## Steps for Performing a d-Sep Test

###### (Shipley 2000, 2003, 2004, 2009)

1. Express the hypothesized causal relationships between the variables in the form of a directed acyclic graph
2. List each of the *k* pairs of variables in the graph that do not have an arrow between them:

|  |  |  |  |
| --- | --- | --- | --- |
| * Row | * Variable Pairs 1 | * Variable Pairs 2 | * Variable Pairs 3 |
| * 1 | * *Topography, Leaf Count* | * *Topography, Prevalence* | * *Host Habitat, Precipitation* |
| * 2 | * *Temp2004, Temp2005* | * *Temp2004, Precip2005* | * *Precip2004, Precip2005* |
| * 3 | * *Temp2004, Temp2006* | * *Temp2004, Precip2006* | * *Precip2004, Precip2006* |
| * 4 | * *Temp2004, Temp2007* | * *Temp2004, Precip2007* | * *Precip2004, Precip2007* |
| * 5 | * *Temp2004, Temp2008* | * *Temp2004, Precip2008* | * *Precip2004, Precip2008* |
| * 6 | * *Temp2004, Temp2009* | * *Temp2004, Precip2009* | * *Precip2004, Precip2009* |
| * 7 | * *Temp2004, Temp2010* | * *Temp2004, Precip2010* | * *Precip2004, Precip2010* |
| * 8 | * *Temp2004, Temp2011* | * *Temp2004, Precip2011* | * *Precip2004, Precip2011* |

This is just an example of what part of this table would look like. I can use the ggm package to find the entire list and define the basis set as described below.

1. For each of the *k* pairs of variables (*Xi,Xj*), list the set of other variables, {*Z*} that are direct causes of either *Xi* or *Xj*. The pair of variables, (*Xi,Xj*), along with its conditioning set, {*Z*}, define an independence claim, (*Xi,Xj*) | {*Z*}, and the full set of the *k* independence claims defines the basis set, *Bu*.
2. For each element in this basis set, obtain the ***exact*** probability, *Pk* that the pair (*Xi,Xj*) is statistically independent conditional on the variables *Z*. In other words, perform a regression model using an appropriate method.
3. Combine the *k* probabilities into the *C*-statistic:

This methodology can be extended into the mixed-model context using standard statistical programs with these capabilities, because you are testing multiple regressions hypothesized to be independent of each other. If too many independence claims are violated, then the model will be found to not fit the data.

It is also possible to apply this method to multilevel Bayesian (MCMC) models (McCarthy 2007) by testing the independence of the partial slopes and determining the 95% credible intervals.

### Example from Shipley 2009 on how to perform the d-sep test in a generalized multilevel context

This is to demonstrate how to assess a multilevel, repeated measures path model by applying mixed models in the tests of d-separation. I am using the supplementary material, including the data provided.

##### Path Model

Latitude --> Degree-days --> Date of bud burst --> Growth --> Live

Load the two libraries and read in the data Shipley uses

library(nlme)  
library(lme4)# masked lmList from nlme package  
data1 <- read.table("d\_sep\_ex/shipley2009.dat")  
str(data1)

## 'data.frame': 1900 obs. of 9 variables:  
## $ site : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ tree : int 1 2 3 4 5 1 2 3 4 5 ...  
## $ lat : num 40.4 40.4 40.4 40.4 40.4 ...  
## $ year : int 1970 1970 1970 1970 1970 1972 1972 1972 1972 1972 ...  
## $ Date : num 115 118 116 111 121 ...  
## $ DD : num 161 159 160 161 157 ...  
## $ Growth : num 61.4 43.8 44.7 48.2 50 ...  
## $ Survival: num 1 0.843 0.944 0.957 0.976 ...  
## $ Live : int 1 1 1 1 1 1 1 1 1 1 ...

summary(data1)

