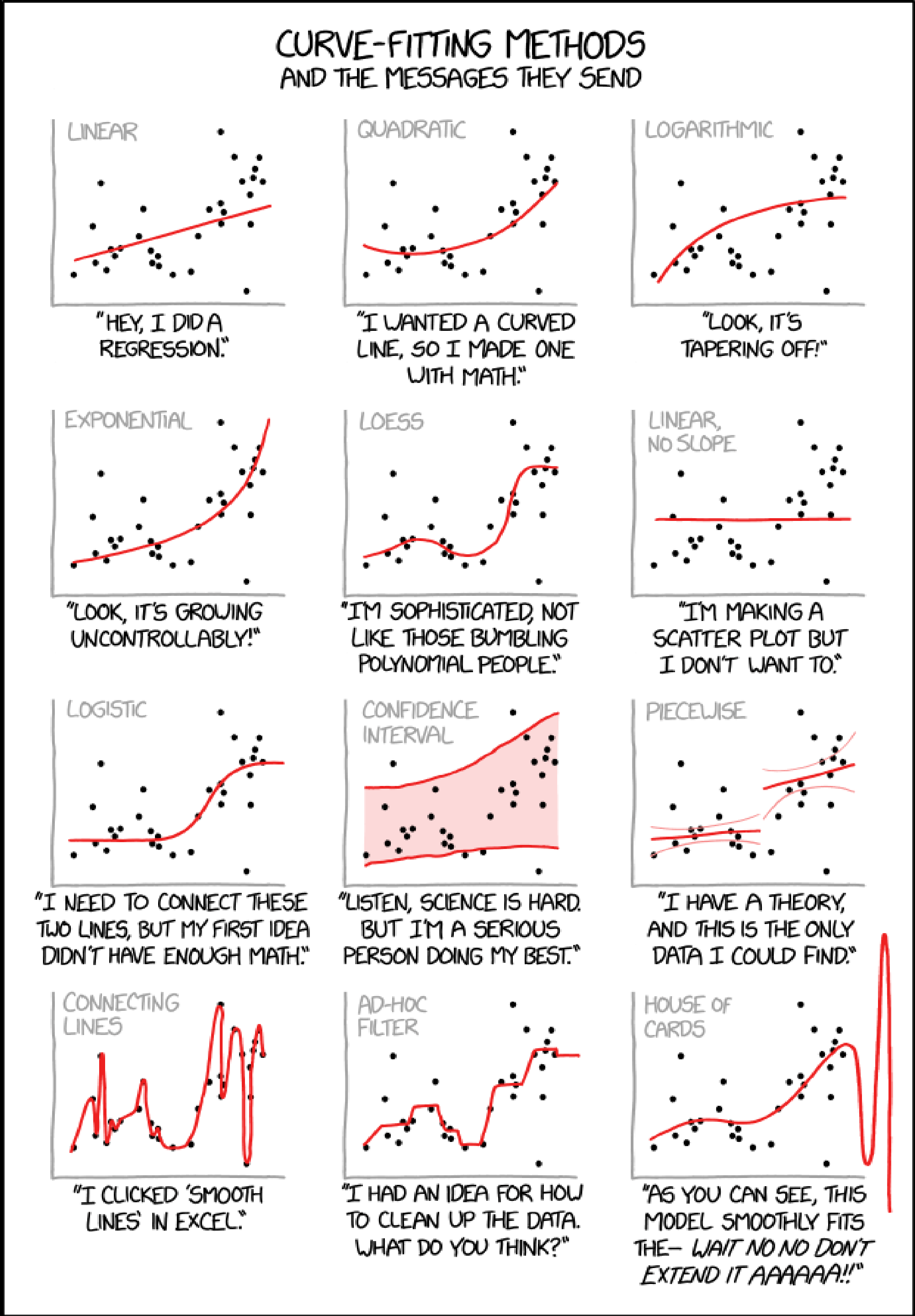
BIOL 599
Model Selection


CURVE-FITTING METHODS
AND THE MESSAGES THEY SEND
LINEAR
"HEY, I DID A REGRESSION."
QUADRATIC
"I WANTED A CURVED LINE, SO I MADE ONE WITH MATH."
LOGARITHMIC
"LOOK, IT'S TAPERING OFF!"
EXPONENTIAL
"LOOK, IT'S GROWING UNCONTROLLABLY!"
LOESS
"I'M SOPHISTICATED, NOT LIKE THOSE BUMBLING POLYNOMIAL PEOPLE."
LINEAR, NO SLOPE
"I'M MAKING A SCATTER PLOT BUT I DON'T WANT TO."
LOGISTIC
"I NEED TO CONNECT THESE TWO LINES, BUT MY FIRST IDEA DIDN'T HAVE ENOUGH MATH."
CONFIDENCE INTERVAL
"LISTEN, SCIENCE IS HARD. BUT I'M A SERIOUS PERSON DOING MY BEST."
PIECEWISE
"I HAVE A THEORY, AND THIS IS THE ONLY DATA I COULD FIND."
CONNECTING LINES
"I CLICKED 'SMOOTH LINES' IN EXCEL."
AD-HOC FILTER
"I HAD AN IDEA FOR HOW TO CLEAN UP THE DATA. WHAT DO YOU THINK?"
HOUSE OF CARDS
"AS YOU CAN SEE, THIS MODEL SMOOTHLY FITS THE— WAIT NO NO DON'T EXTEND IT AAAAAA!!"

Reminders/Updates

Office Hours:

Cancelled this Thursday, Sorry!!

Workshop and Paper Discussion on Thursday:

Model selection

“Essentially, all models are wrong, but some models are useful.”

George E.P. Box

You can make a line for any model, but that doesn’t mean that specific line is a good fit (or the only fit).

All models have error.

Outline

1. What is model selection?
2. Types of model selection
   - Full vs. Reduced LRT
   - Stepwise AIC
   - The single model
   - Multi-model inference
3. Cross-validation

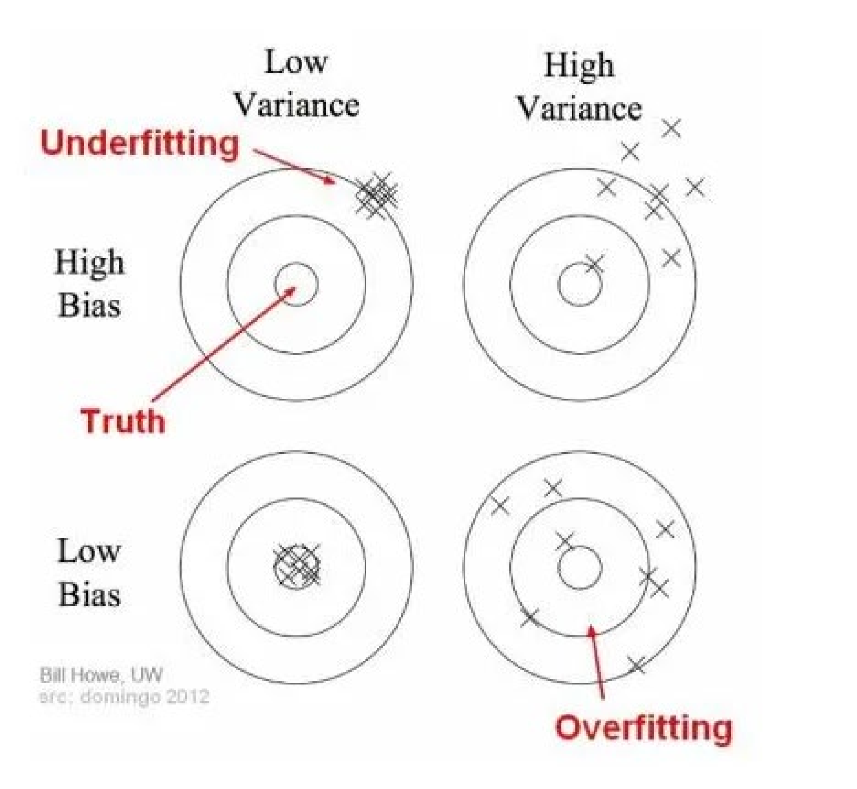
What is model selection?

Finding the “best fit” model:

1. The best subset of predictors should include the most important ones in explaining the variation in the response variable
2. The precision of predictors from the model will be greater with fewer predictors in the model.

Balance goodness of fit with simplicity

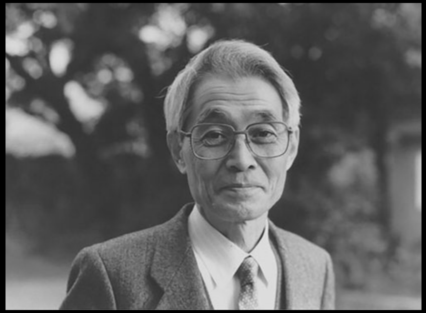
- If a model includes **too many** predictors → common issue could be overfitting
  - Gives good predictions to training data but poor predictions when applied to new data model not trained on
  - Low bias but high variance
- If a model includes **too few** predictors → common issue could be underfitting
  - Gives poor predictions
  - Low variance but high bias
- Criteria most commonly used to do this is AIC!



Bill Howe, UW
src: domingo 2012

Model selection with AIC

- AIC = Akaike Information Criterion
- a measure of model fit that is commonly used in model selection
- balances goodness of fit with # of variables in the model



AIC Penalizes your loglikelihood.

$$AIC = -2\log l(\hat{\theta}) + 2p$$

logl is the log-likelihood of the data evaluated at the estimated parameter values θ^

p is the number of parameters in the model.

The likelihood will always increase as we add more parameters, but AIC may not due to the penalty, 2p.

Simpler models with smaller values of AIC are preferred, all else being equal.

Goals of model selection

1. Explore/Describe = capture the main patterns in the data in a parsimonious way. No clear hypothesis.
2. Explain/infer = test biological hypothesis or set of competing hypotheses about how the world works. use to understand mechanism.
3. Predict = use data we have in hand to make predictions about future data use to test the strength of all predictors.

