Recommendations for Potential Analyses

Bart DiFiore

2024-04-09

Hi all, I've had a chance to dig into the data a bit and wanted to outline some of my recommendations and potential paths forward. As we discussed, we are very limited by the size of the dataset. Therefore, we will need to be **cautious** with any inferences that we draw. However, I do think there are some potentially interesting avenues we could pursue. Some of these choices may not be what you originally had planned for the data set. As an analysist on this project, I'm happy to proceed in whatever manner you choose.

Here I'll start by outlining some of the limitations, and then outline a few potential solutions.

Data Limitations

Right now the data set has 37 observations. In itself that isn't a problem. However there are currently over 120 potential predictor variables in the data set. I fully understand that you don't want to utilize all predictors. However, the simplified predictor variables document that Scott sent along outlines an interest in two crossed categorical predictors and nine continuous predictors. In a simple multiple-linear regression with no interactions that would mean estimating 13 parameters, which is doable with 37 observations. However there were multiple observations collected at 8 of the 30 sites. This would suggest a random effect of site. However, including a random effect of site would mean estimating an additional 30 parameters (e.g. a different random intercept for each site), which means 43 parameters total. And that would far exceed what we can extract from the data (n = 37).

An additional complication is that many of the response and predictor variables that Scott included in the simplified variables doc include NA's. For instance, total trout abundance was only estimated at 34 of the 37 sites (and similarly the estimates for trout within size classes). Similarly, the variable *pct_cover* is only estimated at 24 of the 37 sites. Including that variable immediately reduces the sample size of our model matrix to 24. So one of the requirements is that once we settle on the unit of replication (e.g. are we going to use the 2016 data???), we only include predictor variables and response variables (see below) that are estimated at each site.

In terms of the response variables, I would advise that we focus on trout presence/absence as a binary response. While it would be interesting to model trout abundance / density we have two issues:

1. How do we deal with the dimensionality issues that we have discussed? e.g. is it abundance per m2 or m3? How do we account for different sampling effort? (e.g. similar to issues of standardizing CPUE in the fisheries lit)
2. With 37 points the highly overdispersed and zero inflated distribution of the response is going to require more complex models in which more parameters must be estimated. Fig. 1 is a modified histogram of total trout abundance. Based on this data distribution we would need to use a zero-inflated model with a negative binomial distribution–both of which require additional parameters.

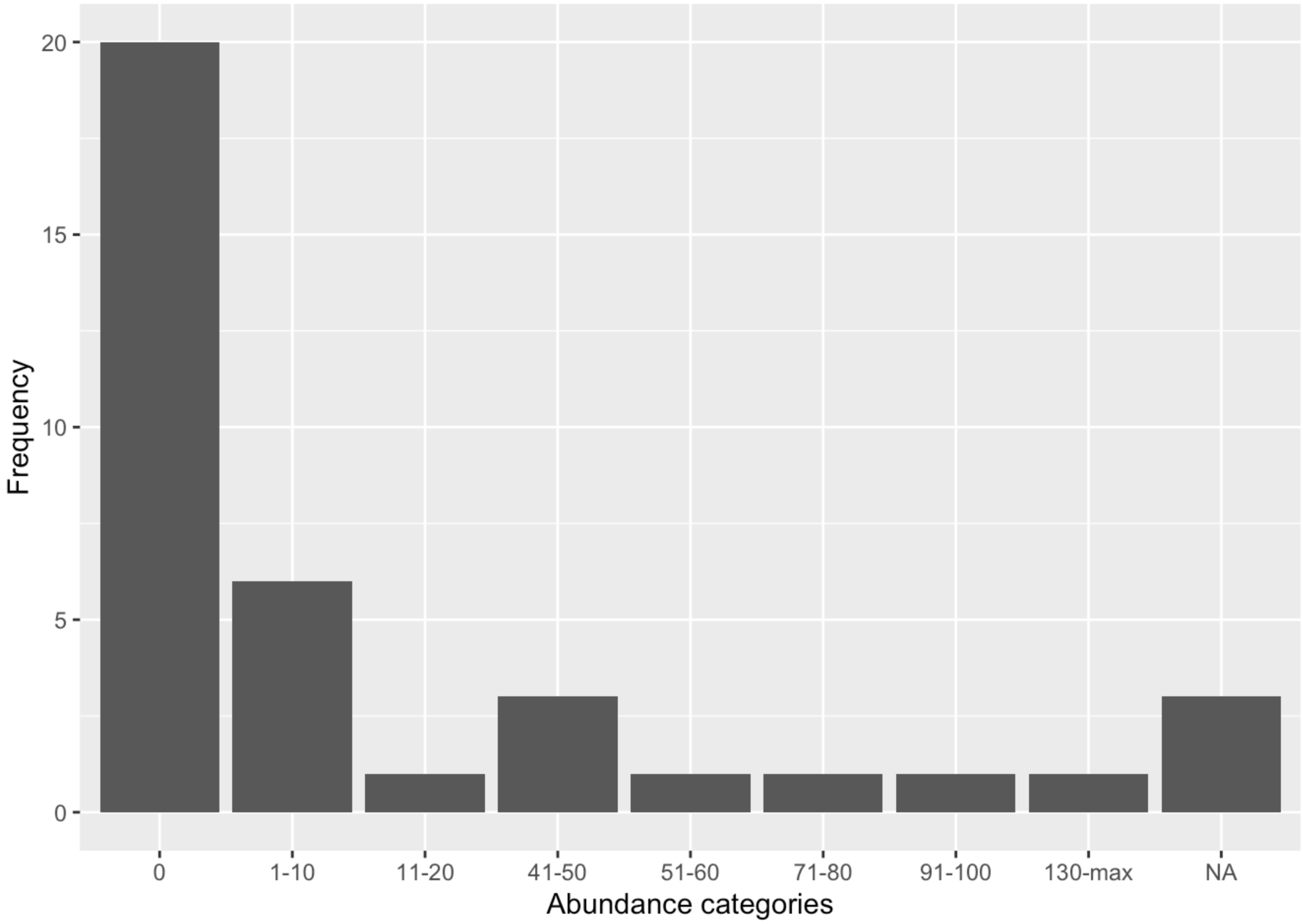


Figure 1. Modified histogram of total trout abundance. Height represents the number of observations of trout in each abundance category.

Therefore, I would suggest that for the time being we model trout presence/absence as we did in the recent paper Cooper et al. 2024 (https://onlinelibrary.wiley.com/doi/full/10.1111/fwb.14212).

Potential solutions

So I'll quickly list the paths that I see moving forward from simplest to more complex. Then I'll dig into each of them.

