**Discrete Multivariate Modeling I and II (SySc 551 and 510) Study Notes**

**Martin Zwick, PhD**

Susan Sienko Thomas  
Systems Science Graduate Program  
Portland State University  
2011

Table of Contents

Reconstructability Analysis Models 5

Basic Concepts 6

Univariate Uncertainty (H); diversity and information 6

H (uncertainty) increases with: 6

Association between A and B conceptually 7

Want entropy to be: 7

Entropy and Diversity 7

Information: -ΔH= Hinitial-Hfinal , 7

Measures and Models 8

Transmission (T) (mutual information, constraint, uncertainty reduction) 8

Bivariate and Conditional Uncertainties 8

Contingency Tables 8

Computations on contingency tables 8

Q distribution 9

(State) decomposition of univariate uncertainty 10

Variable decomposition of transmission 10

T in transmission and sequential situations: MILLER 11

Sequential Situation in communication context 11

T as a likelihood ratio in relation to uncertainty 12

Statistical Significance-Krippendorf (p87)-Chi square 12

Likelihood ratio in Chi-square 12

Assessing Statistical Significance 12

Three variable models 13

Three variable models and contingency tables (of probabilities) 13

Law of distribution for conditional transmissions 14

Simpson’s Paradox Example that demonstrates the T(A:B) can be TC(A:B) 15

Structures 16

Criteria for a good model 16

Uncertainty-Transmission-Information 16

Ways to pick a good model 17

Training versus Test data 17

Choosing models statistically 18

When developing a model 18

Algorithm that detects whether you have loops-Krippendorg p42 19

Q→Transmission 19

Degrees of Freedom-Krippendorff Method 20

Degrees of Freedom-Log linear Methods Krippendorf pg 130-131 20

An alternative way is to write out call unique relations 21

Decomposition of df 21

Lattice of structures, ordinality 23

Computational complexity 23

Complexity of individual structures 23

Generating a Specific Lattice of Structure for 3 Variables 24

Generating a Specific Lattice of Structure for 4 Variables 26

Nearest ancestor and descendent 30

Set-theoretic relation 30

Example: 30

Set Theoretic Reconstructability Analysis (SRA) 31

Information Reconstructability Analysis (IRA) 33

Information Theory 33

What is Information? 33

Shannon Entropy 33

IPF 34

Data 🡪 Projection 🡪 Calculation 🡪 Data 34

Data (ABZ) 34

Projection (AB:BZ) 34

Calculation (ABZAB:AZ) – each q value will be replaced with actual values in terms of parameters a through h 35

Calculated table formulas: 35

q1 for ABZAB:AZ = values for A0B0Z0 for each of the probabilities: 35

IPF (Iterative Proportional Fitting) – method for generating calculated table (Calculation) for LOOPED models 36

Steps to IPF 36

IPF – Impose AB 37

IPF – How to impose AB 38

IFP – Impose BZ – How to do projection of BZ from qA 38

IPF – simplify Projection BZ for qAB table 38

IPF – Impose BZ onto qAB 39

IPF – qAB:BZ calculated table – simplified more 39

IPF – now do it again! For a looped model, you’d need to repeat: impose AB, then impose BZ, over and over again until your calculated table doesn’t change. But this example is loopless, so no need to repeat. 39

State-Based RA 40

Information-theoretic variable-based RA 40

State-Based vs Variable Based 40

State Based Specifics 41

General Concepts of State-Based Models 41

# Reconstructability Analysis Models

* Simple: loopless models, fast processing
* Complex: models with loops, iterative
* Very Complex: state based models, very slow

|  |  |  |
| --- | --- | --- |
| **Variable type** | Nominal-discrete and un ordered, most general  Ordinal-discrete  Quantitative-continuous | |
| **System type** | Directed | Has inputs/outputs (dependent/independent variables)  Deterministic or stochastic  Always has one component that collects together all inputs  Characterized by the # predicting components |
| Neutral | No input/output distinction  Non-deterministic  Variables have equal status |
| **Data type** | Information theoretic (IRA) | Frequency/probability distribution  Most commonly used version  Complexity measured by degrees of freedom (df)-number of parameters needed to specify a model |
| Set theoretic (SRA) | Relations/functions  “Crisp probabilistic” |
| **Problem type** | Reconstruction | Decomposition of frequency or probability distributions, confirmatory |
| Exploratory | Data analysis/mining, identification, composition |
| **Algorithm** | Maximize entropy  Minimize squared deviation | |
| **Method Type** | Variable-based modeling (VBM)  State-based modeling (SBM)  Latent-variable based modeling (LVBM) | |

## Basic Concepts

* Each variable has a probability of being in a specific state (out of a number of possible states)
* If the probabilities of which state the variable will be in is equal for all states, you don’t know what state the variable will be in
* WANT to know (want to have some certainty) about the future state of the variable (don’t want to be surprised)
* WANT to minimize uncertainty (or at least KNOW how much uncertainty you have)
* Uncertainty is measured in terms of entropy, where H is SHANNON entropy
* Difference between uncertainty (initial (what your don’t know) and final (what you do know)) is INFORMATION (change in uncertainty)
* Final uncertainty is conditional as it is the initial uncertainty, knowing something (state of another variable)

## Univariate Uncertainty (H); diversity and information

* H(x)= -Σp(xj) log2p(xj), where p=probability, this is SHANNON entropy, means uncertainty
* Defines the weighted measure of surprise in the outcome
  + If have 50/50, then H=.5 and T=.5, if H=.95 and T=.05, this is much less surprise
* Σpj log21/pj, where pj is the expected or average surprise and log21/pj is the surprise in the outcome j on a log scale

### H (uncertainty) increases with:

* + An increase in the cardinality of states (n)
  + Uniformity of the probability distribution
  + H(A,B)= H(B)- H(B|A), where H(B) is the amount of uncertainty you have about the state variable Z given the probabilities about the states of B

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| B0 | B1 | Some uncertainty because the probabilities are not 0 and 1 | B0 | B1 |
| 0.4 | 0.6 | 0 | 1 |

* + H(B|A) is the amount of uncertainty you have about the state variable A given the probabilities about the states of B AND knowing the state of the other variable A

|  |  |  |
| --- | --- | --- |
|  | B0 | B1 |
| A0 | 0.1 | 0.2 |
| A1 | 0.3 | 0.4 |

* H(A,B)= H(A) – H(A|B) (H of A given B) or H(B)- H(B|A) (H of B given A)
  + T (A:B)
  + Mutual information
  + Constraint is the information about B gained by knowing A, or uncertainty reduction
  + Association
  + Amount of information captured in the model, increases as you go up
  + When looking at data AB=associated, A:B=not associated, independent

### Association between A and B conceptually

Measure conception

Model conception= H(A:B) – H(AB)

### Want entropy to be:

* + Continuous
  + H should be a monotonic increase (function of n)
  + If choice is broken down into two consecutive choices, H should be the weighted sum of the individual values of H

### Entropy and Diversity

* Uncertainty (entropy)=diversity

### Information: -ΔH= Hinitial-Hfinal ,

* + Information and uncertainty are inversely related
  + Information is a measure of the change in uncertainty of a set of variables (set of probabilities for states of variables)
  + Bateson: “information is news of difference”
* Information difference
  + Difference between 2 probability distributions
  + Difference between entropies of 2 different models
* Entropy (error) increases as you go down the lattice
* Measure error from top down

