Estimating Model Parameters

Mark D. Hanigan
Professor
Department of Dairy Science
Virginia Tech

NANP Modeling Subcommittee

Background

- Data Interpretation → Knowledge/Understanding
  - Means separation: Trt A ≠ Trt B
  - Regression slope or intercept
    - Slope differs from 0
    - Slope varies by treatment
    - Slope is a function of other factors
  - Biological process
    - ≠ 0; it exists
    - Is a function of X, Y, or Z
    - Is affected by treatment
- Predict outcomes → a working model
  - Milk fat output = $\alpha(\text{FAIn}) - \beta(\text{C18:2In}) + \chi(\text{NDFIn}) + \delta$

Models Require Parameters

$$\hat{Y}_i = a\left(X_i\right) + b$$

$$res_i = Y_i - \hat{Y}_i$$

$$\sum_{i=1}^{N_i}\left(Y_i - \hat{Y}_i\right)^2 = SSR$$

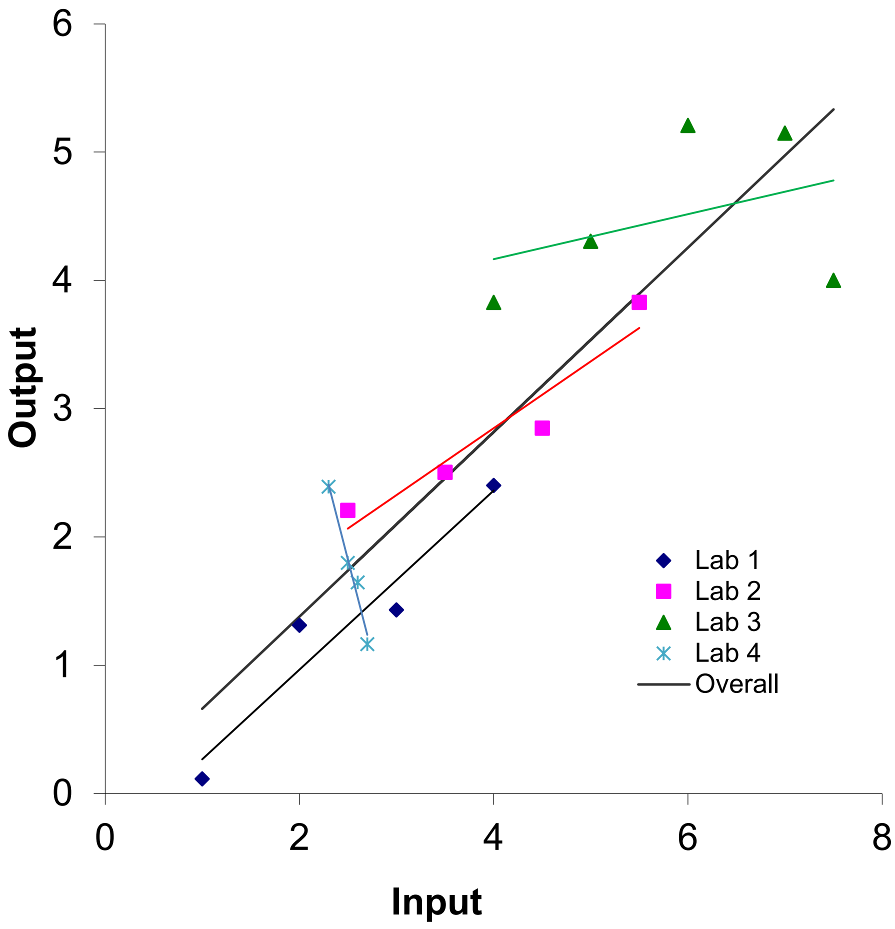
- (a, b) defined by the minimum of SSR
  - (a, b) fitted to data
  - Minimize residuals
- Change in SSR as (a, b) are varied
  - d(SSR)/d(a,b) = Hessian
    - Analytical
    - Numerical (finite diff)
  - SE and *P*
- Optimization methods
  - Slope based → Hessian
    - Quadratics or simplex
  - Random exploration
    - genetic algorithms

What Can Go Wrong?

