Stochastic Systems Group

# Consistent and Efficient Reconstruction of Latent Tree Models

Myung Jin Choi

Joint work with
Vincent Tan, Anima Anandkumar, and Alan S. Willsky

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology

September 29, 2010

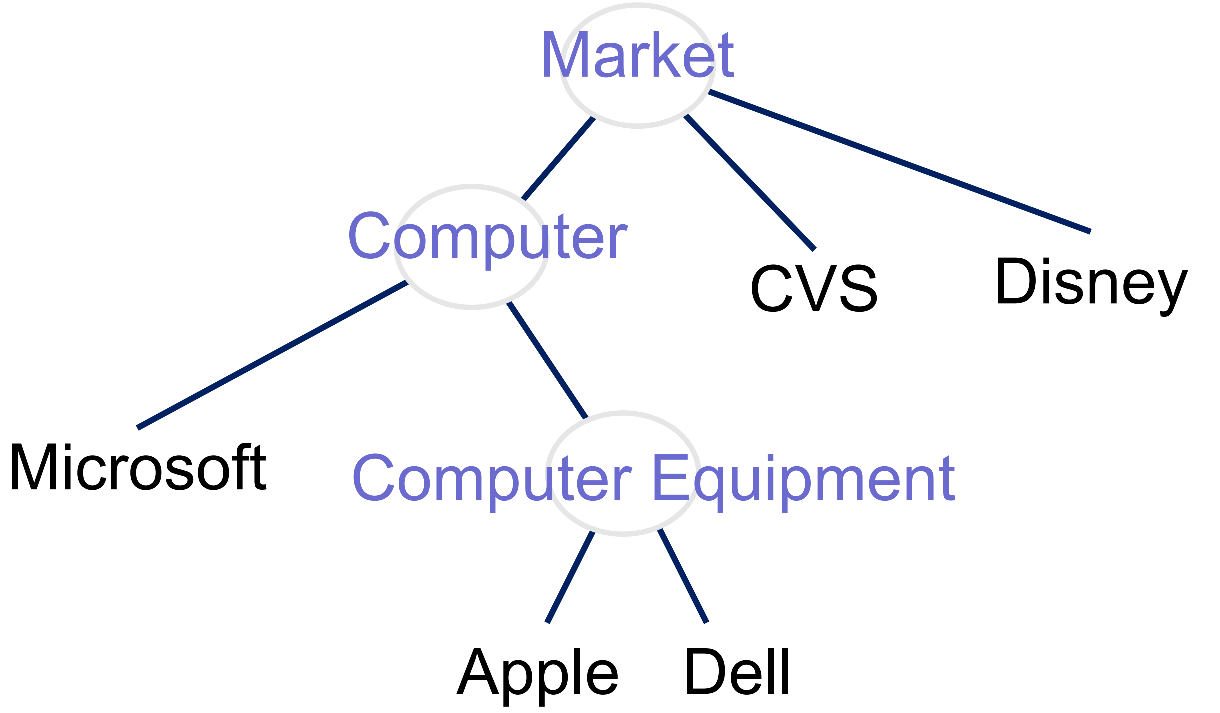
# Latent Tree Graphical Models

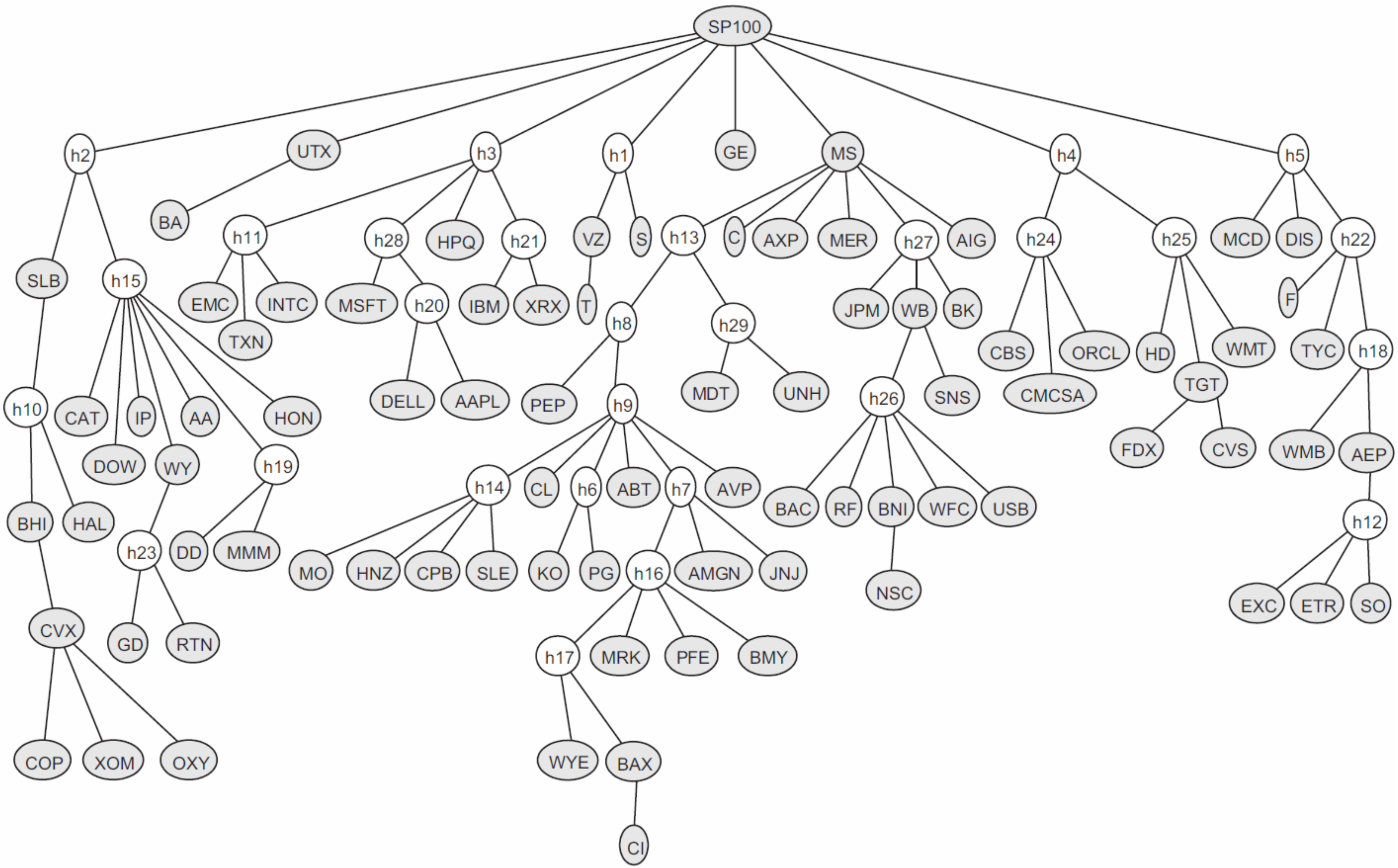
CVS Disney

Microsoft

Apple Dell

# Latent Tree Graphical Models




SP100
h2
UTX
h3
h1
GE
MS
h4
h5
BA
SLB
h15
h11
h28
HPQ
h21
VZ
S
h13
C
AXP
MER
h27
AIG
h24
h25
MCD
DIS
h22
EMC
INTC
TXN
MSFT
h20
IBM
XRX
T
h8
h29
JPM
WB
BK
F
CBS
ORCL
CMCSA
HD
WMT
TYC
h18
TGT
h10
CAT
IP
AA
HON
DOW
WY
h19
DELL
AAPL
PEP
h9
MDT
UNH
h26
SNS
FDX
CVS
WMB
AEP
BHI
HAL
h23
DD
MMM
h14
CL
h6
ABT
h7
AVP
BAC
RF
BNI
WFC
USB
h12
MO
HNZ
CPB
SLE
KO
PG
h16
AMGN
JNJ
NSC
EXC
ETR
SO
CVX
GD
RTN
h17
MRK
PFE
BMY
COP
XOM
OXY
WYE
BAX
CI

# Outline

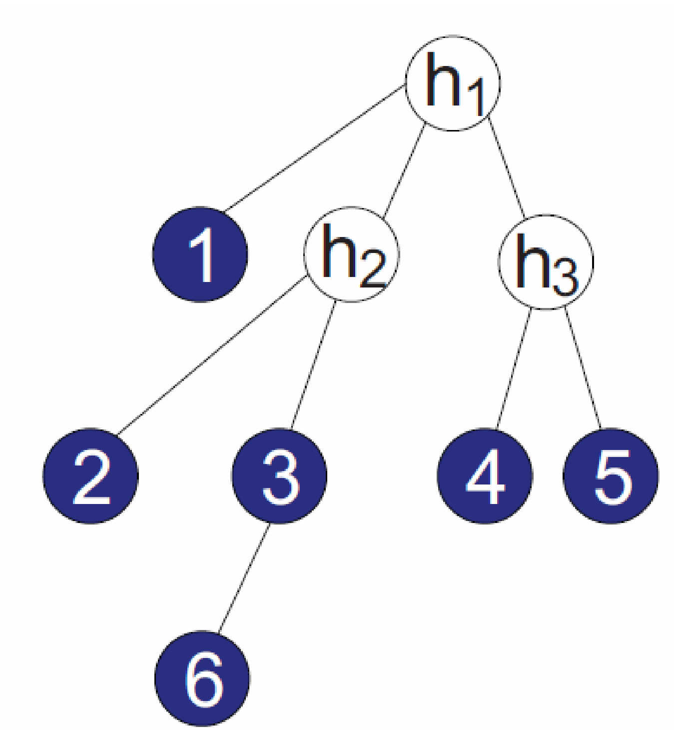
– Reconstruction of a latent tree

– Algorithm 1: Recursive Grouping

– Algorithm 2: CLGrouping

– Experimental results

# Reconstruction of a Latent Tree




- Gaussian model:

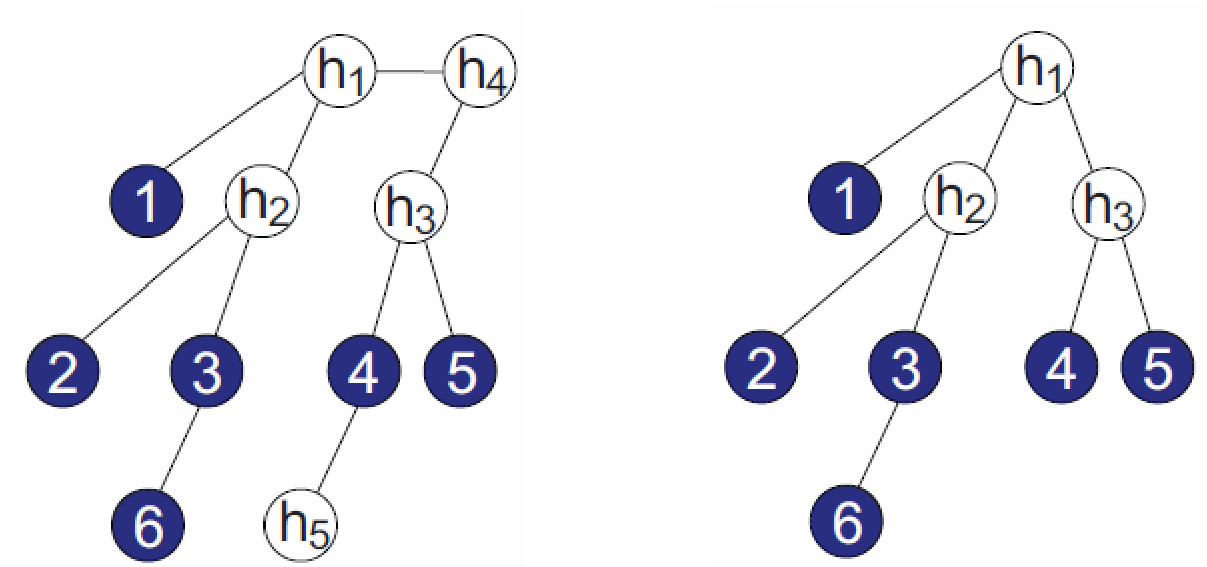
  each node – a scalar Gaussian variable

- Discrete model:

  each node – a discrete variable with K states

Reconstruct a latent tree using samples of the observed nodes.

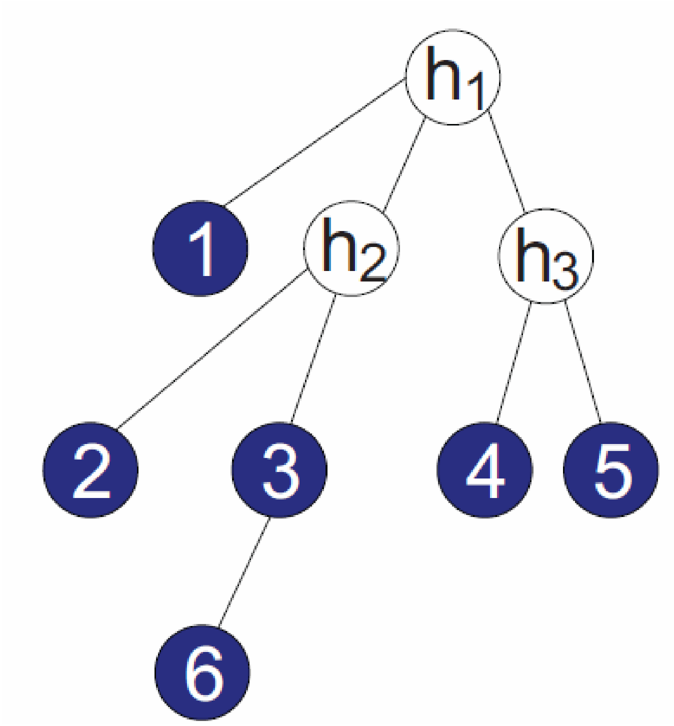
# Minimal Latent Trees (Pearl, 1988)




Conditions for Minimal Latent Trees

- Each hidden node should have at least three neighbors.
- Any two variables are neither perfectly dependent nor independent.

# Desired Properties for Algorithms




1. Consistent for minimal latent trees

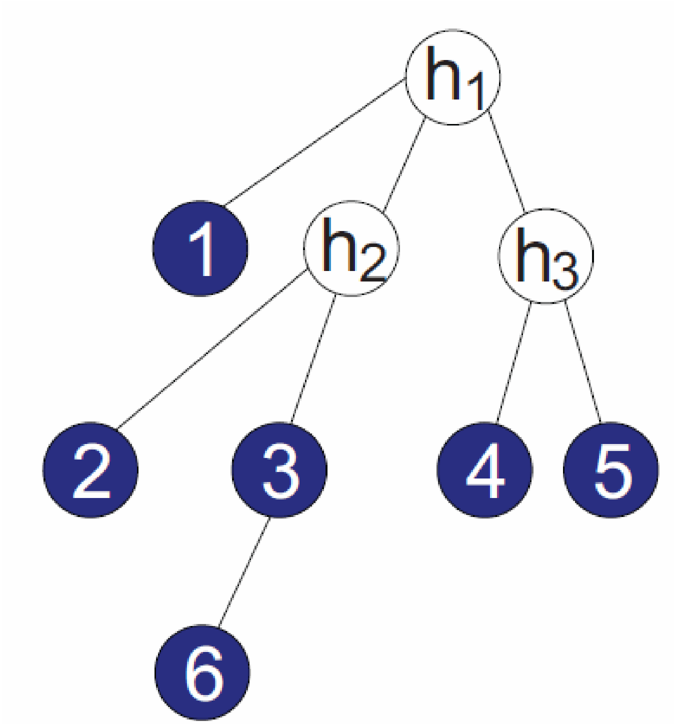
   ⇨ Correct recovery given exact distributions.