## Measures and Models

### Transmission (T) (mutual information, constraint, uncertainty reduction)

* Going UP, T is constraint in the data, since data is AB or top of the lattice
* Going DOWN, T is the error in the X:Y model (the loss of constraint)
* T is always POSITIVE
* T is similar to correlation, constraint of association
* When T is zero= NO association, when T > 0, ASSOCIATION
* Association, means it is explaining it away
* If Tc(A:B) ~ 0, then C explains away the A,B association
* If probabilities between XY and X:Y are different, then T > 0

## Bivariate and Conditional Uncertainties

### Contingency Tables

* Contingency tables can be a set of probabilities OR frequencies
* If you have probabilities, DON’T have sample size, so can’t determine statistical significance
* If have frequencies, then can assess statistical significance, can determine the sample size
* X:Y table is created by cross multiplying the margins
* X:Y means that you know the X and Y distributions SEPARATELY

### Computations on contingency tables

**Bivariate models only have 2 variables so you only have 2 models**

**XY relation (data)**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Y0 | Y1 | Projection for x |
| X0 | .1 | .2 | .3 (sum row) |
| X1 | .3 | .4 | .7 (sum row) |
| Projection for y | .4 (sum column) | .6 (sum column) |  |

H(A:B)=H(A) + H(B)

H(A) =Γ (0.3,0.7) (each value for A rows)

H(B) =Γ (0.4,0.6) (each value for B columns)

H(A:B)= Γ (0.3,0.7) +Γ (0.4,0.6), which is the same as H(A:B)= Γ(.12, .18, .28, .42)

Maximum entropy method is the same as maximum likelihood method

**Model of X:Y from the XY data, if independent, you expect margins to multiply**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Y0 | Y1 |  |
| X0 | .12 (.3\*.4) | .18 (.3\*.6) | .3 (sum row) |
| X1 | .28 (.7\*.4) | .42 (.7\*.7) | .7 (sum row) |
|  | .4 (sum column) | .6 (sum column) |  |

Calculating X:Y (0,0)= p(A=0) \* p(B=0)

If the numbers in the model are the same as in the dataset, there is no loss of information from XY to X:Y, but since they are not the same, there is loss.

H(Y⏐X)=uncertainty of B given A (at a value of A)

H(Y⏐X)=0=Γ(.1/.3,.2/.3) value of XoY0 and XoY1 divided by the projection for X0

H(Y⏐X)=1=Γ(.3/.7,.4/.7) value of X1Y0 and X1Y1 divided by the projection for X1

Therefore, H(Y⏐X)= H(X,Y)- H(X)

=Γ (.1, .2, .3, .4)- Γ (.3, .7)

= Σ H(Y⏐Xi) p (Xi)

=p(X0) H(Y⏐X0) - p(X1) H(Y⏐X1)

= .3 =Γ(.1/.3,.2/.3)- .7=Γ(.3/.7,.4/.7)

### Q distribution

* Solution to the maximum H(Q) subject to the constraints of the model
* Constraints of the model X:Y, know X and know Y
* Want the most uniform distribution that has the margins

H(Q)= Γ (Q1,Q2,Q3,Q4)

Subject to knowledge of X margins

Q1 +Q2 = .3

Q3 +Q4 = .7

Which is really that the observed probabilities are:

Q1 +Q2 = P1 + P2 =.3

Q3 +Q4 = P3 + P4 =.7

Subject to knowledge of Y margins

Q1 +Q3 = .4

Q2 +Q4 = .6

By definition

Q1 +Q2 +Q3 +Q4 =1

While the data has 3 degrees of freedom (df) , the df of the model is 2, since we know the margins

df(X:Y)= df(X) + df(Y), (2-1) +(2-1)= 2

df(X)= cardinality of x-1, where cardinality is the number of states of X

df(Y)= cardinality of y-1

### (State) decomposition of univariate uncertainty

Decomposition, first look at decomposition wrt states, then look at decomposition with wrt variables

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | .1 | .2 | .3 | .4 |  |
| Y | Y1 | Y2 | Y3 | Y4 | Micro states |
| X | X1 | | X2 | | Macro states |

H(Y) =H(X) +H (Y⏐X)

H(Y) =between macro states + between micro states

H(Y) = Γ (.3, .7) + (.3 Γ (.1/.3, .2/.3) + .7 Γ (.3/.7, .4/.7))

H(Y) = p (X1, X2) + p (X1)\* H(Y⏐X1) + p (X2)\* H(Y⏐X2)

### Variable decomposition of transmission

Total constraint= constraint between subsystems + constraint within subsystems

= error in separated subsystems + error in independent model

T(V:W:X:Y)= T(VW:XY) + T(VW) + T(XY)

Where T(V:W:X:Y)= α, T(VW:XY)=β, T(VW) + T(XY)=γ

T of any model= H model – H data

T(VW:XY)= H (VW:XY) – H (VWXY)

Or from the independence model

T(V:W:X:Y)= H(V) + H(W) + H(X) + H (Y) – H(VWXY)

Most systems are nearly decomposable, meaning α, β, γ is nearly 0

Most constraint is within subsystems, therefore can ignore

## T in transmission and sequential situations: MILLER

T= the information that was sent that was also received

Noise= H(Y⏐X)= H(X,Y)- H (X)

### Sequential Situation in communication context

* Measure of constraint at sequential times
* If you know current state, you know the future state

STOCHASTIC SYSTEM

* Stochastic system: H(t+1⏐t) > 0

DETERMINISTIC SYSTEM

* Stochastic system: H(t+1⏐t) = 0
* In a deterministic system, uncertainty is constantly decreasing or staying the same

## T as a likelihood ratio in relation to uncertainty

* Q is the calculated probabilities

## Statistical Significance-Krippendorf (p87)-Chi square

* Zwick’s Q in the equations is the same as Krippendorfs π

### Likelihood ratio in Chi-square

* L2= 2N Σ p ln (P/Q)
* Compare this to T= Σ p log2 (P/Q), information difference between models, where T measure of error going down, T= information distance from the data to the model
* Converts to L2= 2ln2 N \* T
* Therefore, L2= 1.3863 N \* T
* Chi square integrates the information distance and degrees of freedom, when N=#subjects, T=information distance

### Assessing Statistical Significance

* Change in L2 and change in df, use chi square table to determine the probability of a type 1 error, or p value

## Three variable models

* T(X:Y:Z) = H(H:Y:Z) – H (XYZ)
* T(model) = H(model)- H(data)

### Three variable models and contingency tables (of probabilities)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Z1 | | Z2 | |
|  | Y1 | Y2 | Y1 | Y2 |
| X1 | a | b | c | d |
| X2 | e | f | g | h |

Looking specifically at the model XY:Z T(XY:Z)= H(XY:Z)- H(XYZ)

General Form of Calculated XYZ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Z1 | | Z2 | |
|  | Y1 | Y2 | Y1 | Y2 |
| X1 | q1 | q2 | q3 | q4 |
| X2 | q5 | q6 | q7 | q8 |

Next you make projections for each combination of variables XY, XZ and YZ, which is moving down the lattice of structures to the next model

