Lab 13 Structural Equation Modeling (SEM) and Path Analysis

The goal of this lab is to apply SEM techniques to determine relationships within a multivariate data set when variables can be both predictor and response variables simultaneously.

# Set up R session

## Download packages

We will be using the following packages:

library(lavaan)  
library(piecewiseSEM)  
library(semPlot)  
library(mvnormtest)  
library(MVN)  
library(dagitty)

## Import data

Today you will be using a data set looking at fire severity and plant richness in Costal California that contains 8 variables measured over 90 sites.

After downloading the piecewiseSEM library, call in the data set keeley and explore it:

keeley  
str(keeley)  
summary(keeley)

To learn more about the data sets:

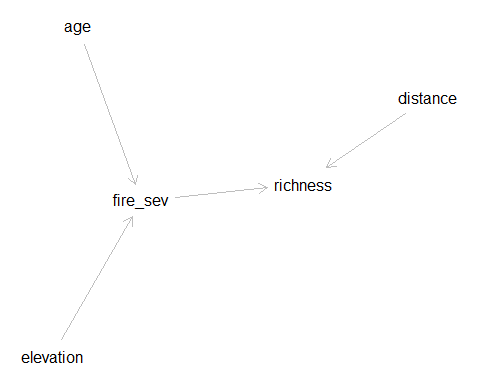
?keeley

# Develop Hypotheses with path diagrams

Path diagrams provide the opportunity to visualize hypotheses to be tested in SEM. Here we use the library dagitty to plot our hypotheses.

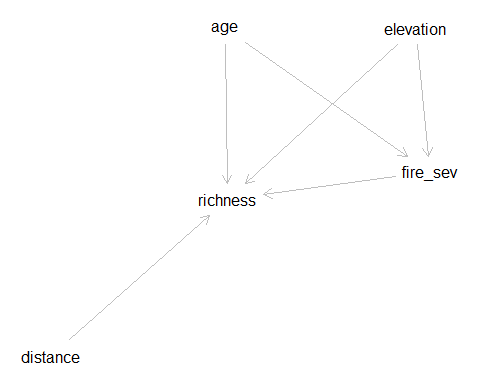
## Indirect Model

#detail all the relationships for the hypothesis/path diagram  
indirect <- dagitty("dag{  
 age -> fire\_sev -> richness  
 elevation -> fire\_sev -> richness  
 distance -> richness  
 }")  
  
#plot the graph  
plot(graphLayout(indirect))



## Direct Model

#detail all the relationships for the hypothesis/path diagram  
direct <- dagitty("dag{  
 age -> fire\_sev -> richness  
 elevation -> fire\_sev -> richness  
 distance -> richness  
 age ->richness  
 elevation ->richness  
 }")  
  
#plot the graph  
plot(graphLayout(direct))



# Model Specification

Our next step is to specify the models for the path diagrams we have developed.

## Indirect Model

Model1<-' firesev ~ age+ elev  
rich ~ firesev + distance'

Run the model using the sem function in lavaan

indirect<-sem(Model1,data=keeley,meanstructure=T)

## Direct Model

Model2<-' firesev ~ age +elev  
rich ~ firesev + age + elev + distance'

Run the model using the sem function in lavaan

direct<-sem(Model2,data=keeley,meanstructure=T)

# Model Assumptions

SEM using a covariance matrix has several assumptions. Arguably, the most important is that the data are multivariate normal. Although we know that tests of multivaraite normality tend to be conservative, we will run one here.

