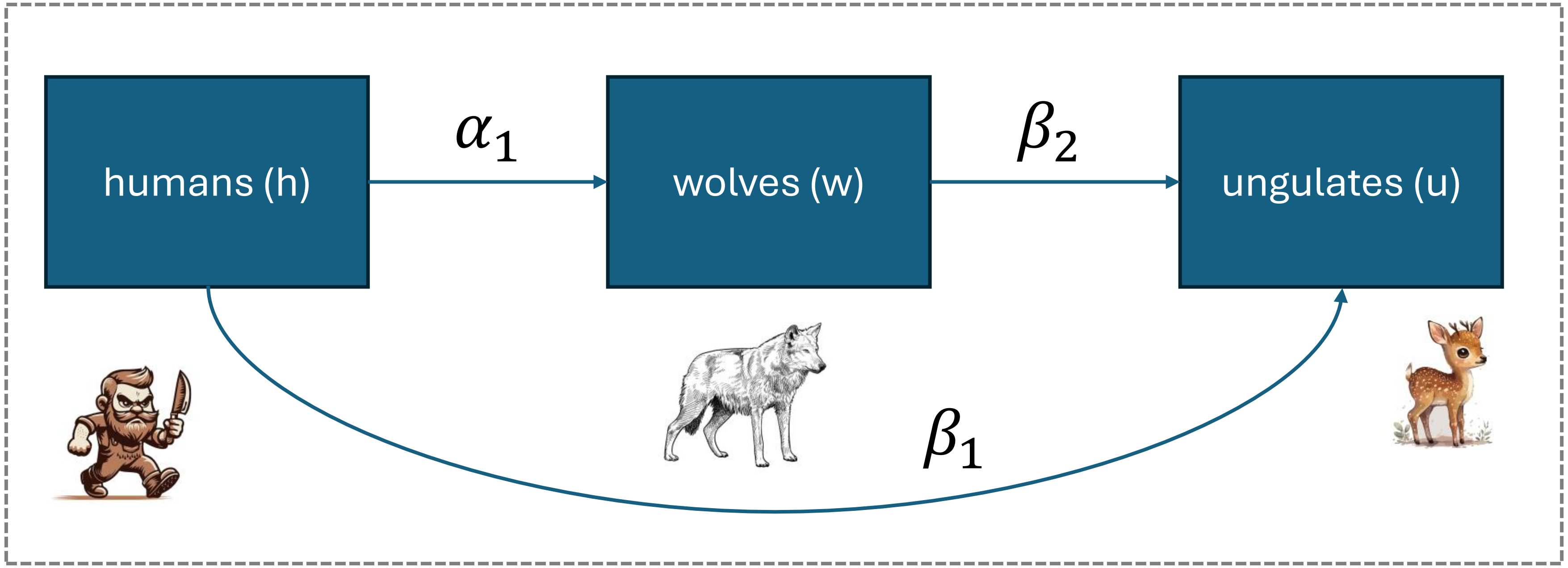
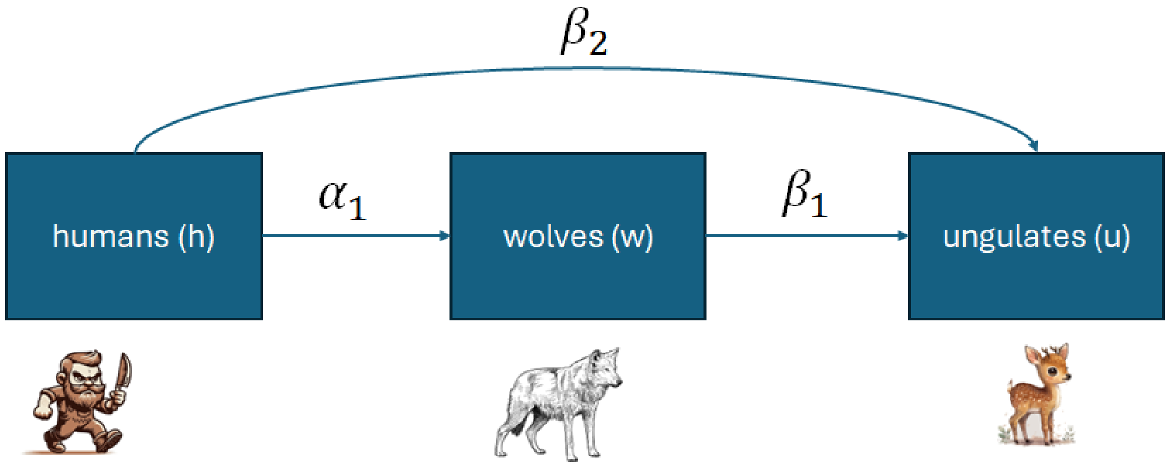
Direct and indirect effects (WILD 595)



Let’s simulate some data (n = 100) and talk about it.

We’ll simulate human (h), wolf (w), and ungulate (u) abundance along a ‘wild-to-suburban’ gradient



β_2

humans (h)

α_1

wolves (w)

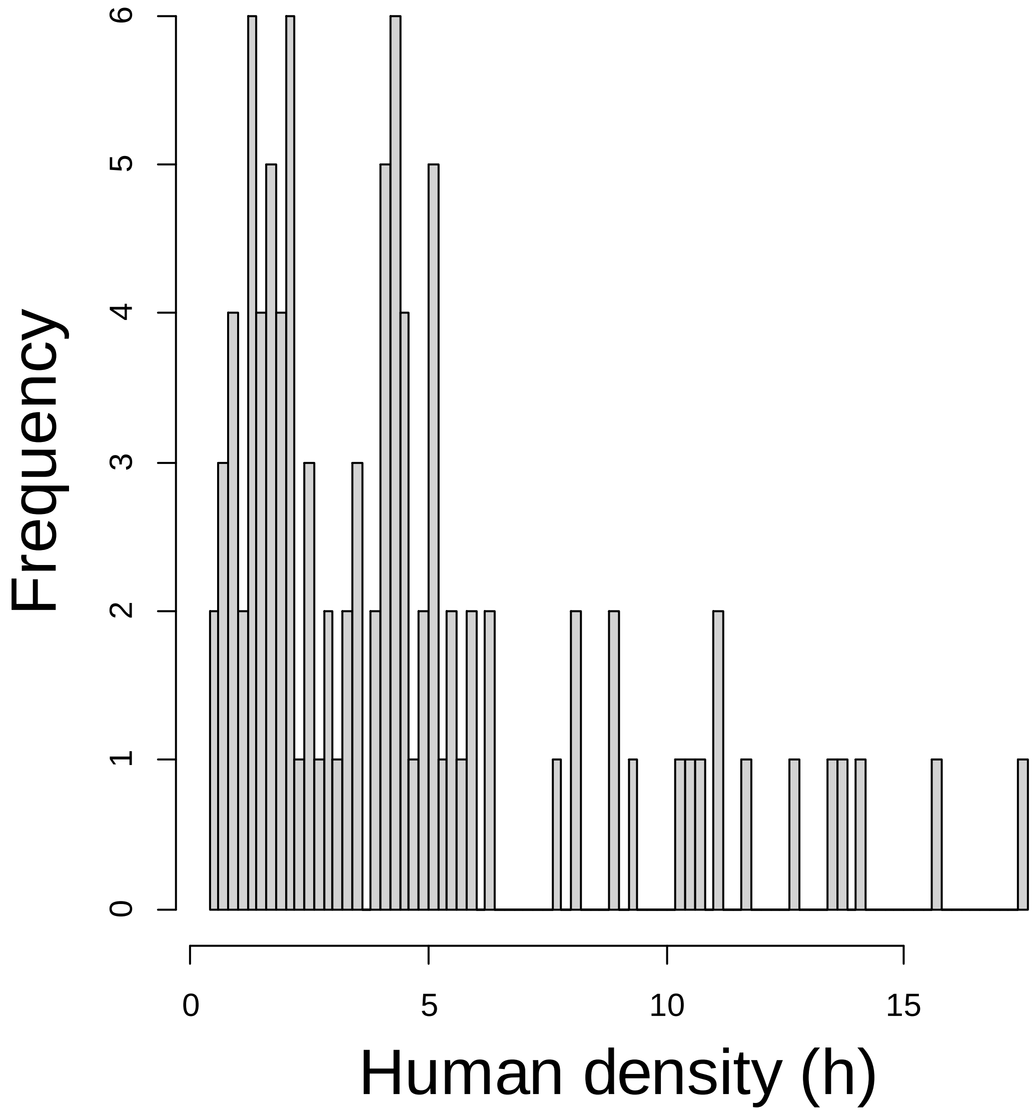
β_1

ungulates (u)

Simulating the data: humans

$$\boldsymbol{h} \sim \text{lognormal}(1, 0.75)$$

humans (h)



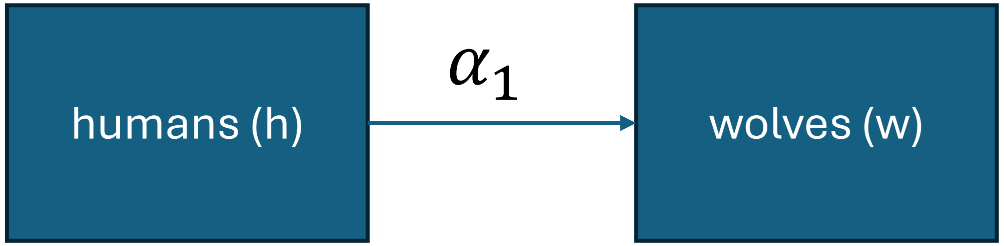
We’ll assume these sites have similar habitat types/qualities

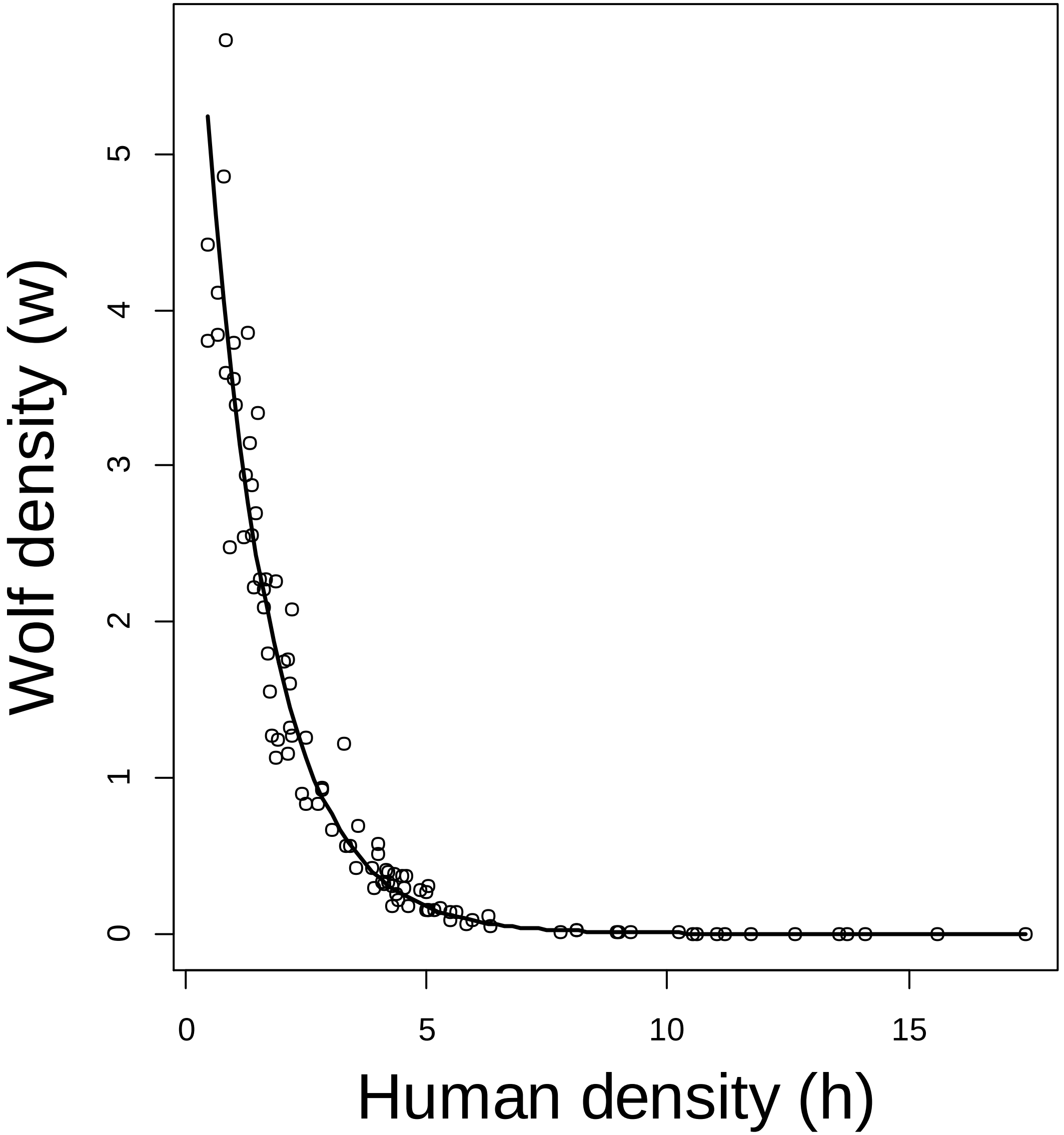
Simulating the data: wolves

$$\boldsymbol{w} \sim \text{lognormal}(\alpha_0 + \alpha_1 \boldsymbol{h}, \sigma_w^2)$$

$$\alpha_0 = 3$$

$$\alpha_1 = -0.75$$





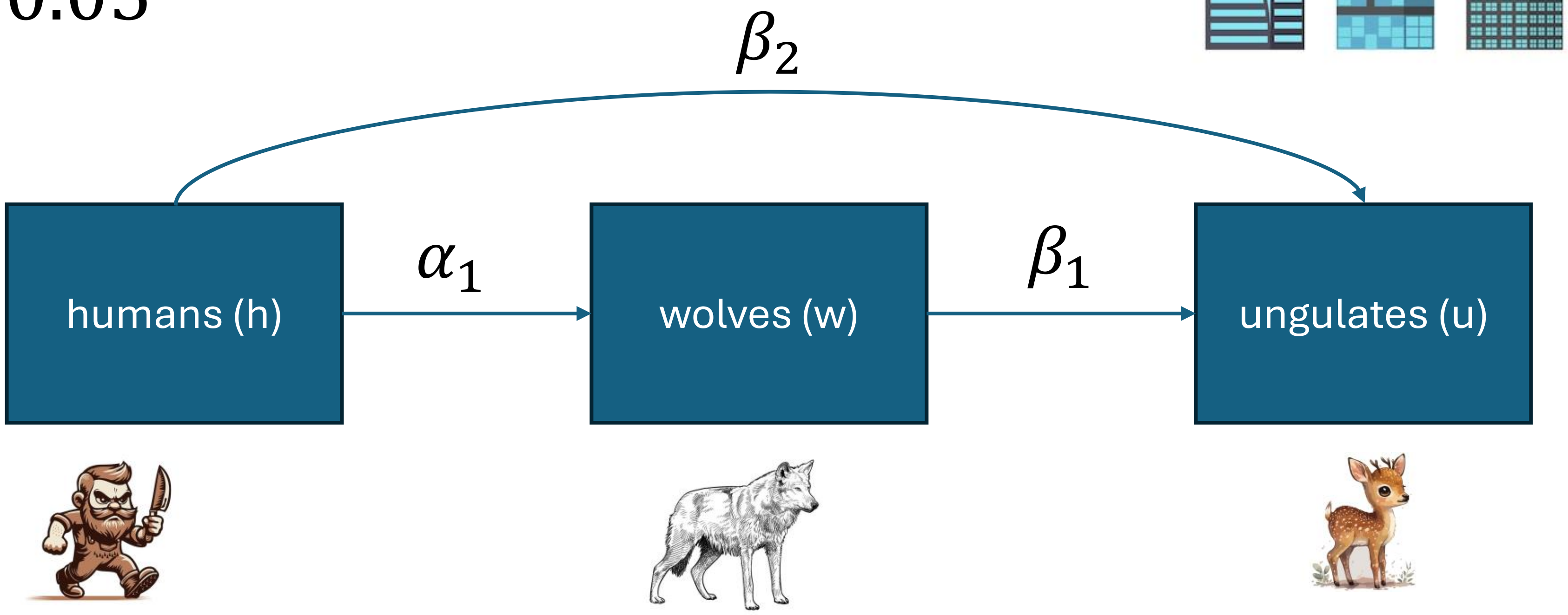
Simulating the data: ungulates

$$\boldsymbol{u} \sim \mathrm{lognormal}(\beta_0 + \beta_1 \boldsymbol{w} + \beta_2 \boldsymbol{h}, \sigma_u^2)$$

$$\beta_0 = 3$$

$$\beta_1 = -0.15$$

$$\beta_2 = -0.05$$



β_2

humans (h)

α_1

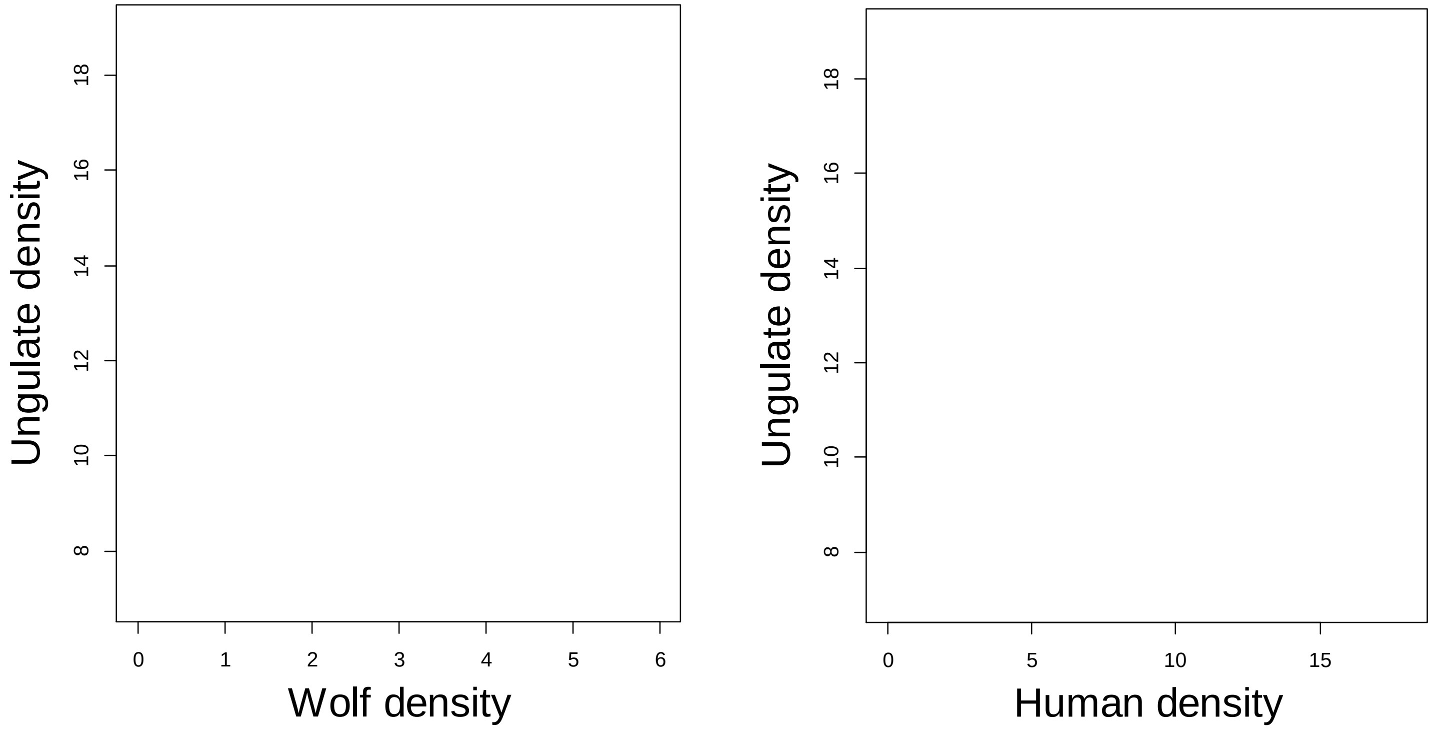
wolves (w)

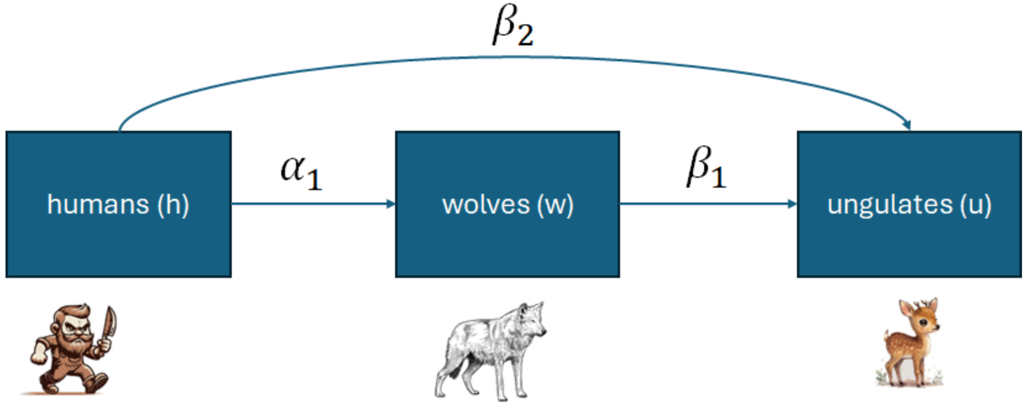
β_1

ungulates (u)

Now let’s look at our observed relationships

What do we expect to see?