- Data Quality
  - Measurement variance
  - Outliers
  - Lack of Range
  - High leverage points
  - Data not normally distributed
  - Inadequate observations
  - Undefined factors

- Extreme caution with random effects
  - Global slope should = w/in study slopes
  - Random effects unknown in the field

- Model Structure
  - Wrong model
  - Not enough complexity
  - Too much complexity



Solutions

- Data Normality
  - Transform: log or other
  - Non-parametric approach
- Outliers & high leverage
  - Remove based on residuals
  - Log-likelihood function
  - also solves normality if log normal data

$$LLF = \frac{1}{2}\sum_{j=1}^{r}\left[n_j\{\log(2\pi)+1\}+n_j\log\left(\frac{1}{n_j}\sum_{i=1}^{n_j}\frac{\left(Y_{i,j}-\hat{Y}_{i,j}\right)^2}{\hat{Y}_{i,j}}\right)\right]$$

- Other effects
  - Represent in the model?
  - Adjust the data for random effects
    - Study
    - Location
    - Laboratory
    - Student ?-)

where i=obs and j=pred variables

Data Weighting

- Why?
  - Confidence($LSM_{N=4}$ = $LSM_{N=12}$)?
  - Confidence(Latin $Sq_{N=6}$ = Rand $Block_{N=6}$)?
  - Confidence(Lab 1 = Lab 2)?

- Solution
  - Weight the data
    - 1/sqrt(SEM) captures all of the above (in a perfect world)
    - SEM vs SED vs SD: must convert to a common reference
    - Older fixed effects models ≠ newer random effects models
    - SAS GLM problem
      - SEM under-reported for random effect models for repeated measure designs
        - » LS, crossover, Youden squares
      - ID software, procedure, fixed vs random, design and attempt to correct
    - Model specification problems or reporting errors (SEM_{Milk}=0.1 kg/d???)
  - Weight by sqrt(N)
    - Highly unlikely to be mis-reported
    - No assumptions or transformations required
    - Captures most of the study design variance
    - Misses laboratory expertise
      - More important for technically difficult measurements, i.e. ruminal outflow

L1_5: Meta Analysis using Mixed Models

- Open and execute ‘Load Observed Data.R’
  - Data loaded into the “o” dataframe
  - >ls(o) to see the list of variables
  - >head(o) to see a sample of the data
  - >o to see all of the data
  - >length(o$TID) for N

- Open Lesson 1 Exercise 5 script
  - >sqrt(o$N_Study) to see the variance in potential weighting
    - Big difference?
  - >lmod <- lmer(Obs_RUPIn ~ Dt_CPIn + (1|PubID), data=o, weights = sqrt(N_study), REML=FALSE)
    - lmer = linear mixed effect regression (lme4 package)
    - Obs_RUPIn = Total N – Micr N – predicted Endog N
    - REML = FALSE yields a Max Likelihood solution
    - What is the DC estimate? And SE?

Non-linear Mixed Effects Model

- NLin models has more complexity
  - formnlmer <- ~ Int + Dt_CPAIn * KpA + Dt_CPBIn * KpB/(KdRUP + KpB) + Dt_CPCIn * KpC
  - form.d <- deriv(formnlmer, parms1, function.arg = args1)
  - Obs_RUPIn ~ form.d(Dt_CPAIn, Dt_CPBIn, Dt_CPCIn, KdRUP,Int, KpA, KpB, KpC) ~ (Int|PubID)
  - Solving for:
    - Int
    - Kp_a
    - Kp_b
    - Kp_c
    - Kd = in situ observations
- Requires an intercept for ME
- Allows data weighting, but doesn’t converge for this problem
- What are the parm estimates?