Projections XY

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Y1 | Y2 |  | Y1 | Y2 |
| X1 | q1 + q3 | q2 +q4 | X1 | a + c | b +d |
| X2 | q5 + q7 | q6 +q8 | X2 | e + g | f +h |

Projections XZ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Y1 | Y2 |  | Z1 | Z2 |
| X1 | q1 + q2 | q3 +q4 | X1 | a + b | c +d |
| X2 | q5 + q6 | q7 +q8 | X2 | e + f | g +h |

Projections YZ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Y1 | Y2 |  | Z1 | Z2 |
| Y1 | q1 + q5 | q3 +q7 | Y1 | a + e | c +g |
| Y2 | q2 + q6 | q4 +q8 | Y2 | b + f | d +h |

To calculate the amount of entropy or error in the XY projection

HZ(X,Y)= H (X,Y|Z) which is the H(X,Y,Z) – H(Z)

HZ(X,Y)= Σ p(Zk) HZk (X,Y)

= (a,b,e,f) Γ (a/a+b+e+f, b/a+b+e+f, e/a+b+e+f, f/a+b+e+f) + (c,d,g,h) Γ (c/c+d+g+h, d/c+d+g+h, g/c+d+g+h, h/c+d+g+h)

which is the entropy of XY knowing Z1 plus the entropy of XY knowing Z2

### Law of distribution for conditional transmissions

TC(A:B)= T(AC:BC)

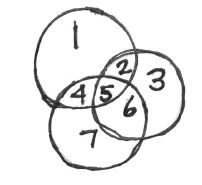
However this is NOT true for H(uncertainty)

HC(A:B) ≠ H(AC:BC) because of the law of uniform subscripting (making things conditional)

HC(A:B) means that C is conditional- H given C

HC(A:B)≠HC(A) + HC(B)

For example using a 3 variable model



HC(A:B) =H(A) + H(B) + H(C)- H(ABC)

HC(A:B) = Areas(1,2,,4,5) + Areas (2,3,4,6) + Areas (4,5,6,7) – Areas (1,2,3,4,5,6,7)

HC(A:B) = Areas(~~1,2,,4,5~~) + Areas (2,~~3~~,5,~~6~~) + Areas (4,5,6,~~7)~~ – Areas (~~1,2,3,4,5,6,7~~)

This leaves areas 2,5,4,5,6 with area 5 being counted twice, it can be either +ve or –ve depending on the relationship to the other variables

McGill and Quaser call area 5 the interaction

Interaction= -H(A) –H(B)- H(C) + H(AB) + H(AC) + H(BC) –H(ABC)

Factor Analysis (linear relation quantitative variables) =Latent Class Analysis (association nominal variables)

### Simpson’s Paradox Example that demonstrates the T(A:B) can be TC(A:B)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Wife | | | |
|  |  | M | S | M | S |
| Husband | M | 10 | 0 | 0 | 10 |
| S | 0 | 10 | 10 | 0 |
|  |  | good | | bad | |

If collapse husband/wife and mountains/seashore

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | Wife | |  |
|  |  | M | S |  |
| Husband | M | 10 | 10 | .5 |
| S | 10 | 10 | .5 |
|  |  | .5 | .5 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | good | |  |
|  |  | M | S |  |
| Bad | M | 10 | 10 | .5 |
| S | 10 | 10 | .5 |
|  |  | .5 | .5 |  |

T(H:W)=0, where S=seashore, M= mountains, and good/bad is the state of the relationship

U(H) + U(W)- U(H,W)

If the probabilities are equal then this is 1, therefore 1 + 1 -2 =0 and the two circle overlap

TR(H:W)= p (Rgood) Tgood(H:W) + p(Rbad) Tbad(H:W)

Tgood(H:W) = U(Hgood) +U(Wgood) –U(H:Wgood) = 1+1-1=1

This is the opposite of explaining away, the latent variable is producing the relation rather than explaining it away

In a directed system with an independent variable and a dependent variable

Tmax = min {H(A), H(B)}

T/H(B) = fraction of uncertainty of B explained by A, COEFFICIENT of CONSTRAINT

= H(B)- H(B⏐A)/ H(B)

T/H(A) = how powerful a predictor A is, measure of EFFICIENCY

= H(A)- H(A⏐B)/ H(A)

# Structures

Structure + data = MODEL

AB:BC:AC

⏐A⏐=2, ⏐B⏐=2, ⏐C⏐=2, when add all of this data from each combination of variables get MODEL

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | B | |  |  | C | |  |  | C | |
| A |  |  |  | B |  |  |  | A |  |  |
|  |  |  |  |  |  |  |  |

## Criteria for a good model

High Information: Low error (T)

High simplicity: low complexity (df)

Need to make tradeoffs between information and complexity

### Uncertainty-Transmission-Information

* ΔH=change in uncertainty
* T= H(X:Y)-H(XY)
* Information=difference between any two models
  + I(M0→Mj)=T(M0)-T(Mij), where T(M0)= difference from data to independence
* Where, H(X:Y) is max uncertainty, least constraint or association, most independent model
* Where, H(XY) is less uncertainty, more constraint/association =DATA
* Therefore, T= H(model)-H(data)
* Going down the lattice
  + T(XY)= H(X:Y)-H(XY)
  + Loss of constraint, increase in uncertainty, loss of information, decreased complexity
* Going up the lattice
  + T(X:Y)= H(X:Y)-H(XY)
  + Gain constraint, increase information, decrease uncertainty

Similar to Transmission

* T=Σ plog2 p/q = Σ p log2 p- Σ p log qj, where Σ p log2 p is the model independent
* L2= Zn Σ p loge p/q =1.3896 NT, where L2 is maximum likelihood, χ2 , goodness of fit

### Ways to pick a good model

1. Statistical Significance (Chisq), looks at p value where the reference is the bottom

* Requires models that are compared to be related (parent/child, nested)

1. Amount of information
2. How well it generalizes in training (training/test splits)
3. AIC and BIC

* AIC (Akaike), little more generous
* AIC of a model(Mj) = -2N Σ p Ln qj + 2dfj, where -2N Σ p Ln qj=error, 2dfj=complexity
* Want AIC to be LOW
* Does not require models to be related (parent/child relationship, “nested”)
* Δ AIC= AICref-AICj, want Δ AIC and Δ BIC to be HIGH
* BIC (Bayesian), very conservative, doesn’t let you go up the lattice very far
* BIC of a model(Mj) = -2N Σ p Ln qj + ln(N)dfj, where -2N Σ p Ln qj=error,

ln(N)dfj =complexity

* Want BIC to be LOW
* Penalizes complexity because it is multiplied by sample size
* Does not require models to be related (parent/child relationship, “nested”)
* Does best to generalize to new data

* Want a balance between complexity and information
* Generalizability (test/training split)
* Final decision can be based on the problem under study

### Training versus Test data

* Want to avoid overfitting
* Have observed data and trying to figure out how decomposable it is: want to increase information and decrease complexity
* Want to stop when you are NOT gaining any more information
* If you go too far up, too complex without information gain (TYPE I error)

## Choosing models statistically

When evaluating a model you want:

1. A model as complex as possible to capture as many interactions as possible
2. A model that is statistically significant relevant to the reference model

|  |  |
| --- | --- |
| **Reference: TOP** | **Reference: Bottom** |
| **Null hypothesis: mi=mo orp =q(mi)** | **Null hypothesis: q(mi) = q(mind)** |
| Information (mo→mi) = T (mi) | Information (mi→m(ind) = T (mi) |
| L2= (mo→mi) = L2 (mi) | L2= ((mi→mind) = L2 (mi) |
| Δ df = df (mo)- df (mi) | Δ df = df (mi)- df (mind) |
| If accept Ho that (mi) is not different from (mo ) the data, then you should try and go down further to a simpler model  If reject Ho that (mi) and (mo) the data are statistically different then you want to go up to a more complex model | If accept Ho, that m(ind) and (mi) are the same, then would tend to go back down because the model cannot be justified, need simpler model  If reject Ho then you have said (mi) and m(ind) are different and try and go up further and get a more complex model |
| Type 1 error  If reject Ho incorrectly, then have said that (mi) is different from (mo ) when it is not  Have a model that is more complex than necessary and you could have gone down further to a simpler model | Type 1 error-MORE SERIOUS  If reject Ho incorrectly, then have said that (mi) is different than m(ind) when it is not  Have a model that is more complex than is statistically justified, have asserted interactions that are not there |
| Type II error-MORE SERIOUS  If accept Ho incorrectly then said that (mi) is the same as (mo ) when it is not  Have a model that is too simple and you need to go back up to a more complex model | Type II error  If accept Ho incorrectly then have said that (mi) is the same as m(ind) when it is not  Have chosen a model that is simpler and less predictive than necessary and you should have gone up higher |

### When developing a model

1. Start with a loopless model, this will help determine which variables may go into the model and then can look at looped models to evaluate ALL relations
2. TOP-DOWN: looking for acceptable simplification, compression of data with fewer df
3. BOTTOM-UP: trying to include as many statistically significant relations (constraints) as possible

* If go TOO far then have overfitting

## Algorithm that detects whether you have loops-Krippendorg p42

* Remove an variable unique to a relation (component) in the model
* Remove any redundant component and then go back up,
* if nothing left=NO LOOPS
* Example
  + ABC:ACD:BCE
  + Step 1: remove D, E
    - ABC:AC~~D~~:BC~~E~~
  + Step 2: remove AC, BC
    - ABC:~~AC~~:~~BC~~
  + Step 3: remove ABC
    - ~~ABC~~
  + Since there is nothing left, there are NO loops
* Example
  + AC:ACD:BCE:DE
  + Since there are NO unique variables and NO redundant components the model has LOOPS
* ANY directed system model with more than one predicting component has a LOOP
  + Predicting component means it has a DV and some IV’s

## Q→Transmission

Independence Model: qA:B(AB)= p(A)p(B)~ H(A)+H(B)

Disjoint Model: qAB::CD(ABCD)= p(AB)p(CD)~ H(AB) +H(CD)

Loopless Model: qAB::BC(ABC)= P(AB)p(BC)/p(B) ~ H(AB)+H(BC)-H(B)

Looped models: cannot do, closed form, MUST use IPF

## Degrees of Freedom-Krippendorff Method

* Calculate df for each component of the model
  + Model: ABC:BCD:CDE:DEF
    - df(ABC)
    - df(BCD)
    - df(CDE)
    - df(DEF)
    - where df(component)=⏐A⏐\*⏐B⏐\*⏐C⏐-1, where ⏐A⏐=number of states of A
* Sum df of the components
  + - df(ABC)+ df(BCD)+ df(CDE)+ df(DEF)
      * If have loops, subtract df of any overlap between components within the model
      * Identify overlaps
* List all pairs of components, triplets of components and Identify overlaps
  + Subtract all variables shared by two components
  + Add all variables shared by three components
  + –df(BC) –df(C)- df(CD) –df (D) -df(DE) + df(C) +df(D)

**therefore: df (ABC:BCD:CDE:DEF)**

**= df(ABC)+ df(BCD)+ df(CDE)+ df(DEF) –df(BC) –df(C)- df(CD) –df (D) -df(DE) + df(C) +df(D)**

|  |  |
| --- | --- |
| Pairs of components | Overlap |
| ABC:BCD | BC |
| ABC:CDE | C |
| ABC:DEF | None |
| BCD:CDE | CD |
| BCD:DEF | D |
| CDE:DEF | DE |
| Triples of components | Overlap |
| ABC:BCD:CDE | C |
| BCD:CDE:DEF | D |
| ABC:BCD:DEF | None |
| ABC:CDE:DEF | none |

## Degrees of Freedom-Log linear Methods Krippendorf pg 130-131

* Model: MER:MV:EV, where ⏐M⏐=⏐R⏐=⏐V⏐= 2 and ⏐E⏐= 3
  + df(MER) +df(MV) + df(EV)-df(M)-df(E)-df(V)
* (2\*3\*2-1) + (2\*2-1) + (3\*2-1) –(2-1) –(3-1)-(2-1)
* 11+3+5-1-2-1=15 degrees of freedom

### An alternative way is to write out call unique relations

* Model: MER:MV:EV, where ⏐M⏐=⏐R⏐=⏐V⏐= 2 and ⏐E⏐= 3
* **Write out all unique relations of the structure or embedded in the product**
* **Calculate the df for each component using the cardinality of each relation minus 1**
* **Sum all the df**
* MER (2-1)\*(3-1)\*(2-1)=2
* ME (2-1)\*(3-1)=2
* MR (2-1)\*(2-1) =1
* ER (3-1)\*(2-1)=2
* MV (2-1)\*(2-1) =1
* EV (3-1)\*(2-1) =2
* M (2-1)=1
* E (3-1)=2
* R (2-1) =1
* V (2-1) =1
* Add these all up and you end up with 15

**Note log linear method makes it easy to compare df of nested models**

## Decomposition of df

ABC data ⏐A⏐=⏐B⏐=⏐C⏐= 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | C0 | | C1 | |
|  | B0 | B1 | B0 | B1 |
| A0 | p1=.10 | p2=.11 | p3=.12 | p4=.13 |
| A1 | p5=.14 | p6=.15 | p7=.16 | P8=.09 |

* df=(2\*2\*2-1)=7
* H(ABC)=Γ(p1, p2, p3, p4, p5, p6, p7, p8)
* H(ABC)=.10+.11+.12+.13+.14+.15+.16+.09=1.0

How can we tell if we can decompose ABC→AB:BC or AB:AC or AC:BC

* Want the same information with decreased degrees of freedom

Process

* DataABC→ProjectionAB:BC→Calculated(q)ABC then compare back to DATA

First Step→Projection

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Projection AB | | Projection BC | | |
|  | B0 | B1 |  | B0 | B1 |
| A0 | p1+ p3 | p2+ p4 | C0 | p1+ p5 | p2+ p6 |
| A1 | p5+ p7 | p6+ p8 | C1 | p3 +p7 | p4+ p8 |

Second step→ replace with probabilities

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Projection AB | | Projection BC | | |
|  | B0 | B1 |  | B0 | B1 |
| A0 | .1+ .12=.22 | .11+.13=.24 | C0 | .10+.14=.24 | .11+.15=.26 |
| A1 | .14+.16=.30 | .15+.09=.24 | C1 | .12+.16=.28 | .13+.09=.22 |

