GILLES KRATZER

JOINT WORK WITH PROF. DR. REINHARD FURRER

NUTRICIA OCTOBER 4, 2018

ADVANCES IN BAYESIAN NETWORK MODELLING APPLIED TO OBSERVATIONAL SYSTEMS EPIDEMIOLOGY DATASETS

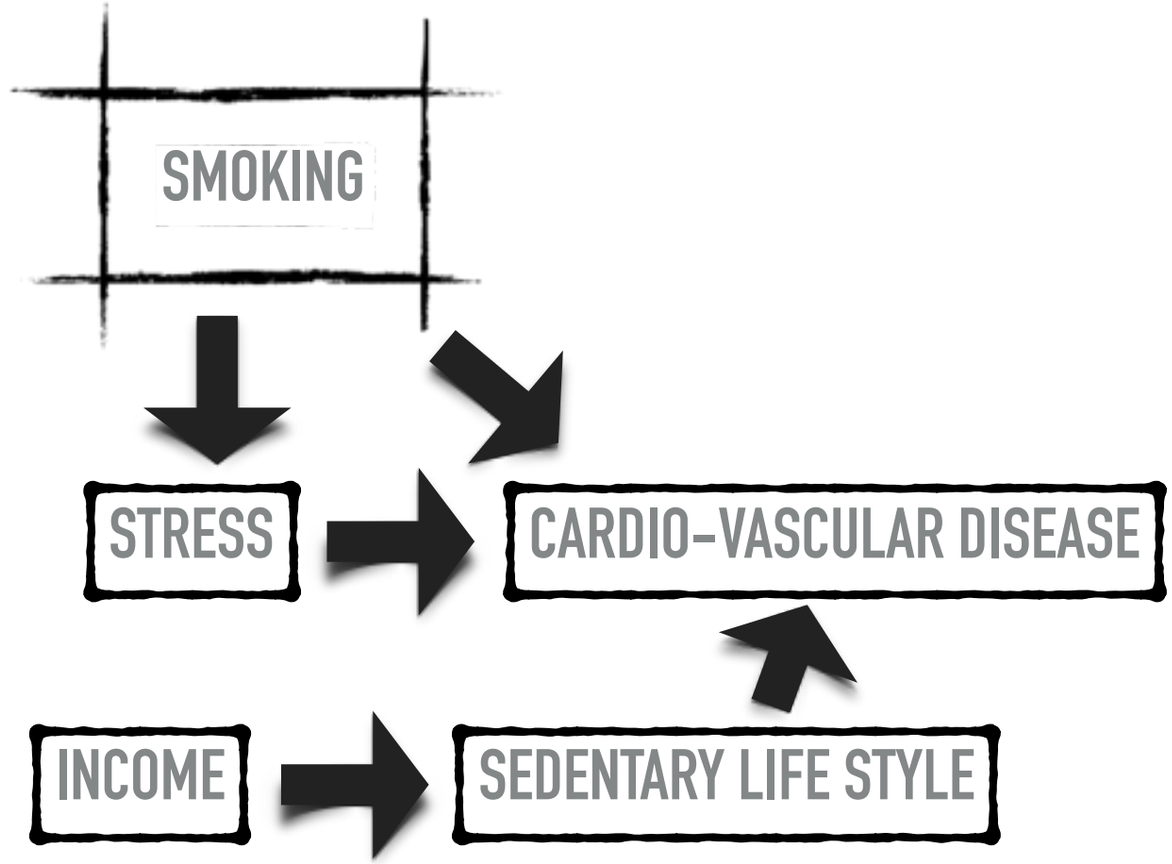
gilles.kratzer@math.uzh.ch

ABN

- *Classical aim in epidemiology is to investigate relationship between covariate and ONE outcome*
- *Typically based on expert knowledge*

Issues:

- Multi-collinearity
- Dependence
- Confounders
- **Multivariate** versus **Multivariables**



Enderlein and al. (1996)

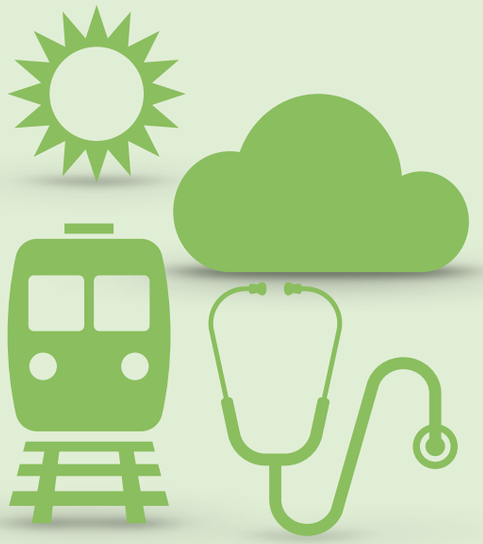
University of Zurich[UZH]

DISEASE LEVEL

- Multiple outcomes/Scores
- Target variables for intervention
- Beginning of the coil of discovery

POPULATION LEVEL

- Demographic data
- Meta population information
- Cluster



ENVIRONMENT LEVEL

- External factors
- Ecology
- Living condition

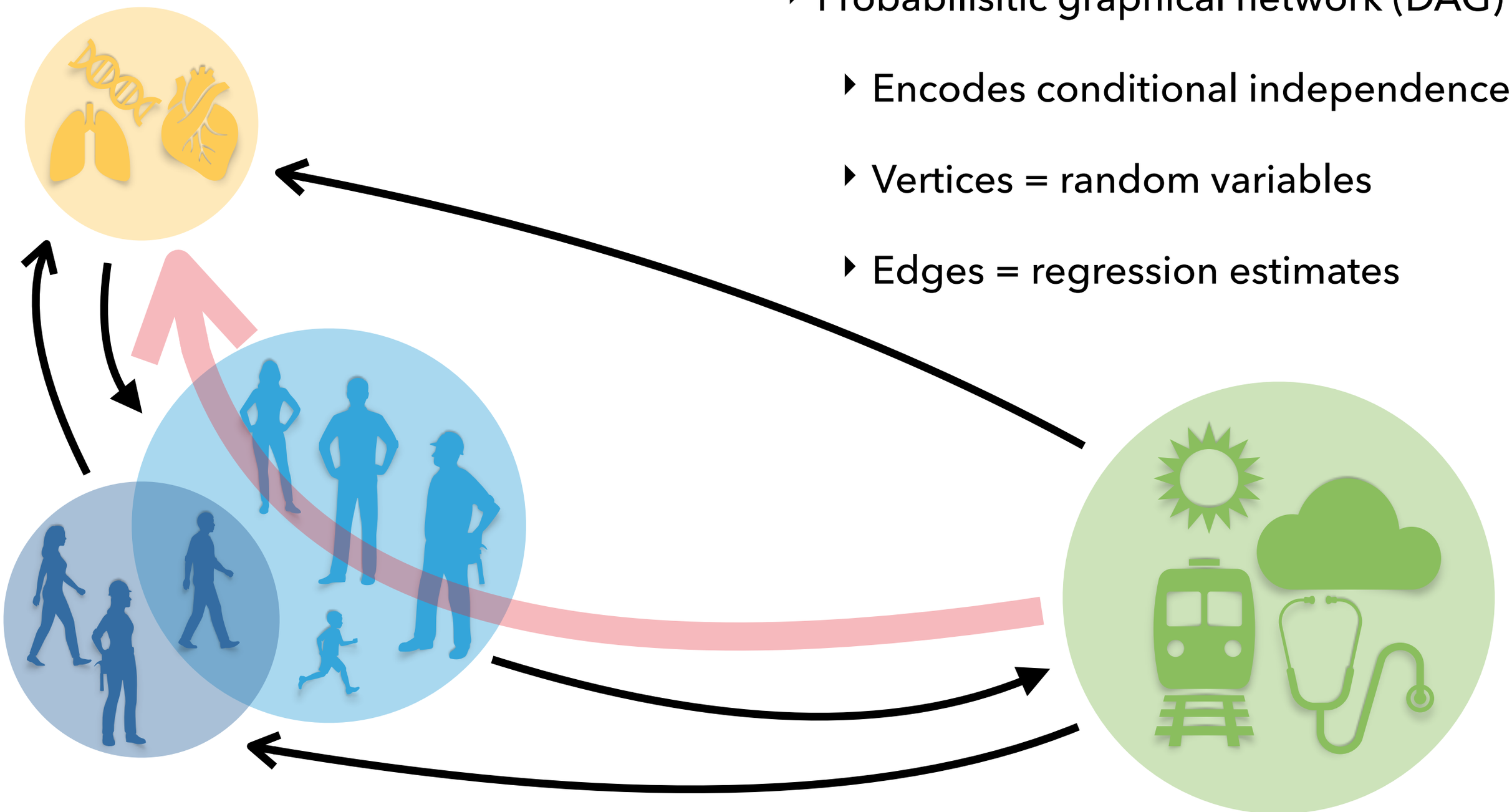
Example

- **Metabolic syndrom**
- A clustering of 3/5 medical conditions

- Observational data
- Age, gender, ...
- Random effect

- Weather condition
- Socio-economic condition
- Housing

Main purpose of ABN: Sort out **directly** associated versus **indirectly** associated, as they are not primary target for intervention



Bayesian Network

- Probabilisitic graphical network (DAG)
  - Encodes conditional independence
  - Vertices = random variables
  - Edges = regression estimates

WHAT IS A BAYESIAN NETWORK?

Bayesian Networks are defined by two elements:

Network structure:

Directed Acyclic Graph (DAG): G = (V, A)

in which each node $v_i \in V$ corresponds to a random variable X_i

Probability distribution:

Probability distribution X with parameters Θ, which can be factorised into smaller local probability distributions according to the arcs $a_{ij} \in A$ present in the graph.

A BN encodes the factorisation of the joint distribution

$$P(\mathbf{X}) = \prod_{j=1}^{n} P(X_j \mid \mathbf{Pa}_j, \Theta_j), \text{ where } \mathbf{Pa}_j \text{ is the set of parents of } X_j$$

$$\mathcal{M} = (\mathcal{S}, \Theta_{\mathcal{M}})$$

Model selection

Structure learning

Parameter estimation

Parameter learning

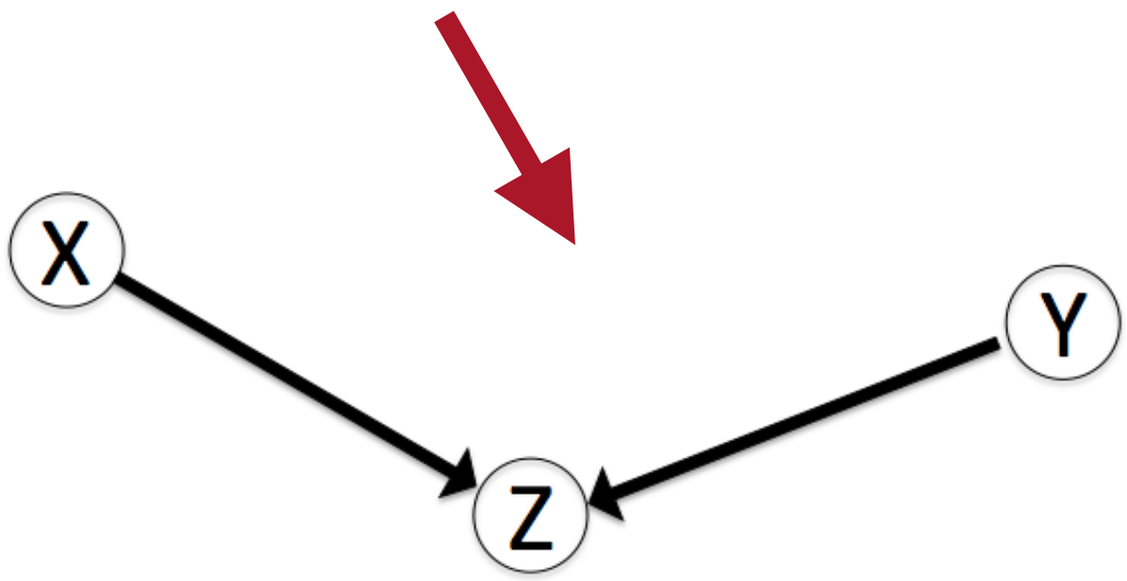
$$P(\mathcal{M}|\mathcal{D}) = \underbrace{P(\Theta_{\mathcal{M}}, \mathcal{S}|\mathcal{D})}_{\text{model learning}} = \underbrace{P(\Theta_{\mathcal{M}}|\mathcal{S}, \mathcal{D})}_{\text{parameter learning}} \cdot \underbrace{P(\mathcal{S}|\mathcal{D})}_{\text{structure learning}}$$

	Fully Observed data	Missing data/hidden variables
Known graph structure	Easy Sample statistics	EM algorithm Gradient ascent Variational inference Doable
Unknown graph structure	Doable **Search-and-score** **PC algorithm**	Hard Structural EM

University of Zurich[UZH]

Constraint based algorithms

Learning independence relationships



Search-and-score algorithms

Maximum a posteriori score

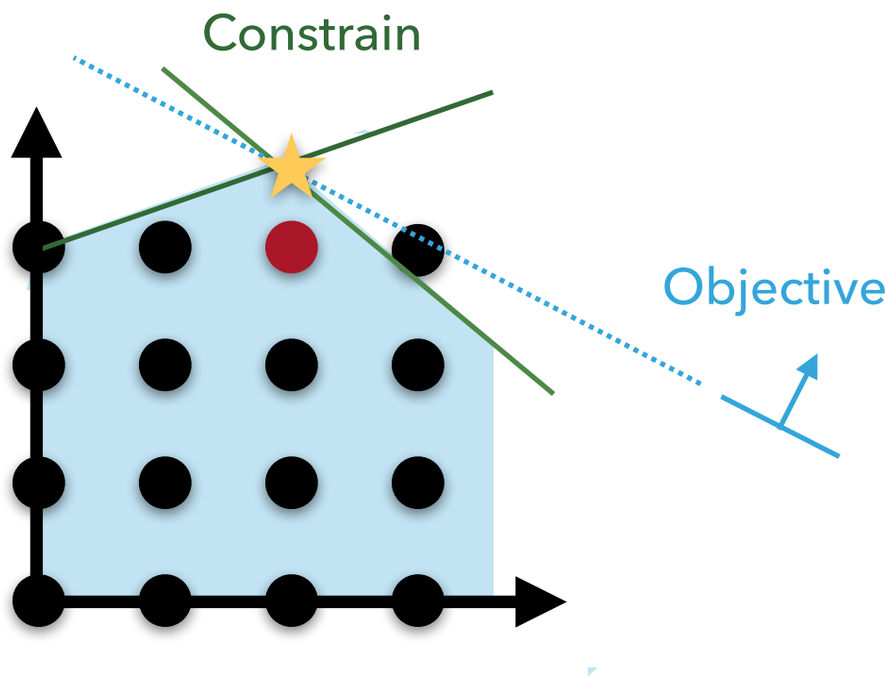
Example of scoring functions:

- Bayesian versus ML scores
  - log marginal likelihood
  - Bayesian-Dirichlet (BDeu,BDs,BDe)
  - Bayesian Information Criterion (BIC)

Score-and-search algorithms

- *Heuristic approaches / Greedy search*
  - Hill-climbing (with possibly random restarts/stochastics …)
  - Tabu search (Glover, 1986)
  - Simulated annealing (Kirkpatrick et al, 1983)
  - Plus an entire zoo of methods …

- *Exact search*
  - Exact node ordering (Koivisto et al. , 2004)
  - Learning with cutting planes (Cussens, 2012)



Scores

- Decomposability!
- Discrete BNs:
  - Bayesian-Dirichlet: BDeu (Heckerman et al. ,1995)
- Score equivalence for additive regression framework:
  - Bayesian based scores: not always score equivalent due to the prior!
  - Information theoretic scores: BIC asymptotically score equivalent

Counter example

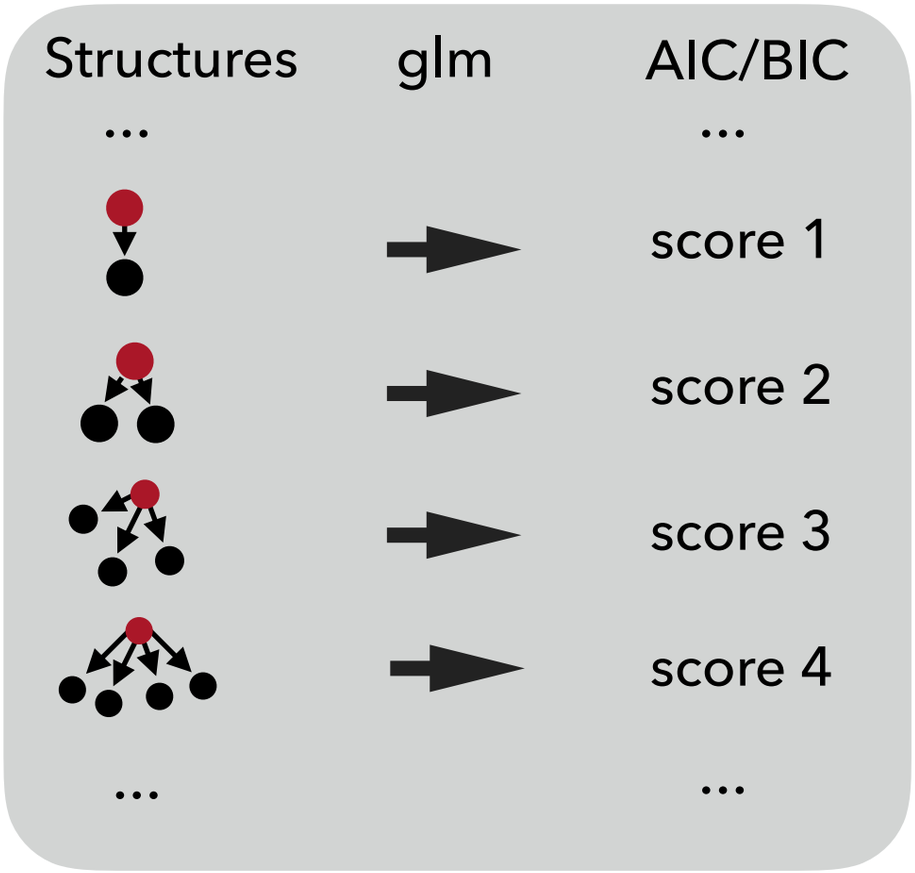
- Maximum likelihood estimator … return fully connected BN!

In a practical perspective:

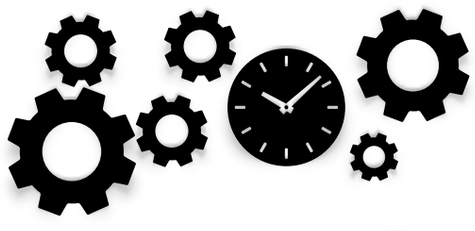
- Scoring mixture of data?
- Score equivalence!

University of Zurich[UZH]

Search and score algorithm

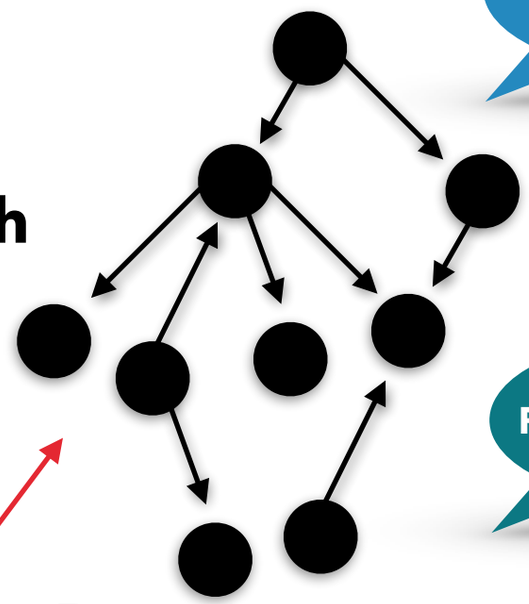


Exact or heuristic search



Causality!

Ban/Retain structures



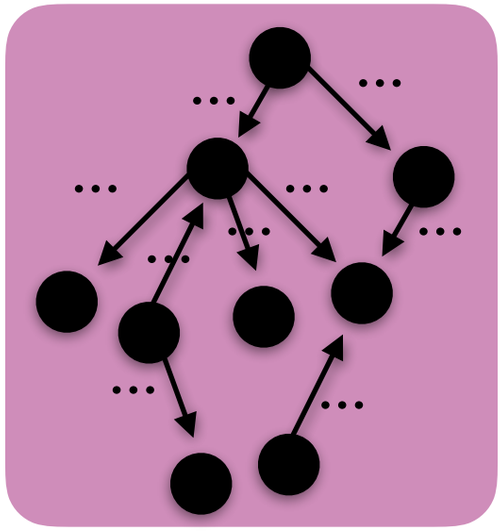
Adjustment

Random effect

Bayesian network with highest posterior probability

Parameter estimation

- compute marginal posterior density
- regression estimate



Using R

buildscorecache()

mostprobable()

fitabn()

CAUSAL THINKING VERSUS ACAUSAL THINKING

- Strong assumptions … but common in statistics, no?