2. Computationally efficient

3. Low sample complexity

4. Good empirical performance

# Desired Properties for Algorithms




1. Consistent for minimal latent trees

   ⇨ Correct recovery given exact distributions.

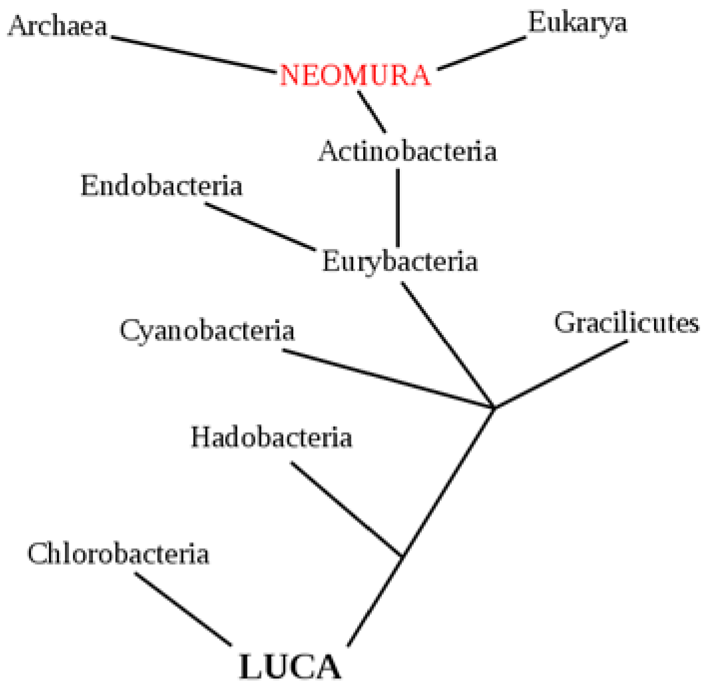
2. Computationally efficient

3. Low sample complexity

4. Good empirical performance

# Related Work

- EM-based approaches
  - ZhangKocka04, HarmelingWilliams10, ElidanFriedman05
  - No consistency guarantees
  - Computationally expensive
- Phylogenetic trees
  - Neighbor-joining (NJ) method (SaitouNei87)



# Information Distance

- Gaussian distributions

$$d_{ij} := -\log|\rho_{ij}|$$

$$\rho_{ij} := \frac{\mathrm{Cov}(X_i, X_j)}{\sqrt{\mathrm{Var}(X_i)\mathrm{Var}(X_j)}}$$

# Information Distance

- Gaussian distributions

$$d_{ij} := -\log|\rho_{ij}| \qquad \rho_{ij} := \frac{\mathrm{Cov}(X_i, X_j)}{\sqrt{\mathrm{Var}(X_i)\mathrm{Var}(X_j)}}$$

- Discrete distributions

$$d_{ij} := -\log \frac{|\det \mathbf{J}^{ij}|}{\sqrt{\det \mathbf{M}^i \det \mathbf{M}^j}}$$

$\mathbf{J}^{ij}$ Joint probability matrix $\mathbf{M}^i$ Marginal probability matrix

ex)
$$\mathbf{J}^{ij} = \begin{pmatrix} p(x_i = 0, x_j = 0) & p(x_i = 0, x_j = 1) \\ p(x_i = 1, x_j = 0) & p(x_i = 1, x_j = 1) \end{pmatrix}$$

$$\mathbf{M}^i = \begin{pmatrix} p(x_i = 0) & 0 \\ 0 & p(x_i = 1) \end{pmatrix}$$

# Information Distance

- Gaussian distributions

$$d_{ij} := -\log |\rho_{ij}|$$

$$\rho_{ij} := \frac{\mathrm{Cov}(X_i, X_j)}{\sqrt{\mathrm{Var}(X_i)\mathrm{Var}(X_j)}}$$

- Discrete distributions

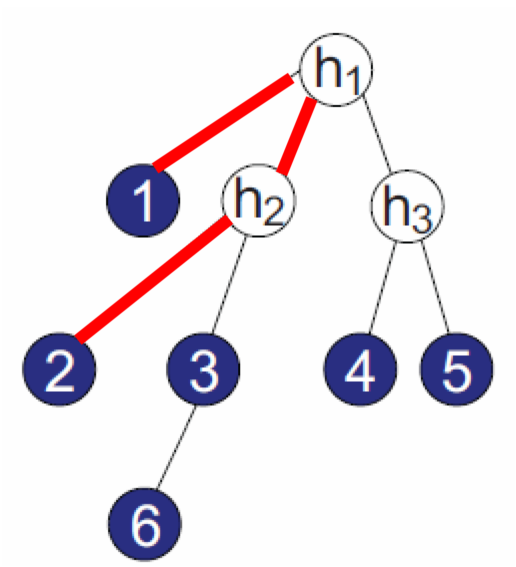
$$d_{ij} := -\log \frac{|\det \mathbf{J}^{ij}|}{\sqrt{\det \mathbf{M}^i \det \mathbf{M}^j}}$$

$\mathbf{J}^{ij}$ Joint probability matrix $\mathbf{M}^i$ Marginal probability matrix

- Algorithms use information distances of observed variables.
- Assume first that the exact information distances are given.

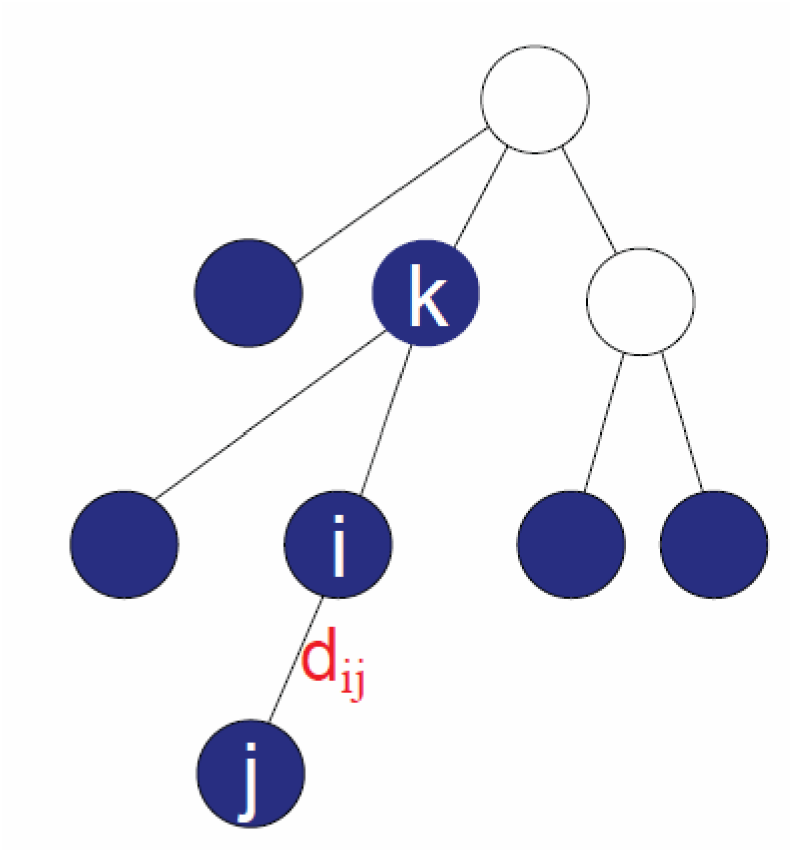
# Additivity of Information Distances on Trees

$$d_{k,l} = \sum_{(i,j) \in \mathrm{Path}((k,l);E_p)} d_{i,j}$$




$$d_{12} = d_{1h_1} + d_{h_1h_2} + d_{2h_2}$$

# Testing Node Relationships




Node j – a leaf node

Node i – parent of j

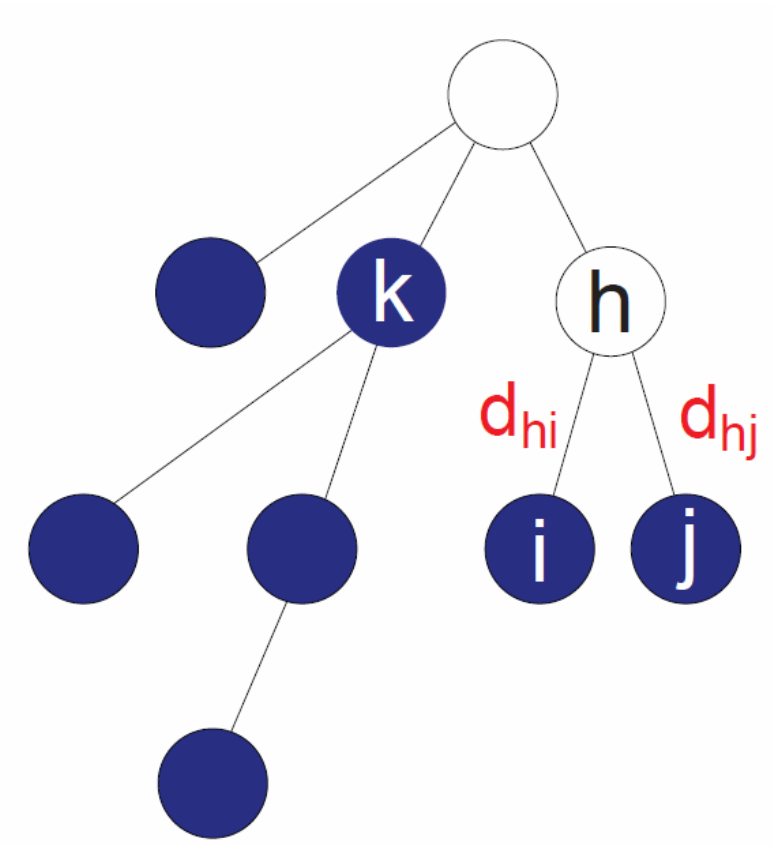
$$\Leftrightarrow d_{jk} - d_{ik} = d_{ij}$$

for all k $\neq$ i, j .

Can identify

(parent, leaf child) pair

# Testing Node Relationships



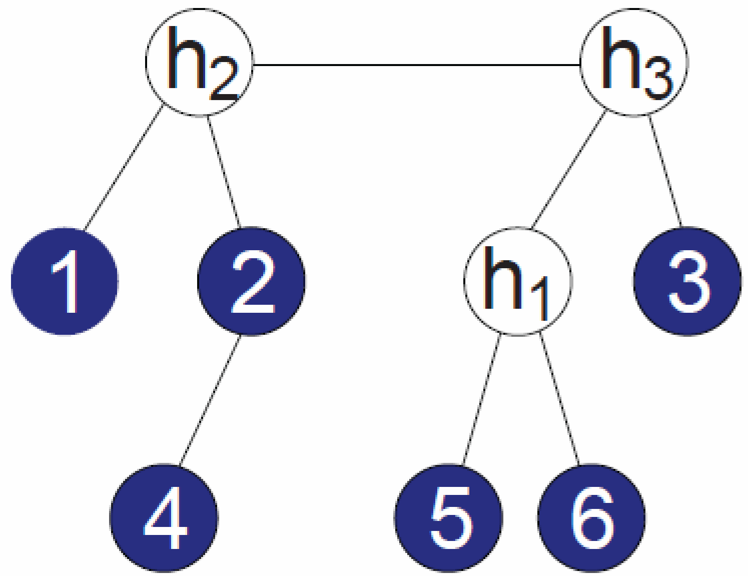

Node i and j – leaf nodes and share the same parent (sibling nodes)

$$\Longleftrightarrow d_{jk} - d_{ik} = d_{hj} - d_{hi}$$

for all k ≠ i, j .

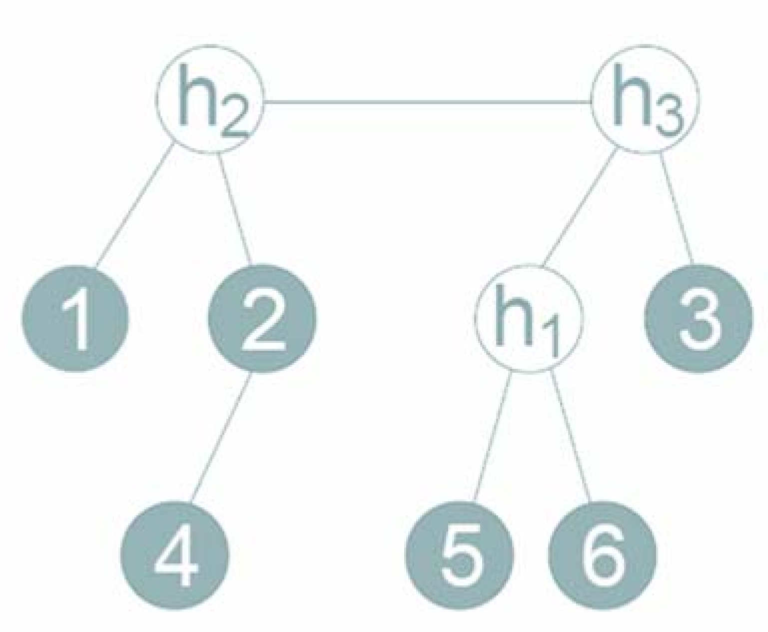
Can identify leaf-sibling pairs.

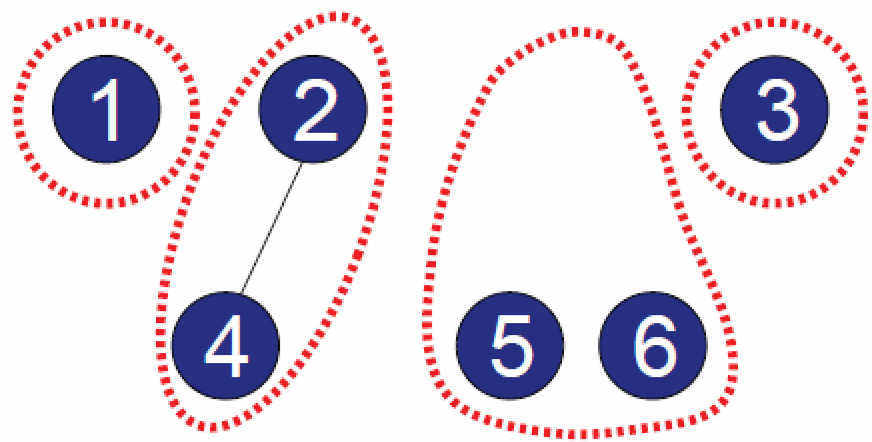
# Recursive Grouping




Step 1. Compute $d_{jk} - d_{ik}$ for all observed nodes (i, j, k).

