

Appendix A: Modeling Wing Morphology

for ‘paper_title’; Bernat, AV, Cenzer, ML

Contents

Details of the Analyses	2
Description of the Data	2
Abbreviations Used in the Data and Code	2
Data Cleaning And Exploration	2
Read Libraries	2
Read Source Files	3
Read the Data	3
Histograms of Wing Morph Data	4
Barplots	5
Plot 1: Collection numbers grouped by Population and Datetime	5
Plot 2: Collection numbers grouped by Host Plant and Datetime	5
Plot 3: Collection numbers grouped by Sex and Datetime	6
Regression Modeling	6
Long-Wing Morph Frequency	6
Best Fit	7
Long-Wing Morph Variance	8
Best Fit	10
Wing-to-Body Ratio	10
Best Fit	12
Wing-to-Body Ratio Variance	13
Best Fit	14
LOESS & Linear Regression Plots	15
Wing Morph Frequency	15
Group significant elements	15
Check for LOESS Residuals	15
Figure: Panels A, B, C, D (long-wing morph freq with month) & E (long-wing morph freq with year)	16
Wing-to-Body Ratio	19
Group significant elements	19
Check for LOESS Residuals	19
Figure: Panels A, B, (wing-to-body ratio with month) & C (wing-to-body ratio with year)	19

Details of the Analyses

This document was generated by R Markdown on 2021-07-30 using R version 4.0.5 (2021-03-31). The document provides the step-by-step analytical methods used in the manuscript by Anastasia Bernat (AVB) and Meredith Cenzer (MLC). Multiple draft scripts were written by AVB between 2021-03-01 and 2021-07-26 until being distilled and compiled by AVB and code reviewed by MLC at the University of Chicago into this comprehensive script. All draft scripts can be viewed in the GitHub repository, SBB-dispersal (<https://github.com/mlcenzer/SBB-dispersal>), within the directory `avbernat_working_on > All_Morphology > stats`.

All code and output from the statistical analyses is shown. Code for data cleaning and the generation of plots is not displayed, but can be viewed in the `wing_summary.Rmd` file and its accompanying three sourced scripts. To repeat analyses and the generation of plots, all data files and sourced scripts should follow the directory structure presented in the SBB-dispersal repository.

Description of the Data

This document analyzes two main datasets, `raw_data` and `data_long`. The `raw_data` set provides morphology measurements for each soapberry bug, *Jadera haematoloma*, collected and measured between the April 2013 and February 2020. There are four morphology measurements: beak length, thorax width, wing length, and body length. The sex, wing morph (long-winged, shot-winged, or ambiguously-winged), and host plant the bug was collected from as well as the month and year each bug was collected in was recorded. The `data_long` set provides the same recordings as the `raw_data` set, but it has been filtered for only long-winged soapberry bugs.

Abbreviations Used in the Data and Code

- **SBB** - soapberry bug, *Jadera haematoloma*
- **S** - short-winged morph
- **L** - long-winged morph
- **LS** or **SL** - ambiguous wing morph
- **pophost** - the host plant soapberry bugs were collected from, which was either *Koelerutera elegans* or *Cardiospermum corindum*, occasionally called (and abbreviated) as goldenrain tree (GRT) or balloon vine (BV), respectively
- **months_since_start** - proxy for year where the first collection occurred on April 2013
- **month_of_year** - proxy for season where collections occurred only in months February, April, May, August, September, October, and September
- **wing2body** - a computed and unitless column that calculates the wing length divided by the body length of a soapberry bug
- **wing2thorax** - a computed and unitless column that calculates the wing length divided by the thorax width of a soapberry bug
- **sd** - standard deviation
- **se** - standard error
- **c** - a column name that ends in `_c` is a column that has been centered. Example columns: `wing2body_c`, `month_of_year_c`, and `months_since_start_c`
- **binom** - a column name that ends in `_binom` is a column that has been recodified into binary data (0's and 1's). Example columns: `sex_binom`, `pophost_binom`, and `wing_morph_binom`

Data Cleaning And Exploration

Read Libraries

The occurrence of long-wing morphology and the wing-to-body ratio of *J. haematoloma* were analyzed using multivariate, generalized linear modeling (GLM) as implemented in the R packages `lme4` and `binom`. The `dplyr` package helped speed data manipulation processes by grouping data quickly. All

plots, except the histograms, were generated using `ggplot` libraries and helper functions found in R packages `ggformula` and `cowplot`.

