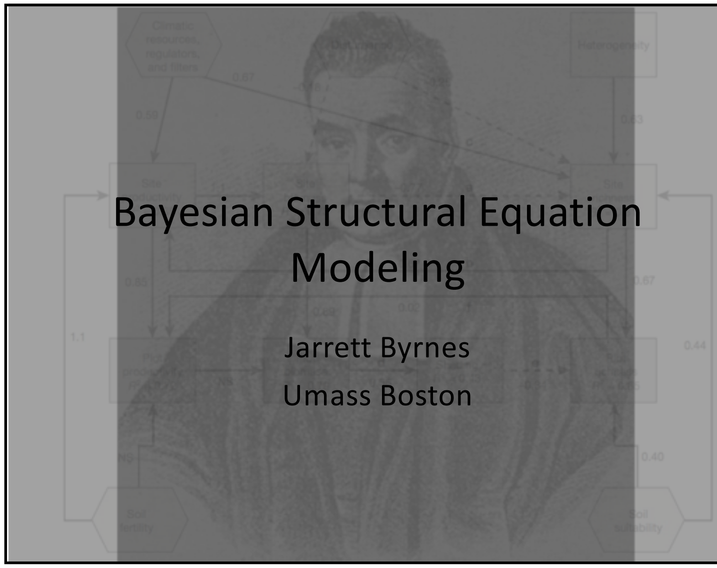
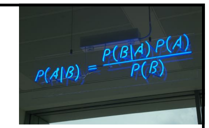


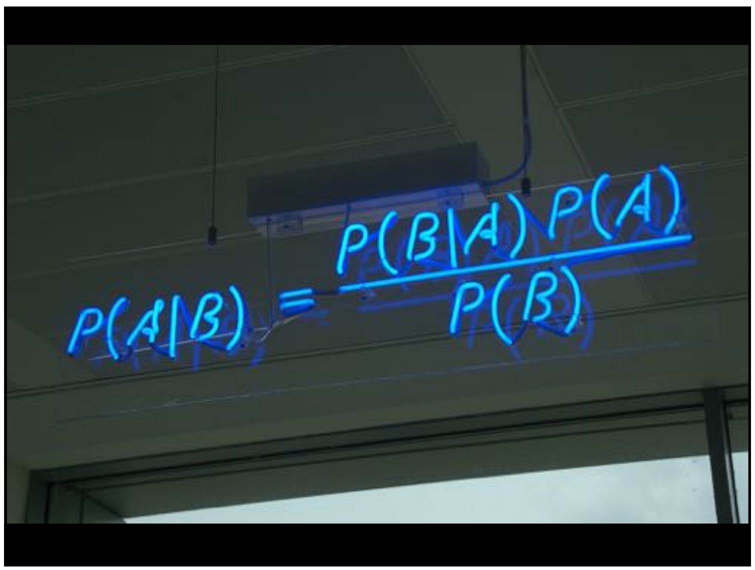
Bayesian Structural Equation Modeling

Jarrett Byrnes
Umass Boston

Why Bayes



- Estimate probability of a parameter
- State degree of belief in specific parameter values
- Evaluate probability of hypothesis given the data
- Incorporate prior knowledge
- Fit crazy complex models



Bayes Theorem and Data

$$p(H|D) = \frac{p(D|H)p(H)}{p(D)}$$

where p(H|D) is your posterior probability of a hypothesis

Credible Intervals

In Bayesian analyses, the 95% ***Credible Interval*** is the region in which we find 95% of the possible parameter values. The observed parameter is drawn from this distribution. For normally distributed parameters:

$$\beta - 2SD \leq \beta \leq \beta + 2SD$$

Considerations for Bayes and SEM

- This is yet another engine – but now we can use priors and credible intervals
- Makes error propagation for prediction easy
- Piecewise approach to local equation estimation
  - brms or directly in STAN or BUGS
- Although can use covariance estimation
  - blavaan

"There are no routine statistical questions, only questionable statistical routines"

- Sir David Cox

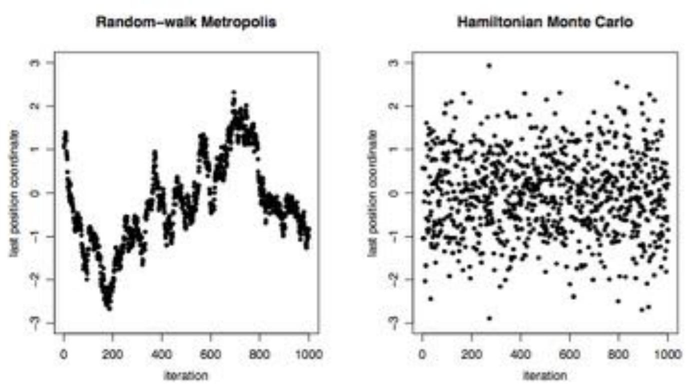
The brms library

- Uses lme4-like syntax
  - brm(y ~ x + 1|group, data = data)
- Calls STAN for HMC fitting
  - Has to compile program first
- Constantly changing and improving