## Testing normality of the observed data

This test is the same for both the direct and indirect model because ther observed data are the same. Univariate normality is also tested

#pull the observed data used in each model  
raw\_data <- inspect(indirect, "data")  
mvn(raw\_data, mvnTest="mardia")

## $multivariateNormality  
## Test Statistic p value Result  
## 1 Mardia Skewness 67.7383677773982 0.000742124645338296 NO  
## 2 Mardia Kurtosis -1.42787779541906 0.153327035801754 YES  
## 3 MVN <NA> <NA> NO  
##   
## $univariateNormality  
## Test Variable Statistic p value Normality  
## 1 Anderson-Darling firesev 0.9288 0.0176 NO   
## 2 Anderson-Darling rich 0.5862 0.1231 YES   
## 3 Anderson-Darling age 0.6426 0.0908 YES   
## 4 Anderson-Darling elev 1.3792 0.0013 NO   
## 5 Anderson-Darling distance 4.0544 <0.001 NO   
##   
## $Descriptives  
## n Mean Std.Dev Median Min Max 25th 75th  
## firesev 90 4.56500 1.652347 4.30000 1.20000 9.200 3.7000 5.55000  
## rich 90 49.23333 15.105658 50.00000 15.00000 85.000 37.0000 62.00000  
## age 90 25.56667 12.566274 25.00000 3.00000 60.000 15.0000 35.00000  
## elev 90 424.66667 258.252337 400.00000 60.00000 1225.000 202.5000 630.00000  
## distance 90 49.23458 8.829480 51.77085 37.03745 60.723 39.4598 58.40224  
## Skew Kurtosis  
## firesev 0.39685828 -0.005405548  
## rich 0.02088941 -0.816859820  
## age 0.50463986 -0.134142371  
## elev 0.63764618 -0.195913417  
## distance -0.11367231 -1.554598328

# Exploring the model parameters and goodness-of-fit

## Indirect Model

indirect<-sem(Model1,data=keeley)  
  
#Summary includes unstandardized coefficents and chi-square test  
sum\_indirect<-summary(indirect)  
  
  
#standardized coefficients  
standardizedSolution(indirect)

## lhs op rhs est.std se z pvalue ci.lower ci.upper  
## 1 firesev ~ age 0.469 0.078 6.047 0.000 0.317 0.621  
## 2 firesev ~ elev -0.161 0.091 -1.768 0.077 -0.340 0.017  
## 3 rich ~ firesev -0.224 0.084 -2.670 0.008 -0.388 -0.060  
## 4 rich ~ distance 0.526 0.069 7.589 0.000 0.390 0.661  
## 5 firesev ~~ firesev 0.768 0.073 10.480 0.000 0.625 0.912  
## 6 rich ~~ rich 0.620 0.073 8.435 0.000 0.476 0.764  
## 7 age ~~ age 1.000 0.000 NA NA 1.000 1.000  
## 8 age ~~ elev 0.093 0.000 NA NA 0.093 0.093  
## 9 age ~~ distance -0.278 0.000 NA NA -0.278 -0.278  
## 10 elev ~~ elev 1.000 0.000 NA NA 1.000 1.000  
## 11 elev ~~ distance 0.606 0.000 NA NA 0.606 0.606  
## 12 distance ~~ distance 1.000 0.000 NA NA 1.000 1.000

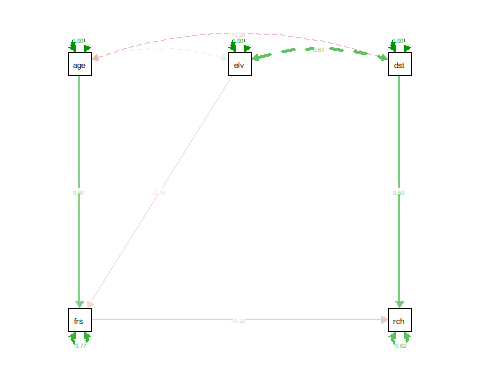
#Rsquared for endogenous variables  
inspect(indirect, "r2")

## firesev rich   
## 0.232 0.380

#Assess Goodness-of-Fit. Look at the lecture slides for "rules of thumb" on goodness-of-fit mesures from Klien et al. 2012.  
fitMeasures(indirect, c("chisq", "df", "pvalue", "cfi", "rmsea","srmr"))