What are your goals?

Model selection:
there is no “right” answer

- Even with a moderate number of predictors, the number of possible models can get large (*p* predictors = 2^p possible models)
- There will rarely be a single best subset of predictors for any data set
- Even if the analysis selects a ‘best’ model or an average of a set of models this does not mean that the model is any good in an explanatory or predictive sense
- **Remove collinear variables FIRST**

Always remember your original hypothesis.

Model selection: there is no "right" answer

- **Be transparent** on the model selection process, packages used, and why parameters were included or excluded
- Your model selection and choice of parameters should **make biological sense with thought ahead of time on which parameters are meaningful**
- Think about experimental design ahead of time—can you minimize confounding factors
- Data analysis should follow your question design and experimental design

Types of model selection

- Full vs. Reduced LRT
- Stepwise AIC
- The single model
- Multi-model inference

Reduced vs. Full – you've already done this!

model (X,Y, both numerical)

1. Fit null (reduced model)
   - null.model<-lme(y~x, weights=(form~1|x), data=mydata)

2. Fit full (+factor1 model) ***that only varies by adding this 1 fixed factor***
   - model1<-lme(y~x + **fixed factor1**, weights=(form~1|x), data=mydata)

3. Compare hierarchically nested models with LRT test
   - anova(null,model1) – **"Which model is better?" or "Does adding this fixed factor improve the model better than the null model?"**

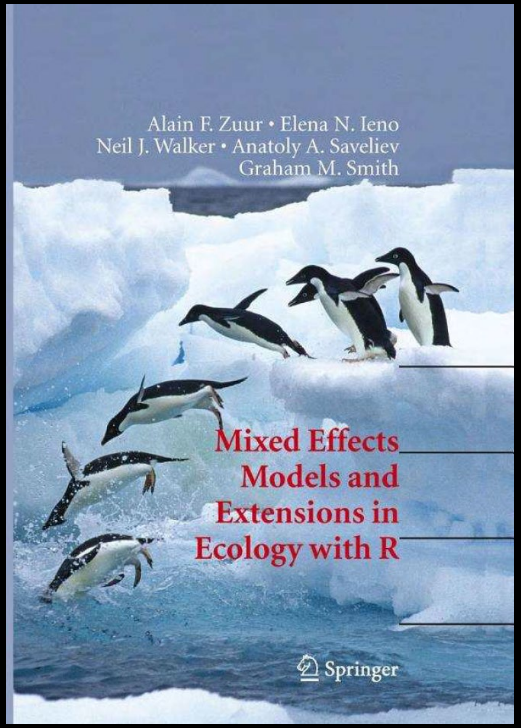
Could you compare the null model to model 2 with LRT? Why or why not?

- null.model<-lme(y~1, random=~x|animal, data=mydata)
- model1<-lme(y~**fixed factor1**, random=~x|animal, data=mydata)
- model2<-lme(y~**fixed factor1+fixed factor2**, random=~x|animal, data=mydata)

You can't compare model 2 with the null model because it differs with 2 factors.

General procedure for mixed model selection

1. Determine optimal random structure using “beyond optimal” fixed structure model
   - Test nested models using LRT with REML estimation
2. Determine optimal fixed effects
   - Lots of different philosophies on how
   - Test nested models using LRT with ML estimation
3. Run a final model
   - Use REML for final parameter estimates
   - Use LRT to get p-values for overall effects


Alain F. Zuur • Elena N. Ieno
Neil J. Walker • Anatoly A. Saveliev
Graham M. Smith
Mixed Effects Models and Extensions in Ecology with R
Springer

LRT: likelihood restricted test.

Optimal random structure

1. If they are introduced by design, the you should ALWAYS include them in your model
   - NEVER appropriate to remove these terms from the model - even if they are non-significant
2. Start with “beyond optimal” model that contains all/most potential predictor variables and interactions
   - This ensures that the model first pulls out any and all variation attributable to any potential fixed effects first (because inherently interested in fixed effects usually, and not so much random effects)
   - Of course, if you have LOTS of predictors a full model may not be possible

Fixed effects & interactions

- Numerous philosophies on how to do this:
  1. Start with a model with no interactions. Apply model and validate. Check residuals and include interactions if needed to explain patterns in residuals.
  2. Decide using biological knowledge of system which effects and interactions to include
  3. Use good data exploration to see which interactions are important
  4. Include only main terms and all two-way interactions
  5. Include all interactions by default and reduce

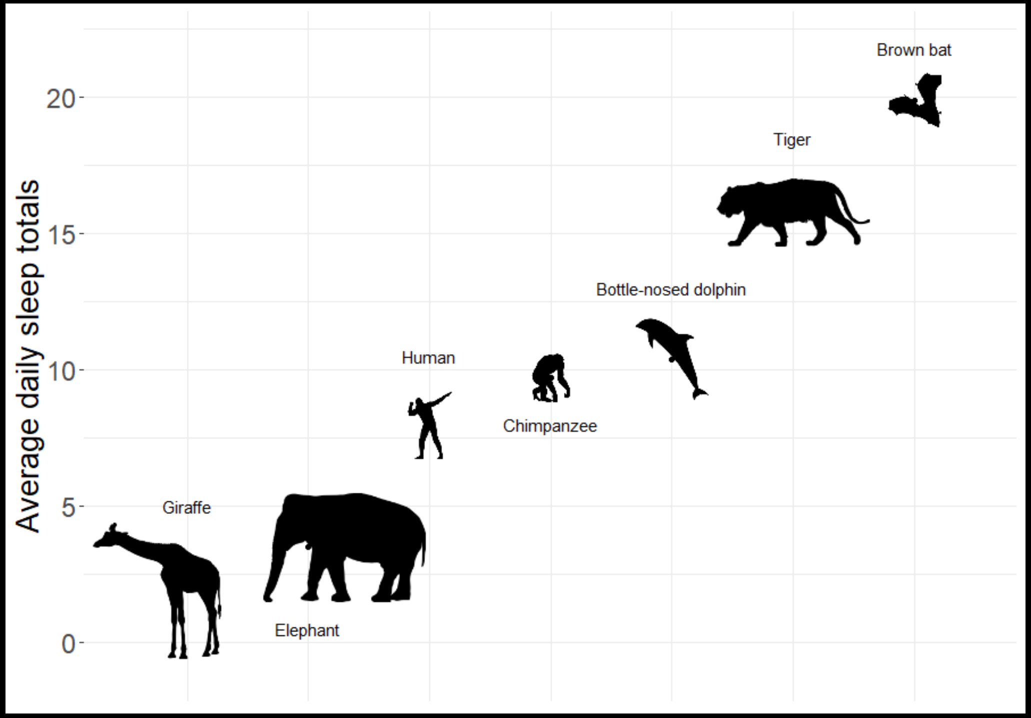
- This applies to non-mixed (regular linear) models too!

Example – mammal sleep

What factors predict how long mammals sleep?

- 39 species of mammals from 7 orders

- use order as a random intercept

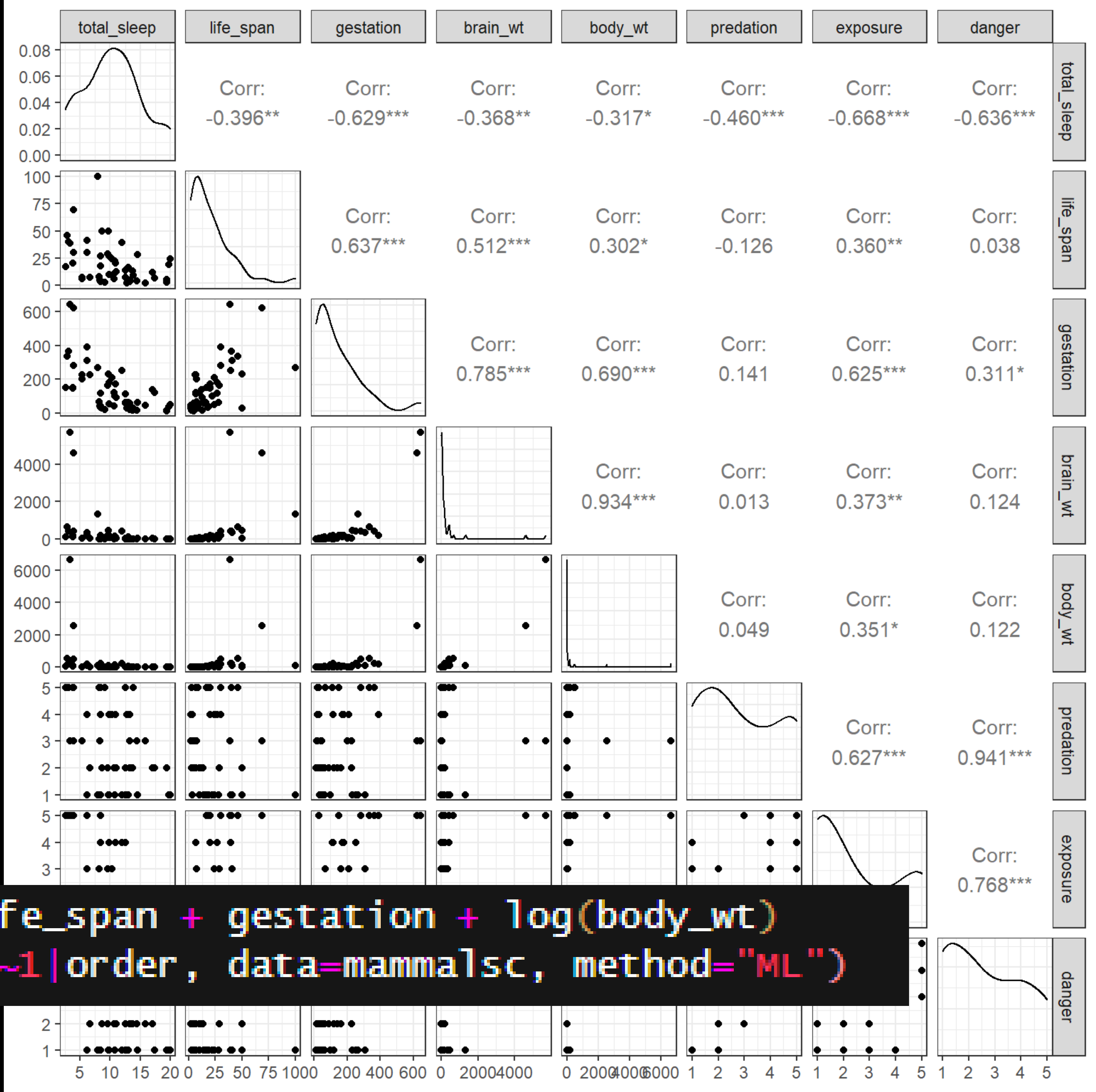


What factors predict how long mammals sleep?