Third step→ Add probabilities for the AB projection and the BC projection

AB=.22+.24+.30.24=1.0

BC=.24+.26+.28+.22=1.0

Fourth step→determine how many state probabilities do you need for AB:BC to be able to calculate ALL states probabilities

|  |  |  |
| --- | --- | --- |
|  | Projection AB | |
|  | B0 | B1 |
| A0 | .22 | .24 |
| A1 | .30 | X |

* All probabilities sum to 1.0
* If know 3 probabilities in table, can calculate the last one
* **So, need to know 3 probabilities to know/calculate all 4 probabilities for table AB**
* From AB table, know sum of all B0 state probabilities and sum of all B1 state probabilities
  + sum of all B0 state probabilities= A0B0+ A1B0=.22+.30=.52
  + sum of all B1 state probabilities= A0B1+ A1B1=.24+.24=.48
* If know probability for either C0B0 of C1B0, can calculate other one
  + C0B0=.24 (from BC projection)
  + Since we know all B0 state probabilities=.52 then C1B0=.52-.24=.28
  + C1B1=.22 (from BC projection)
  + Since we know all B1 state probabilities=.48 then C0B`=.48-.22=.26
* **So, need to know 2 probabilities to know/calculate all 4 probabilities for table BC**
* **Therefore, the df=3+2 or 5**

## Lattice of structures, ordinality

* Composition: going up=Constraint
* Decomposition: going down=Error
* 2 variables
  + AB RELATIONS (variables are connected)
  + A:B (variables are independent)
* 3 variables

General Specific

ABC 1 1

AB:AC:BC (this is a loop) 1 1

AB:AC AB:BC AC:BC 1 3

AB:C AC:B BC:A 1 3

A:B:C 1 1

Model has: 5 9

### Computational complexity

* Increases with the number of variables
* 4 variables
  + 20 general structures, 114 specific structures
* 5 variables
  + 180 general structures, 6894 specific structures

### Complexity of individual structures

* Degrees of freedom= # parameters
* Dependent on the cardinality of each variable
* #parameters (structure) depends on topology of structure-cardinality of variables
* Cardinality= # states the variable can be in
  + Cardinality of A=2, cardinality of B=3