Model Solution Evaluation

Model	DC	Kp/Kp
AIC	59.7	71.7
BIC	68.5	86.6
Log Likelihood	-25.8	-28.3
RMSE (re.form=NA)	47.4%	
RMSE (re.form=T)	18.9%	22.3%
CCC (re.form=NA)	0.489	
CCC (re.form=T)	0.917	0.920
Mean Bias	0.0%	0.0%
Slope Bias	1.9%	1.8%

- corr = with random effects
- uncorr likely reflects novel/field performance

- Lower AIC/BIC is better
- Greater Log Likelihood is better

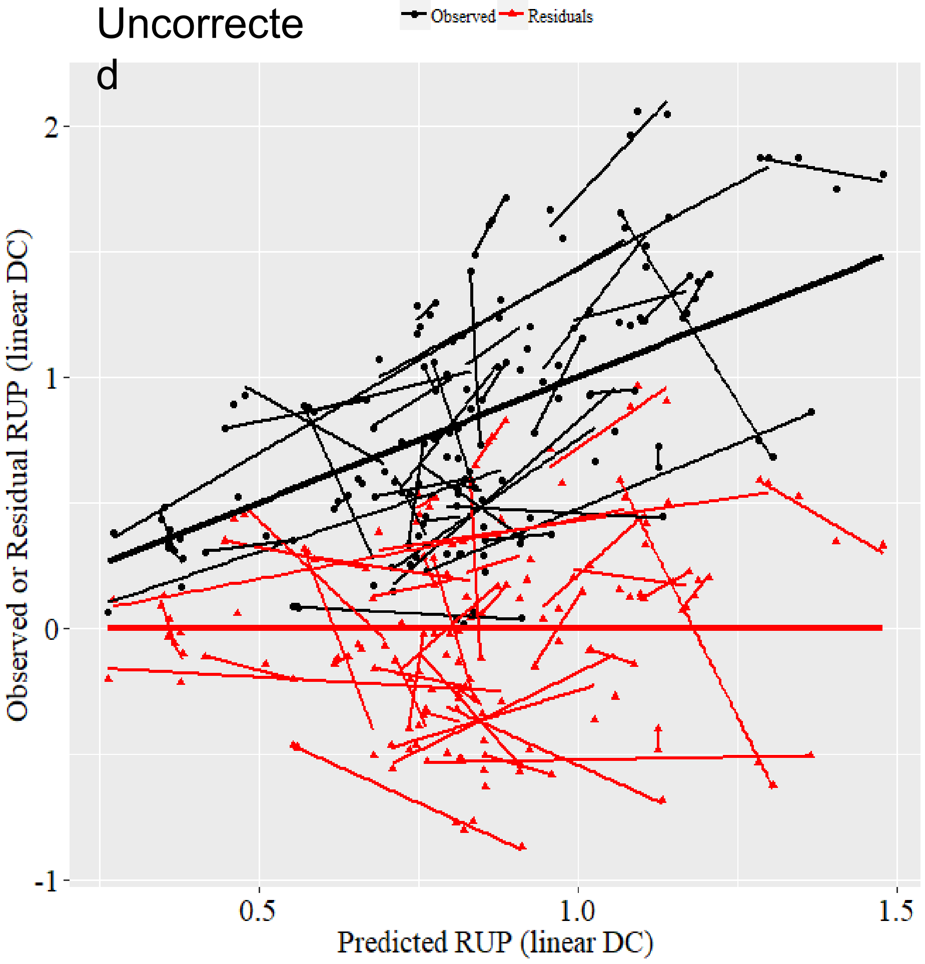
- Simple DC with an intercept is better

- RMSE Anal and Plot Residuals
  - Load RMSE_CCC Functions and execute
  - Load Graph Residuals Functions and exec
  - obs <- o$Obs_RUPIn
  - predl <- predict(lmod, re.form=NA)
    - re.form=NA controls use of random effects (false)
    - Doesn’t work correctly for nlmer
    - re.form=TRUE to correct for random effects
  - RMSE(obs,predl)
  - resprgrpplt(obs,predl,o$PubID,"RUP (linear DC)")

