20	180	16,143
# specific structures <small>neutral</small>	9	114	6,894	7,785,062
one DV <small>directed</small>	5	19	167	7,580
one DV, no loops <small>directed</small>	4	8	16	32

NEED **INTELLIGENT HEURISTICS** TO **SEARCH LATTICE**

Can analyze 100s of variables, & for simple models, many more.

TYPES OF STRUCTURES (GT)

FOR **PREDICTION / CLASSIFICATION** (directed system)

- **Variable-based**

- **no loops** [coarse] *many* variables (**fast**)
IV:ACZ simple prediction, **feature selection**

- **with loops** [fine] up to 100s of variables (slow)
IV:ABZ:BCZ better prediction

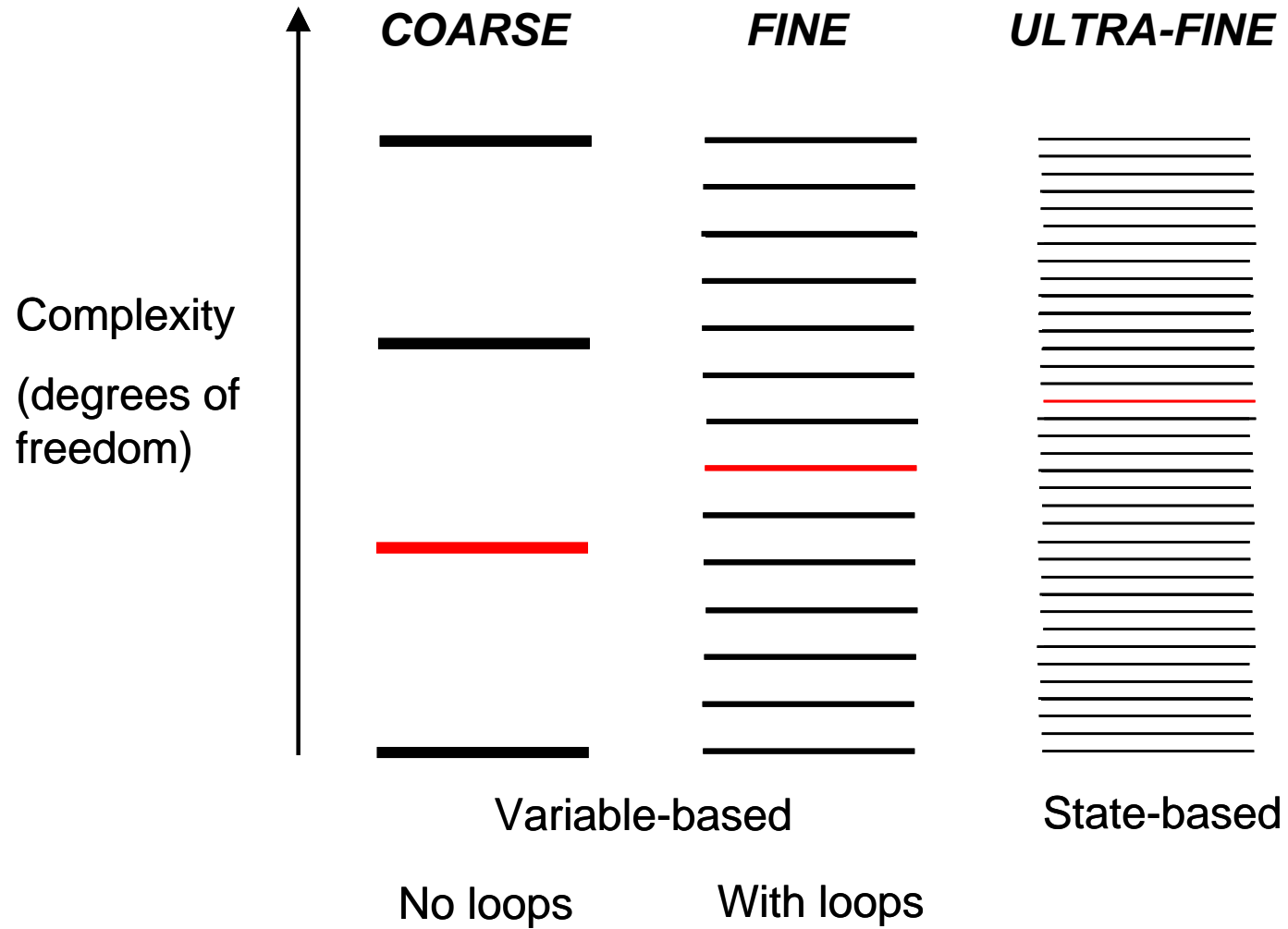
- **State-based** [ultra-fine] < 10 variables (**very slow**)