## chisq df pvalue cfi rmsea srmr   
## 1.551 3.000 0.670 1.000 0.000 0.025

#plot path diagram with standardized coefficients  
  
semPaths(indirect,"std")



## Direct Model

direct<-sem(Model2,data=keeley)  
  
#Summary includes unstandardized coefficents and chi-square test  
sum\_indirect<-summary(direct)  
  
  
#standardized coefficients  
standardizedSolution(direct)

## lhs op rhs est.std se z pvalue ci.lower ci.upper  
## 1 firesev ~ age 0.469 0.078 6.047 0.000 0.317 0.621  
## 2 firesev ~ elev -0.161 0.091 -1.768 0.077 -0.340 0.017  
## 3 rich ~ firesev -0.199 0.093 -2.137 0.033 -0.382 -0.017  
## 4 rich ~ age -0.051 0.101 -0.502 0.616 -0.250 0.148  
## 5 rich ~ elev -0.070 0.111 -0.627 0.530 -0.288 0.148  
## 6 rich ~ distance 0.560 0.102 5.517 0.000 0.361 0.760  
## 7 firesev ~~ firesev 0.768 0.073 10.480 0.000 0.625 0.912  
## 8 rich ~~ rich 0.613 0.073 8.403 0.000 0.470 0.756  
## 9 age ~~ age 1.000 0.000 NA NA 1.000 1.000  
## 10 age ~~ elev 0.093 0.000 NA NA 0.093 0.093  
## 11 age ~~ distance -0.278 0.000 NA NA -0.278 -0.278  
## 12 elev ~~ elev 1.000 0.000 NA NA 1.000 1.000  
## 13 elev ~~ distance 0.606 0.000 NA NA 0.606 0.606  
## 14 distance ~~ distance 1.000 0.000 NA NA 1.000 1.000

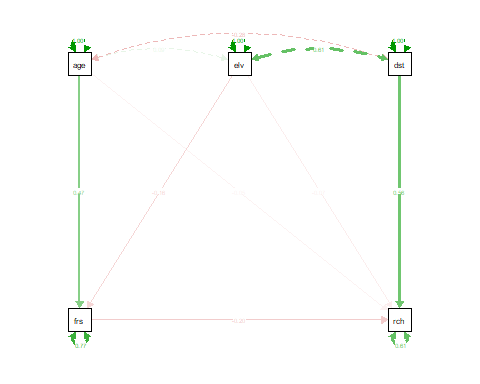
#Rsquared for endogenous variables  
inspect(direct, "r2")

## firesev rich   
## 0.232 0.387

#Assess Goodness-of-Fit. Look at the lecture slides for "rules of thumb" on goodness-of-fit mesures from Klien et al. 2012.  
fitMeasures(direct, c("chisq", "df", "pvalue", "cfi", "rmsea","srmr"))

## chisq df pvalue cfi rmsea srmr   
## 0.562 1.000 0.454 1.000 0.000 0.015

#plot path diagram with standardized coefficients  
  
semPaths(direct,"std")



# Comparing Models

The ability to compare different hypotheses depicted by path diagrams is a strenght of SEM models. Here we will use AIC, BIC, and a likelihood ratio test to compare our direct and indirect hypotheses.

#Likelihood ratio test using the `anova` function  
anova(indirect, direct)

## Chi-Squared Difference Test  
##   
## Df AIC BIC Chisq Chisq diff Df diff Pr(>Chisq)  
## direct 1 1035.1 1055.1 0.5618   
## indirect 3 1032.1 1047.1 1.5514 0.98966 2 0.6097

#Comparing mopdels with AIC and BIC  
AIC(direct, indirect)

## df AIC  
## direct 8 1035.106  
## indirect 6 1032.096

BIC(direct, indirect)