1. Model trout presence/absence as a function of key predictors that are *not* strongly collinear (≤ 5 predictor variables) using a simple glm framework. (I think this one is the least interesting)

2. Build an even simpler SEM that the one that Scott depicted in the simplified predictor document, containing (≤ 7) *continuous* predictors. (I think this one could get at *some* of the questions you were hoping to address)

3. Use a latent-state variable approach to estimate a "stream condition index" for trout presence. (I think this is the most interesting approach, although somewhat of a departure from what you had planned)

Option 1: Simple glm()

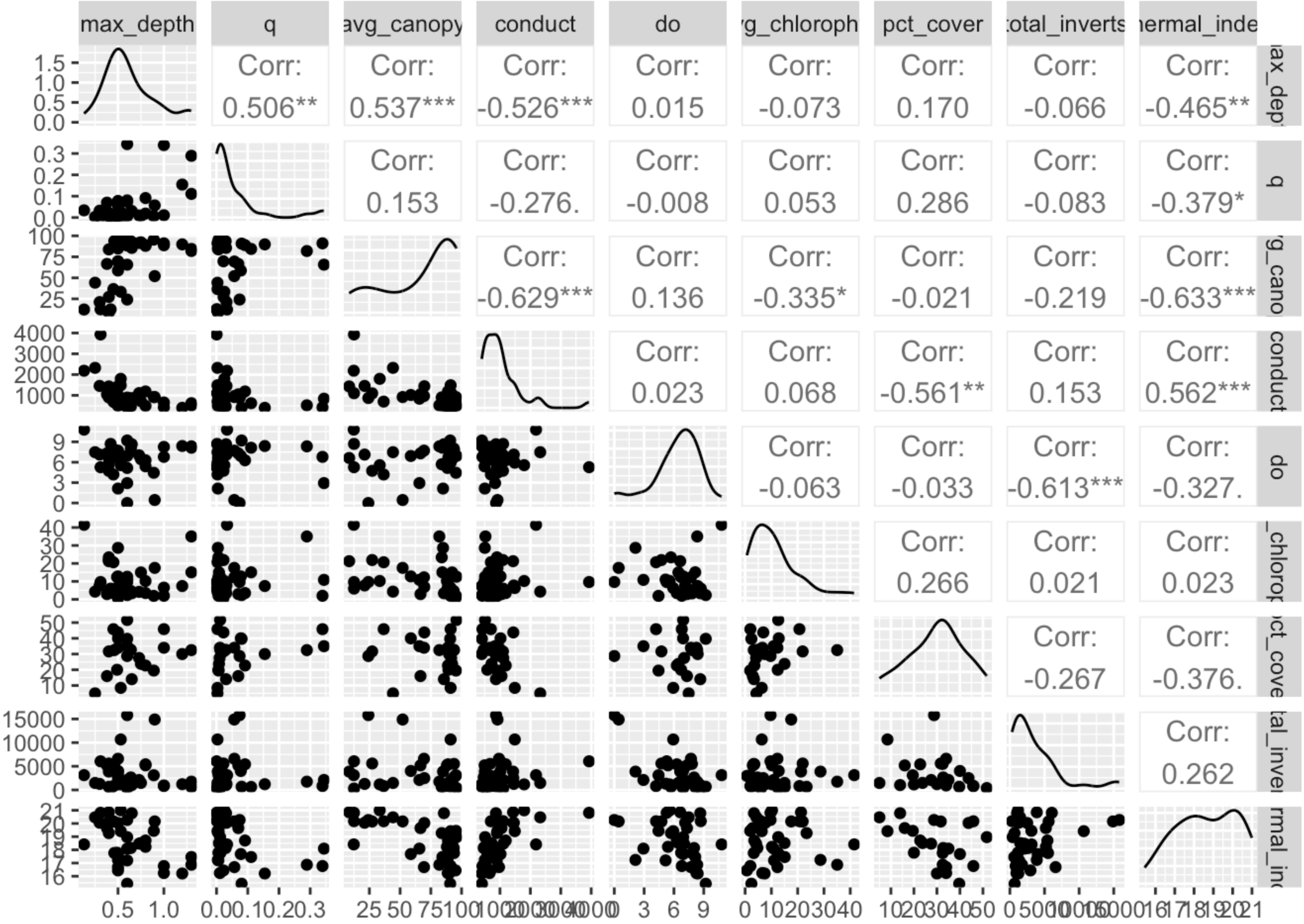


Figure 2. Pairs plot and correlations between predictor variables included in Scott's simplified variable document.

Figure 2 demonstrates the potential collinearity between predictors in a simple glm() framework. In itself this isn't a problem unless there is multicollinearity in the residuals. Lets build out the model and test for that using the variance inflation factor (vif).

VIF's should be less than at least 5 to claim that collinearity isn't an issue. So this model is certainly not specified correctly if our intent is the separate the effects of each predictor on the response. Let's try and transforming some of the predictors. Occasionally, this can reduce collinearity.

```
##                                vif
## scale(max_depth)          43.08333
## scale(q_log)              25.03370
## scale(avg_canopy_logit) 100.49271
## scale(conduct_log)        70.36451
## scale(do)                191.76283
## scale(pct_cover)          74.48260
## scale(thermal_index)      32.38674
## scale(avg_chlorophyll)    52.08025
## scale(total_inverts)     100.24006
```

Doesn't seem to do anything to the multicollinearity issue…

Let's try a different model that may get at some of your questions but with different predictors. Lets pick one environmental variable, lets try conductivity as it is an index of stream quality.

```
## 
## Call:
## glm(formula = trout ~ scale(conduct_log) + scale(avg_chlorophyll) + 
##     scale(total_inverts), family = "binomial", data = df_mod2)
## 
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)   
## (Intercept)            -0.19562    0.46825  -0.418  0.67612   
## scale(conduct_log)     -2.25711    0.78729  -2.867  0.00414 **
## scale(avg_chlorophyll)  0.09555    0.55335   0.173  0.86290   
## scale(total_inverts)   -0.58358    0.53290  -1.095  0.27348   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 51.266  on 36  degrees of freedom
## Residual deviance: 29.677  on 33  degrees of freedom
## AIC: 37.677
## 
## Number of Fisher Scoring iterations: 5
```

