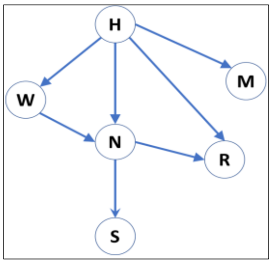
Table of Contents

Q1) Weather condition data:



1.1 Joint distribution of P(H,W,N,M,R,S):

P(H,W,N,M,R,S) = P(H) P(W|H) P(N|H,W) P(S|N) P(M|H) P(R|H,N)

1.2 Minimum number of parameters required by each node is:

Node	**Parameters**	**Comments**
H	3-1 = 2	P(H) has 3 factor levels
W	(3-1)*3 = 6	P(W|H): W has 3 factor levels, also one parent H which has 3 factor levels
N	(4-1)*3*3 = 27	P(N|H,W): N has 4 factor levels, also two parents, H (3 factor levels) and W (3 factor levels)
S	(3-1)*4 = 8	P(S|N): S has 3 factor levels with N as a parent having 4 factor levels
M	(4-1)*3 = 9	P(M|H): M has 4 factor levels with H as a parent having 3 factor levels.
R	(2-1)*3*4 = 12	P(R|H,N): R has 2 factor levels with H (3 factor levels) and N (4 factor levels) as parents
Total	**64**	

Total number of parameters required is 64.

1.3 Answer a,b,c:

a. Joint probability density function if there are no independence among the variable is assumed.
P(H,W,N,M,R,S) = P(H) P(W|H) P(N|H,W) P(M|H,W,N) P(R|H,W,N,M) P(S|H,W,N,M,R)

b. Minimum number of parameters required by each node when no independence is assumed:

Node	**Parameters**	**Comments**
H	3-1 = 2	P(H) has 3 factor levels
W	(3-1)*3 = 6	P(W|H): W has 3 factor levels, also one parent H which has 3 factor levels
N	(4-1)*3*3 = 27	P(N|H,W): N has 4 factor levels and H and W as parents
M	(4-1)*3*3*4 = 108	P(M|H,W,N): M has 4 factor levels and H,W,N as parents
R	(2-1)*3*3*4*4 = 144	P(R|H,W,N,M): R has 2 factor levels and H,W,N, and M as parents
S	(3-1)*3*3*4*4*2 = 576	P(S|H,W,N,M,R): S has 3 factor levels and H,W,N,M, and R as parents
Total	**863**	

Total 863 parameters are required if no independence is assumed.

c. When we consider dependency, then we need 64 parameters and when we consider independence, then we need 863 parameters i.e. 13.48 times more. Thus, to identify dependencies Bayesian network is used.

1.4 Answer a,b,c:

a) M⊥ S | ∅ (M is marginally independent of S)

Paths	Status	Comment
M-H-N-S	Unblocking	H is tail-to-tail and unobserved; unblocking N is head-to-tail and unobserved; unblocking
M-H-R-N-S	Blocking	H is tail-to-tail and unobserved; unblocking R is head-to-head and unobserved; blocking
M-H-W-N-S	Unblocking	H is tail-to-tail and unobserved; unblocking W is head-to-tail and unobserved; unblocking N is head-to-tail and unobserved; unblocking

False => M is not marginally independent of S given no condition as two paths are feasible.

b) W ⊥ R | {N, H} (W is conditionally independent of R given {N, H}

Paths	Status	Comment
W-N-R	Blocking	N is head-to-tail and observed; blocking
W-H-R	Blocking	H is tail-to-tail and observed; blocking
W-H-N-R	Blocking	H is tail-to-tail and observed; blocking

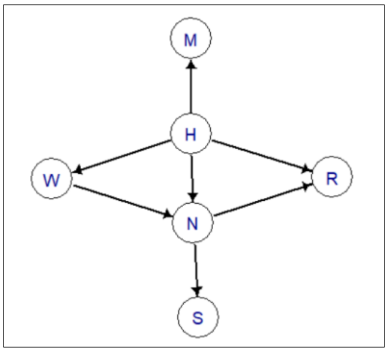
True => W is independent of R given N and H.

c) {R,S} ⊥ W|H

Paths	Status	Comment
R-N-W	Unblocking	N is tail-to-head and unobserved; unblocking
R-H-W	Blocking	H is tail-to-tail and observed; blocking
S-N-W	Unblocking	N is tail-to-head and unobserved; unblocking
S-N-H-W	Blocking	N is tail-to-head and unobserved; unblocking H is tail-to-tail and observed; blocking
S-N-H-R-W	Blocking	N is tail-to-head and unobserved; unblocking H is tail-to-tail and observed; blocking

False => {R,S} is not independent of W given H, as two paths are feasible.

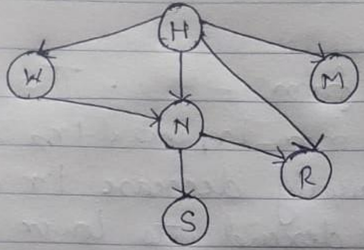
1.5 R program output for 1.4.a to c:




```
> dSep (dag , first ="M", second ="S", cond = NULL)
[1] FALSE
> dSep (dag , first ="W", second ="R", cond =c("N","H"))
[1] TRUE
> dSep (dag , first =c("R","S"), second ="W", cond ="H")
[1] FALSE
```

1.6 Answer a and b:

1.6



$P(W \mid S = low, R = low)$. Variable elimination order is N, H, M

i) $$P(W \mid S = low, R = low) = \frac{P(W, S = low, R = low)}{P(S = low, R = low)}$$

$$= \frac{P(W, S = low, R = low)}{\sum_W P(W, S = low, R = low)}$$

ii) Constructing a factor for each conditional probability :-

$$P(W, S, R) = \sum_{H,N,M} P(H, W, N, M, S, R)$$

$$= \sum_{H,N,M} P(H) \cdot P(W \mid H) \cdot P(N \mid H, W) \cdot P(M \mid H) \cdot P(S \mid N) \cdot P(R \mid H, N)$$

$$= \sum_{H,N,M} f_0(H) \cdot f_1(W, H) \cdot f_2(N, H, W) \cdot f_3(M, H) \, f_4(S, N) \cdot f_5(R, H, N)$$

iii) Assigning observed variables, observed values (states) :-

Observe : $S = low$, $R = low$

$$P(W, S = low, R = low) = \sum_{H,N,M} f_0(H) \cdot f_1(W, H) \cdot f_2(N, H, W) \, f_3(M, H) \cdot f_6(N) \cdot f_7(H, N)$$

where ; $f_4(S, N)_{S = low} = f_6(N)$

$$f_5(R, H, N)_{R = low} = f_7(H, N)$$

iv) Decomposing sum: Elimination order N, H, M (opposite to arrow/direction).