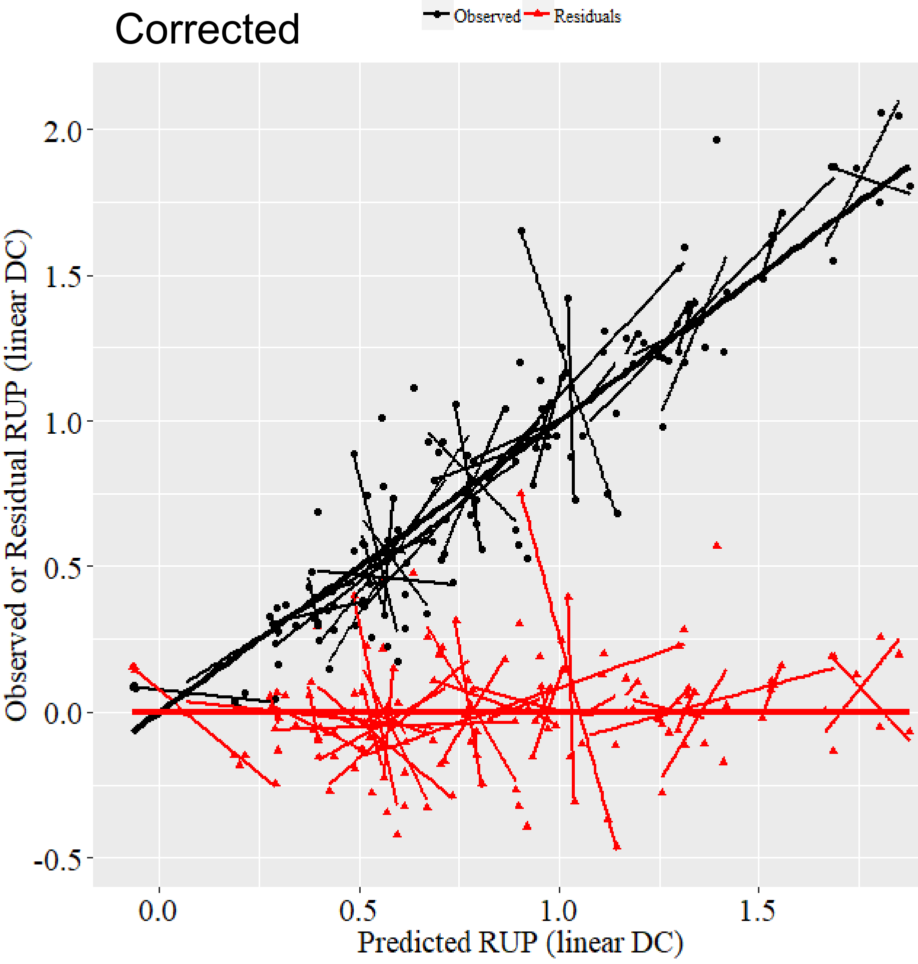
- Plot or regress against all inputs
- Evaluate against other possible inputs

Residuals for the Linear Model

Uncorrecte
d



Corrected



Centering Variables – Script 6

- Easy with a function

```
c. <- function (x) scale(x, scale = FALSE)     #Center x
yielding c.x
```

```
mTPmod <- lmer(Obs_MilkTP_g ~ …  + Abs_EAA_g +
Parity_rl + c.(Year) + (1|PubID), data=o, weights =
sqrt(N_study), REML=FALSE)
```

- Term is independent of the intercept
- Can be removed or added without affecting Intercept
- Does not change correlation with other Parms

Dynamic Model Fitting

Many pieces – test each

1. The model
   - Typically a Rate/State approach
   - Define the flux (rate) equations
   - Define the differential eqns for each pool (state variables)
   - Provide initial pool size estimates
   - Provide initial parameter estimates
   - Provide model inputs
2. Observed data (as before)
3. Residual error data (Obj.f)
   - Run the model to simulate each subject
   - Collect predicted values for appropriate time points (pred.f)
   - Calculate residuals (Obs – Pred)
   - weight? and scale??
4. Optimizer
   - The model function
   - Objective function (Obj.f)
   - List of parameters to fit
   - Initial parameter estimates
   - Parameter bounds

A dynamic rumen model: script 1 3

- Script 3 defines a 3-pool rumen model with linear absorption from the SI
  - CP
  - CHO
  - MiCP
  - MiCP a fn of RDP and RDCHO
  - Absorption a function of RUP and MiCP
  - Uses numerical integration to predict steady state

- Execute the script to define the model function
  - Note model behavior in plots (back arrow to view more)
  - Why are there changes over time in the fluxes?
  - How would we compare model output to animal observations?
  - Where are these data coming from? Type >out
- What do you expect to happen if the rate of passage increases?
  - Kp=0.06
  - Call the model function with revised Kp and collect in out2
  - Compare out with out 2
  - What happens to the ruminal DC for CP and CHO as Kp increases?