ggally ( )

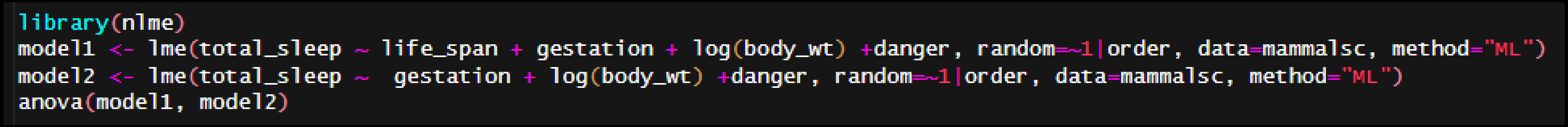
for the figure

- Lifespan (years)
- Gestation (days)
- Brain weight (g)
- Body weight (kg)
- Predation Index (1-5; 1 = least likely to be preyed upon)
- Exposure Index [1-5: 1 = least exposed (e.g., animal sleeps in a den)]
- Danger Index (1:5, combines exposure and predation; 1= least danger from other animals)

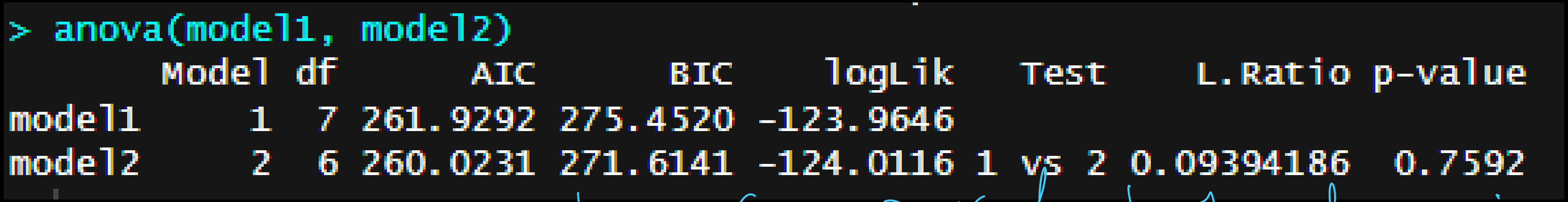


```
model1 <- lme(total_sleep ~ life_span + gestation + log(body_wt)
             +danger, random=~1|order, data=mammalsc, method="ML")
```

LRT of full model vs. reduced



```
library(nlme)
model1 <- lme(total_sleep ~ life_span + gestation + log(body_wt) +danger, random=~1|order, data=mammalsc, method="ML")
model2 <- lme(total_sleep ~  gestation + log(body_wt) +danger, random=~1|order, data=mammalsc, method="ML")
anova(model1, model2)
```



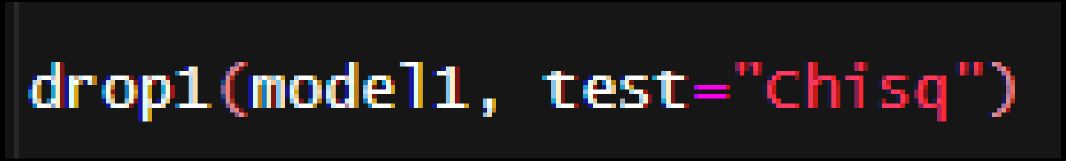
```
> anova(model1, model2)
       Model df      AIC      BIC    logLik   Test    L.Ratio p-value
model1     1  7 261.9292 275.4520 -123.9646
model2     2  6 260.0231 271.6141 -124.0116 1 vs 2 0.09394186  0.7592
```

Model 1 will be compared by dropping
all the predictors and see the effect.
one after the other.

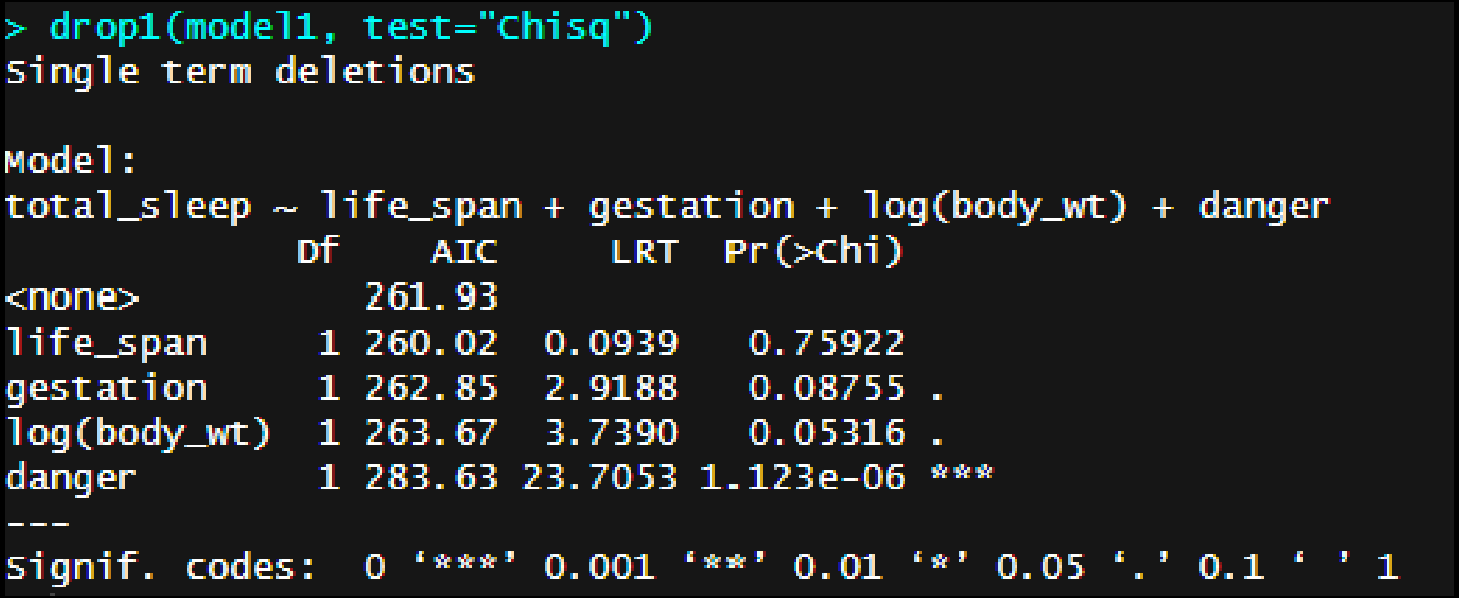
drop1() function will systematically drop each of your variable from the

LRT with drop 1 does the same thing!

full model (model 1)