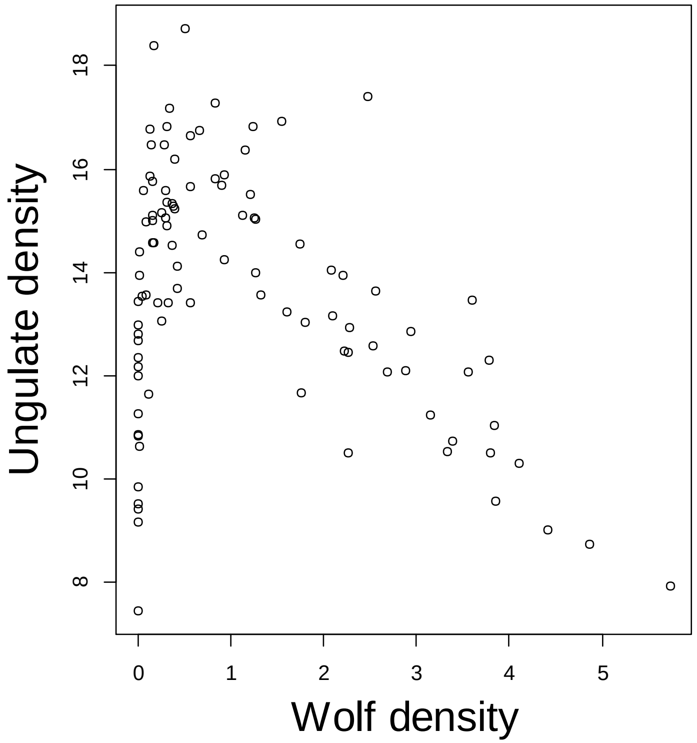


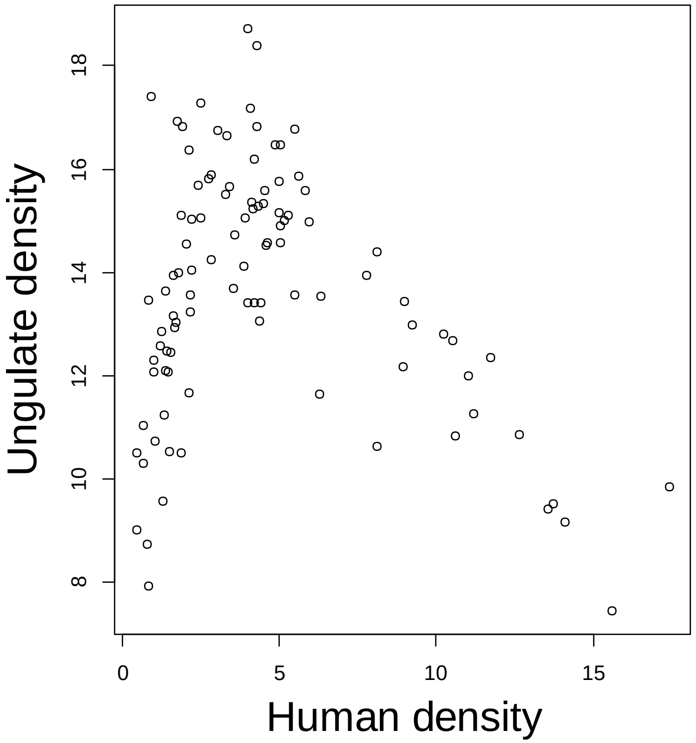


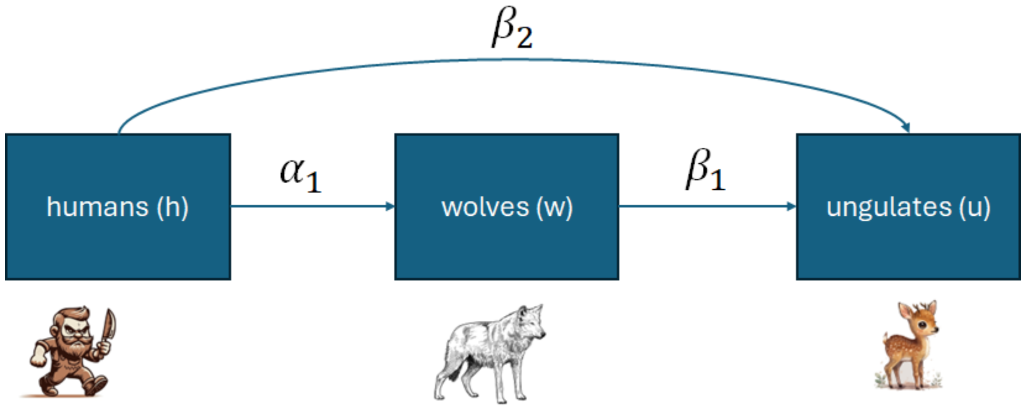
$$\beta_1 = -0.15$$

$$\beta_2 = -0.05$$

What’s happening here?!



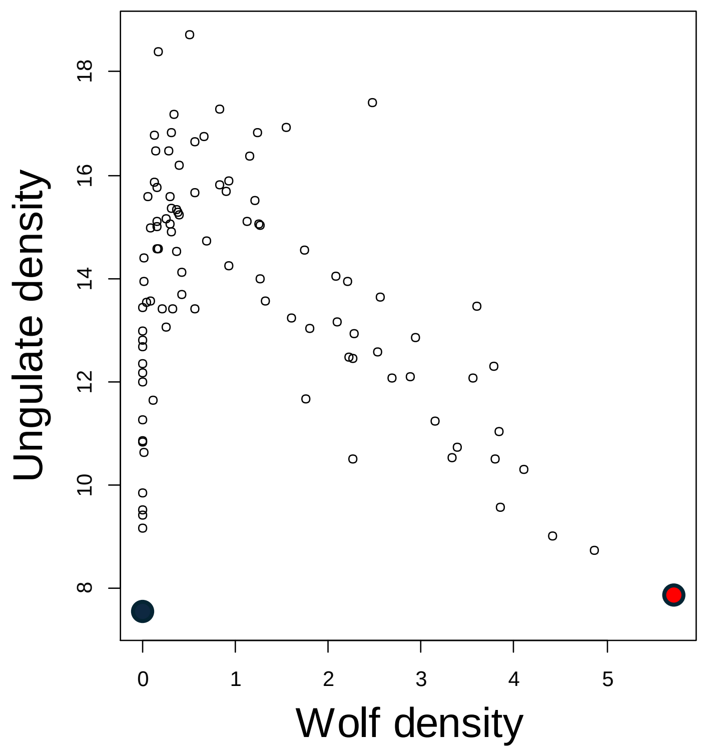


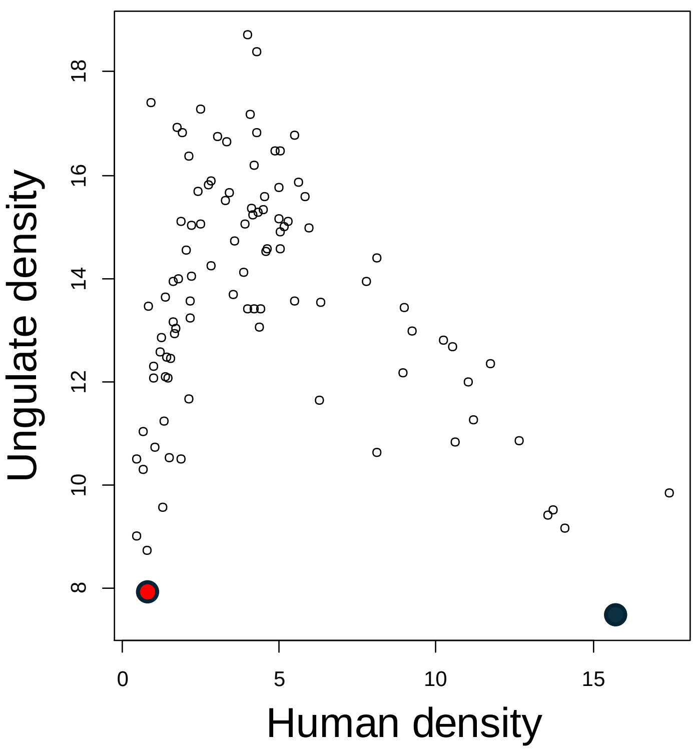


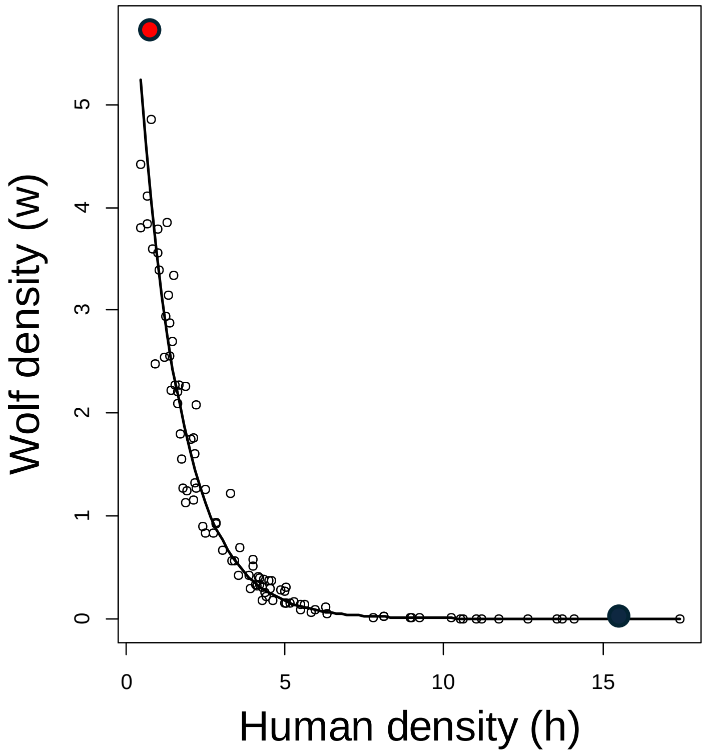
β_2

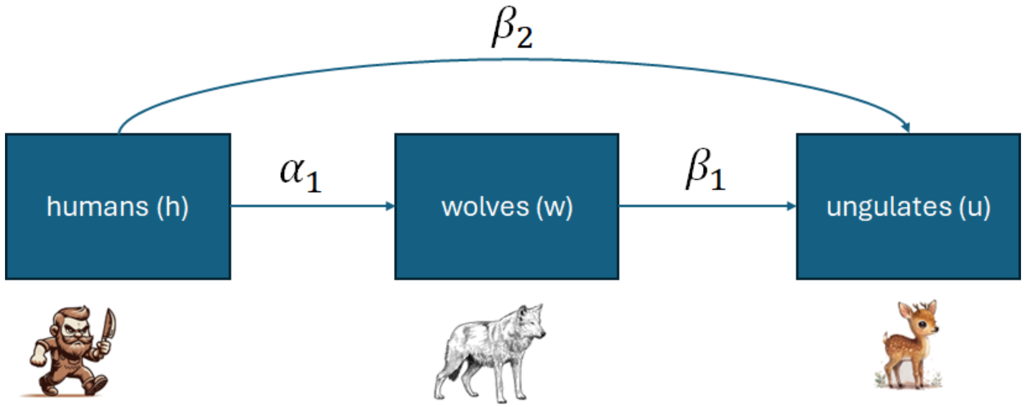
humans (h) $\xrightarrow{\alpha_1}$ wolves (w) $\xrightarrow{\beta_1}$ ungulates (u)

What’s happening here?!





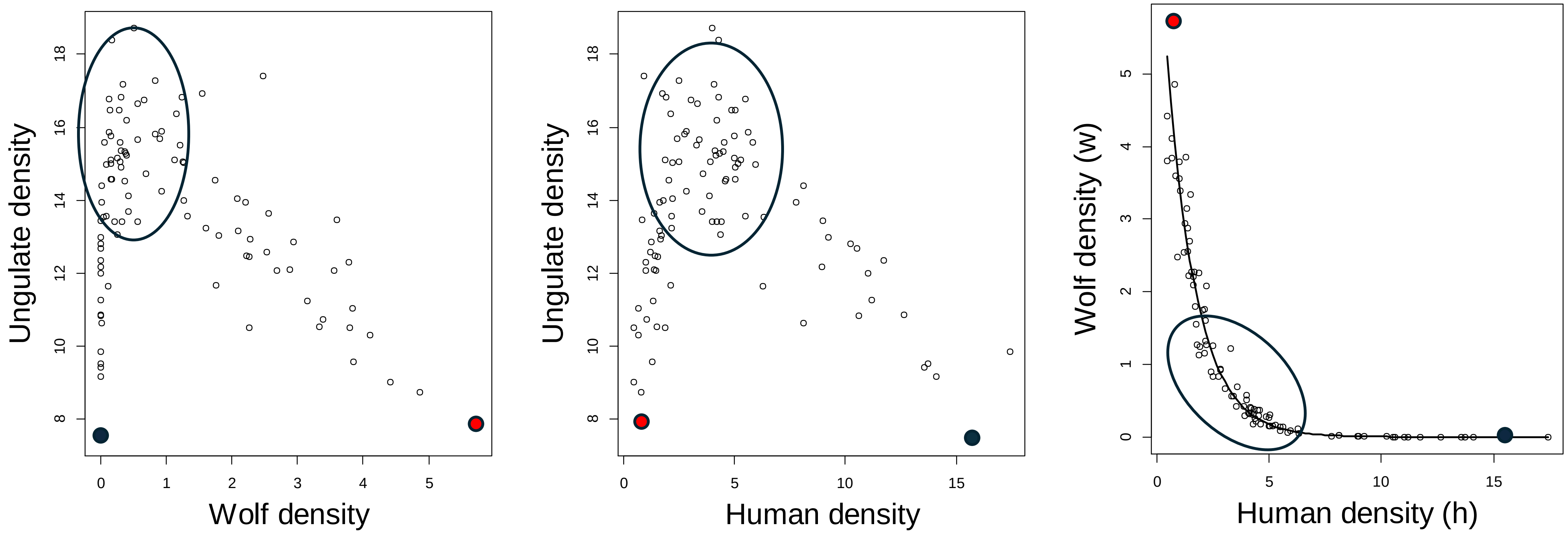


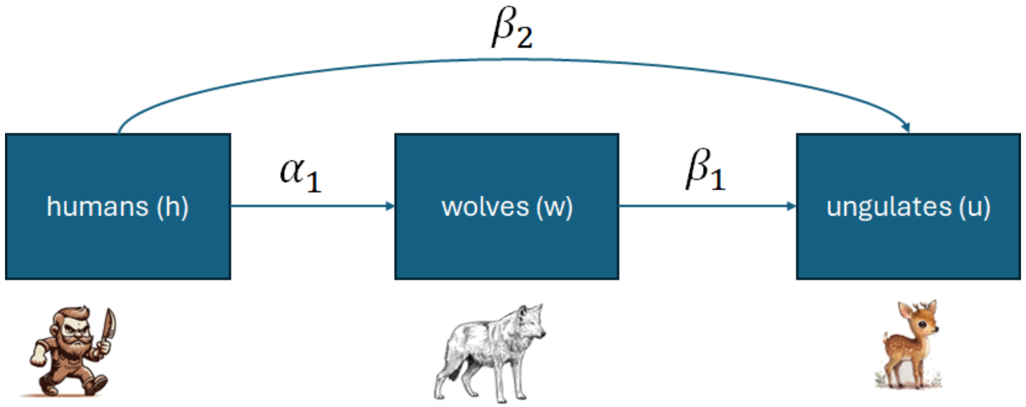


β_2

humans (h) $\xrightarrow{\alpha_1}$ wolves (w) $\xrightarrow{\beta_1}$ ungulates (u)

What’s happening here?!



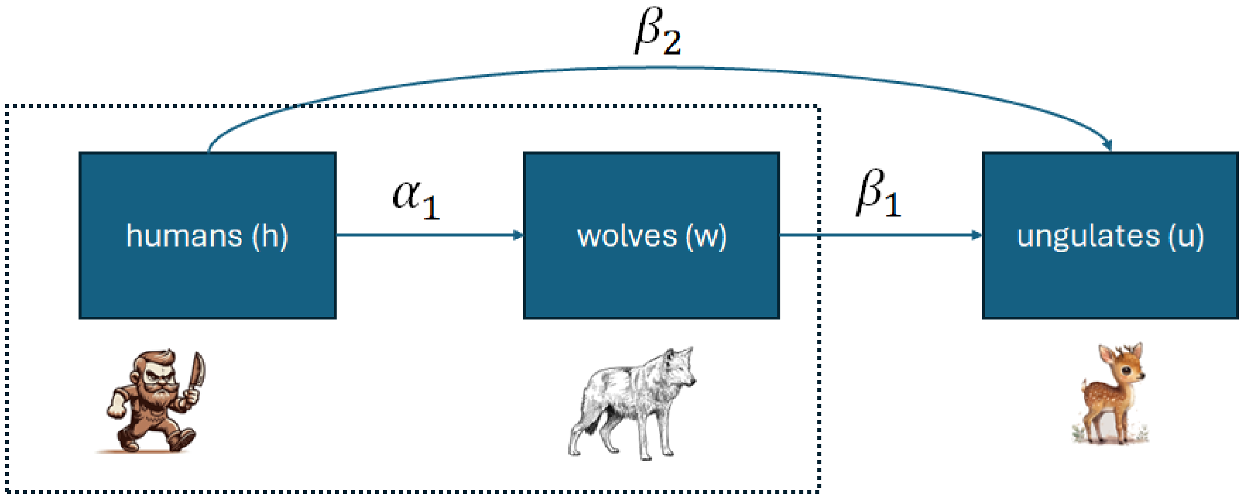


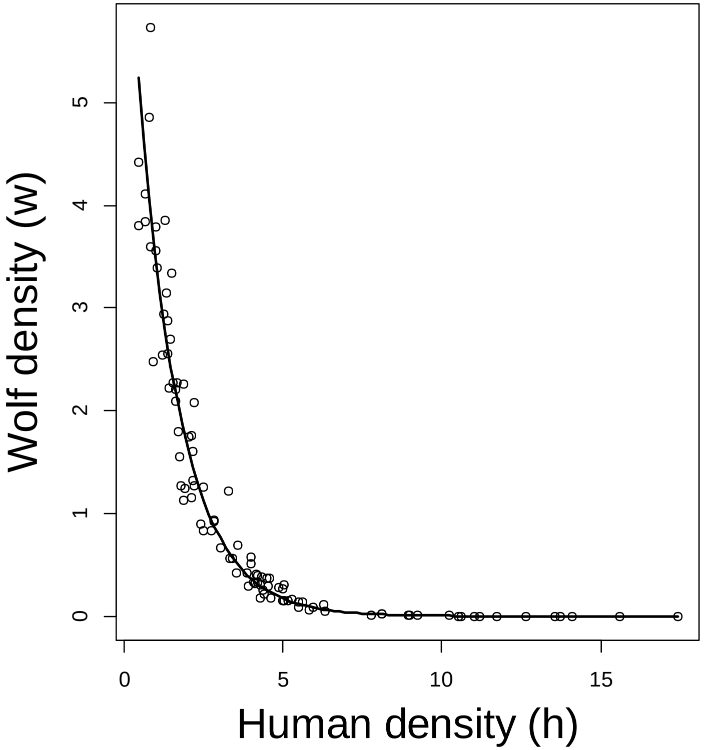
β_2

humans (h) $\xrightarrow{\alpha_1}$ wolves (w) $\xrightarrow{\beta_1}$ ungulates (u)

The total effect of humans on wolves is easy to estimate

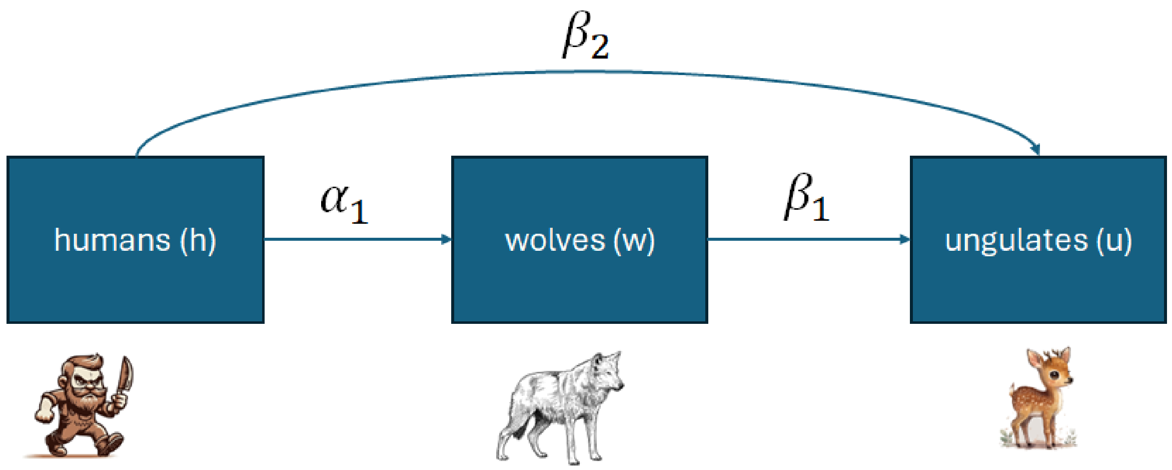
α_1

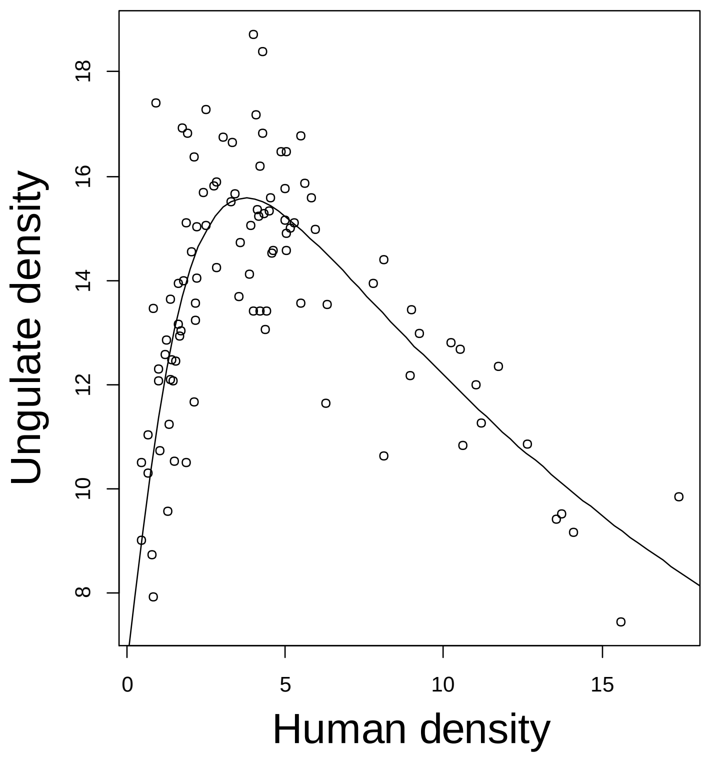




The total effect of humans on ungulates is…

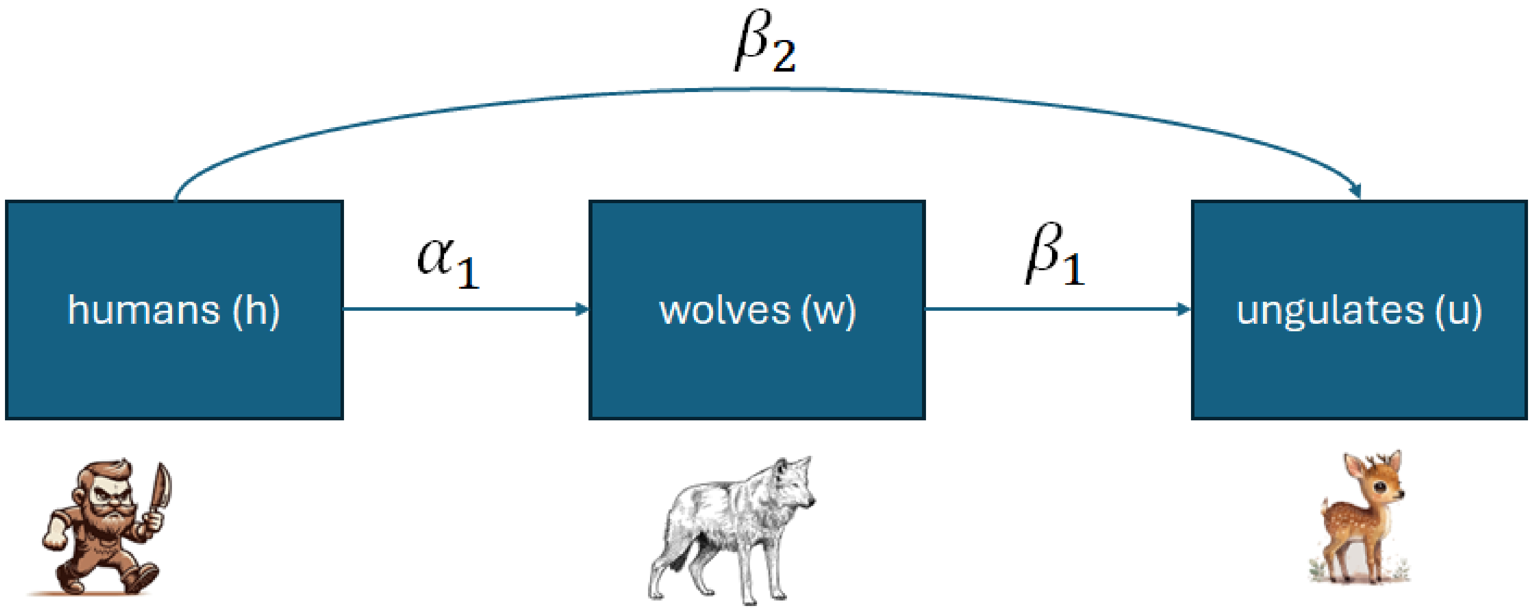
A function of both β$_2$ (the direct effect) as well as the indirect pathway through wolves (a ‘mediator’) which is a function of α$_1$ and β$_1$