Additional R packages not shown below, but embedded in the sourced scripts are `zoo` and `lubridate`, which aid in data manipulation and datetime manipulation, respectively.

```
library(lme4) # fit regressions
library(dplyr) # data manipulation
library(ggformula) # ggplot plotting
library(cowplot) # ggplot helper functions to arrange multi-panel figures
library(binom) # binomial confidence intervals
```

Read Source Files

Each sourced script below aides in either data cleaning (`read_morph_data()`, `remove_torn_wings()`) or multivariate GLM (`model_comparisonsAIC()`, `get_model_probs()`). Additionally, `model_comparisonsAIC()` takes in the path of a generic multi-factor script specific to the GLM family and link function needed to build the predictive models. All aforementioned, sourced scripts are located in the **Rsrc** folder.

```
source_path = paste0(dir, "/Rsrc/")

script_names = c("clean_morph_data.R", # 1 function: read_morph_data()
                 "remove_torn_wings.R", # 1 function: remove_torn_wings()
                 "compare_models.R",    # 1 function: model_comparisonsAIC()
                 "get_Akaike_weights.R") # 1 function: get_model_probs()

for (script in script_names) {
  path = paste0(source_path, script)
  source(path)
}
```

Read the Data

The morphology data were started in 2013-04-28 and last updated on 2021-05-18. The `read_morph_data()` function standardizes population names, host plant names, and month and year inputs. Month and year inputs are also converted into datetimes. Variables of interest like wing-to-body ratio and wing-to-thorax ratio are also calculated and centered. Then, the full dataset, `raw_data` (n=3532), and a long-winged bug only dataset, `data_long` (n=2096) are returned.

```
data_list = read_morph_data(paste0(dir, "All_Morphology/stats/data/allmorphology05.18.21.csv"))

## number of missing dates: 0
##
## morph types: L S NA LS SL
## recoding missing morph types...
## S if wing2thorax <=2.2, L if wing2thorax >=2.5
##
## ambiguous wing morph bug count: 48
##
## filtered out NA wing2body for data_long...

raw_data = data_list[[1]]
data_long = data_list[[2]] # long-wing bugs only

data_long = remove_torn_wings(data_long)

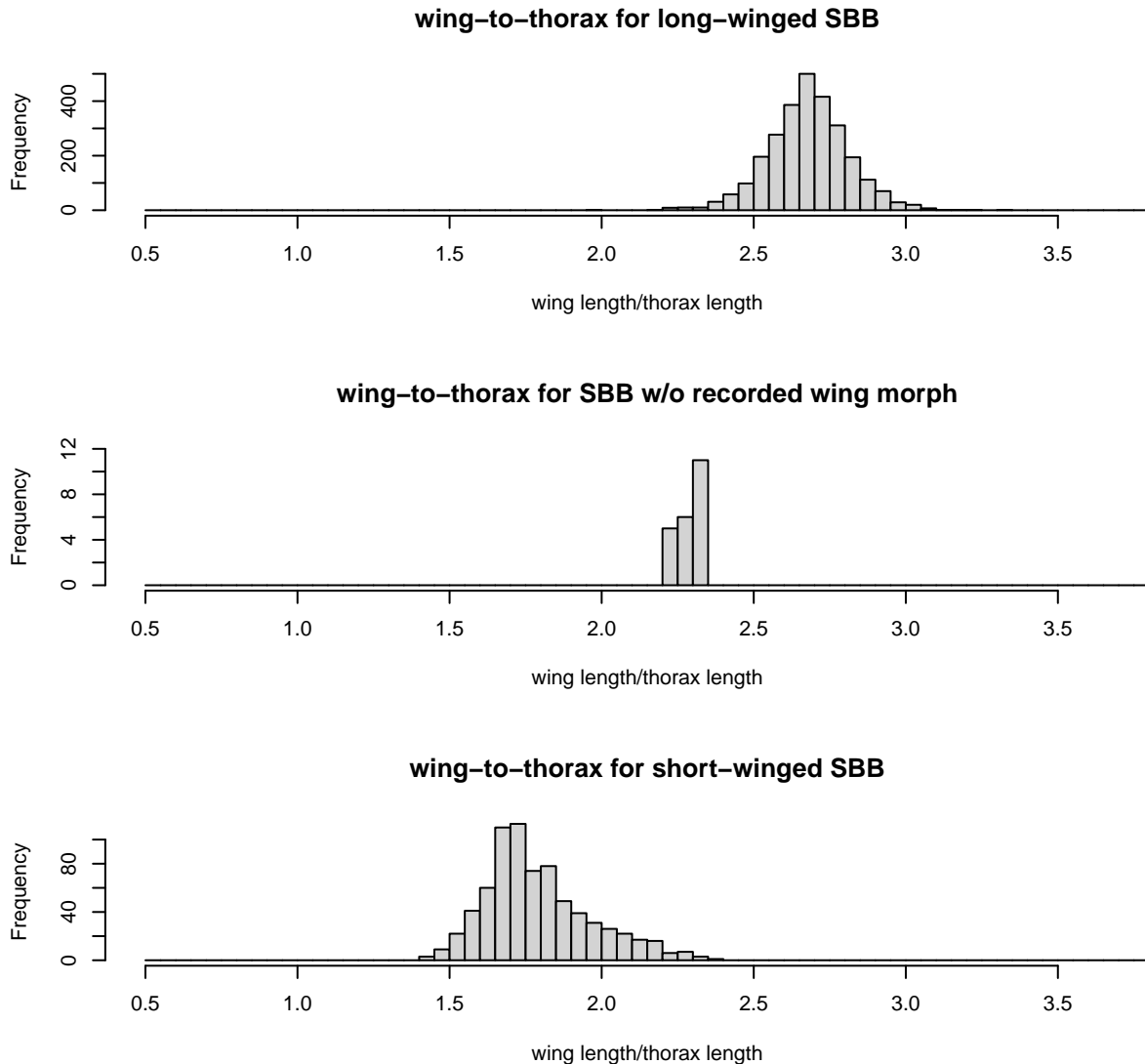
##
```