- IV:Z: A₁B₁Z : B₂C₃Z₁ best prediction; detailed models

“IV” = ABC (all IVs); Z = DV

All directed system models include an IV component

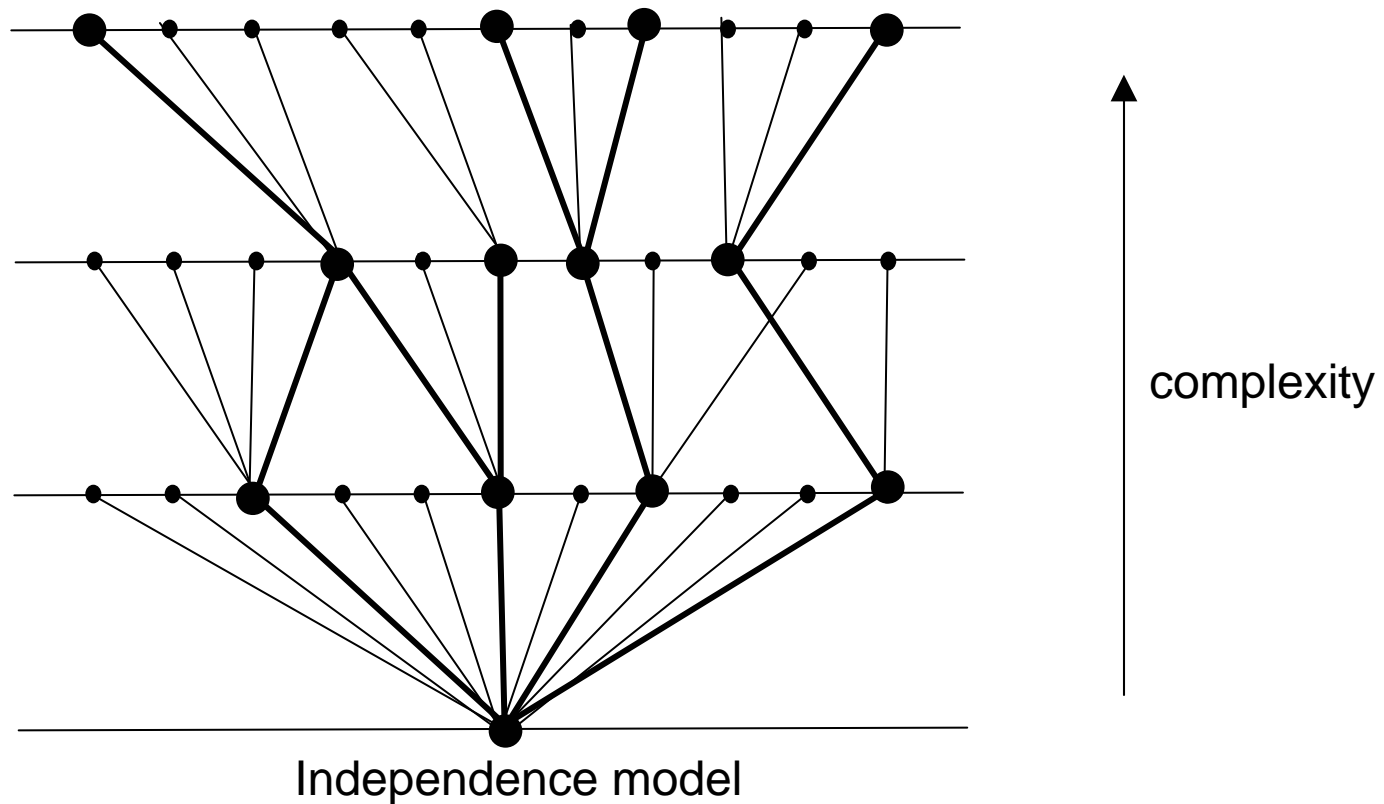
TYPES OF STRUCTURES (GT)