- *"It seems that if conditional independence judgements are byproducts of stored causal relationships, then tapping and representing those relationships directly would be a more natural and more reliable way of expressing what we know or believe about the world. This is indeed the philosophy behind causal Bayesian networks."* (Pearl, 2009)

- The do-calculus
  - Interventions
  - In epidemiology: Randomised Controlled Trial

- So … BN is a nice framework to treat acausal and causal thinking

R CODE: SOFTWARE IMPLEMENTATION

Popular R packages (available on CRAN)

bnlearn

- Learning via constraint-based and score-based algorithms (many!)

pcalg

- Robust estimation of CPDAG via the PC-Algorithm

deal

- Learning BNs with mixed (discrete and continuous) variables

catnet

- Discrete BNs using likelihood-based criteria

abn

- Learning BNs with mixed (discrete, continuous, Poisson) variables
- Score based methods: Bayesian and frequentist estimation
- Exact and heuristic search
- Link strength based on information theory measures

Disclaimer: I am author and maintainer of the abn R package

BAYESIAN NETWORKS IN THE MACHINE LEARNING WORLD

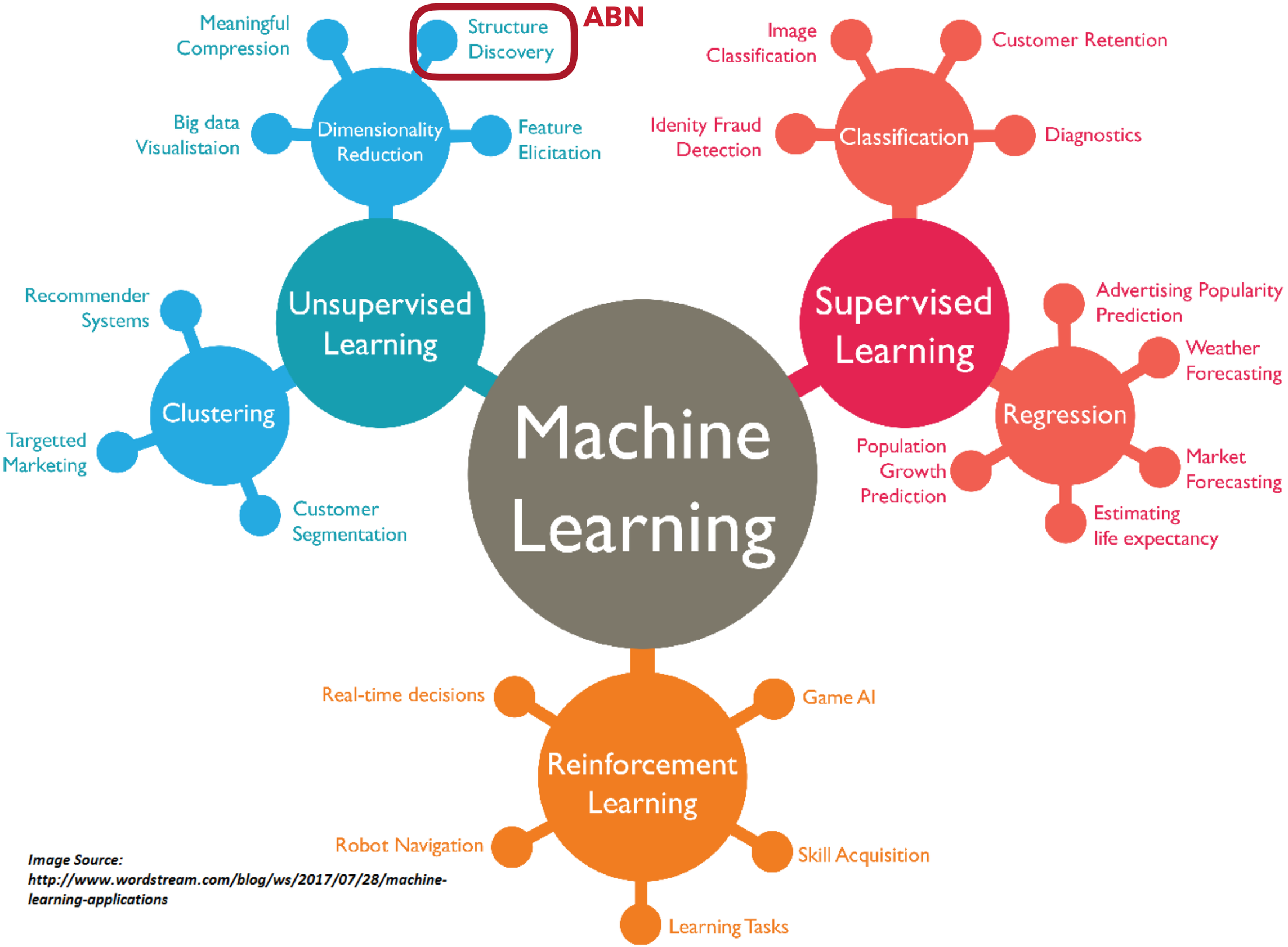
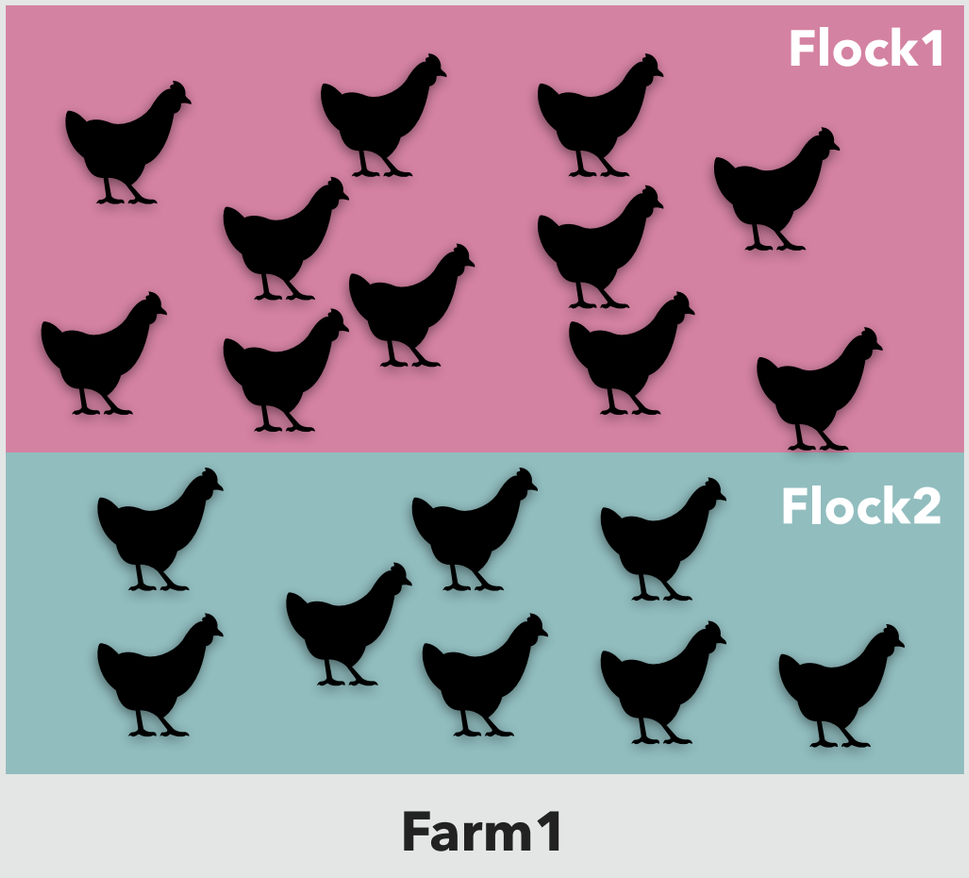


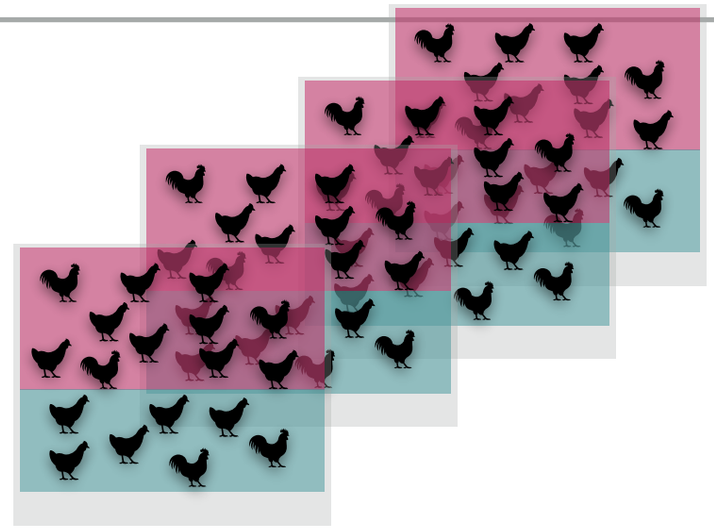
Image Source:
http://www.wordstream.com/blog/ws/2017/07/28/machine-learning-applications

ABN

ANIMAL WELFARE



Farm1



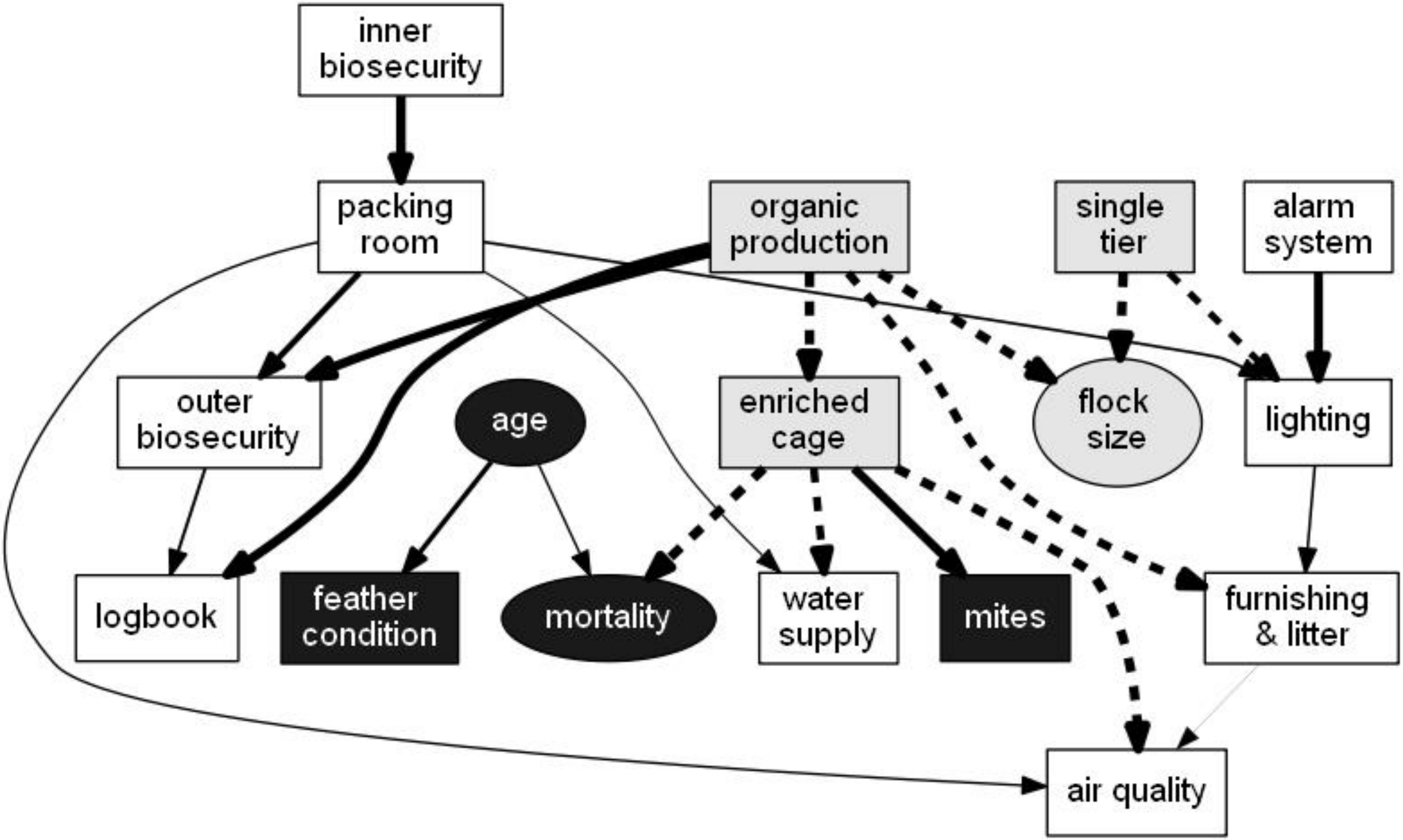
Many farms

Hierachical contribution
- animal level
- Flock level
- Farm level

Risk factor analysis

Animal Welfare

Arianna Comin et al (2017); Revealing the structure of the associations between housing system, facilities, management and welfare of commercial laying hens using Additive Bayesian networks



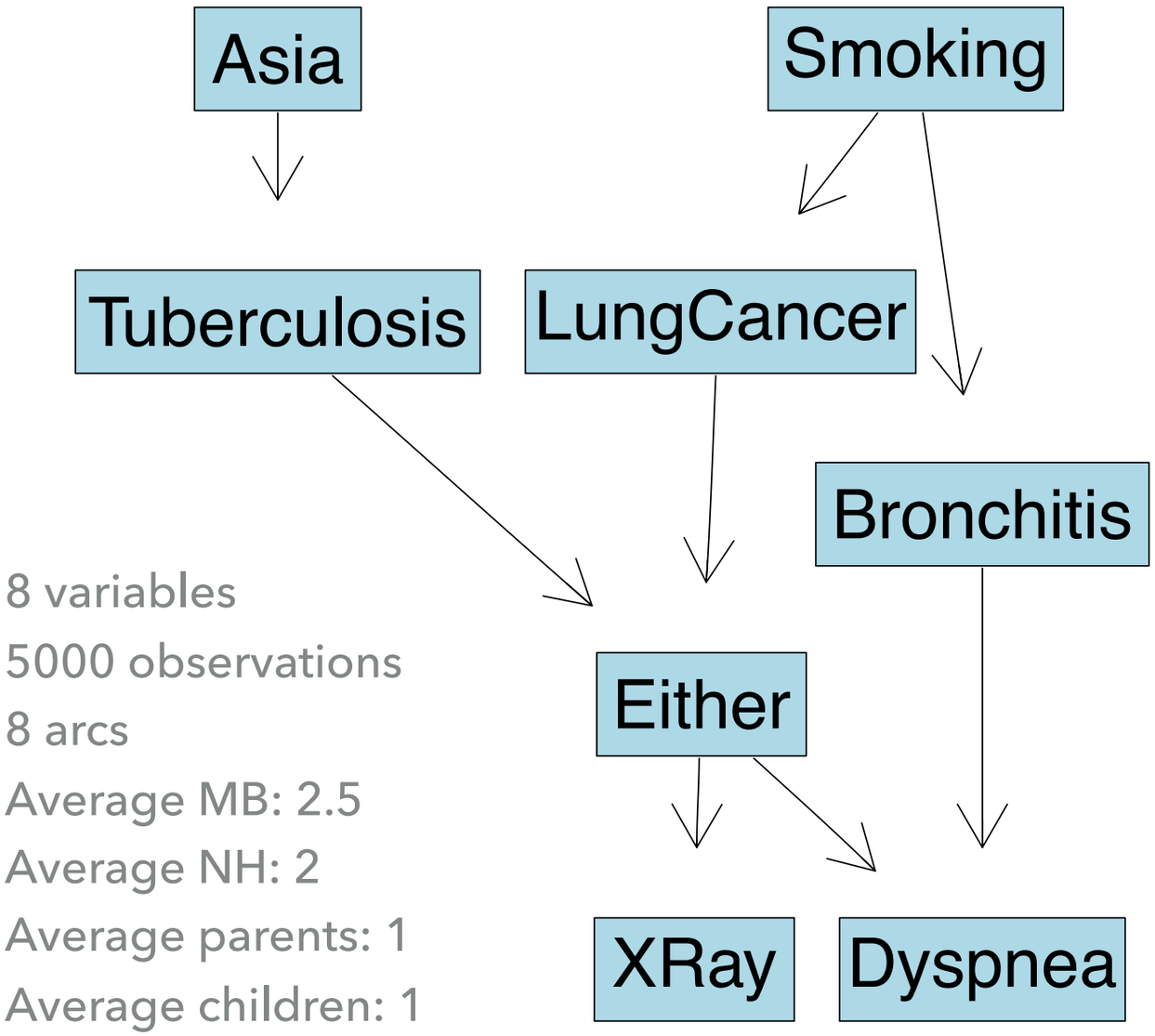
Arianna Comin et al (2017); Revealing the structure of the associations between housing system, facilities, management and welfare of commercial laying hens using Additive Bayesian networks

R CODE: EXAMPLE ASIA

Proposed by Lauritzen et al.,1988 and provided by Scutari, 2009

"Shortness-of-breath (dyspnoea) may be due to tuberculosis, lung cancer or bronchitis, or none of them, or more than one of them. A recent visit to Asia increases the chances of tuberculosis, while smoking is known to be a risk factor for both lung cancer and bronchitis. The results of a single chest X-ray do not discriminate between lung cancer and tuberculosis, as neither does the presence or absence of dyspnoea."

```
##defining distributions
dist = list(Asia = "binomial",
            Smoking = "binomial",
            Tuberculosis = "binomial",
            LungCancer = "binomial",
            Bronchitis = "binomial",
            Either = "binomial",
            XRay = "binomial",
            Dyspnea = "binomial")

#plot BN
plotabn(dag.m = ~Asia|Tuberculosis +
            Tuberculosis|Either +
            Either|XRay:Dyspnea +
            Smoking|Bronchitis:LungCancer +
            LungCancer|Either +
            Bronchitis|Dyspnea,
          data.dists = dist,
          edgedir = "cp",
          fontsize.node = 30,
          edge.arrowwise = 3)
```

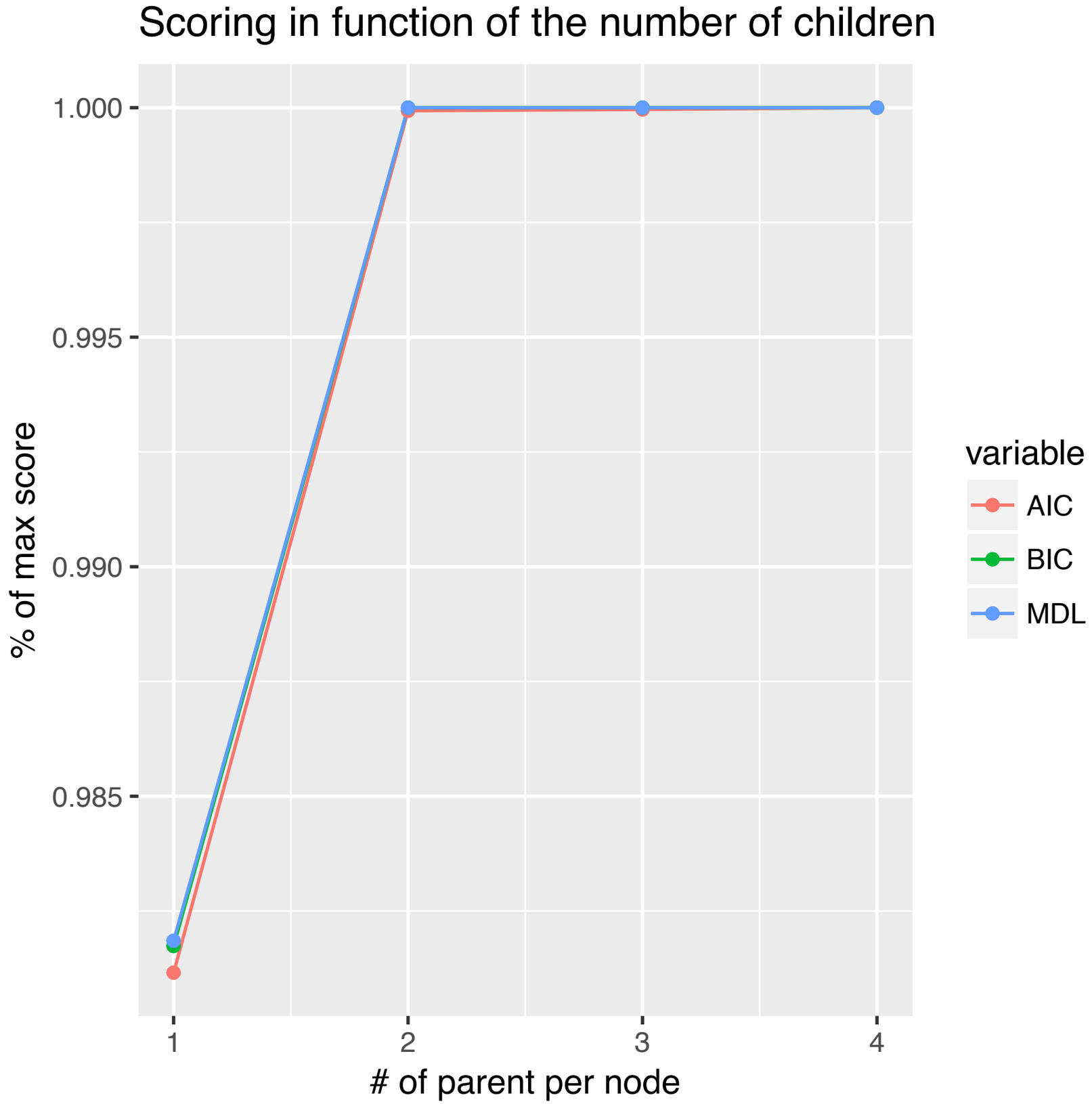


8 variables
5000 observations
8 arcs
Average MB: 2.5
Average NH: 2
Average parents: 1
Average children: 1

ASIA: HOW MANY PARENT ARE NEEDED?