number of bugs with torn wings: 193

Bugs marked as having torn wings during measurements were only filtered out of the `data_long` dataset (n=1903). That was because `data_long` is used only to analyze the wing-to-body ratio, which was computed for long-winged bugs since no short-winged bugs can fly. `raw_data` is only used to analyze long-wing morph frequency.

Histograms of Wing Morph Data

To better visualize how wing morph relates to another allometric measurement, wing-to-thorax ratio, the following histograms were plotted:

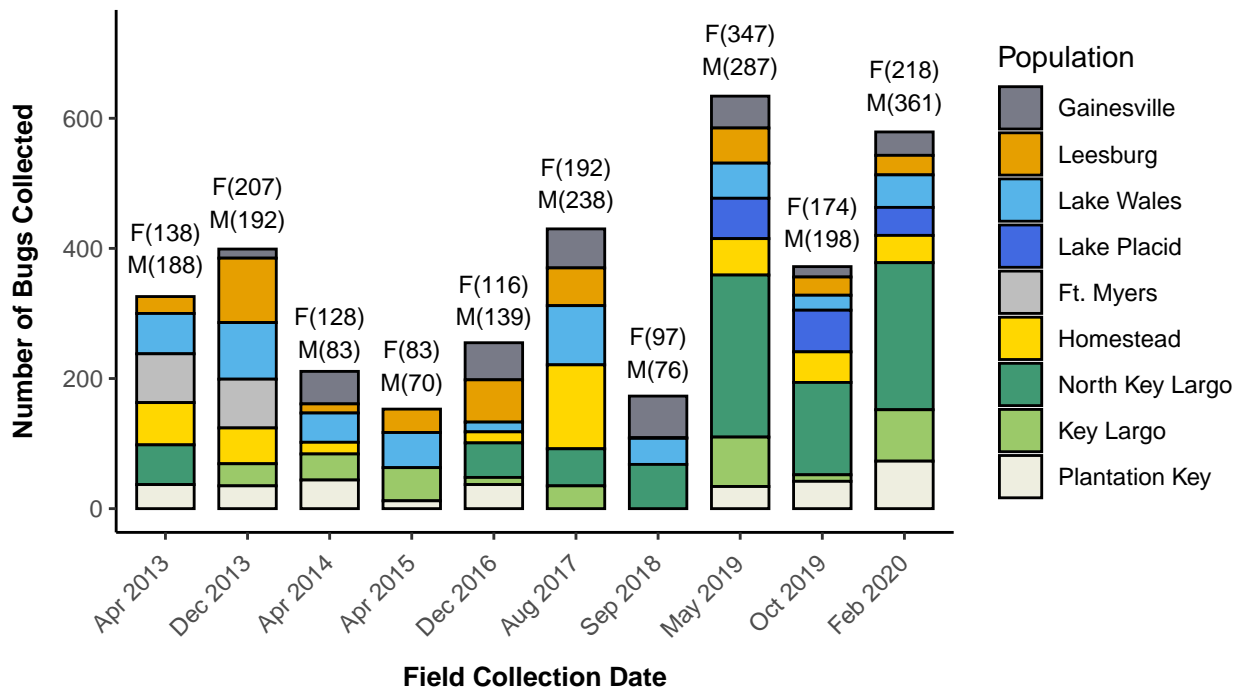