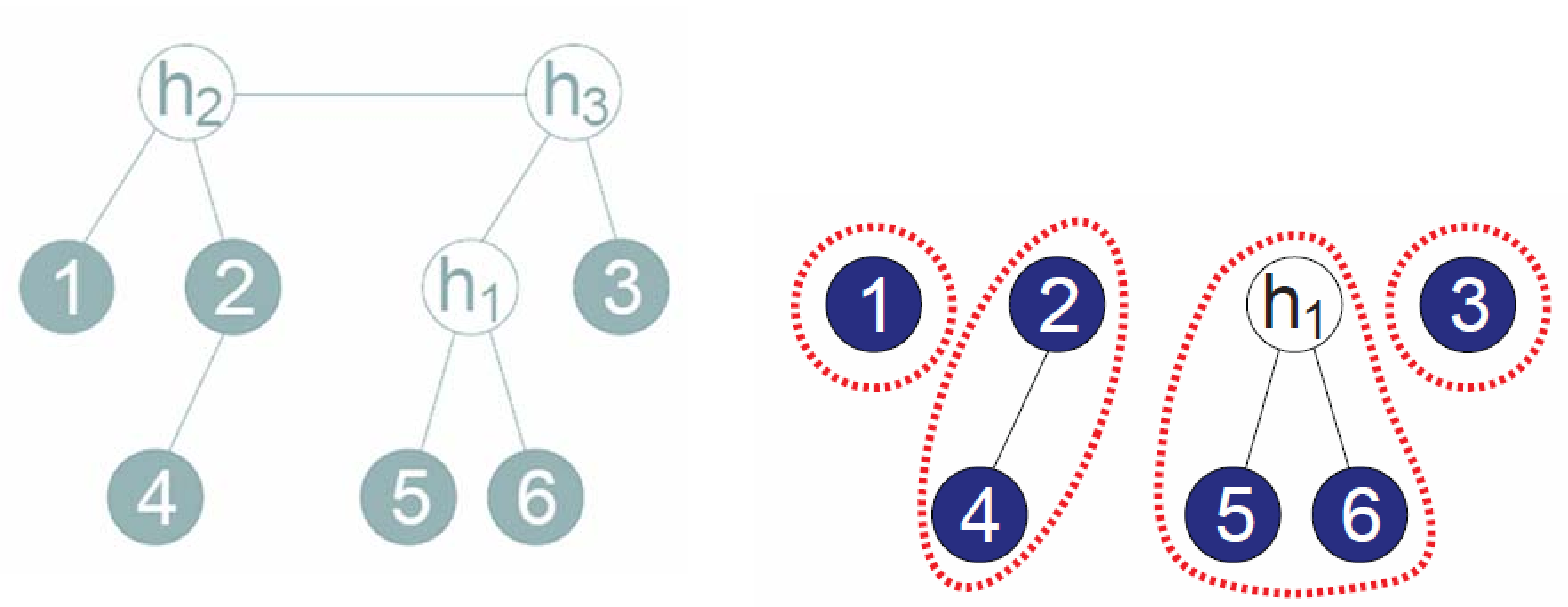
# Recursive Grouping



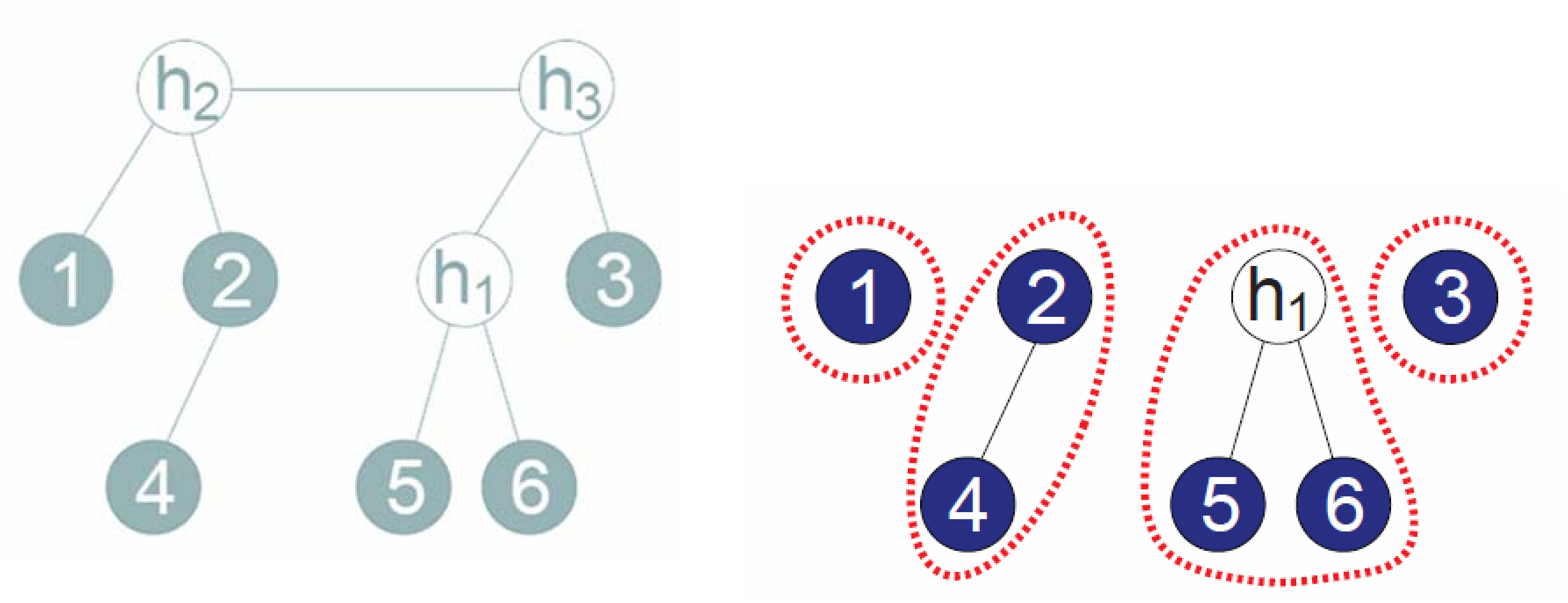

Step 2. Identify (parent, leaf child) or (leaf siblings) pairs.

# Recursive Grouping




Step 3. Introduce a hidden parent node for each sibling group without a parent.

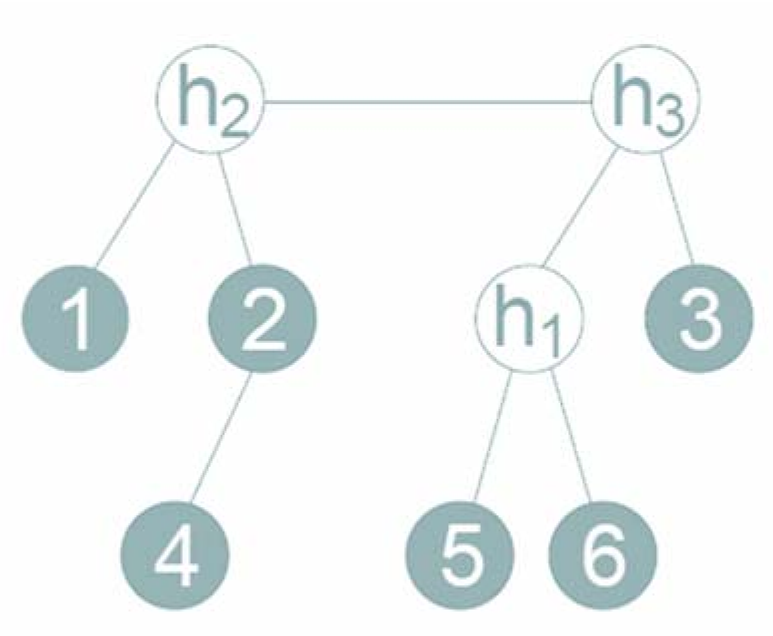
# Recursive Grouping

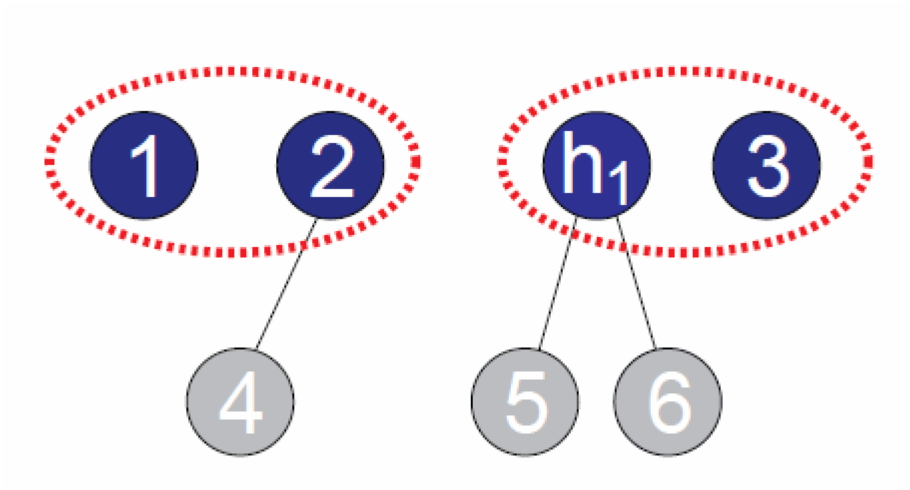



Step 4. Compute the information distance for new hidden nodes.

e.g.) $$d_{5h_1} = \frac{1}{2}(d_{56} + d_{53} - d_{63})$$

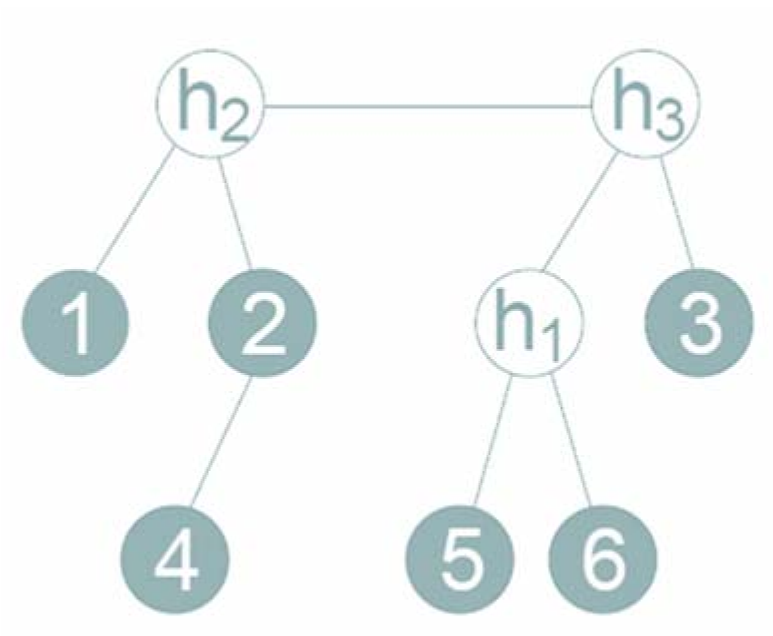
# Recursive Grouping

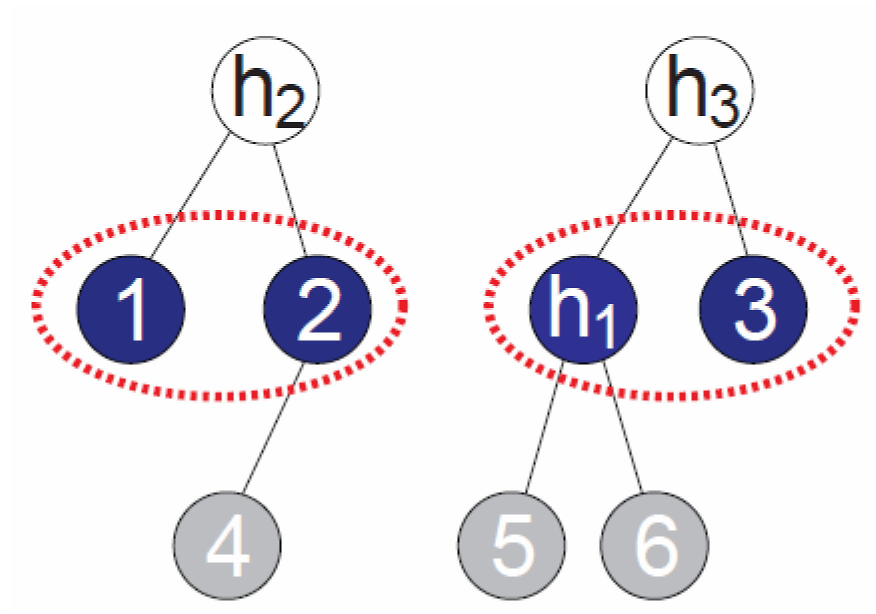



Step 5. Remove the identified child nodes and repeat Steps 2-4.

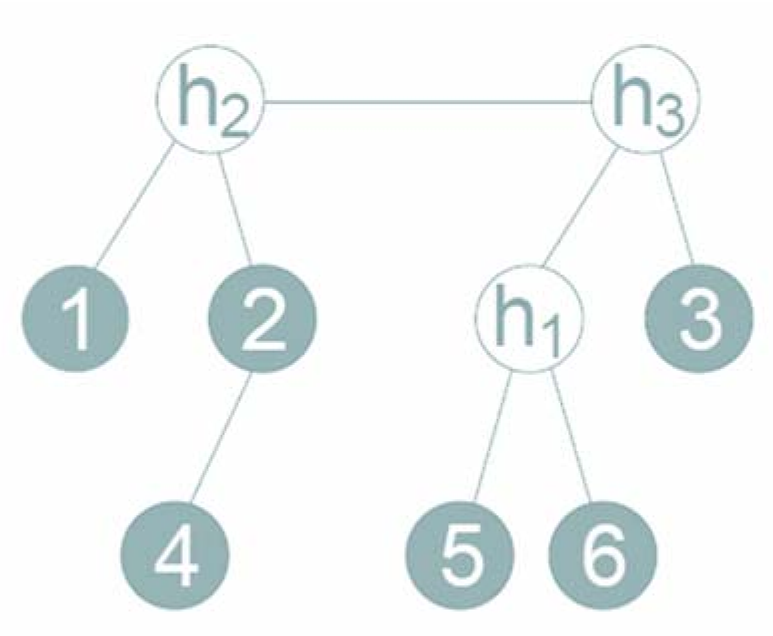
# Recursive Grouping

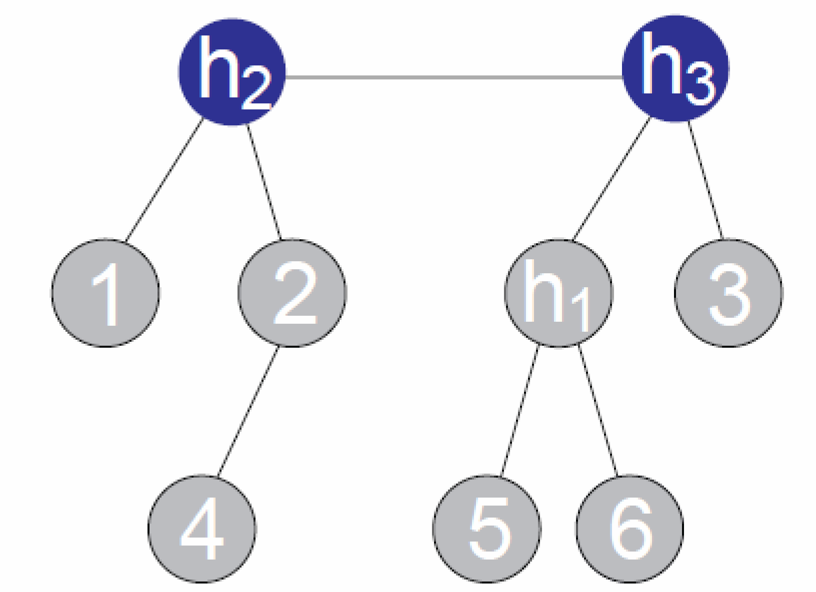



Step 5. Remove the identified child nodes and repeat Steps 2-4.