1) Eliminating N :-

$$P(W, S=\text{low}, R=\text{low}) = \sum_{H,M} f_0(H) \cdot f_1(W,H) \cdot f_3(M,H) \cdot \sum_{N} f_6(N) \cdot f_2(N,H,W) \cdot f_7(H,N)$$

$$= \sum_{H,M} f_0(H) \cdot f_1(W,H) \cdot f_3(M,H) \cdot f_8(H,W)$$

2) Eliminating H :-

$$P(W, S=\text{low}, R=\text{low}) = \sum_{M} \sum_{H} f_0(H) \cdot f_1(W,H) \cdot f_3(M,H) \cdot f_8(H,W)$$

$$= f_9(W, M)$$

3) Eliminating M :-

$$P(W, S=\text{low}, R=\text{low}) = \Sigma f_{10}(W)$$

The posterior distribution over tampering is given by

$$P(W, S=\text{low}, R=\text{low}) = \frac{f_{10}(W)}{\Sigma_W f_{10}(W)}$$

1.6(b) Treewidth of the network :-

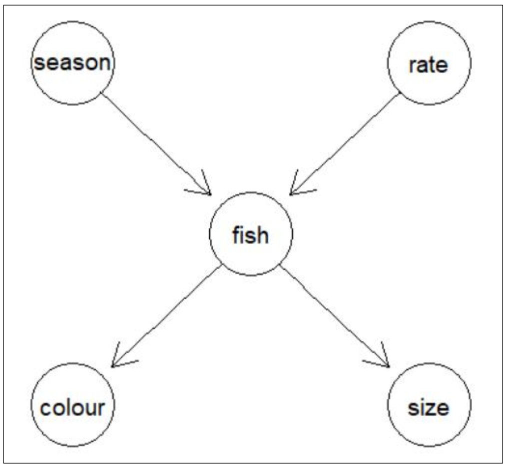
Node eliminated	Variables created	From
N	2	$f_8(H,W)$
H	2	$f_9(W,M)$
M	1	$f_{10}(W)$

The maximum number of variables in a factor created by summing out a variable, given the elimination ordering is 2.

∴ Treewidth = 2.

Q2) Implementing a Bayesian network in R and performing inference:

2.1.a. Belief network for the given distribution:



2.1.b. Probability tables obtained from the R output:

Given probabilities

$P(A = wet) = 0.3$	$P(B = high) = 0.2$
$p(C = bass \mid A = wet, B = high) = 0.4$	$p(C = bass \mid A = dry, B = high) = 0.5$
$p(C = bass \mid A = wet, B = low) = 0.6$	$p(C = bass \mid A = dry, B = low) = 0.3$
$p(D = light \mid C = bass) = 0.2$	$p(D = medium \mid C = bass) = 0.4$
$p(D = light \mid C = cod) = 0.5$	$p(D = medium \mid C = cod) = 0.3$
$p(E = wide \mid C = bass) = 0.6$	$p(E = wide \mid C = cod) = 0.4$

I. Probability for A (Season)

```
> prob_list$season
season
wet dry
0.3 0.7
```

II. Probability for B (River flow rate)

```
> prob_list$rate
rate
high  low
0.2  0.8
```

III. Probability for C (Fish species)

```
> prob_list$fish
, , rate = high

      season
fish    wet dry
  bass 0.4 0.5
  cod  0.6 0.5

, , rate = low

      season
fish    wet dry
  bass 0.6 0.3
  cod  0.4 0.7
```

IV. Probability for D (Colour)

```
> prob_list$colour
        fish
colour   bass cod
  light   0.2 0.5
  medium  0.4 0.3
  dark    0.4 0.2
```

V. Probability for E (Size)

```
> prob_list$size
      fish
size   bass cod
  wide  0.6 0.4
  thin  0.4 0.6
```

2.2 R program to compute the following probabilities:

Note: Order of levels for season and river flow rate respectively impacts the overall probability calculation even if probability table output matches with the given probabilities. For eg. if we give levels as (wet, dry) then the probabilities will be different if we give levels as (dry, wet) along with changing their values according to the levels. I have followed practical i.e. flow as per given CPT.

a) P(E=Thin|B=Low) = 0.522

```
size
wide  thin
0.478 0.522
```

b) P(C=Cod|D=Dark,A=dry) = 0.4925373

```
fish
bass       cod
0.5074627 0.4925373
```

c) P(D,C) => joint distribution of colors and fish species.

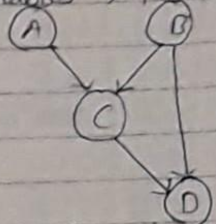
```
               colour
fish    light medium   dark
bass 0.0812 0.1624 0.1624
cod  0.2970 0.1782 0.1188
```

d) P(C)=> marginal distribution of fish species.

```
fish
bass   cod
0.406 0.594
```

Q3) Computed manually:

Q 3 Binary variables A, B, C, D



A=0	A=1
α	$1-\alpha$

B=0	B=1
θ	$1-\theta$

A	B	C=0	C=1
0	0	0.6	0.4
0	1	0.3	0.7
1	0	γ	$(1-\gamma)$
1	1	0.8	0.2

B	C	D=0	D=1
0	0	0.4	0.6
0	1	0.8	0.2
1	0	0.2	0.8
1	1	β	$(1-\beta)$

3.1 $A \perp B \Rightarrow$ A is independent of B?

Soln.

Paths	Status	Comments
A - C - B	Blocking	C node is head-to-head and unobserved.
A - C - D - B	Blocking	i) C node becomes head-to-tail and unobserved, thus unblocking ii) D node is a head-to-head and unobserved, thus blocking

There are two paths available to go from A to B i.e A-C-B and A-C-D-B, but both the paths are blocked or d-separated, thus $A \perp B$ i.e. A and B are deseparated.

3.2) $P(D=1 \mid A=1, B=1)$ in terms of β only.

Solⁿ:- $P(D=1 \mid A=1, B=1) = \dfrac{P(D=1, A=1, B=1)}{P(A=1, B=1)}$

$= \dfrac{\sum_C P(C, D=1, A=1, B=1)}{P(A=1) \times P(B=1)} \quad \{\because A \perp B\}$

$= \cancel{P(A) \cdot P(B) \cdot P(}$

$$= \frac{\sum_C P(A=1) \times P(B=1) \times P(C \mid A=1, B=1) \times P(D=1 \mid C, B=1)}{P(A=1) \times P(B=1)}$$

$$= \sum_C P(C \mid A=1, B=1) \times P(D=1 \mid C, B=1)$$

$$= P(C=0 \mid A=1, B=1) \times P(D=1 \mid C=0, B=1) + P(C=1 \mid A=1, B=1) \times P(D=1 \mid C=1, B=1)$$

$$= (0.8 \times 0.8) + (0.2 \times (1-\beta))$$

$$= 0.64 + 0.2 - 0.2\beta$$

$$= 0.84 - 0.2\beta$$

$\therefore$ $P(D=1 \mid A=1, B=1)$ in terms of β is $= 0.84 - 0.2\beta$

3.3 Summary of data:-

Count of	A	B	C	D
0	7	3	13	6
1	13	17	7	14

i) $A=0 = \underline{\alpha} = \frac{7}{20} = 0.35$ ii) $A=1 = \underline{1-\alpha} = 0.65$

iii) $B=0 = \underline{\theta} = \frac{3}{20} = 0.15$ iv) $B=1 = \underline{1-\theta} = 0.85$

v) $\underline{\gamma} = \frac{\#(C=0, A=1, B=0)}{\#(A=1, B=0)} = \frac{2}{3} \approx 0.67$

vi) $\underline{(1-\gamma)} = 1 - 0.67 = 0.33 = \frac{\#(C=1, A=1, B=0)}{\#(A=1, B=0)}$

vii) $\beta = P(D=0 \mid B=1, C=1)$

$$= \frac{\#(D=0, B=1, C=1)}{\#(B=1, C=1)}$$

$$= \frac{2}{6}$$

$$\approx 0.33$$

ix) $(1-\beta) = P(D=1 \mid B=1, C=1)$

$$= 1 - 0.33$$

$$\approx 0.67$$

3.4) $P(D=1 \mid A=1, B=1) = 0.84 - 0.2\beta$ {From 3.3}

$$= 0.84 - 0.2(0.33)$$

$$= 0.774$$

$\therefore$ $P(D=1 \mid A=1, B=1) = 0.774$

Q4) Bayesian Structure Learning:

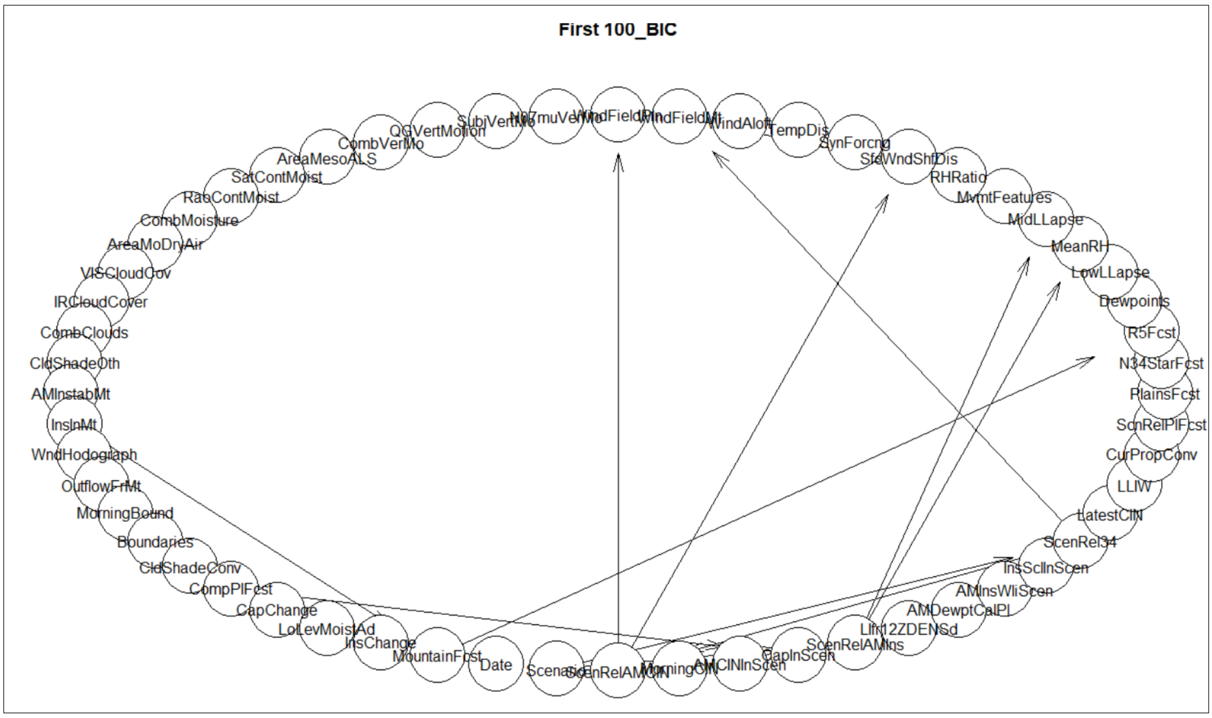
4.1.a. Scores for BIC and BDe:

Dataset	BIC	BDe
First 100	-6429.161	-5980.807
First 1,000	-52906.94	-52256.44
First 10,000	-498383.8	-497403.6

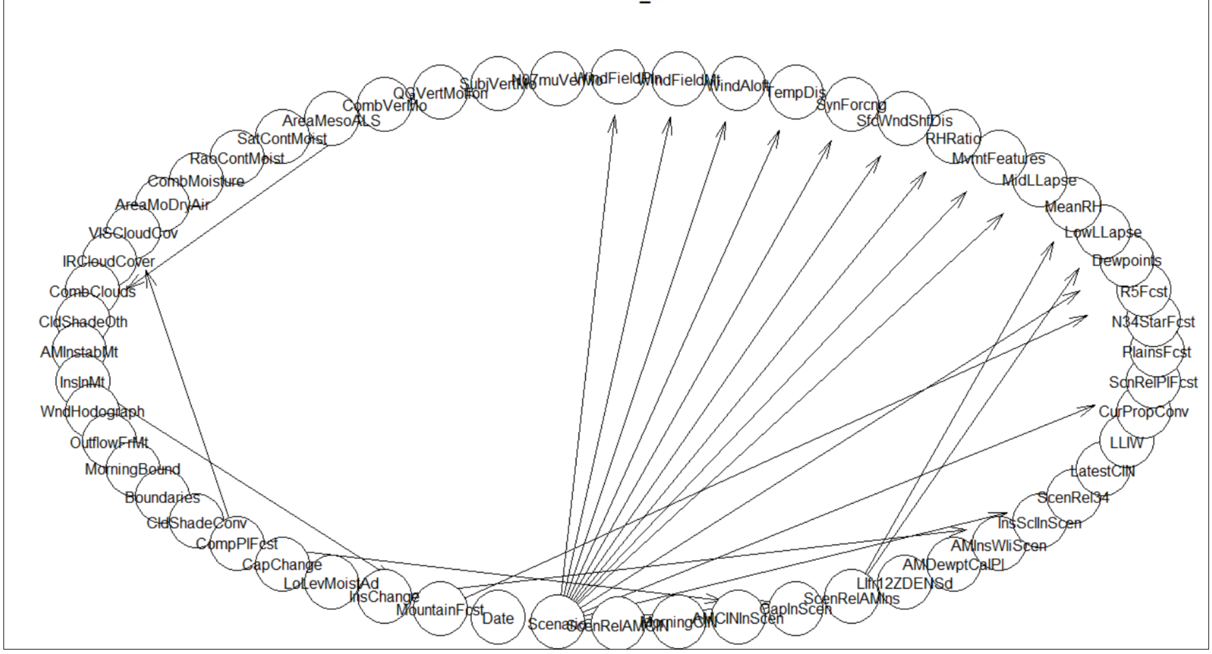
Number of undirected and directed arcs for each model:

Dataset	BIC (Arcs)		BDe (Arcs)	
	Undirected	Directed	Undirected	Directed
First 100	0	30	0	36
First 1,000	0	54	0	60
First 10,000	0	62	0	67

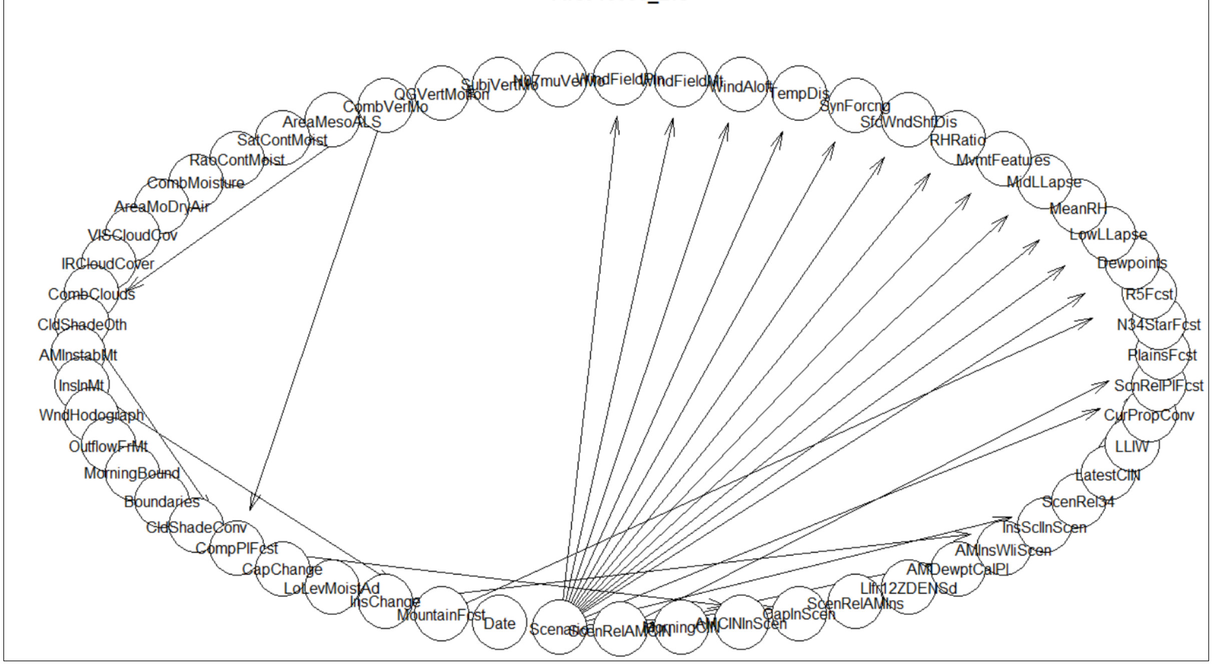
4.1.b. Plots for each fit:

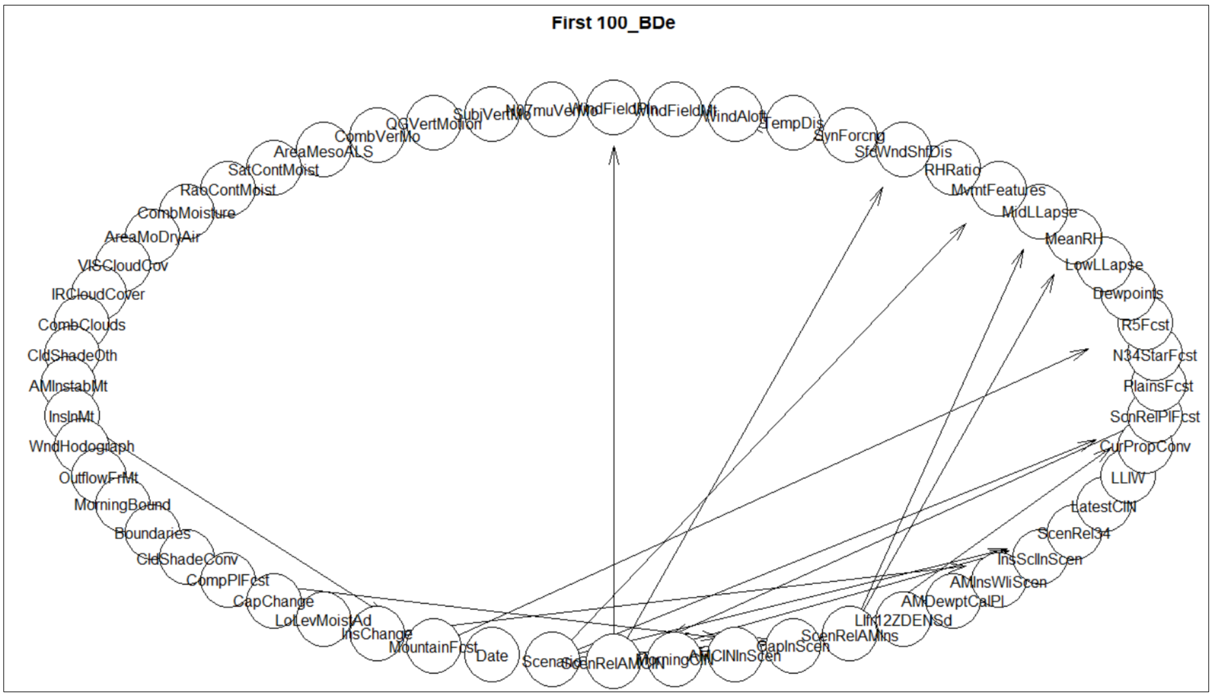


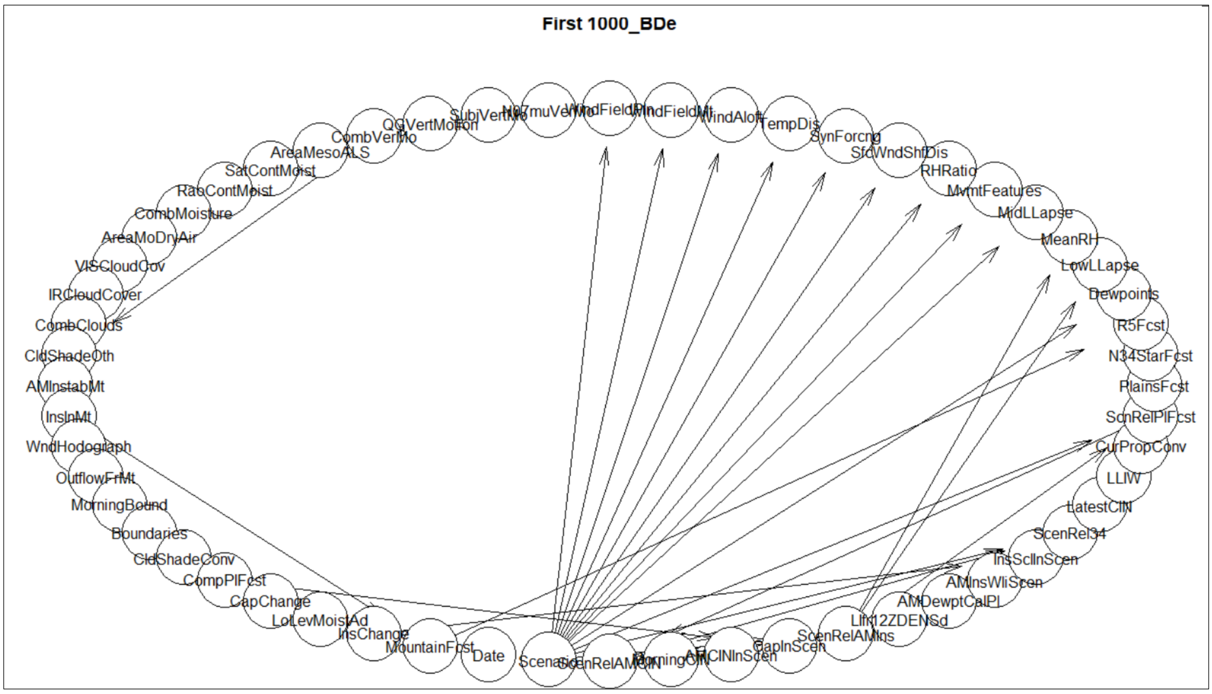
First 1000_BIC

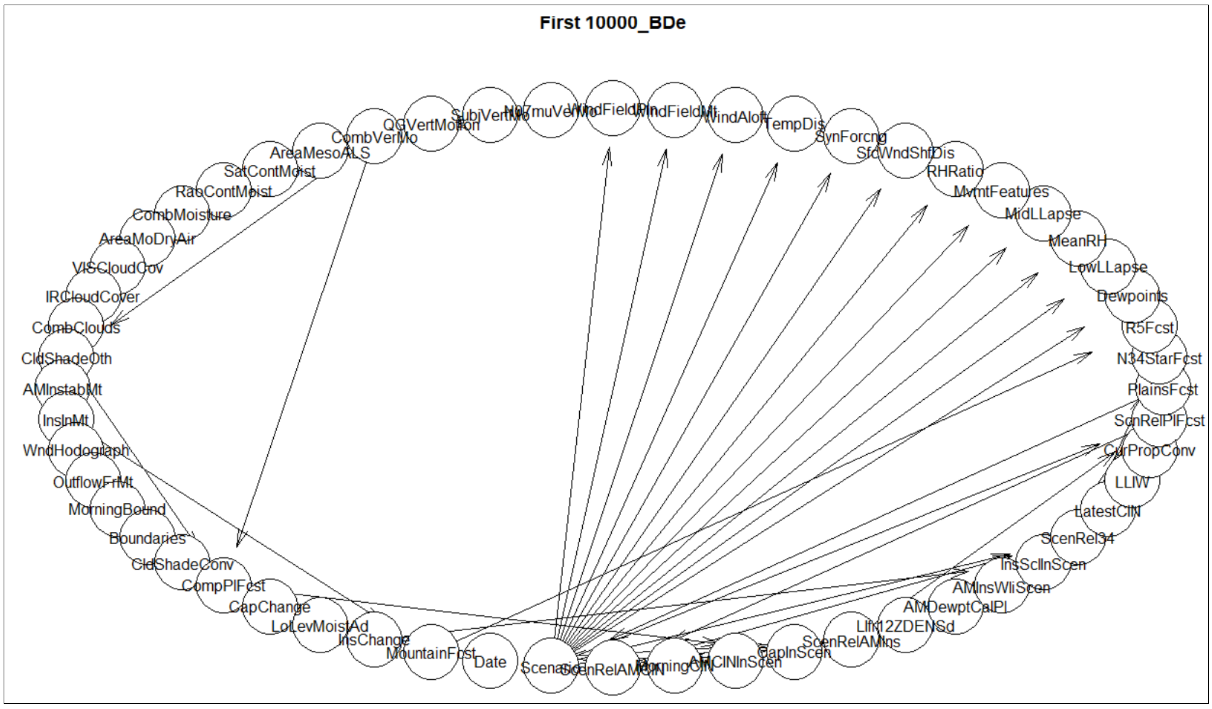


First 10000_BIC




First 100_BDe
QGVertMotion
SubjVertMo
N07muVerMo
WindFieldPln
WindFieldMt
WindAloft
TempDis
SynForcng
SfcWndShfDis
RHRatio
MvmtFeatures
MidLLapse
MeanRH
LowLLapse
Dewpoints
R5Fcst
N34StarFcst
PlainsFcst
ScnRelPlFcst
CurPropConv
LLIW
LatestCIN
ScenRel34
InsSclInScen
AMInsWliScen
AMDewptCalPl
LIfr12ZDENSd
ScenRelAMIns
CapInScen
AMCINInScen
MorningCIN
ScenRelAMCIN
Scenario
Date
MountainFcst
InsChange
LoLevMoistAd
CapChange
CompPlFcst
CldShadeConv
Boundaries
MorningBound
OutflowFrMt
WndHodograph
InsInMt
AMInstabMt
CldShadeOth
CombClouds
IRCloudCover
VISCloudCov
AreaMoDryAir
CombMoisture
RaoContMoist
SatContMoist
AreaMesoALS
CombVerMo


First 1000_BDe
QGVertMotion
SubjVertMo
N07muVerMo
WindFieldPln
WindFieldMt
WindAloft
TempDis
SynForcng
SfcWndShfDis
RHRatio
MvmtFeatures
MidLLapse
MeanRH
LowLLapse
Dewpoints
R5Fcst