King Markov and His Islands

King Hamilton and His Batboat

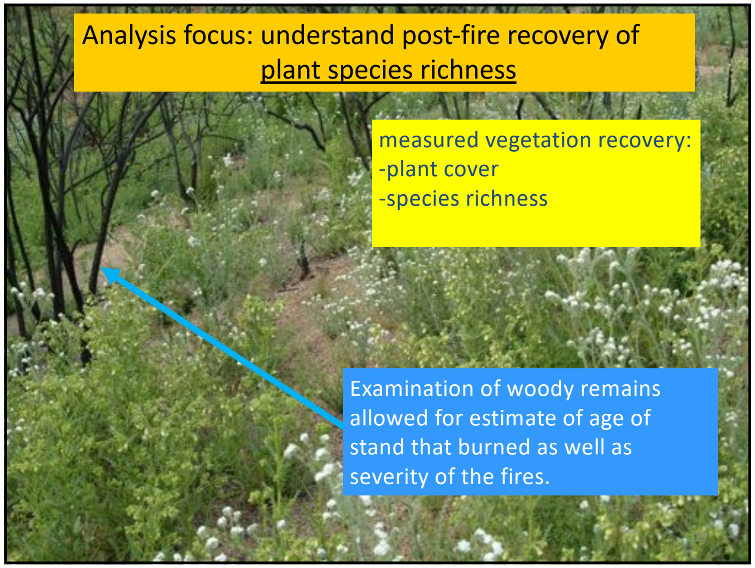
MCMC v. HMC

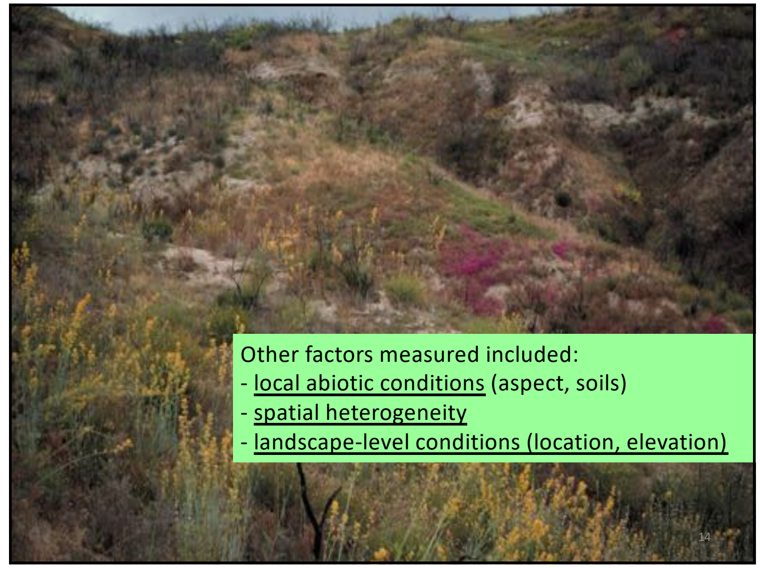


In particular, HMC works better for many correlated parameters – common in mixed models and SEMs

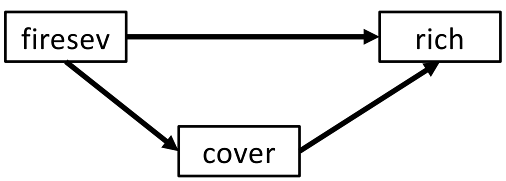
Mediation in Analysis of Post-Fire Recovery of Plant Communities in California Shrublands*

*Five year study of wildfires in Southern California in 1993. 90 plots (20 x 50m), (data from Jon Keeley et al.)