# Recursive Grouping



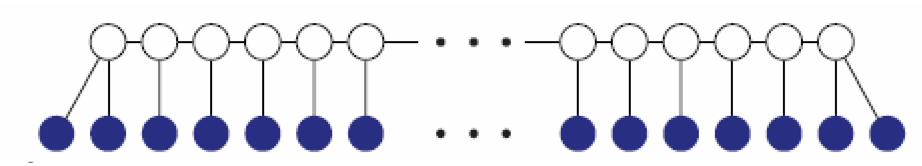




Step 5. Remove the identified child nodes and repeat Steps 2-4.

# Recursive Grouping

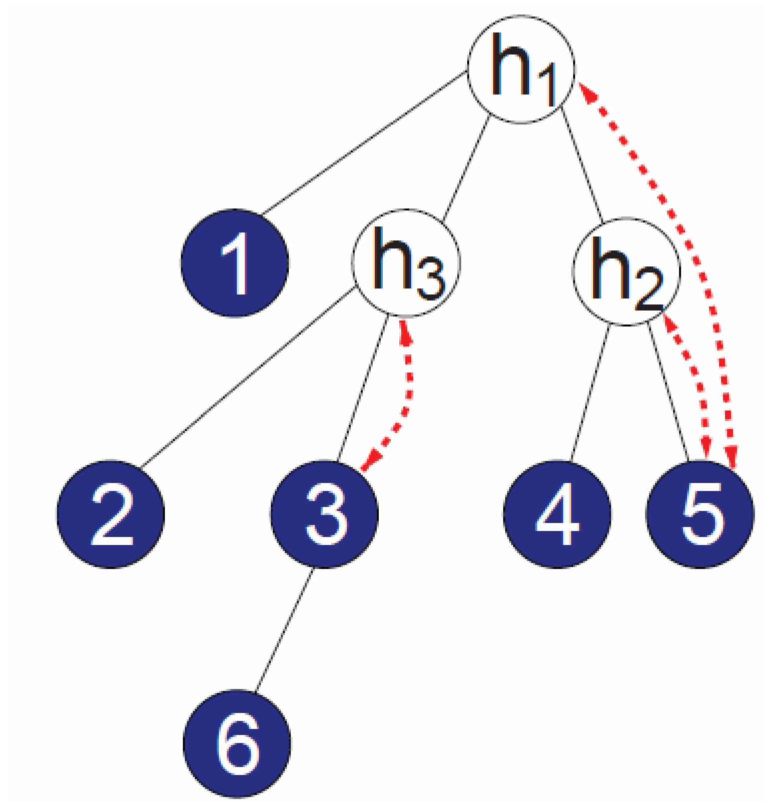
- Identifies a group of family nodes at each step.
- Introduces hidden nodes recursively.
- Correctly recovers all minimal latent trees.

- Computational complexity O(diam(T) $m^3$).



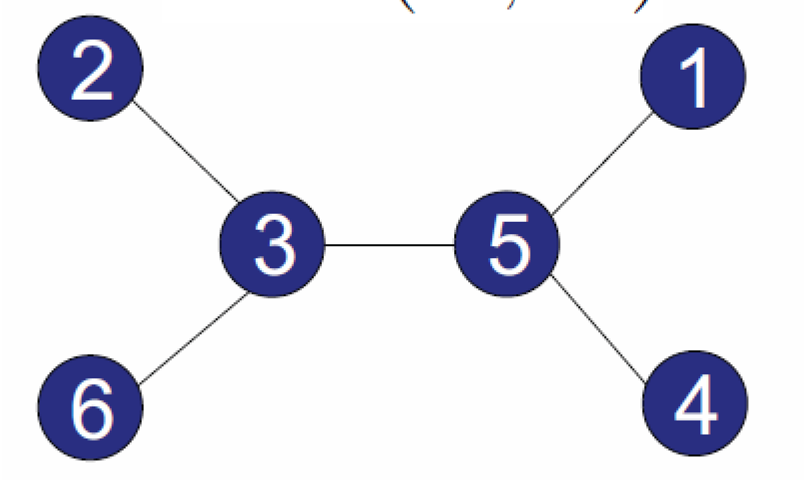
Worst case O($m^4$)

# CLGrouping Algorithm



# Chow-Liu Tree

$$\mathrm{MST}(V; \mathbf{D})$$




Minimum spanning tree of V

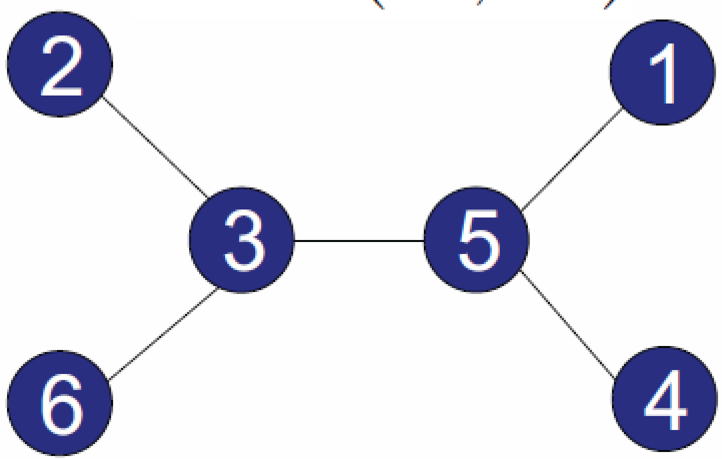
using D as edge weights

V = set of observed nodes

D = information distances

# Chow-Liu Tree

$\mathrm{MST}(V; \mathbf{D})$



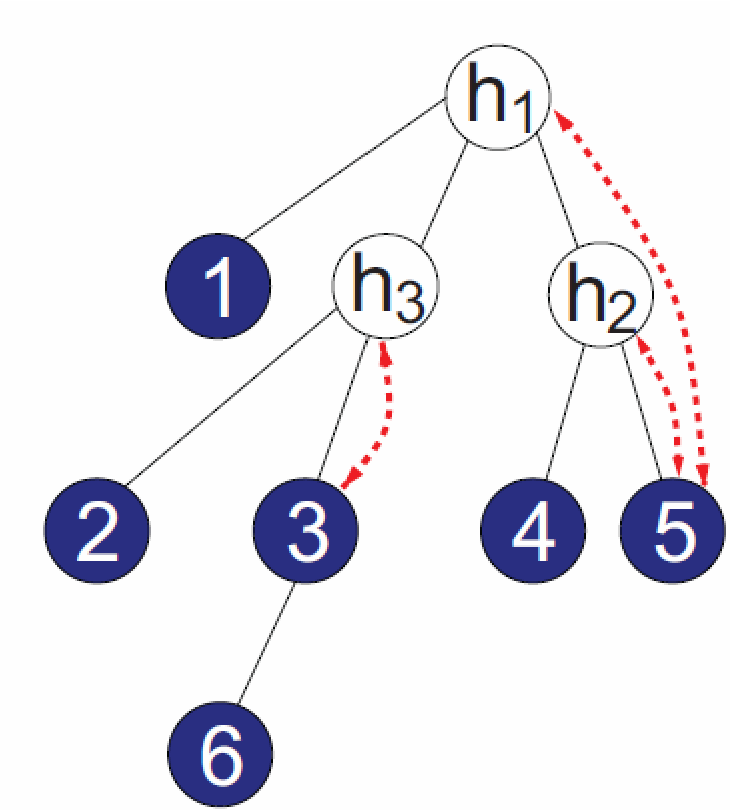

Minimum spanning tree of V using D as edge weights

V = set of observed nodes

D = information distances

- Computational complexity $O(m^2 \log m)$
- For Gaussian models, MST(V; D) = Chow-Liu tree (minimizes KL-divergence to the distribution given by D).

# Surrogate Node



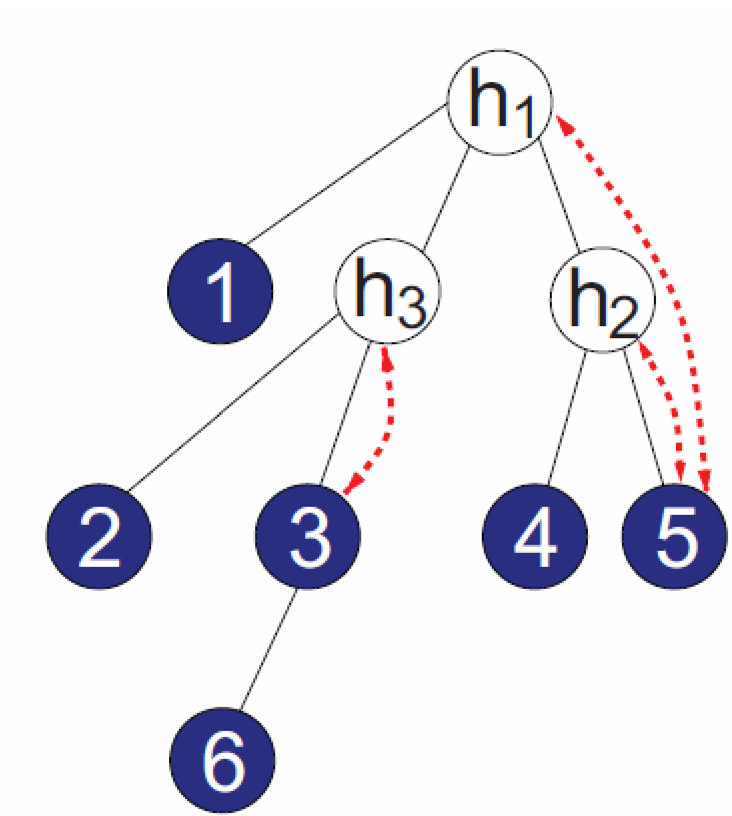

V = set of observed nodes

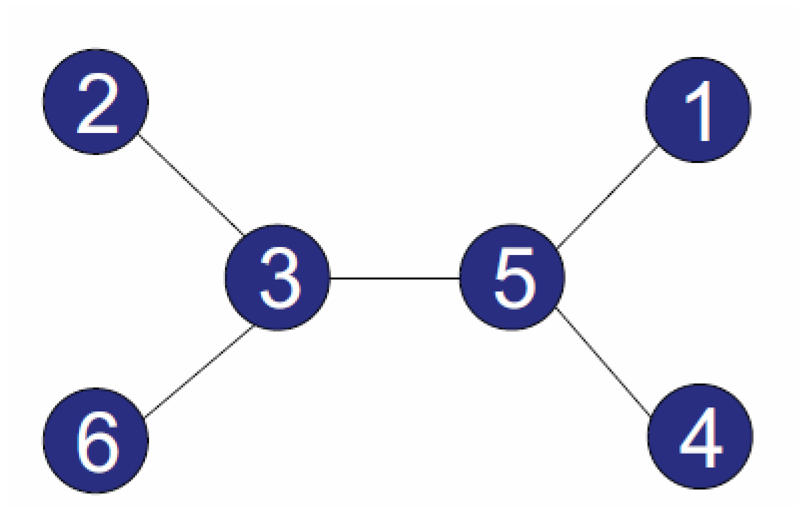
Surrogate node of i

$$\mathrm{Sg}(i) := \operatorname*{argmin}_{j \in V} d_{ij}$$

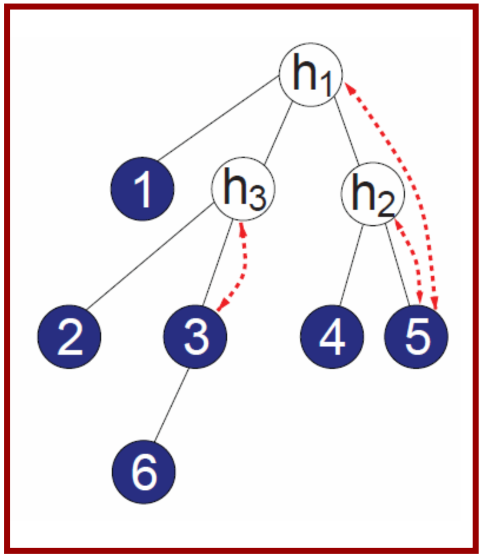
# Property of the Chow-Liu Tree

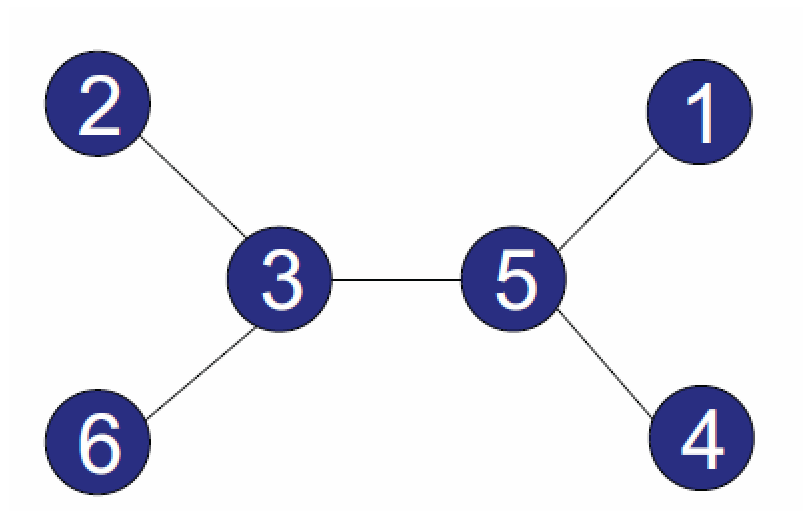
$$\boxed{(i,j) \in E_p \Rightarrow (\mathrm{Sg}(i), \mathrm{Sg}(j)) \in \mathrm{MST}(V; \mathbf{d})}$$



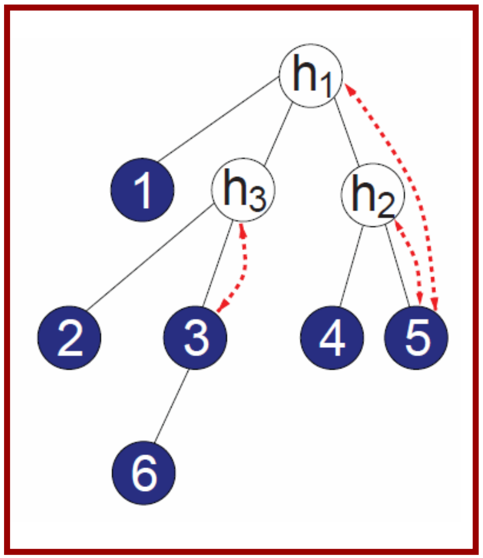
# CLGrouping Algorithm

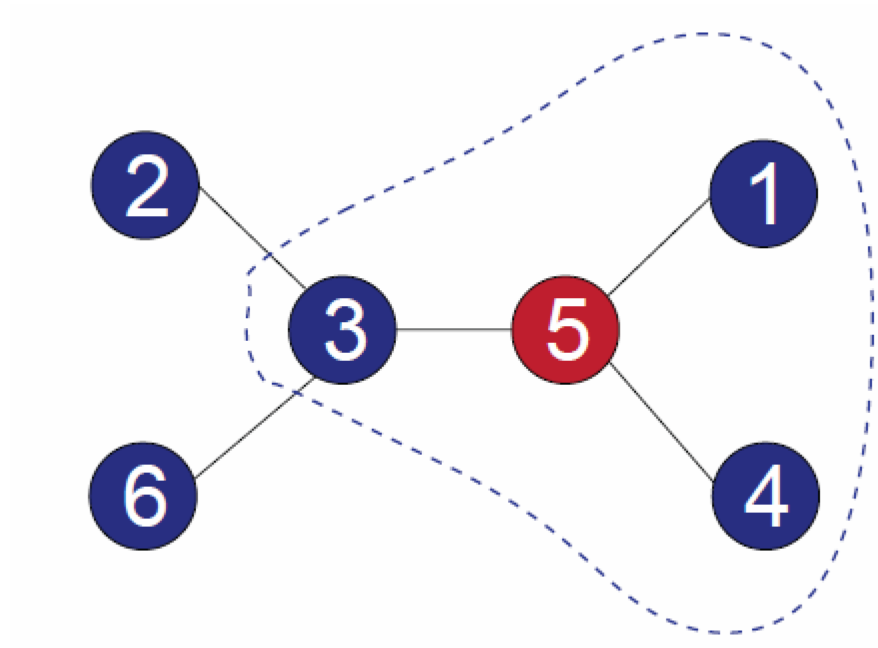



Step 1. Using information distances of observed nodes, construct the Chow-Liu tree, MST(V; D). Identify the set of internal nodes {3, 5}.