From the histograms, the relationship between wing morph and wing-to-thorax ratio is bimodal. Long-winged bugs have larger wing-to-thorax ratios with a frequency peak around 2.75, whereas short-winged bugs have much smaller wing-to-thorax ratios with a frequency peak around 1.75. It is then noticeable that there are 22 bugs who had not been identified as either S or L during measurements, but cannot be categorized into S or L based on wing-to-thorax bimodality because these data reside in-between the two modes.

Barplots

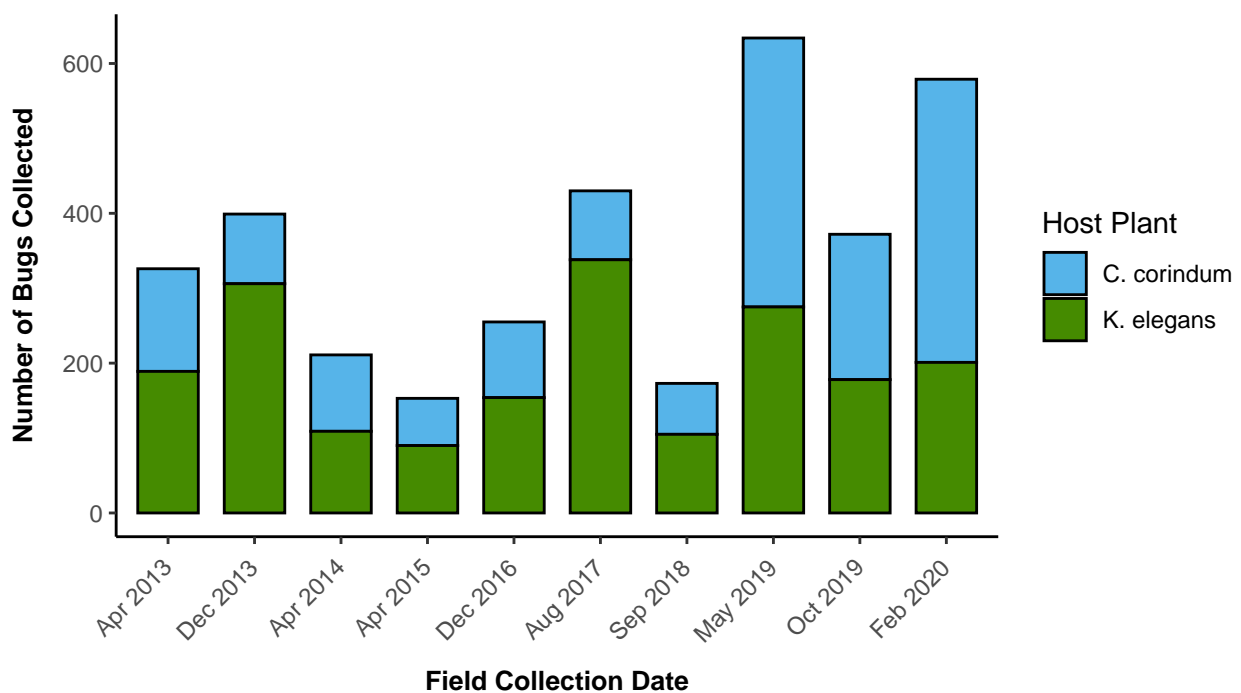
Bugs were collected during different years and months. The barplots below display the bugs collected per **population**, **host plant**, and **sex** across the years and months:

Plot 1: Collection numbers grouped by Population and Datetime

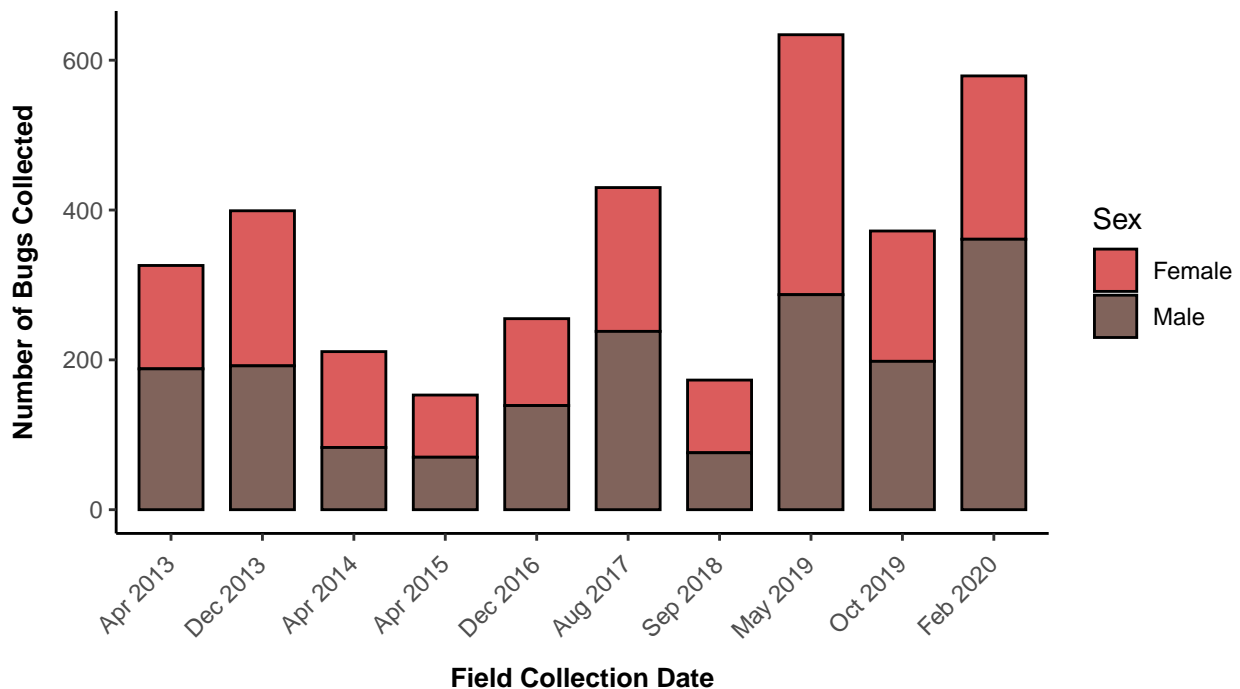


It is noticeable here that there was a heterogeneous population collection pattern across collection datetimes. However, collection numbers by host plant and sex were relatively homogeneous:

Plot 2: Collection numbers grouped by Host Plant and Datetime



Plot 3: Collection numbers grouped by Sex and Datetime



Regression Modeling

Multivariate, GLM was performed using the `glm()` function in the `lme4` package. Models were compared using Akaike Information Criterion (AIC) and model selection was determined using Akaike weights. Model fit was further evaluated between two models using the `anova()` function.