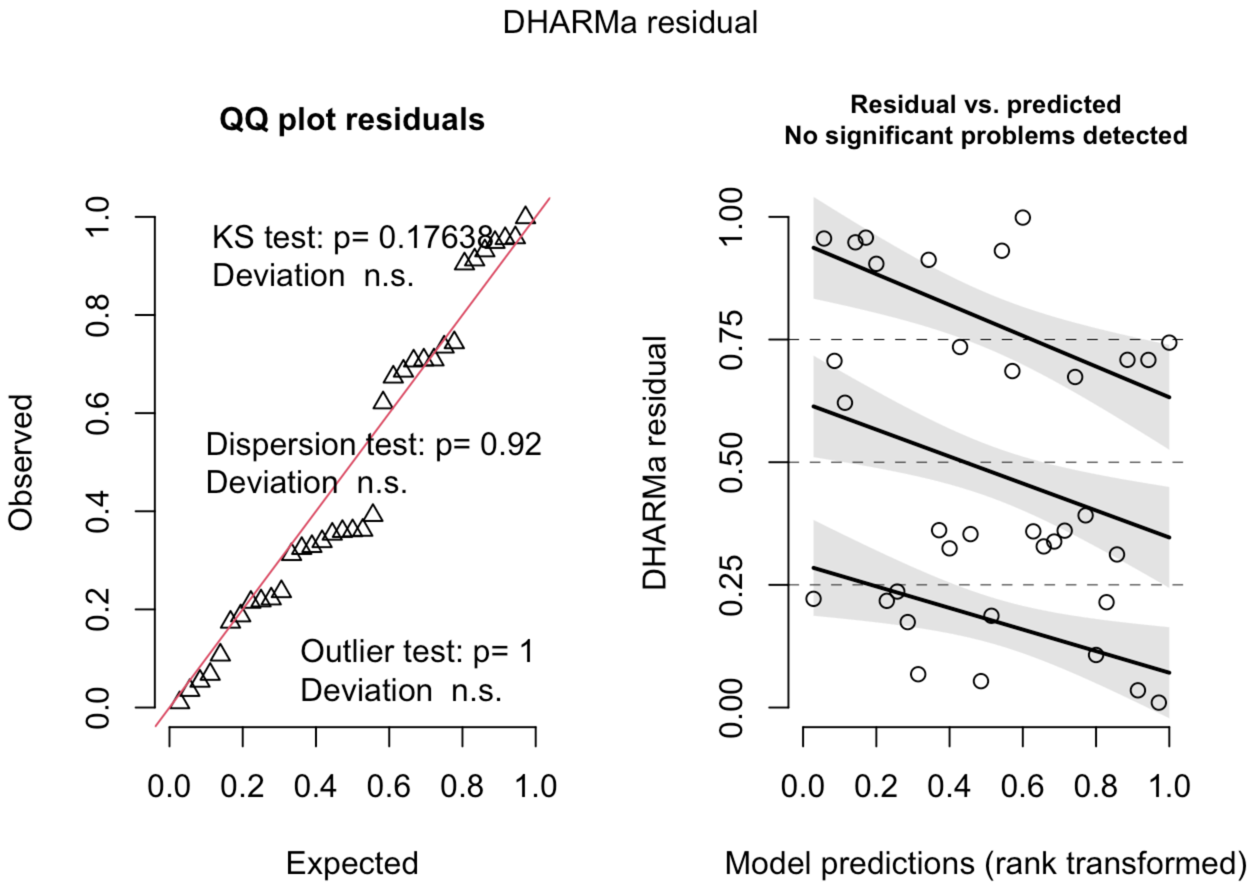
```
##     scale(conduct_log) scale(avg_chlorophyll)   scale(total_inverts) 
##               1.015214               1.001620               1.013606
```

This model doesn’t have issues with collinearity, but it isn’t wholly that informative. It suggests that conductivity is important to the probability of trout presence. And that there is little indication that algae (avg_chlorophyll) or inverts influence the probability of trout presence. We can plot the effect of conductivity on trout presence:

We might be having issues because the pct_cover variables is really reducing our sample size down to 24 sites. Lets cut that and try to rerun some of these models.

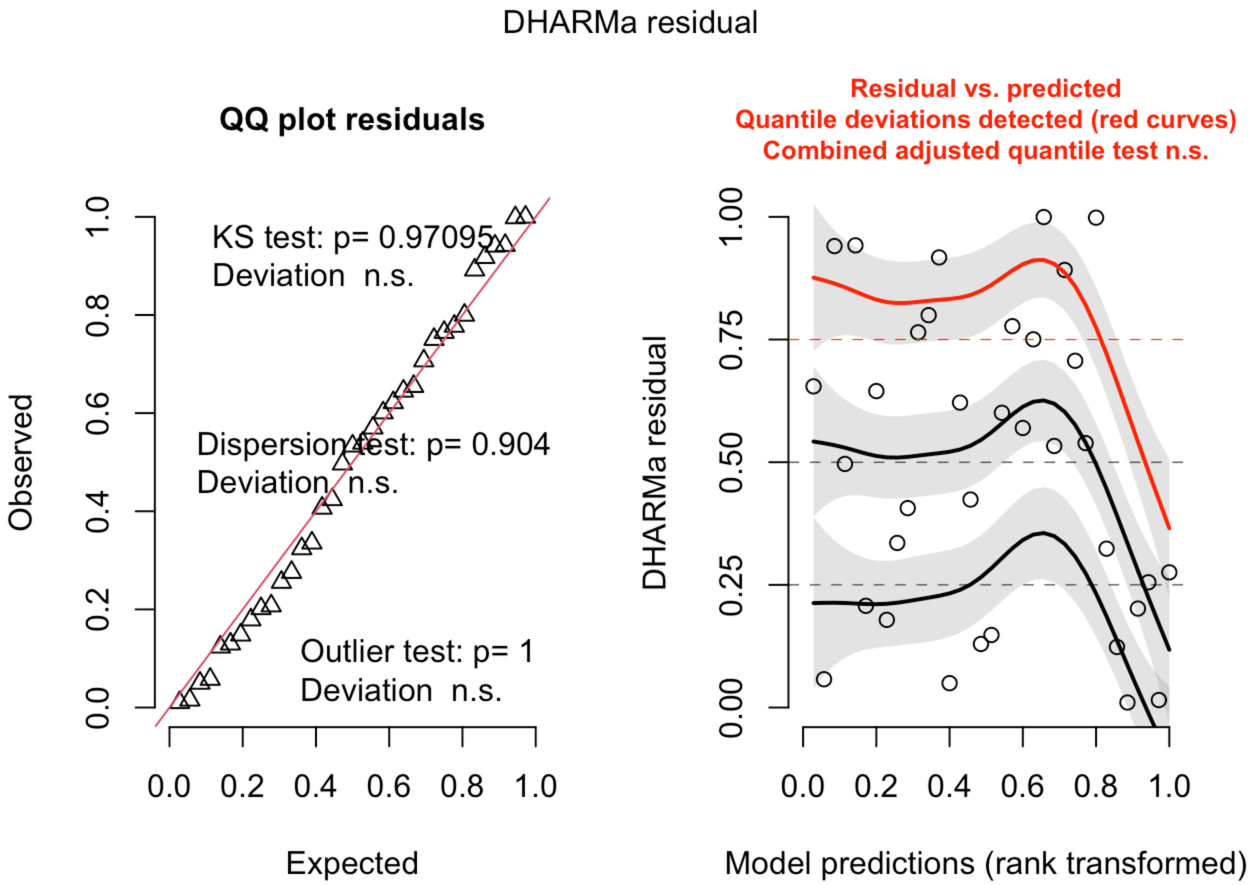
```
##
## Call:
## glm(formula = trout ~ max_depth + q + avg_canopy + conduct +
##     do + thermal_index + avg_chlorophyll + total_inverts, family = "binomial",
##     data = df_mod)
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -4.948e+00  1.358e+01  -0.365   0.7155
## max_depth       -4.126e-01  3.051e+00  -0.135   0.8924
## q                4.797e-02  6.869e+00   0.007   0.9944
## avg_canopy       3.844e-02  3.323e-02   1.157   0.2473
## conduct         -4.955e-03  2.787e-03  -1.778   0.0755 .
## do               3.590e-01  3.648e-01   0.984   0.3251
## thermal_index    2.386e-01  5.647e-01   0.423   0.6726
## avg_chlorophyll  5.337e-02  9.679e-02   0.551   0.5814
## total_inverts   -4.616e-05  2.124e-04  -0.217   0.8280
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 48.492  on 34  degrees of freedom
## Residual deviance: 23.187  on 26  degrees of freedom
##   (2 observations deleted due to missingness)
## AIC: 41.187
##
## Number of Fisher Scoring iterations: 7
```

```
##                      vif
## max_depth       1.576816
## q               1.361337
## avg_canopy      2.111985
## conduct         1.665618
## do              1.784249
## thermal_index   1.931275
## avg_chlorophyll 1.936654
## total_inverts   1.335349
```