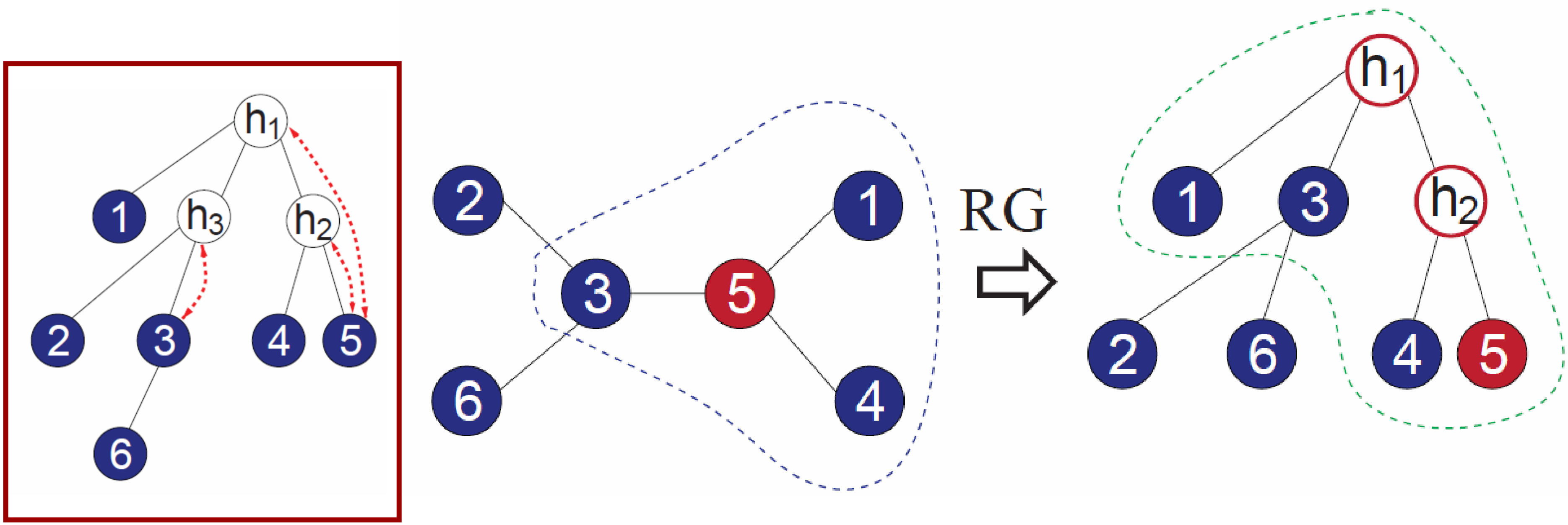
# CLGrouping Algorithm



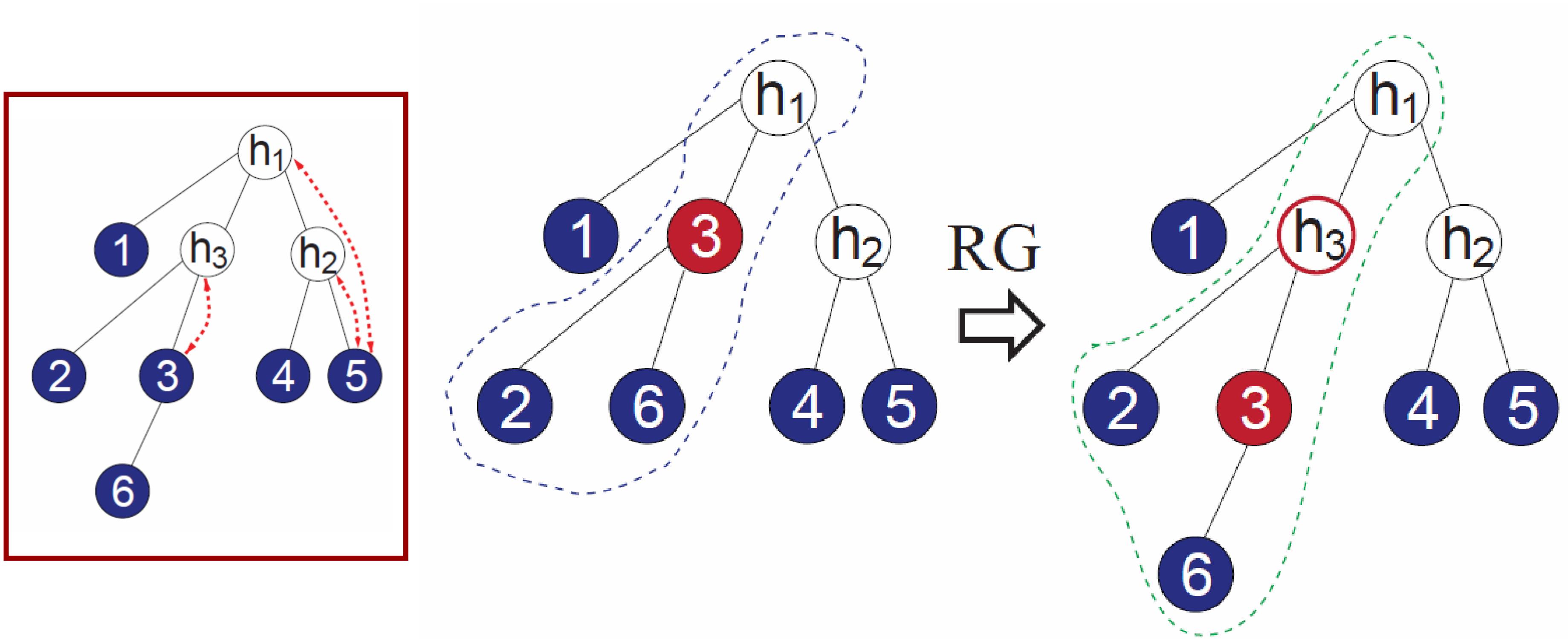

Step 2. Select an internal node and its neighbors, and apply the recursive-grouping (RG) algorithm.

# CLGrouping Algorithm




Step 3. Replace the output of RG with the sub-tree spanning the neighborhood.

# CLGrouping Algorithm

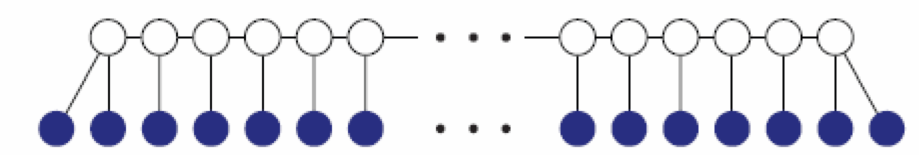



Repeat Steps 2-3 until all internal nodes are operated on.

# CLGrouping

- Step 1: Constructs the Chow-Liu tree, MST(V; D).
- Step 2: For each internal node and its neighbors, applies latent-tree-learning subroutines (RG or NJ).
- Correctly recovers all minimal latent trees.
- Computational complexity

  $O(m^2 \log m + (\#\text{internal nodes})\,(\text{maximum degree})^3)$.



$O(m^2 \log m)$

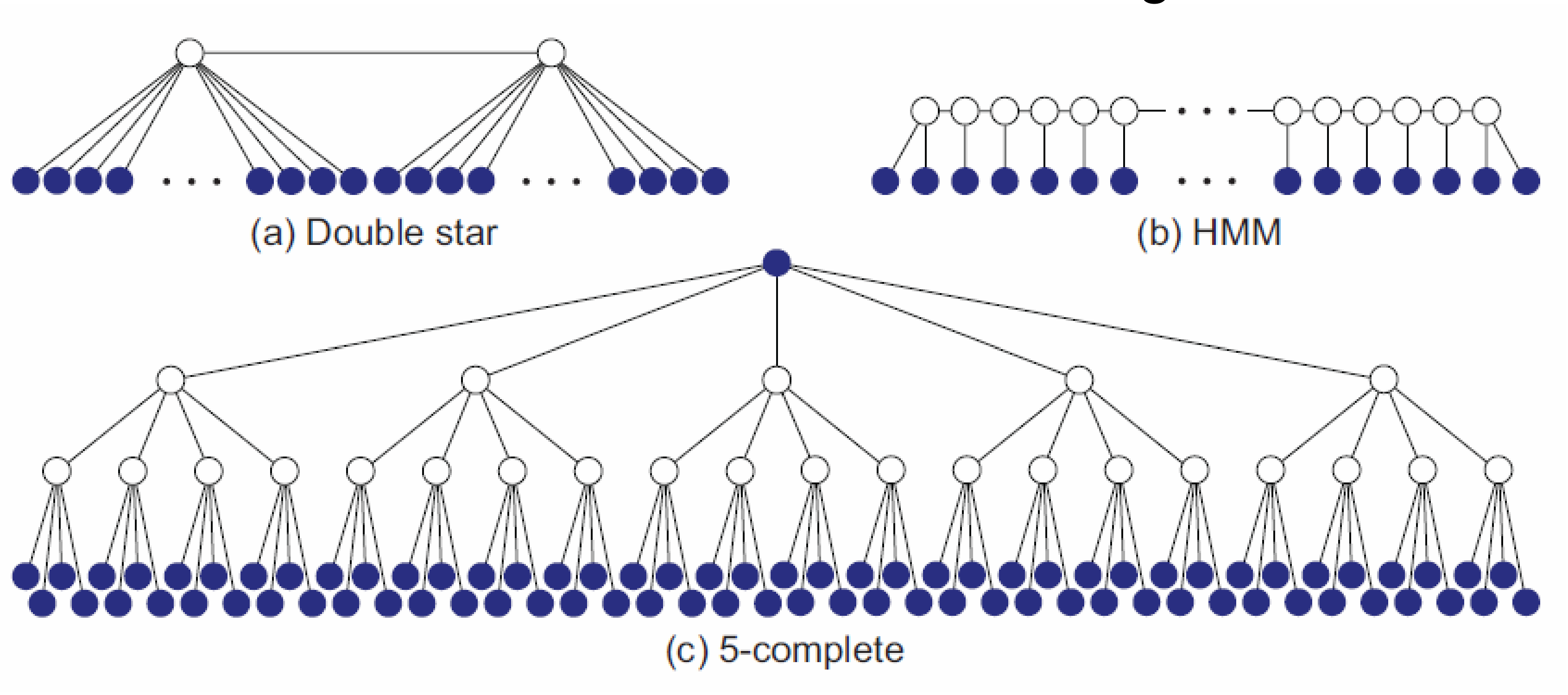
# Sample-based Algorithms

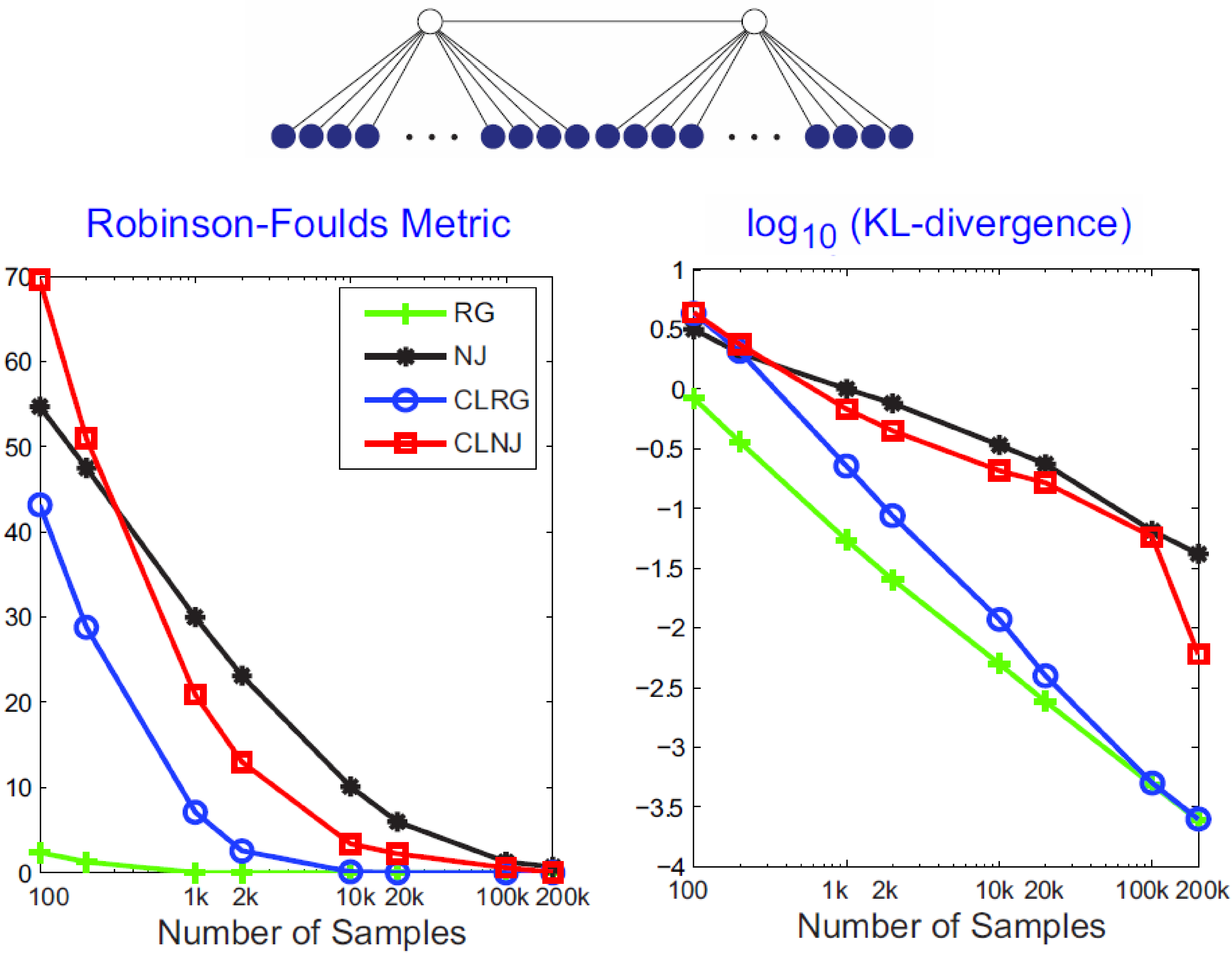
- Compute the ML estimates of information distances.
- Relaxed constraints for testing node relationships.