OCCAM SEARCH of LATTICE of STRUCTURES

beam search, levels = 3, width = 4 (node = model)

(there are many other search algorithms)



MODEL = PROBABILITY DISTRIBUTION (IT)

Neutral system:

- Model = calculated *joint* distribution,
e.g., $p_{ABC:AZ:BZ}(A_i B_j C_k Z_l)$

Directed system:

- Model = calculated *conditional* distribution,
e.g., $p_{ABC:AZ:BZ}(Z_l | A_i B_j C_k)$
- Distribution gives *rule* to *predict* Z from A,B,C
And *increase/decrease risk* relative to margins

SELECTING A MODEL (IT)

1. High **information** (or low **error**) in model

Directed system

- *Info-theory measure: high ΔH , reduction of uncertainty of DV*
- *Generic measure: high %correct, accuracy of prediction*

2. Low **complexity**: df, degrees of freedom

3. Information \leftrightarrow complexity **tradeoff**

- Statistical **significance** (Chi-square p-values)
- **Integrated** measures: AIC, BIC
(Akaike & Bayesian Information Criteria)
- BIC a **conservative** selection criterion

UNCERTAINTY REDUCTION: SIMPLE EXAMPLE

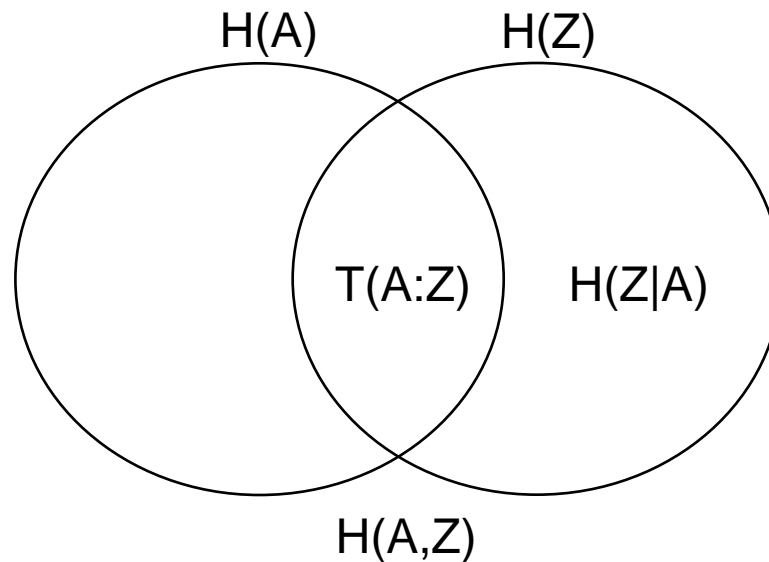
2 variables: $IV=A$; $DV = Z$; $T(A:Z)$ =mutual information (*association*)

- *Uncertainty reduction* is like variance explained

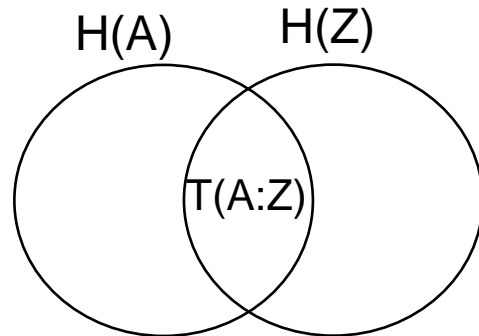
Model AZ = predict Z , i.e., reduce $H(Z)$, by knowing A