## df BIC  
## direct 8 1055.105  
## indirect 6 1047.095

# Estimating and testing indirect effects

We can walk through this together

## Indirect model

Model1<-'   
rich ~ b \* firesev + d \* distance  
firesev ~ a1 \* age + a2 \* elev  
indirect1:=a1\*b  
indirect2:=a2\*b  
total:=d+(a1\*b)+(a2\*b)'  
  
indirect<-sem(Model1,data=keeley)  
standardizedsolution(indirect)

## lhs op rhs label est.std se z pvalue ci.lower  
## 1 rich ~ firesev b -0.224 0.084 -2.670 0.008 -0.388  
## 2 rich ~ distance d 0.526 0.069 7.589 0.000 0.390  
## 3 firesev ~ age a1 0.469 0.078 6.047 0.000 0.317  
## 4 firesev ~ elev a2 -0.161 0.091 -1.768 0.077 -0.340  
## 5 rich ~~ rich 0.620 0.073 8.435 0.000 0.476  
## 6 firesev ~~ firesev 0.768 0.073 10.480 0.000 0.625  
## 7 distance ~~ distance 1.000 0.000 NA NA 1.000  
## 8 distance ~~ age -0.278 0.000 NA NA -0.278  
## 9 distance ~~ elev 0.606 0.000 NA NA 0.606  
## 10 age ~~ age 1.000 0.000 NA NA 1.000  
## 11 age ~~ elev 0.093 0.000 NA NA 0.093  
## 12 elev ~~ elev 1.000 0.000 NA NA 1.000  
## 13 indirect1 := a1\*b indirect1 -0.105 0.044 -2.405 0.016 -0.191  
## 14 indirect2 := a2\*b indirect2 0.036 0.024 1.476 0.140 -0.012  
## 15 total := d+(a1\*b)+(a2\*b) total 0.457 0.089 5.153 0.000 0.283  
## ci.upper  
## 1 -0.060  
## 2 0.661  
## 3 0.621  
## 4 0.017  
## 5 0.764  
## 6 0.912  
## 7 1.000  
## 8 -0.278  
## 9 0.606  
## 10 1.000  
## 11 0.093  
## 12 1.000  
## 13 -0.019  
## 14 0.084  
## 15 0.631

## Direct model

Model2<-'   
rich ~ b \* firesev + d \* distance + c\*age + e\*elev  
firesev ~ a1 \* age + a2 \* elev  
indirect1:=a1\*b  
indirect2:=a2\*b  
total:=d+c+e+(a1\*b)+(a2\*b)'  
  
direct<-sem(Model2,data=keeley)  
standardizedsolution(indirect)