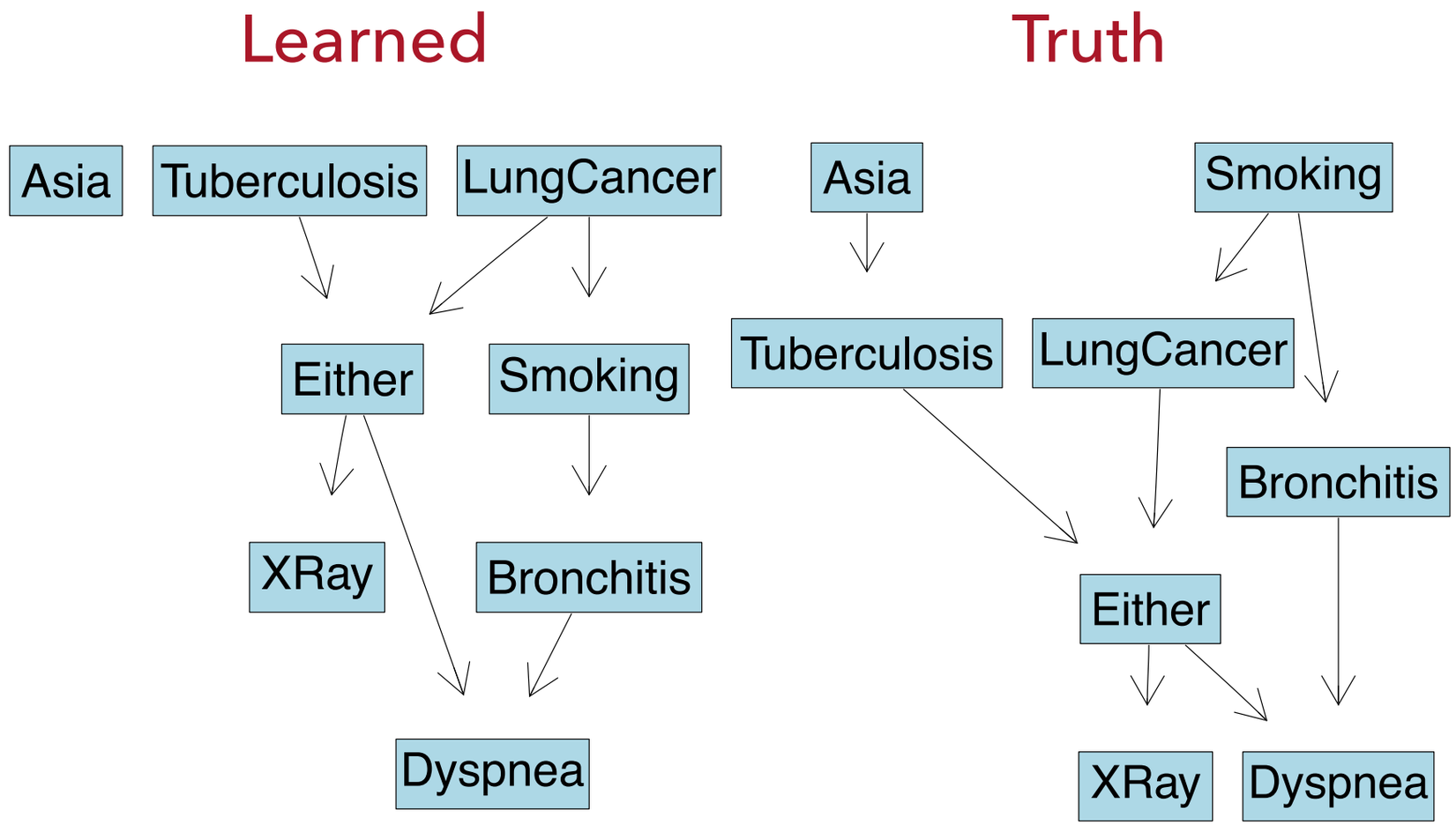
```
res.mlik <- NULL
res.aic <- NULL
res.bic <- NULL
res.mdl <- NULL

for(i in 1:4){
  mycache.computed.mle <- buildscorecache.mle(data.df = asia,
                                              data.dists = dist,
                                              max.parents = i,
                                              dry.run = FALSE,
                                              maxit = 1000,
                                              tol = 1e-11)

  dag <- mostprobable(score.cache = mycache.computed.mle,score = "aic")
  res.aic <- rbind(res.aic,fitabn.mle(dag.m = dag,data.df = mycache.computed.mle$data.df,data.dists = dist)$aic)
  dag <- mostprobable(score.cache = mycache.computed.mle,score = "bic")
  res.bic <- rbind(res.bic,fitabn.mle(dag.m = dag,data.df = mycache.computed.mle$data.df,data.dists = dist)$bic)
  dag<-mostprobable(score.cache = mycache.computed.mle,score = "mdl")
  res.mdl <- rbind(res.mdl,fitabn.mle(dag.m = dag,data.df = mycache.computed.mle$data.df,data.dists = dist)$mdl)
}

library(ggplot2)
library(reshape)
scoring <- data.frame(AIC = max(-res.aic)/-res.aic, BIC = max(-res.bic)/-res.bic, MDL = max(-res.mdl)/-res.mdl, 1:4)

scoring.long <- melt(scoring, id.vars="X1.4")

ggplot(data = scoring.long, aes(x=X1.4, y=(value), group=variable, color=variable)) +
  geom_line() +
  geom_point() +
  ggtitle("Scoring in function of the number of children", subtitle = NULL) +
  xlab("# of parent per node") +
  ylab("% of max score") +
  scale_x_continuous(breaks=c(1,2,3,4,5,6,7))
```

ASIA: HOW MANY PARENT ARE NEEDED?



ASIA: SCORE BASED ALGORITHM

```
##============================
##score based algorithm
##============================

#loglikelihood score
bsc.compute <- buildscorecache(data.df = asia,
                               data.dists = dist,
                               max.parents = 2)

dag <- mostprobable(score.cache = bsc.compute)
plotabn(dag.m = dag,data.dists = dist, fontsize.node = 30, edge.arrc
```




```
> compareDag(ref = t(dag.adj),
+            test = dag)
$TPR
[1] 0.75

$FPR
[1] 0.01785714

$Accuracy
[1] 0.953125

$FDR
[1] 0.2857143

$`G-measure`
[1] 0.8017837

$`F1-score`
[1] 44.8

$PPV
[1] 0.8571429

$FOR
[1] 0.2857143

$`Hamming-distance`
[1] 3
```

ASIA: KNOWN NETWORK

```
fitabn(dag.m = ~Asia|Tuberculosis+
         Tuberculosis|Either +
         Either|XRay:Dyspnea +
         Smoking|Bronchitis:LungCancer +
         LungCancer|Either +
         Bronchitis|Dyspnea,data.df = asia,data.dists = dist)$modes
```

```
$Asia
 Asia|(Intercept) Asia|Tuberculosis
        -4.811200           1.765763

$Smoking
Smoking|(Intercept)  Smoking|LungCancer  Smoking|Bronchitis
          -1.027065            2.356988            1.807460

$Tuberculosis
Tuberculosis|(Intercept)      Tuberculosis|Either
                -12.22120                 10.21823

$LungCancer
LungCancer|(Intercept)      LungCancer|Either
             -12.07565              14.18547

$Bronchitis
Bronchitis|(Intercept)     Bronchitis|Dyspnea
             -1.388644               3.200393

$Either
Either|(Intercept)        Either|XRay     Either|Dyspnea
         -8.656348           8.259773           1.538789

$XRay
XRay|(Intercept)
       -2.052496

$Dyspnea
Dyspnea|(Intercept)
         -0.1201444
```

```
fitabn.mle(dag.m = dag.adj,data.df = asia,data.dists = dist)$coef
```

```
$Asia
     Asia|intercept Tuberculosis
[1,]      -4.811371     1.766849

$Smoking
     Smoking|intercept LungCancer Bronchitis
[1,]         -1.027075   2.357079   1.807472

$Tuberculosis
     Tuberculosis|intercept   Either
[1,]              -8.517393 6.516139

$LungCancer
     LungCancer|intercept   Either
[1,]             -8.517393 10.62598

$Bronchitis
     Bronchitis|intercept  Dyspnea
[1,]            -1.388655 3.200415

$Either
     Either|intercept     XRay  Dyspnea
[1,]        -8.665128 8.268402 1.539146

$XRay
     XRay|intercept
[1,]        -2.0525

$Dyspnea
     Dyspnea|intercept
[1,]        -0.1201443
```

ASIA: BOOTSTRAPPING

University of Zurich[UZH]

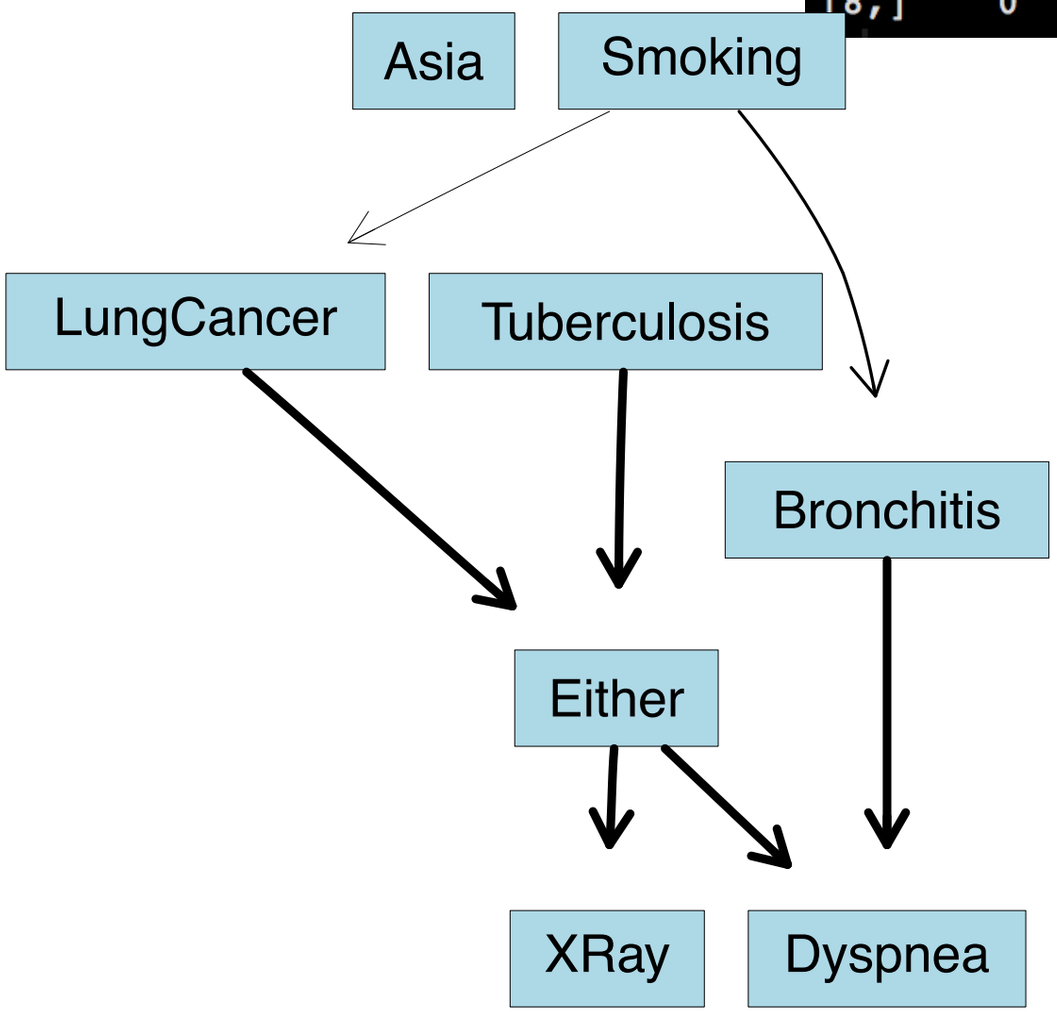
```
library(doParallel)
library(foreach)

cl <- makeCluster(2)
registerDoParallel(cl)

set.seed(1120)
nboot <- 200
nvars <- dim(asia)[2]
nobs <- dim(asia)[1]
bootstrap.dag <- array(data = NA,dim = c(nvars, nvars, nboot))

start_time <- Sys.time()
bootstrap.dag <- foreach(i = 1:nboot,.packages = c("mlabn", "abn")) %dopar% {
  mycache.computed.mle <- buildscorecache.mle(data.df = asia[sample(x = 1:nobs,size = 0.8*nobs,replace = FALSE),],
                                              data.dists = dist,
                                              max.parents = 2,
                                              dry.run = FALSE,
                                              maxit = 1000,
                                              tol = 1e-11)

  dag <- mostprobable(score.cache = mycache.computed.mle, score = "bic")}
compute_time <- Sys.time()-start_time

##analysis
df.boot <- array(data = unlist(bootstrap.dag), dim = c(8, 8, 200))

dag<-apply(df.boot, 1:2, mean)

#dag.mdl<-dag.before

colnames(dag) <- rownames(dag) <- names(dist)

dag.boot.50 <- dag
dag.boot.50[dag>0.5] <- 1
dag.boot.50[dag<=0.5] <- 0

dag[dag<=0.5] <- 0

colnames(dag.boot.50) <- rownames(dag.boot.50) <- names(dist)

plotabn(dag.m = t(dag.boot.50),data.dists = dist,fontsize.node = 30,arc.strength = 10*dag,digit.precision = 2,edge.arrowwise = 3)
```

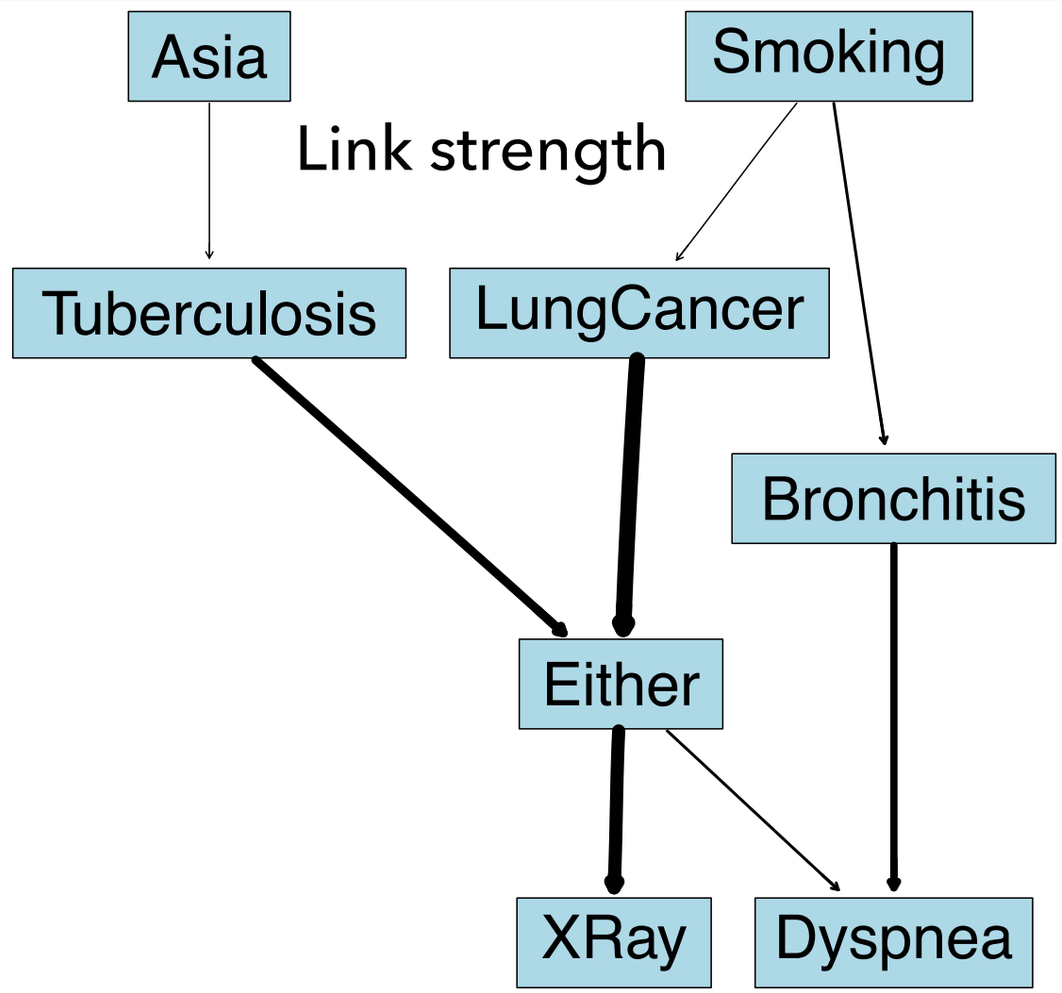
ASIA: BOOTSTRAPPING VERSUS LINK STRENGTH

University of Zurich[UZH]

Bootstrapping

```
> link.strength(dag.m = dag.m,
+               data.dists = dist,
+               data.df = asia,
+               method = "ls.pc",
+               discretization.method = "fd")
     [,1] [,2]        [,3]       [,4]      [,5]      [,6]      [,7]       [,8]
[1,]    0    0 0.007462114 0.00000000 0.0000000 0.0000000 0.0000000 0.00000000
[2,]    0    0 0.000000000 0.03855823 0.1324874 0.0000000 0.0000000 0.00000000
[3,]    0    0 0.000000000 0.00000000 0.0000000 0.4645549 0.0000000 0.00000000
[4,]    0    0 0.000000000 0.00000000 0.0000000 0.8957586 0.0000000 0.00000000
[5,]    0    0 0.000000000 0.00000000 0.0000000 0.0000000 0.0000000 0.34248072
[6,]    0    0 0.000000000 0.00000000 0.0000000 0.0000000 0.7119201 0.08851403
[7,]    0    0 0.000000000 0.00000000 0.0000000 0.0000000 0.0000000 0.00000000
[8,]    0    0 0.000000000 0.00000000 0.0000000 0.0000000 0.0000000 0.00000000
```