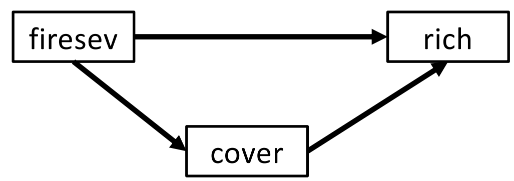


A Partially Mediated Model




```
#Load the data
keeley <- read.csv("../data/Keeley_rawdata_select4.csv")

#The model formulae
rich_mod <- bf(rich ~ firesev + cover)
cover_mod <- bf(cover ~ firesev)
```

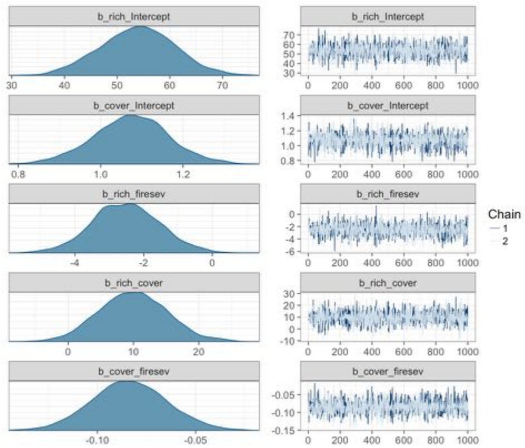
A Partially Mediated Model



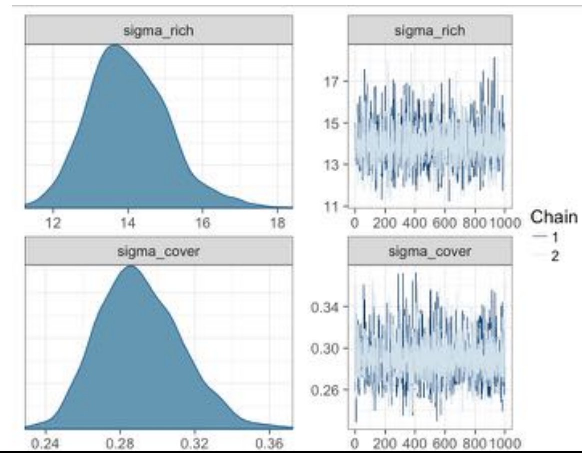

```
k_fit_brms <- brm(rich_mod +
                    cover_mod +
                    set_rescor(FALSE),
                  data=keeley,
                  cores=4, chains = 2)
```

setRescor would allow all endogenous variables to covary if TRUE

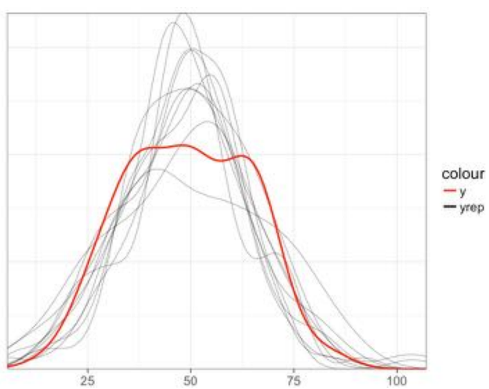
Did it Converge (plot)?



Did it Converge (plot)?



Check Alignment of Posterior with Data




```
pp_check(k_fit_brms, resp="rich") +
  scale_color_manual(values=c("red", "black"))
```

Coefficients from Summary

```
Population-Level Effects:
                Estimate Est.Error l-95% CI u-95% CI Eff.Sample Rhat
rich_Intercept     53.86      7.08    39.73    68.06       2000 1.00
cover_Intercept     1.07      0.09     0.89     1.25       2000 1.00
rich_firesev       -2.52      0.99    -4.44    -0.55       2000 1.00
rich_cover          9.95      5.26    -0.10    20.67       2000 1.00
cover_firesev      -0.08      0.02    -0.12    -0.05       2000 1.00
```

Checking Model Fit in a Bayesian Context

- COULD get posterior p-values and do something like a Fisher's C
  - Untested as of yet… and philosophically odd
- More in line would be testing with WAIC

Widely Applicable Information Criterion

- Like AIC, but, well, Bayesian

WAIC = -2 log likelihood predictive density + 2 effective number of parameters

$$WAIC = -2\ llpd + 2p_{waic}$$

$$= -2\ \Sigma\ \log Pr(y_i|\theta) + 2\ \Sigma\ var(\log Pr(y_i|\theta))$$

WAIC and SEM

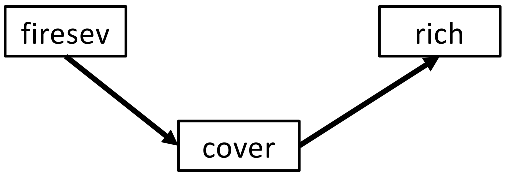
- Each component model has its own WAIC
- We can sum the WAICs to get a modelwide WAIC

$$WAIC_{model} = \Sigma\ WAIC_i$$

Additive WAIC in Action

```
rich_fit <- brm(rich_mod,
                data=keeley,
                cores=2, chains = 2)
cover_fit <- brm(cover_mod,
                 data=keeley,
                 cores=2, chains = 2)

> WAIC(k_fit_brms)
   WAIC    SE
 768.75 15.59

> WAIC(rich_fit)
   WAIC   SE
 734.17 9.51

> WAIC(cover_fit)
  WAIC    SE
 34.64 12.57
```