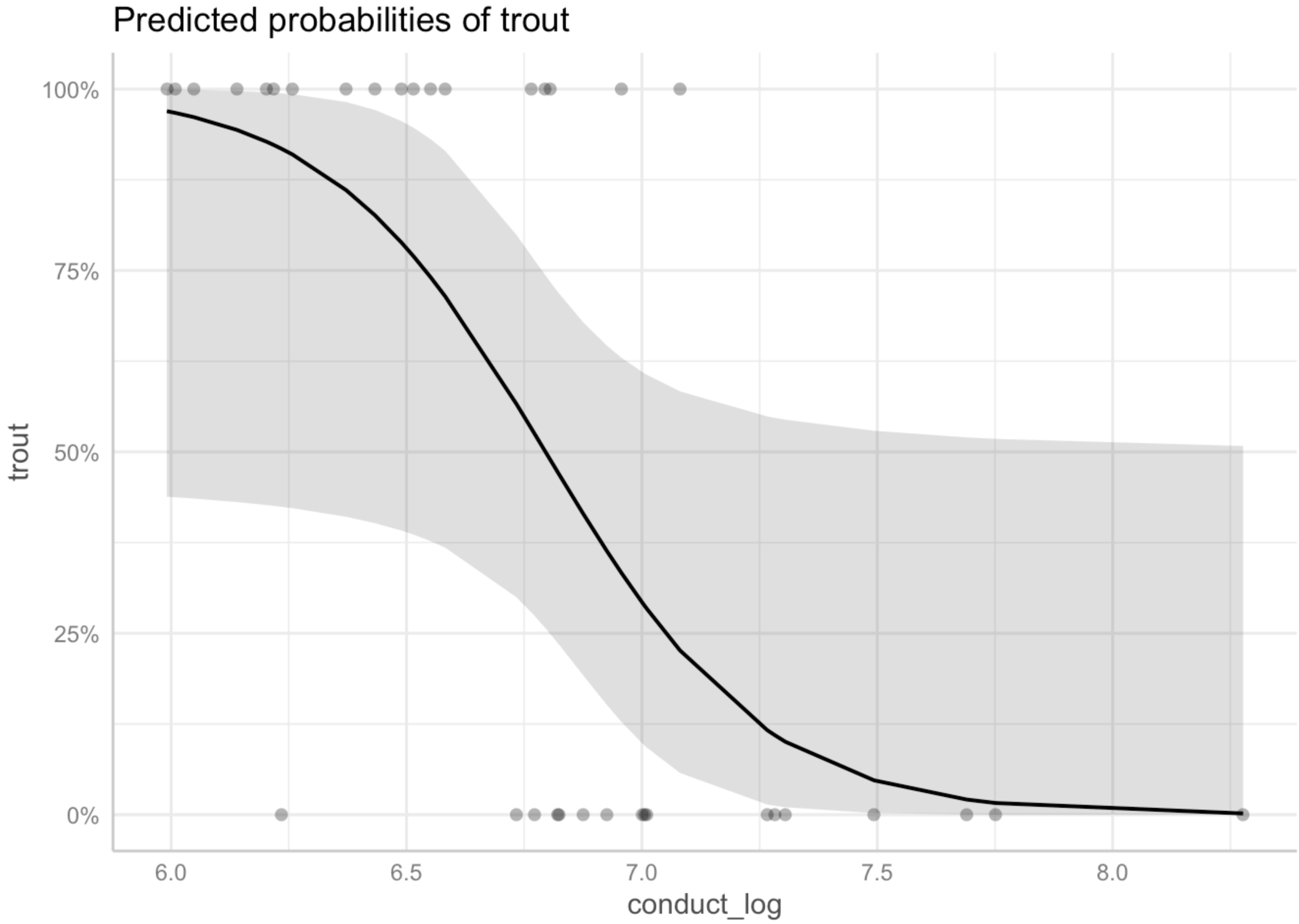


```
##                                 vif
## scale(max_depth)         1.733560
## scale(q_log)             1.359015
## scale(avg_canopy_logit)  2.152467
## scale(conduct_log)       1.646723
## scale(do)                1.442551
## scale(thermal_index)     1.674140
## scale(avg_chlorophyll)   1.648170
## scale(total_inverts)     1.339211
```