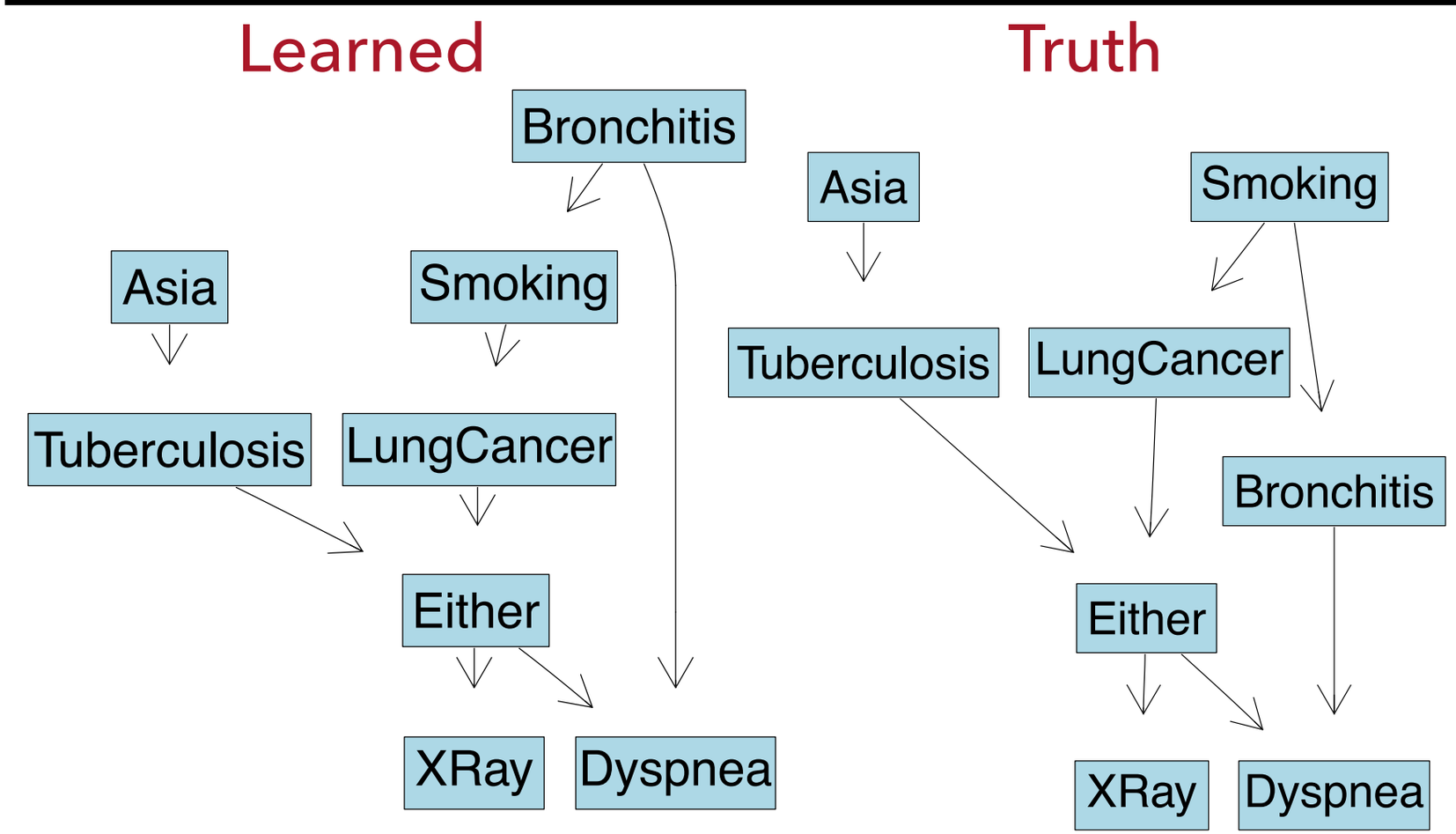




ASIA: EXTERNAL KNOWLEDGE

```
##============================
##external knowledge
##============================

##recent visit to Asia increases risk of tuberculosis
bsc.compute <- buildscorecache.mle(data.df = asia,
                                   data.dists = dist,
                                   max.parents = 2,
                                   dag.retained = ~Tuberculosis|Asia)

dag <- mostprobable(score.cache = bsc.compute,score = "bic")
plotabn(dag.m = dag,data.dists = dist, fontsize.node = 30, edge.arro
```

```
> compareDag(ref = t(dag.adj),
+            test = (dag))
$TPR
[1] 0.875

$FPR
[1] 0.01785714

$Accuracy
[1] 0.96875

$FDR
[1] 0.125

$`G-measure`
[1] 0.875

$`F1-score`
[1] 56

$PPV
[1] 0.875

$FOR
[1] 0.125

$`Hamming-distance`
[1] 2
```



University of Zurich[UZH]

- Simple output
- Arc coefficients: easy to interpret
- Statistical guarantees

Current implementation

- Distributed as an R package (CRAN)
- Bayesian regression based on INLA (lm, logit and Poisson) with possibly **random effect**
- Most probable search (**exact search**) and Hill climber (**heuristic approach**)
- Arc strength based on Mutual Information
  - Significance not p-value based
- GLM implementation (data separation, multinomial variable) with possibly **adjustment**
  - Multiple scores: AIC, BIC, MDL

LARGE NETWORK

Proposed by Beinlich et al.,1989 and provided by Scutari, 2009

ALARM (A Logical Alarm Reduction Mechanism) is a diagnostic application used to explore probabilistic reasoning techniques in belief networks. ALARM implements an alarm message system for patient monitoring; it calculates probabilities for a differential diagnosis based on available evidence.

The medical knowledge is encoded in a graphical structure connecting: (37 variables)

- *8 diagnoses variables*
- *16 findings variables*
- *13 intermediate variables*

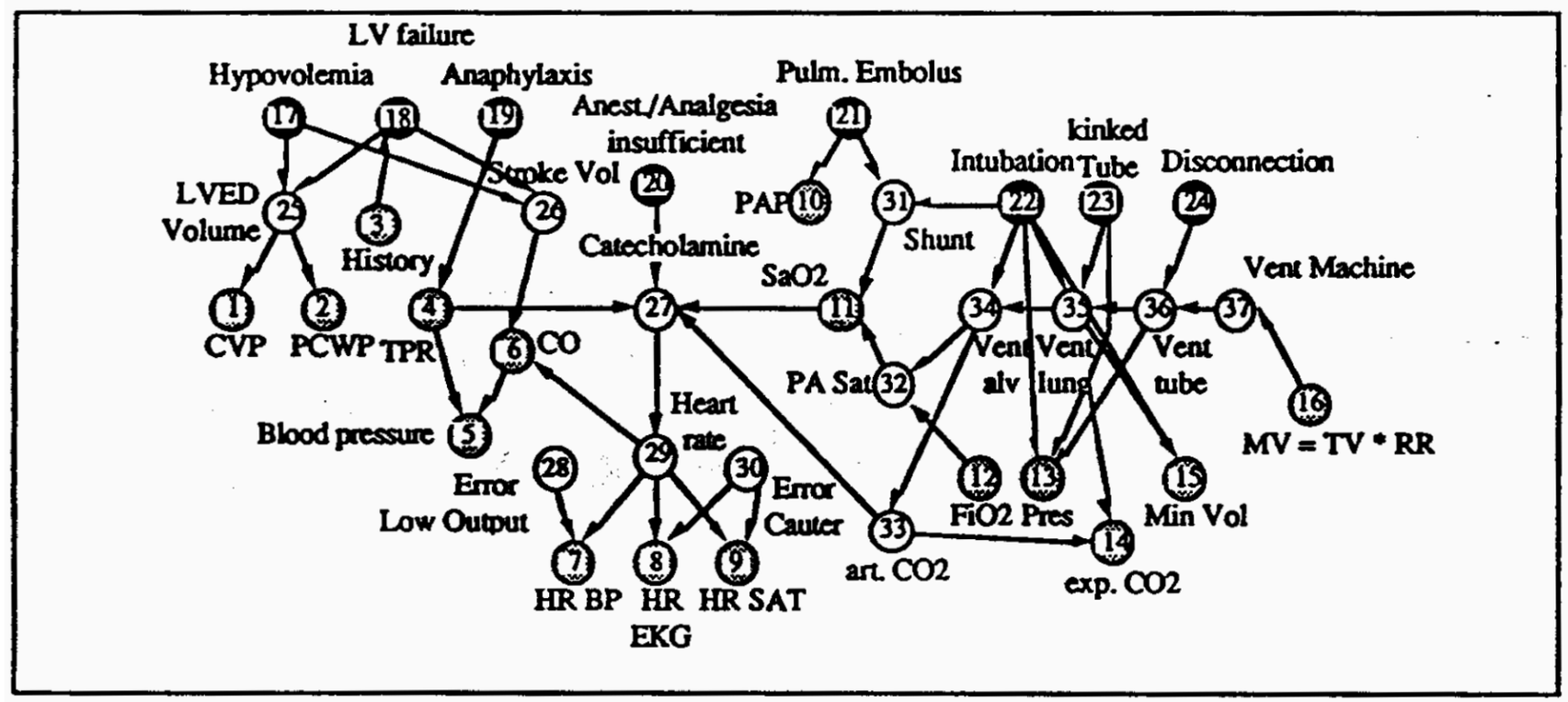
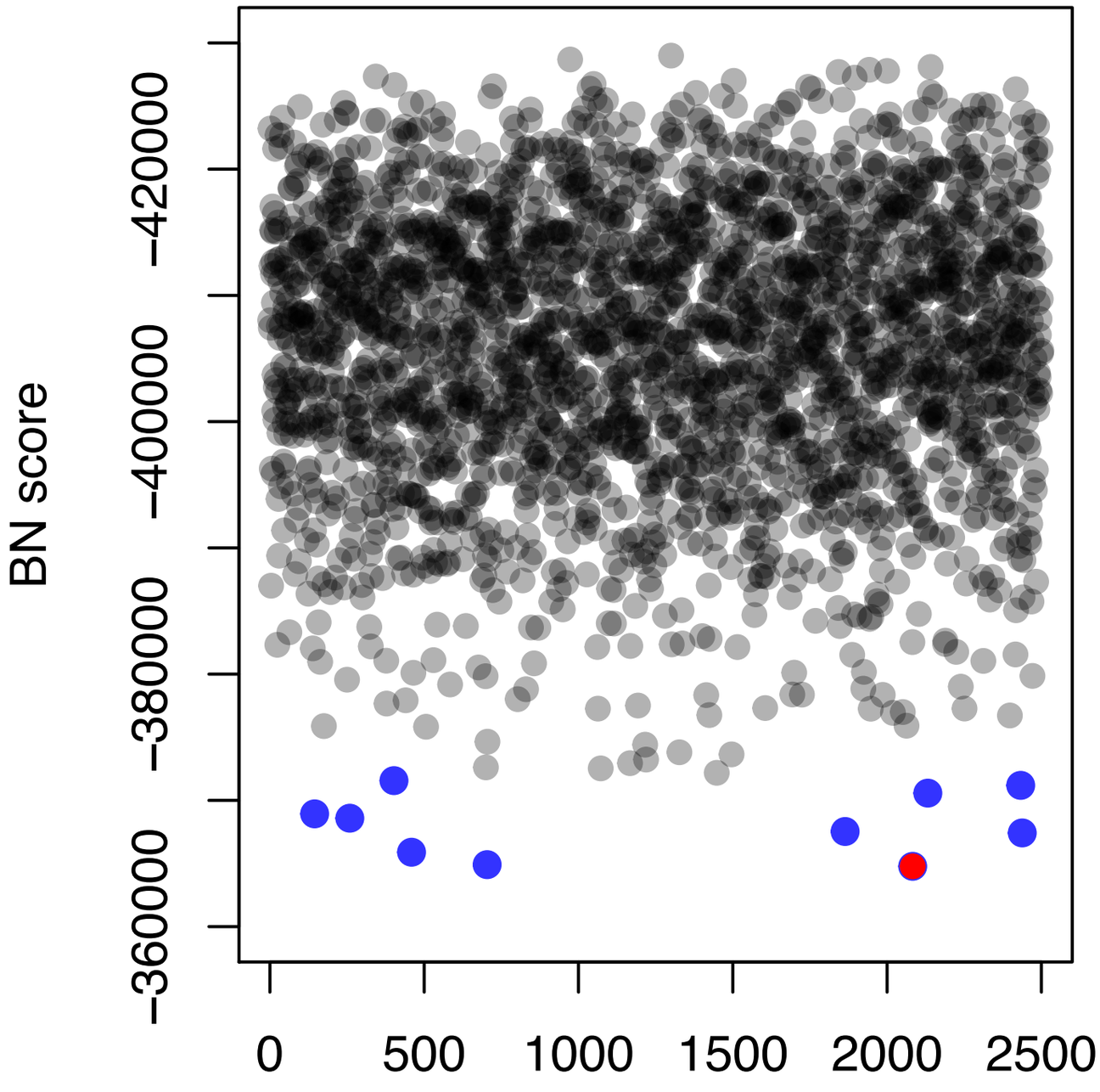


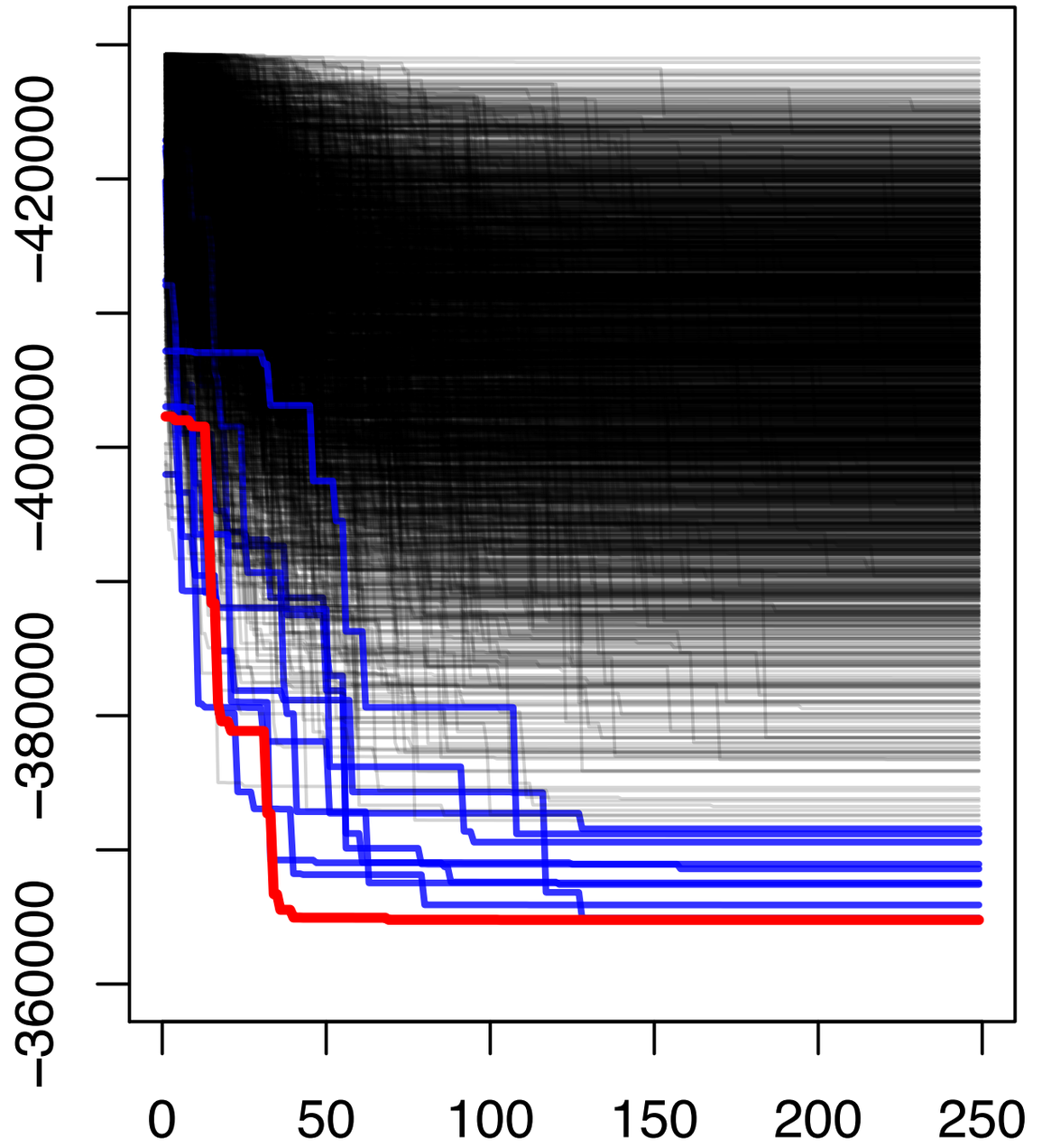
Fig. 1 The ALARM network representing causal relationships is shown with diagnostic (●), intermediate (○) and measurement (◎) nodes. CO: cardiac output, CVP: central venous pressure, LVED volume: left ventricular end-diastolic volume, LV failure: left ventricular failure, MV: minute ventilation, PA Sat: pulmonary artery oxygen saturation, PAP: pulmonary artery pressure, PCWP: pulmonary capillary wedge pressure, Pres: breathing pressure, RR: respiratory rate, TPR: total peripheral resistance, TV: tidal volume

LARGE NETWORK


```
bsc.compute <- buildscorecache.mle(data.df = alarm, data.dists = dist,  max.parents = 1)


bn.hc <- search.heuristic(score.cache = bsc.compute,
                          score = "bic",
                          data.dists = dist,
                          num.searches = 2500,
                          max.steps = 250,
                          start.dag = "random",
                          verbose = TRUE,
                          algo = "tabu",tabu.memory = 10)
```

LARGE NETWORK



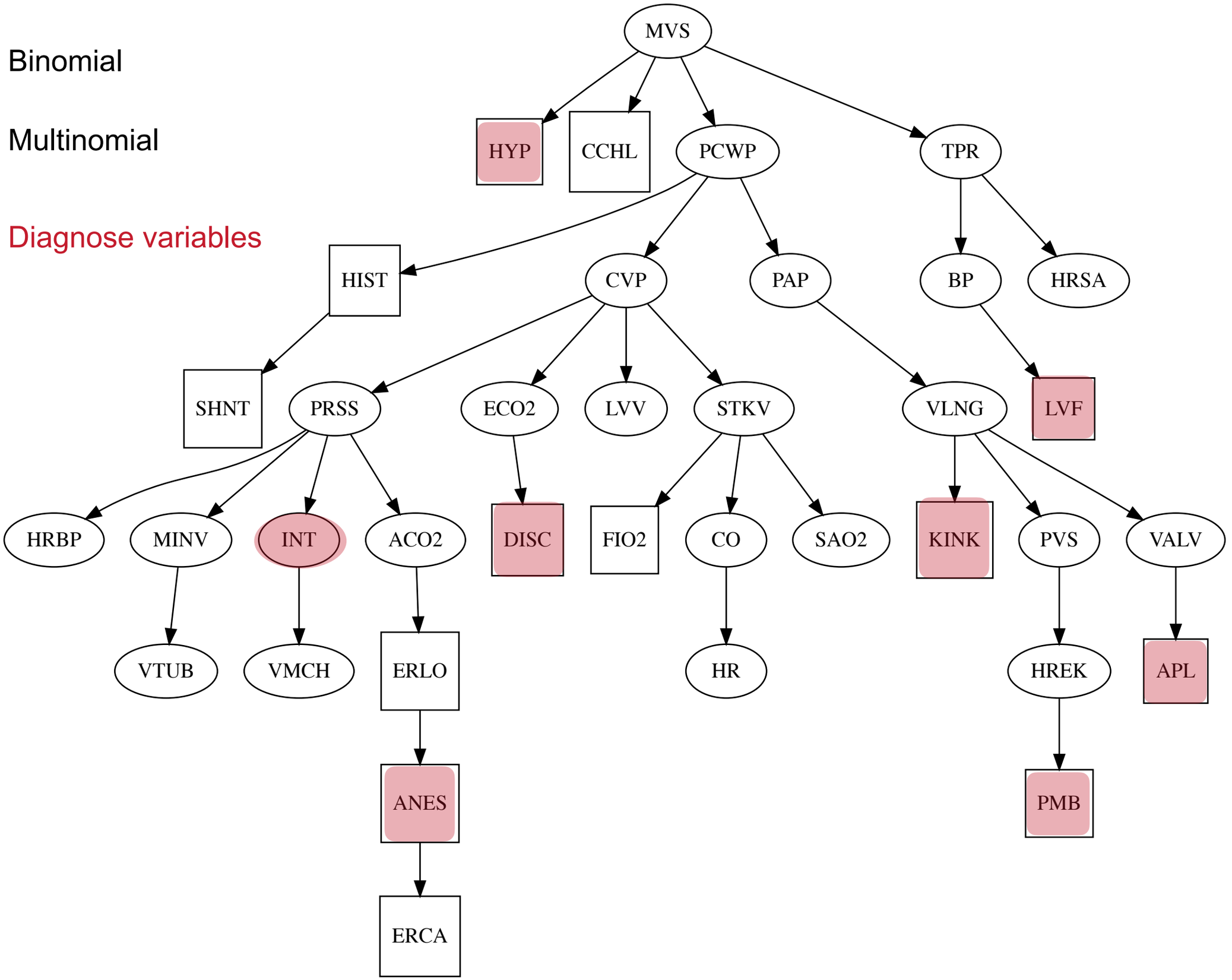


LARGE NETWORK

Binomial

Multinomial

Diagnose variables



University of Zurich[UZH]

System epidemiology

- Typically the set of possible variables is formidable
  - The classical approach for variable selection is based on prior scientific knowledge (29%)[1]
  - Change of estimate (18%)[1]
  - Stepwise model selection (16%)[1]
- No prior model
- Not one outcome experiment

varrank **Variable ranking for better time allocation**

- Variable ranking based on a set of variable of importance
- Model free. Based on information theory metrics
- Mixture of variables (continuous and discrete). Discretisation through rule/clustering
- Ranking of 100 variables with 100’000 observations in ~14 minutes! (forward greedy search)

[1] *Walter et al (2009)*

University of Zurich[UZH]

f_i candidate feature to be ranked

C set of variables of importance

S set of already selected variables

$$\mathrm{H}(X) = \sum_{n=1}^{N} P(x_n) \log P(x_n)$$

Average amount of information of one RV

$$\mathrm{MI}(X;Y) = \sum_{n=1}^{N} \sum_{m=1}^{M} P(x_n; y_m) \log \frac{P(x_n; y_m)}{P(x_n)P(y_m)}$$

Mutual dependence between two RV

Difference (mid) or quotient (miq)