N34StarFcst
PlainsFcst
ScnRelPlFcst
CurPropConv
LLIW
LatestCIN
ScenRel34
InsSclInScen
AMInsWliScen
AMDewptCalPl
LIfr12ZDENSd
ScenRelAMIns
CapInScen
AMCINInScen
MorningCIN
ScenRelAMCIN
Scenario
Date
MountainFcst
InsChange
LoLevMoistAd
CapChange
CompPlFcst
CldShadeConv
Boundaries
MorningBound
OutflowFrMt
WndHodograph
InsInMt
AMInstabMt
CldShadeOth
CombClouds
IRCloudCover
VISCloudCov
AreaMoDryAir
CombMoisture
RaoContMoist
SatContMoist
AreaMesoALS
CombVerMo



4.2) Comparison between BIC and BDE across different sample sizes:

Table 1

Dataset	BIC	BDe	Diff	BIC_Growth	BDe_Growth
First 100	-6429.161	-5980.81	7.50%	-	-
First 1,000	-52906.94	-52256.4	1.24%	723%	774%
First 10,000	-498383.8	-497404	0.20%	842%	852%

Table 2

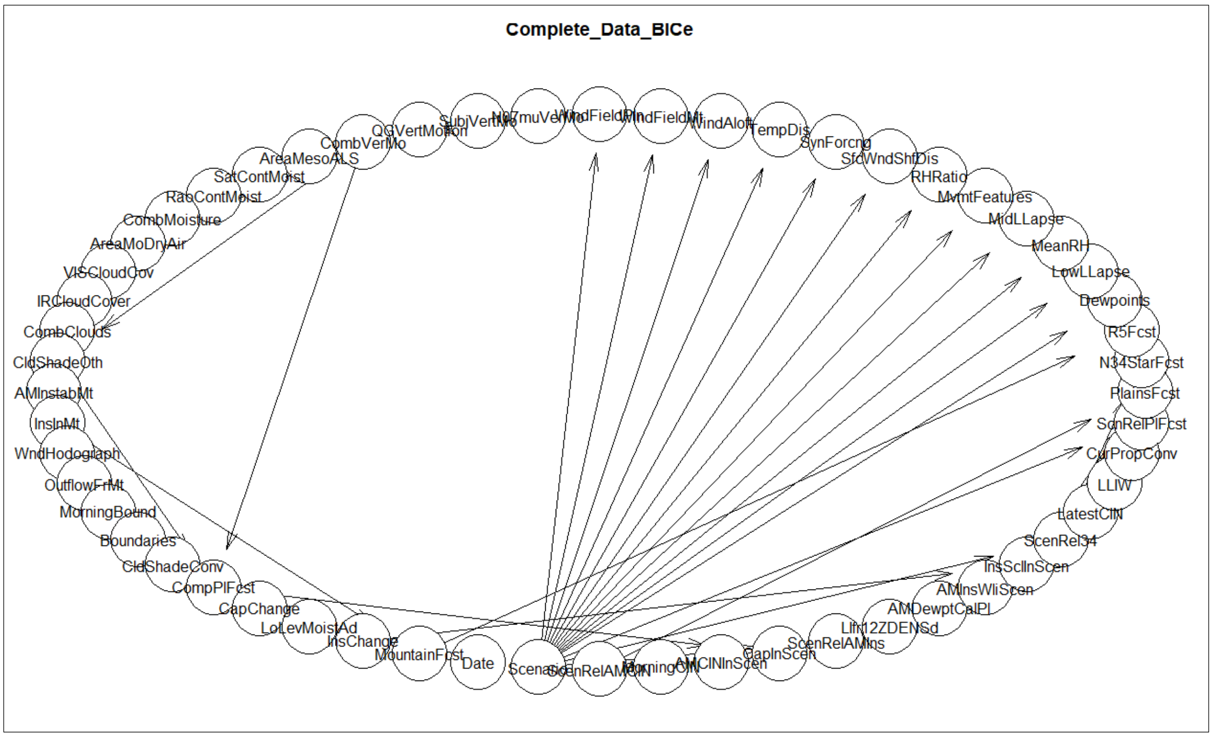
Dataset	BIC (Arcs)		BDe (Arcs)		Summary		
	Undirected	Directed	Undirected	Directed	Diff	BIC_Growth	BDe_Growth
First 100	0	30	0	36	-16.67%		
First 1,000	0	54	0	60	-10.00%	80%	67%
First 10,000	0	62	0	67	-7.46%	15%	12%

