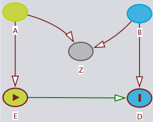
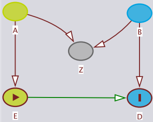
Test your DAGs!

Leeds, May 8th, 2019

Johannes Textor
Tumor Immunology, Radboudumc
Data Science, Radboud University

Nijmegen, The Netherlands





Acknowledgements

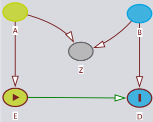
UNIVERSITÄT ZU LÜBECK
Juliane Hardt
Benito van der Zander
Maciej Liskiewicz

ETH ZÜRICH
Ema Perkovic
Markus Kalisch
Marloes Maathuis

UNIVERSITY OF LEEDS
Peter Tennant
Mark Gilthorpe
George Ellison

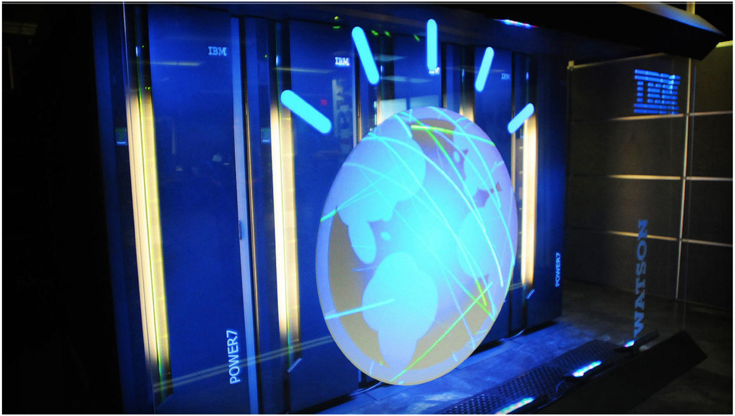
DIFE POTSDAM
Sven Knüppel

FUNDING
KWF Kankerbestrijding

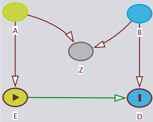


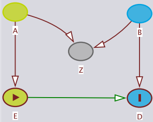
Why should I care about DAGs?

Supercomputer Watson takes on cancer care with Memorial Sloan-Kettering



Caption: This Jan. 13, 2011 photo provided by IBM shows the IBM computer system known as Watson at IBM's T.J. Watson research center in Yorktown Heights, N.Y. Watson, best known for handily defeating the world's best "Jeopardy!" players on TV earlier this year, is on a diet of medical textbooks and journals for health care. IBM says Watson, with its ability to understand plain language, can digest questions about a person's symptoms and medical history and quickly suggest diagnoses and treatments. (AP Photo/IBM) / **AP**





EDITOR'S PICK | 209,668 views | Feb 19, 2017, 03:48pm

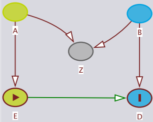
MD Anderson Benches IBM Watson In Setback For Artificial Intelligence In Medicine

Matthew Herper Forbes Staff
Pharma & Healthcare
I cover science and medicine, and believe this is biology's century.

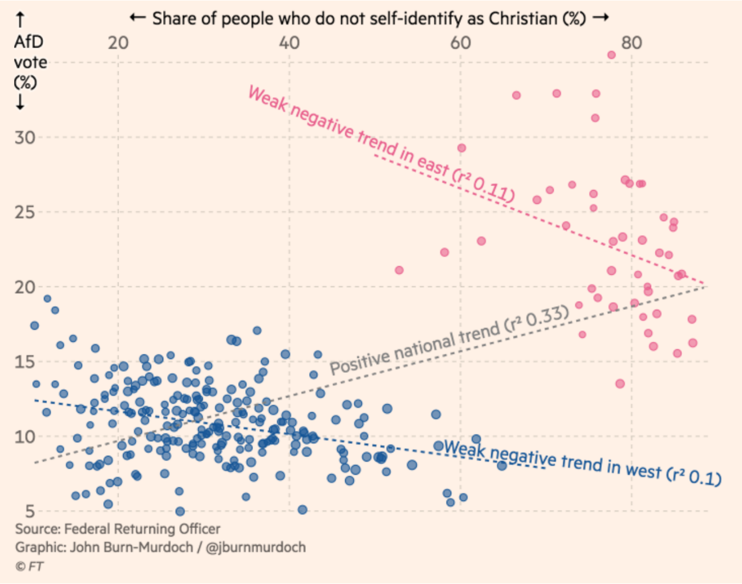
EXCLUSIVE

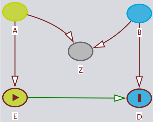
IBM's Watson supercomputer recommended 'unsafe and incorrect' cancer treatments, internal documents show

By CASEY ROSS @caseymross *and* IKE SWETLITZ @ikeswetlitz / JULY 25, 2018



Are Religious People More Right-Wing?





Simpson's Paradox

Suppose a new treatement for a disease is tested in a trial with the following results:

	Cured	Not Cured
Treated	20	20
Not Treated	16	24

$P(C = 1 \mid T = 1) = 0.5$

$P(C = 1 \mid T = 0) = 0.4$

Now the investigator wants to know whether the treatment is more effective in men or women, and gets the following results:

Males	Cured	Not Cured
Treated	18	12
Not Treated	7	3

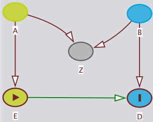
Females	Cured	Not Cured
Treated	2	8
Not Treated	9	21

$P(C = 1 \mid T = 1, S = m) = 0.6$
$P(C = 1 \mid T = 0, S = m) = 0.7$

$P(C = 1 \mid T = 1, S = f) = 0.2$
$P(C = 1 \mid T = 0, S = f) = 0.3$

Do we give the treatment or not?