- Uncertainty *reduced* = $T(A:Z)$; uncertainty *remaining* = $H(Z|A)$

$\Delta H = T(A:Z) / H(Z)$ *fractional uncertainty reduction* (express in %)



UNCERTAINTY REDUCTION: SIMPLE EXAMPLE



	Z_0	Z_1	
A_0	$.67*.5$	$.33*.5$.5
A_1	$.33*.5$	$.67*.5$.5
df=3	.5	.5	

- $p(Z_1)/p(Z_0) = 1:1$, not knowing A \rightarrow 2:1 or 1:2, knowing A
- $\Delta H(Z) = T(A:Z) / H(Z) = 8\%$
- 8% reduction in uncertainty is *large* (unlike variance!)

SELECTING A MODEL DEMENTIA EXAMPLE

<u>Criterion</u>	<u>model</u>	<u>ΔH(%)</u>	<u>Δdf</u>	<u>%c</u>	<u>ΔBIC</u>
<i>Variable-based with loops (fine)</i>					
BIC	IV: Ap Z : Ed Z : K Z	16	5	70	59
p-value	IV: Ap Z : Ed Z : K Z : C Z : L Z	18	9	71	
AIC	IV: B Ap Z : Ed Z : K Z : C Z	20	11	72	

State-based (ultra-fine)

BIC	(model below; each interaction = 1 df)	20	6	72	81
IV:Z: Ap ₁ Z : Ed ₀ Z : K ₂ Z : Ap ₀ Ed ₂ C ₂ Z : Ap ₀ Ed ₁ C ₂ K ₁ Z : Ap ₀ Ed ₁ C ₀ K ₁ Z					

Models integrate multiple predicting interactions

IV = ApEdCKL... (all the independent variables);

