**Assignment --2**

[**Problem Set 2: Exact Inference with Probabilistic Graphical Models**](https://ublearns.buffalo.edu/webapps/assignment/uploadAssignment?content_id=_4499403_1&course_id=_154023_1&group_id=&mode=view)

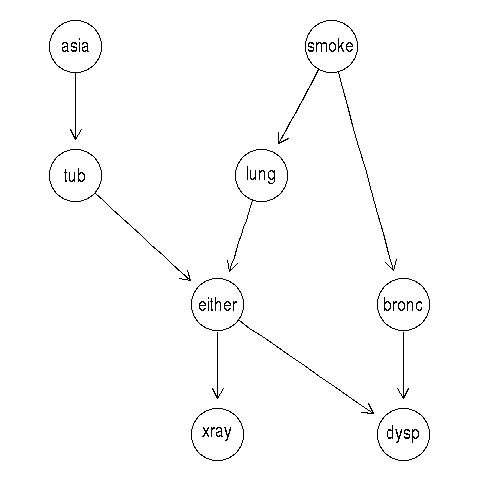
In this problem set, we will make exact inferences about probabilistic graphical models using the state-of-the-art graphical model packages in our most comfortable programming languages, and understand those exact algorithms. You can find tutorials in [Python](https://ublearns.buffalo.edu/bbcswebdav/pid-4499403-dt-content-rid-18203418_1/xid-18203418_1), R ([slides](http://people.math.aau.dk/~sorenh/misc/2014-useR-GMBN/bayesnet-slides.pdf) and [book](https://catalog.lib.buffalo.edu/vufind/Record/003839770)) and [Matlab](https://github.com/bayesnet/bnt). The function calls in different packages are different, but the point here is that we make graphical model our actionable machine learning tool in this course.

We will work with the chest clinic graphical model (below). Please moralize, triangulate and construct a junction tree from this graphical model. Then use message-passing algorithm to find the joint probability of "tub=yes, lung=yes, bronc=yes", given evidence that "asia=yes, xray=yes".

Problem 1: Draw the moral graph, triangulated graph and the junction tree. Explain why the "running intersection property" is satisfied in your junction tree.

Problem 2: Describe how the different terms on the right hand side of "p(V ) = p(a)p(t | a)p(s)p(l | s)p(b | s)p(e | t, l)p(d | e, b)p(x | e)" are distributed among the different juction tree clusters. Write out the messages using these terms and verify that the message passing algorithm indeed gives the cluster marginals.

[Optional] Problem 3: Find the joint probability with MCMC.



> library(gRain) > yn <- c("yes","no") > a <- cptable(~asia, values=c(1,99), levels=yn) > t.a <- cptable(~tub | asia, values=c(5,95,1,99), levels=yn) > s <- cptable(~smoke, values=c(5,5), levels=yn) > l.s <- cptable(~lung | smoke, values=c(1,9,1,99), levels=yn) > b.s <- cptable(~bronc | smoke, values=c(6,4,3,7), levels=yn) > e.lt <- cptable(~either | lung:tub,values=c(1,0,1,0,1,0,0,1),levels=yn) > x.e <- cptable(~xray | either, values=c(98,2,5,95), levels=yn) > d.be<-cptable(~dysp|bronc:either, values=c(9,1,7,3,8,2,1,9),levels=yn)  
> cpt.list <- compileCPT(list(a, t.a, s, l.s, b.s, e.lt, x.e, d.be))

> cpt.list$asia

asia

yes no

0.01 0.99

> cpt.list$tub asia

tub yes no

yes 0.05 0.01

no 0.95 0.99

> cpt.list$smoke smoke

yes no

0.5 0.5

> cpt.list$lung smoke

lung yes no

yes 0.1 0.01

no 0.9 0.99

> cpt.list$bronc smoke

bronc yes no

yes 0.6 0.3

no 0.4 0.7

> ftable(cpt.list$either,row.vars = 1) lung yes no

tub yes no yes no

either

yes 1 1 1 0

no 0 0 0 1

> cpt.list$xray either

xray yes no

yes 0.98 0.05

no 0.02 0.95