Judea Pearl:
Simpson's Paradox: An Anatomy
http://bayes.cs.ucla.edu/R264.pdf



Simpson's Paradox

Suppose a new treatement for a disease is tested in a trial with the following results:

	Cured	Not Cured
Treated	20	20
Not Treated	16	24

$P(C = 1 \mid T = 1) = 0.5$

$P(C = 1 \mid T = 0) = 0.4$

The investigator knows that treatment affects blood pressure, and measures it after treatment. She gets the following results:

High BP	Cured	Not Cured
Treated	18	12
Not Treated	7	3

Low BP	Cured	Not Cured
Treated	2	8
Not Treated	9	21

$P(C = 1 \mid T = 1, S = m) = 0.6$
$P(C = 1 \mid T = 0, S = m) = 0.7$

$P(C = 1 \mid T = 1, S = f) = 0.2$
$P(C = 1 \mid T = 0, S = f) = 0.3$

Do we give the treatment or not? Given that these are exactly the same numbers as on the previous slide, must the answer be the same too?

Using DAGs to Resolve the Paradox

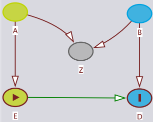
$$\begin{tikzcd} & \text{m/f} \arrow[dl] \arrow[dr] & \\ \text{treatment} \arrow[rr] & & \text{cure} \end{tikzcd}$$

Gender is a confounder of the effect of treatment on cure. Its influence should be removed by conditioning on it. Therefore, given our data, we would not give the treatment.

$$\begin{tikzcd} & \text{blood pressure} \arrow[dr] & \\ \text{treatment} \arrow[ur] \arrow[rr] & & \text{cure} \end{tikzcd}$$

Blood pressure is a mediator of the relationship between treatment and cure. If we were to condition on it, this might obscure or reverse the effect of treatment on cure. Therefore, given our data, we would give the treatment.

You are smarter than your data. Data do not understand causes and effect; humans do.
– Judea Pearl, "Book of Why"



Usage Patterns of DAGs

Theoretical

Use DAGs as a tool in the development of causal inference methodology.

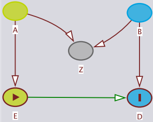
Positively Practical

Draw DAG that accurately represents the data-generating process in a target population, and use that for causal inference.

Negatively Practical

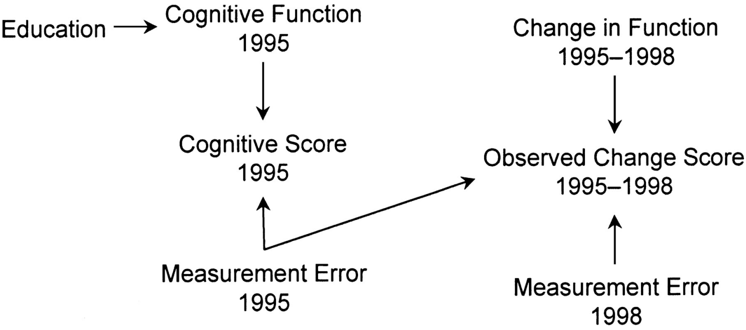
Draw DAG to conceptualize or illustrate a bias that could affect a certain analyis.

It should be no suprise that positive usage of DAGs is much harder than negative usage.

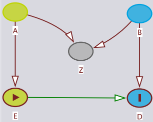


Example of Negatively Practical Usage

Conditioning on the baseline in analysis-of-change can lead to bias in the presence of measurement error.

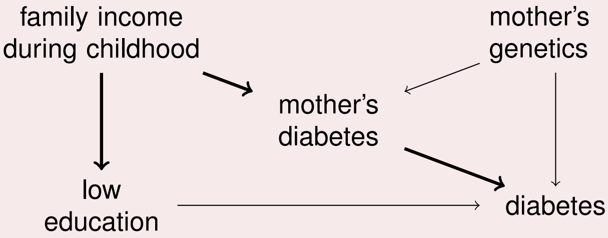


M Glymour et al., Am J Epidemiol, 2005; doi: 10.1093/aje/kwi187



Example of Positively Practical Usage

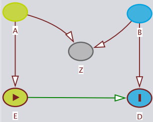
By how much does low education increase diabetes risk?

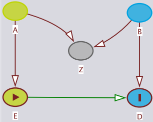


Rothman, Greenland & Lash, Modern Epidemiology, 2008

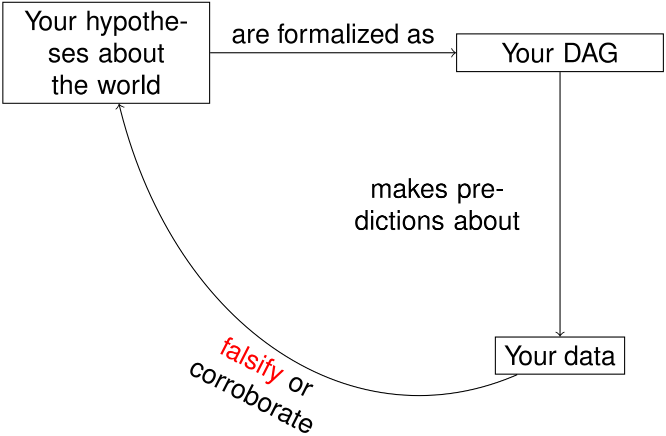
We should condition on family income, but not mother's diabetes, when estimating the effect of low education on diabetes risk.