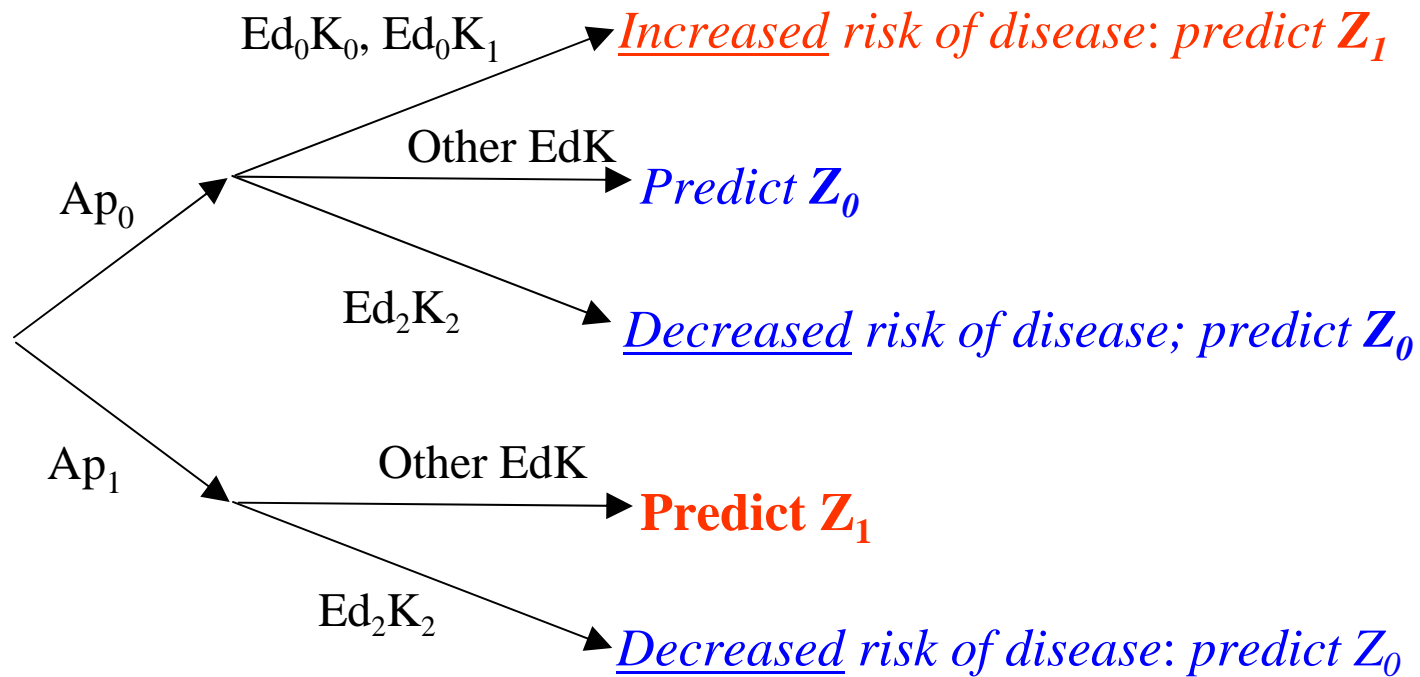
%c(IV:Z) = 52

PROBABILITY DISTRIBUTION DEMENTIA EXAMPLE

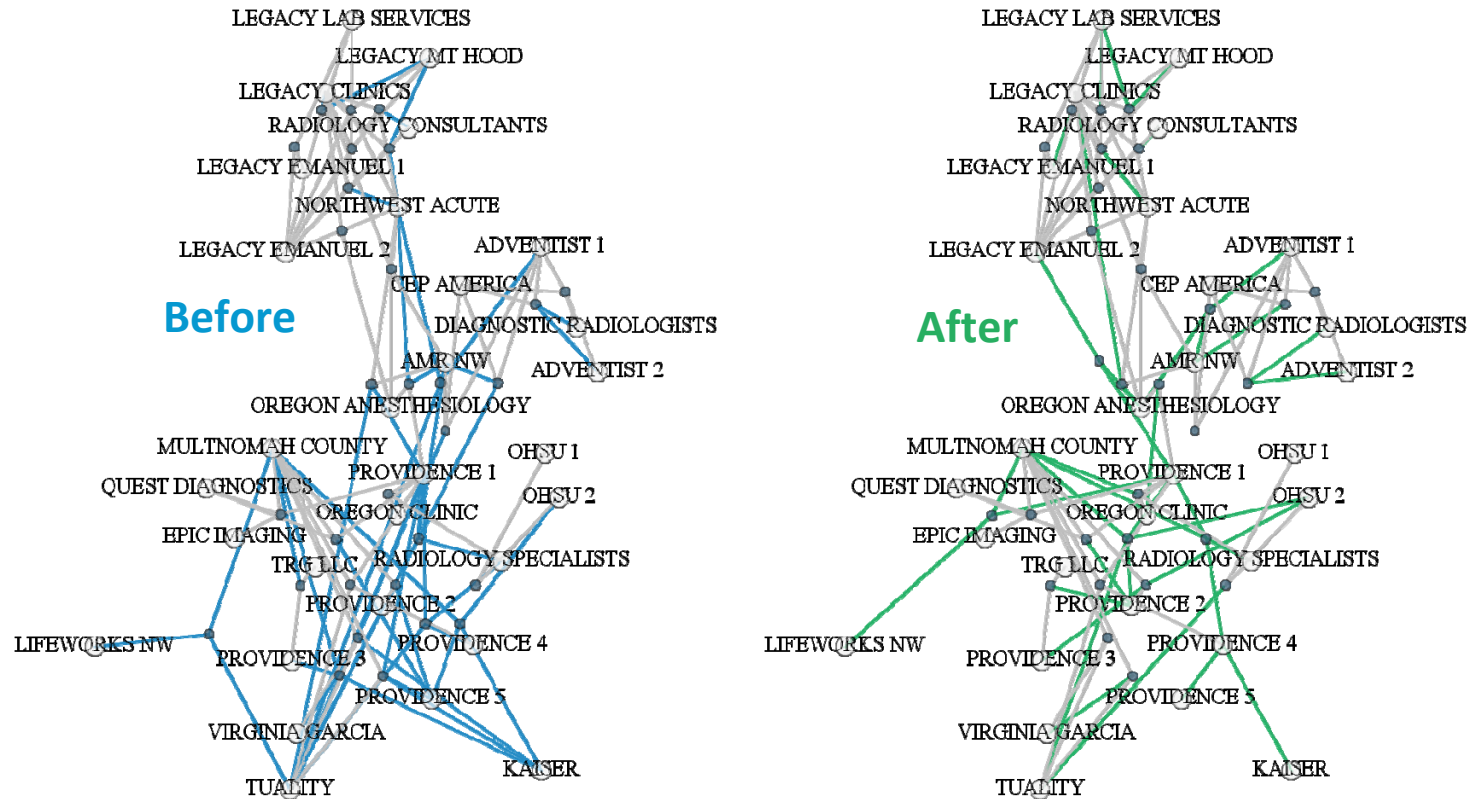
DATA				MODEL IV:ApZ:EdZ:KZ						
IV				obs p(Z IV)		calc p(Z IV)		rule	p-value	
Ap	Ed	K	freq	Z ₀	Z ₁	Z ₀	Z ₁		p _{rule}	P _{Ap}
0	0	0	4	0.0	1.000	.122	.878	1	0.131	0.028
0	0	1	8	.125	.875	.124	.876	1	0.033	0.002
0	0	2	4	.250	.750	.294	.706	1	0.409	0.138
0	1	0	31	.645	.355	.616	.384	0	0.198	0.707
0	1	1	37	.622	.378	.619	.381	0	0.147	0.714
0	1	2	23	.783	.217	.827	.173	0	0.002	0.072
0	2	0	66	.636	.364	.640	.360	0	0.023	0.894
0	2	1	61	.656	.344	.644	.357	0	0.025	0.942
0	2	2	33	.848	.152	.842	.158	0	0.000	0.020
0	--	--	267	.648	.352	.648	.352	0		
1	0	0	1	.000	1.000	.026	.974	1	0.343	0.571
1	0	1	7	.143	.857	.026	.974	1	0.012	0.134
1	0	2	2	.000	1.000	.074	.926	1	0.228	0.514
1	1	0	13	.308	.692	.234	.766	1	0.055	0.709
1	1	1	24	.167	.833	.237	.763	1	0.010	0.633
1	1	2	11	.545	.455	.478	.522	1	0.884	0.146
1	2	0	32	.219	.781	.254	.746	1	0.005	0.732
1	2	1	39	.256	.744	.256	.744	1	0.002	0.735
1	2	2	17	.529	.471	.504	.496	0	0.973	0.040
1	--	--	146	.281	.719	.281	.719	1		
			413	.518	.482	.518	.482	0		

DECISION TREE DEMENTIA EXAMPLE

Obtained from conditional probability distribution
Increase/decrease of risk compared to prediction based only on A_p