|  |  |  |  |
| --- | --- | --- | --- |
|  | B1 | B2 | B3 |
| A1 | 10 | 15 | 3 |
| A2 | 5 | 8 | 11 |

## Generating a Specific Lattice of Structure for 3 Variables

1. Start with a model (set of relations)

**AB:AC:BC**

1. Remove a relation from model, list remaining model

**AB:AC:BC – AB =AB:BC**

**AB is the removed relation, AC:BC is the remaining model**

1. Add all embedded relations from removed relation back in, IF the are NOT already present in the remaining model

**AB removed relation, A and B are the embedded relations**

**AC:BC is the remaining model**

**A and B are both present**

**DO NOT have to add in any relations**

1. Repeat for original model, removing a different relation (ie going down a different pathway)

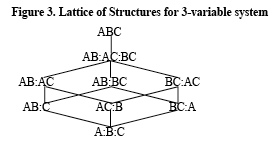
**This time could remove AC**

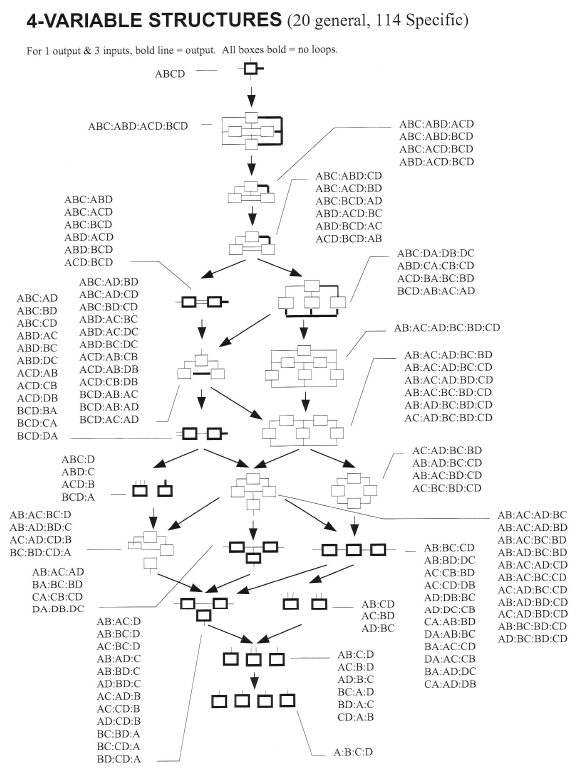
**AB:AC:BC-AC= AB:BC**

**AC removed from relation, A and C are embedded relations**

**A and C are both present**

**Do NOT have to add in any relations**





## Generating a Specific Lattice of Structure for 4 Variables

1. Start with a model (set of relations)

**ABCD**

1. Remove a relation from model, list remaining model

**ABCD – ABCD**

1. Add all embedded relations from removed relation back in, IF the are NOT already present in the remaining model

**ABCD is the removed relation**

**Need to add in ABC +ABD+ ACD +BCD**

**ABC:ABD:ACD:BCD is the remaining model**

1. Repeat removing a different relation (ie continue going down)

**This time could remove BCD**

**ABC:ABD:ACD:BCD –BCD**

**BCD removed from relation, BC +BD + CD are embedded relations**

**BC, BD and CD are all present in ABC:ABD:ACD**

**Do NOT have to add in any relations**

1. Repeat removing a different relation (ie continue going down)

**This time could remove ABC**

**ABC:ABD:ACD-ABC**

**BCD removed from relation, AC +AB + BC are embedded relations**

**AC and AB are all present in ABD:ACD**

**Need to add back in BC, leaving ABD:ACD:BC**

1. Repeat removing a different relation (ie continue going down)

**This time could remove BC, ACD or ABD**

**If removed BC**

**ABD:ACD:BC-BC**

**BC removed from relation, B and C are embedded relations**

**Neither B or C is present in ABD:ACD**

**Need to add back in B and C, leaving ABD:ACD:B:C**

**OR**

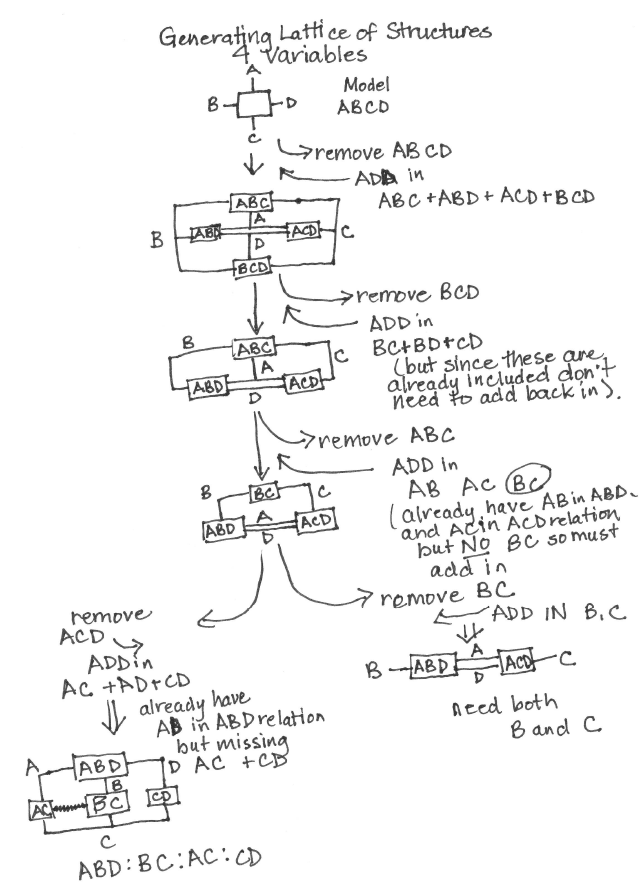
**If removed ACD**

**ABD:ACD:BC-ACD**

**ACD removed from relation, AC, AD and CD are embedded relations**

**Neither AC or CD is present in ABD:BC**

**Need to add back in AC and CD, leaving ABD:BC:AC:CD**



## 

### Nearest ancestor and descendent

* Mi=AC:BCDE Mj=ABD:CD:CE
* Descendent=what is in common with the parent or descendent, INTERSECTION
* AC:BCDE ∩ ABD:CD:CE
* Ancestor=what is common in both
* AC:BCDE ∪ ABD:CD:CE = AC:BCDE:ABD

## Set-theoretic relation

* Subset of Cartesian product (x)
  + Where (x)=all possible combinations of elements in a set
  + The set of states actually observed without regard to frequency
  + Tuples are NOT independent, knowing something about one can tell you something about the other
  + A relation is a constraint (reduction from possible to actual)
  + If possible states = Observed States, then system is a HEAP
  + Strength of the constraint is the difference in uncertainties between the “Heap” AxB and the system AB
    - Strength of the relation (possible)=log2⏐AxB⏐

(actual) ⏐AB⏐

### Example:

A=(r,y,g) B=(r,y,g)

AxB=rr, ry,rg,yr,yy,yg,gr,gy,gg Cardinality =9 nine tuples

AB=rr,ry,rg,yr,gr Cardinality=5

Uncertainty

* H(A⊗B)-H(AB) = amount of constraint in subset relative to the Cartesian product

Constraint (T)

* Loss of constraint or amount of error, difference between uncertainty of sets

H (A⊗B) =Hartley entropy or the amount of uncertainty, A⊗B=HEAP of A:B and AB (system)

= log2⏐A⊗B⏐- log2⏐AB⏐

= log29- log25

## Set Theoretic Reconstructability Analysis (SRA)

There are 2 ways to figure out the tuples allowed for a model

1. ABC(model)=all tuples allowed by projections specified by the model=’maximum entropy’
   1. Takes all tuples except those that are forbidden by the relation

ABC

AB:AC:BC

AB:AC AB:BC AC:BC

AB:C AC:B BC:A

A:B:C

This shows that there are 5 states

ABC A:B:C=A⏐2⏐, B⏐2⏐, C⏐2⏐

= A⊗B⊗C

=2x2x2 =8

Within the specific set ABC, we look at AB, AC and BC. Where the “.” is “don’t care”. The initial number of tuples is 5, since there are 5 separate states for ABC

ABC AB. A.C .BC ABC

000 00. 0.0 .00 000

010 01. 0.0 .10 001 b

011 01. 0.1 .11 010

110 11. 1.0 .11 011

111 not 10. (a) 1.1 not .01 (b) 100 a

101 a,b

110

111

Since we are left with the same 5 states as ABC, since you have everything in AC covered with AB and BC then, you can decompose this structure to AB:BC with no loss of information

1. AB⊗C ∩ BC⊗A ABC

~~000~~ ~~000~~ 000

001 100 010

~~010~~ ~~010~~ 011

~~011~~ ~~110~~ 110

~~110~~ ~~011~~ 111

~~111~~ ~~111~~

Here you create tuples compatible with AB and then BC. So for AB you try all combinations of C and for BC you try all combinations of C, all tuples on both lists are compatible with the AB:BC relation

T (A:B:C)=H(A:B:C model)-H (ABC data)

T(A:B:C)= log2|A⊗B⊗C|- log2|ABC|

T(A:B:C)=log2 8/5

T= the amount of constraint that is lost or the amount of error in the model.

# Information Reconstructability Analysis (IRA)

* Model complexity defined as the number of parameters needed to specify the model
* Type of complexity is the number of components in the model

## Information Theory

* Measure of the syntactic information
* Doesn’t measure meaning (semantic) or value (pragmatic)
* Provides a yardstick for measuring organization
* About constraint, order, organization, association and relation

### What is Information?

* Information is the opposite of uncertainty
* Information in message=Hinitial-Hfinal= -ΔH
* If Hfinal is equal to 0, then the amount of information in the message is Hinitial
* When all pj (probabilities) are equal (1/n), then H=log2n, log22=1, where 2 is the number of variables

## Shannon Entropy

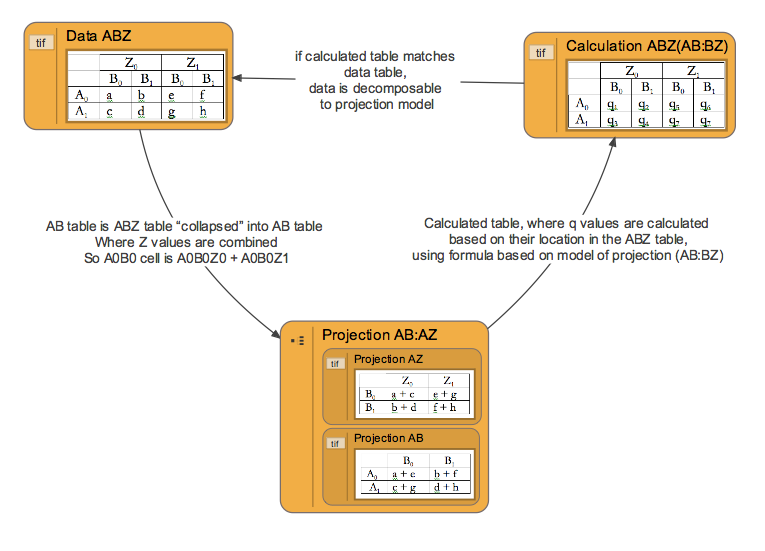
* Univariate uncertainty
* H(x)= -Σp(xj) log2p(xj), where p=probability
* Σpj log21/pj, where pj is the expected or average surprise and log21/pj is the surprise in the outcome j on a log scale
* Defines the amount of surprise in the outcome
* When pj is low, then one is surprised if this occurs, ie .01, happens 1 out 100x
* When pj is high, .99, then one is not surprised at the outcome
* H increases as J increases and the probabilities are more uniform and have more possible values
* **Cannot** be used if you have possibility

For 2 states H and T,

* H maximizes at ½ (50:50)
* H is a measure of diversity

# IPF

## Data 🡪 Projection 🡪 Calculation 🡪 Data



## Data (ABZ)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Z0 | | Z1 | |
|  | B0 | B1 | B0 | B1 |
| A0 | a | b | e | f |
| A1 | c | d | g | h |

## Projection (AB:BZ)

|  |  |  |
| --- | --- | --- |
|  | B0 | B1 |
| A0 | a + e | b + f |
| A1 | c + g | d + h |

AB table is ABZ table “collapsed” into AB table

Where Z values are combined

So A0B0 cell is A0B0Z0 + A0B0Z1

|  |  |  |
| --- | --- | --- |
|  | Z0 | Z1 |
| B0 | a + c | e + g |
| B1 | b + d | f + h |

|  |  |
| --- | --- |
| B0 | a + c + e + g |
| B1 | b + d + f + h |

## Calculation (ABZAB:AZ) – each q value will be replaced with actual values in terms of parameters a through h

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Z0 | | Z1 | |
|  | B0 | B1 | B0 | B1 |
| A0 | q1 | q2 | q5 | q6 |
| A1 | q3 | q4 | q7 | q7 |

Calculated table, where qvalues are calculated based on their location in the ABZ table, using formula based on model of projection (AB:BZ)

For LOOPLESS models, the calculated table can be determined algebraically, but for LOOPED models, another method, Iterative Proportional Fitting, must be used.