Long-Wing Morph Frequency

We tested how sex, host plant, month, and/or year effected whether a soapberry bug is long-winged (`wing_morph_binom=1`) or short-winged (`wing_morph_binom=0`).

```
data = data.frame(R=raw_data$wing_morph_binom,
                  A=raw_data$sex_binom,
                  B=raw_data$pophost_binom,
                  C=(raw_data$month_of_year),
                  D=raw_data$months_since_start)
```

```
model_script = paste0(source_path,"generic models-binomial glm 4-FF.R")
model_comparisonsAIC(model_script)
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]
## AICs    3145.306  3146.842  3147.157  3147.201  3148.521
## models   98       110       84       107       105
## probs   0.2529382 0.1173602 0.1002697 0.09808583 0.05068685
##
## m98  glm(formula = R ~ A * B + A * D + B * C + C * D, family = binomial,
##        data = data)
## m110  glm(formula = R ~ A * B + A * D + B * C + B * D + C * D, family = binomial,
##        data = data)
## m84  glm(formula = R ~ A * D + B * C + C * D, family = binomial, data = data)
## m107  glm(formula = R ~ A * B + A * C + A * D + B * C + C * D, family = binomial,
```

```
## data = data)
## m105 glm(formula = R ~ A * D + B * C + B * D + C * D, family = binomial,
## data = data)
```

```
anova(m98, m110, test="Chisq") # adding B*D does not improve fit
anova(m84, m98, test="Chisq") # adding A*B improves fit
anova(m63, m84, test="Chisq") # Adding C*D improves fit
anova(m51, m63, test="Chisq") # Adding B improves fit
```

```
## Analysis of Deviance Table
```

```
##
## Model 1: R ~ A * B + A * D + B * C + C * D
## Model 2: R ~ A * B + A * D + B * C + B * D + C * D
## Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1 3461 3127.3
## 2 3460 3126.8 1 0.46421 0.4957
```

```
## Analysis of Deviance Table
```

```
##
## Model 1: R ~ A * D + B * C + C * D
## Model 2: R ~ A * B + A * D + B * C + C * D
## Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1 3462 3131.2
## 2 3461 3127.3 1 3.8506 0.04973 *
```

```
## ---
```

```
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Analysis of Deviance Table
```

```
##
## Model 1: R ~ A * D + C * D + B
## Model 2: R ~ A * D + B * C + C * D
## Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1 3463 3137.3
## 2 3462 3131.2 1 6.1886 0.01286 *
```

```
## ---
```

```
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Analysis of Deviance Table
```

```
##
## Model 1: R ~ A * D + C * D
## Model 2: R ~ A * D + C * D + B
## Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1 3464 3497.3
## 2 3463 3137.3 1 359.93 < 2.2e-16 ***
```

```
## ---
```

```
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The best fit model is m98. That is confirmed by its minimum AIC value, maximum Akaike weight, and the addition of A*B (sex_binom*pophost_binom) leading to a significant improvement in model fit as detected by the ANOVA test.

Best Fit

```
M1 = glm(wing_morph_binom ~ sex_binom * pophost_binom + sex_binom * months_since_start +
pophost_binom * month_of_year + month_of_year * months_since_start,
data=raw_data, family="binomial")
```

```
summary(M1)
```

```
##
## Call:
## glm(formula = wing_morph_binom ~ sex_binom * pophost_binom +
##       sex_binom * months_since_start + pophost_binom * month_of_year +
##       month_of_year * months_since_start, family = "binomial",
##       data = raw_data)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.3803   0.3597   0.4321   0.8450   1.2552
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.7516501  0.1841942   4.081 4.49e-05 ***
## sex_binom     -0.2597900  0.0902673  -2.878 0.004002 **
## pophost_binom  1.1256358  0.1142931   9.849 < 2e-16 ***
## months_since_start  0.0107239  0.0029582   3.625 0.000289 ***
## month_of_year   0.0995560  0.0255307   3.899 9.64e-05 ***
## sex_binom:pophost_binom  0.0973323  0.0495811   1.963 0.049635 *
## sex_binom:months_since_start  0.0037212  0.0015337   2.426 0.015254 *
## pophost_binom:month_of_year -0.0379395  0.0150617  -2.519 0.011771 *
## months_since_start:month_of_year -0.0014557  0.0004553  -3.198 0.001386 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 3562.3  on 3469  degrees of freedom
## Residual deviance: 3127.3  on 3461  degrees of freedom
##   (62 observations deleted due to missingness)
## AIC: 3145.3
##
## Number of Fisher Scoring iterations: 5
```

All single effects and their interactions are significant in the best fit model for predicting wing morph. It may be because of the size of the dataset that the model is more sensitive at detecting weak interactions as significant.

Long-Wing Morph Variance

We then tested how sex, host plant, month, and/or year effected long-wing morph frequency variance.

First, the long-wing morph mean frequency was computed using `aggregate()` to group the long-wing morph recordings in `raw_data` according to sex, host plant, month, and year. The subsequent subset data created was `wmorph_table` (n=40) Then, summary statistics were applied to the data subset and variance (`sd`) was modeled.