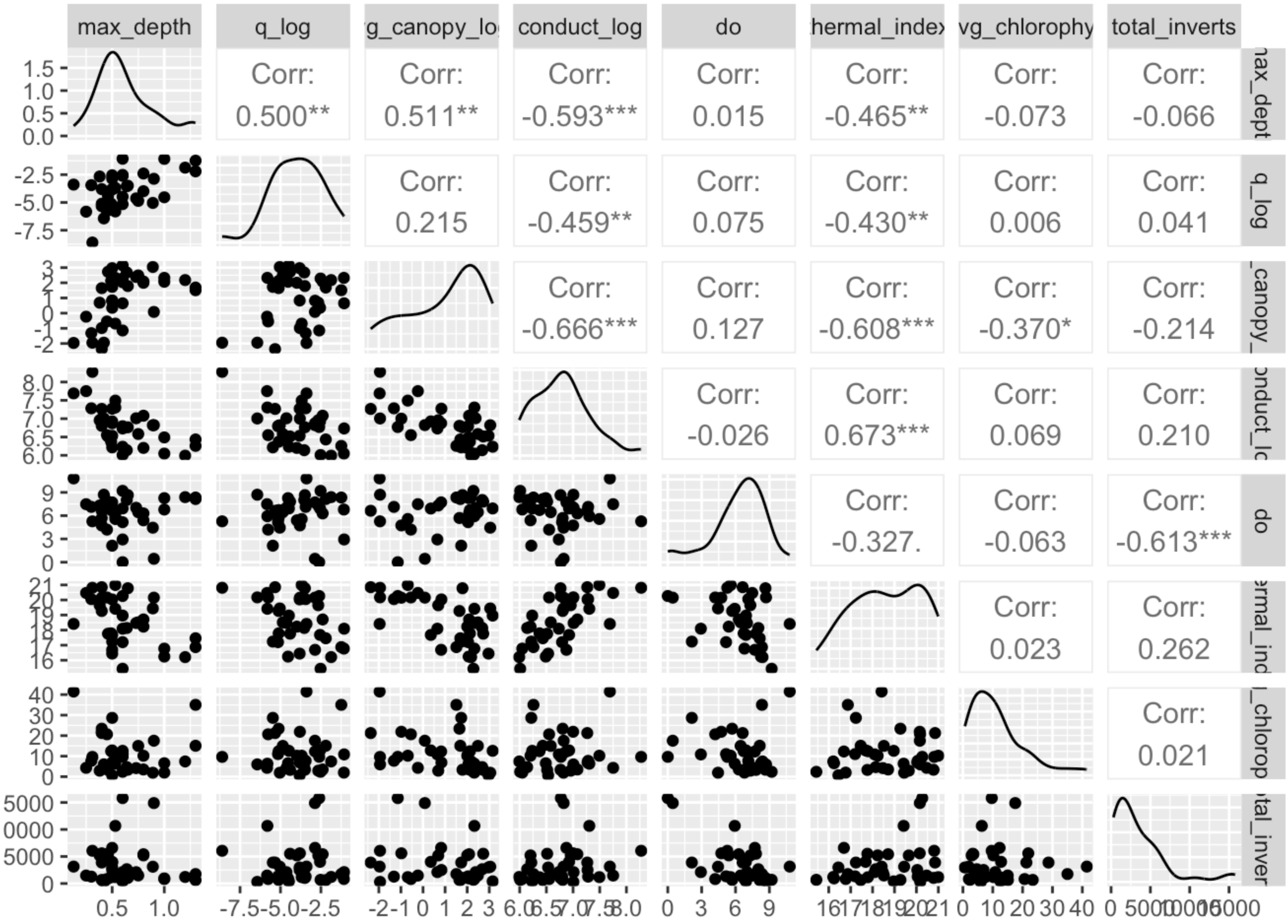
```
##
## Call:
## glm(formula = trout ~ scale(max_depth) + scale(q_log) + scale(avg_canopy_logit) +
##     scale(conduct_log) + scale(do) + scale(thermal_index) + scale(avg_chlorophyll) +
##     scale(total_inverts), family = "binomial", data = df_mod2)
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)              0.02823    0.52887   0.053   0.9574
## scale(max_depth)        -0.05209    0.86972  -0.060   0.9522
## scale(q_log)             0.12768    0.66611   0.192   0.8480
## scale(avg_canopy_logit)  0.72601    0.86743   0.837   0.4026
## scale(conduct_log)      -2.24500    1.14989  -1.952   0.0509 .
## scale(do)                0.67039    0.71203   0.942   0.3464
## scale(thermal_index)     0.18991    0.79834   0.238   0.8120
## scale(avg_chlorophyll)   0.28376    0.74028   0.383   0.7015
## scale(total_inverts)    -0.29066    0.74689  -0.389   0.6972
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 48.492  on 34  degrees of freedom
## Residual deviance: 24.811  on 26  degrees of freedom
##   (2 observations deleted due to missingness)
## AIC: 42.811
##
## Number of Fisher Scoring iterations: 6
```



So there doesn’t appear to be any issues with multicollinearity once we drop the pct_cover variable, which is a good thing. The model residuals also look acceptable. From this analysis it appears that the only variable related to the probability of trout presence is conductivity. Where increases in conductivity decrease the probability of trout presence. We can visualize the partial regression plot.



While the VIF for this model suggests that the multicollinearity isn’t an issue. I suspect that the fact that the only variable that appears correlated with trout presence is an indication that collinearity is still an issue with model interpretation.



Conductivity is strongly correlated with almost every other variable. So I do suspect that despite the low VIF scores this is causing issues in model interpretation.

Long story short, we could certainly use a glm() approach. However, I think we will be very limited in the inference we can draw from these models due to the implicit collinearity of the predictors you are interested in.

Option 2: SEM

As SEM (structual equation model) approach could be powerful in this situation because it allows us to model cascading influences of variables. While it would be very interesting to fit the model that Scott diagrammed in the word document, I think that would be far too complex for this data set. Lets try and reduce the number of variables and fit a simplified version of the model.

Here is a plot of the DAG. This is obviously a hypersimplified hypothesis of the relationships. Lets fit the SEM for this hypothesis.