A Fully Mediated Model for Comparison




```
# fit the fully mediated model
rich_mod_fullmed <- bf(rich ~  cover)

fit_brms_fullmed <- brm(rich_mod_fullmed +
                          cover_mod +
                          set_rescor(FALSE),
                        data=keeley,
                        cores=4, chains = 2)
```

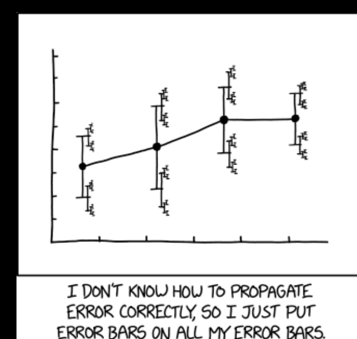
Model Comparison

```
> WAIC(k_fit_brms, fit_brms_fullmed)
                                  WAIC    SE
k_fit_brms                      768.75 15.59

fit_brms_fullmed                772.67 16.88

k_fit_brms - fit_brms_fullmed    -3.92  5.42
```

Models are not different
Parsimony suggest full mediation model

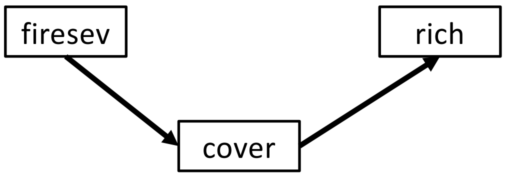
Prediction with Error Propagation

1. Create a new data frame with exogenous variables
2. Calculate posterior simulations of most proximate endogenous variables (those with only exogenous predictors)
3. Use simulated values to calculate next set of endogenous variables
   - Take diag of prediction matrix to keep nsims constant
4. Rinse and repeat…



xkcd

What do we Propagate?




```
Population-Level Effects:
                Estimate Est.Error l-95% CI u-95% CI Eff.Sample Rhat
rich_Intercept     53.86      7.08    39.73    68.06       2000 1.00
cover_Intercept     1.07      0.09     0.89     1.25       2000 1.00
rich_firesev       -2.52      0.99    -4.44    -0.55       2000 1.00
rich_cover          9.95      5.26    -0.10    20.67       2000 1.00
cover_firesev      -0.08      0.02    -0.12    -0.05       2000 1.00
```

The First Posterior Fitted Values

```
#First, make new data
follow_this <- data.frame(firesev = 5)

#Get fitted sims of new data
cover_fit <- fitted(k_fit_brms,
                   newdata=follow_this,
                    resp = "cover",
                    nsamples = 1000,
                    summary = FALSE)
```

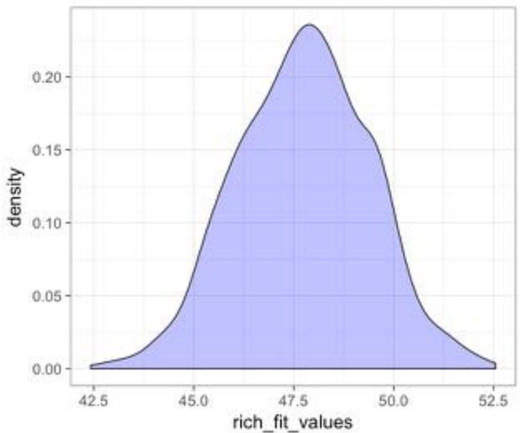
New Data

```
follow_this_fit <-
     expand.grid(firesev = follow_this$firesev,
                 cover = as.vector(cover_fit))
```

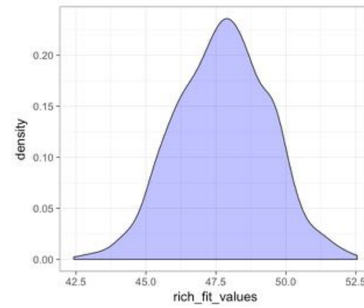
The Second Endogenous Prediction

```
#second fit
rich_fit_values <- fitted(k_fit_brms,
                    newdata=follow_this_fit,
                    resp = "rich",
                   nsamples = 1000,
                    summary = FALSE)

#remove excess simulations
#(from a 1000 x 1000 matrix)
rich_fit_values <- diag(rich_fit_values)
```