Greedy search

Forward - argmax

Backward - argmax

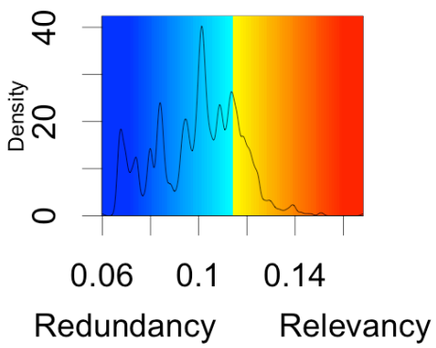
$$\text{score}_i = \underbrace{\mathrm{MI}(f_i; \mathbf{C})}_{\text{Relevance}} - \beta \sum_{\mathbf{S}} \underbrace{\alpha(f_i, f_s, \mathbf{C})}_{\text{Normalization}} \underbrace{\mathrm{MI}(f_i; f_s)}_{\text{Redundancy}}$$

Discretization

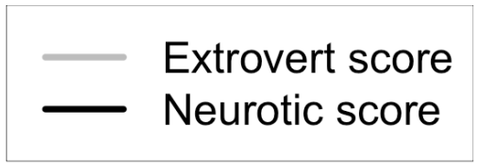
Estévez and al. (2009)

$$\beta = 1/|\mathbf{S}| \text{ and } \alpha(f_i, f_s, \mathbf{C}) = \frac{1}{\min(\mathrm{H}(f_i), \mathrm{H}(f_s))}$$

EYSENCK PERSONALITY INVENTORY

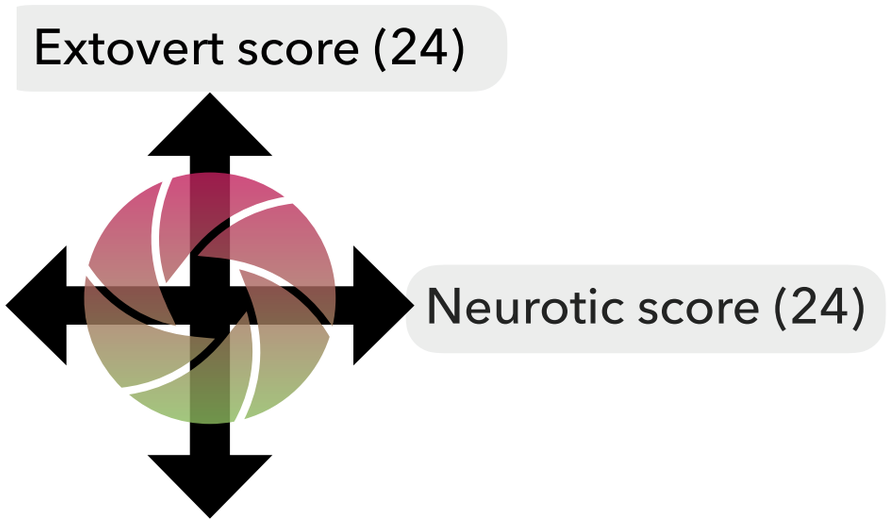


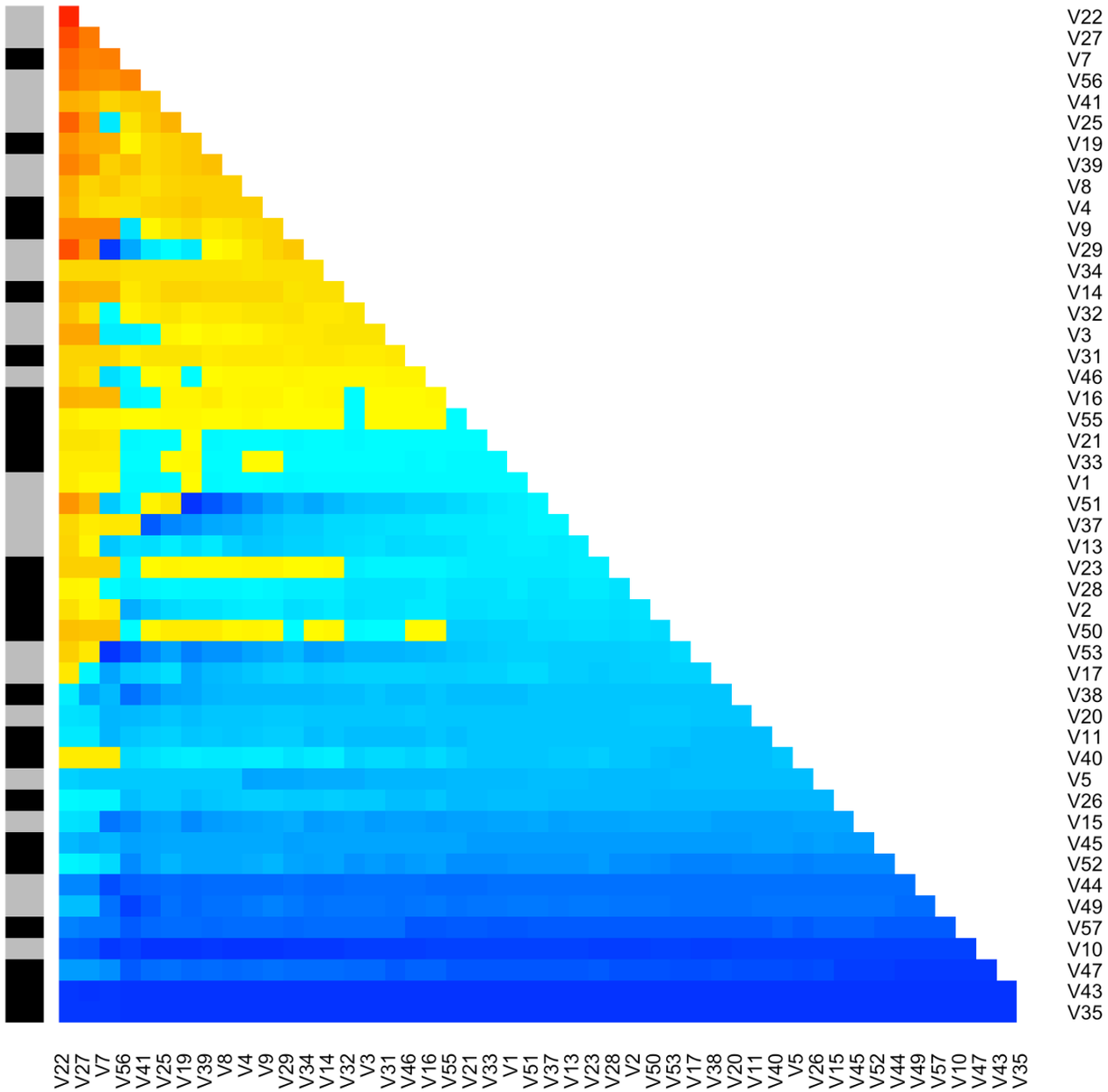
EPI: 3570 observations and 57 variables



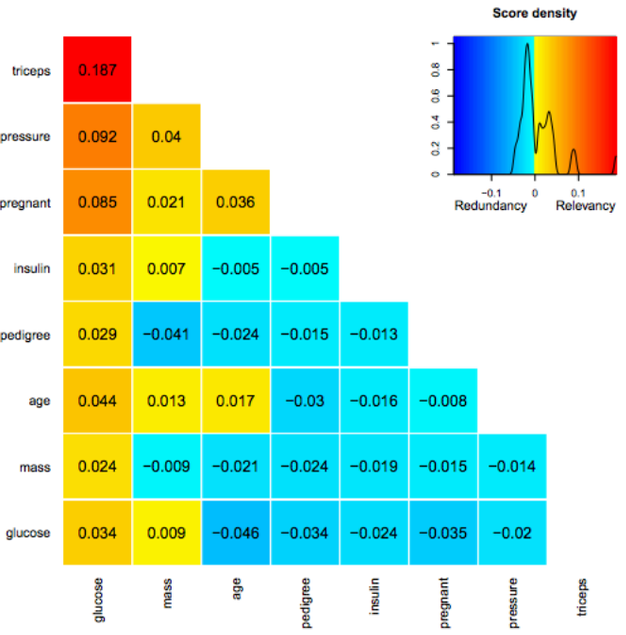
Structure of EPI:

✓ Lie scale (9 responses)

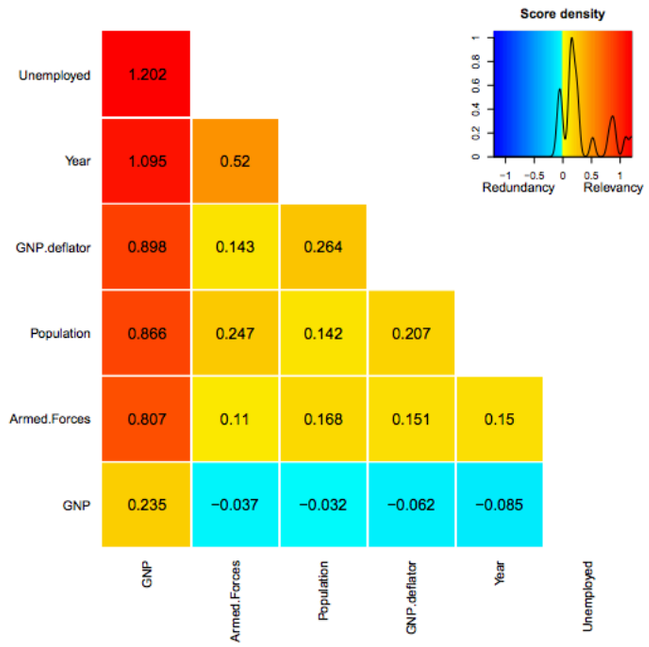




VARRANK



(a) A: Pima Indians Diabetes



(b) B: Longley

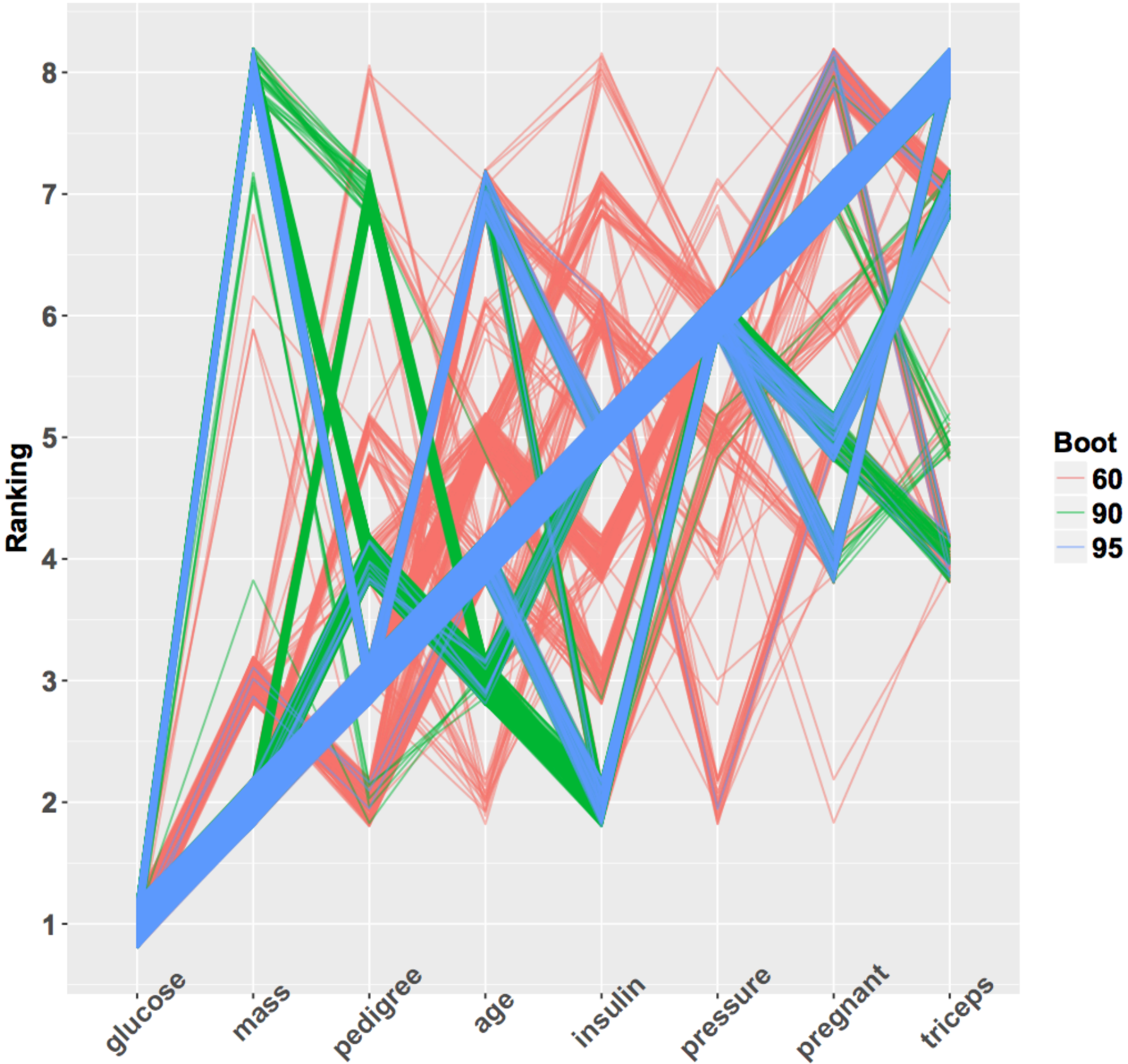
	varrank	caret	Boruta
#1	glucose	glucose	glucose
#2	mass	mass	mass
#3	age	age	age
#4	pedigree	pregnant	pregnant
#5	insulin	pedigree	insulin
#6	pregnant	pressure	pedigree
#7	pressure	triceps	triceps
#8	triceps	insulin	pressure
Bootstrapping 80%	29%	24%	17%
Running time [s]	2.72	4.31	31.22

Table 1: Variable ranking comparison between varrank, caret and Boruta for the Pima Indians Diabetes dataset.

	varrank	caret	Boruta
#1	GNP	GNP	GNP
#2	Armed.Forces	GNP.deflator	Year
#3	Population	Year	GNP.deflator
#4	GNP.deflator	Population	Population
#5	Year	Armed.Forces	Armed.Forces
#6	Unemployed	Unemployed	Unemployed
Bootstrapping 80%	15%	0%	0%
Running time [s]	0.07	0.89	0.57

Table 2: Variable ranking comparison between varrank, caret and Boruta for the Longley dataset.

VARRANK


Ranking
8
7
6
5
4
3
2
1
glucose
mass
pedigree
age
insulin
pressure
pregnant
triceps
Boot
60
90
95

BAYESIAN NETWORKS IN THE MACHINE LEARNING WORLD

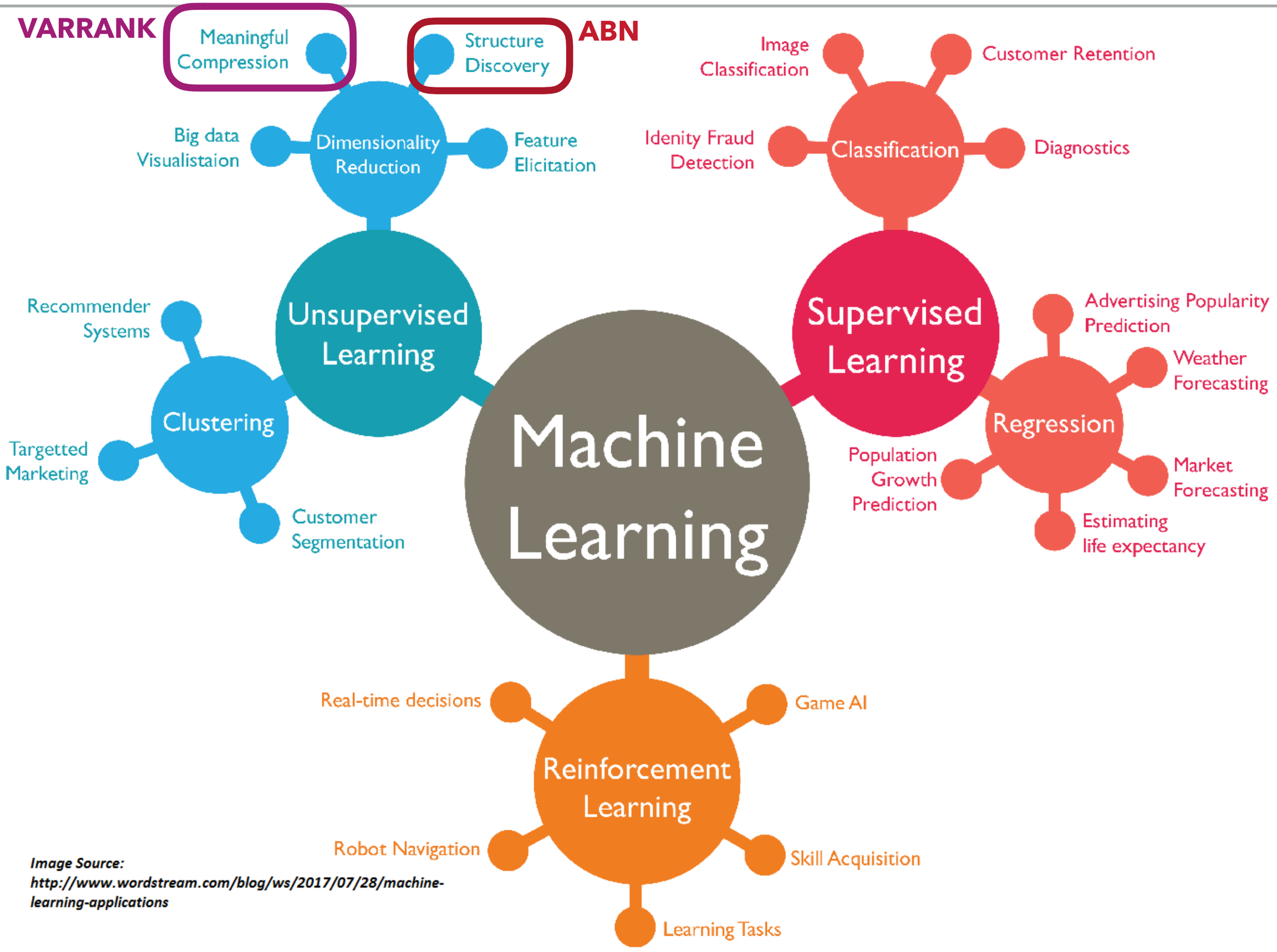


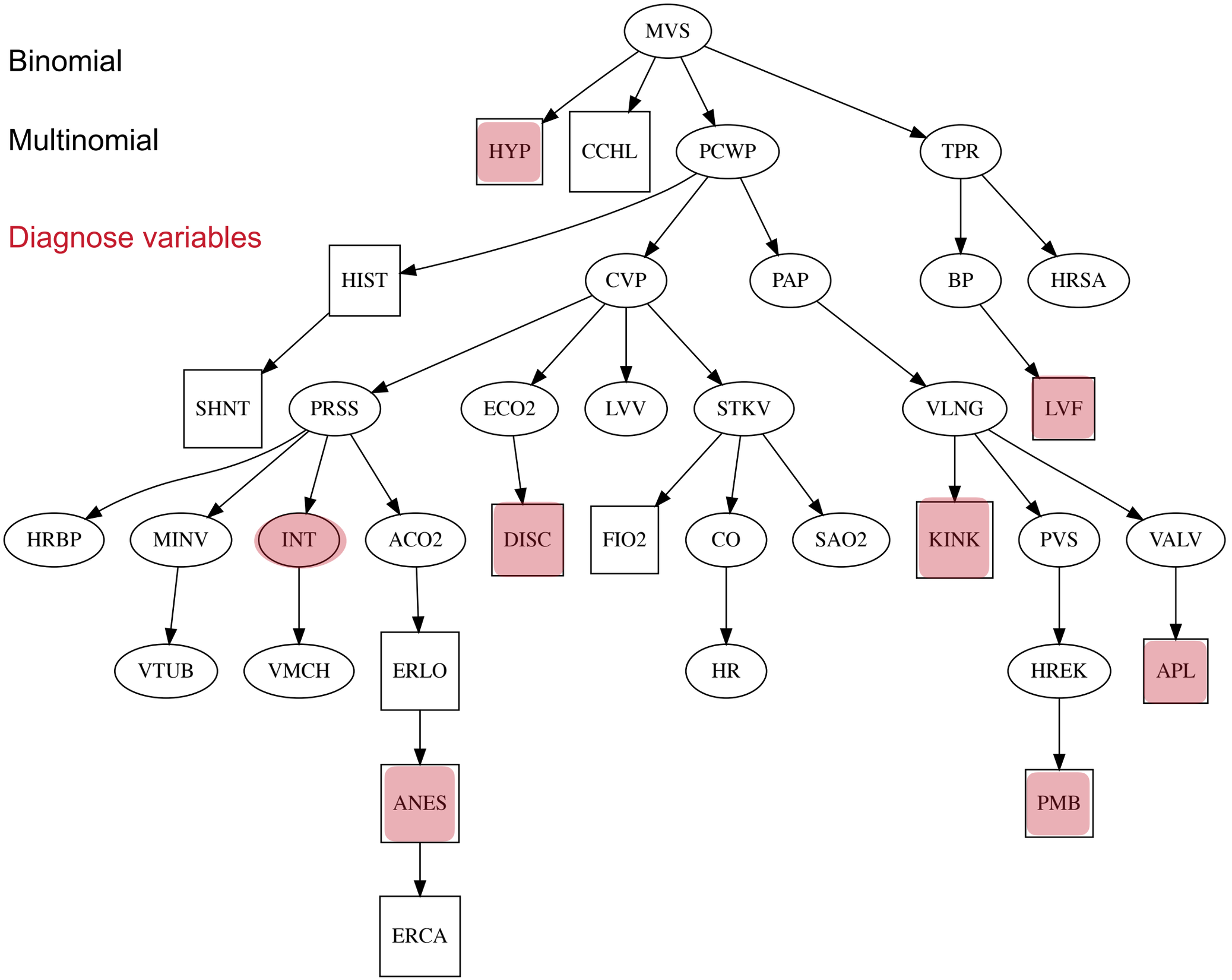
Image Source:
http://www.wordstream.com/blog/ws/2017/07/28/machine-learning-applications