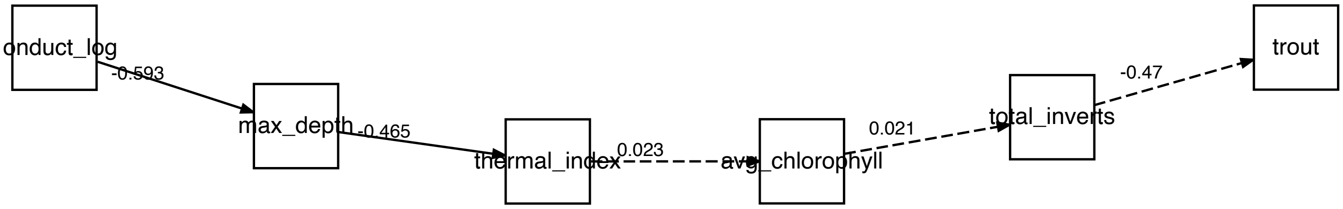
```
library(piecewiseSEM)

sem1 <- psem(
  lm(max_depth ~ conduct_log, df_mod2),
  lm(thermal_index ~ max_depth, df_mod2),
  lm(avg_chlorophyll ~ thermal_index, df_mod2),
  lm(total_inverts ~ avg_chlorophyll, df_mod2),
  glm(trout ~ total_inverts, df_mod2, family = "binomial")
)

summary(sem1)
```

```
##   |                                                                              |
|   0%  |                                                                              |
=======                                                               |  10%  |
|==============                                                        |  20%  |
|=====================                                                 |  30%  |
|============================                                          |  40%  |
|===================================                                   |  50%  |
|==========================================                            |  60%  |
|=================================================                     |  70%  |
|========================================================              |  80%  |
|===============================================================       |  90%  |
|======================================================================| 100%
```

```
## 
## Structural Equation Model of sem1 
## 
## Call:
##   max_depth ~ conduct_log
##   thermal_index ~ max_depth
##   avg_chlorophyll ~ thermal_index
##   total_inverts ~ avg_chlorophyll
##   trout ~ total_inverts
## 
##     AIC
##  1176.400
## 
## ---
## Tests of directed separation:
## 
##                         Independ.Claim Test.Type DF Crit.Value P.Value    
##     thermal_index ~ conduct_log + ...      coef 34     3.9109  0.0004 ***
##   avg_chlorophyll ~ conduct_log + ...      coef 34     0.4255  0.6732    
##     total_inverts ~ conduct_log + ...      coef 34     1.2489  0.2202    
##             trout ~ conduct_log + ...      coef 34    -2.8703  0.0041  **
##     avg_chlorophyll ~ max_depth + ...      coef 33    -0.2530  0.8019    
##       total_inverts ~ max_depth + ...      coef 33     0.4329  0.6679    
##               trout ~ max_depth + ...      coef 33     0.5972  0.5504    
##   total_inverts ~ thermal_index + ...      coef 33     1.5630  0.1276    
##           trout ~ thermal_index + ...      coef 33    -1.5344  0.1249    
##         trout ~ avg_chlorophyll + ...      coef 33    -0.0298  0.9762    
## 
## --
## Global goodness-of-fit:
## 
## Chi-Squared = 34.27 with P-value = 0 and on 10 degrees of freedom
## Fisher's C = 41.142 with P-value = 0.004 and on 20 degrees of freedom
## 
## ---
## Coefficients:
## 
##          Response       Predictor Estimate Std.Error DF Crit.Value P.Value
##         max_depth     conduct_log  -0.3156    0.0724 35    -4.3600  0.0001
##     thermal_index       max_depth  -2.6109    0.8404 35    -3.1068  0.0037
##   avg_chlorophyll   thermal_index   0.1370    1.0202 35     0.1342  0.8940
##     total_inverts avg_chlorophyll   8.2850   65.6244 35     0.1262  0.9003
##             trout   total_inverts  -0.0003    0.0002 35    -1.7588  0.0786
##   Std.Estimate    
##        -0.5933 ***
##        -0.4649  **
##         0.0227    
##         0.0213    
##        -0.4700    
## 
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
## 
```

```
## ---
## Individual R-squared:
##
##          Response     method R.squared
##         max_depth       none      0.35
##     thermal_index       none      0.22
##   avg_chlorophyll       none      0.00
##     total_inverts       none      0.00
##             trout nagelkerke      0.16
```

```
plot(sem1)
```



So the global goodness-of-fit test suggest that this model isn’t a good representation of the data. The tests of directed separation suggest including two additional paths: thermal_index ~ conductivity and trout ~ conductivity. Lets include these paths and refit the model.