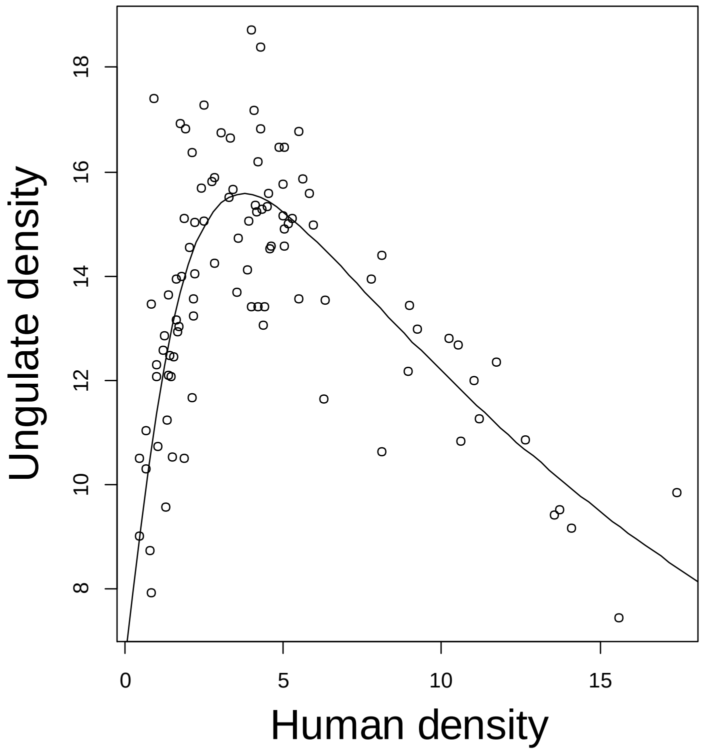
If identity link, simply α$_1$β$_1$ + β$_2$ (we won’t use that link much…)

Direct and indirect effects



β_2

humans (h) $\xrightarrow{\alpha_1}$ wolves (w) $\xrightarrow{\beta_1}$ ungulates (u)



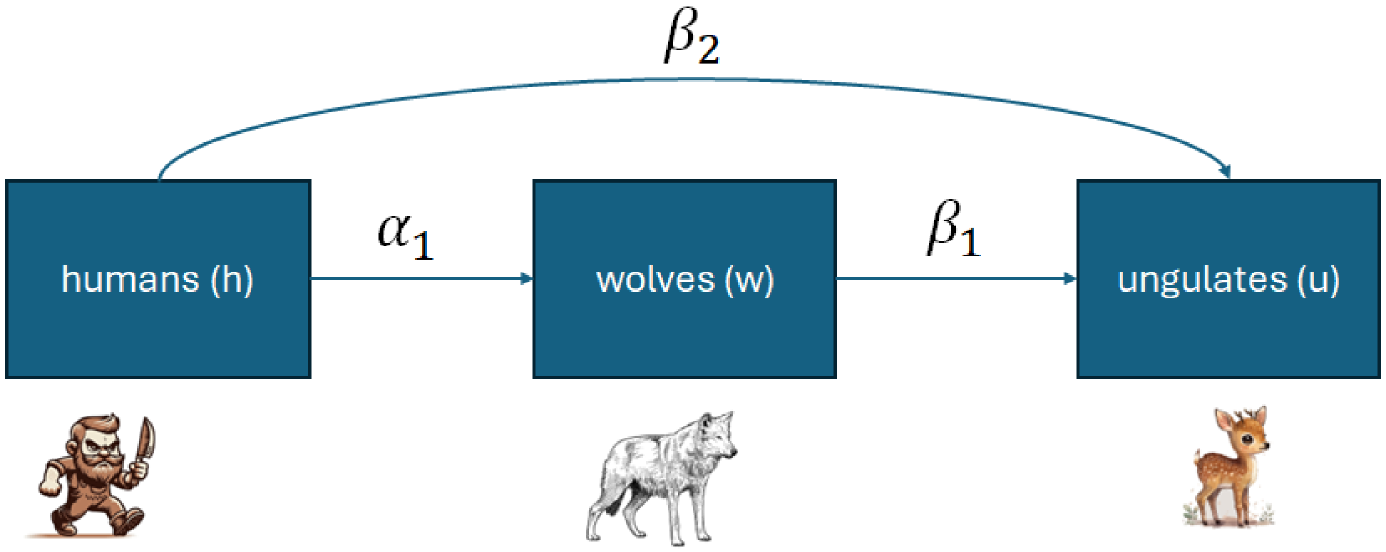
α_1, β_1, & β_2 are direct effects, the indirect effect is a function of α_1 & β_1

Today we have three main goals:

1. Fit a glm() and piecewiseSEM

2. Fit this in lavaan and blavaan (and export a JAGS file)

3. Fit this in JAGS and make predictions/figures



β_2

humans (h)

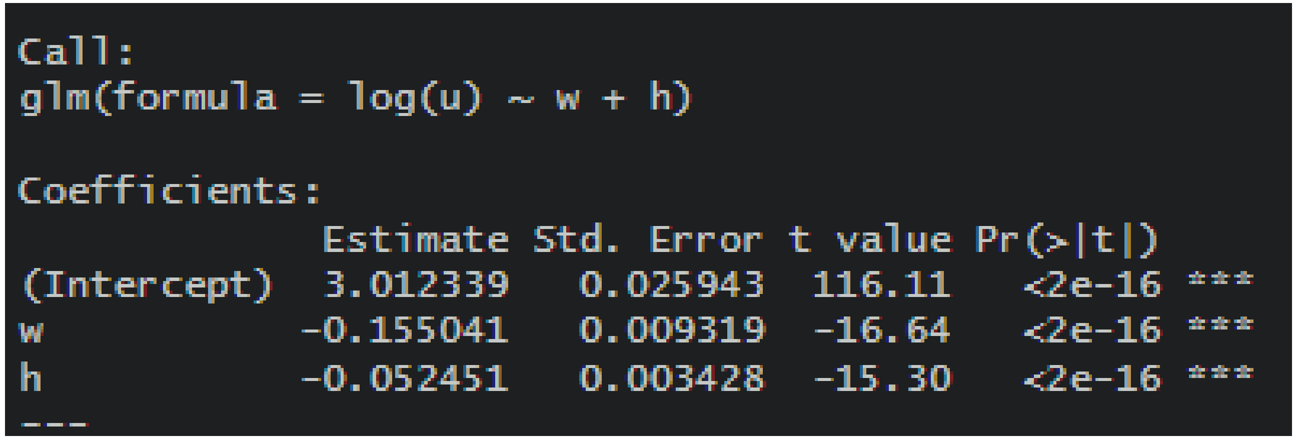
α_1

wolves (w)

β_1

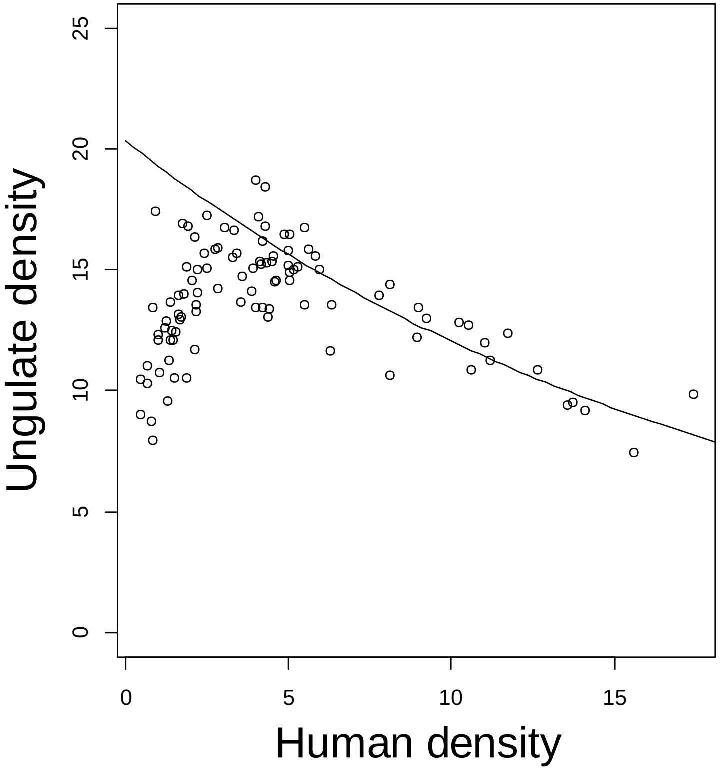
ungulates (u)

Fit a glm()



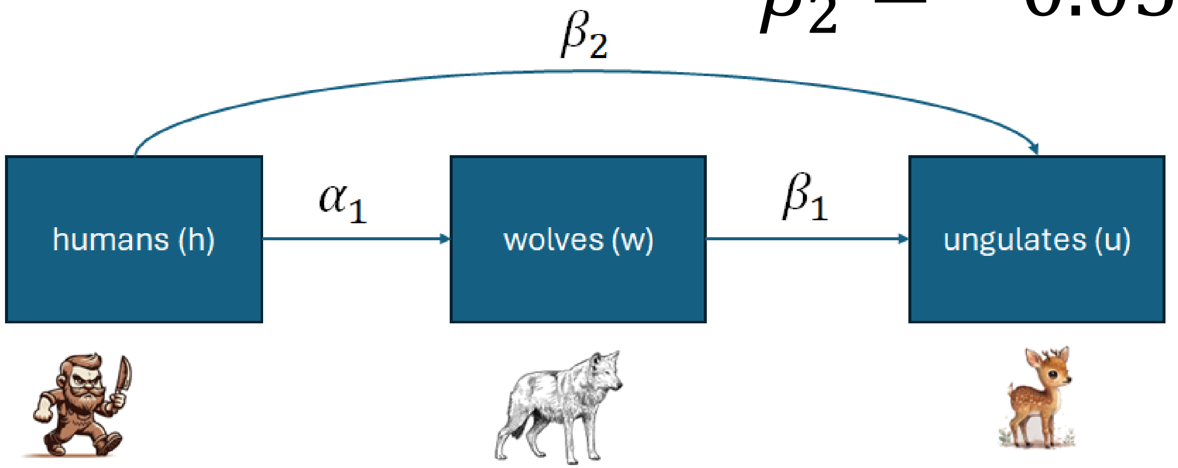
```
Call:
glm(formula = log(u) ~ w + h)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.012339   0.025943  116.11   <2e-16 ***
w           -0.155041   0.009319  -16.64   <2e-16 ***
h           -0.052451   0.003428  -15.30   <2e-16 ***
---
```

glm(log(u) ~ w + h)

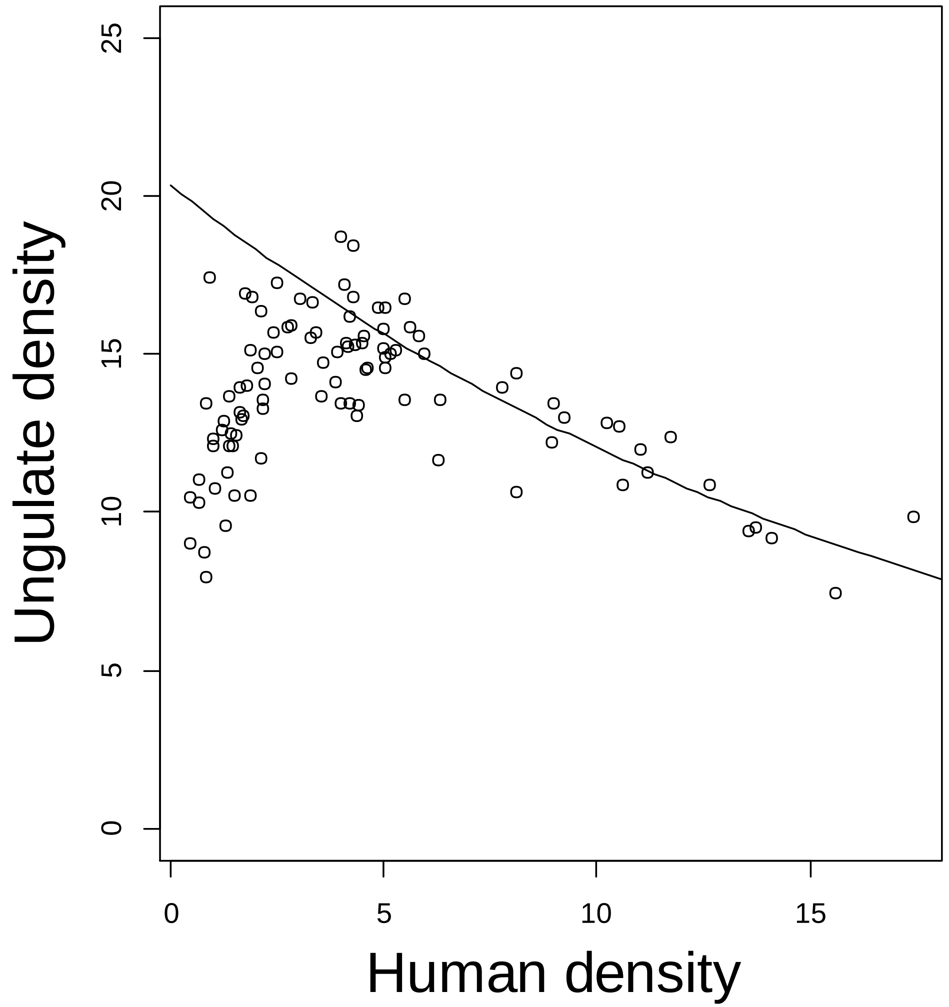


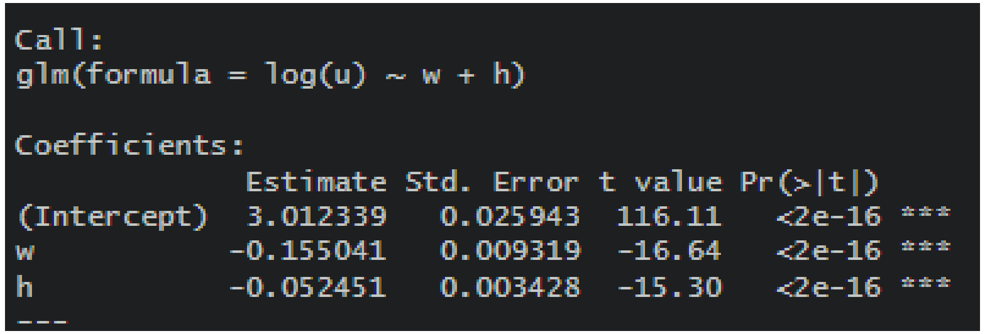
$$\beta_1 = -0.15$$

$$\beta_2 = -0.05$$

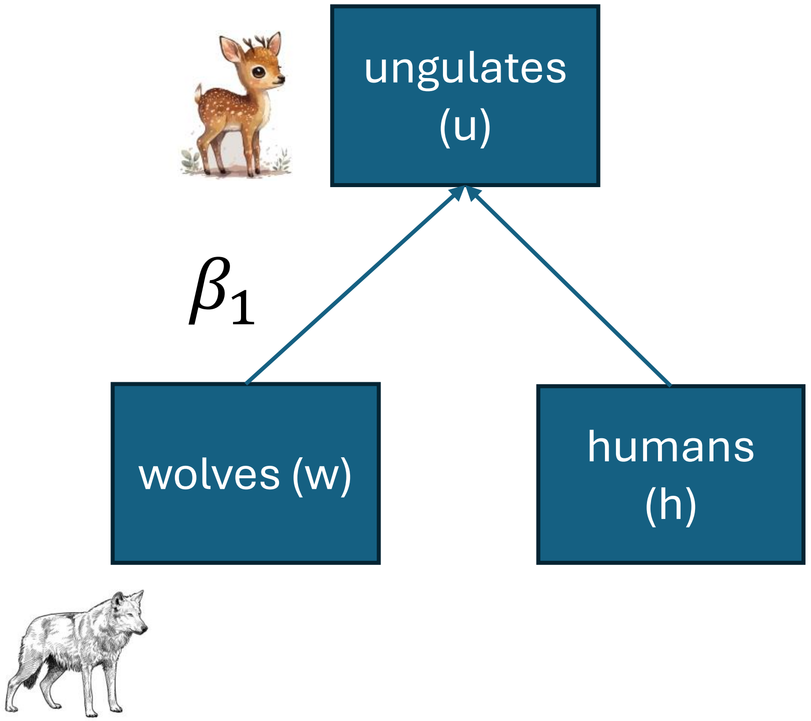


This is the relationship we'd predict if we held *w* = 0 and varied *h*





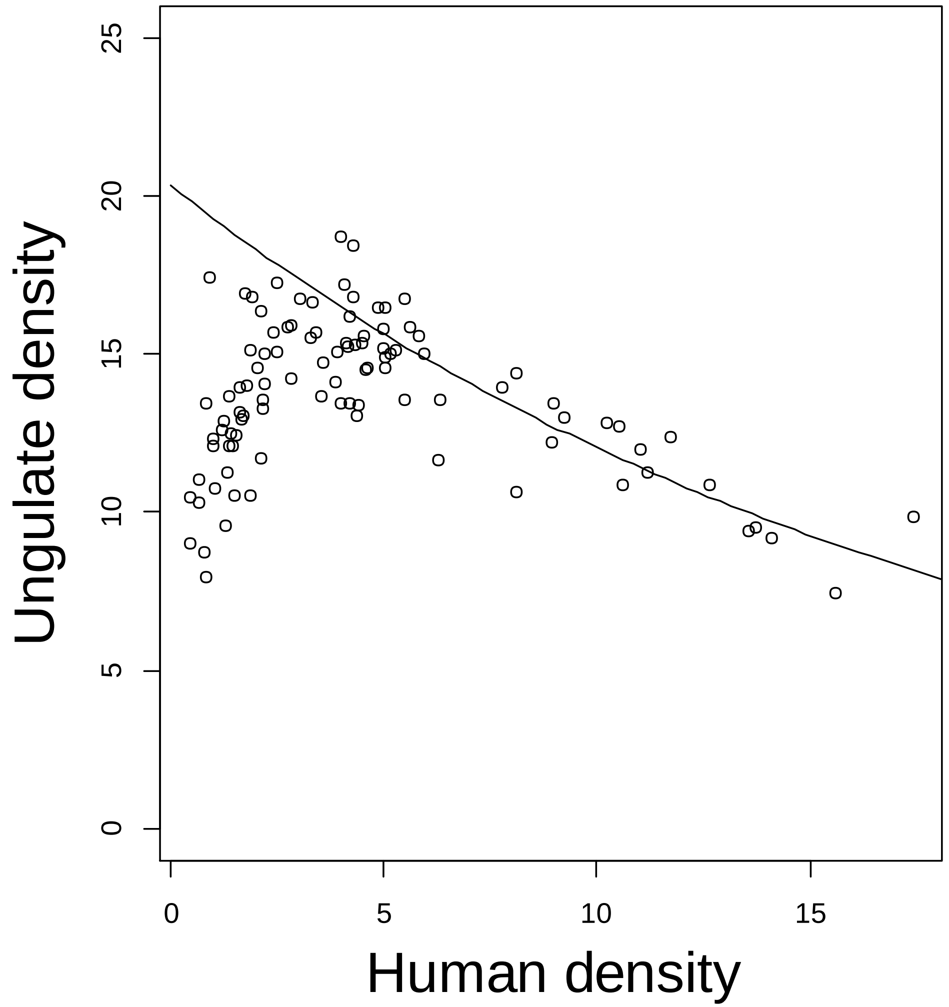
```
Call:
glm(formula = log(u) ~ w + h)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   3.012339   0.025943  116.11   <2e-16 ***
w            -0.155041   0.009319  -16.64   <2e-16 ***
h            -0.052451   0.003428  -15.30   <2e-16 ***
---
```