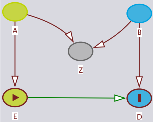
This DAG has drawn intensive criticism (e.g. George Davey-Smith)





DAG Modeling Cycle





- Negatively practical DAG usage is “one-shot” and is used to question, not build, substantive theory.
- Positively practical DAG usage requires iterations of model building, falsification, and refinement.

Motivation

- DAGs are models of variable relationships in a certain domain.
- (Sparse) DAGs net models encode certain assumptions about these relationships.
- Incorrect assumptions may lead to incorrect inferences.
- Once a DAG is constructed, we can test some of the assumptions it encodes against data.

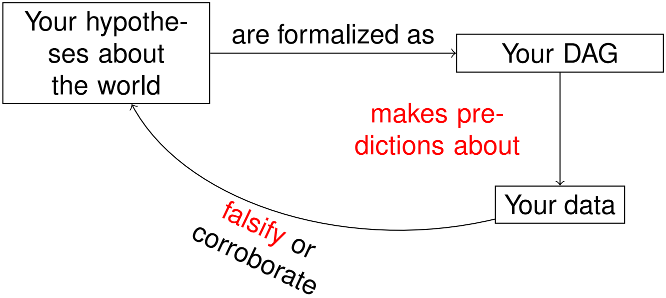
No free lunch!

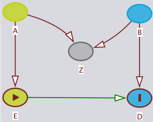
Model testing never guarantees a correct model! It can only refute, but never prove it.

Popper: only falsifiable theories are scientific.

Moving On

- Hypothesis: DAG users in Epidemiology are stuck at the first stage of the modelling cycle.
- The next stage requires attempting to falsify DAGs and produce less bad DAGs in the process.
- I don't know if this will ever be done, but I'll spend the rest of this talk showing how it can be done.





Types of Testable Implications

For DAGs without latent variables:

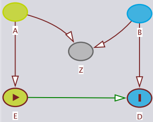
- Conditional Independence

For DAGs with latent variables:

- Instrumentality Test

For linear DAGs with latent variables:

- Vanishing Tetrad Constraints
- Conditional Independence



d-Separation

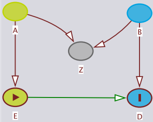
A path is a sequence of variables connected by arrows. Moving against arrow directions is allowed – e.g.
$Z \leftrightarrow A \rightarrow E \leftarrow Z \rightarrow D$.

- A collider is a path of length 3 that looks like $X \rightarrow M \leftarrow Y$.
- All other paths of length 3 are called non-colliders: $X \rightarrow M \rightarrow Z$, $X \leftarrow M \leftarrow Y$, $X \leftarrow M \rightarrow Z$

d-separation

A set $\mathbf{Z}$ of variables (possibly empty!) d-separates a path, if

- The midpoint M of some non-collider is in $\mathbf{Z}$; or
- The midpoint M of some collider is not an ancestor of any variable in Z.

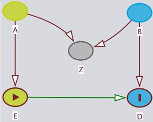


d-Separation By Example

Path: $X \to Y$

There is no collider or non-collider on this path, since it is too short. So it is not d-separated, no matter what $\mathbf{Z}$ is.

- X and Y are not d-separated (by $\mathbf{Z} = \{\}$).
- X and Y are not d-separated by $\mathbf{Z} = \{Z\}$

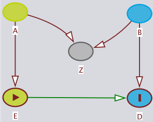


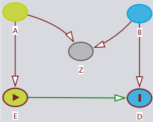
d-Separation By Example

Path: $X \to Z \to Y$

This path is a non-collider.

- X and Y are not d-separated (by $\mathbf{Z} = \{\}$).
- X and Y are d-separated by $\mathbf{Z} = \{Z\}$





d-Separation By Example

Path: $X \rightarrow Z \leftarrow Y$

This path is a collider.

- The path is not d-separated using $\mathbf{Z} = \{Z\}$.
- The path is d-separated (by $\mathbf{Z} = \{\}$).

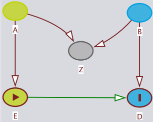
d-Separation By Example

Path: $X \rightarrow Z \rightarrow W \leftarrow Y$

We check d-separation on this path by checking each 3-variable sub-path independently. The path is d-separated if

- $X \rightarrow Z \rightarrow W$ is d-separated; or
- $Z \rightarrow W \leftarrow Y$ is d-separated; or
- both are d-separated.

- The path is d-separated (by $\mathbf{Z} = \{\}$).
- The path is d-separated by $\mathbf{Z} = \{Z\}$.
- The path is not d-separated by $\mathbf{Z} = \{W\}$.