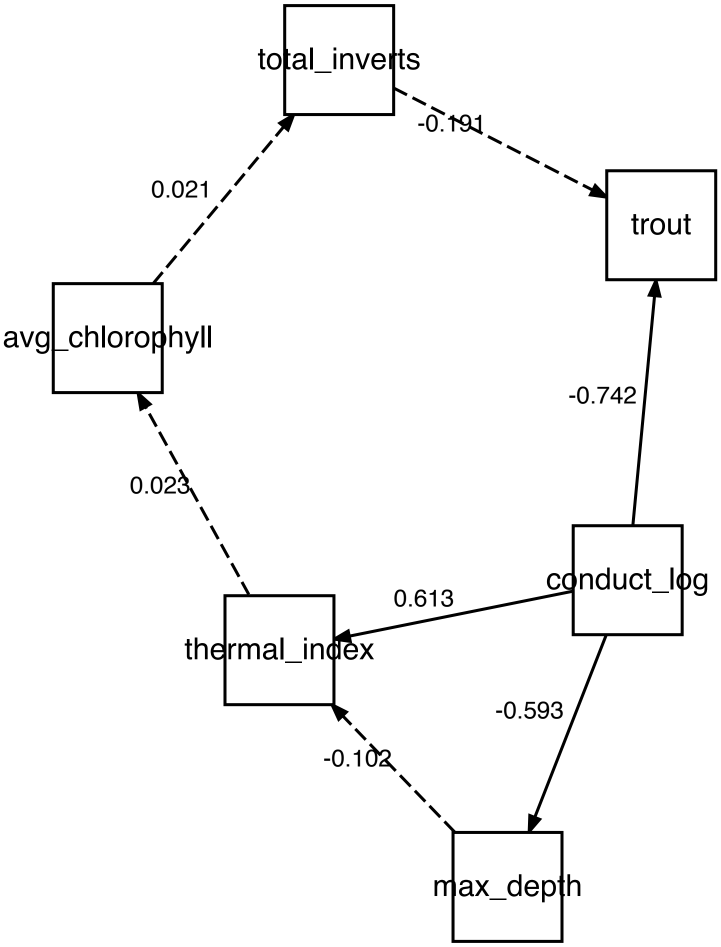
```
sem2 <- psem(
  lm(max_depth ~ conduct_log, df_mod2),
  lm(thermal_index ~ max_depth + conduct_log, df_mod2),
  lm(avg_chlorophyll ~ thermal_index, df_mod2),
  lm(total_inverts ~ avg_chlorophyll, df_mod2),
  glm(trout ~ total_inverts + conduct_log, df_mod2, family = "binomial")
)

summary(sem2)
```

```
##   |                                                                              |
|   0%  |                                                                              |
=========                                                             |  12%  |
|==================                                                    |  25%  |
|==========================                                            |  38%  |
|===================================                                   |  50%  |
|============================================                          |  62%  |
|====================================================                  |  75%  |
|=============================================================         |  88%  |
|======================================================================| 100%
```

```
##
## Structural Equation Model of sem2
##
## Call:
##   max_depth ~ conduct_log
##   thermal_index ~ max_depth + conduct_log
##   avg_chlorophyll ~ thermal_index
##   total_inverts ~ avg_chlorophyll
##   trout ~ total_inverts + conduct_log
##
##     AIC
##  1149.931
##
## ---
## Tests of directed separation:
##
##                         Independ.Claim Test.Type DF Crit.Value P.Value
##   avg_chlorophyll ~ conduct_log + ...      coef 34     0.4255  0.6732
##     total_inverts ~ conduct_log + ...      coef 34     1.2489  0.2202
##     avg_chlorophyll ~ max_depth + ...      coef 33    -0.2530  0.8019
##       total_inverts ~ max_depth + ...      coef 33     0.4329  0.6679
##               trout ~ max_depth + ...      coef 33     0.5972  0.5504
##   total_inverts ~ thermal_index + ...      coef 32     1.0196  0.3156
##           trout ~ thermal_index + ...      coef 32     0.1960  0.8446
##         trout ~ avg_chlorophyll + ...      coef 32     0.1600  0.8728
##
## --
## Global goodness-of-fit:
##
## Chi-Squared = 3.801 with P-value = 0.875 and on 8 degrees of freedom
## Fisher's C = 9.177 with P-value = 0.906 and on 16 degrees of freedom
##
## ---
## Coefficients:
##
##           Response       Predictor Estimate Std.Error DF Crit.Value P.Value
##          max_depth     conduct_log  -0.3156    0.0724 35    -4.3600  0.0001
##      thermal_index       max_depth  -0.5699    0.8797 34    -0.6479  0.5214
##      thermal_index     conduct_log   1.8304    0.4680 34     3.9109  0.0004
##    avg_chlorophyll   thermal_index   0.1370    1.0202 35     0.1342  0.8940
##      total_inverts avg_chlorophyll   8.2850   65.6244 35     0.1262  0.9003
##              trout   total_inverts  -0.0002    0.0001 34    -1.0890  0.2762
##              trout     conduct_log  -4.3428    1.5130 34    -2.8703  0.0041
##   Std.Estimate
##        -0.5933 ***
##        -0.1015
##         0.6126 ***
##         0.0227
##         0.0213
##        -0.1911
##        -0.7423  **
##
```

```
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##          Response     method R.squared
##         max_depth       none      0.35
##     thermal_index       none      0.46
##   avg_chlorophyll       none      0.00
##     total_inverts       none      0.00
##             trout nagelkerke      0.59
```

```
plot(sem2)
```



We can see that the Global goodness of fit statistics now suggest that the model is a apt representation of the data (or at least that we fail to reject the null hypothesis that the model is a poor representation of the data). There are also no additional paths suggested by the tests of directed separation. What I draw from this model, in addition to the glm() work previously, is that conductivity is a key predictor of trout presence. Also, at least in this data set, there is little evidence for a bottom up cascade extending from stream condition —> algae —> prey —> trout. Rather conductivity seems to be a key predictor of many other environmental variables, and is strongly correlated with the probability of trout presence.

Option 3: Latent-variable approach

The most interesting option (in my opinion) is to analyze the data using a latent-variable SEM style approach. I think that the advantage of this approach is that the goal of the paper could really be about developing an index of stream quality that is a strong predictor of trout presence. I feel like this would be something of interest to the U.S. forest service / California DFW? The general idea is that we have all these indicators of stream quality. Each of these indicators contributes something to an overall index of stream quality–a latent state that we cannot observe but we believe exists–and it is this overall index that we think is connected to trout presence.

There are two ways that I know of to fit latent state SEM's. The first is using the "Lavann" package in R. This is a decently user friendly approach, and I'll give it an initial attempt in some code below. The more powerful way we could fit this latent-state SEM is using Bayesian approaches. Most of the stats that I'm currently doing use Bayesian methods and this wouldn't be much of a stretch. The models are all custom built from scratch. So before I go into constructing one, I would need to know that you want to proceed in this direction. The power of fitting the model using Bayesian methods is that we could generate predictions across space for our latent state–stream quality–and robust confidence intervals for those predictions. Given our data limitations, this paper would likely be limited to in-sample predictions. In other words, we could produce a map of stream quality at the sites you sampled. However, the power would be that future field efforts may be able to just collect key indicators (conductivity) and generate predictions (think the probability of trout presence) based on those indicators.

If you want to read about a really cool application for latent state modeling check out Chris Brown's recent paper (https://www.sciencedirect.com/science/article/pii/S0048969723002851). This paper is what gave me the idea to use a latent variable approach. And Chris Brown made all the code accessible including tutorials here (https://www.seascapemodels.org/rstats/2023/06/15/bayesian-sem-tute.html) and here (https://github.com/cbrown5/ecological-condition-latent-model). One thing to note in this paper is that their entire analysis is based off of a time series of annual observations over ~ 30 years. So at least we would have some justification for using a similar approach to our data set of a similar size.

Before diving into the Bayesian approach, let me try and get some Lavaan code up and running.