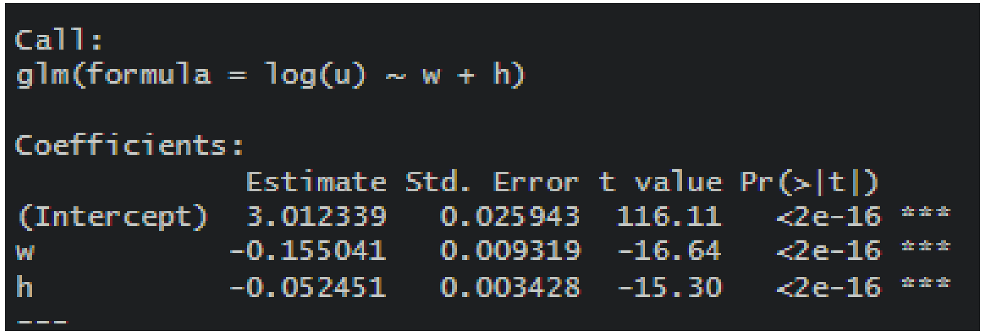


$$\beta_1 = -0.15$$

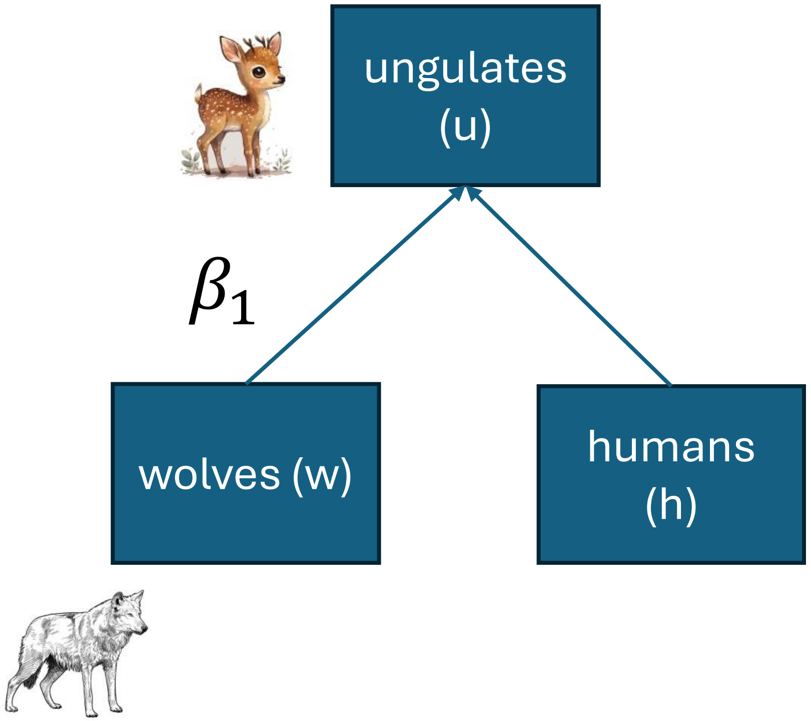
$$\beta_2 = -0.05$$

We can’t hold wolves constant and vary *h*





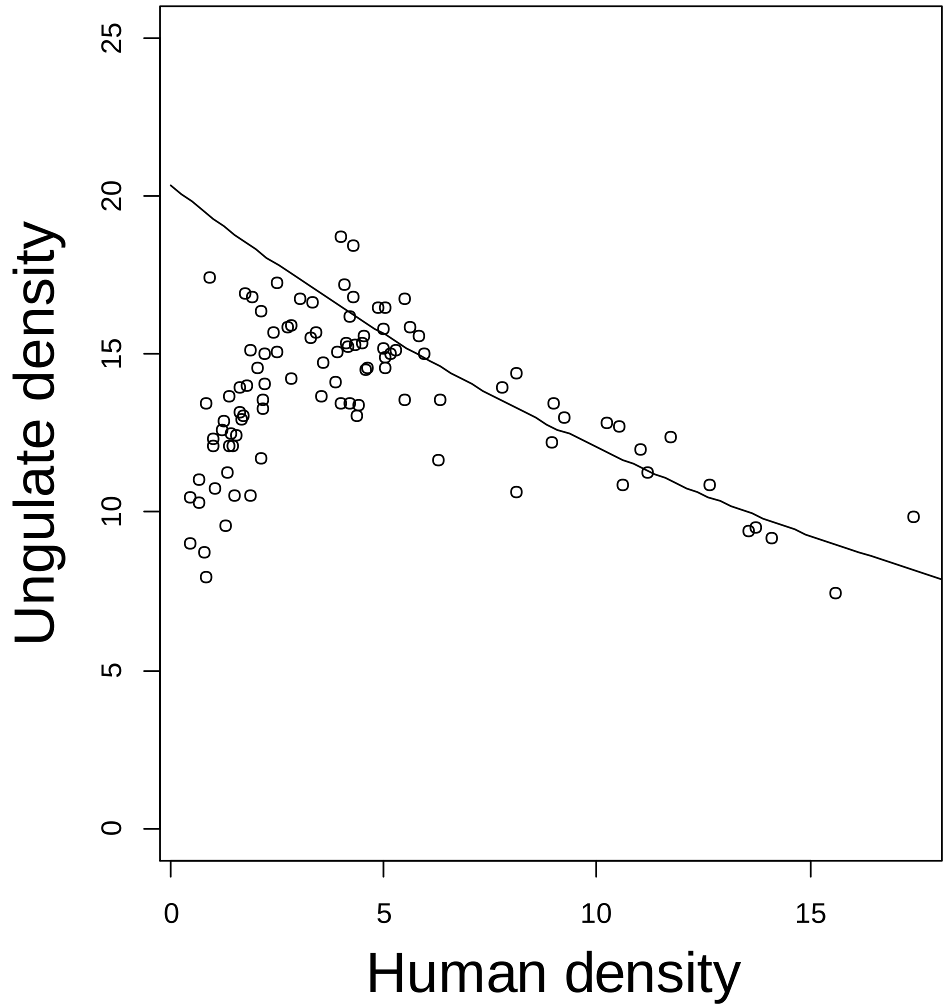
```
Call:
glm(formula = log(u) ~ w + h)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.012339   0.025943  116.11   <2e-16 ***
w           -0.155041   0.009319  -16.64   <2e-16 ***
h           -0.052451   0.003428  -15.30   <2e-16 ***
---
```

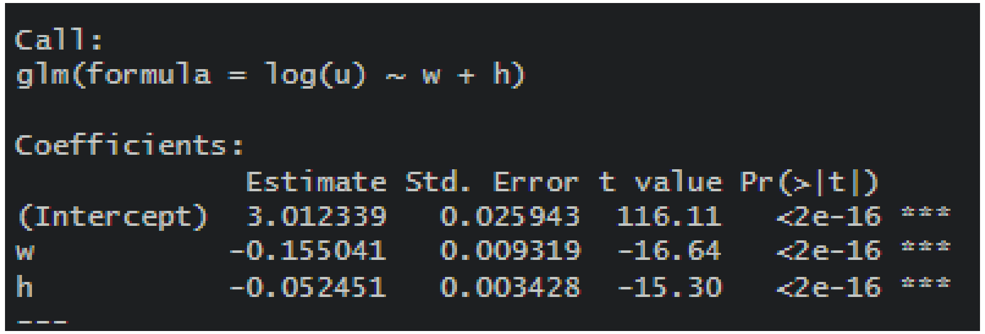


$$\beta_1 = -0.15$$

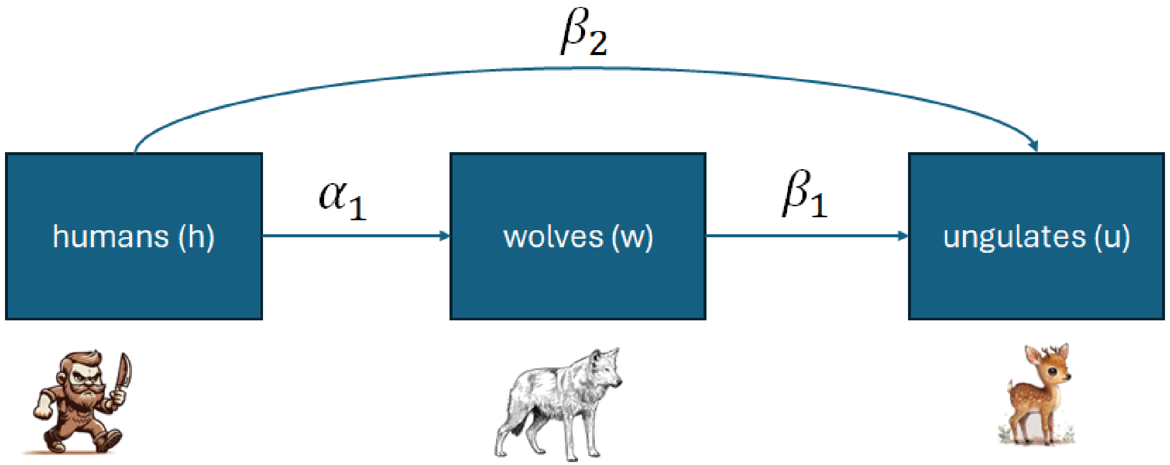
$$\beta_2 = -0.05$$

We can't hold *w* constant and vary *h*, because *h* affects *w*!

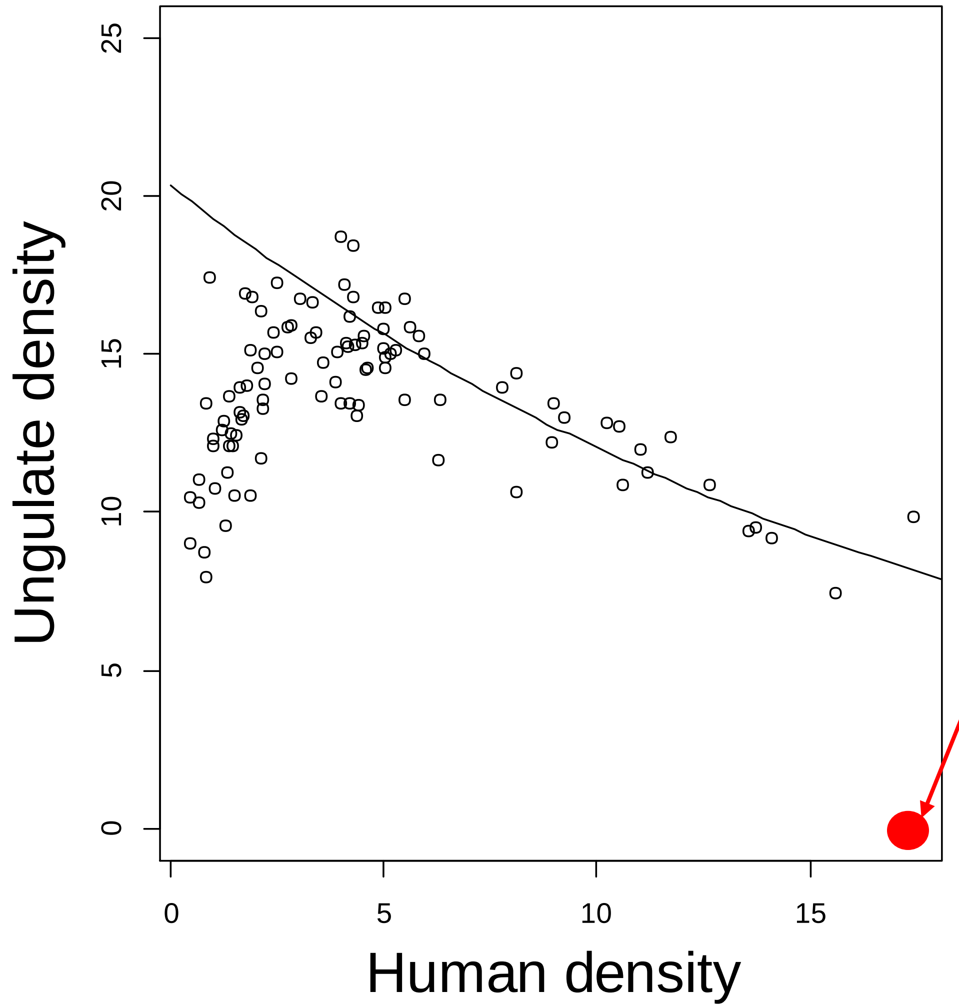


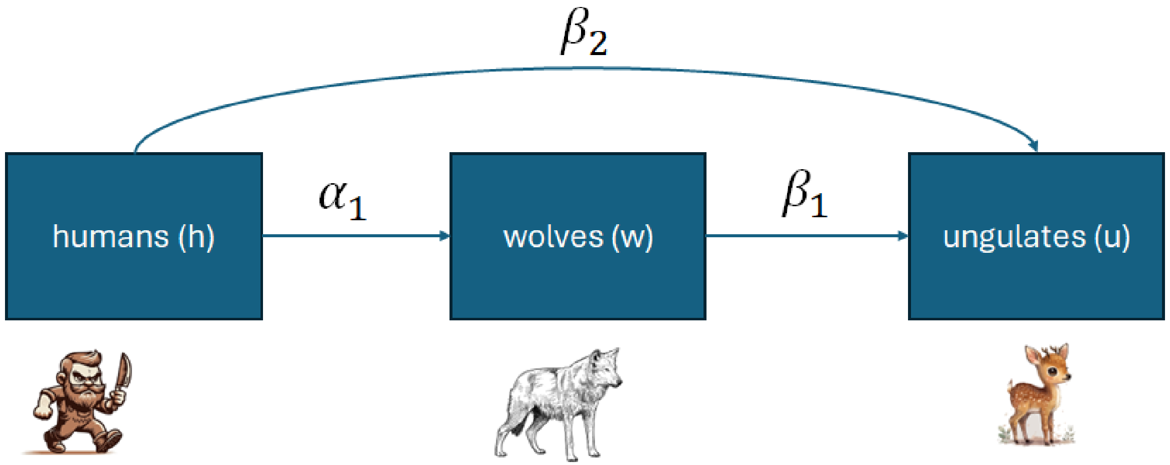


```
Call:
glm(formula = log(u) ~ w + h)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.012339   0.025943  116.11   <2e-16 ***
w           -0.155041   0.009319  -16.64   <2e-16 ***
h           -0.052451   0.003428  -15.30   <2e-16 ***
---
```



This also helps avoid silly predictions (what if *h* and *w* were high?)



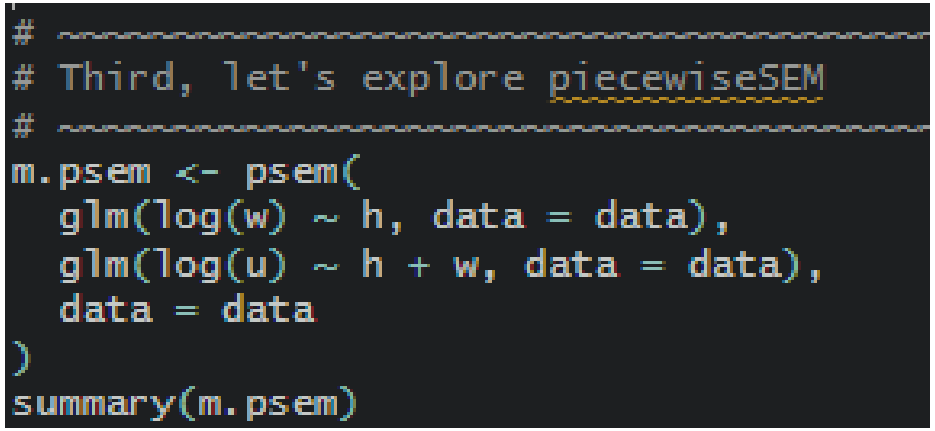


humans (h)

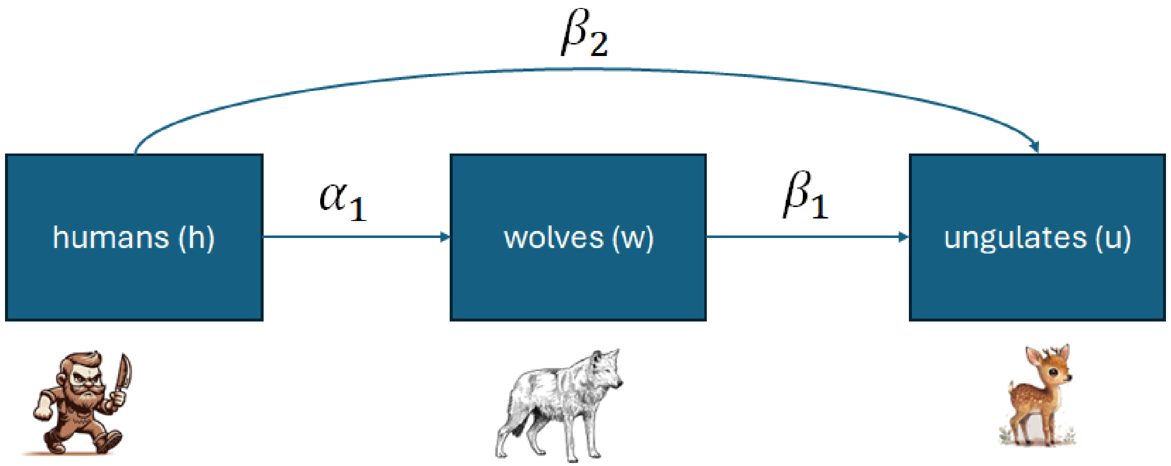
wolves (w)

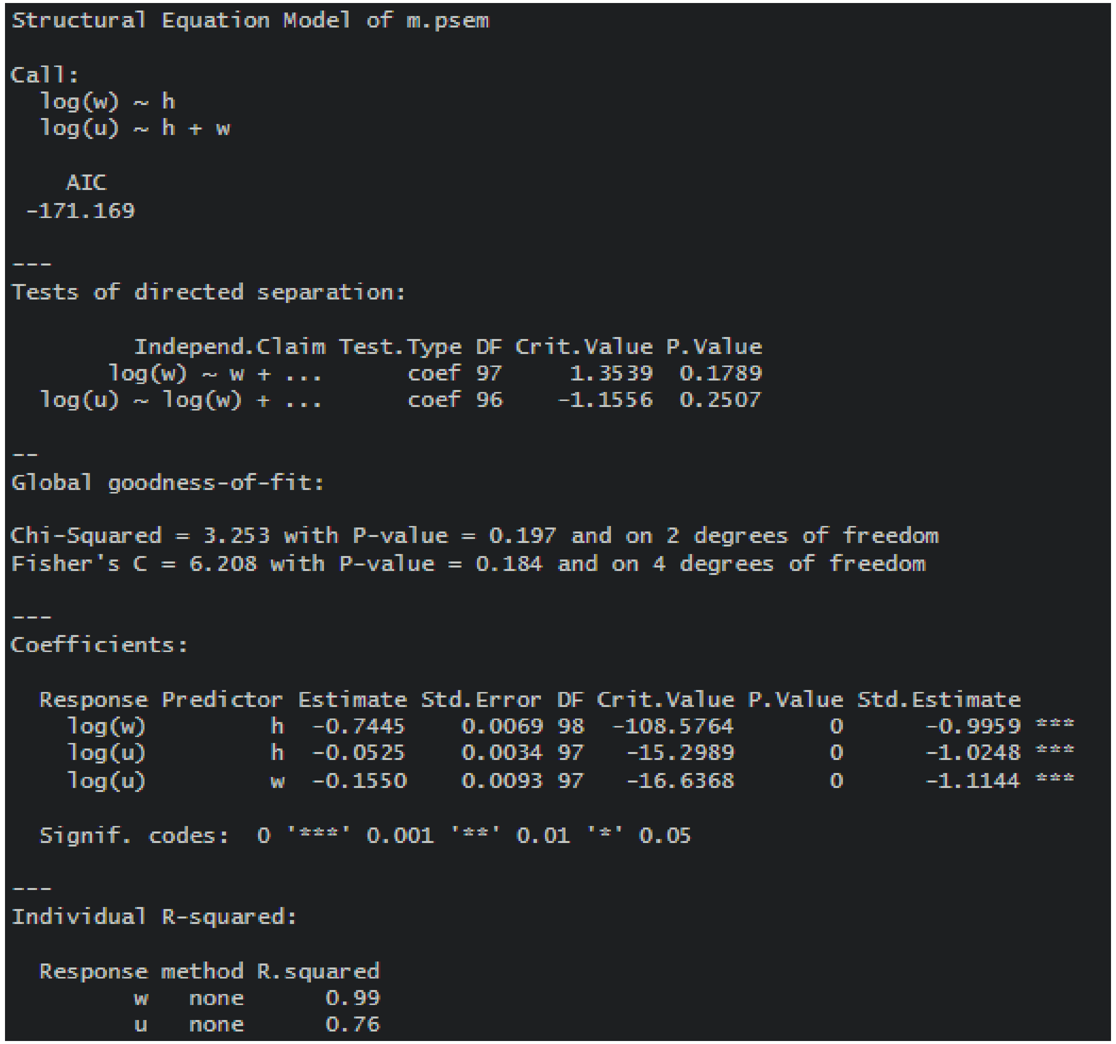
ungulates (u)

Fit in piecewiseSEM (we can add link functions)



```
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
# Third, let's explore piecewiseSEM
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
m.psem <- psem(
  glm(log(w) ~ h, data = data),
  glm(log(u) ~ h + w, data = data),
  data = data
)
summary(m.psem)
```

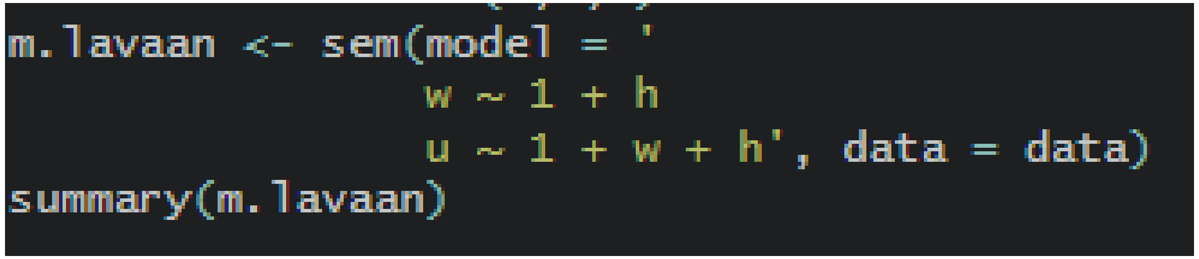




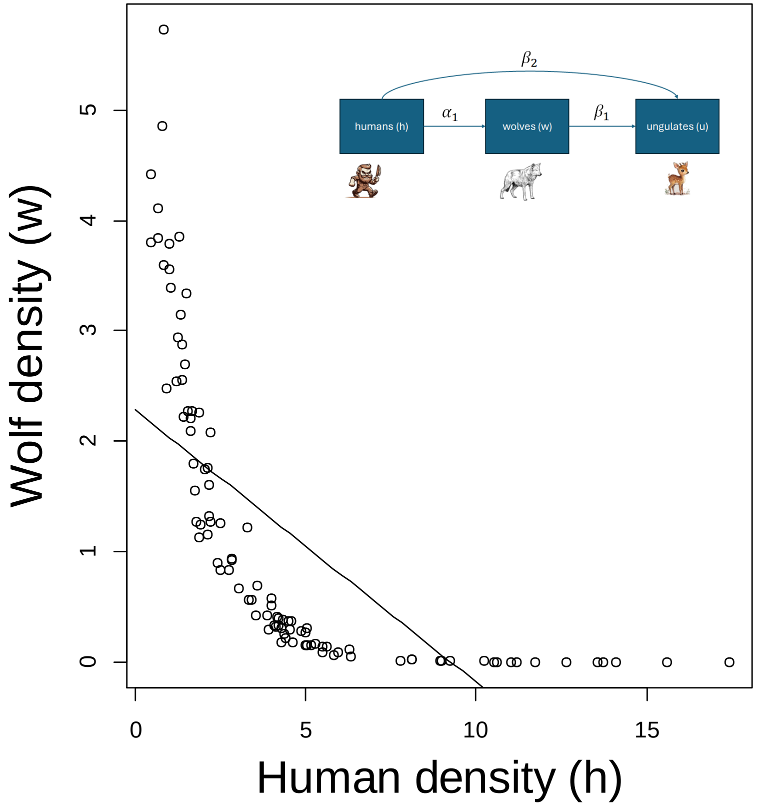
```
Structural Equation Model of m.psem

Call:
  log(w) ~ h
  log(u) ~ h + w

    AIC
 -171.169

---
Tests of directed separation:

         Independ.Claim Test.Type DF Crit.Value P.Value 
       log(w) ~ w + ...      coef 97     1.3539  0.1789 
  log(u) ~ log(w) + ...      coef 96    -1.1556  0.2507 

--
Global goodness-of-fit:

Chi-Squared = 3.253 with P-value = 0.197 and on 2 degrees of freedom
Fisher's C = 6.208 with P-value = 0.184 and on 4 degrees of freedom

---
Coefficients:

  Response Predictor Estimate Std.Error  DF Crit.Value P.Value Std.Estimate    
    log(w)         h  -0.7445    0.0069 98  -108.5764       0      -0.9959 ***
    log(u)         h  -0.0525    0.0034 97   -15.2989       0      -1.0248 ***
    log(u)         w  -0.1550    0.0093 97   -16.6368       0      -1.1144 ***

  Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05

---
Individual R-squared:

  Response method R.squared
         w   none      0.99
         u   none      0.76
```