## site tree lat year   
## Min. : 1.00 Min. : 1.00 Min. :40.38 Min. :1970   
## 1st Qu.: 5.75 1st Qu.: 25.75 1st Qu.:60.60 1st Qu.:1978   
## Median :10.50 Median : 50.50 Median :64.23 Median :1988   
## Mean :10.50 Mean : 50.50 Mean :63.55 Mean :1988   
## 3rd Qu.:15.25 3rd Qu.: 75.25 3rd Qu.:66.79 3rd Qu.:1998   
## Max. :20.00 Max. :100.00 Max. :79.24 Max. :2006   
##   
## Date DD Growth Survival   
## Min. :103.3 Min. :115.1 Min. :32.71 Min. :0.1791   
## 1st Qu.:121.2 1st Qu.:136.8 1st Qu.:46.66 1st Qu.:0.9392   
## Median :127.8 Median :143.4 Median :50.46 Median :0.9882   
## Mean :126.9 Mean :143.8 Mean :50.82 Mean :0.9332   
## 3rd Qu.:132.8 3rd Qu.:150.3 3rd Qu.:55.02 3rd Qu.:0.9981   
## Max. :145.1 Max. :179.4 Max. :69.61 Max. :1.0000   
## NA's :469 NA's :469 NA's :469   
## Live   
## Min. :0.0000   
## 1st Qu.:0.0000   
## Median :1.0000   
## Mean :0.7326   
## 3rd Qu.:1.0000   
## Max. :1.0000   
##

head(data1)

## site tree lat year Date DD Growth Survival Live  
## 1 1 1 40.38063 1970 115.4956 160.5703 61.36852 0.9996238 1  
## 2 1 2 40.38063 1970 118.4959 158.9896 43.77182 0.8433521 1  
## 3 1 3 40.38063 1970 115.8836 159.9262 44.74663 0.9441110 1  
## 4 1 4 40.38063 1970 110.9889 161.1282 48.20004 0.9568525 1  
## 5 1 5 40.38063 1970 120.9946 157.3778 50.02237 0.9759584 1  
## 6 1 1 40.38063 1972 114.2315 160.6120 56.29615 0.9983398 1

Shipley (2009) describes these data in the text as a study beginning in 1970 where 20 sites are chosen with different latitudes. Five individual trees are chosen within each site, and each tree is followed every second year until 2006 or death. In each site, each year, for each individual four measurements are recorded:

1. DD - the number of growing degree days until bud break
2. Date - the Julian date (day of year) of bud break
3. Growth - individual tree growth (change in diameter)
4. Live - binary indicator of survival (1) or death (0)

I am now going to load the ggm library for identifying the directed acyclic graph and defining the basis set (a.k.a. the condition independence claims) without writing it out by hand.

"The DAG function defines the adjacency matrix of a directed acyclic graph. An adjacency matrix is a square Boolean [e.g. 0/1] matrix that is equal to the number of nodes of the graph, with a one in a given position (i,j) if there is an arrow from i to j and zero otherwise. The row names of the adjacency matrix are the nodes of the DAG."

This means composing each the regressions between variables directly connected by arrows in the graph, which is pretty easy in this simple example because each variable has no more than one direct cause.

library(ggm)

## Loading required package: igraph

dag <- DAG(Live ~ Growth, Growth ~ Date, Date ~ DD, DD ~ Latitude)  
isAcyclic(dag)# Logical test to check if the translated graph is acyclic, though a cyclic graph should return a warning, e.g.:

## [1] TRUE

DAG(x~y, y~z, z~x)

## Warning in DAG(x ~ y, y ~ z, z ~ x): The graph contains directed cycles!

## x y z  
## x 0 0 1  
## y 1 0 0  
## z 0 1 0

Bu <- basiSet(dag)# Get the basis set as a list of vectors. The first two items of each vector are the names of the two non-adjacent nodes, followed by the names of the nodes in the conditioning set, which may be empty (though not in this instance).

Following the directions of the Shipley (2009) supplement I will fit separate mixed models for each independence claim in the basis set.