Our Posterior Simulation

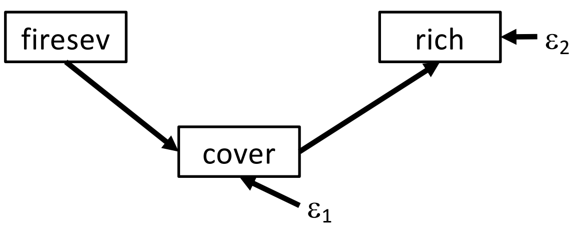


Our Posterior Simulation




```
> median(rich_fit_values)
[1] 47.80541

> posterior_interval(as.matrix(rich_fit_values))
           5%      95%
[1,] 45.13998 50.3635
```

But What About Residuals?

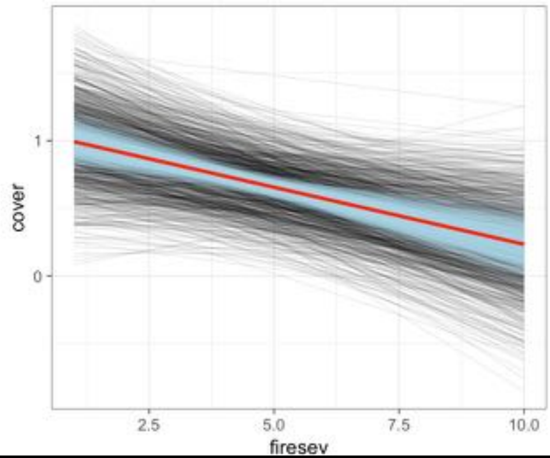


- Prediction error: we propagate coefficient variation, residual variation, and variation in our residual variation
- We use the full posterior!

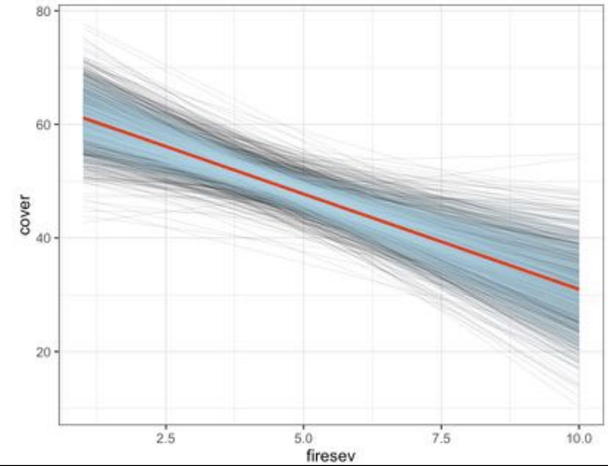
`predict` versus `fitted`

- fitted gives simulations using only variance in coefficient values
  - Allows for exploration of how an exogenous data generating process propagates through a network
- predict incorporates residual variability
  - True prediction credible intervals
  - Effects of variability magnify through a network
- Both are affected by smaller sample sizes, particularly as you move through a network

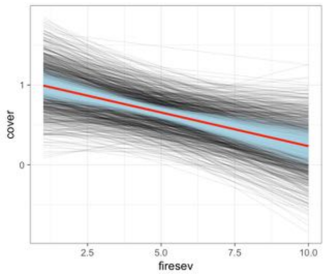
Prediction v. Fit Credible Intervals

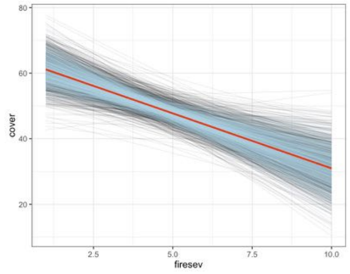


Letting Results Propagate Through



Note how the Intervals Spread Out





Same plot formatting – fainter = flatter overall density

Final Thoughts on Bayes

- Scientists often use Bayesian logic in their writing, even if their statistics are frequentist
- Bayesian SEM is merely another ***engine*** to drive the SEM ***framework***
- Are merely another implementation of piecewise methodologies, with more flexibility
- Bayesian methods enable natural propagation of error in predictions via draws from the posterior – makes life easy!


BAYES IS MAGICKS?!
(heavy breathing)