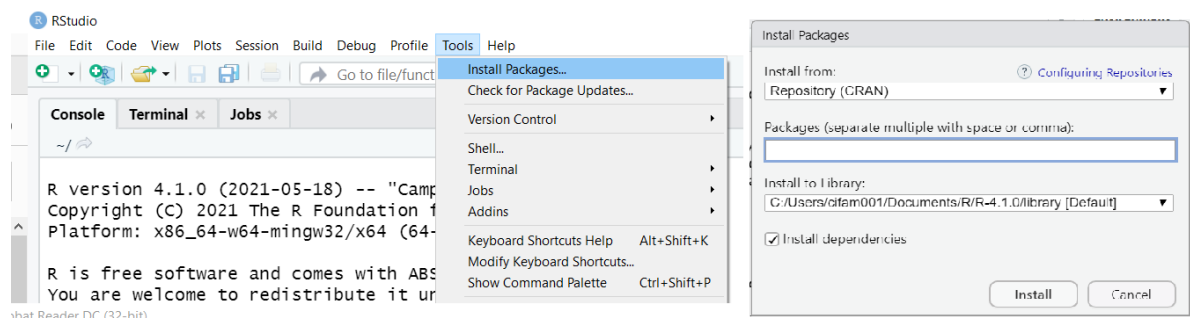


Practical 2: Bayesian Network Inference

I. Bayesian network inference with gRain

1. Start R or Rstudio (Recommended).
2. Install the gRain package.
 - a. Select Tools → install packages → specify the name of the package you want to install.
 - b. Tick the “Install dependencies” box to install all the dependent packages.
 - c. Click “Install”.



note: You may need to install additional dependencies from Bioconductor

```
> library(gRain)
Loading required package: gRbase
Required packages from Bioconductor are not installed: RBGL
Please execute these lines and re-install gRbase again:
source("https://bioconductor.org/biocLite.R");biocLite(c("graph", "RBGL",
"Rgraphviz"))
```

if so, please run

```
if (!requireNamespace("BiocManager", quietly = TRUE))
  install.packages("BiocManager")
```

```
BiocManager::install(version = "3.17")
#install additional dependencies
BiocManager::install(c("graph", "RBGL", "Rgraphviz"))
```

3. Run the following codes to create the conditional probability table for the Asia network



This example uses the following binary variables: `asia`, `smoker`, `tub` (tuberculosis), `lung` (lung cancer), `bronc` (bronchitis), `either` (either tuberculosis or lung cancer), `dysp` (dyspnoea) and `xray`. Each variable is binary and can take the values “yes” and “no” and `either` is a logical variable which is **true** (`yes`) if either `tub` or `lung` are **true** (`yes`) and **false** (`no`) otherwise.

```
library(gRain)
yn <- c("yes", "no")
a <- ctable(~asia, values=c(1,99), levels=yn)
t.a <- ctable(~tub|asia, values=c(5,95,1,99), levels=yn)
s <- ctable(~smoke, values=c(5,5), levels=yn)
l.s <- ctable(~lung|smoke, values=c(1,9,1,99), levels=yn)
b.s <- ctable(~bronc|smoke, values=c(6,4,3,7), levels=yn)
e.lt <- ctable(~either|lung:tub, values=c(1,0,1,0,1,0,0,1), levels=yn)
x.e <- ctable(~xray|either, values=c(98,2,5,95), levels=yn)
d.be <- ctable(~dysp|bronc:either, values=c(9,1,7,3,8,2,1,9), levels=yn)
plist <- compileCPT(list(a, t.a, s, l.s, b.s, e.lt, x.e, d.be))
plist
#Checking the (conditional) probability of some nodes
plist$tub
plist$either
#Another way to visualize tables
plist$tub %>% as.data.frame.table
```

```
> plist$tub %>% as.data.frame.table
```

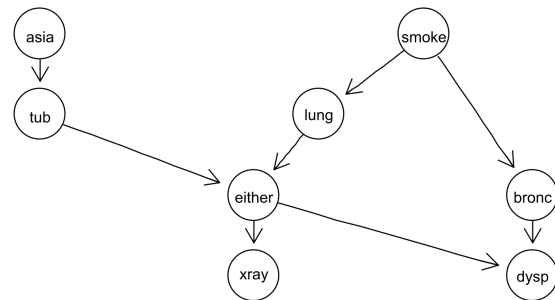
	tub	asia	Freq
1	yes	yes	0.05
2	no	yes	0.95
3	yes	no	0.01
4	no	no	0.99

Table from: `t.a <- ctable(~tub|asia, values=c(5,95,1,99), levels=yn)`

It is read as $P(\text{tub} = \text{yes} | \text{asia} = \text{yes}) = 0.05$, $P(\text{tub} = \text{no} | \text{asia} = \text{yes}) = 0.95$, and so on...

4. Draw the network:

```
net1=grain(plist)
plot(net1) # undirected network
plot(net1$dag) # directed network
```



5. Query the marginal probabilities $P(\text{lung})$ and $P(\text{bronc})$:

```
querygrain(net1, nodes=c("lung","bronc"), type="marginal")
```

\$lung		\$bronc	
lung		bronc	
yes	no	yes	no
0.055	0.945	0.45	0.55

6. Query the joint probability $P(\text{lung}, \text{bronc})$:

```
querygrain(net1, nodes=c("lung","bronc"), type="joint")
```

	bronc	
lung	yes	no
yes	0.0315	0.0235
no	0.4185	0.5265

7. Query the conditional probability $P(\text{lung}|\text{bronc})$

```
querygrain(net1, nodes=c("lung","bronc"),
  , type="conditional")
```

	bronc	
lung	yes	no
yes	0.07	0.04272727
no	0.93	0.95727273

8. Calculate the following probabilities:

```
#P(lung=yes,bronc=yes)
querygrain(net1, nodes=c("lung","bronc"), type="joint")
```

	bronc	
lung	yes	no
yes	0.0315	0.0235
no	0.4185	0.5265

```
#P(bronc=yes)
```

```
querygrain(net1, nodes=c("bronc"), type="marginal")
```

		smoke	
		yes	no
bronc			
yes	no	0.0235	
0.45	0.55	0.5265	

```
#P(lung=yes|smoke=yes)
querygrain(net1, nodes=c("lung","smoke"), type="conditional")
```

		smoke	
		yes	no
lung			
yes	yes	0.1	0.01
no	0.9	0.99	

```
#P(xray=yes|smoke=yes)
querygrain(net1, nodes=c("xray","smoke"),
, type="conditional")
```

		smoke	
		yes	no
xray			
yes	yes	0.1517048	0.06887528
no	0.8482952	0.93112472	

```
#P(xray=yes|smoke=yes, asia=yes)
querygrain(net1, nodes=c("xray","smoke","asia"),
, type="conditional")
```

		smoke	
		yes	no
xray			
yes	yes	0.18485	0.105335
no	0.81515	0.894665	

```
#P(lung=yes|asia=yes)
querygrain(net1, nodes=c("lung","asia"),
, type="conditional")
```

		asia	
		yes	no
lung			
yes	yes	0.055	0.055
no	0.945	0.945	

```
##P(bronc=yes|smoke=yes, asia=yes)
querygrain(net1, nodes=c("bronc","smoke","asia"),
, type="conditional")
```

		smoke	
		yes	no
bronc			
yes	yes	0.6	0.3
no	0.4	0.7	