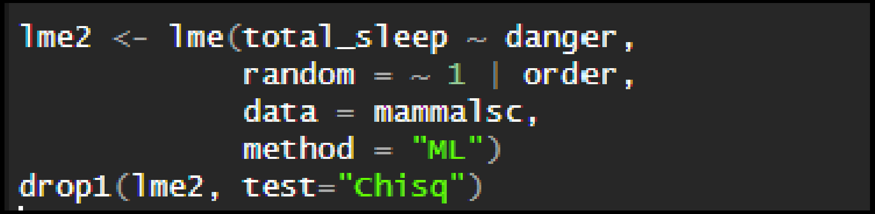
```
drop1(model1, test="Chisq")
```



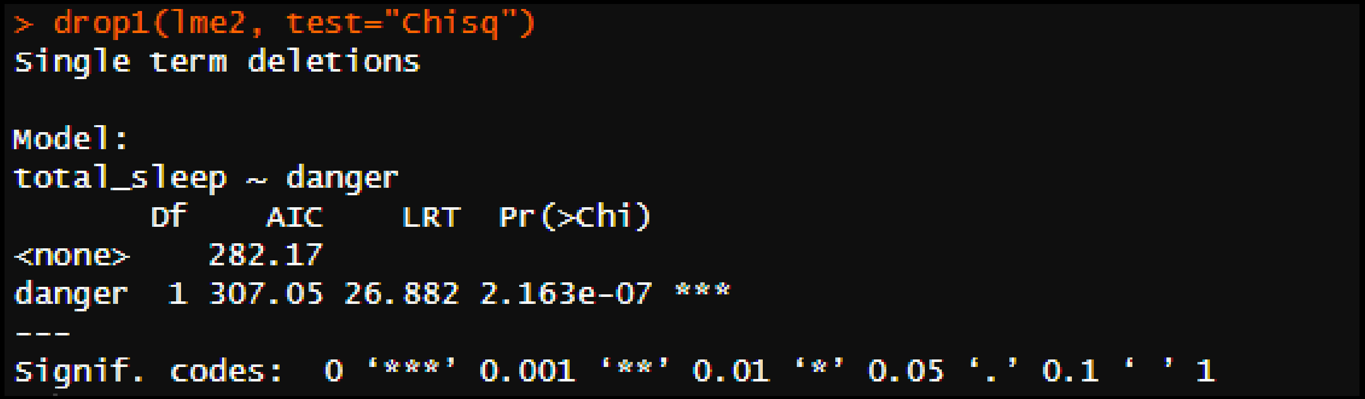
```
> drop1(model1, test="Chisq")
Single term deletions

Model:
total_sleep ~ life_span + gestation + log(body_wt) + danger
             Df    AIC     LRT  Pr(>Chi)    
<none>          261.93                      
life_span     1 260.02  0.0939   0.75922    
gestation     1 262.85  2.9188   0.08755 .  
log(body_wt)  1 263.67  3.7390   0.05316 .  
danger        1 283.63 23.7053 1.123e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Tests the effect of dropping each predictor one at a time.
- Reports AIC of the reduced model and LRT p-value against the full model.
- Same as manually fitting reduced models, but automated

We’re not done yet!

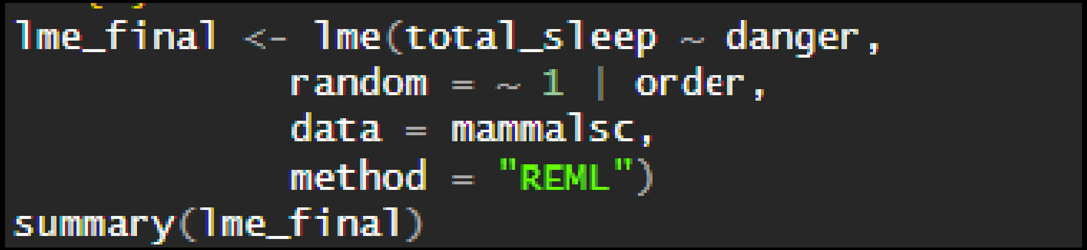


```
lme2 <- lme(total_sleep ~ danger,
            random = ~ 1 | order,
            data = mammalsc,
            method = "ML")
drop1(lme2, test="Chisq")
```

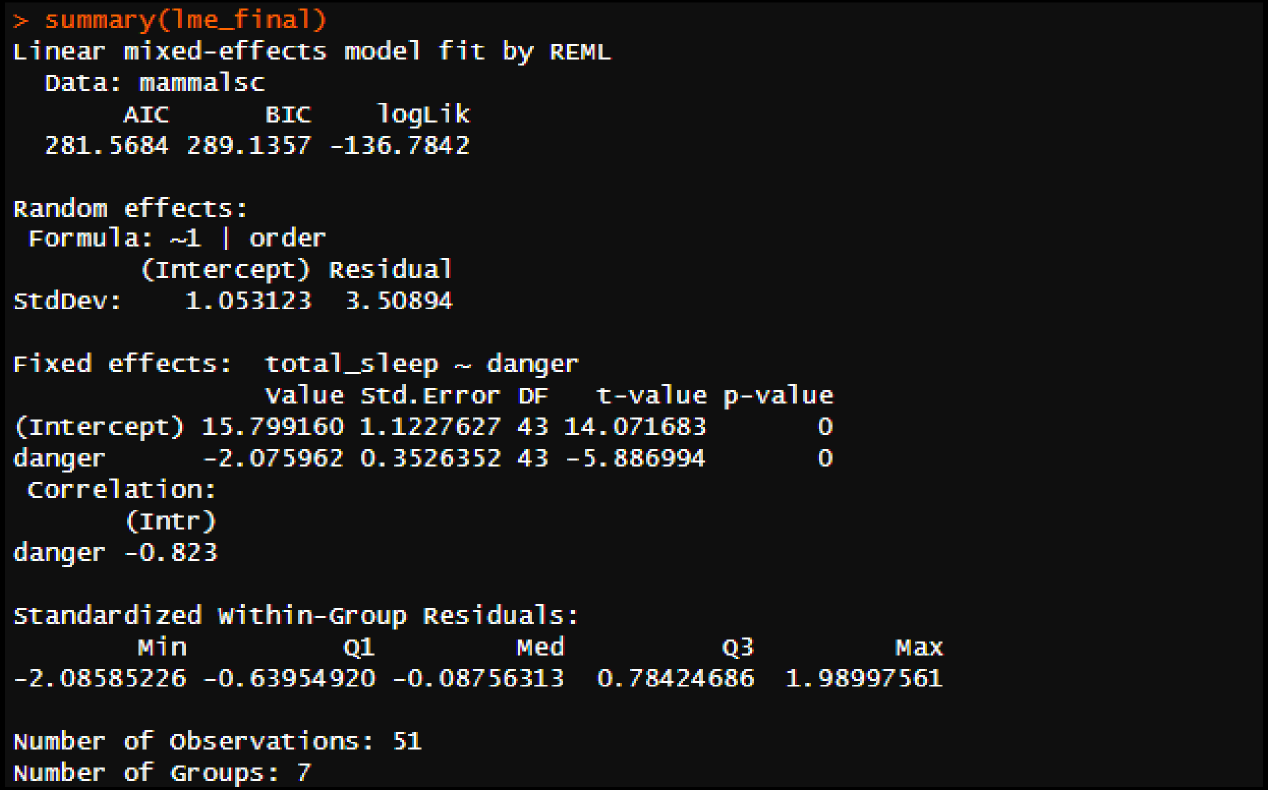


```
> drop1(lme2, test="Chisq")
Single term deletions

Model:
total_sleep ~ danger
       Df    AIC    LRT  Pr(>Chi)
<none>    282.17
danger  1 307.05 26.882 2.163e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Interpret Final model (use REML)



```
lme_final <- lme(total_sleep ~ danger,
                 random = ~ 1 | order,
                 data = mammalsc,
                 method = "REML")
summary(lme_final)
```



```
> summary(lme_final)
Linear mixed-effects model fit by REML
  Data: mammalsc
       AIC      BIC    logLik
  281.5684 289.1357 -136.7842

Random effects:
 Formula: ~1 | order
        (Intercept) Residual
StdDev:    1.053123  3.50894

Fixed effects:  total_sleep ~ danger
                Value Std.Error DF   t-value p-value
(Intercept) 15.799160 1.1227627 43 14.071683       0
danger      -2.075962 0.3526352 43 -5.886994       0
 Correlation:
       (Intr)
danger -0.823

Standardized Within-Group Residuals:
        Min          Q1         Med          Q3         Max
-2.08585226 -0.63954920 -0.08756313  0.78424686  1.98997561

Number of Observations: 51
Number of Groups: 7
```

We use LRTs to decide what belongs in the model, but we report the estimates from the final fitted model

What do I write?

Methods:

- State the *full model you started with* (list fixed and random effects).
- Say *how* you did model selection (e.g., LRT + AIC, stepwise reduction).

Results:

- Report the **LRT results** for dropping predictors (χ^2, p-values, AIC).
- Be transparent: mention which predictors were kept/dropped.

Final Model:

- Report the **estimates and SEs** from the final reduced model.
- Add a short interpretation of the effect(s).

Reduced vs. Full models

- Start with a strong a priori hypothesis (this is why stats helps you become a better scientist!)
- Present your initial full model and your reduced model to be transparent

Types of model selection

- Full vs. Reduced LRT
- Stepwise AIC
- The single model
- Multi-model inference

Stepwise multiple regression

- Using stepwise elimination or addition of terms, is a common practice, but CRITICIZED
- Fitting a multiple regression with many variables, cycle of adding/deleting model terms and then refitting.
- Continue until only statistically significant terms are in the model
- Mass package = stepAIC function (direction="backward", "forward", "both")
- lmerTest package = step function (options for both random and fixed effects)

NOTE: LRT of full vs reduced with drop1 is NOT backward stepwise selection

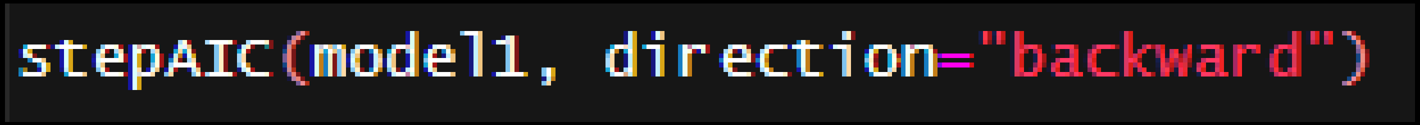
Backward stepwise selection

1. Fit a full model containing all predictors of interest.
2. Consider all possible models formed by dropping 1 of these predictors
3. Keep the current model, or drop the “worst” predictor depending on:
   - p-values from the individual t-tests (drop the variable with the highest p-value, if >0.05)
   - Adjusted R2 values (higher values are better) for lm
   - AIC (lower values are better) for lme
4. Rinse and repeat until you can no longer improve the