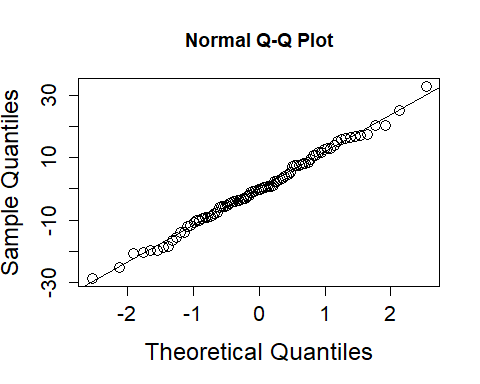
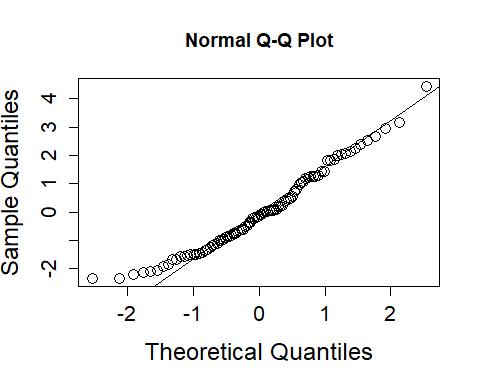
## lhs op rhs label est.std se z pvalue ci.lower  
## 1 rich ~ firesev b -0.224 0.084 -2.670 0.008 -0.388  
## 2 rich ~ distance d 0.526 0.069 7.589 0.000 0.390  
## 3 firesev ~ age a1 0.469 0.078 6.047 0.000 0.317  
## 4 firesev ~ elev a2 -0.161 0.091 -1.768 0.077 -0.340  
## 5 rich ~~ rich 0.620 0.073 8.435 0.000 0.476  
## 6 firesev ~~ firesev 0.768 0.073 10.480 0.000 0.625  
## 7 distance ~~ distance 1.000 0.000 NA NA 1.000  
## 8 distance ~~ age -0.278 0.000 NA NA -0.278  
## 9 distance ~~ elev 0.606 0.000 NA NA 0.606  
## 10 age ~~ age 1.000 0.000 NA NA 1.000  
## 11 age ~~ elev 0.093 0.000 NA NA 0.093  
## 12 elev ~~ elev 1.000 0.000 NA NA 1.000  
## 13 indirect1 := a1\*b indirect1 -0.105 0.044 -2.405 0.016 -0.191  
## 14 indirect2 := a2\*b indirect2 0.036 0.024 1.476 0.140 -0.012  
## 15 total := d+(a1\*b)+(a2\*b) total 0.457 0.089 5.153 0.000 0.283  
## ci.upper  
## 1 -0.060  
## 2 0.661  
## 3 0.621  
## 4 0.017  
## 5 0.764  
## 6 0.912  
## 7 1.000  
## 8 -0.278  
## 9 0.606  
## 10 1.000  
## 11 0.093  
## 12 1.000  
## 13 -0.019  
## 14 0.084  
## 15 0.631

# piecewiseSEM

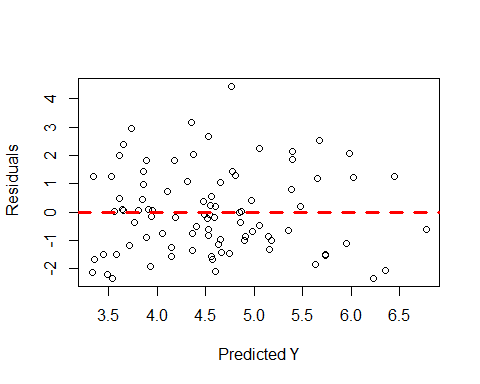
Let’s runs the same models using local estimation in the package ‘piecewiseSEM’

## Indirect model

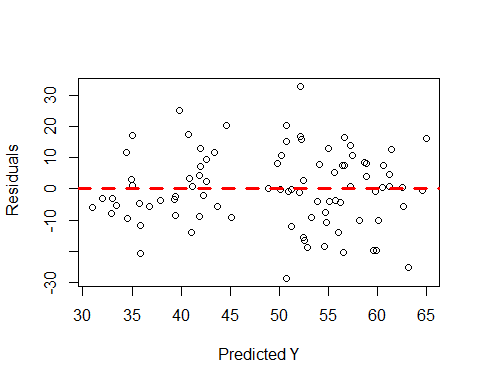
#model "pieces"  
  
fire<-lm(firesev ~ age+ elev,data=keeley)  
richness<-lm(rich ~ firesev + distance,data=keeley)  
  
#check model assumptions  
fire\_res<-residuals(fire)  
richness\_res<-residuals(richness)  
  
res\_indirect<-cbind(fire\_res,richness\_res)  
  
#qqplots  
apply(res\_indirect[,1:2], 2, function(x){  
 qqnorm(x, cex=1.5, cex.lab=1.5, cex.axis=1.3)  
 qqline(x)  
})



#Plotting residuals vs. predicted values to assess homogeneity of variance and linearity  
  
#fire  
predicted\_fire<-predict.lm(fire)  
  