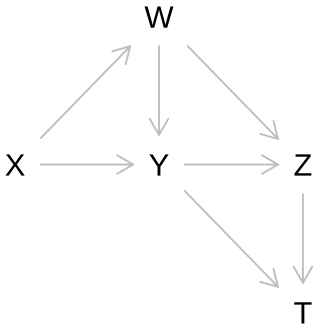


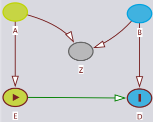
Paths in R

For detailed inspection of paths and their statuses, dagitty provides the function ‘paths‘. This function returns a list with two components, ‘paths‘ and ‘open‘ (=not d-separated).

```r
paths( g, "Y", "Z" )$paths
```

```
## Y -> T <- Z
## Y -> Z
## Y <- W -> Z
## Y <- X -> W -> Z
```

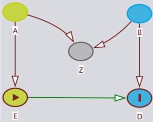




d-Separation and Conditional Independence

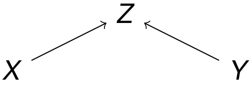
Theorem (Verma & Pearl, 1984)

If all paths between X and Y are d-separated by the set $\mathbf{Z}$, then $X \perp\!\!\!\perp Y \mid \mathbf{Z}$ in every distribution that can possibly be generated by the DAG.



Example 1: Testing a Collider Model

Let us simulate data in R that follows the collider DAG:

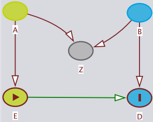



```r
set.seed(123)
# Number of samples to be generated
n <- 10000
# X and Y are simple coin tosses
# P(X=1)=P(Y=1)=0.5
X <- 2*rbinom(n,1,p=.5)-1
Y <- 2*rbinom(n,1,p=.5)-1
# Z is a variable that depends
# on both X and Y.
# P(Z=1) = e^(X+Y)/(e^(X+Y)+1)
Z <- 2*rbinom(n,1,
  p=exp(X+Y)/(exp(X+Y)+1))-1
```

```r
# continued
as.data.frame(
  table(X,Y,Z))

##    X  Y  Z Freq
## 1 -1 -1 -1 2263
## 2  1 -1 -1 1298
## 3 -1  1 -1 1256
## 4  1  1 -1  272
## 5 -1 -1  1  295
## 6  1 -1  1 1242
## 7 -1  1  1 1243
## 8  1  1  1 2131
```

Or, simply:

```r
simulateLogistic("dag{X->Y<-Z}",1)
```

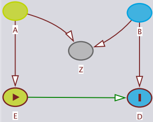



Example 1: Testing a Collider Model

Using *d*-separation, we can derive one implication from our collider model: $X \perp\!\!\!\perp Y$.

```
paths( "dag{ X -> Z <- Y }", "X", "Y" )

## $paths
## [1] "X -> Z <- Y"
##
## $open
## [1] FALSE

impliedConditionalIndependencies( "dag{ X -> Z <- Y }" )

## X _||_ Y
```

Assume we have categorical data, then we can test this implication using a standard chi-square test.



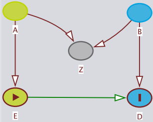
Example 1: Performing the Chi-Square Test

```
chisq.test(X,Y)

##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  X and Y
## X-squared = 0.61238, df = 1, p-value = 0.4339
```

The resulting p-value does not provide strong evidence against independence of X and Y. In other words, our data do not provide strong evidence against our model.

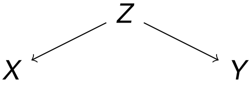
Philosophical digression

If you hate p-values, consider this: the test we performed was a significance test, but the hypothesiswe tested was not a null hypothesis – this was a direct model test!



Example 2: Testing a Fork Model

Now let us simulate some continuous data that follows the fork structure:

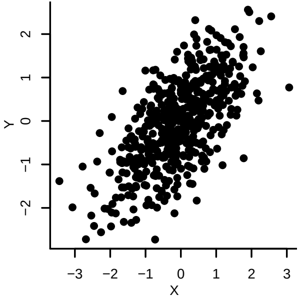


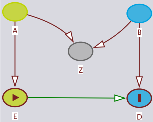

```r
d <- simulateSEM("dag{X <- Z -> Y}", b.default=.8)
```

The fork model does not imply $X \perp\!\!\!\perp Y$, because X and Y are d-connected. So what will happen if we test $X \perp\!\!\!\perp Y$? For continuous data, we could do that using a simple correlation.

```r
cor.test(X,Y)$p.value

## [1] 2.676702e-64
```

Our data and our DAG are not consistent.





Example 2: Testing a Fork Model

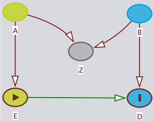
Our fork model $X \leftarrow Z \rightarrow Y$ has a conditional independence implication: $X \perp\!\!\!\perp Y \mid Z$.
Most statistical procedures for testing conditional independence fall in one of these categories:

Regression

Regress both X and Y on Z, and test independence of the residuals.

Stratification

Perform separate independence tests for each value of Z and combine the results.



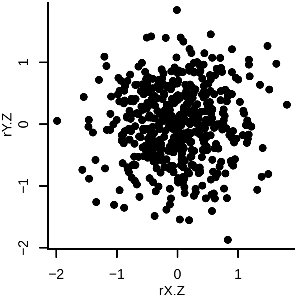
The Regression Strategy

Our fork model $X \leftarrow Z \rightarrow Y$ has a conditional independence implication: $X \perp\!\!\!\perp Y \mid Z$.