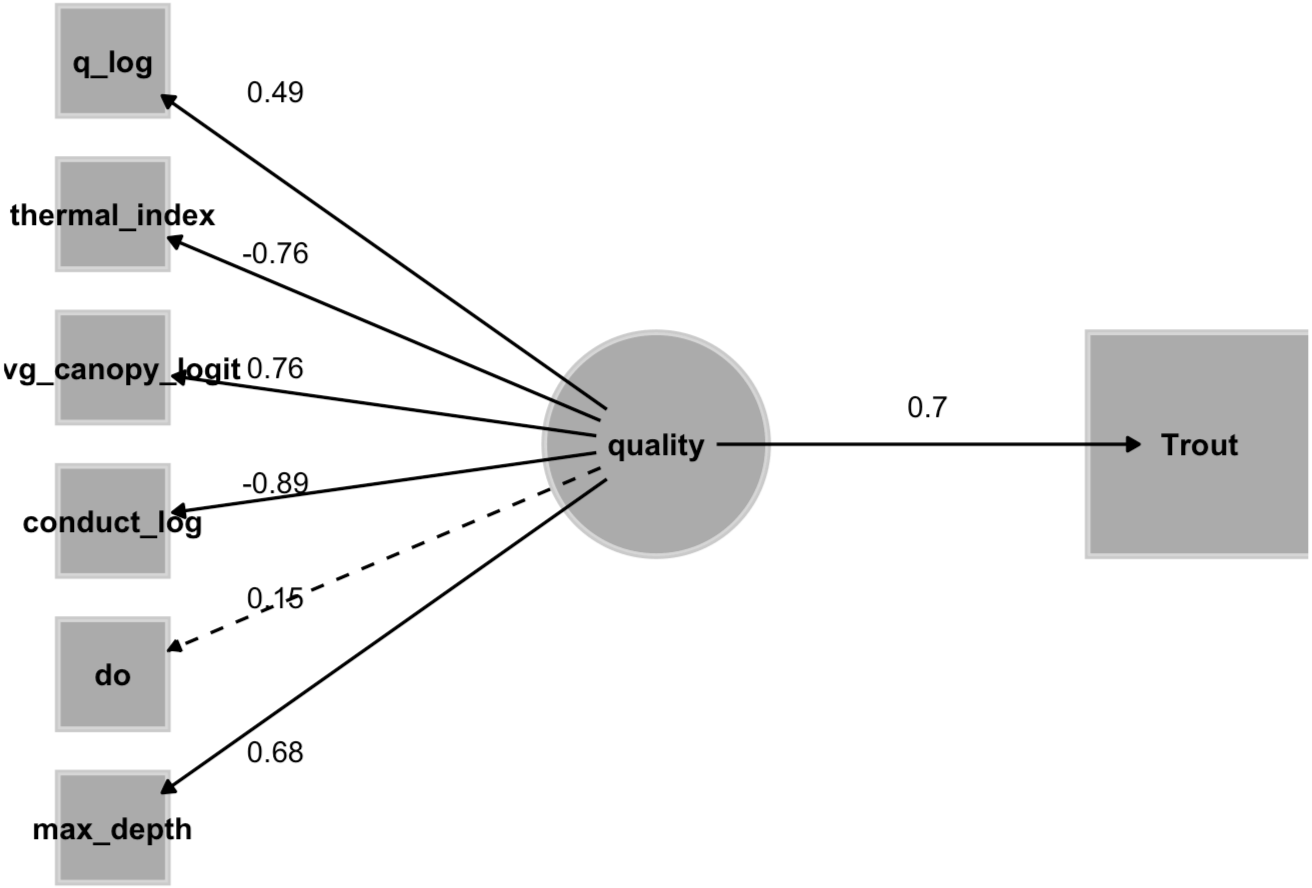
```
library(lavaan)

lv1 <- '
# latent
quality =~ q_log + thermal_index + avg_canopy_logit + conduct_log + do + max_depth

# structural paths
trout ~ quality

# correlated errors
'

lv1_mod <- sem(lv1, df_mod2, std.lv = T)

summary(lv1_mod, standardized = T)
```

```
## lavaan 0.6.17 ended normally after 29 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        14
##
##                                                   Used       Total
##   Number of observations                            35          37
##
## Model Test User Model:
##
##   Test statistic                                18.306
##   Degrees of freedom                                14
##   P-value (Chi-square)                           0.193
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   quality =~
##     q_log             0.805    0.274    2.940    0.003    0.805    0.490
##     thermal_index    -1.157    0.226   -5.115    0.000   -1.157   -0.762
##     avg_canopy_lgt    1.232    0.243    5.063    0.000    1.232    0.756
##     conduct_log      -0.465    0.072   -6.473    0.000   -0.465   -0.893
##     do                0.342    0.408    0.840    0.401    0.342    0.149
##     max_depth         0.190    0.044    4.369    0.000    0.190    0.679
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   trout ~
##     quality           0.349    0.077    4.541    0.000    0.349    0.699
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .q_log             2.049    0.508    4.031    0.000    2.049    0.760
##    .thermal_index     0.968    0.278    3.484    0.000    0.968    0.419
##    .avg_canopy_lgt    1.135    0.323    3.509    0.000    1.135    0.428
##    .conduct_log       0.055    0.025    2.250    0.024    0.055    0.203
##    .do                5.171    1.239    4.172    0.000    5.171    0.978
##    .max_depth         0.042    0.011    3.762    0.000    0.042    0.539
##    .trout             0.128    0.034    3.711    0.000    0.128    0.512
##     quality           1.000                               1.000    1.000
```

```
# library(lavaanPlot)
# lavaanPlot(lv1_mod)
```

The excerpt above is a pretty complex summary of the lavaan SEM output. There are a few things to note. First the p-value on the Chi-square test indicates the model is a potentially accurate representation of the data (technically, we fail to reject the null hypothesis that the model doesn't fit the data. And yes its funky because we are looking for $p \geq 0.05$). If you look at the "Latent Variables:" portion of the table you can see the loadings (e.g. "Estimates") of each of the exogenous indicators on the latent state, which here I've called "quality" (short for stream quality). Much of this is consistent with previous analyses. Everything (except DO) significantly loads on the latent state ($p \leq 0.05$). If you look at the "Std.all" column you can see standardized estimates of the loadings. So the indicator that loads the strongest is conductivity (technically the log of conductivity), followed by thermal index, the logit of average canopy, max depth, and finally q (or discharge). Finally, and likely most important, we can look at the "Regression" table. I a priori assumed that the latent state, quality, predicted trout presence. We can see that indeed, trout presence was significantly related to the latent variable, stream quality. Furthermore, the standardized magnitude of this effect was high (0.699) relative to the loadings of the indicators on the latent state.

Plotting these up in R is a bit tricky but here a crude attempt.



All this goes to say: environmental indicators load onto a latent state variable, that we will call stream quality. And stream quality is a good predictor of trout presence. The arrows in this figure are a bit tricky to interpret. You would expect the arrows to go **from** the indicators (environmental variables) **to** the latent state. However, in the parlance of SEM, the latent state is thought to be an emergent property of the indicators. Therefore, the arrows extend from the latent state to the indicators. Conversely, the arrow from quality to trout points in the same direction (from latent state to trout). However, the interpretation is different. Here, we have assumed that the latent state is predicting trout presence.

We could stop here if you wished. However, one of the deficiencies of lavaan is that we cannot use different data distributions. So here the regression trout ~ quality is a simple linear regression. Trout is a binary predictor (0, 1), so really it should be modeled as a glm(…, family = binomial(link = “logit”)). However, that isn’t possible in lavaan. Furthermore, generating predictions and CI’s from lavaan is not intuitive or strait-forward. I don’t think that building out the model in STAN (Bayesian analysis) would be particularly onerous, BUT I do think it would add significantly to the logic of the paper and the story. So I would recommend going that direction if you choose to pursue the latent state modeling approach.