#plot residuals vs. predicted values  
plot(predicted\_fire,fire\_res, ylab=" Residuals", xlab= "Predicted Y")  
abline(a=0,b=0, col="red", lwd=3,lty="dashed")



#Richness  
predicted\_richness<-predict.lm(richness)  
  
#plot residuals vs. predicted values  
plot(predicted\_richness,richness\_res, ylab=" Residuals", xlab= "Predicted Y")  
abline(a=0,b=0, col="red", lwd=3,lty="dashed")



#"cobble" the model together using the "psem" function   
model\_indirect <- psem(fire,richness)  
  
#use the summary function to evaluate model  
summary(model\_indirect)

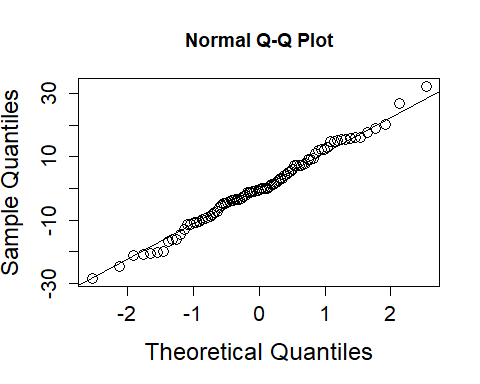
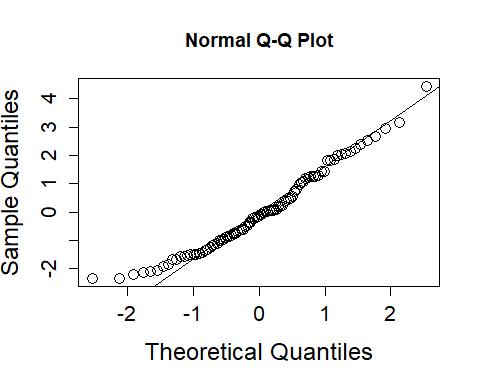
## Structural Equation Model of model\_indirect   
##   
## Call:  
## firesev ~ age + elev  
## rich ~ firesev + distance  
##   
## AIC BIC  
## 20.928 40.926  
##   
## ---  
## Tests of directed separation:  
##   
## Independ.Claim Test.Type DF Crit.Value P.Value   
## rich ~ age + ... coef 86 -0.7546 0.4526   
## rich ~ elev + ... coef 86 -0.8382 0.4042   
## firesev ~ distance + ... coef 86 -0.7338 0.4651   
##   
## Global goodness-of-fit:  
##   
## Fisher's C = 4.928 with P-value = 0.553 and on 6 degrees of freedom  
##   
## ---  
## Coefficients:  
##   
## Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate   
## firesev age 0.0617 0.0124 87 4.9681 0.0000 0.4689 \*\*\*  
## firesev elev -0.0010 0.0006 87 -1.7071 0.0914 -0.1611   
## rich firesev -2.0348 0.7986 87 -2.5479 0.0126 -0.2226 \*  
## rich distance 0.8941 0.1495 87 5.9825 0.0000 0.5226 \*\*\*  
##   
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05  
##   
## ---  
## Individual R-squared:  
##   
## Response method R.squared  
## firesev none 0.23  
## rich none 0.39

## Goodness of fit

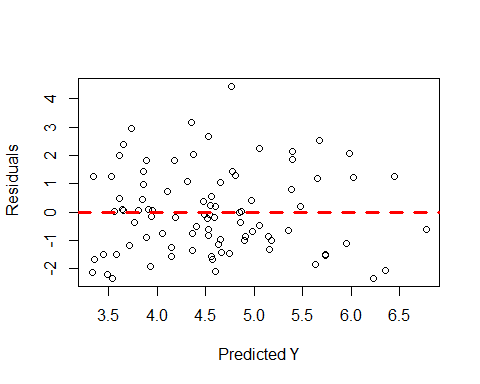
The goodness-of-fit of a piecewise structural equation model is obtained using ‘tests of directed seperation’. The P-values of these tests are then combined in a single Fisher’s C statistic which is χ2-distributed with 2k degrees of freedom. As with other goodness-of-fit tests, we are looking for p-values >0.05.