- Consistent.
- More details in the paper

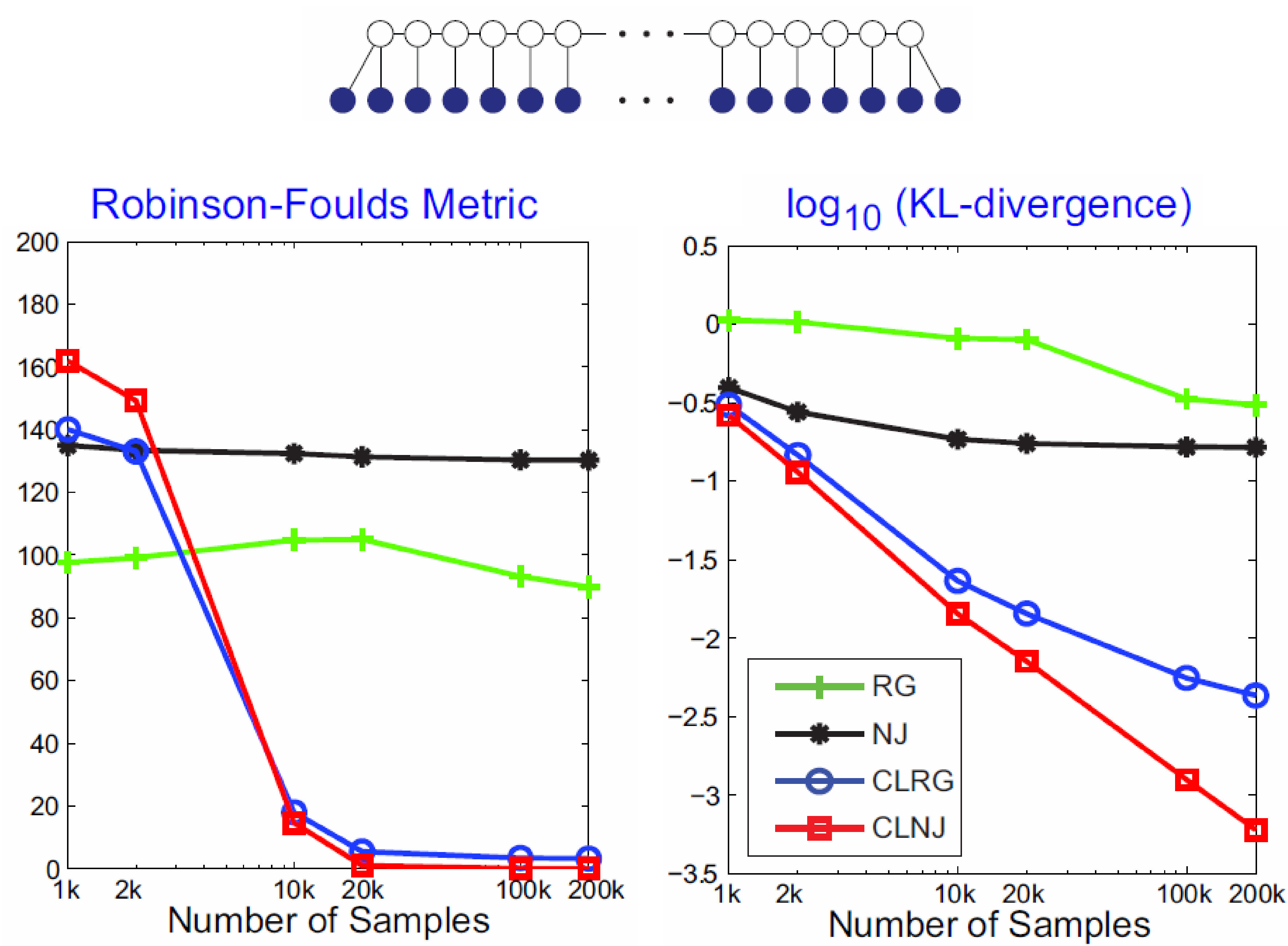
# Experimental Results

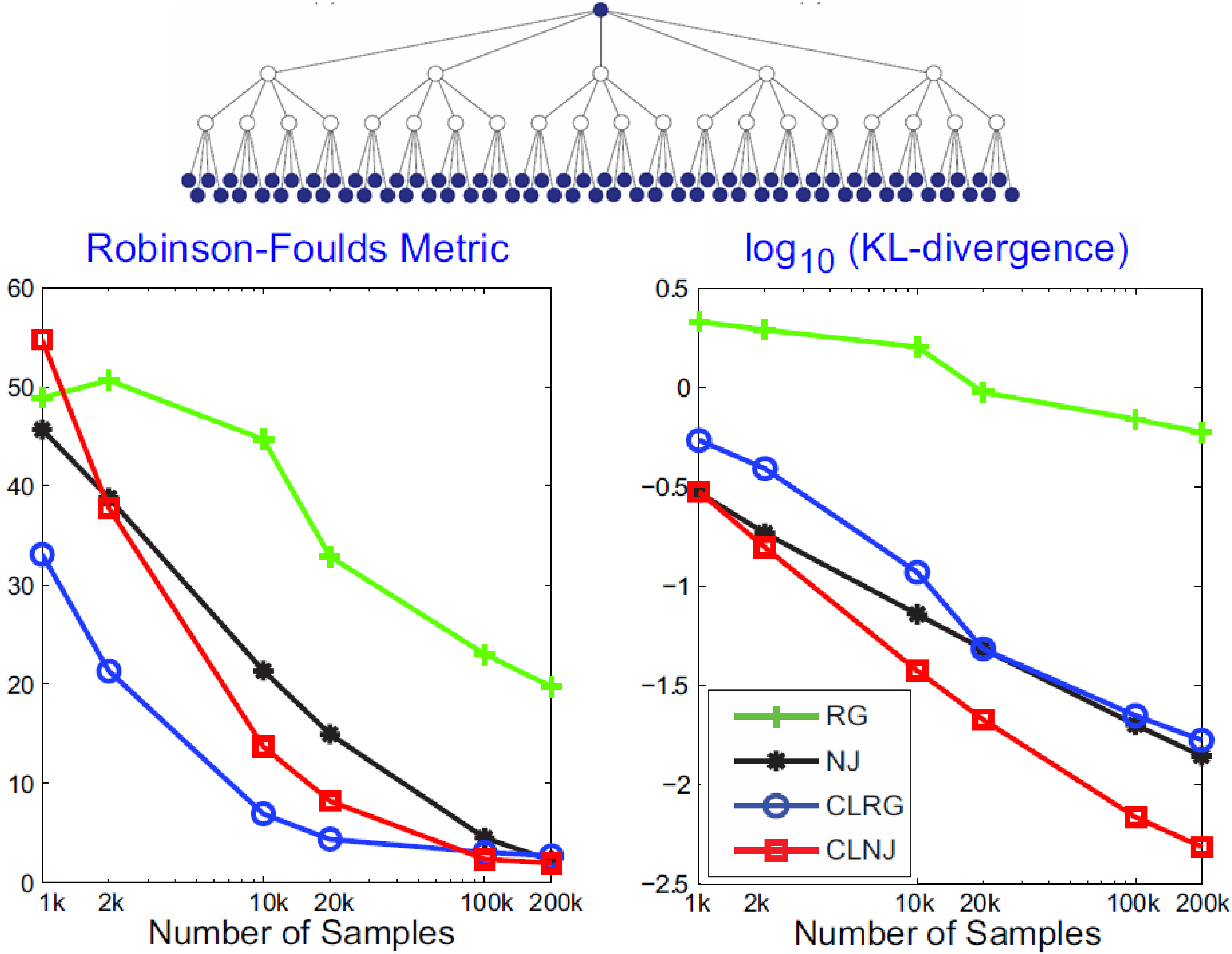
- Simulations using Synthetic Datasets
  - Compares RG, NJ, CLRG, and CLNJ.
  - Robinson-Foulds Metric and KL-divergence.





Robinson-Foulds Metric
log10 (KL-divergence)
RG
NJ
CLRG
CLNJ
70
60
50
40
30
20
10
0
1
0.5
0
−0.5
−1
−1.5
−2
−2.5
−3
−3.5
−4
100
1k
2k
10k
20k
100k
200k
Number of Samples
Number of Samples


Robinson-Foulds Metric
log10 (KL-divergence)
200
180
160
140
120
100
80
60
40
20
0
0.5
0
−0.5
−1
−1.5
−2
−2.5
−3
−3.5
1k
2k
10k
20k
100k
200k
Number of Samples
RG
NJ
CLRG
CLNJ


Robinson-Foulds Metric
log10 (KL-divergence)
60
50
40
30
20
10
0
0.5
0
−0.5
−1
−1.5
−2
−2.5
1k
2k
10k
20k
100k
200k
Number of Samples
RG
NJ
CLRG
CLNJ

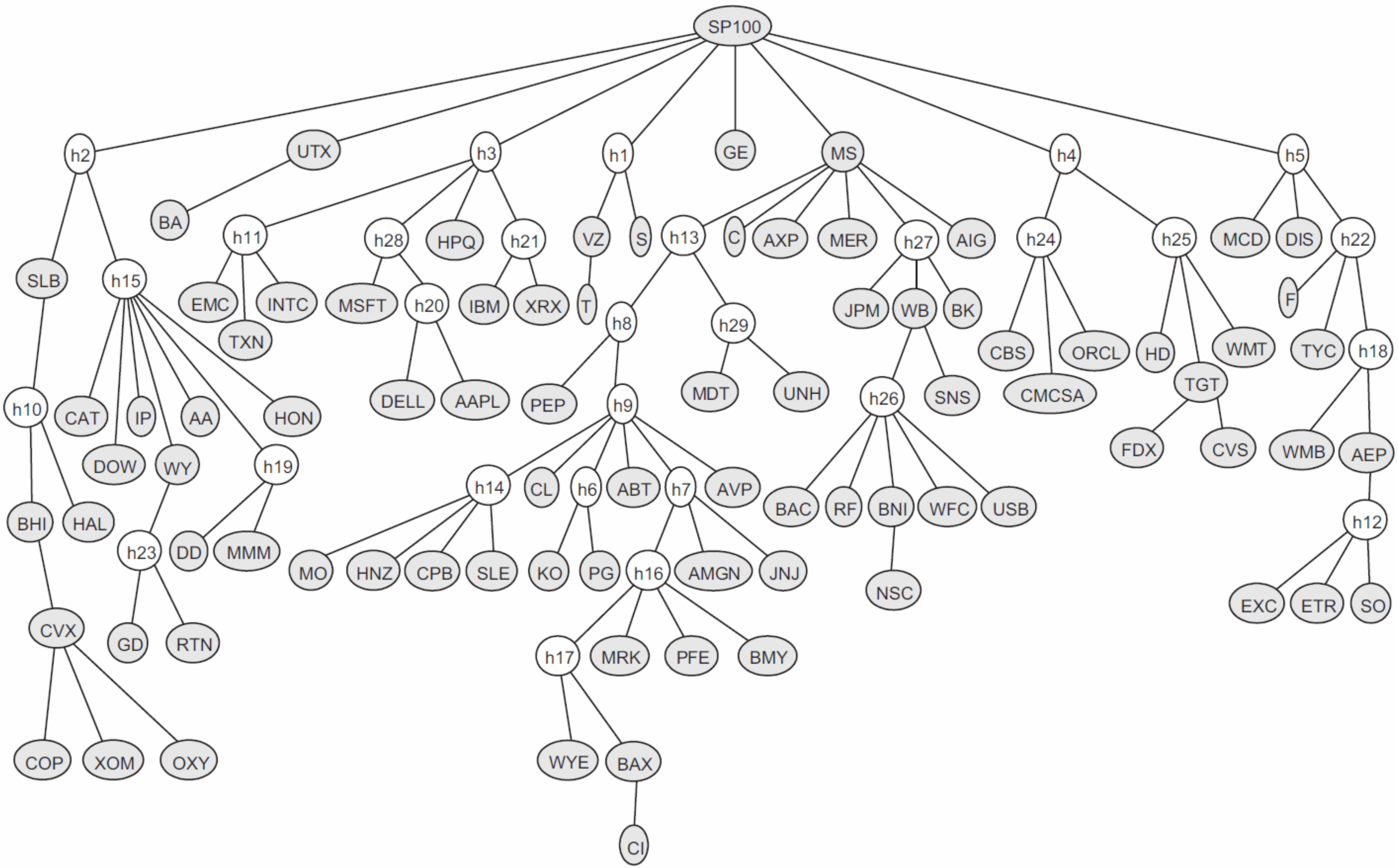
# Performance Comparisons

- For a double star, RG is clearly the best.
- NJ is poor in recovering HMM.
- CLGrouping performs well in all three structures.

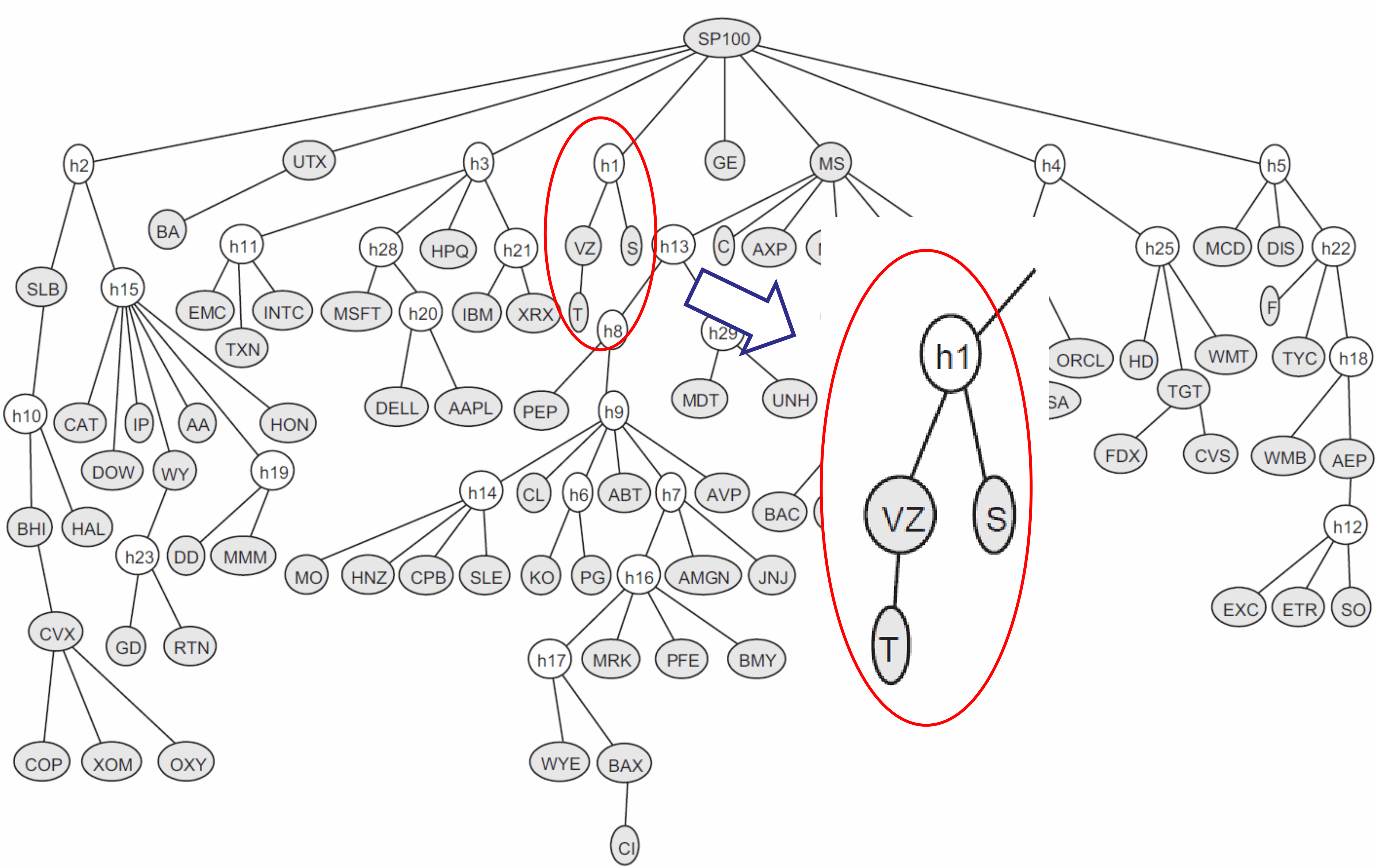
- Average running time for CLGrouping < 1 second.