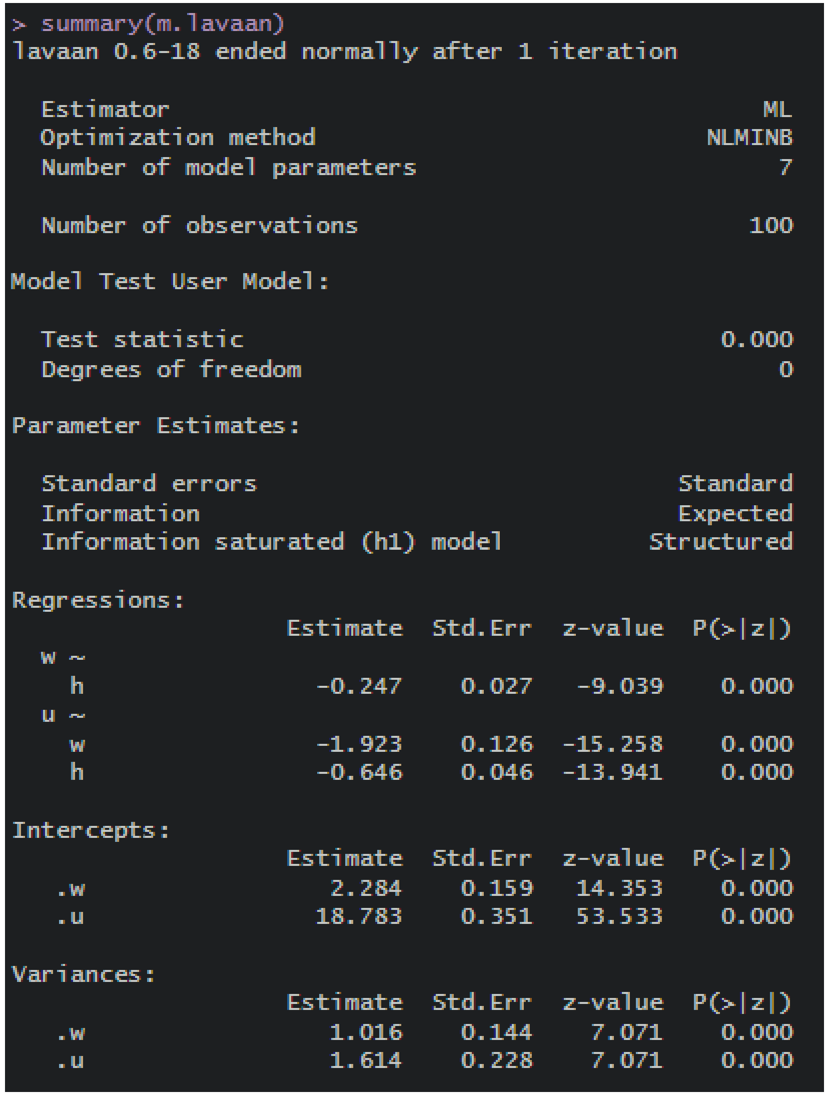
And we accurately recover our parameters!

Fit in lavaan



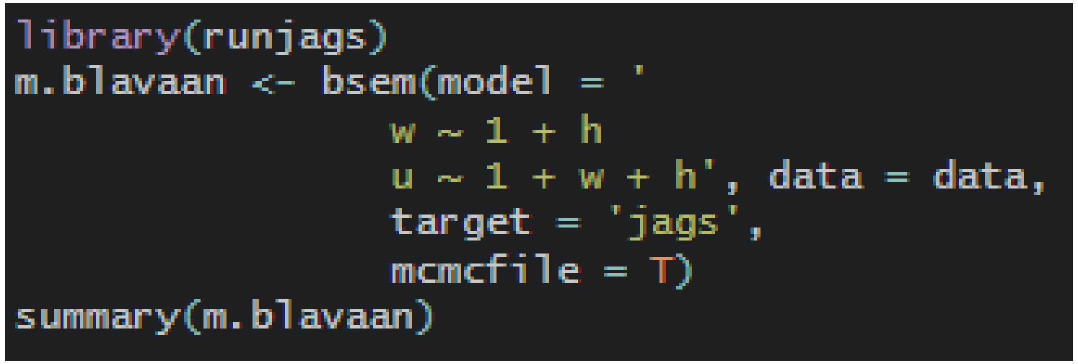
```
m.lavaan <- sem(model = '
                w ~ 1 + h
                u ~ 1 + w + h', data = data)
summary(m.lavaan)
```



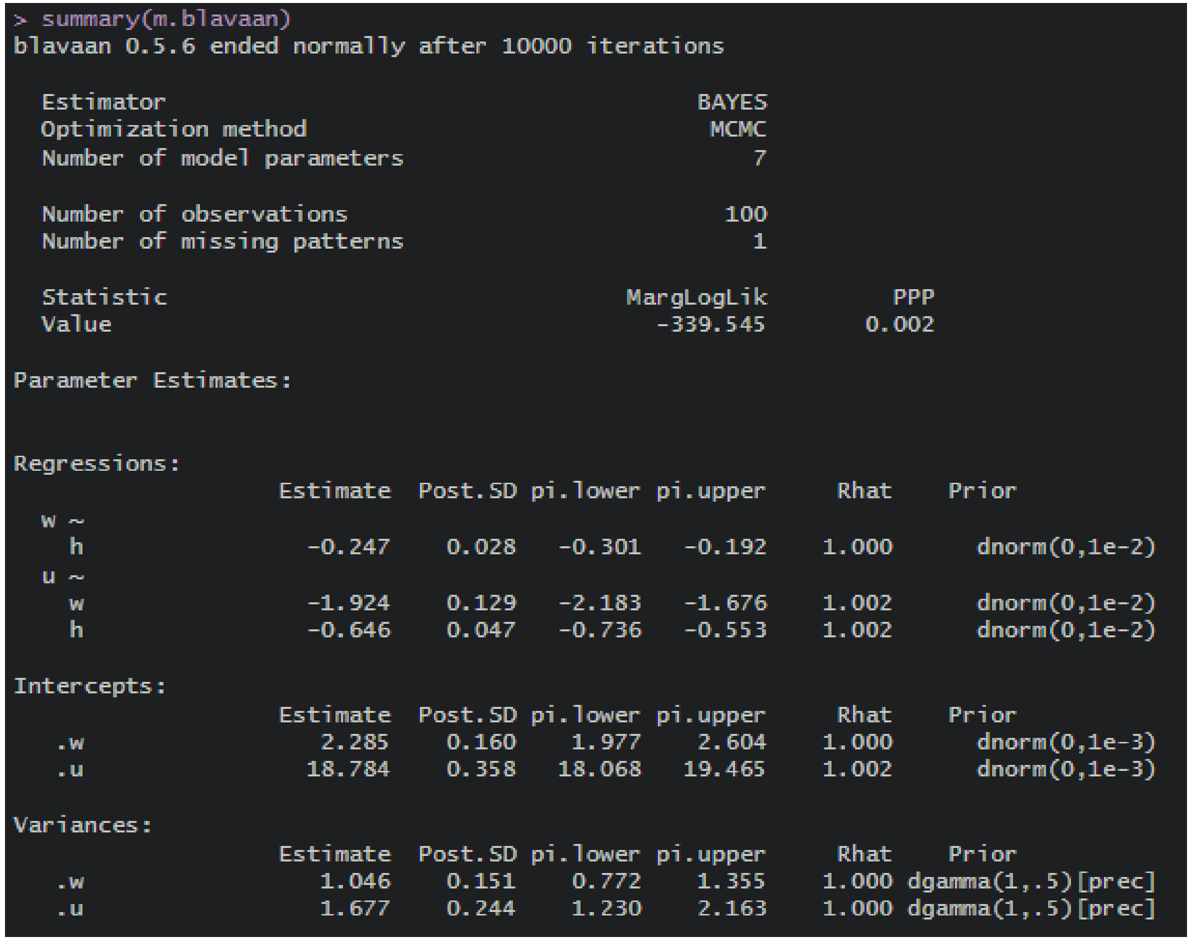


```
> summary(m.lavaan)
lavaan 0.6-18 ended normally after 1 iteration

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                         7

  Number of observations                           100

Model Test User Model:

  Test statistic                                 0.000
  Degrees of freedom                                 0

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Expected
  Information saturated (h1) model          Structured

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)
  w ~
    h                -0.247    0.027   -9.039    0.000
  u ~
    w                -1.923    0.126  -15.258    0.000
    h                -0.646    0.046  -13.941    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .w                 2.284    0.159   14.353    0.000
   .u                18.783    0.351   53.533    0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .w                 1.016    0.144    7.071    0.000
   .u                 1.614    0.228    7.071    0.000
```

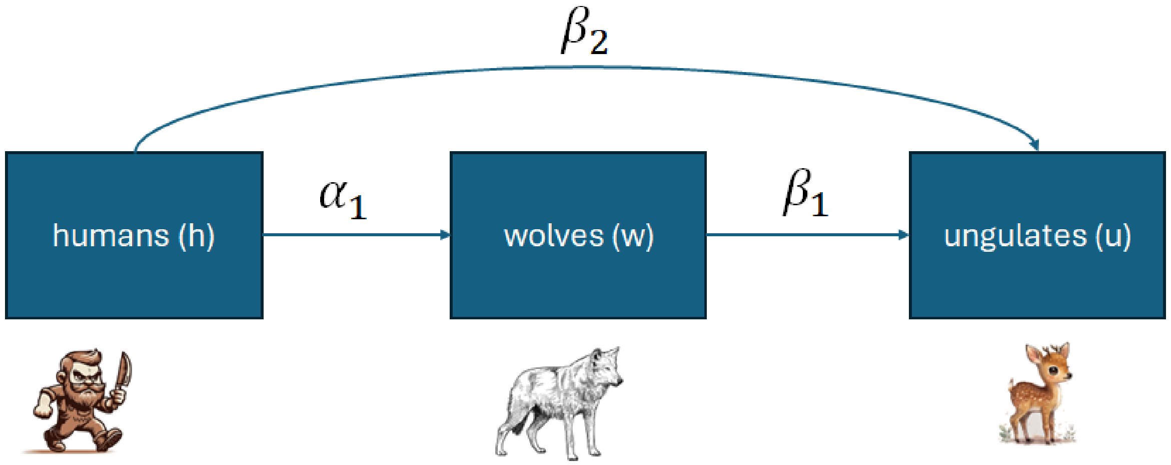
Fit in blavaan



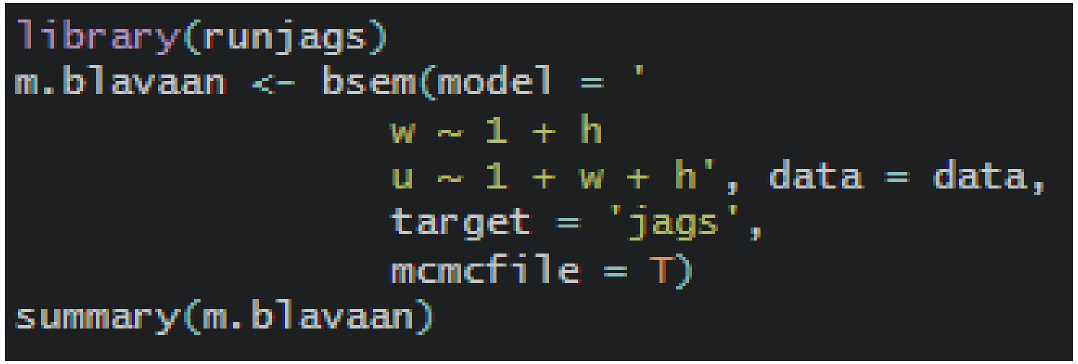
```
library(runjags)
m.blavaan <- bsem(model = '
                w ~ 1 + h
                u ~ 1 + w + h', data = data,
                target = 'jags',
                mcmcfile = T)
summary(m.blavaan)
```



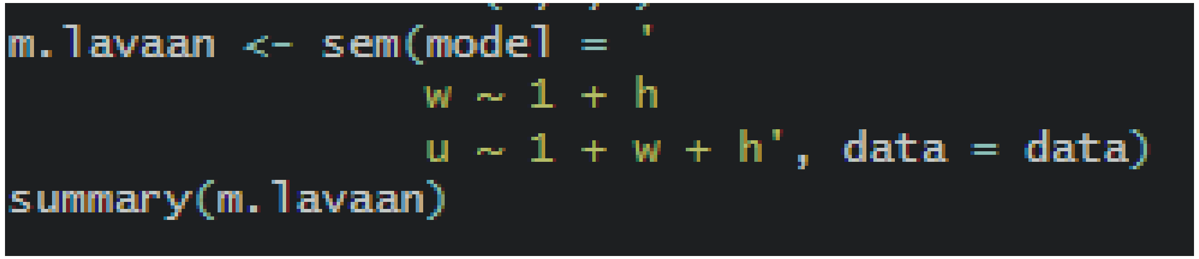
```
> summary(m.blavaan)
blavaan 0.5.6 ended normally after 10000 iterations

  Estimator                                      BAYES
  Optimization method                             MCMC
  Number of model parameters                         7

  Number of observations                           100
  Number of missing patterns                         1

  Statistic                                 MargLogLik         PPP
  Value                                       -339.545       0.002

Parameter Estimates:


Regressions:
                   Estimate  Post.SD pi.lower pi.upper     Rhat    Prior
  w ~
    h                -0.247    0.028   -0.301   -0.192    1.000      dnorm(0,1e-2)
  u ~
    w                -1.924    0.129   -2.183   -1.676    1.002      dnorm(0,1e-2)
    h                -0.646    0.047   -0.736   -0.553    1.002      dnorm(0,1e-2)

Intercepts:
                   Estimate  Post.SD pi.lower pi.upper     Rhat    Prior
   .w                 2.285    0.160    1.977    2.604    1.000      dnorm(0,1e-3)
   .u                18.784    0.358   18.068   19.465    1.002      dnorm(0,1e-3)

Variances:
                   Estimate  Post.SD pi.lower pi.upper     Rhat    Prior
   .w                 1.046    0.151    0.772    1.355    1.000 dgamma(1,.5)[prec]
   .u                 1.677    0.244    1.230    2.163    1.000 dgamma(1,.5)[prec]
```



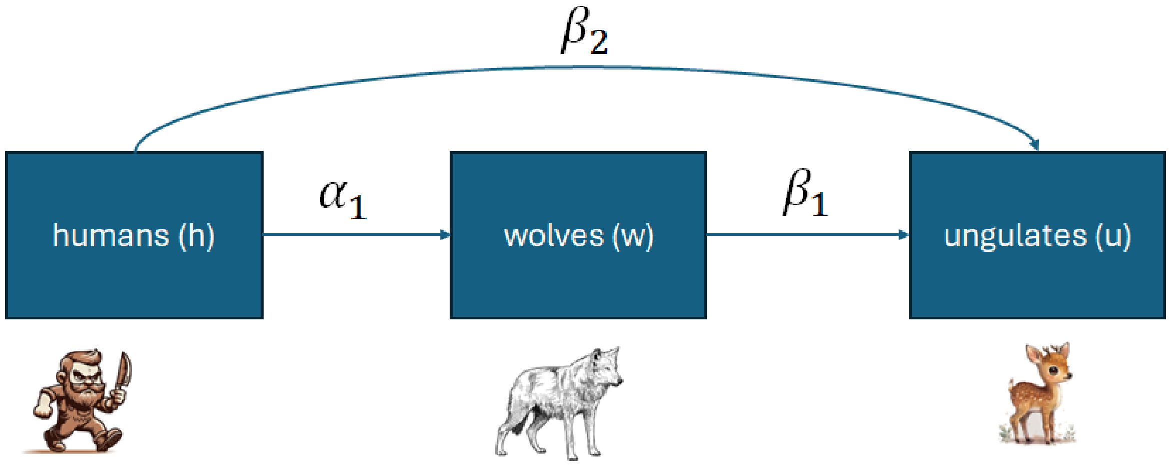
Note the similarities between blavaan and lavaan



```
library(runjags)
m.blavaan <- bsem(model = '
                w ~ 1 + h
                u ~ 1 + w + h', data = data,
                target = 'jags',
                mcmcfile = T)
summary(m.blavaan)
```



```
m.lavaan <- sem(model = '
                w ~ 1 + h
                u ~ 1 + w + h', data = data)
summary(m.lavaan)
```



β_2

humans (h)

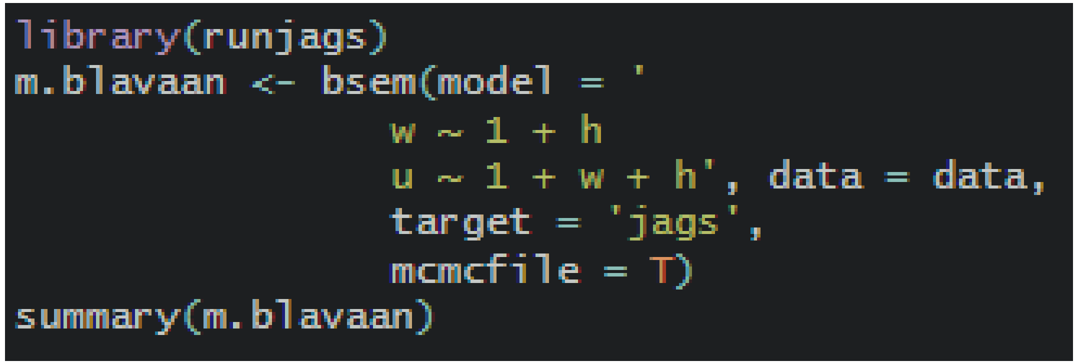
α_1

wolves (w)

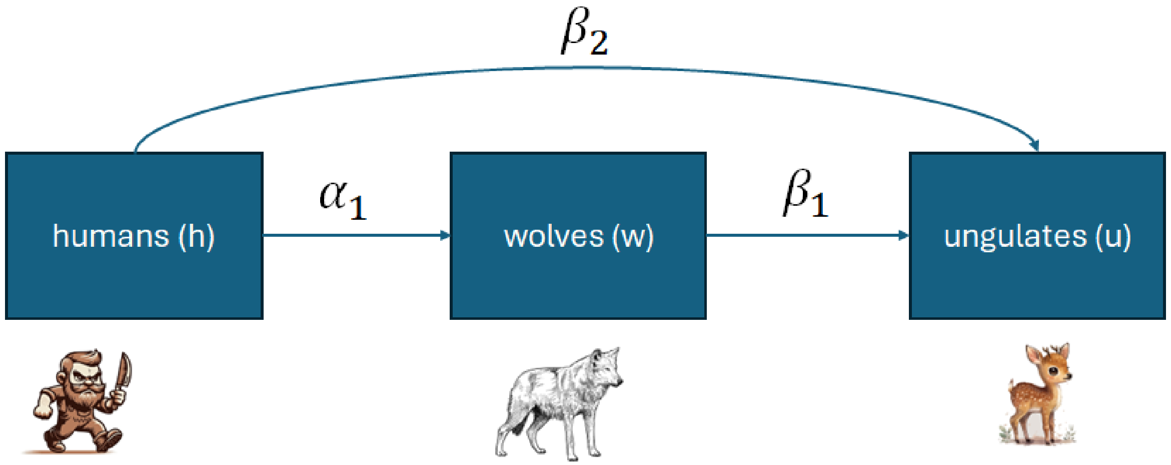
β_1

ungulates (u)

mcmcfile = T saves a working model in Stan or JAGS



```
library(runjags)
m.blavaan <- bsem(model = '
                w ~ 1 + h
                u ~ 1 + w + h', data = data,
                target = 'jags',
                mcmcfile = T)
summary(m.blavaan)
```



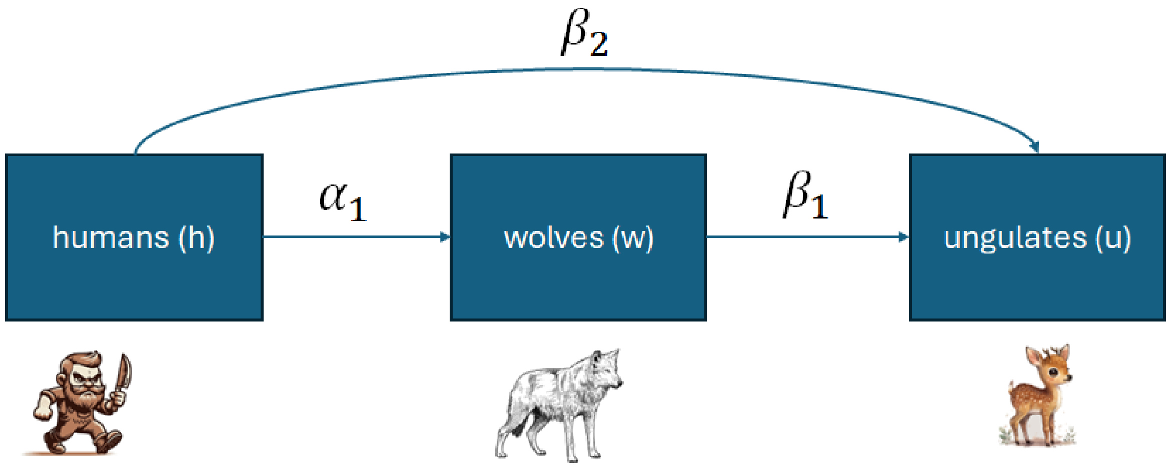
lavExport (folder in Documents or your working directory)

Let’s take a quick break, then we’ll go to our R script,
work through our own custom JAGS model,
and explore our first counter-factual*.