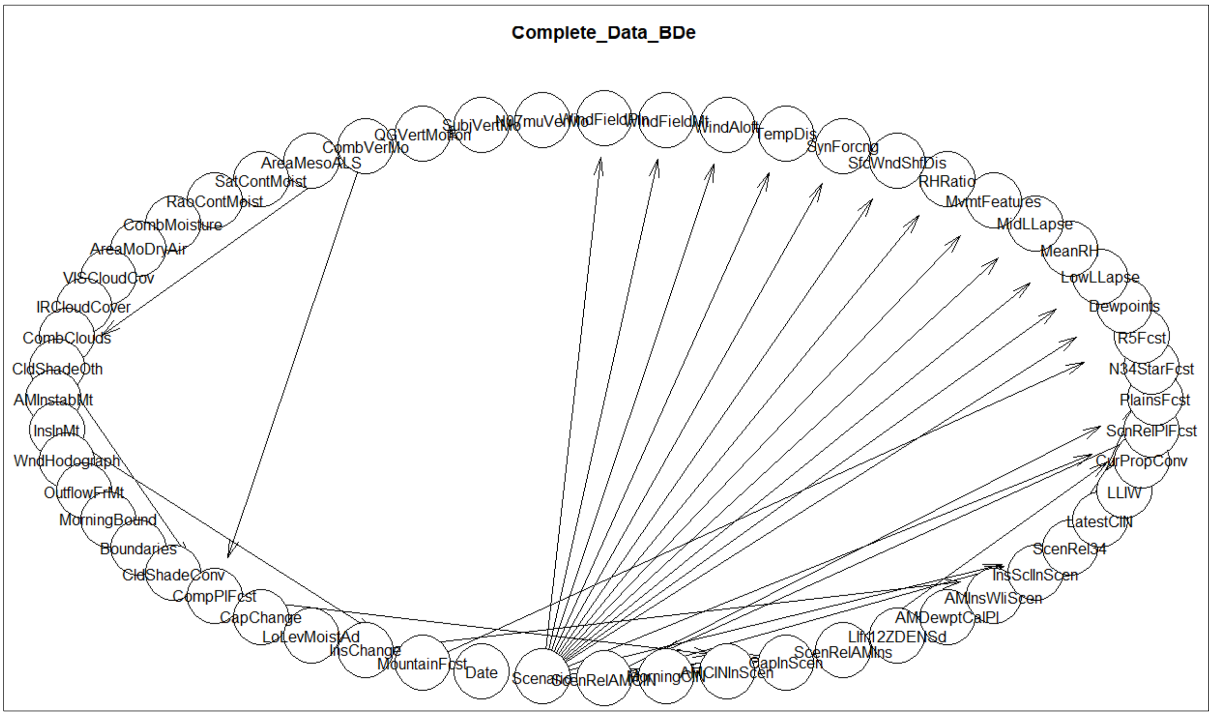
BIC penalizes more heavily and has less arcs comparatively than BDE. As data (M) increases, model complexity increases logarithmically, which reduces score for both, BIC and BDe. BIC and BDe scores become similar as sample size increases. This is evident from the drop in difference percentage in Table 1 and 2. Difference is the percentage of BIC w.r.t. BDE.
Drop in scores and increase in arcs for both BIC and BDe gives evidence that both tends to fit the model towards data as data (M) increases.

In summary, BIC score is asymptotically equivalent to BDe, which is evident from reducing difference in scores and number of arcs as sample size increases. This also implies that, at small sizes, BIC penalizes complex model more heavily than BDe. As information increases (data) fitted network move towards true network w.r.t. number of directed arcs.

4.3.a. BIC and BDe score for complete dataset:

Value	BIC	BDe
Score	-990474.8	-989470
Arcs (Directed)	64	70





4.3.b. Comparison between subset dataset models and complete dataset model:

Number of arcs BIC fitted network is closer to the number of arcs in true network. True network has 66 arcs, BIC has 64 and BDE has 70 arcs.

1) Summary comparison:
   i) True network's structure is different than the structure of BIC and BDe.

```
> all.equal(true_network,bicnet20000)
[1] "Different number of directed/undirected arcs"
> all.equal(true_network,bdenet20000)
[1] "Different number of directed/undirected arcs"
```

   ii) Based on the below table, we can say that BIC network is closest to the true network comparatively. It has more true positive arcs, i.e. arcs which are present in target as well. BDe has more false positive arcs, i.e. arcs which are not present in target but present in the fitted network. BIC has less false negative arcs, i.e. arcs which are missed by the fitted network w.r.t. true network.

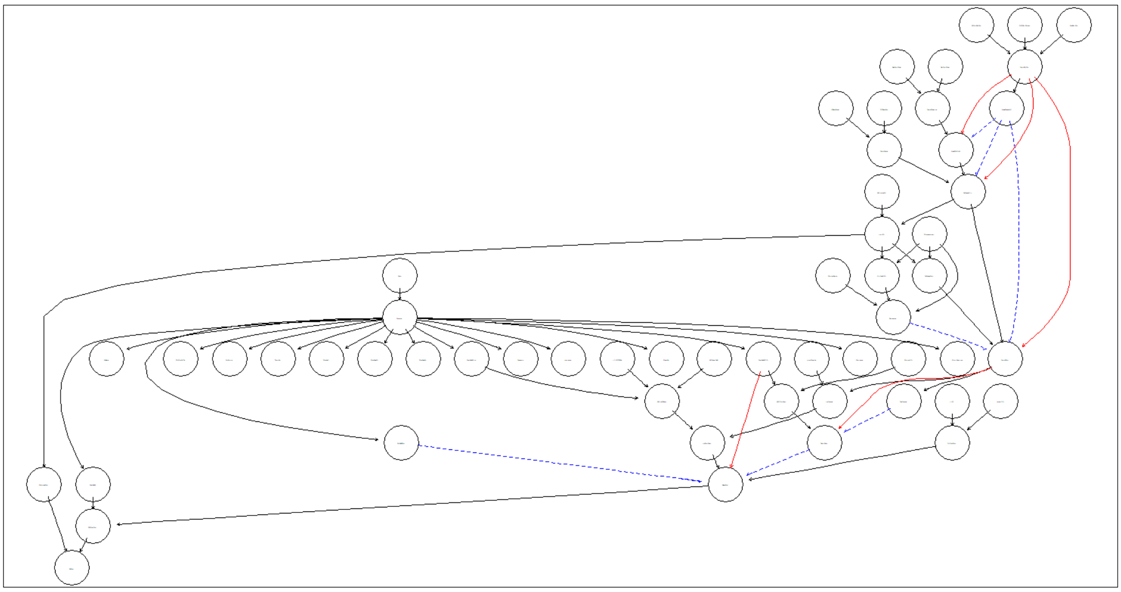
Score	True Positive	False Positive	False negative
BIC	59	5	7
BDE	57	13	9

2) Structural distance:
   Hamming distance, which is the count of arcs that are different between two networks is less between BIC fitted network and true network comparatively.