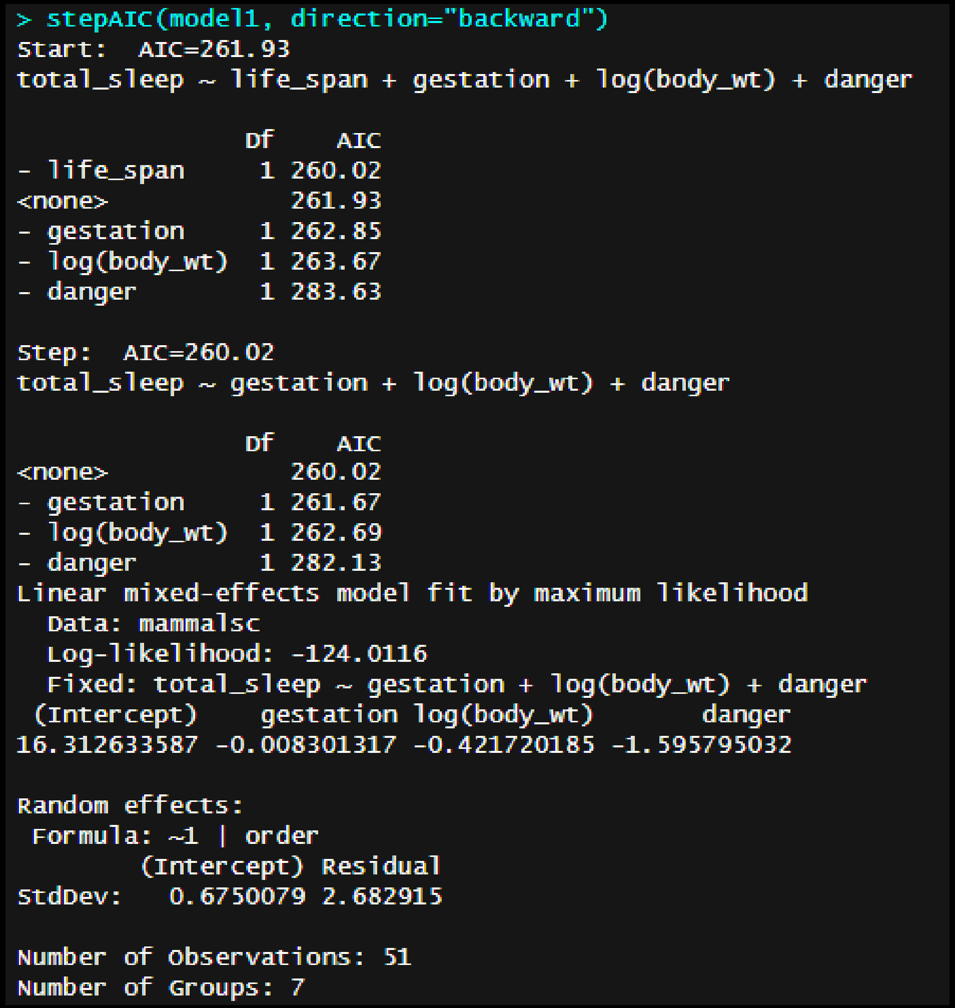
Forwards stepwise selection

1. Fit a null model containing no predictors
2. Consider all possible models formed by adding 1 predictor at a time
3. Keep the current model, or add the predictor based on comparisons between nested models
4. Rinse and repeat until you can no longer improve the model

Example – stepwise



```
stepAIC(model1, direction="backward")
```

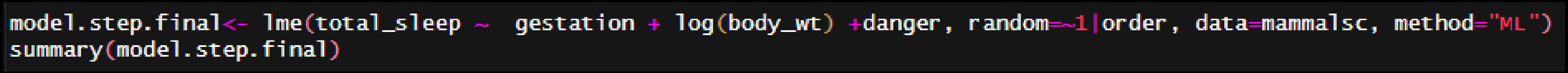


```
> stepAIC(model1, direction="backward")
Start:  AIC=261.93
total_sleep ~ life_span + gestation + log(body_wt) + danger

               Df    AIC
- life_span     1 260.02
<none>            261.93
- gestation     1 262.85
- log(body_wt)  1 263.67
- danger        1 283.63

Step:  AIC=260.02
total_sleep ~ gestation + log(body_wt) + danger

               Df    AIC
<none>            260.02
- gestation     1 261.67
- log(body_wt)  1 262.69
- danger        1 282.13
Linear mixed-effects model fit by maximum likelihood
  Data: mammalsc
  Log-likelihood: -124.0116
  Fixed: total_sleep ~ gestation + log(body_wt) + danger
 (Intercept)    gestation log(body_wt)       danger
16.312633587 -0.008301317 -0.421720185 -1.595795032

Random effects:
 Formula: ~1 | order
        (Intercept) Residual
StdDev:   0.6750079 2.682915

Number of Observations: 51
Number of Groups: 7
```

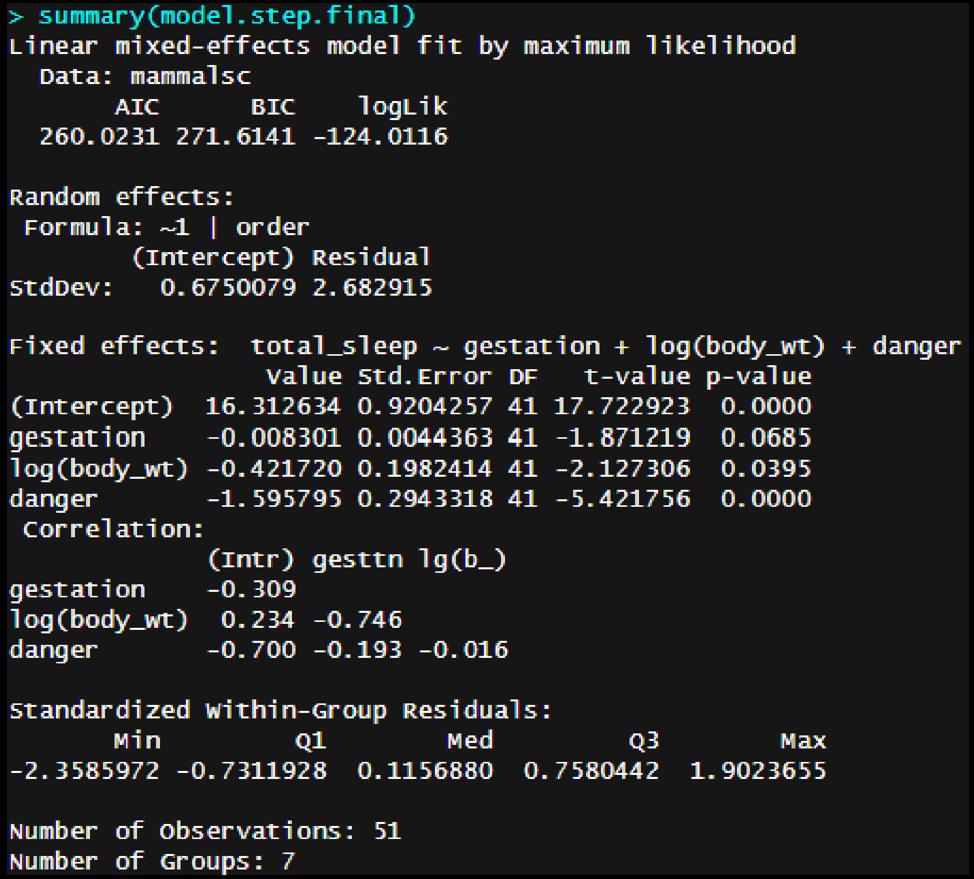
•Always drops the term with the **lowest AIC**, even if the improvement is tiny.

•ΔAIC < 2 usually means models are **essentially tied** → you must decide if the drop is meaningful.

Example – stepwise final model



```
model.step.final<- lme(total_sleep ~  gestation + log(body_wt) +danger, random=~1|order, data=mammalsc, method="ML")
summary(model.step.final)
```



```
> summary(model.step.final)
Linear mixed-effects model fit by maximum likelihood
  Data: mammalsc
       AIC      BIC    logLik
  260.0231 271.6141 -124.0116

Random effects:
 Formula: ~1 | order
        (Intercept) Residual
StdDev:   0.6750079 2.682915

Fixed effects:  total_sleep ~ gestation + log(body_wt) + danger
                 Value Std.Error DF   t-value p-value
(Intercept)  16.312634 0.9204257 41 17.722923  0.0000
gestation    -0.008301 0.0044363 41 -1.871219  0.0685
log(body_wt) -0.421720 0.1982414 41 -2.127306  0.0395
danger       -1.595795 0.2943318 41 -5.421756  0.0000
 Correlation: 
             (Intr) gesttn lg(b_)
gestation    -0.309              
log(body_wt)  0.234 -0.746       
danger       -0.700 -0.193 -0.016

Standardized Within-Group Residuals:
       Min         Q1        Med         Q3        Max 
-2.3585972 -0.7311928  0.1156880  0.7580442  1.9023655 

Number of Observations: 51
Number of Groups: 7 
```

It is best to consider these methods as useful for generating new hypotheses; these hypotheses should be tested with new, independent data

Types of model selection

- Full vs. Reduced LRT
- Stepwise AIC
- The single model
- Multi-model inference

The single model approach

Given the potential for overfitting and issues with multiple testing and inference when using model selection algorithms, it can be advantageous at times to just fit a single model and use it for inference

Select variables that are:

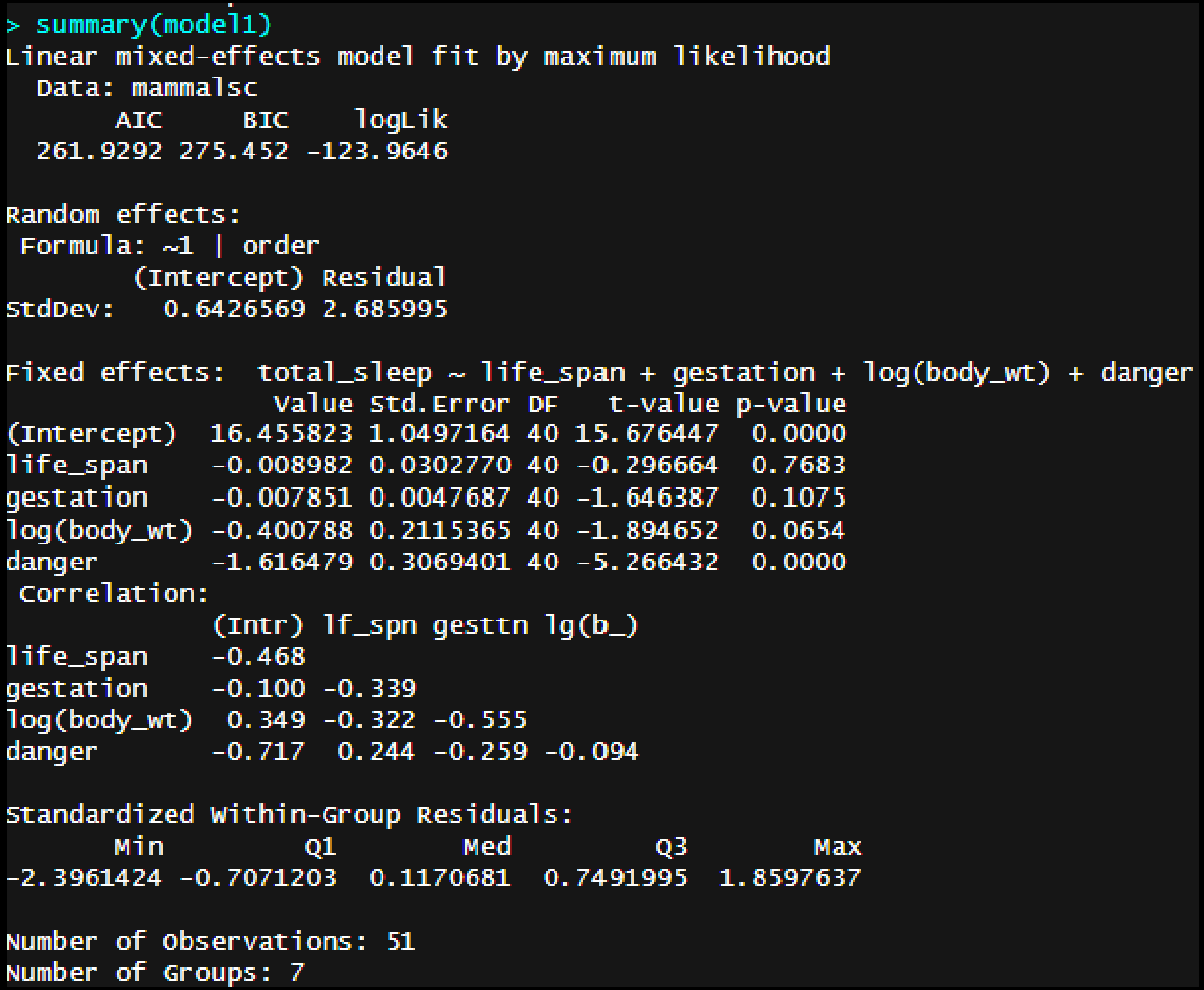
a) are ecologically relevant
b) are feasible to collect data on
c) are closer to the mechanism

The single model approach

1. fit a “full model” without further simplification.
2. assess significance:
   - measures of effect size and their uncertainty (summary)
   - LRT between reduced and full but KEEP predictors in the final model for parameter estimates

Main downside: prediction error may be larger due to including predictors that just add noise

Single model – final model



```
> summary(model1)
Linear mixed-effects model fit by maximum likelihood
  Data: mammalsc
       AIC     BIC    logLik
  261.9292 275.452 -123.9646

Random effects:
 Formula: ~1 | order
        (Intercept) Residual
StdDev:   0.6426569 2.685995

Fixed effects:  total_sleep ~ life_span + gestation + log(body_wt) + danger
                 Value Std.Error DF   t-value p-value
(Intercept)  16.455823 1.0497164 40 15.676447  0.0000
life_span    -0.008982 0.0302770 40 -0.296664  0.7683
gestation    -0.007851 0.0047687 40 -1.646387  0.1075
log(body_wt) -0.400788 0.2115365 40 -1.894652  0.0654
danger       -1.616479 0.3069401 40 -5.266432  0.0000
 Correlation: 
             (Intr) lf_spn gesttn lg(b_)
life_span    -0.468                     
gestation    -0.100 -0.339              
log(body_wt)  0.349 -0.322 -0.555       
danger       -0.717  0.244 -0.259 -0.094

Standardized Within-Group Residuals:
       Min         Q1        Med         Q3        Max 
-2.3961424 -0.7071203  0.1170681  0.7491995  1.8597637 

Number of Observations: 51
Number of Groups: 7 
```

Types of model selection

- Full vs. Reduced LRT
- Stepwise AIC
- The single model
- Multi-model inference

Multi-model inference

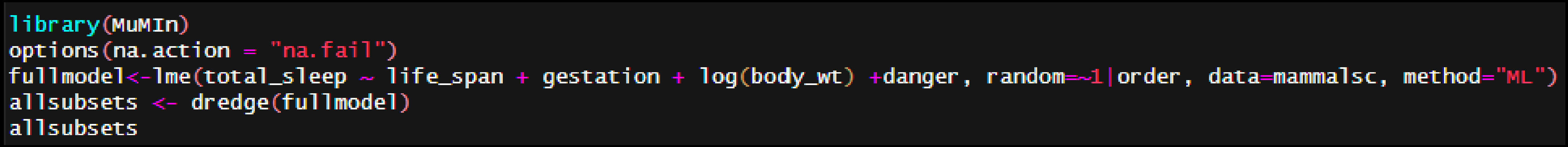
- AIC is often used to choose among competing models, not just nested models
- balances the fitness of a model with the number of predictors employed; best model minimizes AIC
- Rather than choose a best model, we can choose to average predictions among multiple “good” models.
- Model averaging incorporates uncertainty by weighting the parameter estimates of a model by the models’ Akaike weight

Symonds and Moussalli: A brief guide to model selection, multimodel inference and model averaging in behavioural ecology using AIC
https://link.springer.com/article/10.1007/s00265-010-1037-6

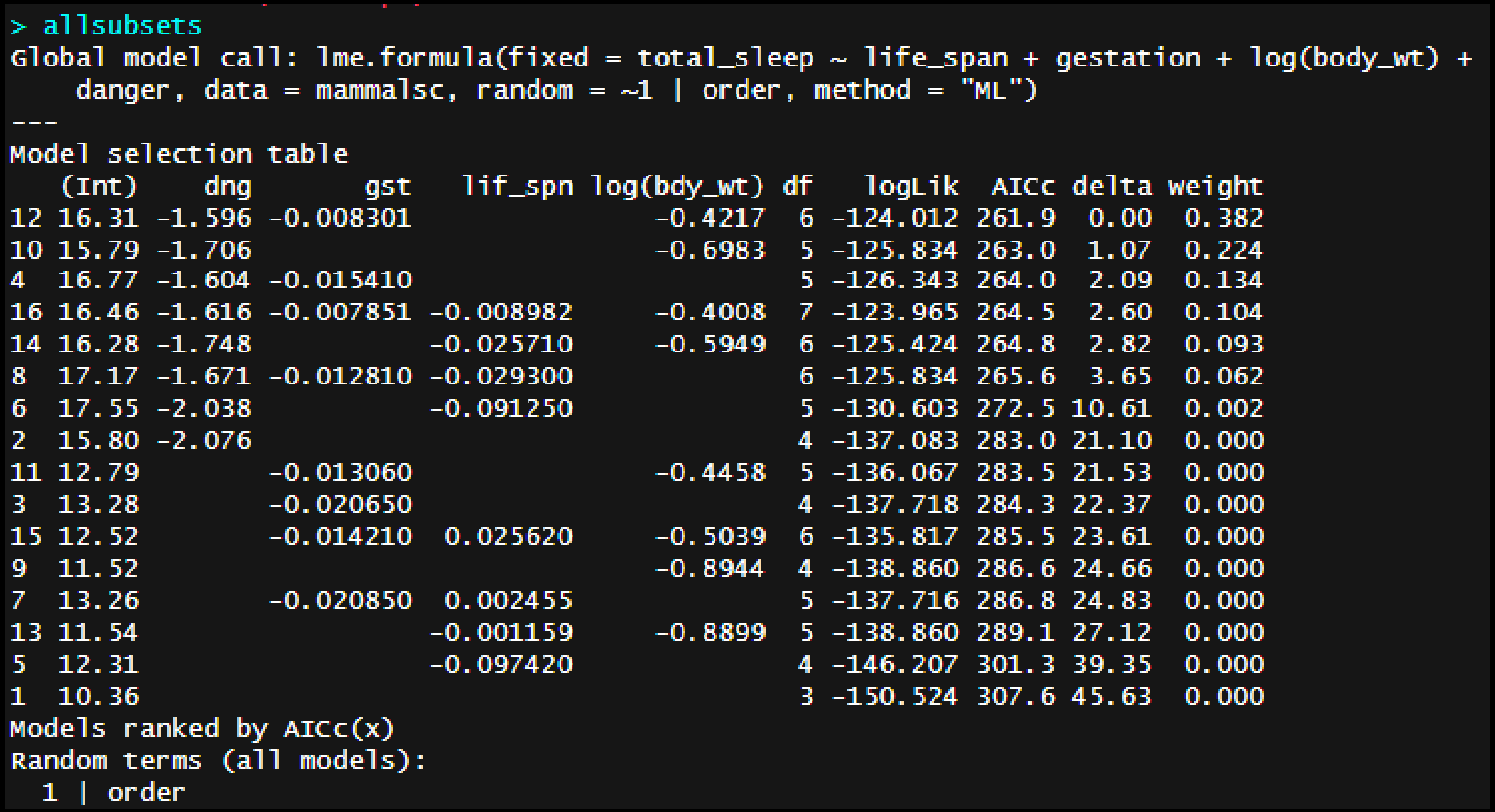
Multi-model inference

1. Write down all biologically plausible models (or all subsets)
2. Fit these models and calculate AIC for each
3. Compute model weights using the AIC values, reflecting the "relative plausibility" of the different models
4. Identify model set based on change in AIC (ΔAIC) relative to best model (Δ4 or 6 generally recommended)
5. Calculate weighted predictions and SEs that reflect model uncertainty and sampling uncertainty

Multi-model inference



```
library(MuMIn)
options(na.action = "na.fail")
fullmodel<-lme(total_sleep ~ life_span + gestation + log(body_wt) +danger, random=~1|order, data=mammalsc, method="ML")
allsubsets <- dredge(fullmodel)
allsubsets
```