LARGE NETWORK

Binomial

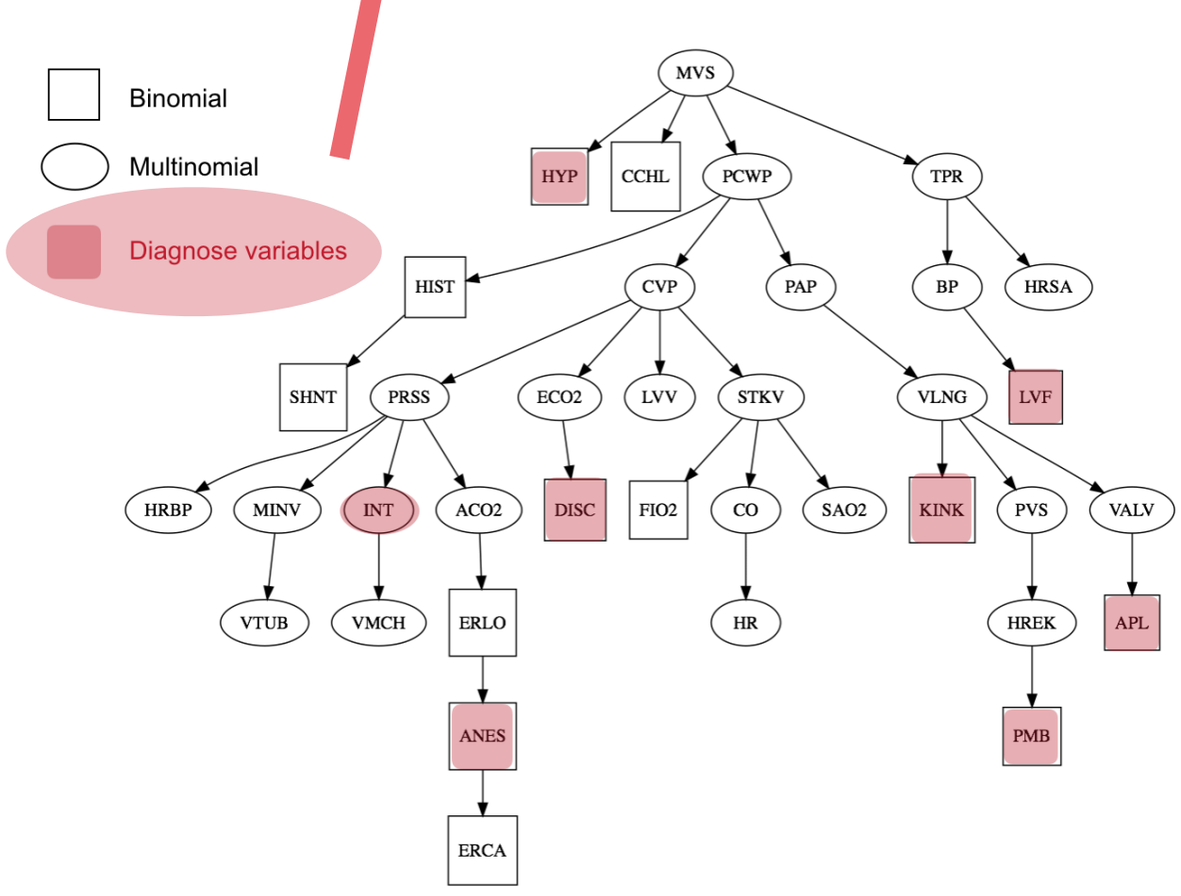
Multinomial

Diagnose variables

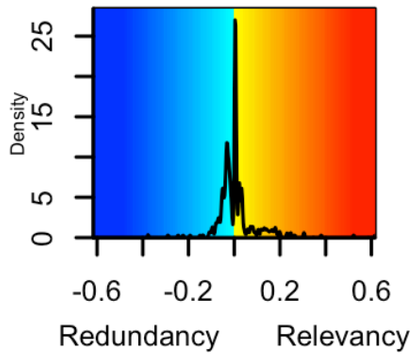


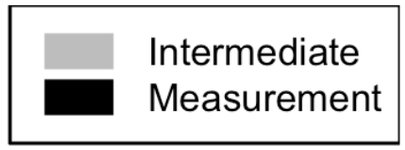
LARGE NETWORK

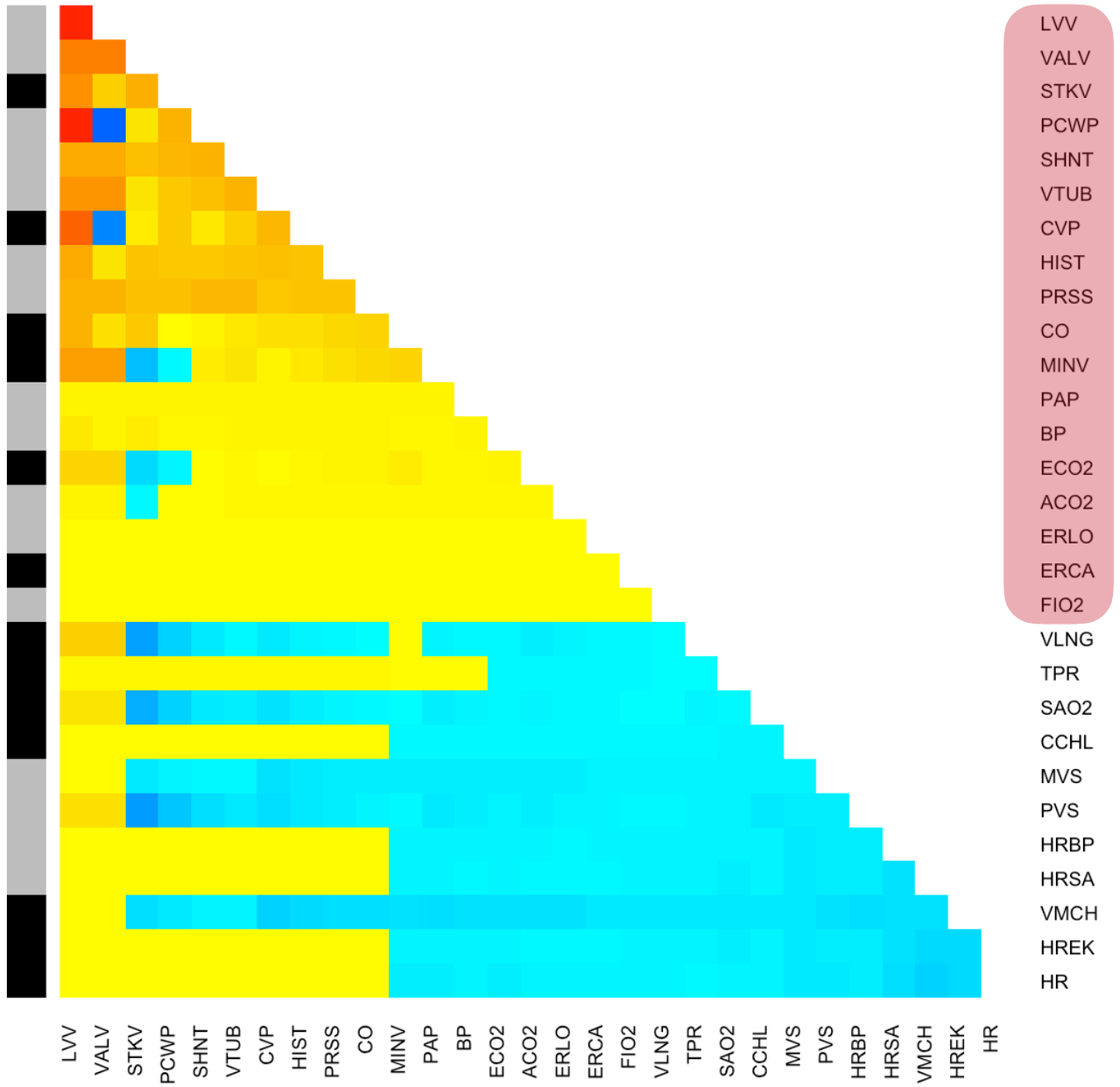
```
var.varrank <- varrank(data.df = alarm,
                       variable.important = c("HYP","LVF","APL","ANES","PMB","INT","KINK","DISC"),
                       method = "peng",
                       algorithm = "forward",
                       scheme = "mid",
                       discretization.method = "fd",
                       verbose = TRUE)

plot(var.varrank)
```



LARGE NETWORK

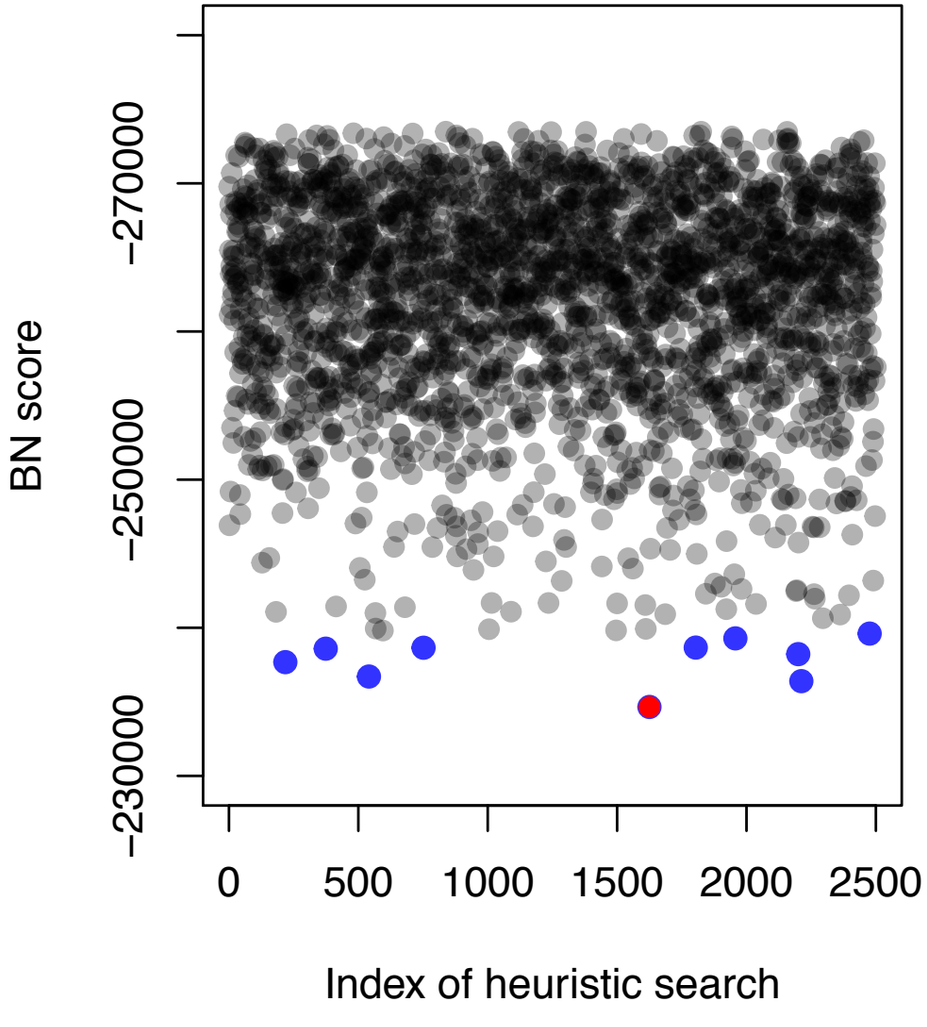


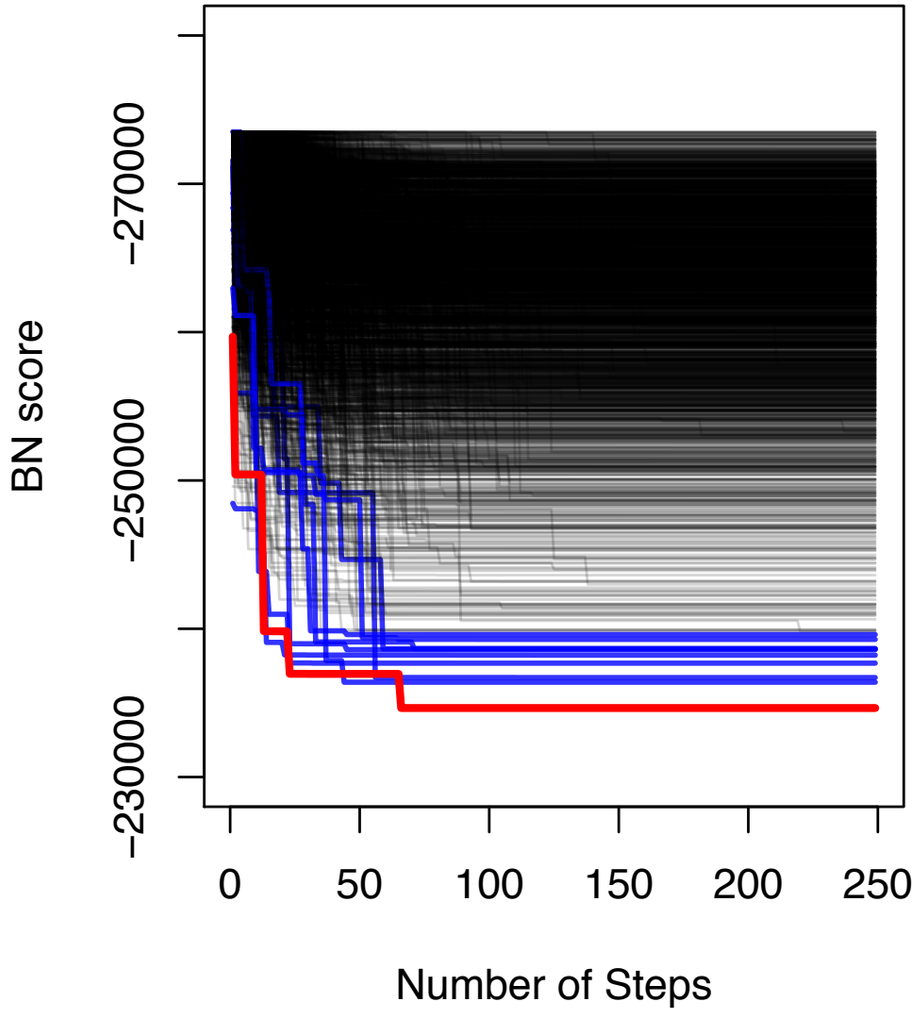




LARGE NETWORK

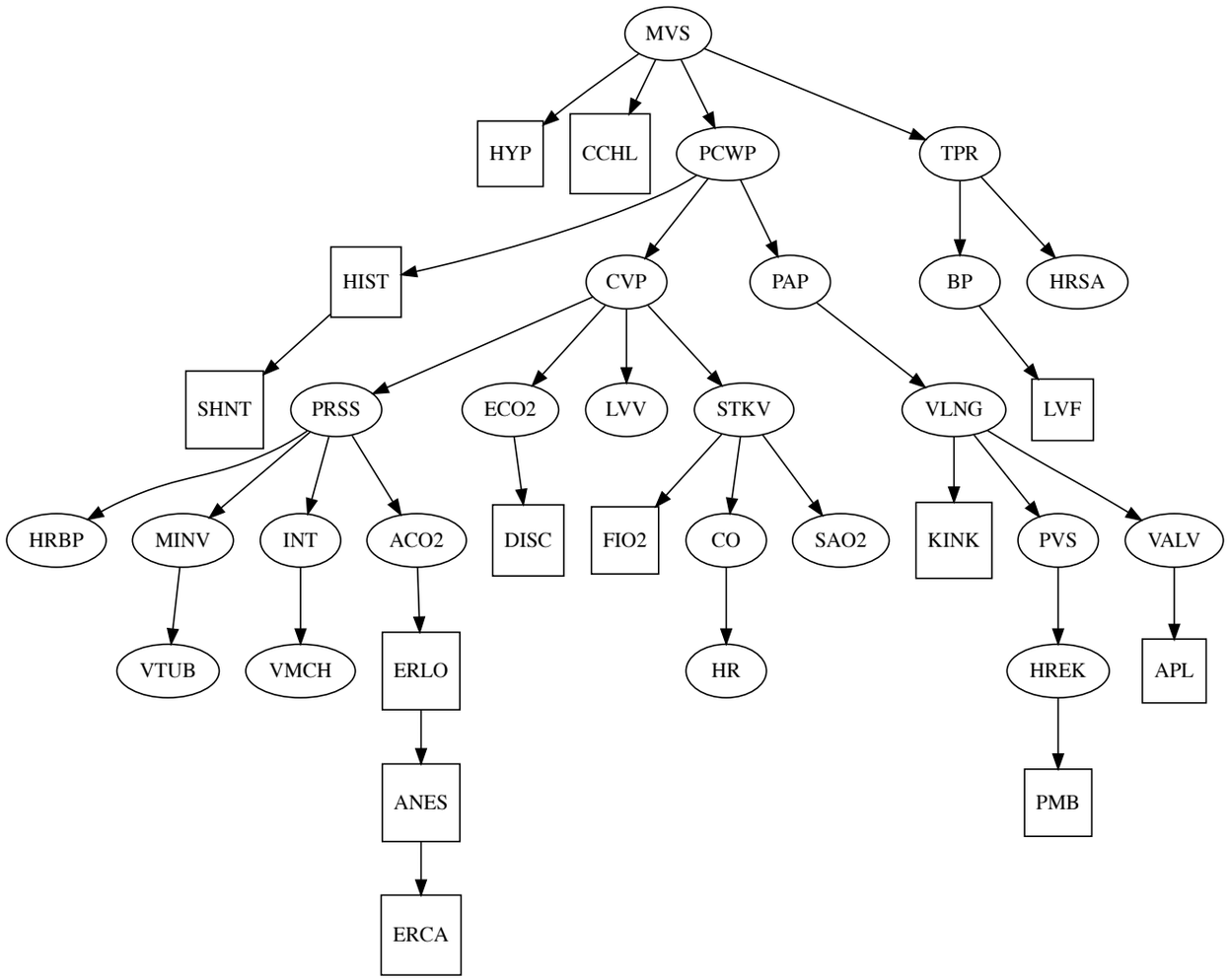
```
bsc.compute <- buildscorecache.mle(data.df = alarm[,sel], data.dists = dist[sel],  max.parents = 1)

bn.hc <- search.heuristic(score.cache = bsc.compute,
                          score = "bic",
                          data.dists = dist[sel],
                          num.searches = 2500,
                          max.steps = 250,
                          start.dag = "random",
                          verbose = TRUE,
                          algo = "hc")
```