## Direct model

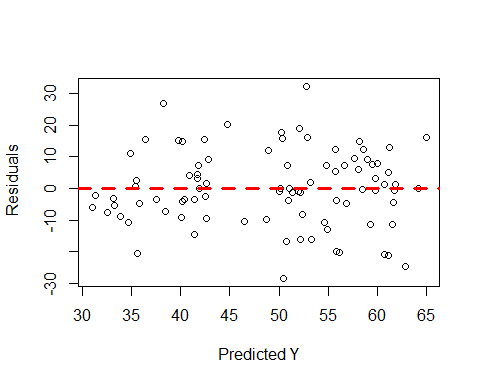
#model "pieces"  
  
fire2<-lm(firesev ~ age+ elev,data=keeley)  
richness2<-lm(rich ~ firesev + distance+age+elev,data=keeley)  
  
#check model assumptions  
fire\_res2<-residuals(fire2)  
richness\_res2<-residuals(richness2)  
  
res\_indirect2<-cbind(fire\_res2,richness\_res2)  
  
#qqplots  
apply(res\_indirect2[,1:2], 2, function(x){  
 qqnorm(x, cex=1.5, cex.lab=1.5, cex.axis=1.3)  
 qqline(x)  
})



#residuals vs. predicted values  
  
#fire  
predicted\_fire2<-predict.lm(fire2)  
  
## plot residuals vs. the predicted values:  
plot(predicted\_fire2,fire\_res2, ylab=" Residuals", xlab= "Predicted Y")  
abline(a=0,b=0, col="red", lwd=3,lty="dashed")



#Richness  
predicted\_richness2<-predict.lm(richness2)  
## plot residuals vs. the predicted values:  
plot(predicted\_richness2,richness\_res2, ylab=" Residuals", xlab= "Predicted Y")  
abline(a=0,b=0, col="red", lwd=3,lty="dashed")



#cobble the model together using the "psem" function   
model\_direct <- psem(fire2,richness2)  
  
#use the summary function to evaluate model  
summary(model\_direct)

##   
## Structural Equation Model of model\_direct   
##   
## Call:  
## firesev ~ age + elev  
## rich ~ firesev + distance + age + elev  
##   
## AIC BIC  
## 21.531 46.529  
##   
## ---  
## Tests of directed separation:  
##   
## Independ.Claim Test.Type DF Crit.Value P.Value   
## firesev ~ distance + ... coef 86 -0.7338 0.4651   
##   
## Global goodness-of-fit:  
##   
## Fisher's C = 1.531 with P-value = 0.465 and on 2 degrees of freedom  
##   
## ---  
## Coefficients:  
##   
## Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate   
## firesev age 0.0617 0.0124 87 4.9681 0.0000 0.4689 \*\*\*  
## firesev elev -0.0010 0.0006 87 -1.7071 0.0914 -0.1611   
## rich firesev -1.8108 0.8834 85 -2.0498 0.0435 -0.1981 \*  
## rich distance 0.9536 0.2010 85 4.7453 0.0000 0.5574 \*\*\*  
## rich age -0.0609 0.1234 85 -0.4933 0.6230 -0.0506   
## rich elev -0.0041 0.0066 85 -0.6120 0.5422 -0.0694   
##   
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05  
##   
## ---  
## Individual R-squared:  
##   
## Response method R.squared  
## firesev none 0.23  
## rich none 0.39

## Compare models using AIC

AIC(model\_direct,model\_indirect)

## df AIC  
## x 10 21.531  
## y 8 20.928

Not a huge difference between the models. I would go with the more parsimonious model (i.e., indirect)