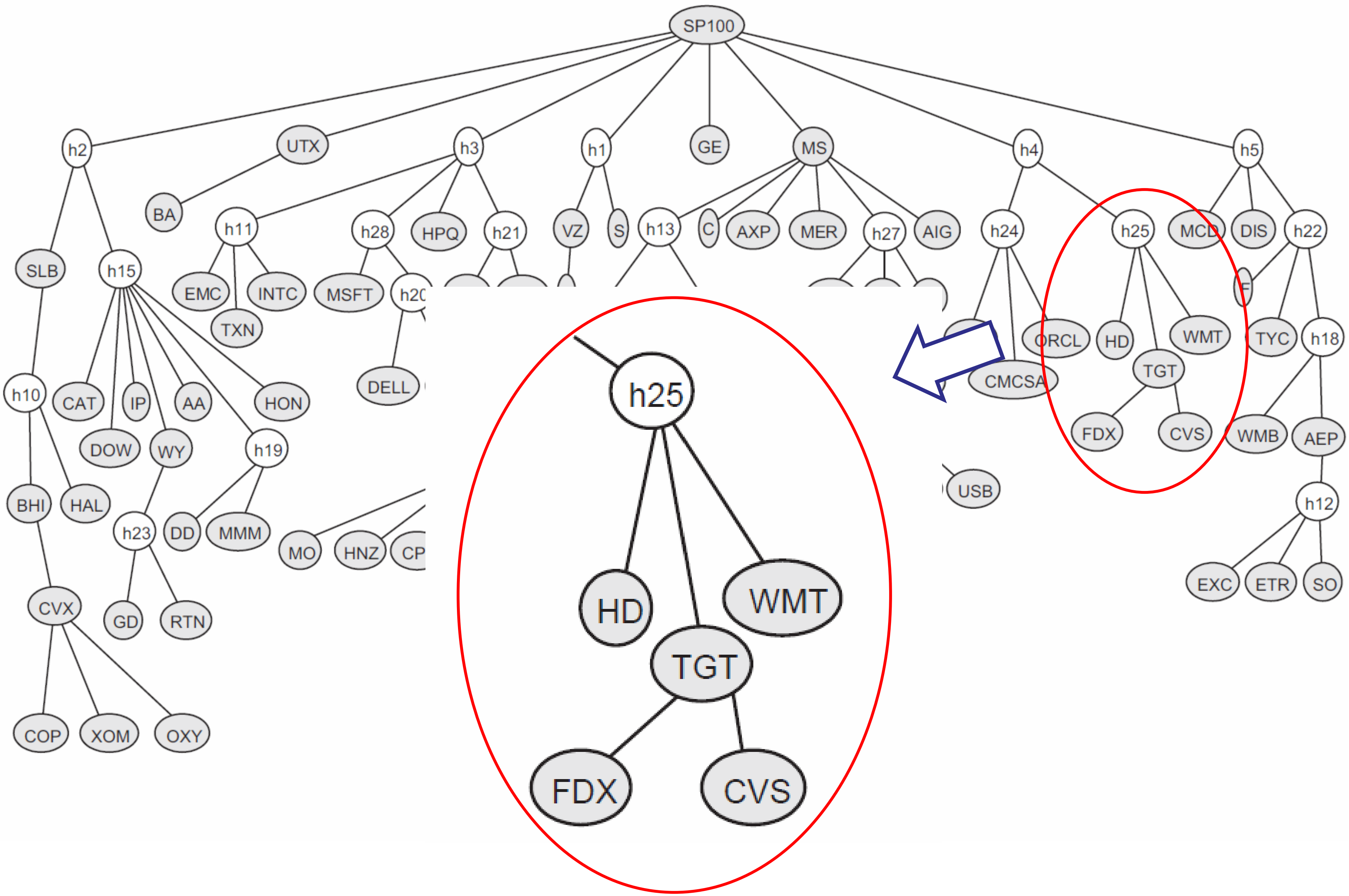
# Monthly Stock Returns

- Monthly returns of 84 companies in S&P 100.
- Samples from 1990 to 2007.
- Latent tree learned using CLNJ.

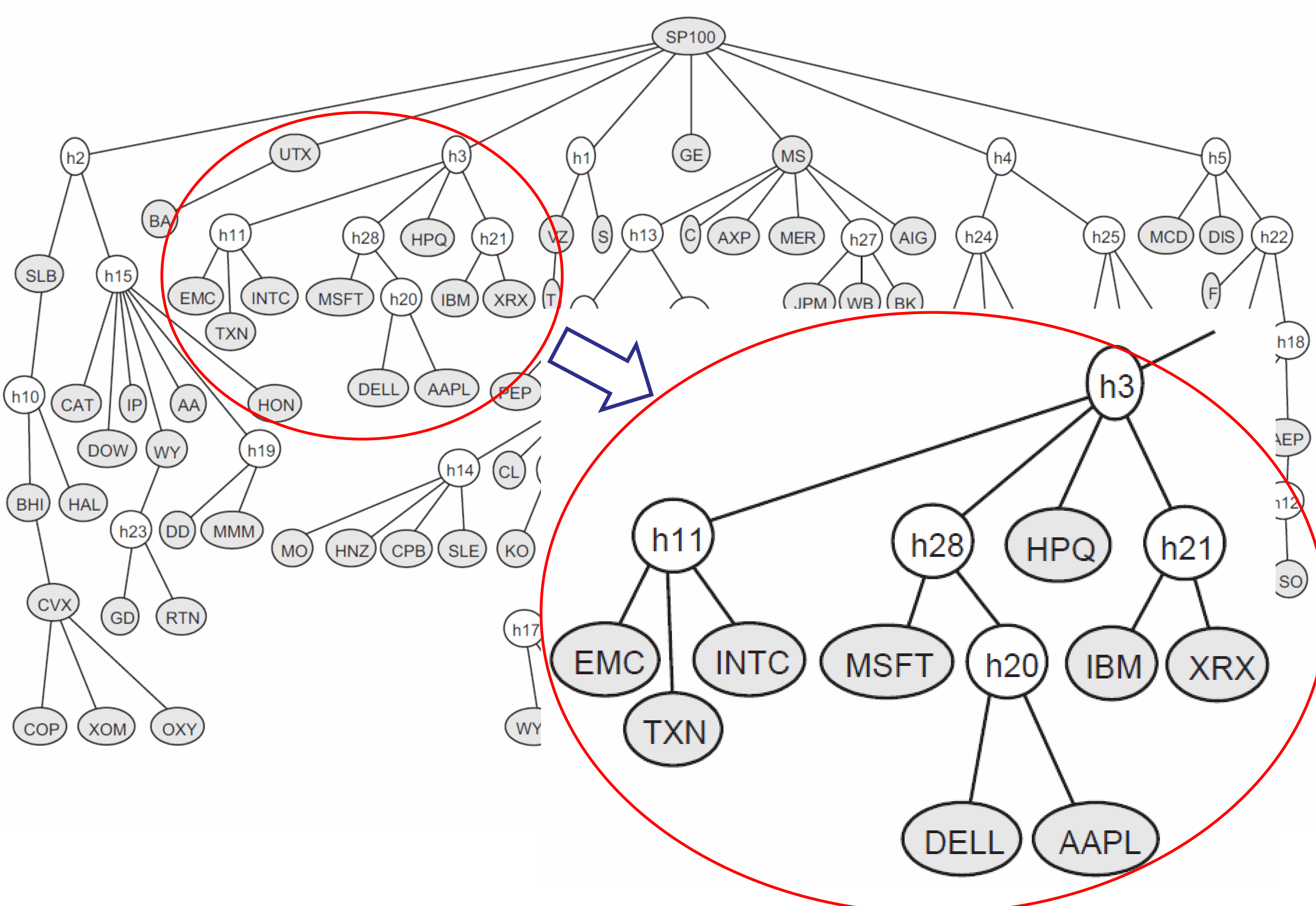


SP100
h2
UTX
h3
h1
GE
MS
h4
h5
BA
h11
h28
HPQ
h21
VZ
S
h13
C
AXP
MER
h27
AIG
h24
h25
MCD
DIS
h22
SLB
h15
EMC
INTC
MSFT
h20
IBM
XRX
T
h8
h29
JPM
WB
BK
F
TXN
CBS
ORCL
HD
WMT
TYC
h18
CMCSA
TGT
h10
CAT
IP
AA
HON
DELL
AAPL
PEP
h9
MDT
UNH
h26
SNS
DOW
WY
h19
FDX
CVS
WMB
AEP
h14
CL
h6
ABT
h7
AVP
BAC
RF
BNI
WFC
USB
BHI
HAL
h23
DD
MMM
h12
MO
HNZ
CPB
SLE
KO
PG
h16
AMGN
JNJ
NSC
EXC
ETR
SO
CVX
GD
RTN
h17
MRK
PFE
BMY
COP
XOM
OXY
WYE
BAX
CI


SP100
h2
UTX
h3
h1
GE
MS
h4
h5
BA
h11
h28
HPQ
h21
VZ
S
h13
C
AXP
h25
MCD
DIS
h22
SLB
h15
EMC
INTC
MSFT
h20
IBM
XRX
T
h8
h29
F
TXN
ORCL
HD
WMT
TYC
h18
h10
CAT
IP
AA
HON
DELL
AAPL
PEP
h9
MDT
UNH
TGT
DOW
WY
h19
FDX
CVS
WMB
AEP
h14
CL
h6
ABT
h7
AVP
BAC
BHI
HAL
h23
DD
MMM
MO
HNZ
CPB
SLE
KO
PG
h16
AMGN
JNJ
h12
EXC
ETR
SO
CVX
GD
RTN
h17
MRK
PFE
BMY
COP
XOM
OXY
WYE
BAX
CI