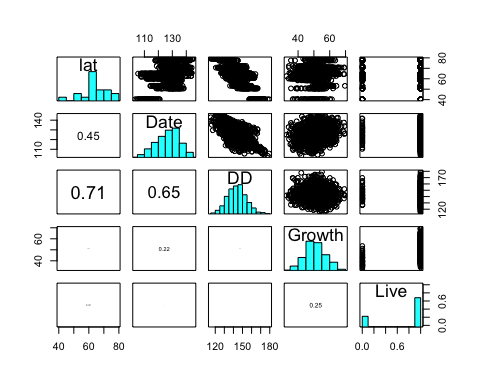
print(Bu)

## [[1]]  
## [1] "Latitude" "Date" "DD"   
##   
## [[2]]  
## [1] "Latitude" "Growth" "Date"   
##   
## [[3]]  
## [1] "Latitude" "Live" "Growth"   
##   
## [[4]]  
## [1] "DD" "Growth" "Latitude" "Date"   
##   
## [[5]]  
## [1] "DD" "Live" "Latitude" "Growth"   
##   
## [[6]]  
## [1] "Date" "Live" "DD" "Growth"

Six claims equals six models. The first two variables listed in each item are claimed to be independent based on the following set of zero or more variables.

The first independence claim is that Date is independent of Latitude after controlling for DD. So, I need to obtain the *precise* null probability for this claim. Choosing the correct modeling procedure requires identifying the appropriate distribution of the dependent variable, in this case Date. Using some helper fucntions and the pairs() function in the base graphics package I can look at the correlations and distributions simultaneously.

#pairs(subset(data1, select = c(lat, Date, DD, Growth, Live)), diag.panel = panel.hist, lower.panel = panel.cor)  
sub\_data1 <- data1[ ,c(3,5:7,9)] #subset(data1, , select = c("lat","Date","DD","Growth","Live"))  
#library(dplyr)  
#sub\_data1 <- data1 %>% select(lat, Date, DD, Growth, Live)  
  
pairs(sub\_data1, diag.panel = panel.hist, lower.panel = panel.cor)



The Date variable appears close to a normal distribution, so either the lmer() in the lme4 package or the lme() function in the nlme package are appropriate. Shipley says that the lme() prints out degrees of freedom but lmer() does not. I am going to check to see if that is still true.

fit1 <- lme(Date ~ DD + lat, data = data1,   
 random = ~1|site/tree,   
 na.action = na.omit)  
fit1lme4 <- lmer(Date ~ DD + lat + (1|site) + (1|tree), data = data1,  
 na.action = na.omit)  
summary(fit1)

## Linear mixed-effects model fit by REML  
## Data: data1   
## AIC BIC logLik  
## 4706.589 4738.173 -2347.295  
##   
## Random effects:  
## Formula: ~1 | site  
## (Intercept)  
## StdDev: 3.803355  
##   
## Formula: ~1 | tree %in% site  
## (Intercept) Residual  
## StdDev: 4.816661 1.014405  
##   
## Fixed effects: Date ~ DD + lat   
## Value Std.Error DF t-value p-value  
## (Intercept) 198.91522 7.337100 1330 27.11088 0.0000  
## DD -0.49766 0.004937 1330 -100.80609 0.0000  
## lat -0.00905 0.113477 18 -0.07976 0.9373  
## Correlation:   
## (Intr) DD   
## DD -0.132   
## lat -0.986 0.036  
##   
## Standardized Within-Group Residuals:  
## Min Q1 Med Q3 Max   
## -4.03862871 -0.60616453 -0.01426884 0.62856908 2.80922832   
##   
## Number of Observations: 1431  
## Number of Groups:   
## site tree %in% site   
## 20 100

summary(fit1lme4)

## Linear mixed model fit by REML ['lmerMod']  
## Formula: Date ~ DD + lat + (1 | site) + (1 | tree)  
## Data: data1  
##   
## REML criterion at convergence: 4694.6  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -4.0386 -0.6062 -0.0143 0.6286 2.8092   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## tree (Intercept) 23.200 4.817   
## site (Intercept) 14.466 3.803   
## Residual 1.029 1.014   
## Number of obs: 1431, groups: tree, 100; site, 20  
##   
## Fixed effects:  
## Estimate Std. Error t value  
## (Intercept) 198.915224 7.337100 27.11  
## DD -0.497660 0.004937 -100.81  
## lat -0.009051 0.113477 -0.08  
##   
## Correlation of Fixed Effects:  
## (Intr) DD   
## DD -0.132   
## lat -0.986 0.036

It is correct that lmer() does not provide degrees of freedom, and thus no probability estimate (p-value) for the coefficients. Aside from that it appears that the values estimate from both algorithms match up very well. Based on these results the null probability of observing a t-value at least as extreme for latitude (lat) is 0.9373, so pretty darn likely. That means that this independence claim is supported by the data.