### Calculated table formulas:

For calculated value, depends on (projection) model. For model AB:BZ:

qspecific location= [p(AB)same locationp(BZ)same location]/p(B)same location

Where “location” is specified by variable states, so q1 for example is at “location” A0B0Z0 in the ABZ table,

so p(AB) for q1 is the value in the cell A0B0Z0+1 in the ABZ table, or value in cell A0B0 in the projected table AB

p(BZ) for q1 is the value in the cell A0+1B0Z0 in the ABZ table, or value in cell B0Z0 in the projected table BZ

and p(B) for q1 is the value in the cell A0+1B0Z0+1 in the ABZ table, or value in cell B0 in the projected table B

### q1 for ABZAB:AZ = values for A0B0Z0 for each of the probabilities:

**[p(AB)A0B0Z0p(BZ)A0B0Z0]/p(B)A0B0Z0**

p(AB)A0B0Z0 for q1 = a + e

p(BZ)A0B0Z0 for q1 = a + c

p(B)A0B0Z0 for q1 = a + c + e + g

so, q1 for ABZAB:AZ = [(a + e)(a+ c)]/(a + c + e + g)

then repeat for each value for q (e.g. q2, q3, . . .q6) to complete the calculated table

## IPF (Iterative Proportional Fitting) – method for generating calculated table (Calculation) for LOOPED models

### Steps to IPF

1. Make a version of the calculated table that has uniform distribution (quniform table)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Z0 | | Z1 | |
|  | B0 | B1 | B0 | B1 |
| A0 | 1/8 | 1/8 | 1/8 | 1/8 |
| A1 | 1/8 | 1/8 | 1/8 | 1/8 |

1. Choose any individual q value (e.g. q1 or q7) in the calculated table (choose any cell) OR just decide to calculate the whole table (to process for each q1 value at the same time)

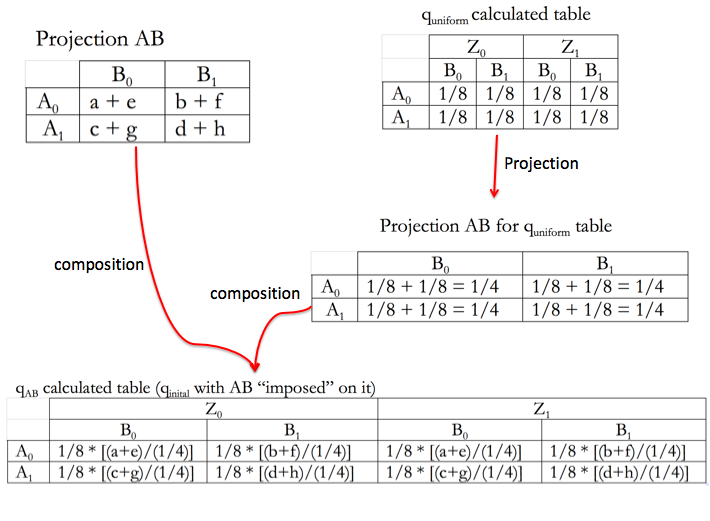
qinitial table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Z0 | | Z1 | |
|  | B0 | B1 | B0 | B1 |
| A0 | q1 | q2 | q5 | q6 |
| A1 | q3 | q4 | q7 | q7 |

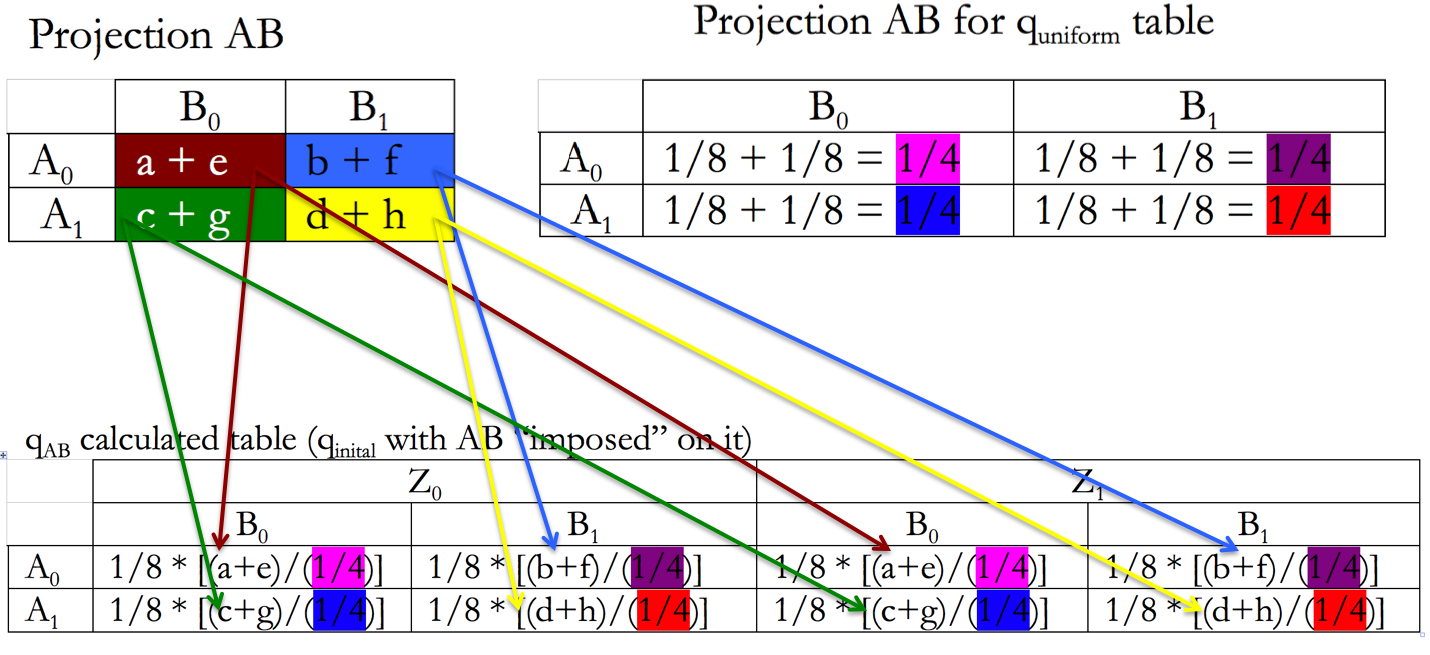
1. See what marginal value(s) the chosen q value contributes to in the relation being imposed = See what value’s “location” is, state of each variable for specific q value (e.g. q2 is at A0B1Z0)
2. Multiply value of chosen cell (or each cell in turn) in (quniform table) by: [(desired marginal value)/calculated marginal value)]
   1. Desired marginal value = specific marginal value for specific projection of data table (initially just choose one projection table, next time do the other) for specific q value = (for q3, which is at A1B0Z0, which is A1B0 for AB projection, or B0Z0 for BZ projection)
   2. Calculated marginal value = specific marginal value for projection of current calculated table (initially quniform) for specific q value = (for q3, which is A1B0Z0, project from current calculated table and get value from A1B0 cell from AB table, and B0Z0 cell from BZ table.
3. Repeat process, using new calculated table, and “impose” the other data projection (e.g. if you imposed the AB table the first time, now impose the BZ table).