Parameter Estimation: Script 1 7

- Contains the code to fit the model to the data

1. Load initial parameters and test model output to verify
2. Select parms to fit
   - Start with 1 parm (Kp)
3. Provide initial parameter guesses
4. Create lb and ub vectors
5. Select obs vars to fit against
6. Specify which model times to use for comparison to data
   - Start with 1 obs var (FCpSI)
7. Execute Obj.f and pred.f functions to get them in memory
8. Scale the residuals = TRUE
9. Execute modFit statement to fit the parameters to the data
10. Review output
    - >summary(m1)
    - Converged? Note the list elements in m1
      - >m1$info
      - Print(m1) to list all element contents
    - Logical answers?
    - SE acceptable (<50% of the estimate)

Other Observations/Questions

- What is the RMSE?
  - Transfer final parms to the model inputs
  - Collect pred vals using pred.f function
  - Calculate residuals
  - Execute RMSE function
  - Good or bad?
  - Is RUP flow biased?

- What is the CCC?
  - Good or bad?

- Plot residuals
  - Load plotting functions
  - resprplt(obs,pred,”RUP”) to plot residuals without lines by study
  - Make studies vector >studies <- o[, "PubID"]
  - resprgrpplt(obs,pred,studies,”RUP”) to plot with lines by study

Fit Other Parms

- Update the Kp value in the parameters vector list
- Select KdCho and fit it against FChoSi
- Update parameters and try fitting Kp and KdCho at the same time
- Update KdCho in parameters and repeat to fit KRdp to FCpMiSi
- Update parameters and fit all 3 at once