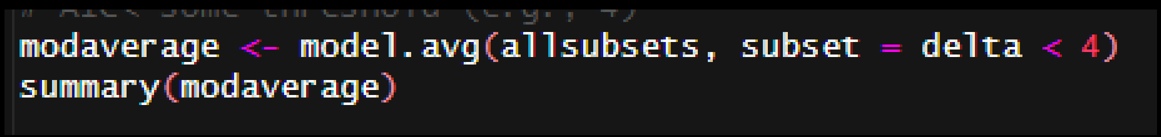


```
> allsubsets
Global model call: lme.formula(fixed = total_sleep ~ life_span + gestation + log(body_wt) +
    danger, data = mammalsc, random = ~1 | order, method = "ML")
---
Model selection table
   (Int)    dng       gst   lif_spn log(bdy_wt) df   logLik  AICc delta weight
12 16.31 -1.596 -0.008301               -0.4217  6 -124.012 261.9  0.00  0.382
10 15.79 -1.706                         -0.6983  5 -125.834 263.0  1.07  0.224
4  16.77 -1.604 -0.015410                        5 -126.343 264.0  2.09  0.134
16 16.46 -1.616 -0.007851 -0.008982     -0.4008  7 -123.965 264.5  2.60  0.104
14 16.28 -1.748           -0.025710     -0.5949  6 -125.424 264.8  2.82  0.093
8  17.17 -1.671 -0.012810 -0.029300              6 -125.834 265.6  3.65  0.062
6  17.55 -2.038           -0.091250              5 -130.603 272.5 10.61  0.002
2  15.80 -2.076                                  4 -137.083 283.0 21.10  0.000
11 12.79        -0.013060               -0.4458  5 -136.067 283.5 21.53  0.000
3  13.28        -0.020650                        4 -137.718 284.3 22.37  0.000
15 12.52        -0.014210  0.025620     -0.5039  6 -135.817 285.5 23.61  0.000
9  11.52                                -0.8944  4 -138.860 286.6 24.66  0.000
7  13.26        -0.020850  0.002455              5 -137.716 286.8 24.83  0.000
13 11.54                  -0.001159     -0.8899  5 -138.860 289.1 27.12  0.000
5  12.31                  -0.097420              4 -146.207 301.3 39.35  0.000
1  10.36                                         3 -150.524 307.6 45.63  0.000
Models ranked by AICc(x)
Random terms (all models):
  1 | order
```