### IPF – Impose AB

Take quniform, choose Projection table (from data projections) to “impose”, make projection of quniform with same variables as chosen projection table of data, take data projection and calculated projection and “impose” onto calculated table (quniform)



### IPF – How to impose AB



### IFP – Impose BZ – How to do projection of BZ from qA:IPF:Projection BZ of qAB process.tiff

### IPF – simplify Projection BZ for qAB table

B0Z0= 1/8 \* [(a+e)/(1/4)] + 1/8 \* [(c+g)/(1/4) = ((a+e)/2) + ((c+g)/2) = (a+e+c+g)/2

Projection BZ for qAB table

|  |  |  |
| --- | --- | --- |
|  |  |  |
| B0 | (a+e+c+g)/2 | (a+e+c+g)/2 |
| B1 | (b+f+d+h)/2 | (b+f+d+h)/2 |
|  | Z0 | Z1 |

### IPF – Impose BZ onto qAB

### :IPF:qAB-BZ w BZ imposed simple process.tiff

### IPF – qAB:BZ calculated table – simplified more

Simplified more

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Z0 | | Z1 | |
|  | B0 | B1 | B0 | B1 |
| A0 |  |  |  |  |
| A1 |  |  |  |  |

### IPF – now do it again! For a looped model, you’d need to repeat: impose AB, then impose BZ, over and over again until your calculated table doesn’t change. But this example is loopless, so no need to repeat.

# State-Based RA

## Information-theoretic variable-based RA

* + Can detect and quantify relationships among variables for contingency tables
  + In reconstruction, a distribution is decomposed (compressed, simplified) into component distributions
  + Joint distribution associated with a model has the maximum possible information-theoretic uncertainty (Shannon entropy) subject to the constraints of its component projections with corresponding projections of the data
  + Quality of the approximation can be assess with respect to both the information it retains (relative to the data) and its complexity
  + The information (constraint) present in the data is the information-theoretic transmission between p (joint distribution of the data) and q(joint distribution of the model, where transmission is defined as
    - T = Σ p log (p/qmodel)
* Complexity of a model in VBRA is defined as its degrees of freedom (df)

### State-Based vs Variable Based

|  |  |
| --- | --- |
| **State-Based** | **Variable-Based** |
| STATE-based is MORE GENERAL in that the set of state-based models for a given system contains all possible variable-based models, therefore more state-based models |  |
| Best state models ALWAYS retain at least as much information as VB with NO ↑in model complexity |  |
| Constraint in the DV is explained in terms of SPECIFIC STATES of subsets of the IV’ | Constraint in the DV is explained in terms of all the states of the IV (main effects of the individual IVs and interactions among them) |
| Lattice of structures enormously greater posing a problem of how to search the lattice |  |
| Do NOT specify complete marginal distributions (projections), they correspond to any linearly independent set of individual elements (cells) of the joint distribution or any of the marginal distributions | Parameters are values of complete marginal distributions (q distribution) that will be constrained to match the corresponding marginal distributions derived from the data (p distribution) |
| Can be categorized with respect to degrees of freedom required for specification | Can be categorized with respect to degrees of freedom required for specification |
| Since state-based structures are less constrained than the variable based independence model A:B then A:B is NOT the bottom of the lattice of structures (not limited to uniform distribution) | A:B (independence model) is the bottom of the lattice |
| Represent behavior systems more accurately and more parsimoniously that VBRA |  |

## State Based Specifics

* Powerful and broadly applicable generalization of established VB modeling
* State-based modeling can also be used to explore the analysis of event and decision trees, providing an alternative to the standard variable-based sensitivity methods
* State-based modeling provides insight into which outcomes or combinations of outcomes lead to a favorable or unfavorable utility (ie. decision-maker wants to know which outcomes or combinations of outcomes are most responsible for the variability in the utility)
* In state-based the relationship between the DVs and the IVs is probabilistic so that the system of interest can be defined in terms of a contingency table (joint probability distribution)
* **State-based RA parameters DO NOT specify complete marginal distributions (projections), rather they correspond to any linearly independent set of individual elements (cells) of the joint distribution or any of the marginal distributions.**

DISADVANTAGES:

* Fourier coefficients are not as interpretable as a distribution and projection (therefore, Fourier constitutes and alternative to state-based)

ADVANTAGES:

* State-based models RETAIN more information while simultaneously reducing model complexity (greater compression)
* Promising application is the simplification of event and decision tree models

### General Concepts of State-Based Models

* Bush Jones linked the concept of state-based modeling idea to the concept of going from a “g to K” transformation
* “System” is what Klir terms a behavior system- a contingency table which assigns frequencies or probabilities to system states
* A directed system is one in which each variable is distinguished as being either an “independent variable (IV)” or a “dependent variable (DV)
* In state-based the IV’s define the system state and the DV’s depend upon this state
* IV’s qualitative (catergorical or ordinal) IV’s with one or more qualitative DV’s.
* Binning does sacrifice some information of the original quantitative variable
* State based is one aspect of the k-systems and encompasses two concepts
  + An RA model need not be defined in terms of univariate or multivariate projections of the data but can instead be identified by specifying the set of system states for which the model frequencies or probabilities are constrained to match the data
    - Only requirement is that the constraints imposed on the set of system
  + Second concept is that any function that maps the system states defined by the IV’s goes onto a finite section of the real line can be transformed, using a “g-to-k” transformation into a function with real values in the range (0,1) that sum to 1
* These combinations of IV’s (levels) viewed as events that are associated with the DV outcomes
* Data comprise a frequency distribution in the form of counts (observations at each point in the discrete domain defined by the levels of the independent and the dependent variables)
* When normalized to the total number of observations, these counts can be interpreted as a joint probability distribution
* ALL state models must include both IVs and DV components- this forces those states which are not explicitly constrained in the state-based model to be maximally consistent with the data
* **Behavior of a state based models encompasses two ideas: given a p distribution and a candidate structure S, the q distribution is constrained to:**
  + **Maximize information-theoretic uncertainty**
  + **All elements of the q distribution will be greater than or equal to zero**
* IPF is used to maximize the information-theoretic uncertainty of the joint distribution, subject to the imposed constraints
* Projections of the model distribution must be consistent with the corresponding marginal projections observed in the data AND the distribution over DV conditioned on the state constrained to match the observed conditional distribution for that state
* This has the effect that in the model the conditional distributions on Z for all states other than the one of interest are as similar as possible subject to the satisfaction of the AB and Z constraints