***i.e., what would’ve happened if…?**

lavExport (folder in Documents or your working directory)

Take-home 1: complex paths produce complex signals



β_2

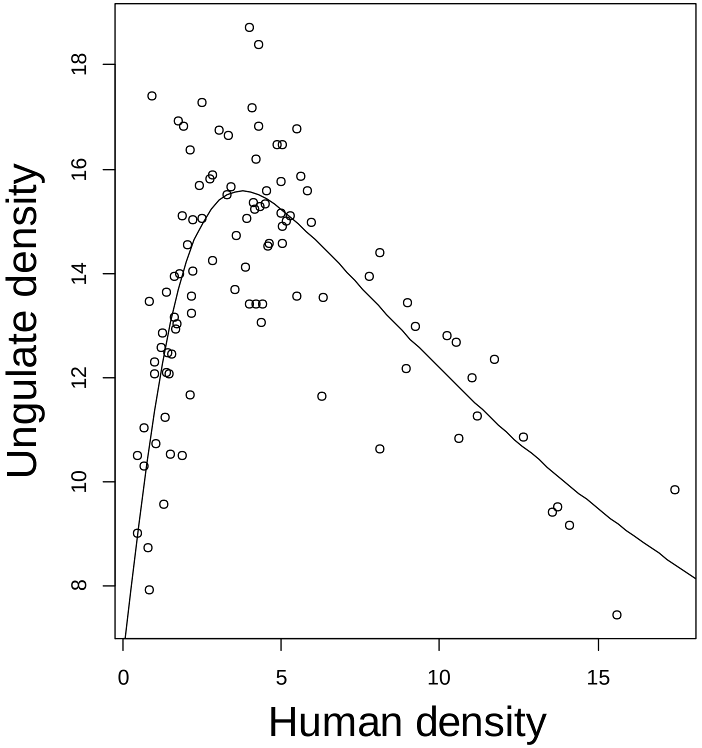
humans (h)

α_1

wolves (w)

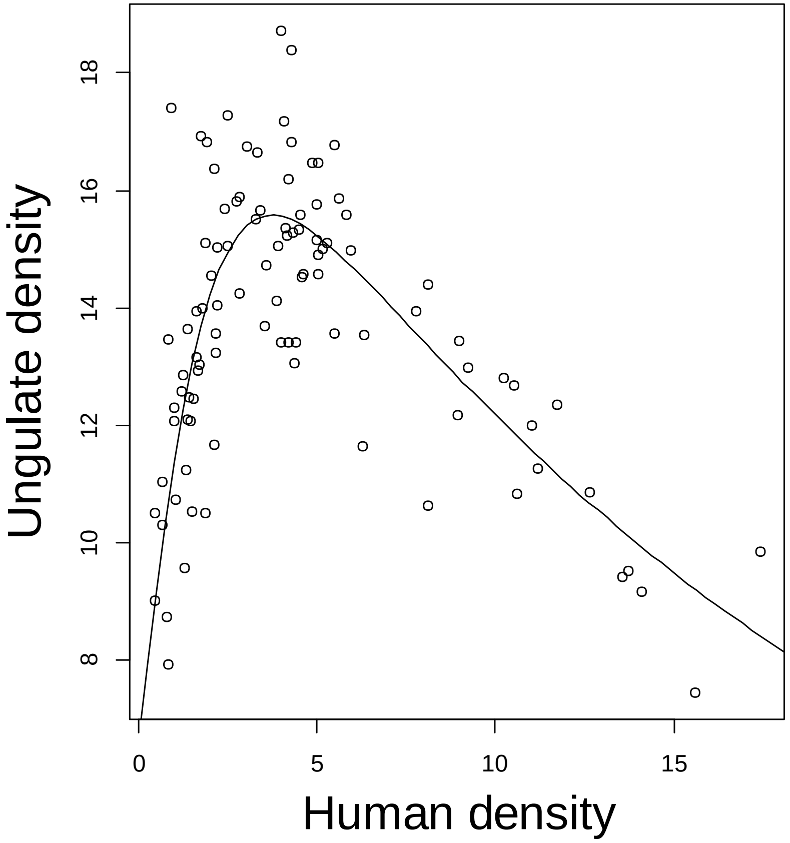
β_1

ungulates (u)



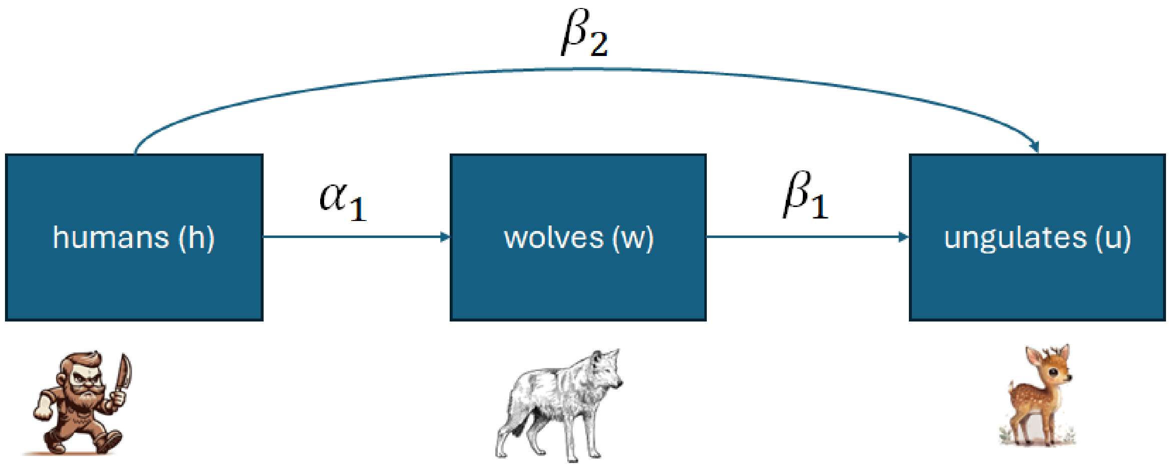
***Almost everything (worth studying) is the result of a complex path!**

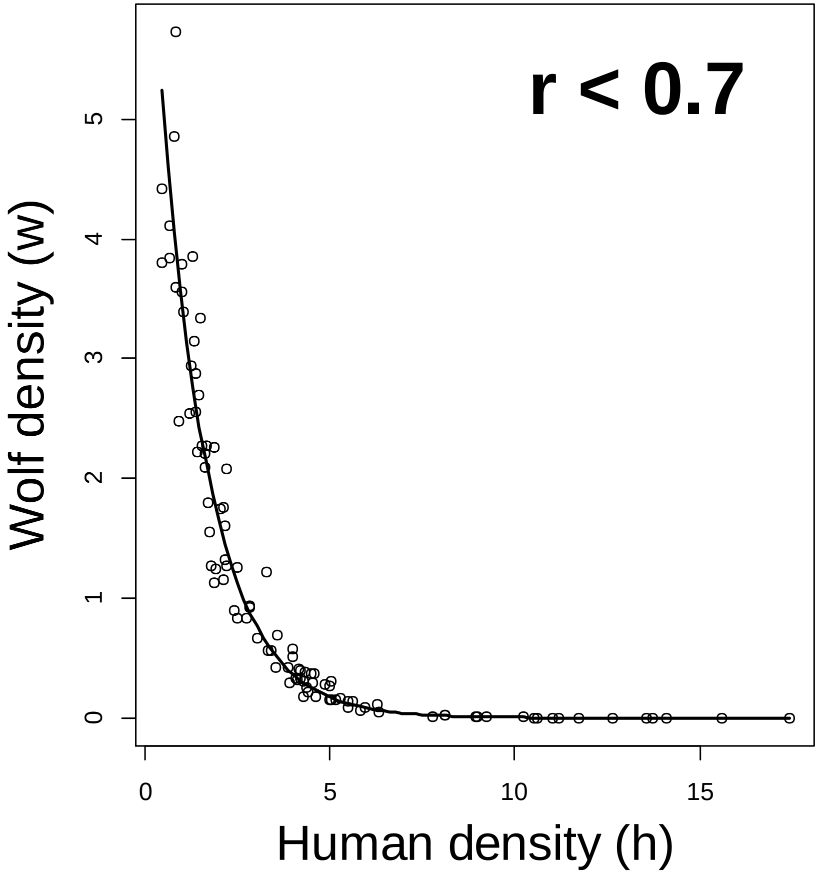
Take-home 2: univariate relationships can be misleading!



lm(u ~ h)

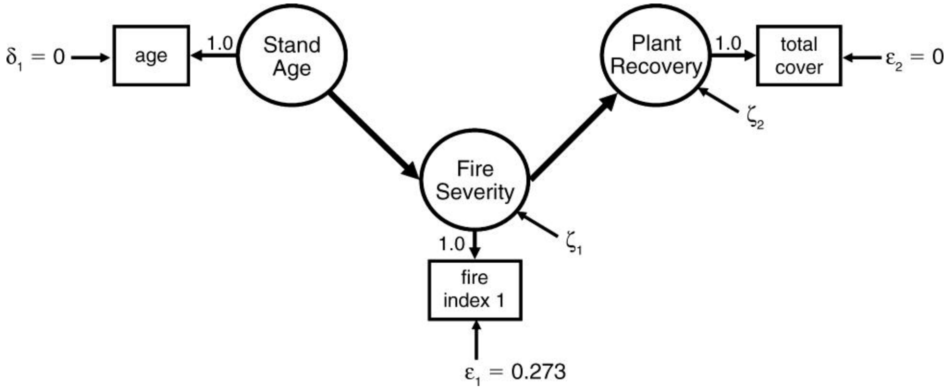
Take-home 3: Causal paths lead to multicollinearity





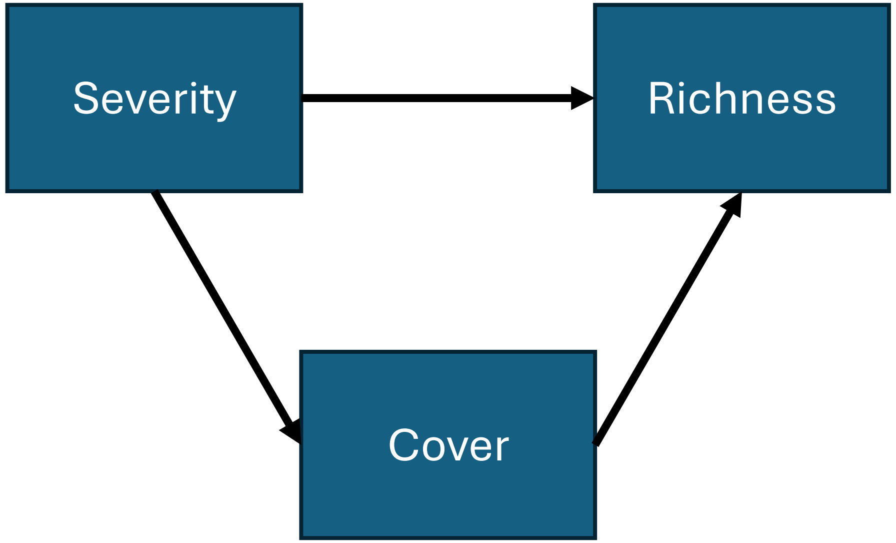
We can model those pathways to obtain more accurate predictive inference

Another example(s): the Keeley data [in ‘`psem`’ package]



Grace et al. **(2010)** *Ecological Monographs*

Grace & Keeley **(2006)** *Ecological Applications*

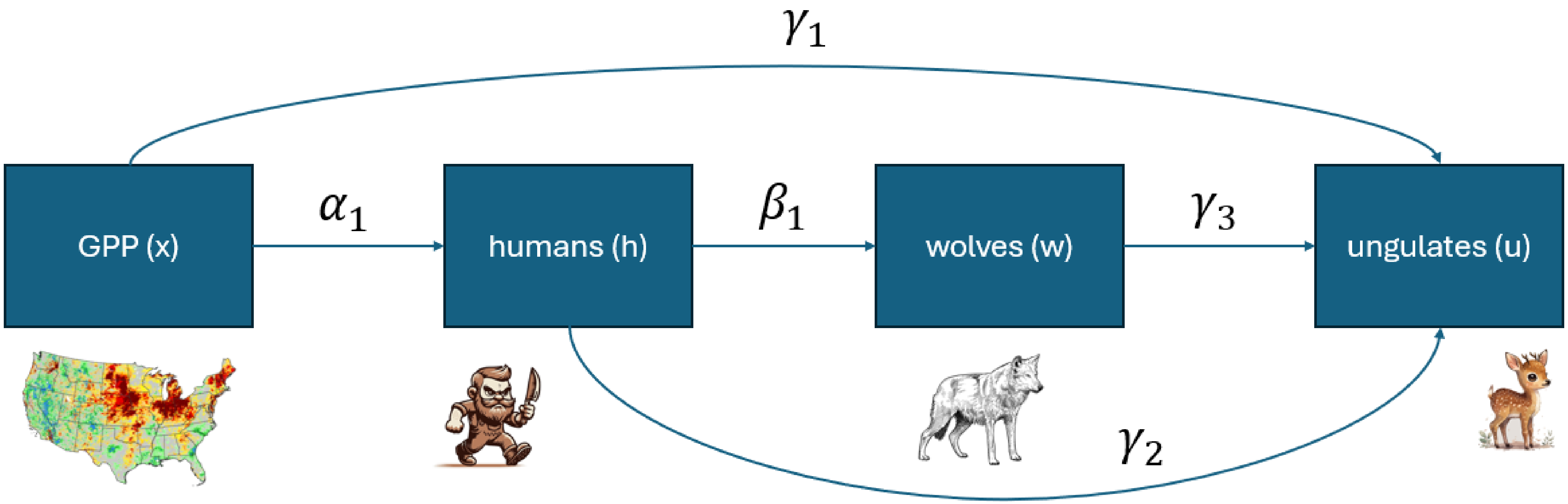


Some notes from Jon Lefcheck (jonlefcheck.net)

https://www.youtube.com/watch?v=VT-gw_VVP1E

https://jslefche.github.io/sem_book/

Following slides correspond to the supplementary script



γ_1

GPP (x)

α_1

humans (h)

β_1

wolves (w)

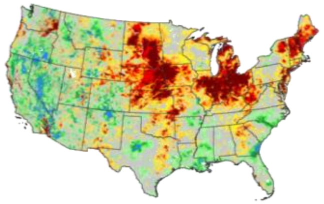
γ_3

ungulates (u)

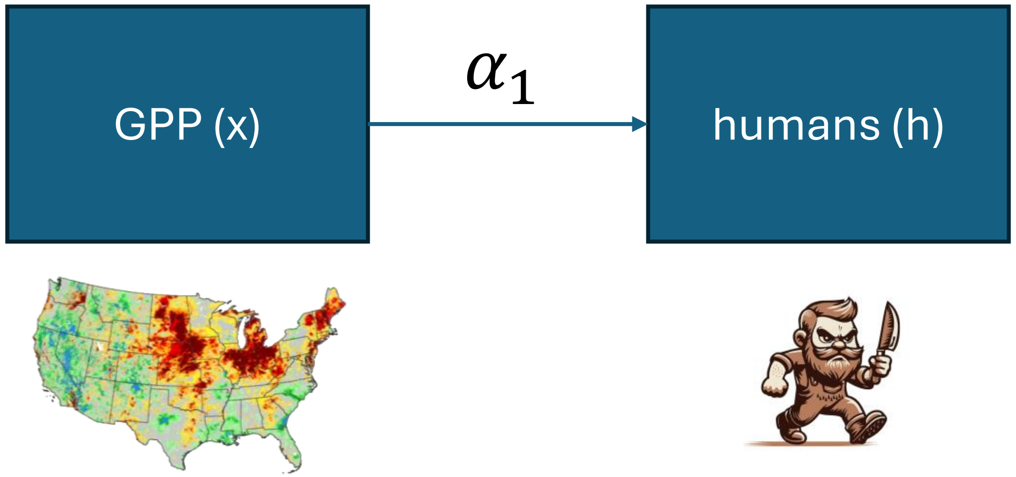
γ_2

Simulating the data: GPP

GPP (x)



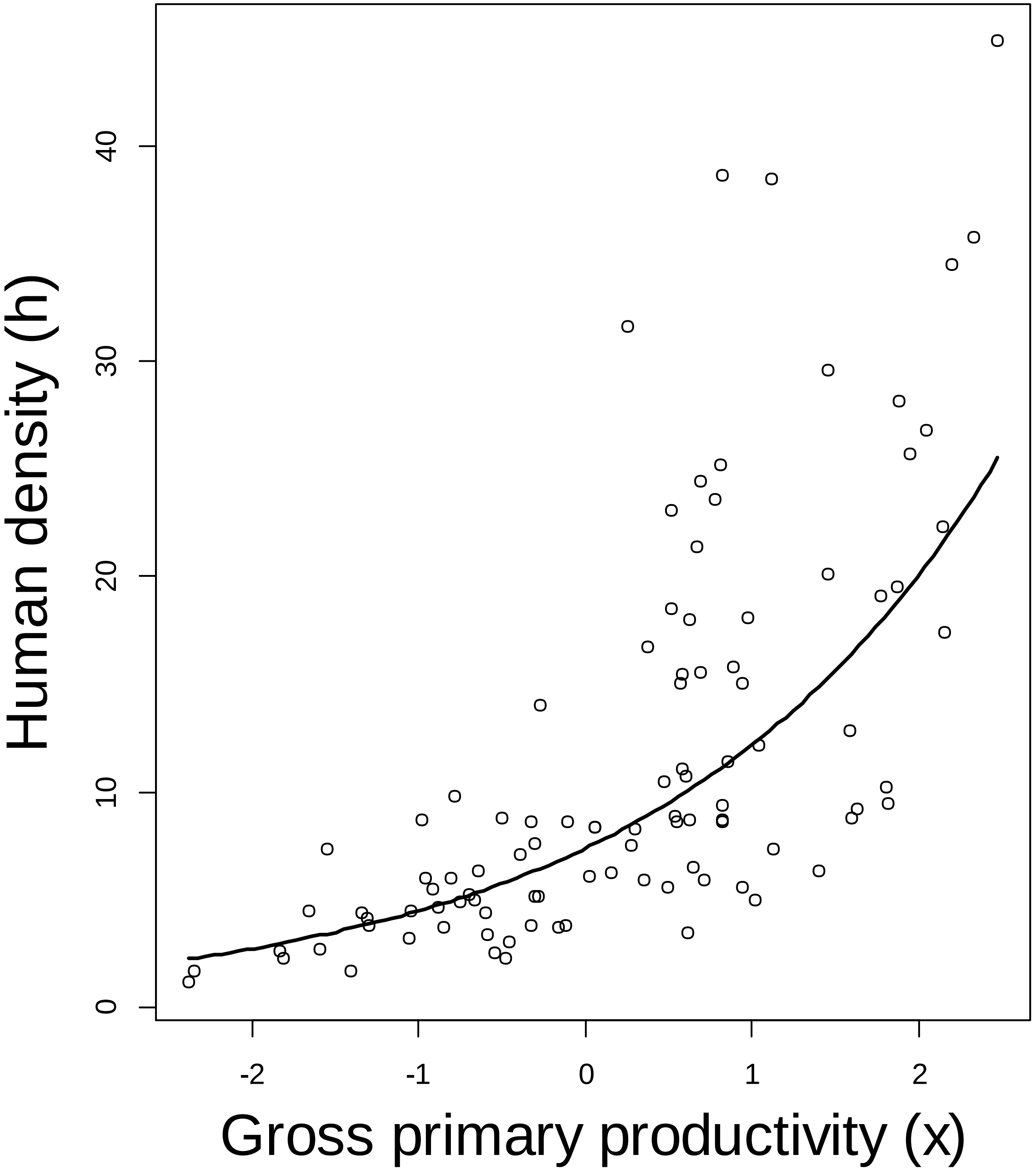
Simulating the data: humans



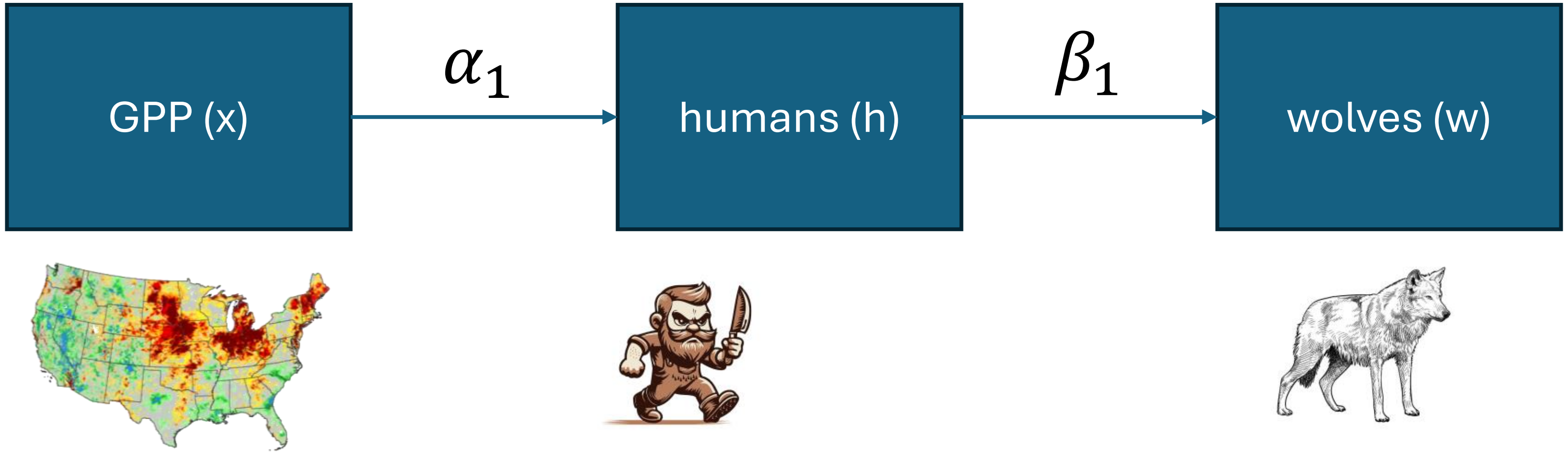
GPP (x)