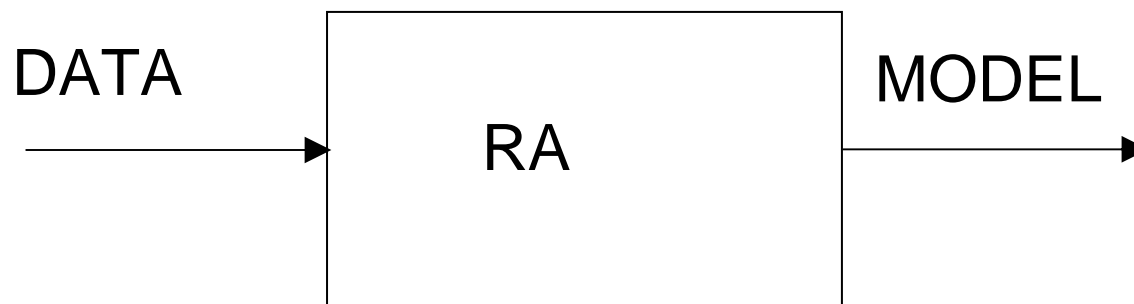


NEUTRAL ANALYSIS EXAMPLE



- THANK YOU.
- zwick@pdx.edu

1. Introduction: what is RA
2. Input data to RA
3. Output model from RA
4. RA methodology

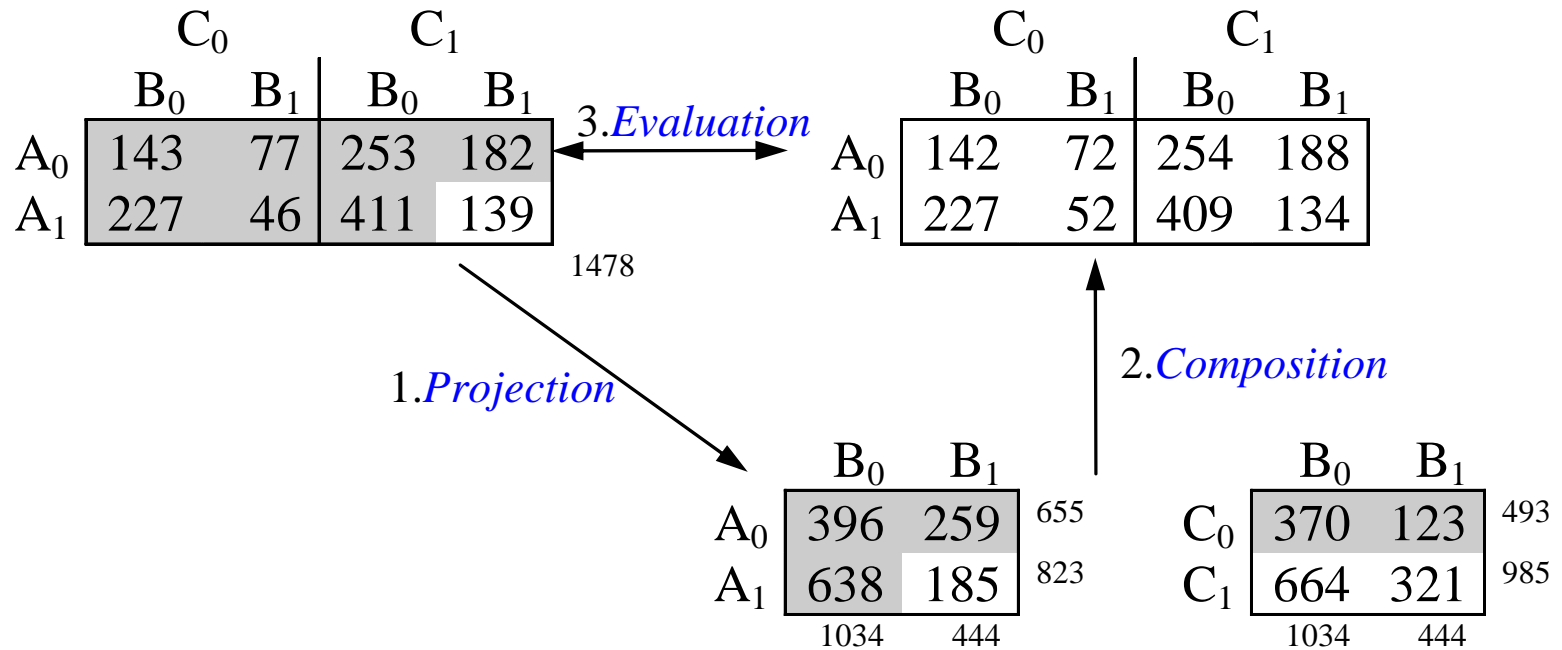


GENERATE MODEL

frequencies shown, not probabilities

data: observed ABC (df=7)

model: calculated ABC_{AB:BC}



model: AB:BC (df=5)

GENERATE MODEL (*Projection, Composition*)