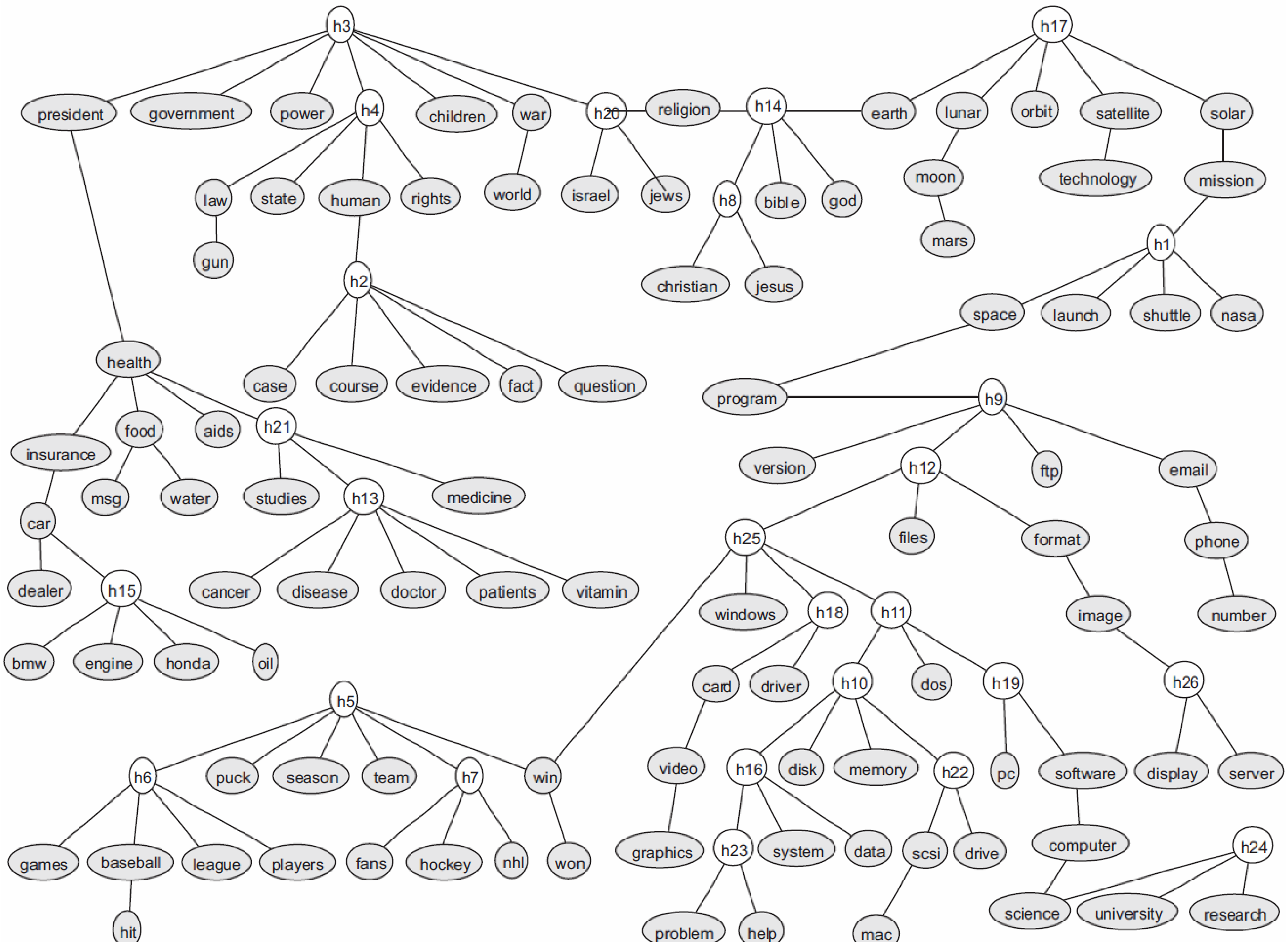


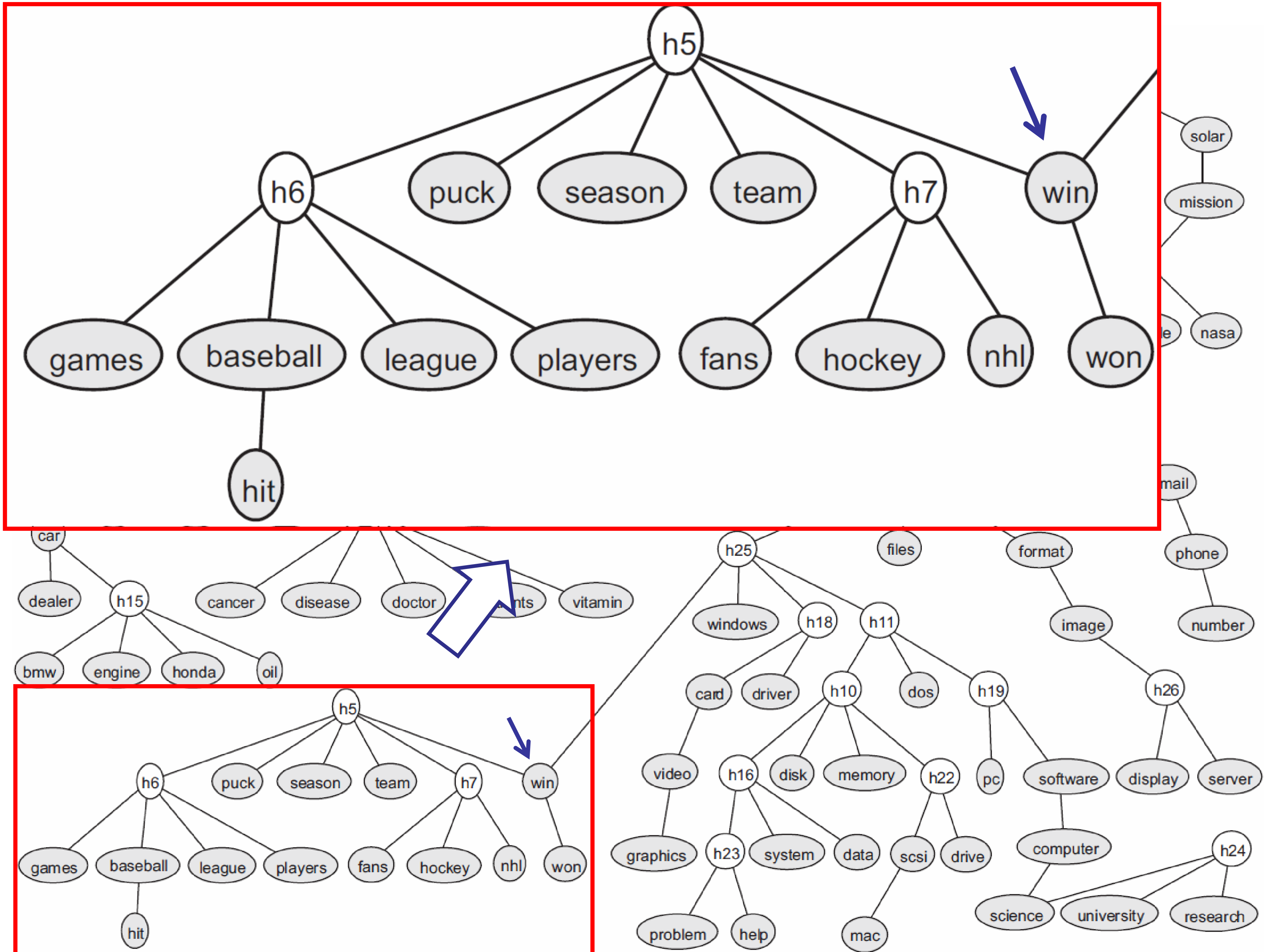
SP100
h2
UTX
h3
h1
GE
MS
h4
h5
BA
h11
h28
HPQ
h21
VZ
S
h13
C
AXP
MER
h27
AIG
h24
h25
MCD
DIS
h22
SLB
h15
EMC
INTC
TXN
MSFT
DELL
ORCL
HD
WMT
TGT
FDX
CVS
CMCSA
TYC
h18
h10
CAT
IP
AA
HON
DOW
WY
h19
WMB
AEP
BHI
HAL
h23
DD
MMM
MO
HNZ
USB
h12
GD
RTN
CVX
EXC
ETR
SO
COP
XOM
OXY

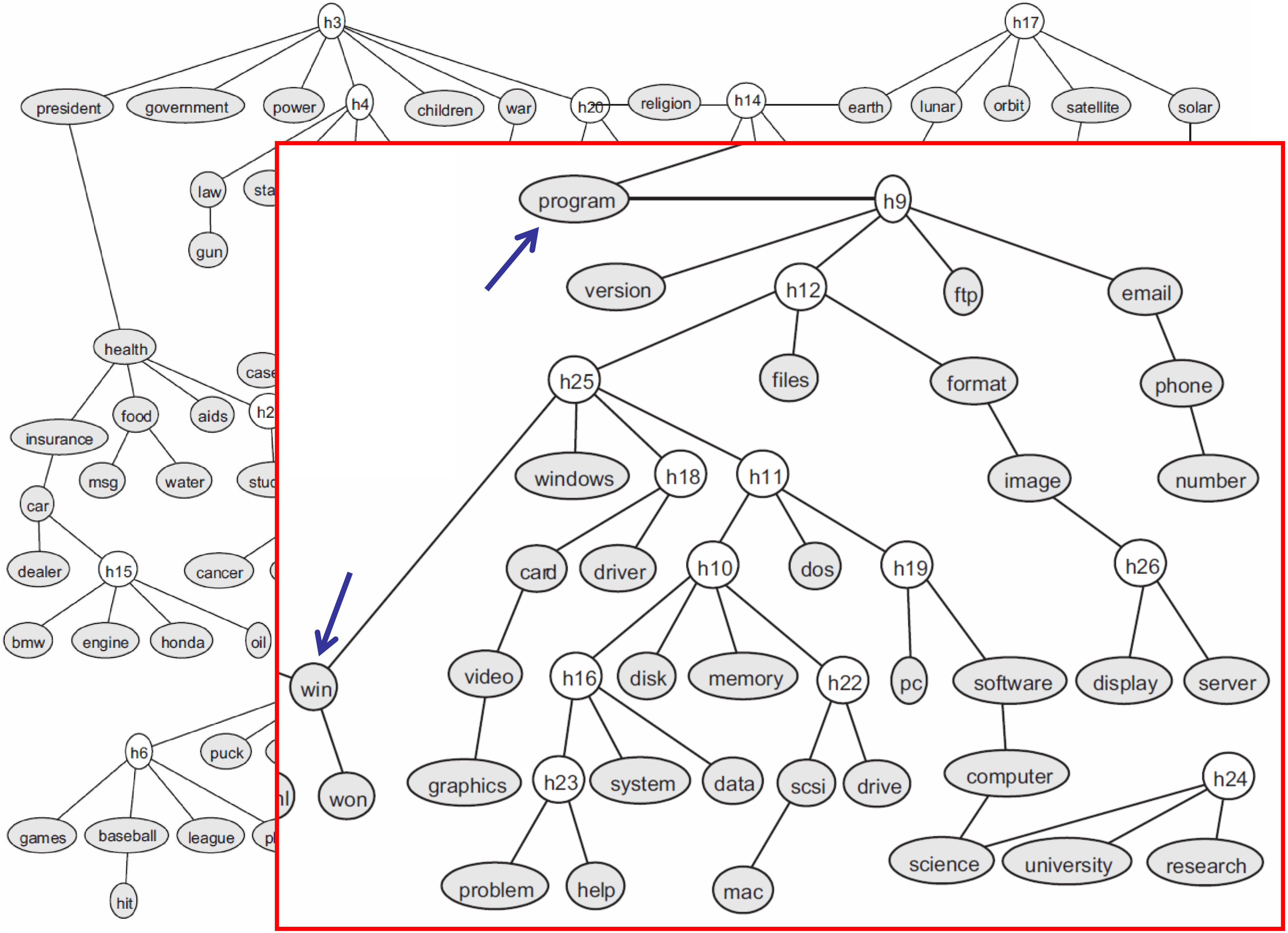

SP100
h2
UTX
h3
h1
GE
MS
h4
h5
BA
h11
h28
HPQ
h21
VZ
S
h13
C
AXP
MER
h27
AIG
h24
h25
MCD
DIS
h22
SLB
h15
EMC
INTC
MSFT
h20
IBM
XRX
T
JPM
WB
BK
F
TXN
h18
h10
CAT
IP
AA
HON
DELL
AAPL
PEP
AEP
DOW
WY
h19
h14
CL
BHI
HAL
h23
DD
MMM
MO
HNZ
CPB
SLE
KO
SO
CVX
GD
RTN
h17
COP
XOM
OXY
WY
h3
h11
h28
HPQ
h21
EMC
INTC
MSFT
h20
IBM
XRX
TXN
DELL
AAPL

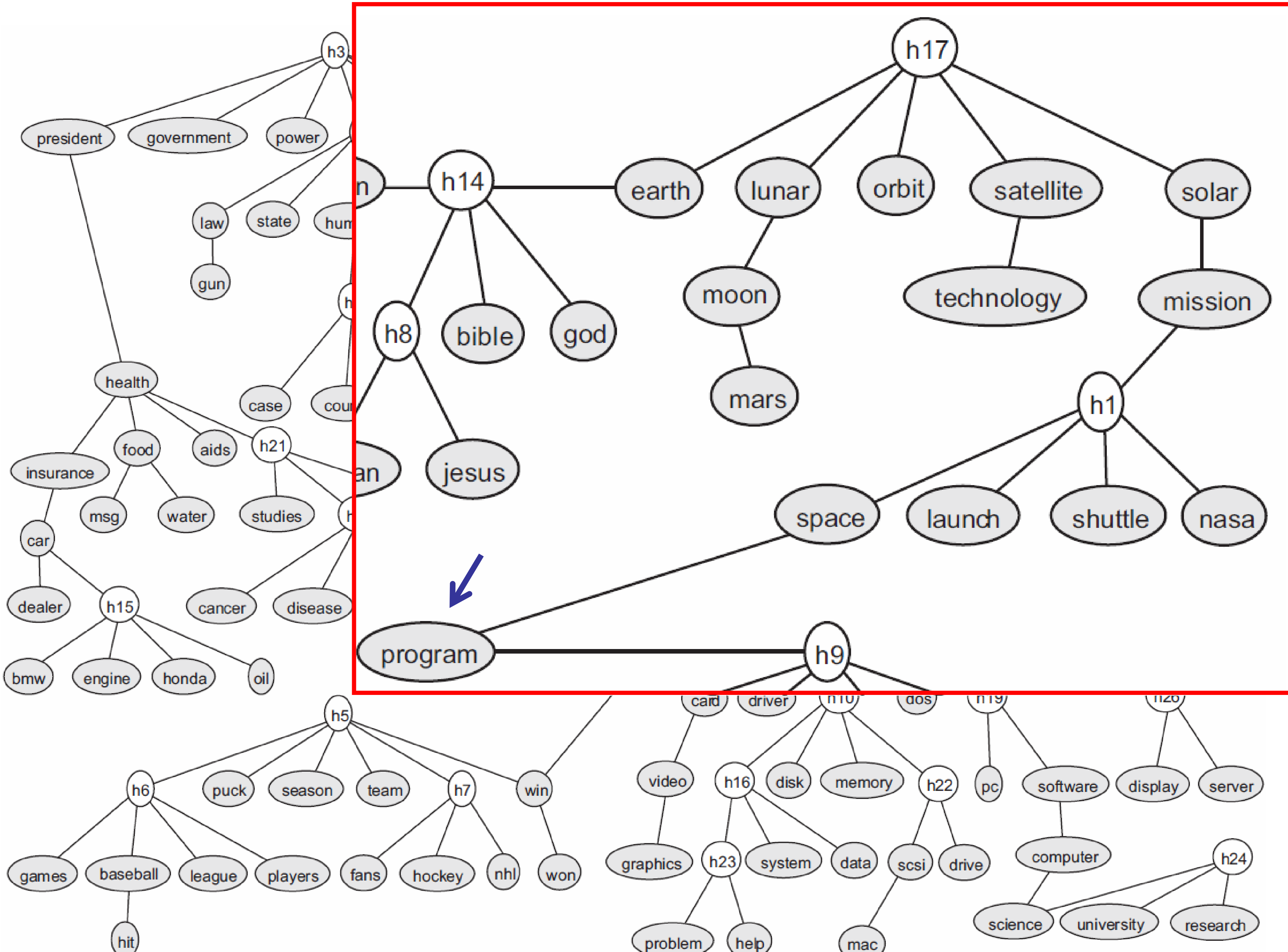
# 20 Newsgroups with 100 Words

- 16,242 binary samples of 100 words
- Latent tree learned using regCLRG.


h3
h17
president
government
power
h4
children
war
h20
religion
h14
earth
lunar
orbit
satellite
solar
law
state
human
rights
world
israel
jews
h8
bible
god
moon
technology
mission
gun
mars
h1
h2
christian
jesus
space
launch
shuttle
nasa
health
case
course
evidence
fact
question
program
h9
food
aids
h21
insurance
version
h12
ftp
email
msg
water
studies
h13
medicine
car
h25
files
format
phone
dealer
h15
cancer
disease
doctor
patients
vitamin
windows
h18
h11
image
number
bmw
engine
honda
oil
h5
card
driver
h10
dos
h19
h26
h6
puck
season
team
h7
win
video
h16
disk
memory
h22
pc
software
display
server
games
baseball
league
players
fans
hockey
nhl
won
graphics
h23
system
data
scsi
drive
computer
h24
hit
problem
help
mac
science
university
research


h5
h6
puck
season
team
h7
win
games
baseball
league
players
fans
hockey
nhl
won
hit
solar
mission
nasa
mail
phone
number
car
dealer
h15
bmw
engine
honda
oil
cancer
disease
doctor
vitamin
h25
files
format
windows
h18
h11
image
card
driver
h10
dos
h19
h26
video
h16
disk
memory
h22
pc
software
display
server
graphics
h23
system
data
scsi
drive
computer
h24
problem
help
mac
science
university
research


h3
h17
president
government
power
h4
children
war
h20
religion
h14
earth
lunar
orbit
satellite
solar
law
gun
program
h9
version
h12
ftp
email
health
h25
files
format
phone
food
aids
insurance
windows
h18
h11
image
number
msg
water
car
card
driver
h10
dos
h19
h26
dealer
h15
cancer
bmw
engine
honda
oil
win
video
h16
disk
memory
h22
pc
software
display
server
h6
puck
won
graphics
h23
system
data
scsi
drive
computer
h24
games
baseball
league
problem
help
mac
science
university
research
hit


h3
president
government
power
law
state
gun
health
case
insurance
food
aids
h21
msg
water
studies
car
dealer
h15
cancer
disease
bmw
engine
honda
oil
h17
h14
earth
lunar
orbit
satellite
solar
h8
bible
god
moon
technology
mission
mars
h1
jesus
space
launch
shuttle
nasa
program
h9
card
driver
dos
h5
h6
puck
season
team
h7
win
games
baseball
league
players
fans
hockey
nhl
won
hit
video
h16
disk
memory
h22
pc
software
display
server
graphics
h23
system
data
scsi
drive
computer
h24
problem
help
mac
science
university
research

# Contributions

- Recursive-grouping
  - Identifies families and introduces hidden nodes recursively.
- CLGrouping
  - First learns the Chow-Liu tree
  - Then applies latent-tree-learning subroutines locally.

# Contributions

- Recursive-grouping
- CLGrouping

- Consistent.
- CLGrouping - superior experimental results in both accuracy and computational efficiency.

- Longer version of the paper and MATLAB implementation available at the project webpage.

  http://people.csail.mit.edu/myungjin/latentTree.html