Causal Method for Mediation Analysis for Count Data (Poisson Regression)

Disclaimer:

This pdf is part of a YouTube tutorial (https://youtu.be/l3igREAFRXE).

1. Please make sure to read the published papers mentioned in the video description. Since I am not the author of the papers, I cannot guarantee that the R code and information in the tutorial correctly reflect the idea of those papers. Please read the papers by yourself and make your own judgment.
2. This video tutorial and this pdf are not peer-reviewed and the correctness and quality of the information and R code presented in this tutorial are not guaranteed. Thus, please do NOT cite the video tutorial or this pdf as a reference, if you are writing an academic paper.
3. This pdf is for your personal usage only. Please do NOT distribute this pdf.

Generate Sample Data

```
# set the size of the sample
n=500
# set seed
set.seed(123)

# simulate x (normal distribution)
X <- rep(0:1, each=n/2)
#print(X)


# simulate x (normal distribution)
#X <- rnorm(n, 5, 4)
# calculate the mean of X
#mean_x=mean(X)

# simulate a residual for M
residual_1<-rnorm(n,0,1)
M<-0.3+0.5*X+residual_1

# mu for Poisson regression via a log link
mu_1 <- exp(0.2 + 0.2*M+0.08*X)
# use rpois to generate Y
Y <- rpois(n, lambda=mu_1)

# combine into a dataframe and print out the first 6 rows
```

```r
data <- data.frame(X=X, M=M, Y=Y)
head(data)
```

```
##   X           M Y
## 1 0 -0.26047565 0
## 2 0  0.06982251 1
## 3 0  1.85870831 0
## 4 0  0.37050839 2
## 5 0  0.42928774 2
## 6 0  2.01506499 2
```

a and b paths

```
a_path <- lm(M ~ X, data = data)
summary(a_path)
```

```
##
## Call:
## lm(formula = M ~ X, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7387 -0.6172 -0.0324  0.6350  3.2496
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.29144    0.06152   4.737 2.83e-06 ***
## X            0.58630    0.08701   6.738 4.44e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9728 on 498 degrees of freedom
## Multiple R-squared:  0.08356,    Adjusted R-squared:  0.08172
## F-statistic: 45.41 on 1 and 498 DF,  p-value: 4.437e-11
```

```
b_path <- glm(Y ~ X + M, data = data, family=poisson(link = "log"))
summary(b_path)
```

```
##
## Call:
## glm(formula = Y ~ X + M, family = poisson(link = "log"), data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1113  -0.8894  -0.1481   0.5511   3.3210
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.13812    0.05956   2.319   0.0204 *
## X            0.11475    0.07846   1.463   0.1436
## M            0.23370    0.03805   6.142 8.13e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 598.43  on 499  degrees of freedom
## Residual deviance: 549.29  on 497  degrees of freedom
## AIC: 1486.6
##
## Number of Fisher Scoring iterations: 5
```

Use mediate()

```
library(mediation)
```

```
## Warning: package 'mediation' was built under R version 4.1.3
```

```
## Loading required package: MASS
```

```
## Loading required package: Matrix
```

```
## Loading required package: mvtnorm
```

```
## Loading required package: sandwich
```

```
## Warning: package 'sandwich' was built under R version 4.1.2
```

```
## mediation: Causal Mediation Analysis
## Version: 4.5.0
```

```
# set seed
#set.seed(123)
# Estimate the indirect effect using the mediate function
result <- mediate(a_path, b_path,sims = 1000, boot = TRUE, treat = "X", mediator = "M")
```

```
## Running nonparametric bootstrap
```

```
# Summarize the results
summary(result)
```

```
##
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
##
##                          Estimate 95% CI Lower 95% CI Upper p-value
## ACME (control)             0.1845       0.1150         0.27  <2e-16 ***
## ACME (treated)             0.2069       0.1311         0.30  <2e-16 ***
## ADE (control)              0.1527      -0.0456         0.37   0.126
## ADE (treated)              0.1752      -0.0525         0.42   0.126
## Total Effect               0.3596       0.1396         0.59   0.002 **
## Prop. Mediated (control)   0.5129       0.2425         1.37   0.002 **
## Prop. Mediated (treated)   0.5753       0.3146         1.33   0.002 **
## ACME (average)             0.1957       0.1234         0.28  <2e-16 ***
## ADE (average)              0.1639      -0.0494         0.39   0.126
## Prop. Mediated (average)   0.5441       0.2800         1.36   0.002 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 500
##
##
## Simulations: 1000
```

Double Check

```
Mediation_function_poisson_Checking<-function(data_used,i)
{
  # sample a data
  data_temp=data_used[i,]

  # a path
a_path <- lm(M ~ X, data = data_temp)
a_0<-a_path$coefficients[1]
a_1<-a_path$coefficients[2]

  # b path
b_path <- glm(Y ~ X + M, data = data_temp, family=poisson(link = "log"))
b_0<-b_path$coefficients[1]
b_1<-b_path$coefficients[3]
c_1_apostrophe<-b_path$coefficients[2]

 # M when t=0 (control condition)
x_predetermined=0
M_estimated_0=a_0+a_1*x_predetermined

 # M when t=1 (treatment condition)
x_predetermined=1
M_estimated_1=a_0+a_1*x_predetermined

# indirect effect: t=0 (control condition)
x_predetermined=0
Indirect_Effect_0=exp(b_0+b_1*M_estimated_1+c_1_apostrophe*x_predetermined)-
  exp(b_0+b_1*M_estimated_0+c_1_apostrophe*x_predetermined)


# indirect effect: t=1 (treatment condition)
x_predetermined=1
Indirect_Effect_1=exp(b_0+b_1*M_estimated_1+c_1_apostrophe*x_predetermined)-
  exp(b_0+b_1*M_estimated_0+c_1_apostrophe*x_predetermined)

return(c(Indirect_Effect_0,Indirect_Effect_1))

}
```

Apply Checking Function

```
library(boot)
```

```
## Warning: package 'boot' was built under R version 4.1.3
```

```
#set.seed(23)
Indirect_Effect <- boot(data=data,Mediation_function_poisson_Checking,R=1000)
E_Indirect_Effect_0 <- Indirect_Effect$t[,1]
E_Indirect_Effect_1 <- Indirect_Effect$t[,2]
IE_0<-mean(E_Indirect_Effect_0)
IE_1<-mean(E_Indirect_Effect_1)

print(IE_0)
```

```
## [1] 0.1791486
```

```
print(IE_1)
```

```
## [1] 0.2008047
```

```
print((IE_0+IE_1)/2)
```

```
## [1] 0.1899766
```