- *Projection* = sum frequencies or probabilities
- *Composition*

Maximize model *entropy subject to* model *constraints*

Model entropy: $H(p_{\text{model}}) = - \sum p_{\text{model}} \log_2 p_{\text{model}}$

E.g., for model AB:BC, *maximize* $H(p_{\text{AB:BC}})$ *subject to*

$$p_{\text{AB:BC}}(\text{AB}) = p_{\text{data}}(\text{AB})$$

$$p_{\text{AB:BC}}(\text{BC}) = p_{\text{data}}(\text{BC})$$

Composition is *critical computational step*; done

(a) Algebraically (very fast)

loopless models

(b) *Iteratively* (Iterative Proportional Fitting)

models with loops

EVALUATE MODEL (1/2)

- *Evaluation* (1 = data dependent; 2 = data independent)

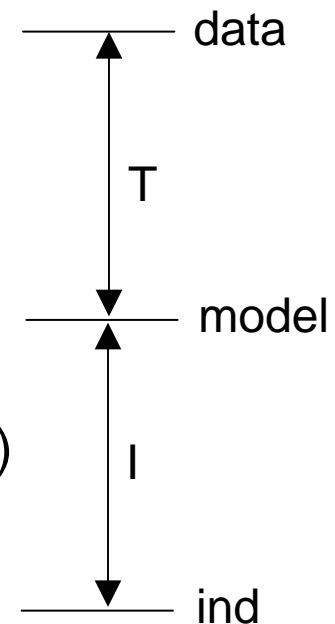
1. [reference=data]

$$\begin{aligned} \text{error, } T_{\text{model}} &= H_{\text{model}} - H_{\text{data}} \\ &= \sum p_{\text{data}} \log_2(p_{\text{data}}/p_{\text{model}}) \end{aligned}$$

[reference=independence]

$$\begin{aligned} \text{information, } I_{\text{model}} &= H_{\text{ind}} - H_{\text{model}} \\ &= \sum p_{\text{data}} \log_2(p_{\text{model}}/p_{\text{ind}}) \end{aligned}$$

$$\text{uncertainty reduction} = H(\text{DV}) - H_{\text{model}}(\text{DV} | \text{IV})$$



2. [reference=independence]

$$\text{complexity} = \Delta df = df_{\text{model}} - df_{\text{ind}}$$

EVALUATE MODEL (2/2)

Trade off information (or error) & complexity, define **best model** criterion, via:

Use likelihood ratio Chi-square, $LR = k N T$

- **p-values** from ΔLR , Δdf , Chi-square table

Or linear combinations of information & complexity

- **$\Delta AIC = \Delta LR + 2 \Delta df$**
- **$\Delta BIC = \Delta LR + \ln(N) \Delta df$**


BASIC OCCAM ACTIONS

- **Search** = **exploratory** modeling, examine many models, find best or good ones
(OCCAM actions: Search, SB-Search)
- **Fit** = **confirmatory** modeling, look at one model in detail (see probability distribution) & use for prediction
(OCCAM actions: Fit, SB-Fit)

(OCCAM actions: Show Log, Manage Jobs = managerial functions)

OCCAM Initial Screen

← → ↻ 🏠 🖨️ dmit.sysc.pdx.edu/weboccam.cgi

 **Portland State**
UNIVERSITY

Occam version 3.4.0 — Tue Jun 19 14:41:08 2018

Do Search Do SB-Search Do Fit Do SB-Fit | Do Compare

Show Log Manage Jobs Cached Data Mode

© 2000-2017

INFORMATION ON RA

- Review articles on DMM page
 - “Wholes & Parts in General Systems Methodology” (accessible)
 - “An Overview of Reconstructability Analysis” (encompassing)
- Krippendorff, Klaus (1986). *Information Theory. Structural Models for Qualitative Data* (Quantitative Applications in the Social Sciences Monograph #62). New York: Sage Publications.
- *International Journal of General Systems*
- *Kybernetes*, Vol. 33, No. 5/6 2004: special RA issue