Now to do the same for the other normally distributed dependent variables

# Independence of Growth and lat conditioned on Date  
fit2 <- lme(Growth ~ Date + lat, data = data1,   
 random = ~1|site/tree,   
 na.action = na.omit)  
# Independence of Growth and DD conditioned on Date and lat  
fit3 <- lme(Growth ~ Date + lat + DD, data = data1,   
 random = ~1|site/tree,   
 na.action = na.omit)

The other three independence claims have Live as the dependent variable. This is binary so I need to use the binomial error distribution with the logit link, requiring the lmer() function.

# Independence of Live and lat conditioned on Growth  
fit4 <- glmer(Live ~ scale(Growth) + scale(lat) + (1|site) + (1|tree),  
 data = data1, na.action = na.omit, family = binomial)  
summary(fit4)

## Generalized linear mixed model fit by maximum likelihood (Laplace  
## Approximation) [glmerMod]  
## Family: binomial ( logit )  
## Formula: Live ~ scale(Growth) + scale(lat) + (1 | site) + (1 | tree)  
## Data: data1  
##   
## AIC BIC logLik deviance df.resid   
## 262.0 288.3 -126.0 252.0 1426   
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -12.1722 0.0262 0.0593 0.1195 1.0592   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## tree (Intercept) 0.80933 0.8996   
## site (Intercept) 0.05326 0.2308   
## Number of obs: 1431, groups: tree, 100; site, 20  
##   
## Fixed effects:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 5.5580 0.5889 9.438 < 2e-16 \*\*\*  
## scale(Growth) 2.2742 0.3786 6.006 1.9e-09 \*\*\*  
## scale(lat) 0.2631 0.2559 1.028 0.304   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr) scl(G)  
## scal(Grwth) 0.912   
## scale(lat) 0.311 0.249

This currently throws a warning about a large eigenvalue ratio with suggestion to rescale the variables.

Some of the current estimates are fairly different from those shown in the Shipley supplement, particularly for standard errors, z-values, and p-values. The actual coefficient estimates are much more similar. Nonetheless, the interpretation that the independence claim is supported holds true, with a null probability for observing a z-value associated with lat at least that extreme equaling 0.304.

Now to test the remaining independence claims, Live & DD conditioned on Growth & lat, and Live & Date conditioned on DD & Growth.

# Independence of Live and DD conditioned on Growth and lat  
fit5 <- glmer(Live ~ scale(Growth) + scale(lat) + scale(DD) + (1|site) + (1|tree),  
 data = data1, na.action = na.omit, family = binomial)  
# Indpendence of Live and Date conditioned on DD and Growth  
fit6 <- glmer(Live ~ scale(Growth) + scale(DD) + scale(Date) + (1|site) + (1|tree),  
 data = data1, na.action = na.omit, family = binomial)  
summary(fit6)

## Generalized linear mixed model fit by maximum likelihood (Laplace  
## Approximation) [glmerMod]  
## Family: binomial ( logit )  
## Formula: Live ~ scale(Growth) + scale(DD) + scale(Date) + (1 | site) +   
## (1 | tree)  
## Data: data1  
##   
## AIC BIC logLik deviance df.resid   
## 263.1 294.7 -125.6 251.1 1425   
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -21.2095 0.0339 0.0710 0.1331 1.2657   
##   
## Random effects:  
## Groups Name Variance Std.Dev.   
## tree (Intercept) 4.521e-14 2.126e-07  
## site (Intercept) 0.000e+00 0.000e+00  
## Number of obs: 1431, groups: tree, 100; site, 20  
##   
## Fixed effects:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 5.2426 0.3830 13.687 <2e-16 \*\*\*  
## scale(Growth) 2.1236 0.2523 8.417 <2e-16 \*\*\*  
## scale(DD) -0.2321 0.2466 -0.941 0.347   
## scale(Date) -0.3791 0.2416 -1.569 0.117   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr) scl(G) sc(DD)  
## scal(Grwth) 0.882   
## scale(DD) -0.083 -0.142   
## scale(Date) -0.121 -0.230 0.629

Again got warnings about large eigenvalues, as well as failure of the model to converge, with suggestion to rescale the variables.