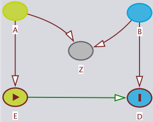
Regression

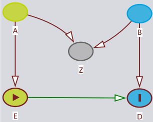
Regress both X and Y on Z, and test independence of the residuals.

```
rX.Z <- resid( lm( X ~ Z ) )
rY.Z <- resid( lm( Y ~ Z ) )
cor( rX.Z, rY.Z )

## [1] 0.04679324
```



This finds no evidence against our model.





The Stratification Strategy

Our fork model $X \leftarrow Z \rightarrow Y$ has a conditional independence implication: $X \perp\!\!\!\perp Y \mid Z$.

Stratification

Perform separate independence tests for each value of Z and combine the results.

```
T.Z0 <- chisq.test( X[Z==0], Y[Z==0] )
T.Z1 <- chisq.test( X[Z==1], Y[Z==1] )
T.Z0

##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  X[Z == 0] and Y[Z == 0]
## X-squared = 0.45078, df = 1, p-value = 0.502

T.Z1

##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  X[Z == 1] and Y[Z == 1]
## X-squared = 0.040947, df = 1, p-value = 0.8396
```

Example 2: A Single Conditional Independence Test

Making decisions based on multiple statistical tests is hard. It is more convenient to combine the results for the different levels of Z into a single test, especially when Z has many levels.

For chi-square tests, when $T_0 \sim \chi^2(a)$ and $T_1 \sim \chi^2(b)$ are two chi-square distributed variables with a and b degrees of freedom, then

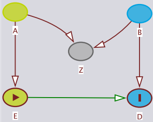
$$T_1 + T_2 \sim \chi^2(a + b)$$

So we can combine the results as follows:

```
chisq.combined <- T.Z0$statistic + T.Z1$statistic
df.combined <- T.Z0$parameter + T.Z1$parameter
1-pchisq(chisq.combined,df.combined)
```

```
## [1] 0.7820309
```

There is no strong evidence agaist $X \perp\!\!\!\perp Y \mid Z$.



A More Complex Example

In larger DAGs, there will be multiple implications we can test. Such local tests can point us to the part of the DAG where a problem lies.
In the example DAG below, we fail to take into account measurement error at U_1. How could we detect such a mistake?

$$
\begin{tikzcd}
 & M_1 \arrow[dr] & \\
X \arrow[ur] \arrow[dr] & & Y \\
 & M_2 \arrow[ur] &
\end{tikzcd}
$$

assumed model

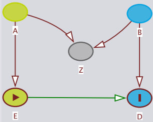
$M_1 \perp\!\!\!\perp M_2 \mid X$

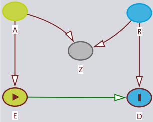
$X \perp\!\!\!\perp Y \mid \{M_1, M_2\}$

$$
\begin{tikzcd}
 & & M_1 \\
 & U_1 \arrow[ur] \arrow[dr] & \\
X \arrow[ur] \arrow[dr] & & Y \\
 & M_2 \arrow[ur] &
\end{tikzcd}
$$

true model

$M_1 \perp\!\!\!\perp M_2 \mid X$

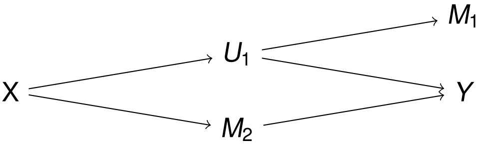




A More Complex Example

Let us generate some data from the “true” DAG.

```r
set.seed(123)
d<-simulateLogistic("dag{ X -> {U1 M2} -> Y ; U1 -> M1 }",1,N=1000)
```

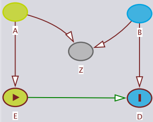


Testing the Assumed Model (I)

Let us start with the first implied independence: $M_1 \perp\!\!\!\perp M_2 \mid X$.

```
chisq <- 0 ; df <- 0
for( x in unique(X) ){
        tst <- chisq.test( M1[X==x], M2[X==x] )
        chisq <- chisq + tst$statistic
        df <- df + tst$parameter
}
chisq

## X-squared
## 0.3632495

df

## df
##  2
```

There is no strong evidence against dependence here.

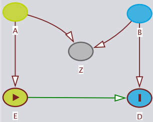


Testing the Assumed Model (II)

Let's now test the second implied independence: $X \perp\!\!\!\perp Y \mid \{M1, M2\}$. We do this by running separate chi-square tests for $X \perp\!\!\!\perp Y$ for each combination of M_1 and M_2.

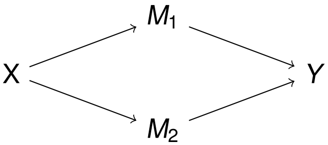
```
chisq <- 0 ; df <- 0
for( m1 in unique(M1) ){
        for( m2 in unique(M2) ){
                tst <- chisq.test( X[M1==m1 & M2==m2],
                        Y[M1==m1 & M2==m2] )
                chisq <- chisq + tst$statistic
                df <- df + tst$parameter

        }
}
chisq

## X-squared
##  18.78956

df

## df
##  4
```

The data strongly contradict $X \perp\!\!\!\perp Y \mid \{M1, M2\}$.