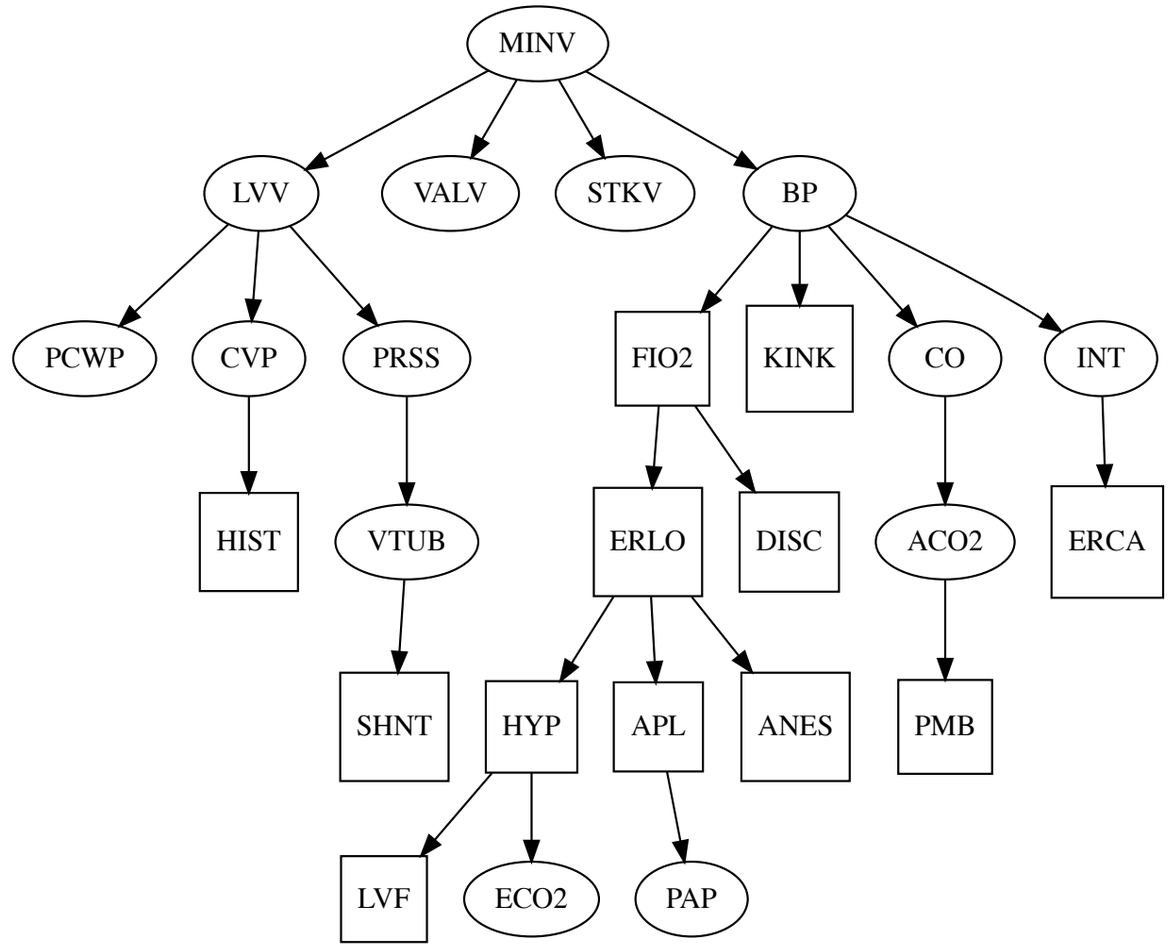


LARGE NETWORK

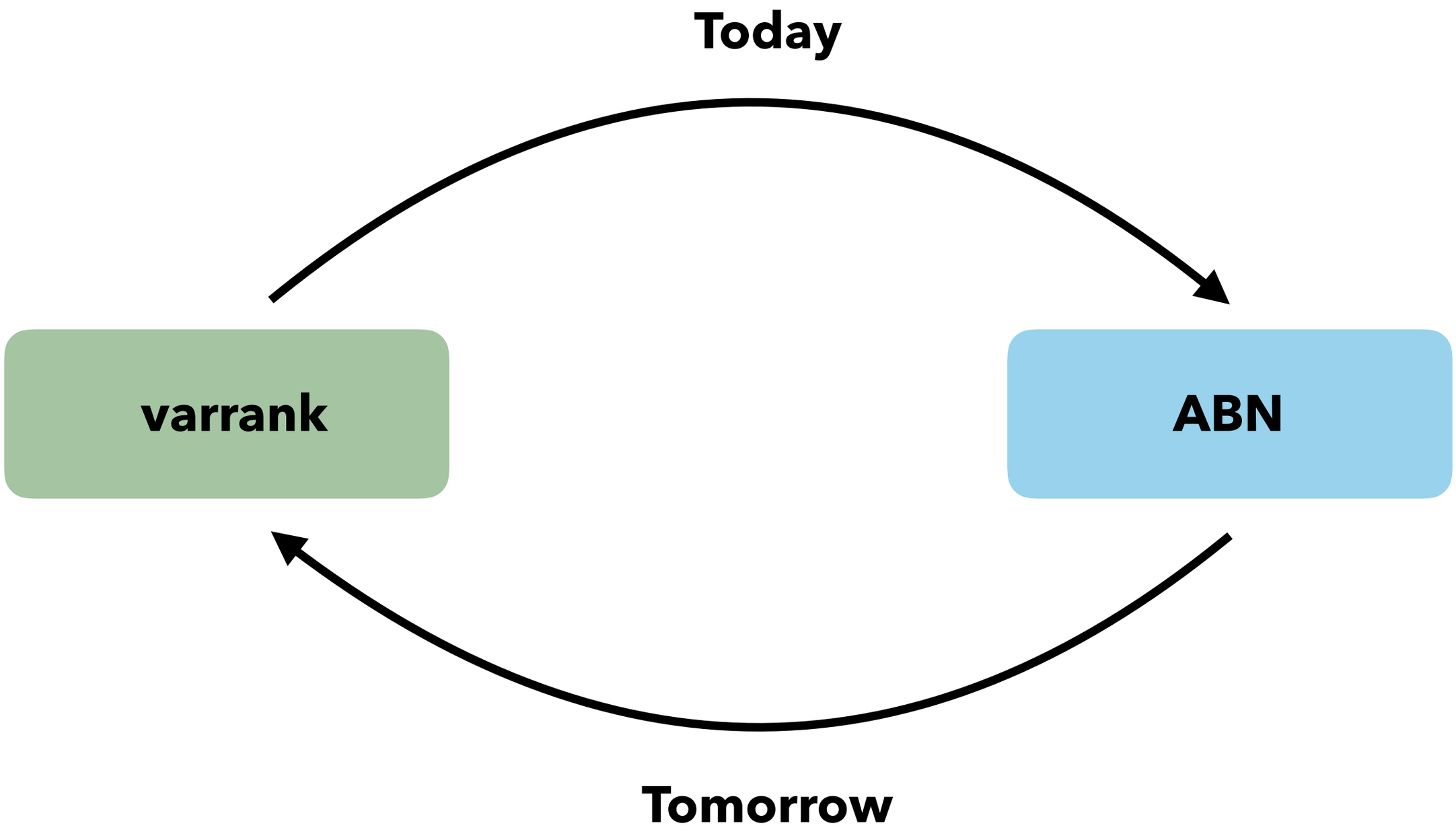
Full set of variables



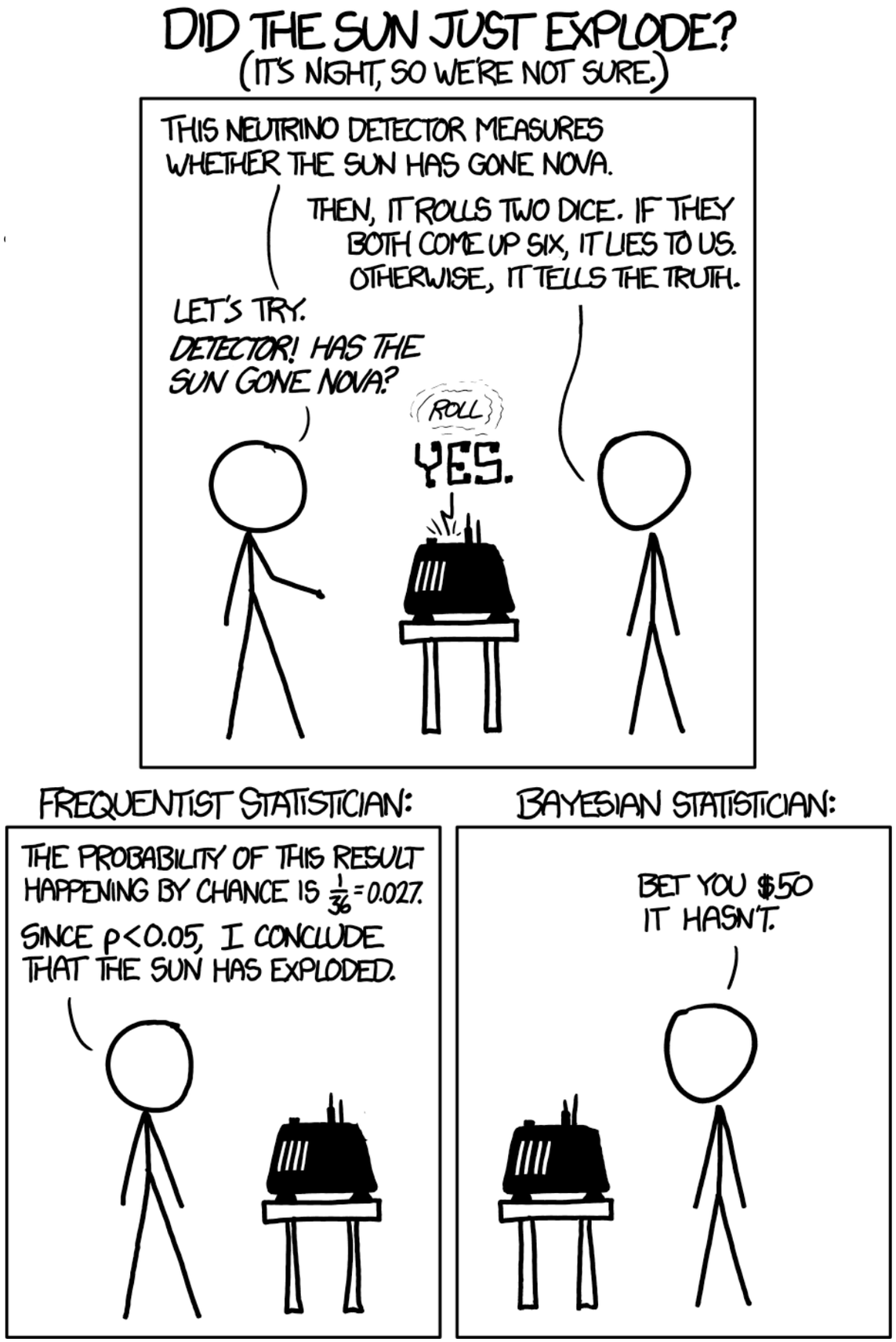
Selected variables



KEY MESSAGE/OUTLOOK



Looking forward for your questions, inputs or remarks ...



xkcd.com

REFERENCES

- Lewis, Fraser I., and Michael P. Ward. "Improving epidemiologic data analyses through multivariate regression modelling." *Emerging themes in epidemiology* 10.1 (2013): 4.
- Lauritzen, Steffen L., and David J. Spiegelhalter. "Local computations with probabilities on graphical structures and their application to expert systems." *Journal of the Royal Statistical Society. Series B (Methodological)* (1988): 157-224.
- M. Scutari. Learning Bayesian Networks with the bnlearn R Package. *Journal of Statistical Software*, 35(3):1-22, 2010.
- I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks. In *Proceedings of the 2nd European Conference on Artificial Intelligence in Medicine*, pages 247-256. Springer-Verlag, 1989.
- Kratzer, Gilles, and Reinhard Furrer. "varrank: an R package for variable ranking based on mutual information with applications to observed systemic datasets." *arXiv preprint arXiv: 1804.07134* (2018).
- Kratzer, Gilles, and Reinhard Furrer. "Information-Theoretic Scoring Rules to Learn Additive Bayesian Network Applied to Epidemiology." *arXiv preprint arXiv:1808.01126* (2018).

Backup slides

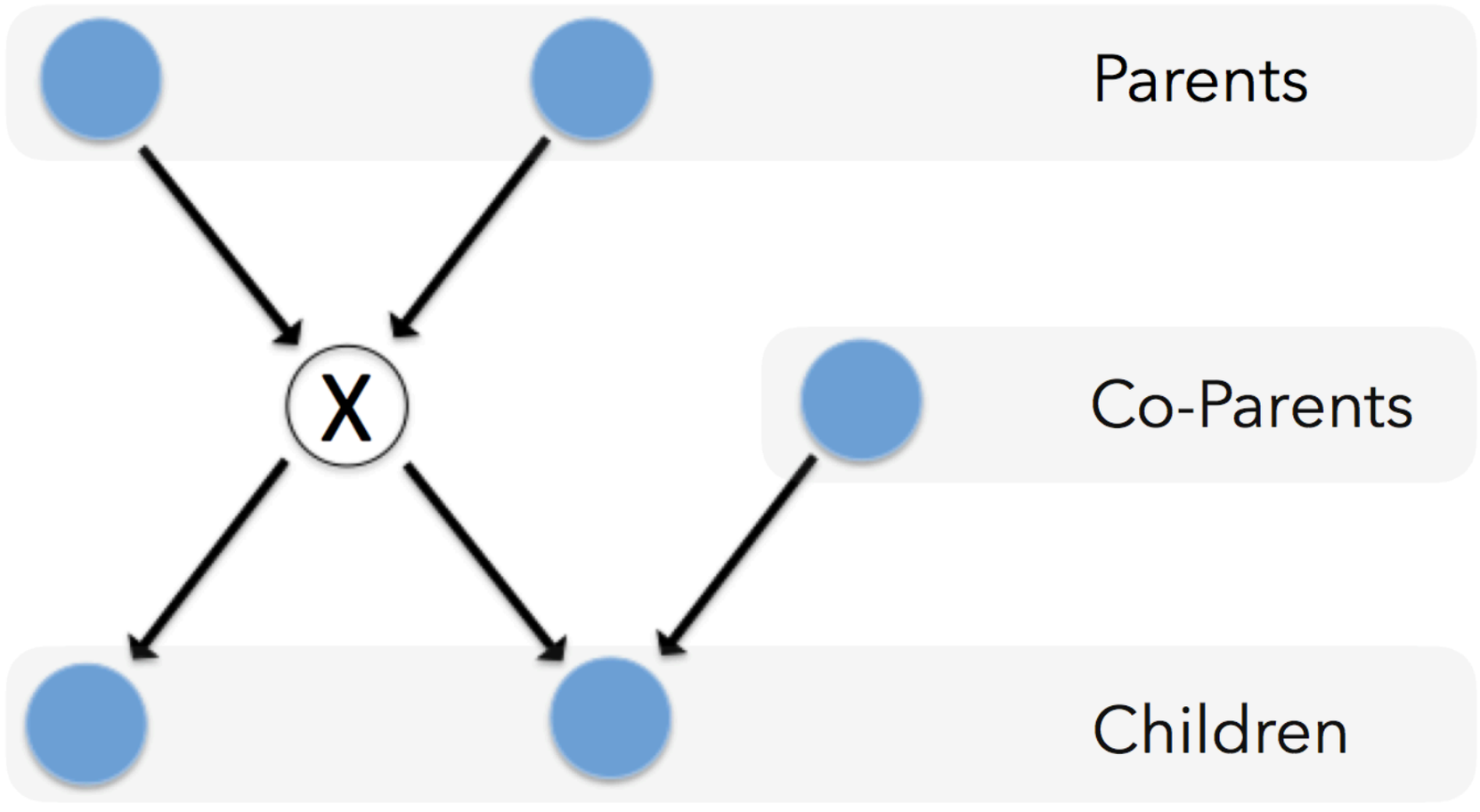
Constraint-based algorithms

- *Inductive Causation* (IC): (Verma and Pearl, 1991)
  - Provides a framework for learning the structure of Bayesian networks using conditional independence tests in three steps
  - A major problem of the IC algorithm is that the first two steps cannot be applied to any real-world problem due to computational complexity …

- *PC*: first practical application of the IC algorithm (Spirtes et al., 2001)
  - backward selection procedure from the saturated graph

- *Grow-Shrink* (GS) (Margaritis, 2003)
  - Simple forward selection MB detection approach

The Markov Blanket of a node is the set of parents, co-parents and children.



$$P(X_k \mid X_n, k \neq n) = P(X_k \mid X_{\mathrm{MB}(k)}), \forall k$$

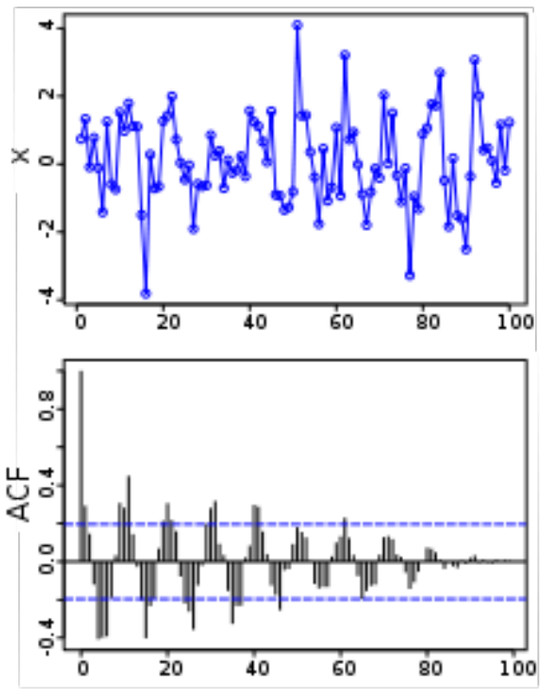
Constraint-based algorithms

- *Inductive Causation* (IC): (Verma and Pearl, 1991)
  - Provides a framework for learning the structure of Bayesian networks using conditional independence tests in three steps
  - A major problem of the IC algorithm is that the first two steps cannot be applied to any real-world problem due to computational complexity …

- *PC*: first practical application of the IC algorithm (Spirtes et al., 2001)
  - backward selection procedure from the saturated graph

- *Grow-Shrink* (GS) (Margaritis, 2003)
  - Simple forward selection MB detection approach

TSABN



Time series regression

- OLS estimates
- Goodness of fit metrics

Variance-Covariance

$$\begin{pmatrix} \sigma^2_{y_1} & \sigma_{y_1 y_2} & \cdots & \sigma_{y_1 y_n} \\ \sigma_{y_1 y_2} & \sigma^2_{y_2} & \cdots & \sigma_{y_2 y_n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{y_1 y_n} & \sigma_{y_2 y_n} & \cdots & \sigma^2_{y_n} \end{pmatrix}$$

tsabn as a time series extension of abn

- Extending ABN to correlated errors
- Several implemented scores: AIC, BIC, MDL
- Errors Autocorrelation: ARMA procedure with Autoregressive modelling
  - Kalman filter

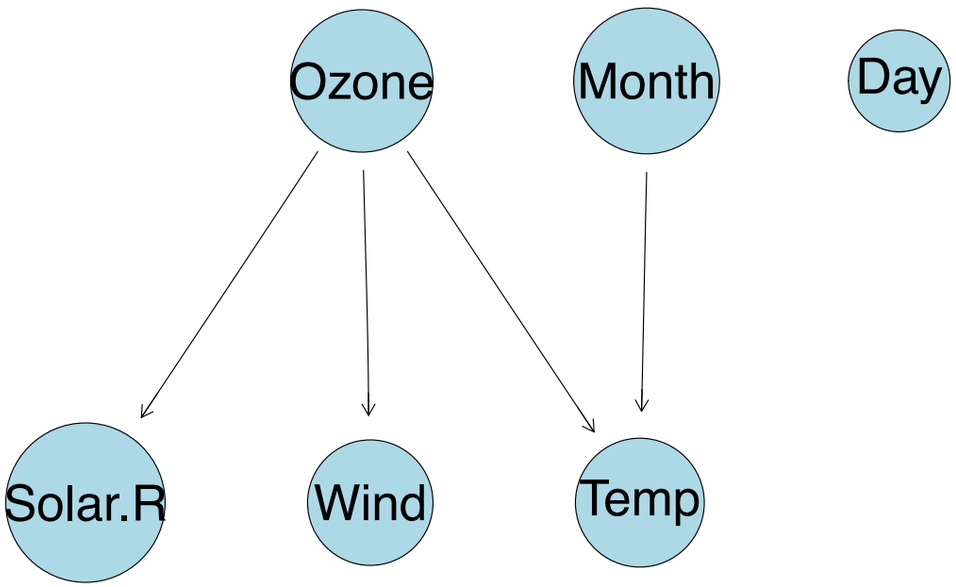
Future work

- Implementation of Granger causality score for BN learning

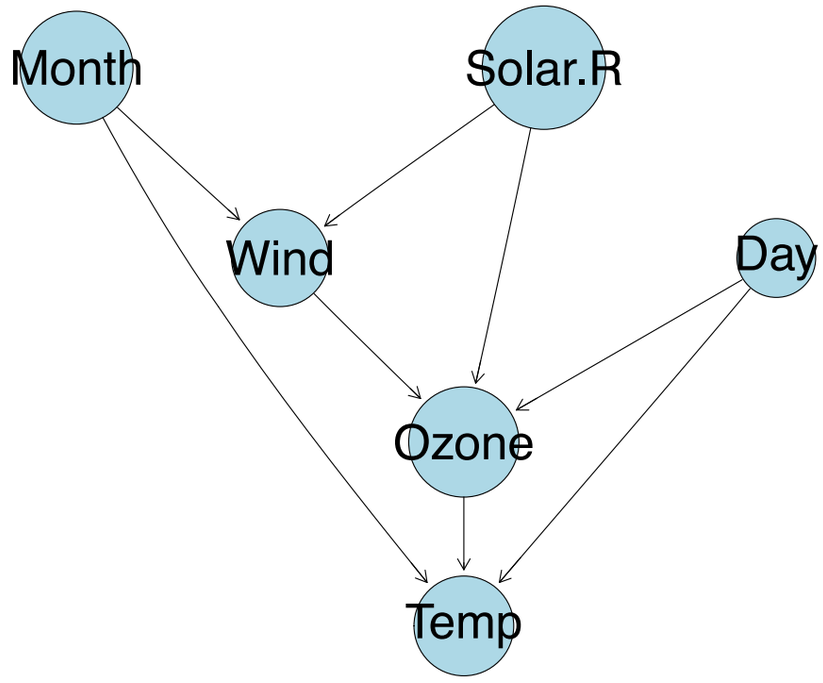
OZONE DATASET

Daily readings of the air quality values from May to September 1973

111 observations on 6 variables



BIC with AR



Future work:

Hourly readings of the PM2.5 and 6 other chemical compounds data of US embassy in Beijing with meteorological data from Beijing Capital International Airport from 2013 to 2017

- Constraint-based methods require a Markov and faithfulness assumption
- Conditional independencies in the distribution exactly equal the ones encoded in the DAG via d-separation

$$A \perp\!\!\!\perp_G B|C \overset{\text{Markov}}{\underset{\text{Faithful}}{\rightleftarrows}} A \perp\!\!\!\perp_P B|C$$

- Causal sufficiency: no unmeasured common causes

In a pratical perspective:

- Testing mixture of data?
- Testing assumptions?