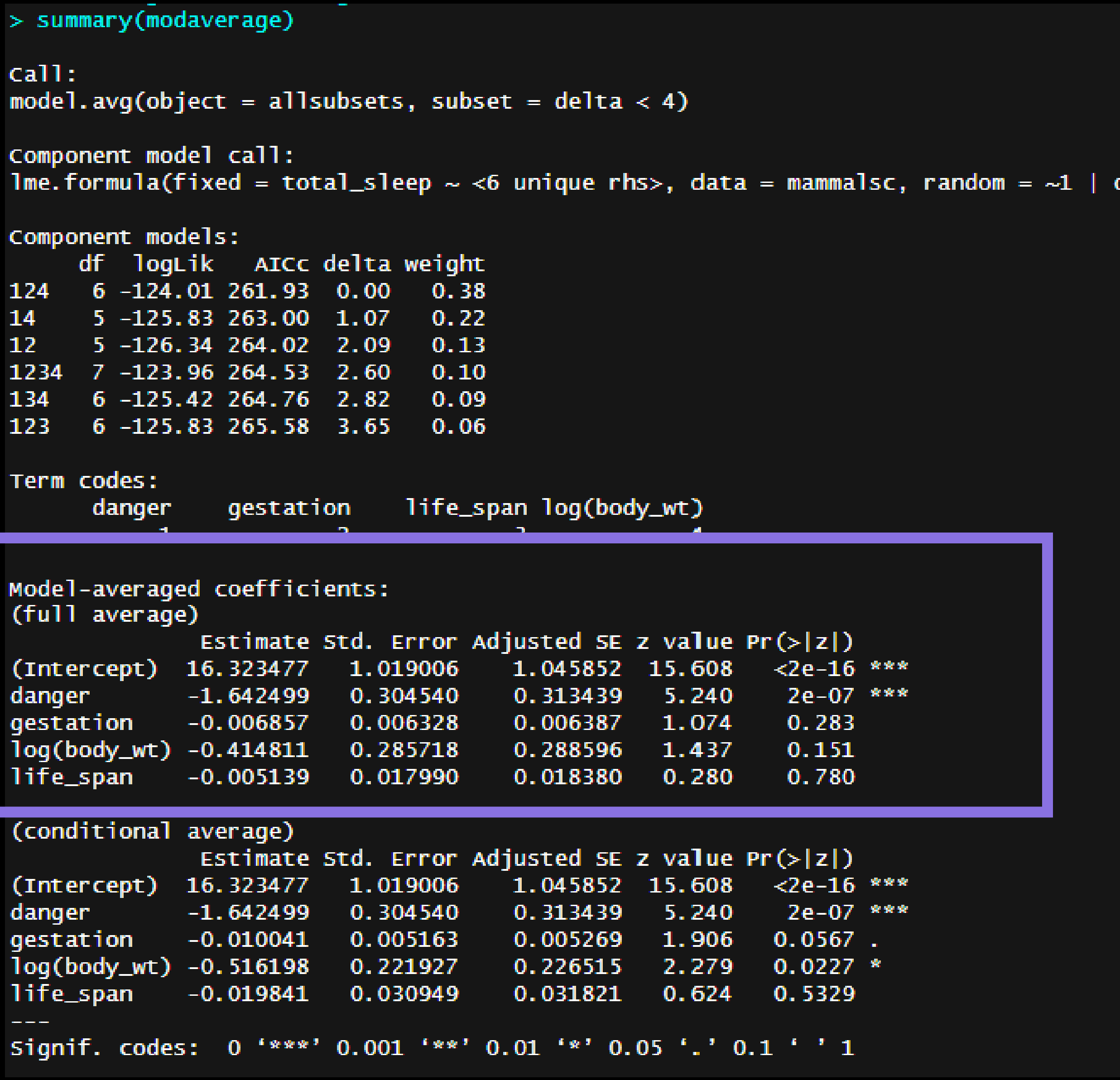
dredge = all possible models

AIC ranked from best to worst

Multi-model inference



```
# AIC< some threshold (e.g., 4)
modaverage <- model.avg(allsubsets, subset = delta < 4)
summary(modaverage)
```



```
> summary(modaverage)

Call:
model.avg(object = allsubsets, subset = delta < 4)

Component model call:
lme.formula(fixed = total_sleep ~ <6 unique rhs>, data = mammalsc, random = ~1 | c

Component models:
     df  logLik   AICc delta weight
124   6 -124.01 261.93  0.00   0.38
14    5 -125.83 263.00  1.07   0.22
12    5 -126.34 264.02  2.09   0.13
1234  7 -123.96 264.53  2.60   0.10
134   6 -125.42 264.76  2.82   0.09
123   6 -125.83 265.58  3.65   0.06

Term codes:
      danger    gestation    life_span log(body_wt)
           1            2            3            4

Model-averaged coefficients:
(full average)
             Estimate Std. Error Adjusted SE z value Pr(>|z|)
(Intercept)  16.323477   1.019006    1.045852  15.608   <2e-16 ***
danger       -1.642499   0.304540    0.313439   5.240    2e-07 ***
gestation    -0.006857   0.006328    0.006387   1.074    0.283
log(body_wt) -0.414811   0.285718    0.288596   1.437    0.151
life_span    -0.005139   0.017990    0.018380   0.280    0.780

(conditional average)
             Estimate Std. Error Adjusted SE z value Pr(>|z|)
(Intercept)  16.323477   1.019006    1.045852  15.608   <2e-16 ***
danger       -1.642499   0.304540    0.313439   5.240    2e-07 ***
gestation    -0.010041   0.005163    0.005269   1.906   0.0567 .
log(body_wt) -0.516198   0.221927    0.226515   2.279   0.0227 *
life_span    -0.019841   0.030949    0.031821   0.624   0.5329
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Similar estimate to the single model approach

Multi-model inference

- **Exploratory:**
  - **All-subsets** selection of enormous global models containing large numbers of predictors and their interactions makes analyses extremely prone to including uninformative parameters and 'overfitted' models
- Inference:
  - consider only a handful of hypotheses and then build a single statistical model to reflect each hypothesis (candidate models)

Model selection depends on: question, data, and goals

Selection

- Good for large number of predictors
- Familiar yes/no for each parameter
- LRT: good for inference and prediction
- Stepwise algorithms: good for exploration

Single Model

- NOT good for large number of predictors
- Each predictor should be hypothesis driven
- Greater uncertainty of parameter estimates
- Good for inference

Multi-model

- Good for large number of predictors – but overfitting!
- More realistic (some support for many models)
- All subsest = good for exploration
- Candidate models = inference and prediction

Cross-validation

You found your “best” model!

But is that model a “good” predictor?

Cross validation

You can also do a variety of cross-validation experimental designs to test predictions

1. Refit the model on a new dataset

- Build model on one dataset (training)
- Test model predictions on another dataset (test)

2. Split data into training and testing dataset:

- Randomly subset some % (usually 50%)
- Grouped data (random effect):
  - Train model on 6 ‘sites’, and Hold-out 1 ‘site’ to test model
  - Randomly subset 50% of each ‘sites’ data points to put in test or train

3. Leave-one-out cross-validation:

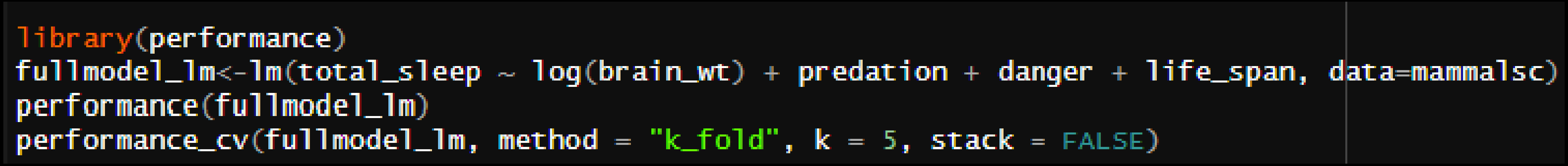
- Split data into training (all but one observation) and predict the one observation; repeat for each observation in your dataset

K-fold cross-validation

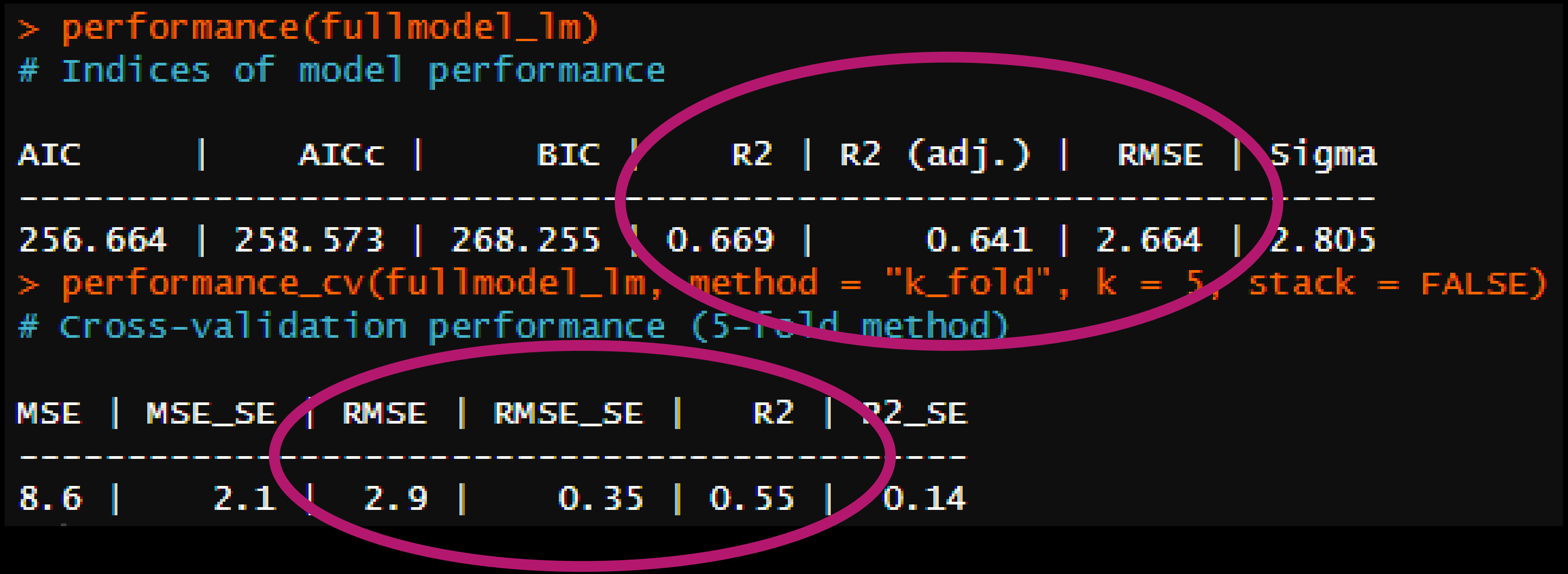
- Data are partitioned into k different subsets, referred to as folds.
- Model is trained using all data except for 1 of the folds, which is used as the test data
- Process is repeated (k-fold repeated cross-validation).
- Results are pooled to evaluate model performance.

- Repeated k-fold cross-validation is generally recommended when the dataset is limited, and you want a more reliable estimate of model performance.

K-fold cross-validation – LM

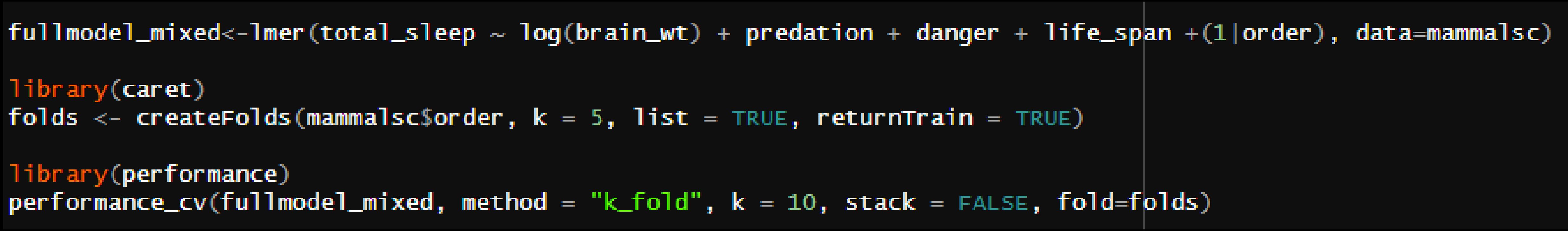


```
library(performance)
fullmodel_lm<-lm(total_sleep ~ log(brain_wt) + predation + danger + life_span, data=mammalsc)
performance(fullmodel_lm)
performance_cv(fullmodel_lm, method = "k_fold", k = 5, stack = FALSE)
```

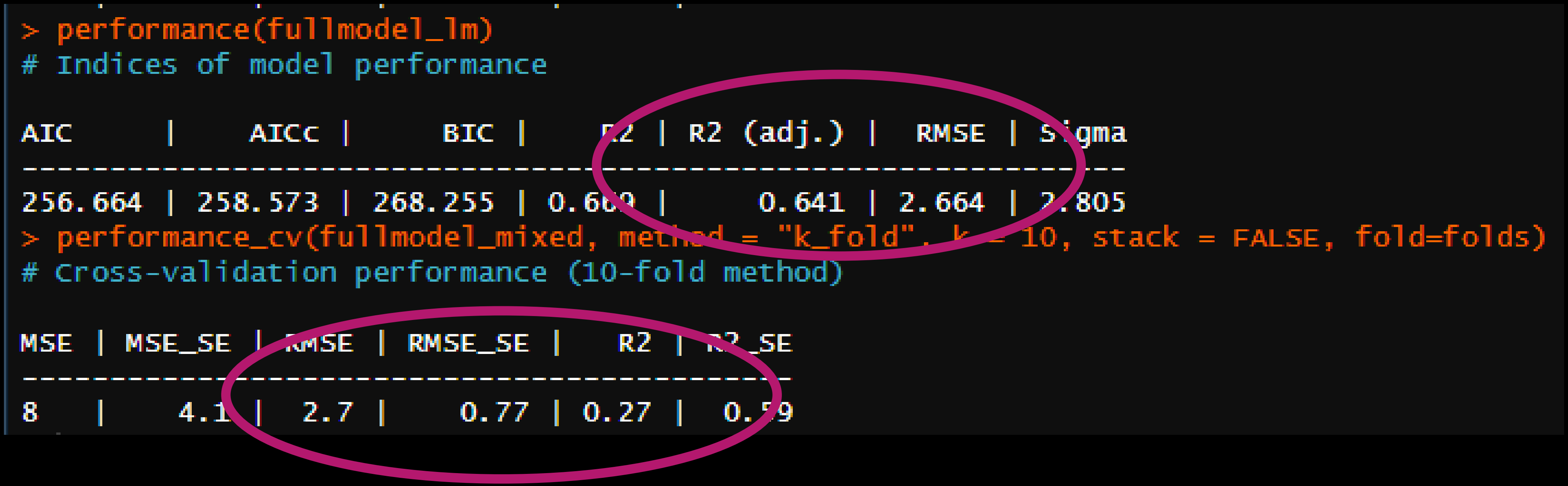


```
> performance(fullmodel_lm)
# Indices of model performance

AIC     |    AICc |     BIC |    R2 | R2 (adj.) |  RMSE | Sigma
---------------------------------------------------------------
256.664 | 258.573 | 268.255 | 0.669 |     0.641 | 2.664 | 2.805
> performance_cv(fullmodel_lm, method = "k_fold", k = 5, stack = FALSE)
# Cross-validation performance (5-fold method)

MSE | MSE_SE | RMSE | RMSE_SE |   R2 | R2_SE
--------------------------------------------
8.6 |    2.1 |  2.9 |    0.35 | 0.55 |  0.14
```

K-fold cross-validation – LME



```
fullmodel_mixed<-lmer(total_sleep ~ log(brain_wt) + predation + danger + life_span +(1|order), data=mammalsc)

library(caret)
folds <- createFolds(mammalsc$order, k = 5, list = TRUE, returnTrain = TRUE)

library(performance)
performance_cv(fullmodel_mixed, method = "k_fold", k = 10, stack = FALSE, fold=folds)
```



```
> performance(fullmodel_lm)
# Indices of model performance

AIC      |    AICc |     BIC |    R2 | R2 (adj.) |  RMSE | Sigma
-----------------------------------------------------------------
256.664 | 258.573 | 268.255 | 0.669 |     0.641 | 2.664 | 2.805
> performance_cv(fullmodel_mixed, method = "k_fold", k = 10, stack = FALSE, fold=folds)
# Cross-validation performance (10-fold method)

MSE | MSE_SE | RMSE | RMSE_SE |   R2 | R2_SE
---------------------------------------------
8   |    4.1 |  2.7 |    0.77 | 0.27 |  0.59
```

Pros and Cons of Cross-validation

- Pros
  - Assesses ability to predict on new data
- Cons
  - Takes more time in analysis and data collection
  - Reduces samples (and groups) that model is trained on
- Some scientists don’t really want to know their model has very low ability to predict on new data – sometimes it doesn’t matter

Model Selection Summary

- There is no one correct way; depends on question, data, and goals
- Be transparent on the model selection process, packages used, and why parameters were included or excluded
- If random factors are introduced by design, then you should always include them in your model
- Start with a strong a priori hypothesis – use biological knowledge
- Even if the analysis selects a ‘best’ model or an average of a set of models this does not mean that the model is any good in an explanatory or predictive sense