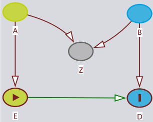
Summary of Test Results

- We assumed the following DAG:



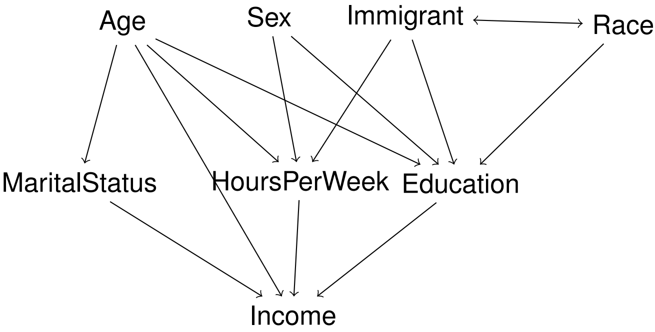
- Using *d*-Separation, we derived two conditional independencies from the net:
  1. $M_1 \perp\!\!\!\perp M_2 \mid X \Rightarrow$ OK
  2. $X \perp\!\!\!\perp Y \mid M_1, M_2 \Rightarrow$ Not OK!

```r
localTests( "dag{ X -> {M1 M2} -> Y }", d, type="cis.chisq" )[,2:4]

##                           x2 df      p.value
## M1 _||_ M2 | X     0.5062151  2 0.7763843914
## X _||_ Y | M1, M2 21.5183312  4 0.0002498793
```

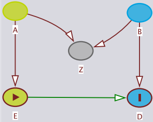
A More Realistic Example

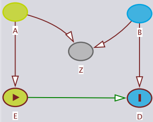
Let us consider a hypothetical DAG for a part of the “Adult census income” dataset:



We will test this model on a cleaned version of the data with $\sim 30,000$ records.

```
d <- read.csv("http://dagitty.net/learn/adult.csv")
```

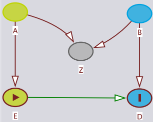


Big Data Brings Low P-Values

With enough data, most DAG models can be falsified.

```
g <- 'dag{
   Age -> {MaritalStatus HoursPerWeek Education} -> Income
   Age -> Income
   Sex -> {HoursPerWeek Education} <- Immigrant
   Immigrant <-> Race -> Education }'
localTests( g, d, type="cis.chisq",
   max.conditioning.variables=2 )[,4,drop=FALSE]

##                                               p.value
## Age _||_ Immigrant                       8.605148e-12
## Age _||_ Race                            1.768747e-11
## Age _||_ Sex                             1.876589e-61
## Education _||_ MaritalStatus | Age       1.489605e-85
## HoursPerWeek _||_ MaritalStatus | Age 2.025719e-241
## HoursPerWeek _||_ Race | Immigrant       1.957253e-85
## Immigrant _||_ MaritalStatus             9.741660e-11
## Immigrant _||_ Sex                       6.016682e-01
## MaritalStatus _||_ Race                  1.625036e-62
## MaritalStatus _||_ Sex                   0.000000e+00
## Race _||_ Sex                            1.681026e-74
```

Most p-values are extremely low. It seems that everything depends on everything. But how strong are the dependencies?



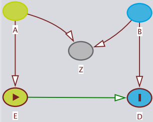
The Root Mean Square Error of Approximation (RMSEA)

Instead of a p-value, which conflates information about dependence strength and sample size, an effect size is often more useful. For chi-square tests, various effect sizes can be defined. An important one is the RMSEA:

$$\text{RMSEA} = \sqrt{\frac{\chi^2/\text{df} - 1}{N - 1}}$$

Properties of the RMSEA

- The expected RMSEA of a “true” model (independence) is 0.
- For a wrong model, the RMSEA converges to a constant positive value as $N \to \infty$ (the p-value converges to 0).
- Higher RMSEA means worse model fit.



Interpretation of Test Results

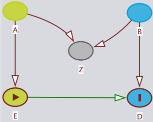
```
localTests( g, d, type="cis.chisq",
   max.conditioning.variables=2 )[,1,drop=FALSE]

##                                               rmsea
## Age _||_ Immigrant                       0.02110776
## Age _||_ Race                            0.02081284
## Age _||_ Sex                             0.04865666
## Education _||_ MaritalStatus | Age       0.04919928
## HoursPerWeek _||_ MaritalStatus | Age 0.07491911
## HoursPerWeek _||_ Race | Immigrant       0.04431913
## Immigrant _||_ MaritalStatus             0.02703967
## Immigrant _||_ Sex                       0.00000000
## MaritalStatus _||_ Race                  0.06843991
## MaritalStatus _||_ Sex                   0.31780131
## Race _||_ Sex                            0.10499237
```

It seems weird that sex and marital status would be so strongly dependent, especially in data from the 1990s.

```
table(d$Sex, d$MaritalStatus)

##
##          Is-Married Never-married Was-Married
##   Female       1681          4312        3789
##   Male        12775          5414        2191
```

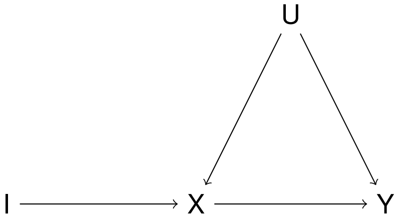
Why DAGs are Not Interesting

- All DAGs we have seen so far assumed that all variables are observed and perfectly measured, such that we can easily stratify for them or regress on them.
- This is often not true.