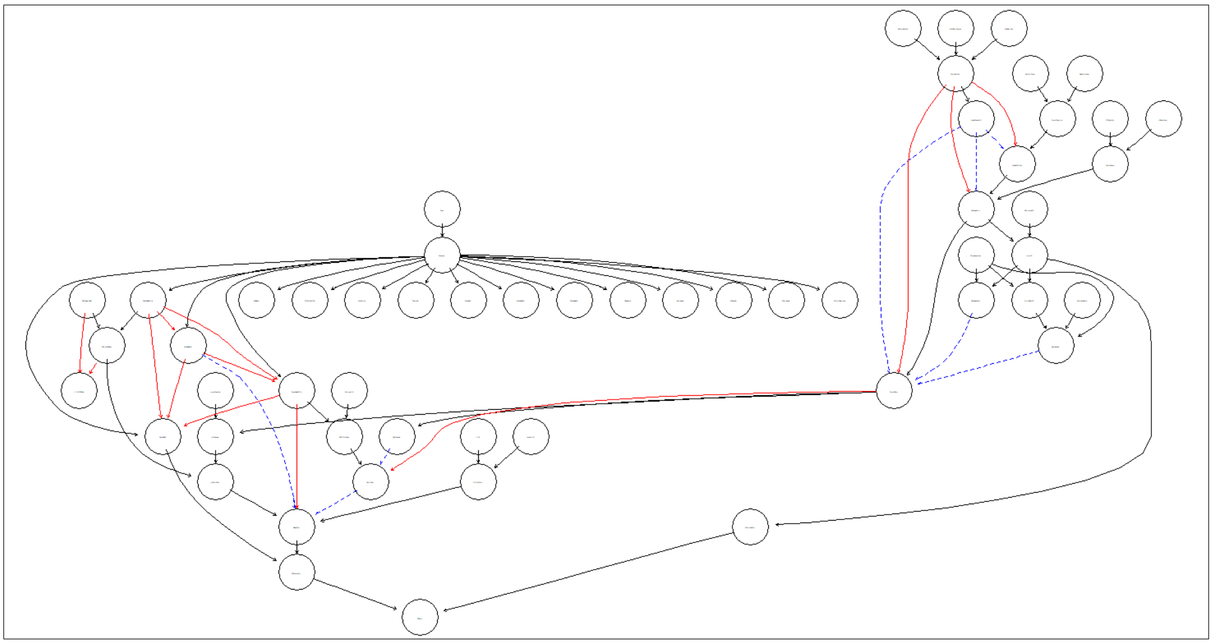
```
> hamming(true_network,bicnet20000)
[1] 12
> hamming(true_network,bdenet20000)
[1] 20
```

3) Graphical comparison:
   We can see the same result of compare function using graphical form. In this we can see, false positive arcs are in red, false negative arcs are in blue and dashed, true positive arcs are in black
   Thus, we can see 5 red and 7 blue arcs in first figure which is comparison between BIC fitted network and true network.
   For BDe, we can see 13 red and 9 blue lines in comparison with true network.

BIC fitted network comparison with true network



BDe fitted network comparison with true network



4.3.c. Fitting data to the network using BIC score and computing conditional probability distribution table:

```
Parameters of node CombClouds (multinomial distribution)

Conditional probability table:

, , IRCloudCover = Clear

          VISCloudCov
CombClouds       Clear      Cloudy          PC
    Clear  0.980757483 0.073929961 0.496460825
    Cloudy 0.000000000 0.801556420 0.044910910
    PC     0.019242517 0.124513619 0.458628265

, , IRCloudCover = Cloudy

          VISCloudCov
CombClouds       Clear      Cloudy          PC
    Clear  0.500905797 0.018248175 0.027590848
    Cloudy 0.087862319 0.941605839 0.442799462
    PC     0.411231884 0.040145985 0.529609690

, , IRCloudCover = PC

          VISCloudCov
CombClouds       Clear      Cloudy          PC
    Clear  0.693900392 0.008705114 0.104643413
    Cloudy 0.019585898 0.871599565 0.103532548
    PC     0.286513710 0.119695321 0.791824039
```

4.3.d

Probability of below with set.seed(1) is;

P(CombClouds ="Cloudy" | MeanRH = "VeryMoist", IRCloudCover ="Cloudy") = 0.3423423

```
> cpquery(fittedParams, event = (CombClouds=="Cloudy"),
+         evidence = ((MeanRH=="VeryMoist") & (IRCloudCover =="Cloudy")))
[1] 0.3423423
```

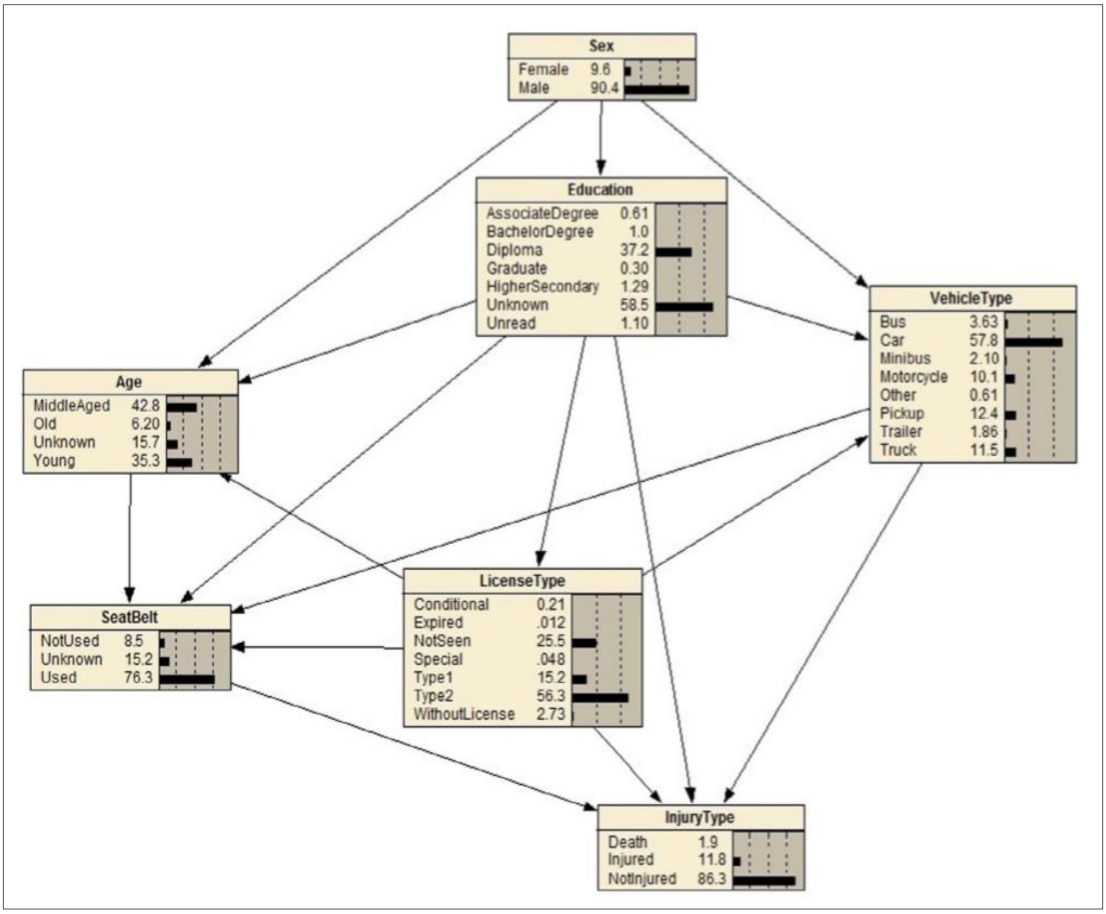
Q5) Research based question:

a) Road accident data analysis using Bayesian networks:

   i) Dataset used in the experiment was collected at Hamedan-Qazvin highway, one of the popular highways in Iran (March 2009-December 2013). Dataset consists of 65,535 accidents records collected over a period 56-month. Below are the variables;