Now I want to get the fixed-effects summaries for each model, and in particular the null probability estimates (p-values) for the pair of variables in each independence claim.

# These four lines were to help me figure out how to access the right part of the summary object  
#sum\_fit1 <- summary(fit1)  
#str(sum\_fit1)  
#sum\_fit1$coefficients$fixed# Same as fixed.effects(fit1)  
#sum\_fit1$tTable# Gets the full t-test table including coefficient estimates and probabilities  
summary(fit1)$tTable# lat, Date independence

## Value Std.Error DF t-value p-value  
## (Intercept) 198.915223483 7.337099813 1330 27.11087876 3.185667e-129  
## DD -0.497660383 0.004936809 1330 -100.80608521 0.000000e+00  
## lat -0.009051378 0.113476607 18 -0.07976426 9.373049e-01

summary(fit2)$tTable# Growth, lat independence

## Value Std.Error DF t-value p-value  
## (Intercept) 16.76021995 7.56928260 1330 2.2142415 2.698123e-02  
## Date 0.30315800 0.02676548 1330 11.3264559 1.861527e-28  
## lat -0.09886283 0.11072020 18 -0.8929069 3.836896e-01

summary(fit3)$tTable# Growth, DD indpendence

## Value Std.Error DF t-value p-value  
## (Intercept) 20.64099496 15.12956610 1329 1.3642820 1.727098e-01  
## Date 0.28538608 0.06561009 1329 4.3497288 1.467228e-05  
## lat -0.10048765 0.11142160 18 -0.9018686 3.790366e-01  
## DD -0.01061371 0.03576722 1329 -0.2967440 7.667083e-01

#sum\_fit4 <- summary(fit4)  
#str(sum\_fit4)  
#sum\_fit4$coefficients  
summary(fit4)$coefficients# Independence of Live and lat

## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) 5.5579930 0.5889136 9.437705 3.810332e-21  
## scale(Growth) 2.2741901 0.3786327 6.006323 1.897777e-09  
## scale(lat) 0.2630913 0.2559048 1.028083 3.039110e-01

summary(fit5)$coefficients# Independence of Live and DD

## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) 5.2567803 0.4048942 12.983095 1.525870e-38  
## scale(Growth) 2.0634844 0.2653458 7.776586 7.450791e-15  
## scale(lat) 0.3718368 0.2848878 1.305204 1.918233e-01  
## scale(DD) 0.2794098 0.2779952 1.005088 3.148543e-01

summary(fit6)$coefficients# Independence of Live and Date

## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) 5.2426050 0.3830279 13.6872648 1.209781e-42  
## scale(Growth) 2.1235856 0.2522910 8.4172079 3.855630e-17  
## scale(DD) -0.2321480 0.2466107 -0.9413539 3.465235e-01  
## scale(Date) -0.3790956 0.2415644 -1.5693357 1.165697e-01

Create a vector of the probabilities for the independence claims (6 values).

pvalues <- c(summary(fit6)$coefficients[4,4], summary(fit5)$coefficients[4,4],  
 summary(fit4)$coefficients[3,4], summary(fit3)$tTable[3,5],  
 summary(fit2)$tTable[3,5], summary(fit1)$tTable[3,5])  
print(pvalues)

## [1] 0.1165697 0.3148543 0.3039110 0.3790366 0.3836896 0.9373049

These values do not all match up with the estimates printed in Table 2 of Shipley (2009). Assuming I did the modeling correctly (a substantial assumption), what gives?

Combine the p-values into the *C*-statistic and see what we get anyway.

ln\_pvalues <- log(pvalues)  
C\_stat <- -2\*sum(ln\_pvalues)  
print(C\_stat)

## [1] 12.97744

Checking this value in a $\Chi^2$ look up table against 2*k*(12) degrees of freedom provides an estimated probability of a larger value being somewhere between 0.25 and 0.50.

I would like to learn how to calculate the chi-square cumulative probability based on the degrees of freedom and *C*-statistic, a la the handy tool on this [website](http://stattrek.com/online-calculator/chi-square.aspx).

It's probably simple...

The tool from this website calculates the probability of observing a value smaller than this as 0.63, indicating the probability of observing a larger value is 0.37.