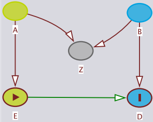


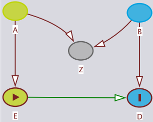
Latent Confounding

Pessimistic epidemiologists do not believe that confounding can ever be controlled by adjustment. Instead, some people prefer to use instrumental variables (I).



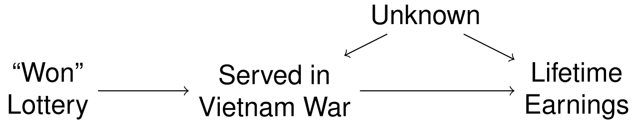
This DAG implies a conditional independence: $I \perp\!\!\!\perp Y \mid X, U$.
But: if U is unobserved, so we cannot condition on it!
Pearl, UAI 1995; Bonet, UAI 2001





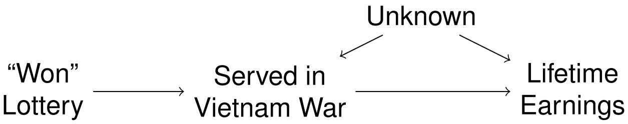
The Vietnam Draft Lottery

In 1969, men were called in a random order determined by their birthdays, and asked to serve in the war. 195 out of 366 possible birthdays were “drafted”. For example, men born on a September 14th were drafted, but men born on a June 20th were not. Not every drafted person enlisted.



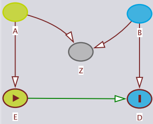
Angrist used “draft” as an IV to determine the effect of serving in the war on lifetime earnings.

The Vietnam Draft Lottery

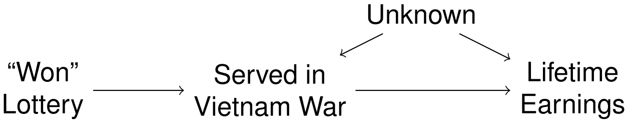


The following findings would be incompatible with this DAG:

- All those who won the lottery went to Vietnam and now earn a lot.
- All those who lost the lottery went to Vietnam and now earn little.

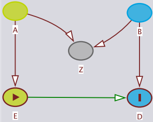


The Vietnam Draft Lottery



The following findings would be incompatible with this DAG:

- 90% of those who won the lottery went to Vietnam and now earn a lot.
- 90% those who lost the lottery went to Vietnam and now earn little.



Pearl's Instrumentality Test

Even though U is unobserved, there is limited information that can pass from I to Y once we hold X constant.

$$\begin{tikzcd} & U \arrow[d] \arrow[dr] & \\ I \arrow[r] & X \arrow[r] & Y \end{tikzcd}$$

If X is binary, we have, for example,

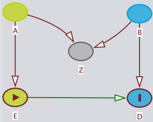
$$P(Y = 1, X = 1 \mid I = 0) + P(Y = 0, X = 1 \mid I = 1) \leq 1$$

For example, suppose these data for the Vietnam lottery:

Won lottery	Went to Vietnam	Didn't go to Vietnam
High earner	70	10
Low earner	10	10

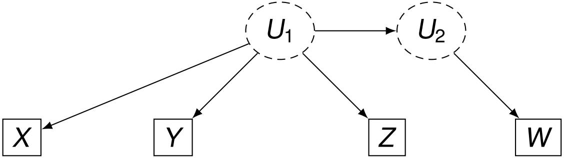
Lost lottery	Went to Vietnam	Didn't go to Vietnam
High earner	20	20
Low earner	40	20

Then there was likely a direct effect of the phone call on lifetime earnings.



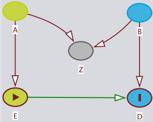
Measurement Models

We can recover from measurement errors by combining several imperfect measurements of the target variable.

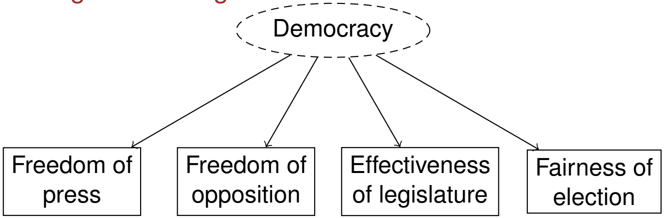


X and Y are indicators of U_1.
Z and W are indicators of U_2.
We need to assume some kind of functional form of the measurement error to recover from it. For example, we might assume Gaussian error

$$X = U_1 + \mathcal{N}(0, \sigma)$$

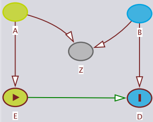


Defining and Testing Measurement Models




```
## lavaan 0.6-2 ended normally after 26 iterations
##
##   Optimization method                           NLMINB
##   Number of free parameters                          8
##
##   Number of observations                            75
##
##   Estimator                                         ML
##   Model Fit Test Statistic                      10.006
##   Degrees of freedom                                 2
##   P-value (Chi-square)                           0.007
```

```
library(lavaan)
sem('dem65 =~ y1 + y2 + y3 + y4', data=PoliticalDemocracy)
```

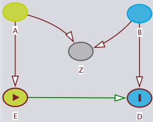
Measurement Models and Structural Models

In structural equation modeling, models are often split in two parts:

- The measurement model links the latent variables to ovservations.
- The structural model links the latent variables to each other.