α_1

humans (h)



Simulating the data: wolves



GPP (x)

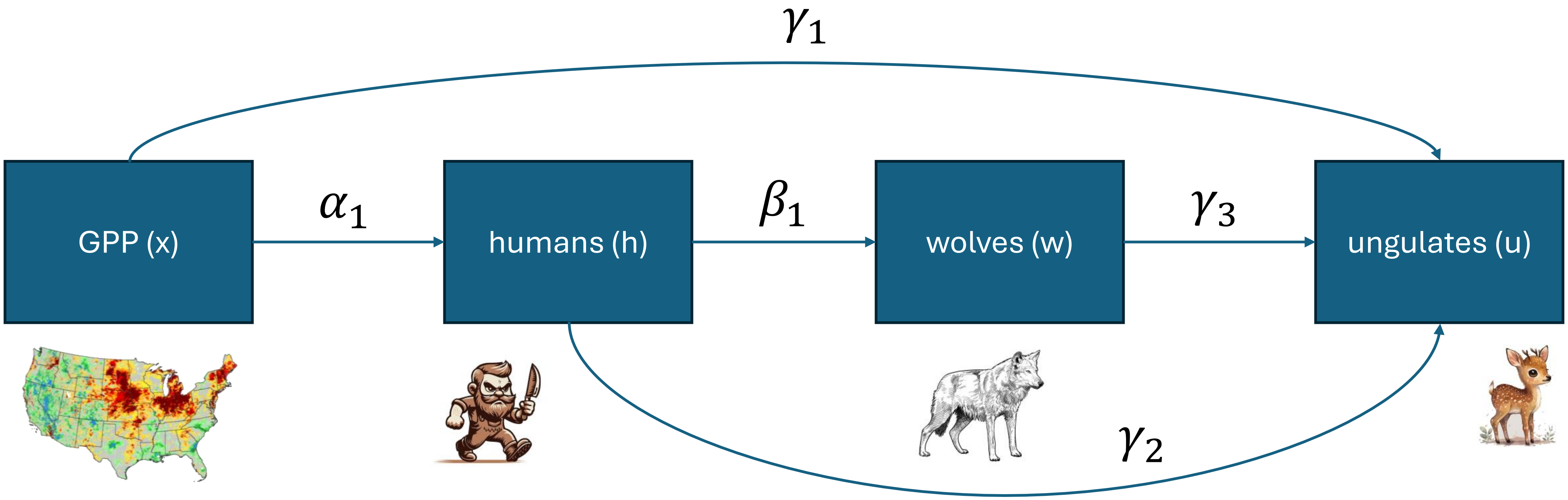
α_1

humans (h)

β_1

wolves (w)

Simulating the data: ungulates



γ_1

GPP (x)

α_1

humans (h)

β_1

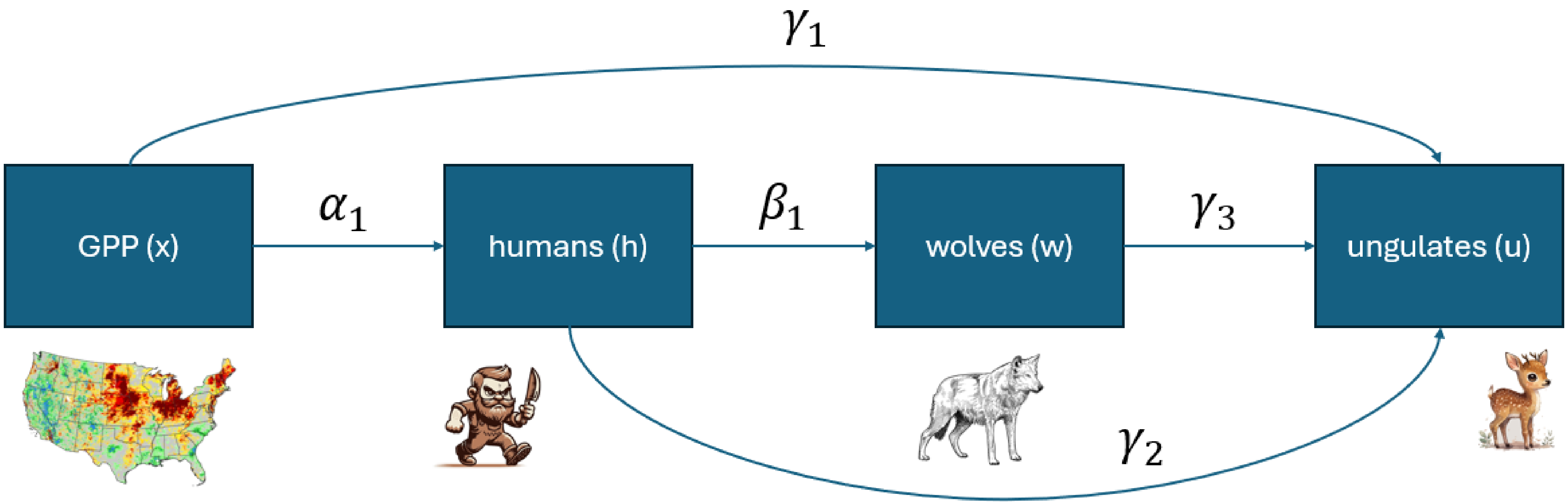
wolves (w)

γ_3

ungulates (u)

γ_2

So what does this all mean?!



γ_1

GPP (x)

α_1

humans (h)

β_1

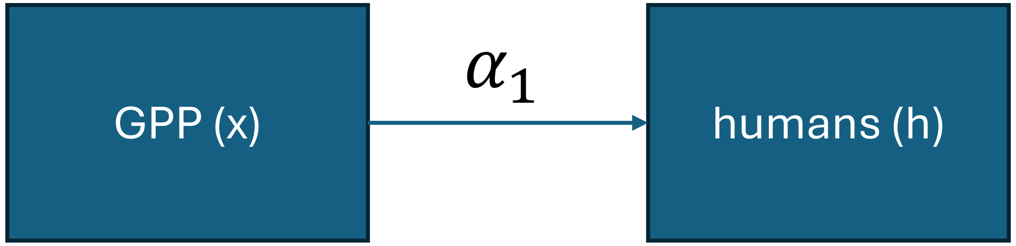
wolves (w)

γ_3

ungulates (u)

γ_2

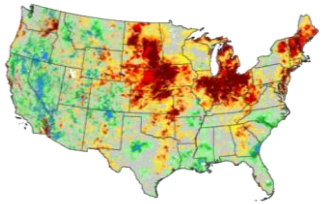
Simulating univariate relationships is simple

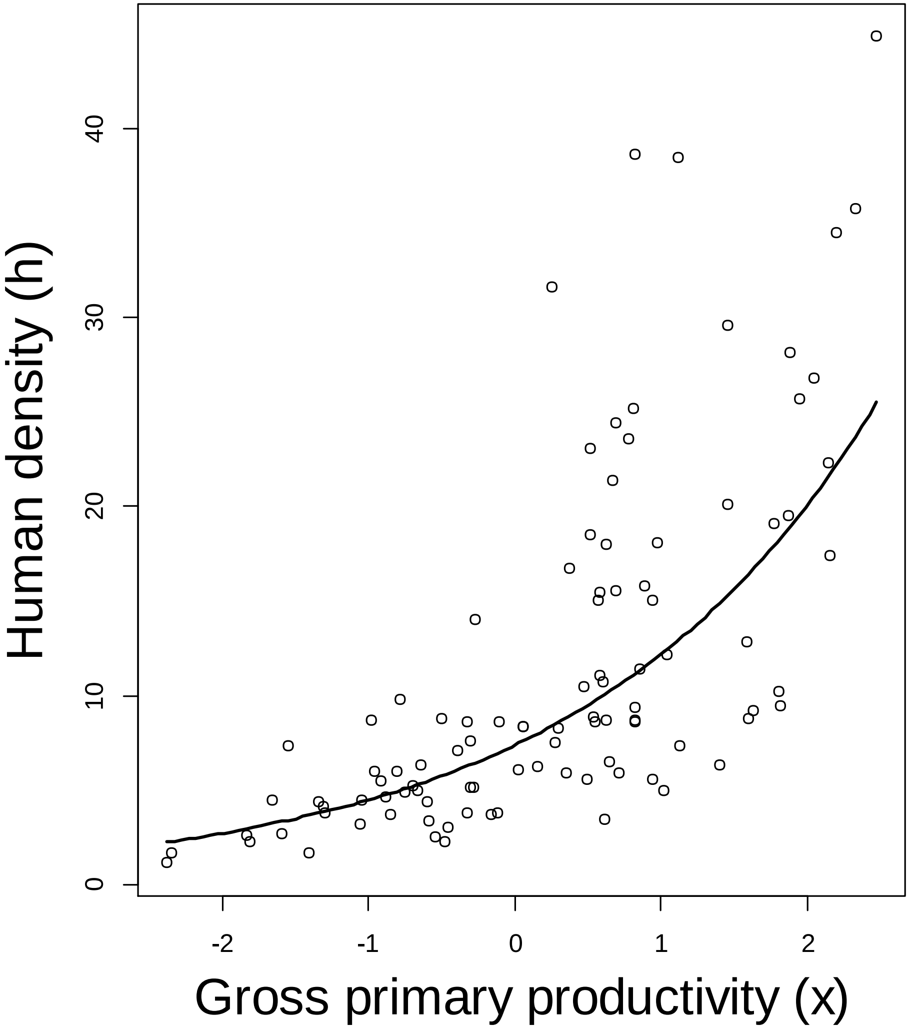


GPP (x)

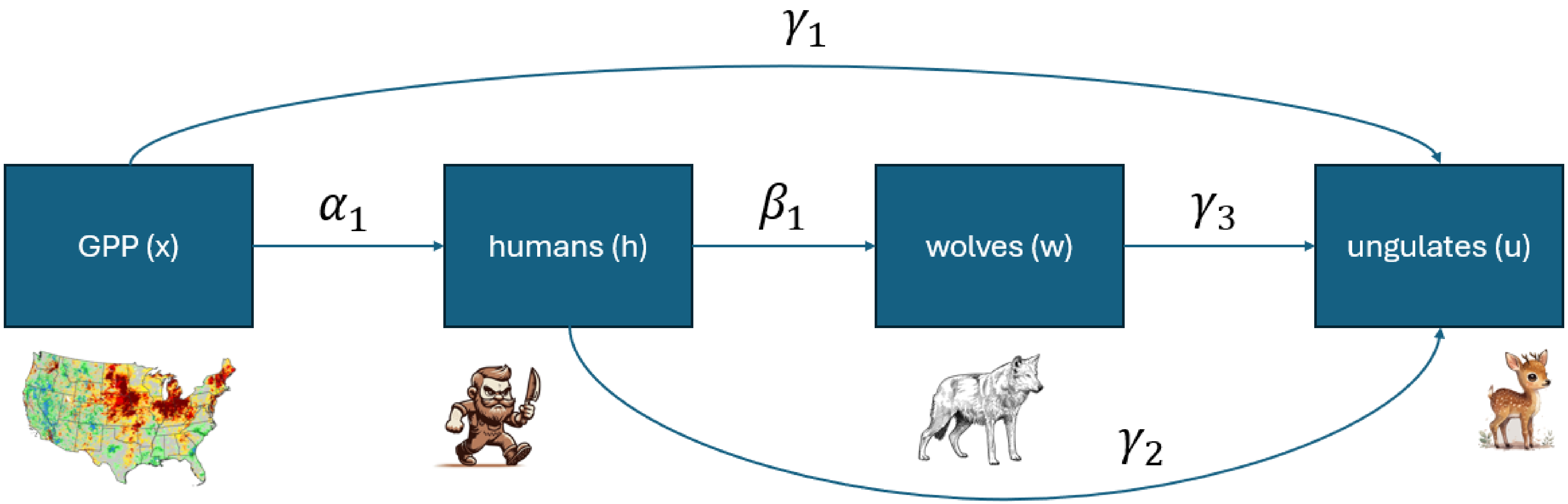
α_1

humans (h)





Do the presence of direct and indirect effects muddle things up?



γ_1

GPP (x)

α_1

humans (h)

β_1

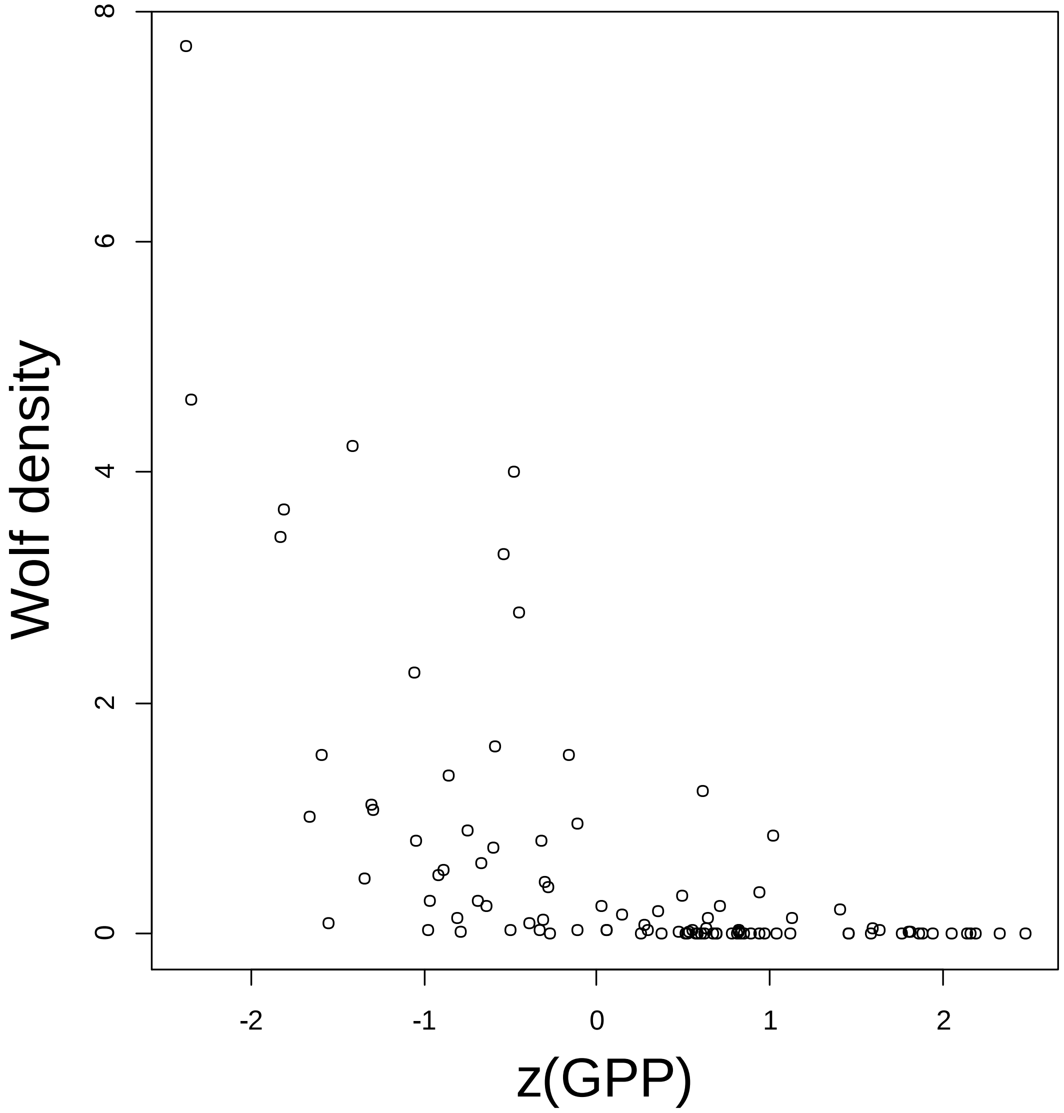
wolves (w)

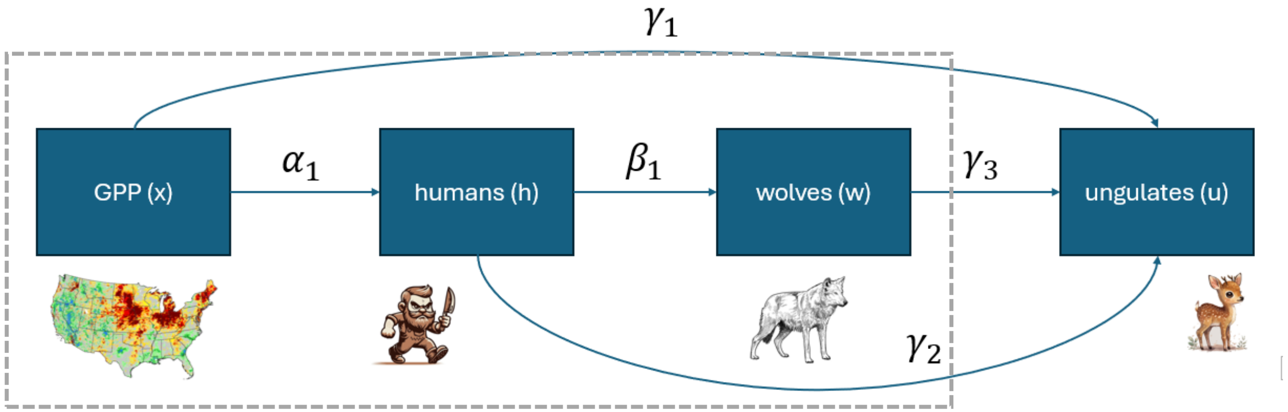
γ_3

ungulates (u)

γ_2

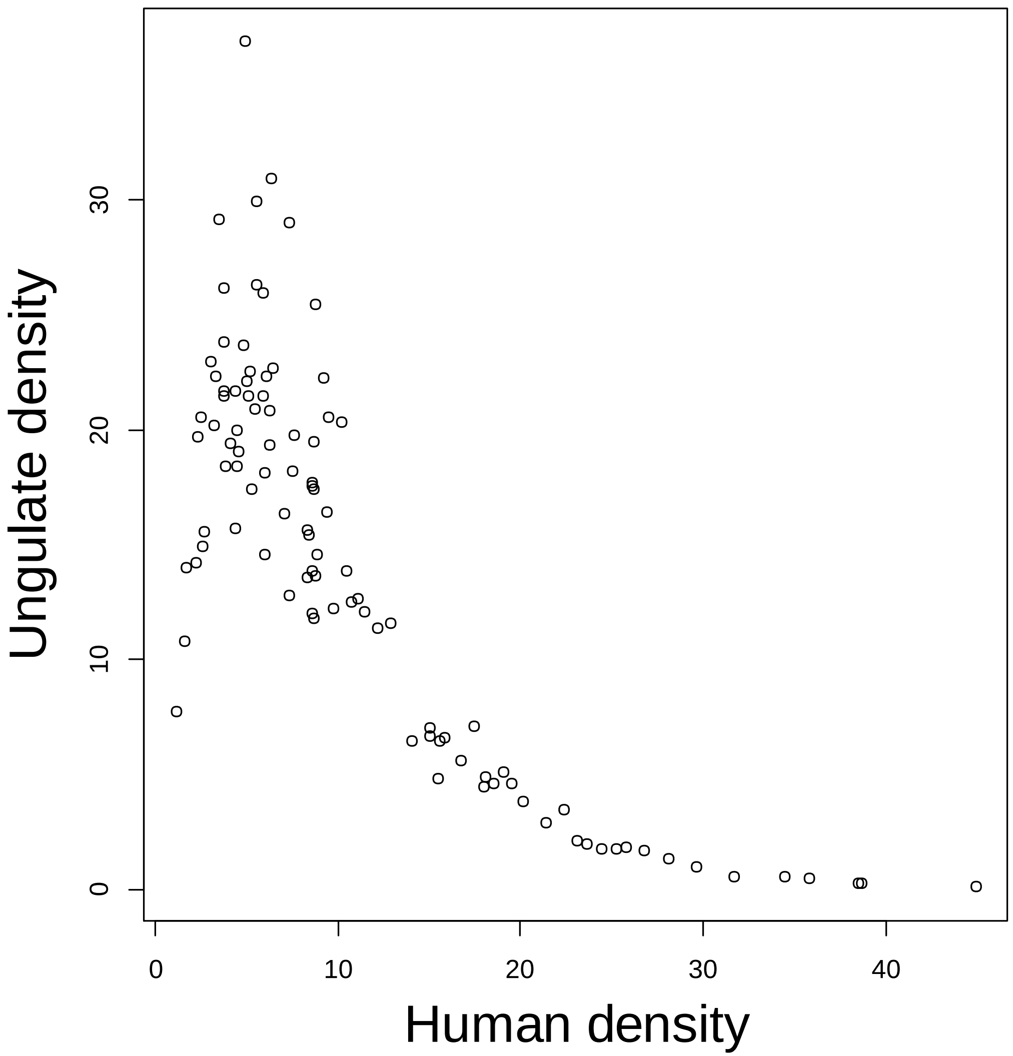
Here is the simulated relationship between wolves and GPP

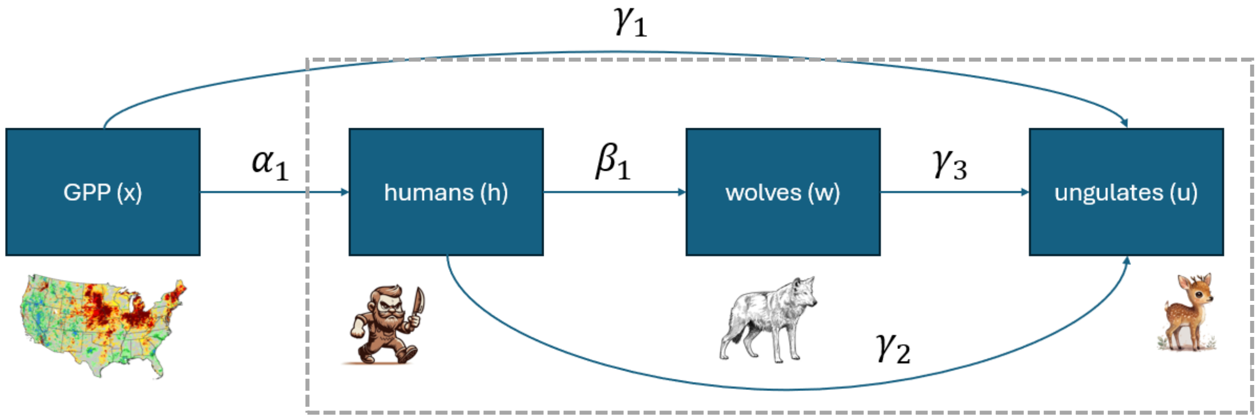




It's negative, not because GPP is bad for wolves, but b/c GPP means more humans. Humans are bad for wolves.