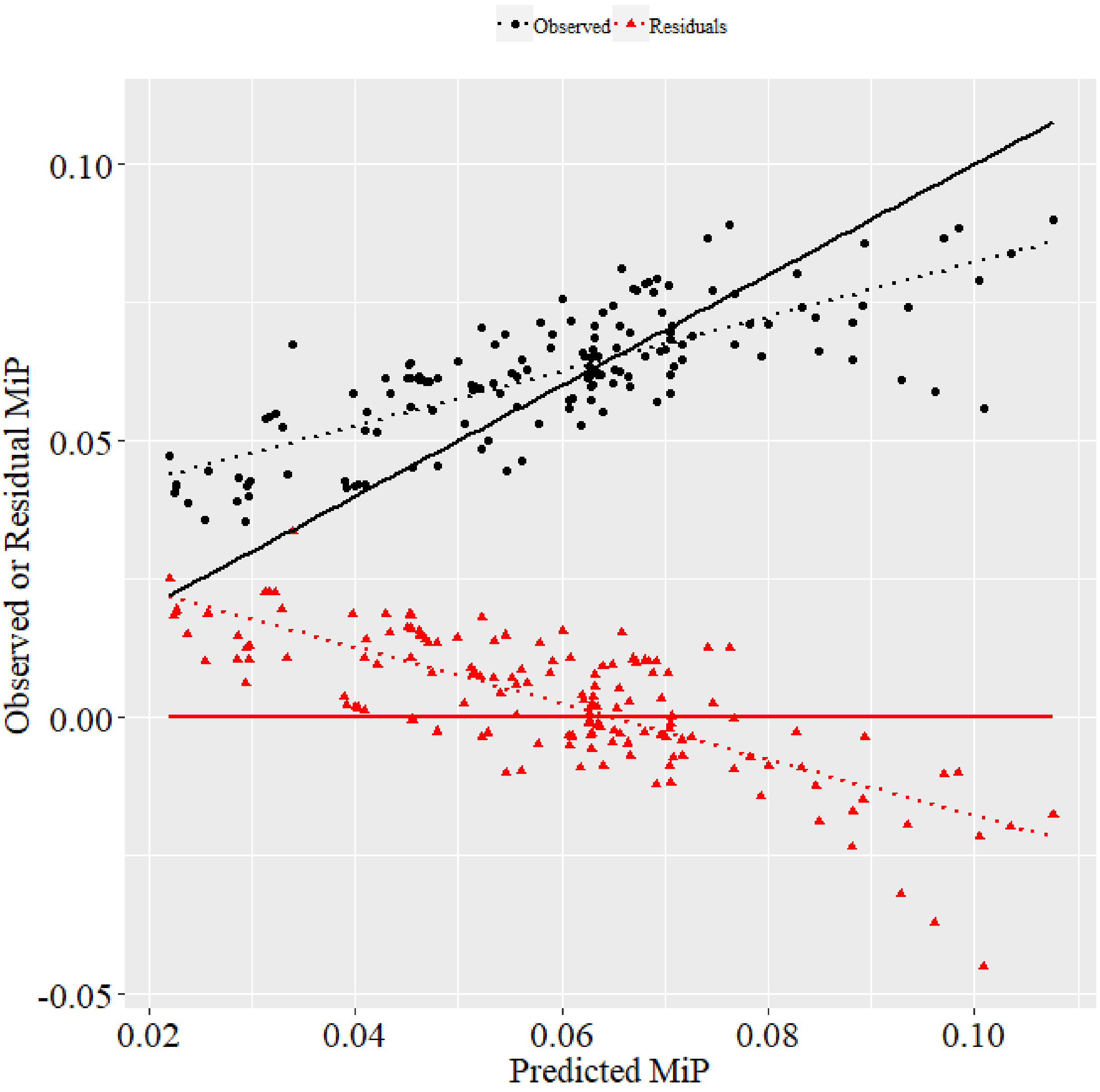
SE and Correlations

```
Parameters:
       Estimate Std. Error t value Pr(>|t|)    
Kp    4.478e-02  7.540e-03    5.94 5.72e-09 ***
KdCho 4.862e-02  8.200e-03    5.93 6.06e-09 ***
KRdp  5.493e-16  3.868e-03    0.00        1    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1

Residual standard error: 0.002585 on 450 degrees of
freedom

Parameter correlation:
           Kp   KdCho    KRdp
Kp    1.00000 0.99855 0.07624
KdCho 0.99855 1.00000 0.07996
KRdp  0.07624 0.07996 1.00000
```

Predicted Microbial CP Flow



RMSE and CCC

Microbial CP Flow

```
 RMSE(obs$FCpSi,pred$FCpSi)
           Statistic     Values
1                  N 151.000000
2      Observed Mean   0.045886
3     Predicted Mean   0.044131
4               RMSE   0.004714
5       RMSE, % mean  10.272802
6   Mean Bias, % MSE  13.865398
7  Slope Bias, % MSE  26.464249
8  Dispersion, % MSE  59.670353
9          Mean Bias   0.001755
10        Slope Bias  -0.157152
11       P-Mean Bias   0.000100
12      P-Slope Bias   0.000100
13               RSR   0.347864
14               CCC   0.947544
```

CHO Outflow

```
RMSE(obs$FChoSi,pred$FChoSi)
           Statistic     Values
1                  N 151.000000
2      Observed Mean   0.207109
3     Predicted Mean   0.204671
4               RMSE   0.021165
5       RMSE, % mean  10.219235
6   Mean Bias, % MSE   1.326393
7  Slope Bias, % MSE  12.814169
8  Dispersion, % MSE  85.859438
9          Mean Bias   0.002438
10        Slope Bias  -0.114925
11       P-Mean Bias   0.157690
12      P-Slope Bias   0.000100
13               RSR   0.342693
14               CCC   0.944976
```

Next

- Fit intestinal absorption coefficients against fecal outputs
- Address the slope bias in microbial CP flow
    - Other drivers may be required
    - Plot residuals against other available observations (hypothesis testing)
- Remove the RDP driver in MiCP??
- Check for normality of residuals
- Check for residuals outliers and remove?
- Finalize fits

Notes

- Need to code an LLF and try with that