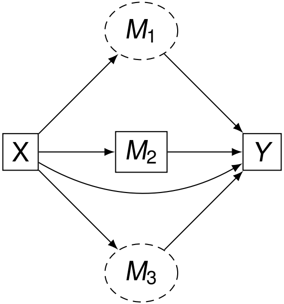
We can test models of this type in a two-step procedure:

1. Test the measurement model. If this fails, stop.
2. Test the structural model.



Dancing the Two-Step

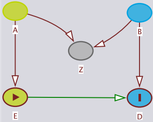
Consider the following multiple mediation model:



Implications:

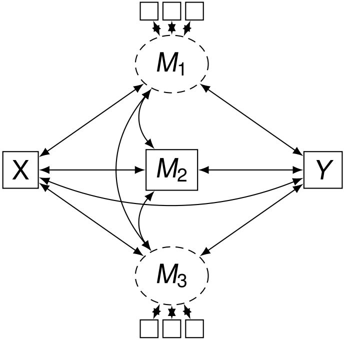
- $M_1 \perp\!\!\!\perp M_2 \mid X$
- $M_2 \perp\!\!\!\perp M_2 \mid X$
- $M_1 \perp\!\!\!\perp M_3 \mid X$
- $X \perp\!\!\!\perp Y \mid M_1, M_2, M_3 X$

Suppose we know that M_1 and M_3 can only be measured with error. Then we cannot expect any of these implications to hold.

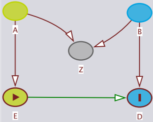


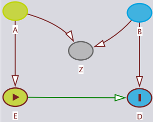
Dancing the Two-Step

We start by building a measurement model and testing that separately. To do that, we saturate the structural model such that it does not imply any constraints.



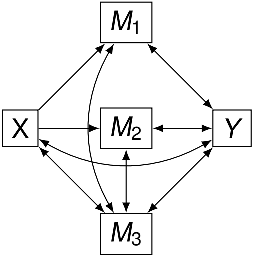
Measurement models can be tested by e.g confirmatory factor analysis or through their implied covariance matrix.



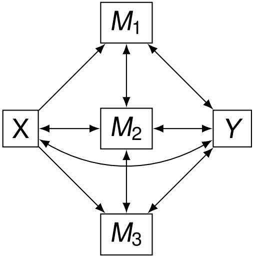


Single-Implication Models

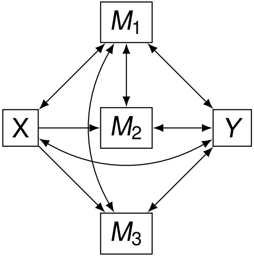
We now construct one separate model per implication of the original structural model, and test these separately.



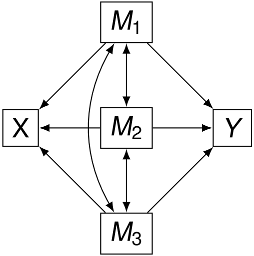
$M_1 \perp\!\!\!\perp M_2 \mid X$



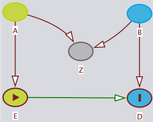
$M_1 \perp\!\!\!\perp M_3 \mid X$



$M_1 \perp\!\!\!\perp M_3 \mid X$

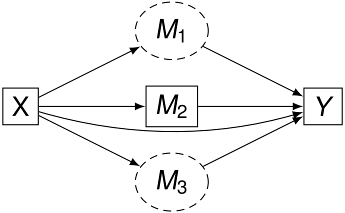


$X \perp\!\!\!\perp Y \mid M_1, M_2, M_3$



Example Results for a Mediation Model

The single-implication trick leverages any existing goodness-of-fit test from structural equation modeling to test DAGs.



	χ^2	df	p	RMSEA	C9(χ^2/df)	C10(χ^2/df)
Total	132	84	7.11^{-04}	.034	-	-
Structural	62.6	4	8.44^{-13}	.171	.896	.104
$M_1 \perp\!\!\!\perp M_2 \mid X$	0.37	1	.543	.000	.999	.001
$M_1 \perp\!\!\!\perp M_3 \mid X$	0.04	1	.838	.000	1.000	.000
$M_2 \perp\!\!\!\perp M_3 \mid X$	53	1	3.39^{-13}	.323	.912	.088
$X \perp\!\!\!\perp Y \mid M_1, M_2, M_3$	12.2	1	4.71^{-04}	.150	.980	.020

The individual test allow us to see which particular implication of the model is false.

Thoemmes, Rosseel & Textor, Psych Methods 2018, doi: https://doi.org/10.1037/met0000147

Summary

1. DAGs put conditional independence constraints on compatible probability distributions.
2. The *d*-separation criterion allows to read off these constraints from the graphical model structure.
3. The constraints can be tested statistically.
4. There are many different kinds of constraints you can use to falsify your DAG models.

Which of these DAGs is better?

$$\begin{tikzcd}[column sep=small] & \text{Job choice} \arrow[dr] & \\ \text{Y chromosome} \arrow[ur] \arrow[rr] & & \text{Pension} \end{tikzcd} \qquad \begin{tikzcd}[column sep=small] & \text{Job choice} \arrow[dr] & \\ \text{Y chromosome} \arrow[ur] & & \text{Pension} \end{tikzcd}$$