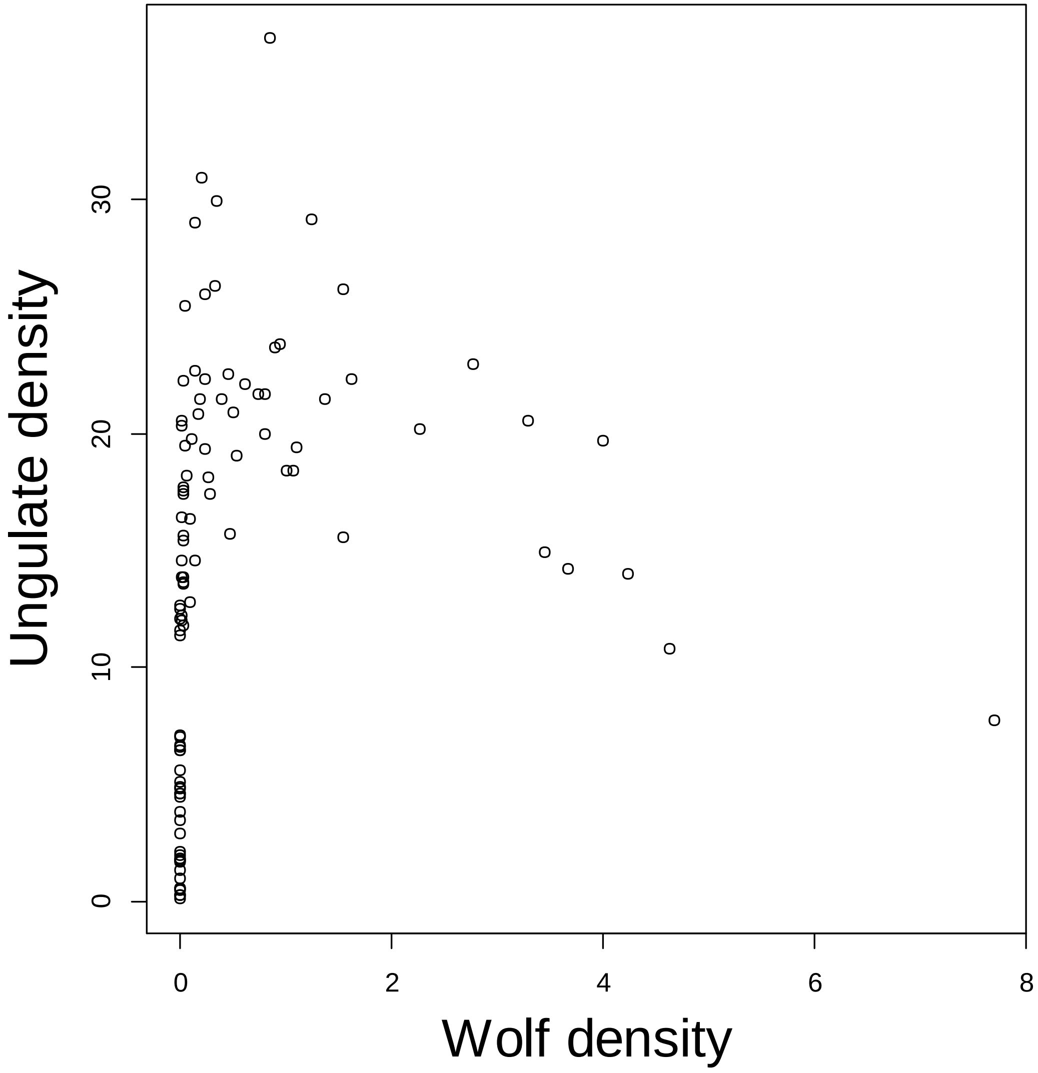
Here is the simulated relationship between ungulates and humans

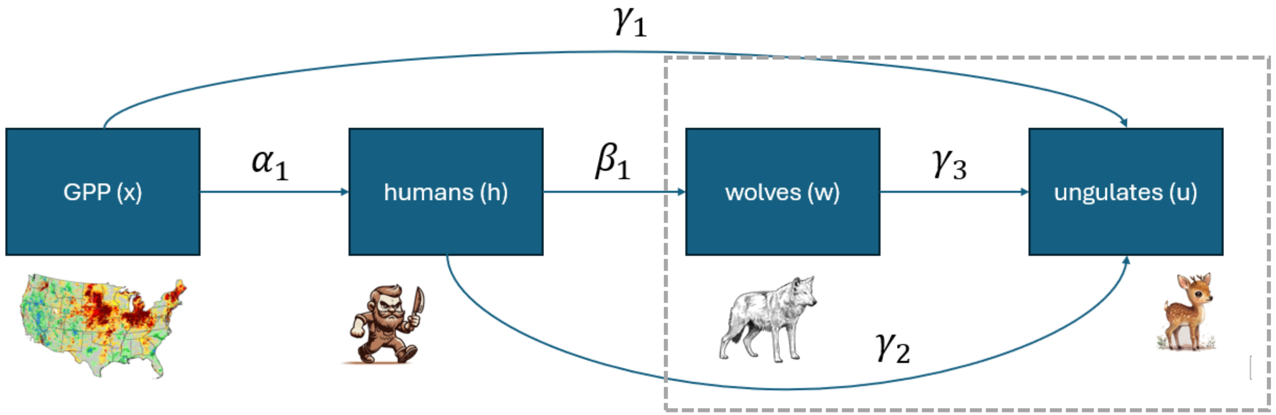




It's quadratic (peaks around 5, then declines). This is because initial human presence removes wolves (a key predator), but continued human presence has a major negative impact on deer. i.e., imagine ahuman density of ~5 corresponds to suburbs or Greenough Park, a human density of 30-40 would be more like downtown Missoula!

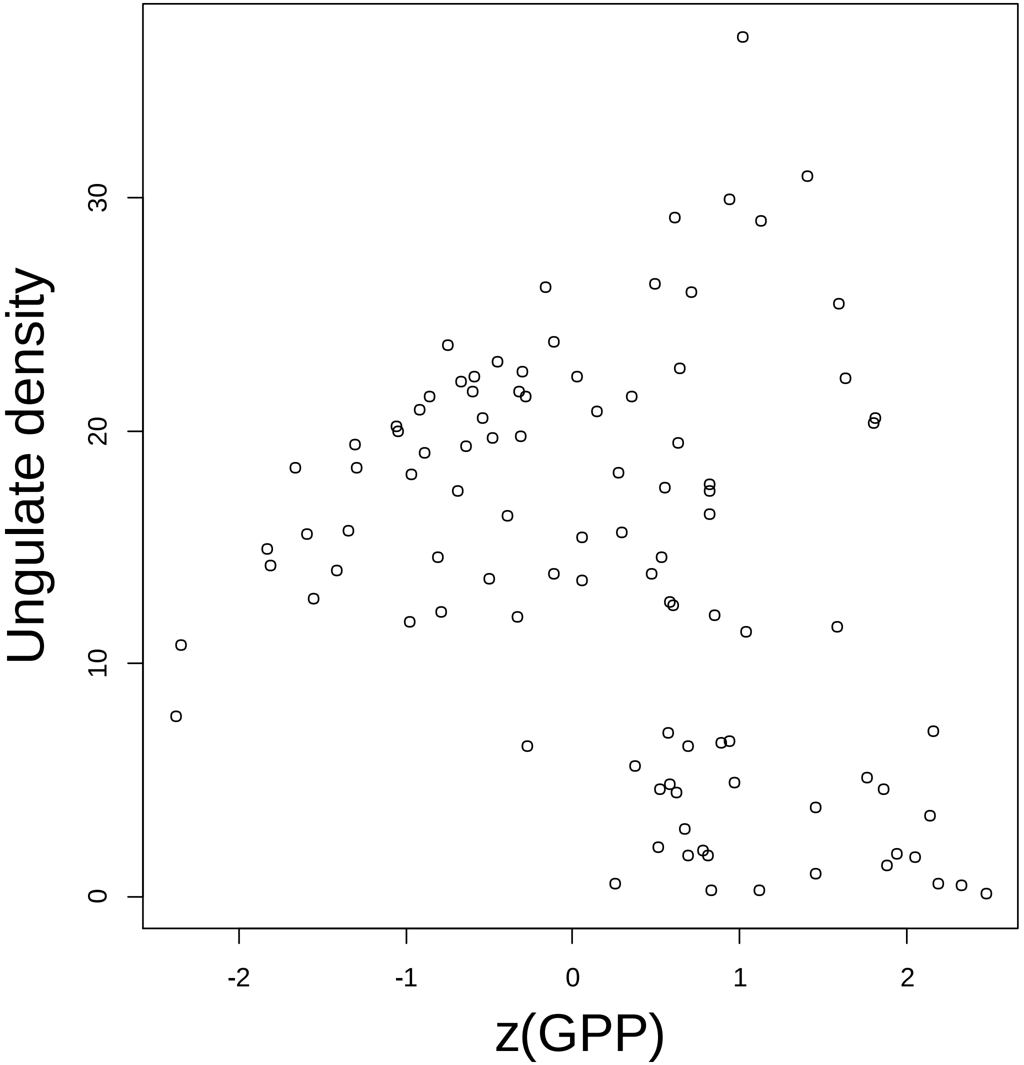
Here is the simulated relationship between ungulates and wolves

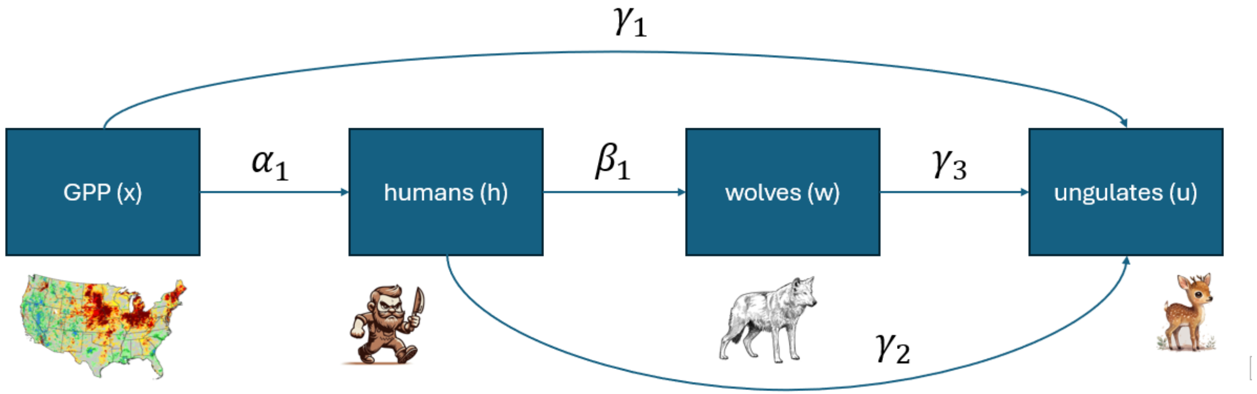




It's quadratic (hitting 0 at very low wolf densities because that's where lots of humans are) and declining at high wolf densities because wolves eat deer

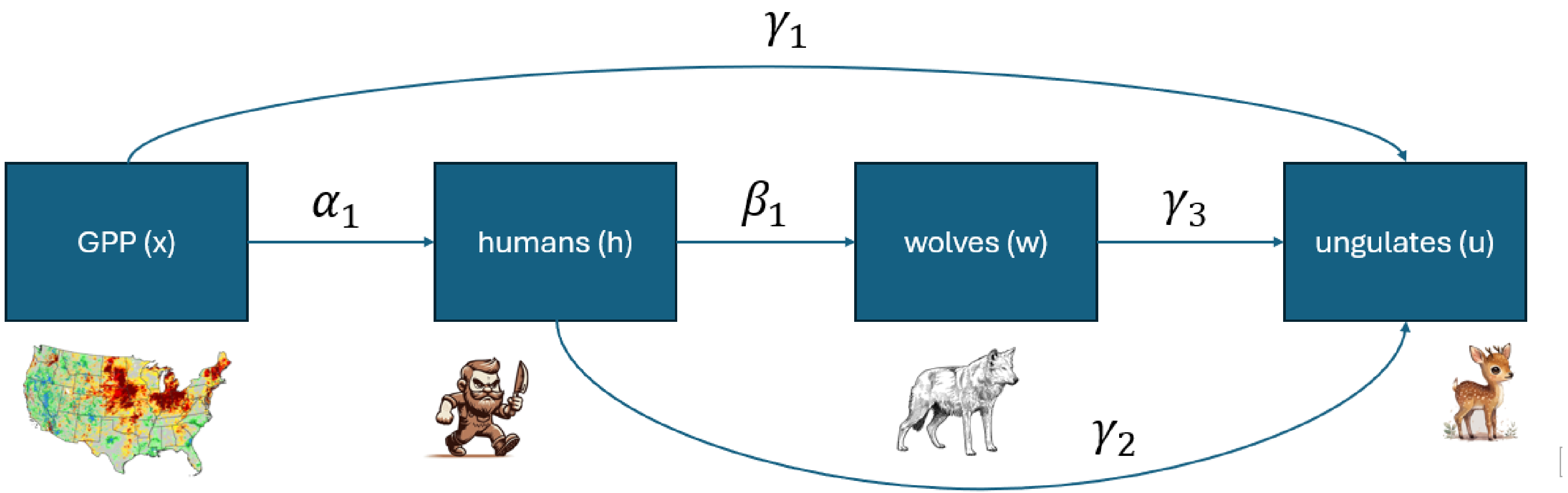
Here is the simulated relationship between ungulates and GPP





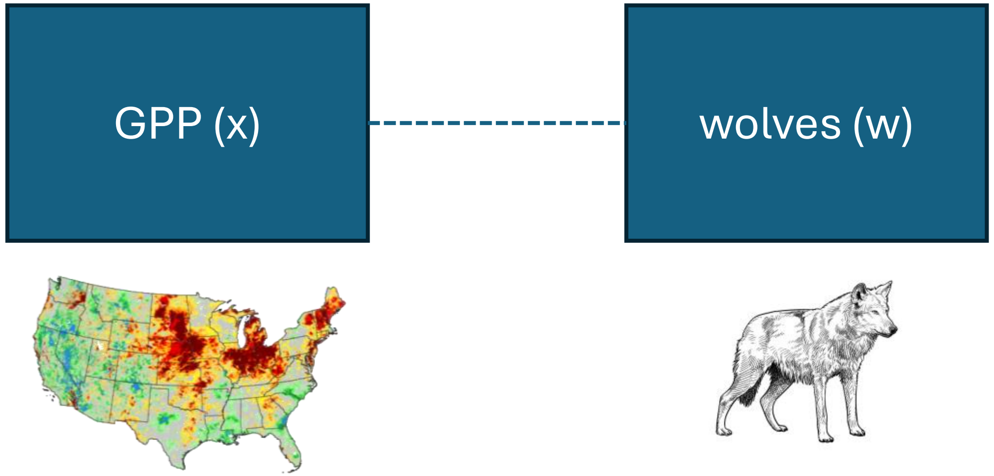
High GPP valleys have lots of people, and thus few ungulates. So the direct effect of GPP on ungulate density (γ_1) is positive, but the 'total' effect of GPP on ungulates is negative or quadratic because high GPP leads to lots of people.

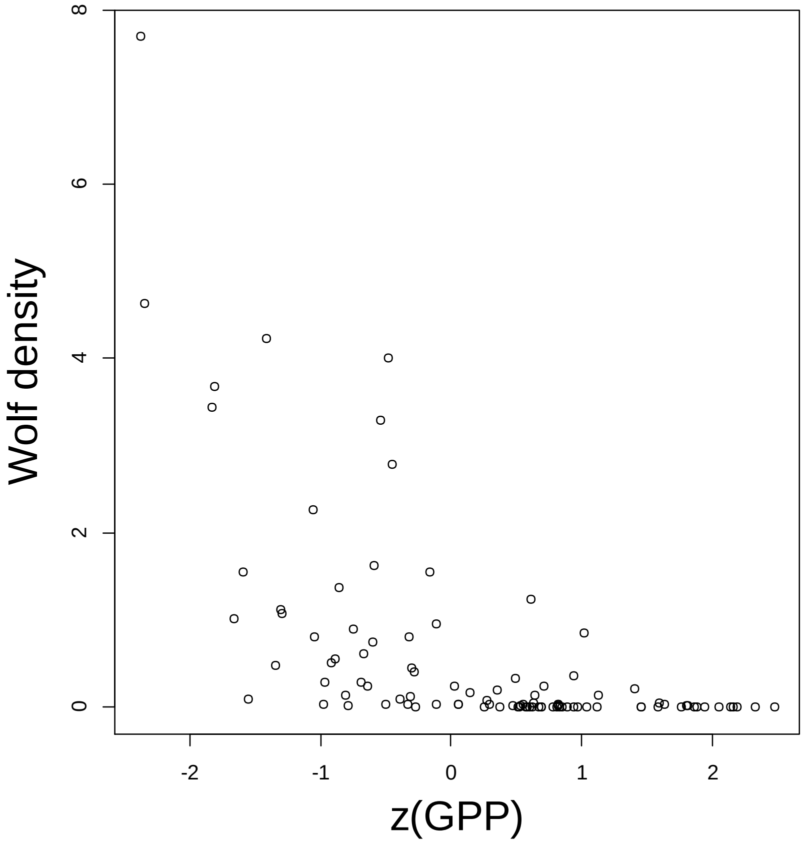
Take-home 1: complex paths produce complex signals



***Almost everything (worth studying) is the result of a complex path!**

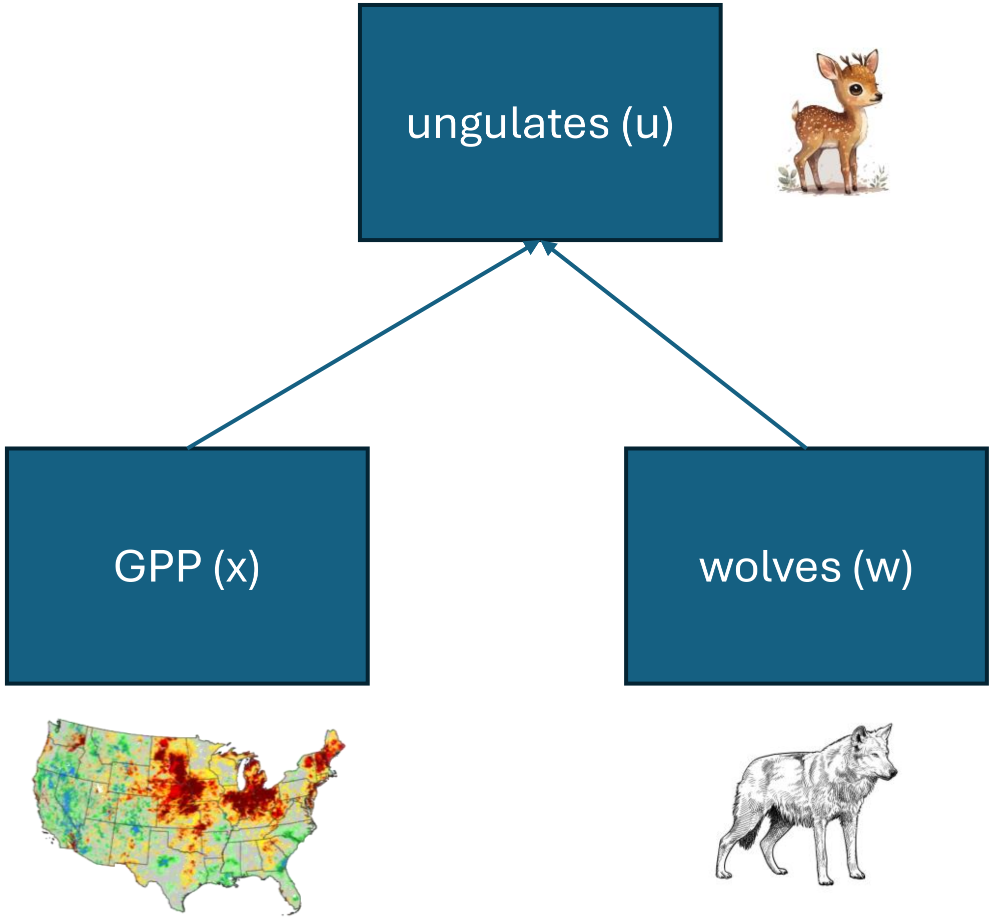
Take-home 2: univariate relationships can be wildly misleading!





This can create problems for inference (it’s predictive, but not causal)

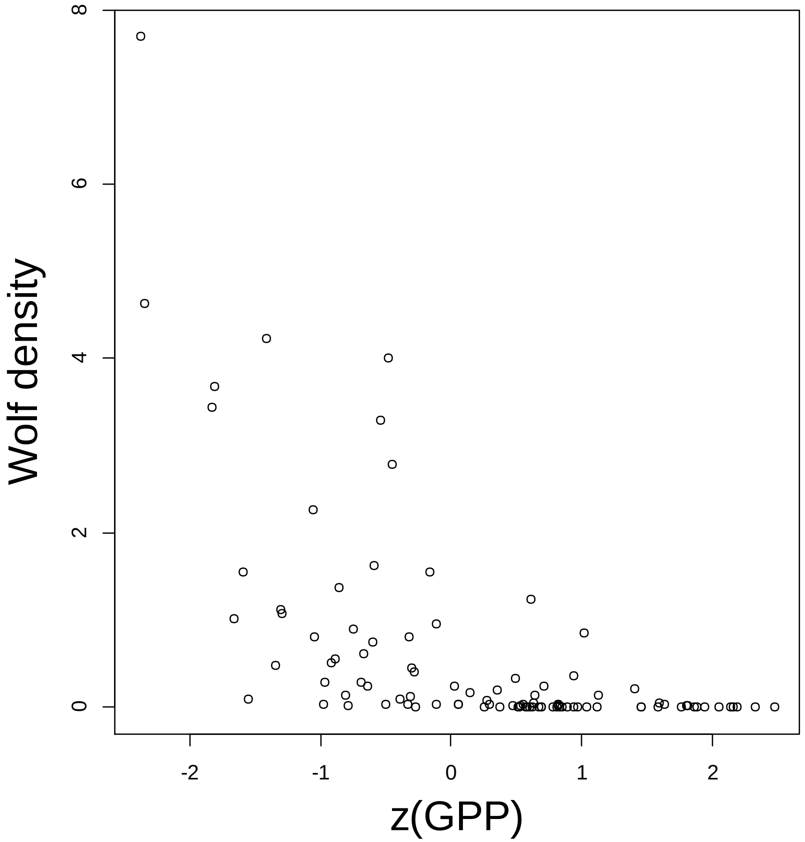
e.g., imagine we wanted to build this model



ungulates (u)

GPP (x)

wolves (w)



This can create problems for inference