SOME ELEMENTS OF PROBABILITY THEORY

The conditional probability of A given B is: $$P(A \mid B) = \frac{P(A, B)}{P(B)}$$

Bayes theorem: $$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Let A, B and C non intersecting subsets of nodes in a DAG G

A is conditionally independent of B given C if: $A \perp\!\!\!\perp_P B | C$

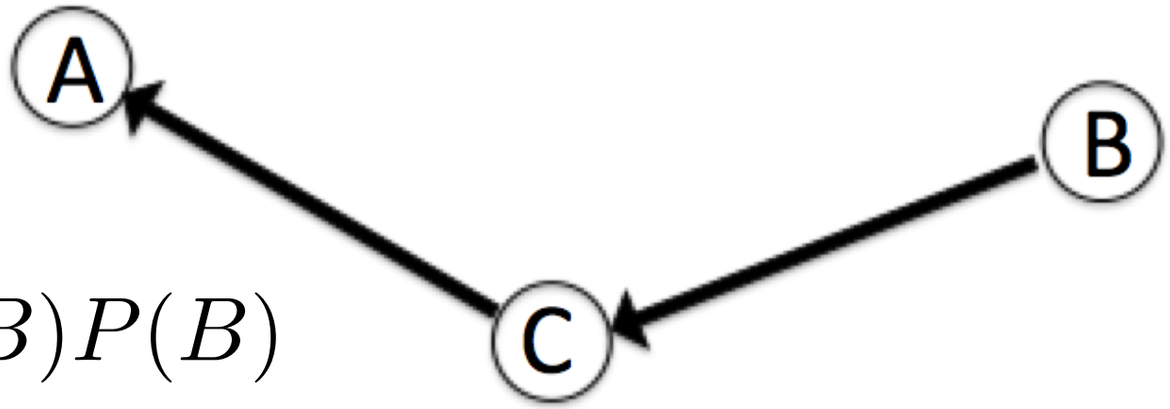
$$P(A, B \mid C) = P(A \mid C)P(B \mid C)$$

ELEMENT OF GRAPH THEORY

Let A, B and C non intersecting subsets of nodes in a DAG G

A is conditionally independent of B given C if: $A \perp\!\!\!\perp_P B | C$

$$P(A, B \mid C) = P(A \mid C)P(B \mid C)$$



$$P(A, B, C) = P(A \mid C)P(C \mid B)P(B)$$

$$P(A, B \mid C) = \frac{P(A \mid C)P(C \mid B)P(B)}{P(C)}$$

$$= \frac{P(A \mid C)P(B, C)}{P(C)}$$

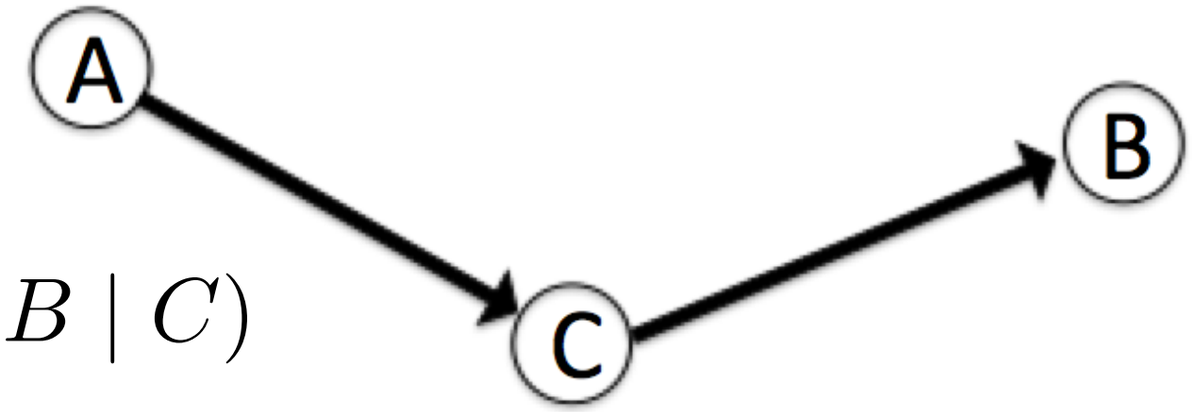
$$= P(A \mid C)P(B \mid C)$$

ELEMENT OF GRAPH THEORY

Let A, B and C non intersecting subsets of nodes in a DAG G

A is conditionally independent of B given C if: $A \perp\!\!\!\perp_P B | C$

$$P(A, B \mid C) = P(A \mid C)P(B \mid C)$$



$$P(A, B, C) = P(A)P(C \mid A)P(B \mid C)$$

$$P(A, B \mid C) = \frac{P(A)P(C \mid A)P(B \mid C)}{P(C)}$$

$$= \frac{P(A, C)P(B \mid C)}{P(C)}$$

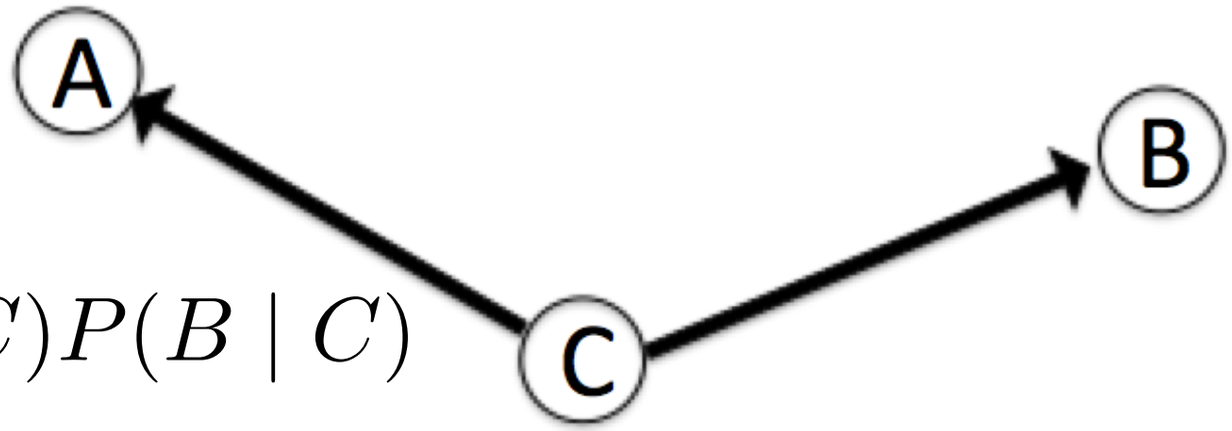
$$= P(A \mid C)P(B \mid C)$$

ELEMENT OF GRAPH THEORY

Let A, B and C non intersecting subsets of nodes in a DAG G

A is conditionally independent of B given C if: $A \perp\!\!\!\perp_P B | C$

$$P(A, B \mid C) = P(A \mid C)P(B \mid C)$$



$$P(A, B, C) = P(C)P(A \mid C)P(B \mid C)$$

$$P(A, B \mid C) = \frac{P(C)P(A \mid C)P(B \mid C)}{P(C)}$$

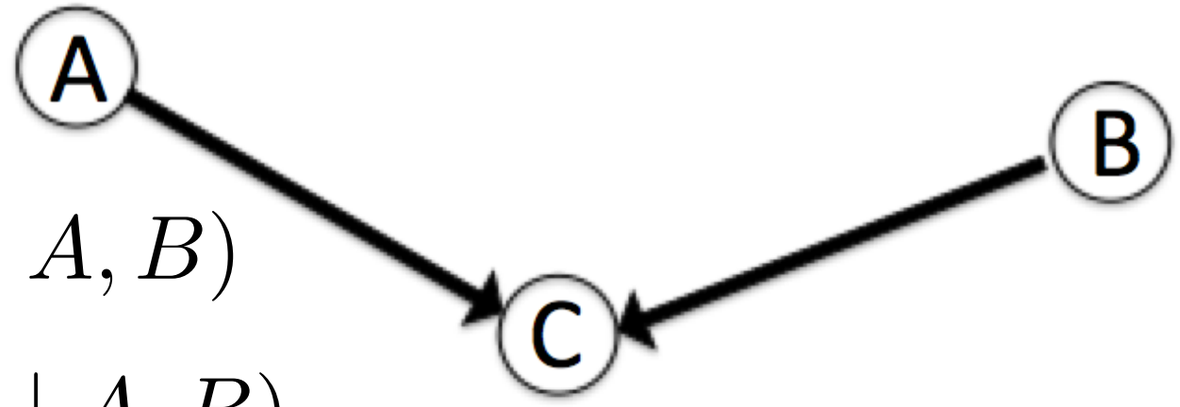
$$= P(A \mid C)P(B \mid C)$$

ELEMENT OF GRAPH THEORY

Let A, B and C non intersecting subsets of nodes in a DAG G

A is conditionally independent of B given C if: $A \perp\!\!\!\perp_P B|C$

$$P(A, B \mid C) = P(A \mid C)P(B \mid C)$$



$$P(A, B, C) = P(A)P(B)P(C \mid A, B)$$

$$P(A, B \mid C) = \frac{P(A)P(B)P(C \mid A, B)}{P(C)}$$

$$= \frac{P(A)P(B)P(A, B, C)}{P(A)P(B)P(C)}$$

$$= P(A, B \mid C)$$

$A \not\perp\!\!\!\perp_P B|C$

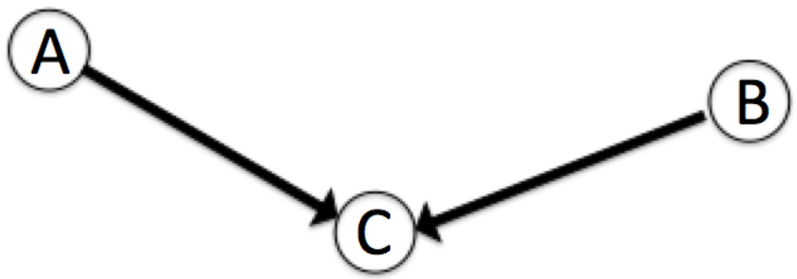
ELEMENT OF GRAPH THEORY

Let A, B and C non intersecting subsets of nodes in a DAG G

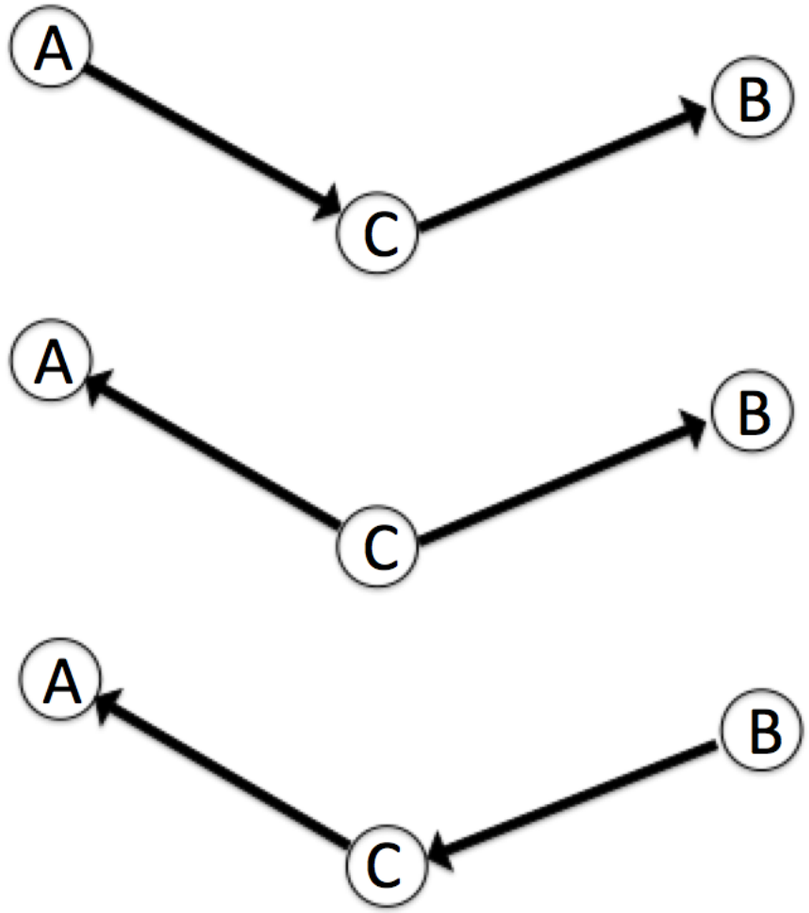
A is conditionally independent of B given C if: $A \perp\!\!\!\perp_P B|C$

$$P(A, B \mid C) = P(A \mid C)P(B \mid C)$$

$A \not\perp\!\!\!\perp_P B|C$



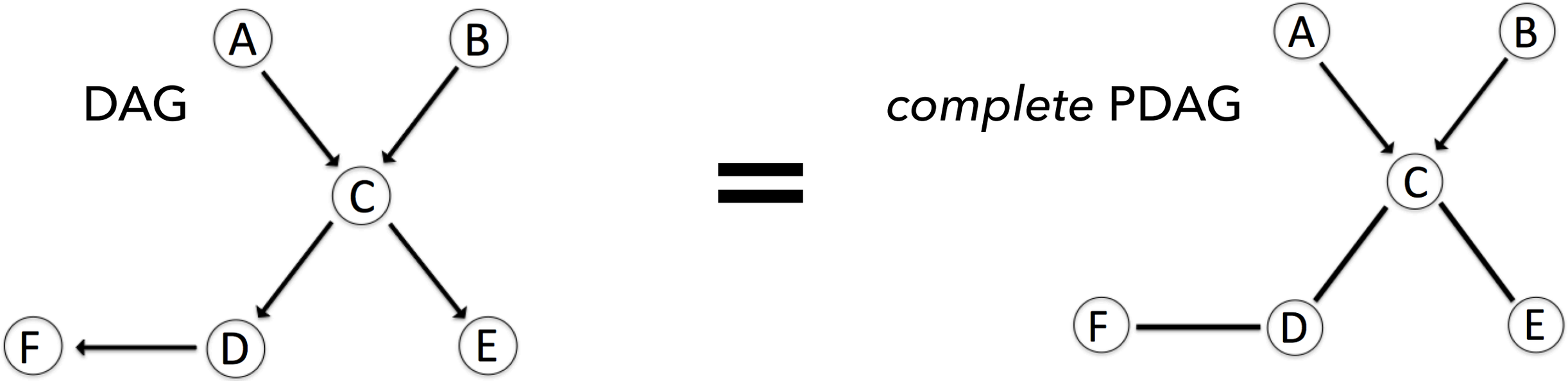
$A \perp\!\!\!\perp_P B|C$



LEARNING BAYESIAN NETWORKS

- In a practical perspective, for observational data, if learning algorithms rely on probabilistic learning algorithm. Then one can learn up to the Markov equivalence class.

- Markov equivalence class are the set of DAGs that have the same skeleton and v-structure.



A path from A to B is blocked if it contains a node s.t. either

- the arrows on the path meet either head-to-tail or tail-to-tail at the node, and the node is in the set C, or
- the arrows meet head-to-head at the node, and neither the node, nor any of its descendants, are C.

If all paths from A to B are blocked, A is said to be d-separated from B by C.

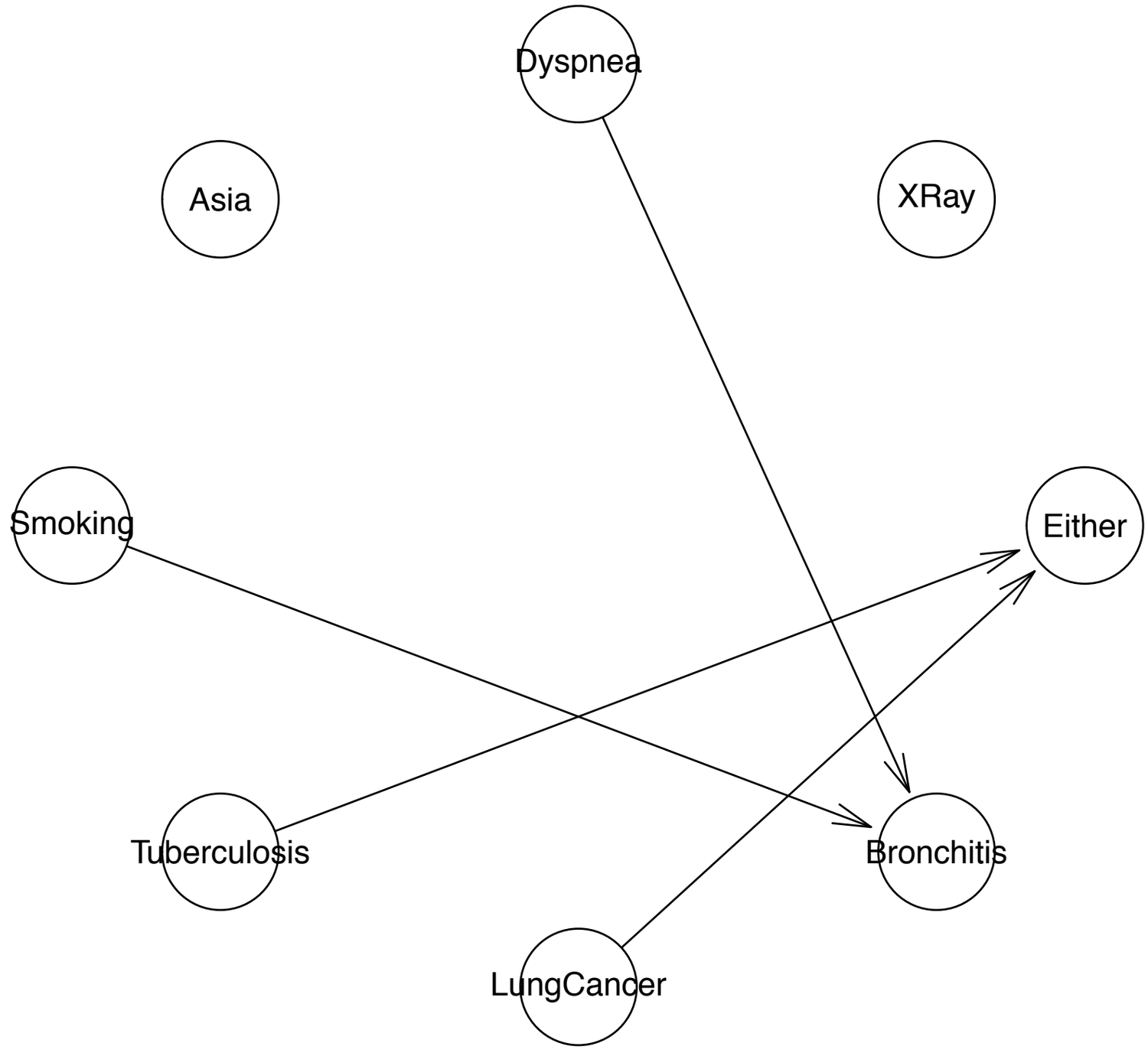
Theorem (Verma & Pearl, 1988): A is d-separated from B by C if, and only if, the joint distribution over all variables in the graph satisfies:

$$A \perp\!\!\!\perp_G B | C$$

Link between statistical statement (conditionally independent) and a graph propriety (d-separation)

ASIA: CONSTRAINT-BASED LEARNING

```
##============================
## constraint-based algorithm
##============================

bn.gs <- gs(asia)
plot(bn.gs)

bn.iamb <- iamb(asia)
plot(bn.iamb)
```



That is all folks!