   (1) X_1; Sex (male, female)
   (2) X_2; Education (unread, higher secondary, diploma, associate's degree, bachelor's degree, graduate degree, unknown)
   (3) X_3; Vehicle type (pickup, motorcycle, bus, minibus, truck, trailer, car, other)
   (4) X_4; Age (young, middle-aged, old, unknown)
   (5) X_5; License type (type 1, type2, expired, conditional, special, not seen, without license),
   (6) X_6; Seatbelt status (used, unused, unknown)
   (7) X_7; Injury type (not injured, injured, death)

Except the Age variable, all the variables were discrete. The Age variable was discretized separately due to implementation of the PC algorithm.
Of the 65,535 total recorded accidents, male drivers were (about 90.4%). The recorded information of drivers' education shows that percentage of drivers with the education types unread, unknown, higher secondary, diploma, associate's degree, bachelor's degree and graduate degree was 1.10, 58.53, 1.29, 37.17, 0.61, 1.00 and 0.30, respectively. Percentage of accidents for the different eight types pickup, motorcycle, bus, minibus, truck, trailer and car was reported as 12.39, 10.1, 3.64, 2.10, 11.50, 1.86 and 57.80, respectively. The remaining 0.61% were categorized as others. 42.8% of drivers were middle-aged, while 6.20% and 35.3% were categorized as old and young drives. In the remaining 15.7% of the cases, driver age was not recorded. Driver license type was reported in seven different categories. Due to the rules in Iran, not every driving license holder can drive any vehicles. For example, type 2 license holders are not allowed to drive buses, trucks and trailers. Percentage of license holders with type 1, type 2, expired, conditional and special driving licenses was 15.21, 56.29, 0.012, 0.21, 0.048, respectively. In 2.73% of the cases, drivers had no driving license and in the remaining 25.5%, it was reported as unseen. 76.3% of drivers had worn their seat belt, while 8.5% of them had not used their seat belt. In the remaining 15.2% of the accidents, seat belt use was reported unknown. Of the 65,535 recorded accidents, 1.9% led to driver death and in 11.8% of the cases, injuries were reported. The remaining 86.3% of the accidents led to no significant injuries. All the percentages described here are summarized in below figure.



ii) PC algorithm has been used for learning the Bayesian network structure.
iii) Netica software has been used to build and visualize the Bayesian Network.
Website: https://www.norsys.com/

iv)

I) P(InjuryType=Injured | SeatBelt=Used, VehicleType=Car, Education=Diploma, LicenseType=Type2) = 1.66%

II) P(InjuryType=Death | SeatBelt=Used, VehicleType=Car, Education=Diploma, LicenseType=Type2) = 0.12%

III) P(InjuryType=Injured | SeatBelt=NotUsed, VehcileType=Car, Education=Diploma, LicenseType=Type2) = 11.40%

IV) P(InjuryType=Death | SeatBelt=NotUsed, VehcileType=Car, Education=Diploma, LicenseType=Type2) = 1%

V) Based on probability values we can conclude that wearing seat belt reduces probability of being injured and death by 86% and 88% respectively.

[(1.66-11.4)/11.4] = -86% ➔ Calculation for Injury
[(0.12-1)/1] = -88% ➔ Calculation for Death

b) Brief : “Bayesian Network Based Prediction Algorithm of Stock Price Return” by Yi Zuo and Eisuke Kita:

Introduction:

In Financial Econometrics, there are various time-series models to model stock price return. For eg, Autoregressive(AR), Moving Average(MA), Autoregressive Moving Average(ARMA), etc. Time-series models past stock returns under the assumption of stationarity and the error term following normal distribution. Stock returns does not follow normal distribution as per studies in econophysics. Thus, these models cannot predict stock returns accurately.
Authors of this paper proposed a stock price prediction model using Bayesian Network.

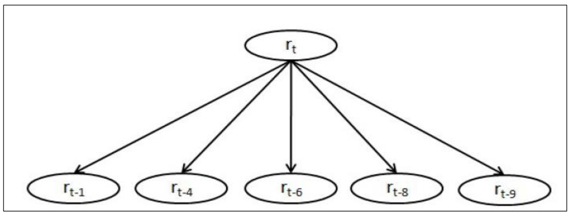
Steps:

Stock price of TOYOTA Motor Corporation from 22-Feb-1985 to 30-Dec-2008 was used to fit the model. Prediction of stock returns was done from 1-Jan to 30-Mar-2009.

Stock price discretised:
Stock price return is defined as below where P_t denotes the closing price at time t.

$$\boxed{r_t = (\ln P_t - \ln P_{t-1}) \times 100}$$

Ward method is used to discretised stock returns to maximise its occurrence and to define clusters so that Euclid distances from samples to the cluster centers are minimized. This gave below clusters which was used to form BN. Also, due to these clusters time-series element of each data point was preserved.



Building BN:

Conditional probability table was computed considering relationship between $\mathbf{x_i}$ and $\mathbf{x_j}$ as parent or child. Conditional dependency probability between x_i and x_j is represented with $P(x_i|x_j)$, i.e. conditional probability of x_i given x_j. The strength of relationships is quantified using CPT below; where M and L are total numbers of the states of $Pa(x_i)$ and x_i respectively; and Y^m and X^l denote the m-th state of $Pa(x_i)$ and the l-th of x_i respectively.

$$\boxed{\begin{array}{c} P(X^1|Y^1), P(X^2|Y^1), \cdots, P(X^L|Y^1) \\ \vdots \qquad\qquad \vdots \\ P(X^1|Y^M), P(X^2|Y^M), \cdots, P(X^L|Y^M) \end{array}}$$

Bayesian networks (BN) were determined using K2 algorithm using metrics [9,10,12] as the estimator of the network.

$$K2 = \prod_{i=1}^{N} \prod_{j=1}^{M} \frac{(L-1)!}{(N_{ij}+L-1)!} \prod_{k=1}^{L} N_{ijk}!$$

where

$$N_{ij} = \sum_{k=1}^{L} N_{ijk}.$$

N, L, and M denote total number of nodes and states for xi and $Pa(x_i)$, respectively. The notation N_{ijk} denotes the number of samples of $x_i = X_k$ when $Pa(x_i) = Y_j$. K2 algorithm determines the network from the totally ordered set of the random variables which is summarized as follows.

Probability $P(x_i = X^l | e)$ is given by the marginalization as follows where e is an evidence of the random variable.

$$P(x_i = X^l|e) = \frac{\sum_{j=1, j\neq i}^{N} \sum_{x_j=X^1}^{X^L} P(x_1, \cdots, x_i = X^l, \cdots, x_N, e)}{\sum_{j=1}^{N} \sum_{x_j=X^1}^{X^L} P(x_1, \cdots, x_N, e)} \tag{9}$$

where the notation $\sum_{x_j=X^1}^{X^L}$ denotes the summation over all states $X^1, X^2, \cdots X^L$ of the random variable x_j.

Prediction:

Stock return r_t was determined from the discrete value set of the stock prices so that its occurrence probability $P(r^l \mid B)$ is maximized where is B is BN.

$$\boxed{r_t = \arg\max_{r^l} (P(r^l|B))}$$

References:

Watada, J., Watanabe, T., Phillips-Wren, G., Howlett, R. and Jain, L., 2012. *Intelligent Decision Technologies*. Berlin, Heidelberg: Springer, pp.397-406.

References:

Karimnezhad, A. and Moradi, F., 2016. Road accident data analysis using Bayesian networks. *Transportation Letters*, 9(1), pp.12-19.

Bnlearn.com. 2020. *Bnlearn - Comparing Bayesian Network Structures*. [online] Available at: <https://www.bnlearn.com/examples/compare-dags/> [Accessed 26 May 2020].

Nagarajan, R., Scutari, M. and Lèbre, S., 2013. *Bayesian Networks In R*. New York, NY: Springer.