```
wmorph_table = aggregate(wing_morph_binom ~
                          sex_binom*pophost_binom*month_of_year*months_since_start,
                          data=raw_data, FUN=mean)
```

```
SE = function(x){sd(x)/sqrt(length(x))}
```



```

wmorph_table$sd = aggregate(wing_morph_binom ~
                             sex_binom*pophost_binom*month_of_year*months_since_start,
                             data=raw_data,FUN=sd)$wing_morph_binom
wmorph_table$se = aggregate(wing_morph_binom ~
                             sex_binom*pophost_binom*month_of_year*months_since_start,
                             data=raw_data,FUN=SE)$wing_morph_binom
wmorph_table$n = aggregate(wing_morph_binom ~
                             sex_binom*pophost_binom*month_of_year*months_since_start,
                             data=raw_data,FUN=length)$wing_morph_binom

```

```

data = wmorph_table
data = data.frame(R=data$sd,
                  A=data$sex_binom,
                  B=data$pophost_binom,
                  C=(data$month_of_year),
                  D=data$months_since_start)

```

```

model_script = paste0(source_path,"generic models-gaussian glm 4-FF.R")
model_comparisonsAIC(model_script)

```

```

##          [,1]      [,2]      [,3]      [,4]
## AICs    -92.39855 -90.95292 -90.75898 -90.41465
## models  2         5         8         9
## probs   0.183788  0.0892081 0.08096352 0.06815837
##
## m2      glm(formula = R ~ B, family = gaussian, data = data)
## m5      glm(formula = R ~ A + B, family = gaussian, data = data)
## m8      glm(formula = R ~ B + C, family = gaussian, data = data)
## m9      glm(formula = R ~ B + D, family = gaussian, data = data)

```

```

anova(m2, m5, test="Chisq") # Adding A does not improve fit
anova(m2, m8, test="Chisq") # Adding C does not improve fit
anova(m2, m9, test="Chisq") # Adding D does not improve fit
anova(m0, m2, test="Chisq") # Adding B improves fit

```

```

## Analysis of Deviance Table
##
## Model 1: R ~ B
## Model 2: R ~ A + B
##   Resid. Df Resid. Dev Df   Deviance Pr(>Chi)
## 1         38      0.20009
## 2         37      0.19734  1 0.0027541   0.4724
## Analysis of Deviance Table
##
## Model 1: R ~ B
## Model 2: R ~ B + C
##   Resid. Df Resid. Dev Df   Deviance Pr(>Chi)
## 1         38      0.20009
## 2         37      0.19830  1 0.0017949   0.5628
## Analysis of Deviance Table
##
## Model 1: R ~ B
## Model 2: R ~ B + D

```

```
##   Resid. Df Resid. Dev Df    Deviance Pr(>Chi)
## 1         38      0.20009
## 2         37      0.20001  1 8.0534e-05   0.9029
## Analysis of Deviance Table
##
## Model 1: R ~ 1
## Model 2: R ~ B
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1         39      0.62439
## 2         38      0.20010  1  0.42429 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The best fit model is m2. That is confirmed by its minimum AIC value, maximum Akaike weight, and the addition of B (pophost_binom) to the null model leading to a significant improvement in model fit as detected by the ANOVA test.

Best Fit

```
M2 = glm(sd ~ pophost_binom, data=wmorph_table, family="gaussian")
summary(M2)

##
## Call:
## glm(formula = sd ~ pophost_binom, family = "gaussian", data = wmorph_table)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.249168 -0.041487  0.005877  0.041147  0.171269
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.35216    0.01147  30.693 < 2e-16 ***
## pophost_binom -0.10299    0.01147  -8.976 6.28e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.005265667)
##
##      Null deviance: 0.62439  on 39  degrees of freedom
## Residual deviance: 0.20010  on 38  degrees of freedom
## AIC: -92.399
##
## Number of Fisher Scoring iterations: 2
```

Host plant (*K. elegans* = 1, *C. corindum* = -1) is significant in predicting long-wing morph frequency variance. Soapberry bugs collected from *C. corindum*, balloon vine, experience more variance in long-wing morph frequency than those collected from *K. elegans*, goldenrain tree.

Wing-to-Body Ratio

We tested how sex, host plant, month, and/or year effected whether the wing-to-body ratio of long-winged soapberry bugs.

```

data = data.frame(R=data_long$wing2body_c,
                  A=data_long$sex_binom,
                  B=data_long$pophost_binom,
                  C=data_long$month_of_year_c,
                  D=data_long$months_since_start_c)

model_script = paste0(source_path,"generic models-gaussian glm 4-FF.R")
model_comparisonsAIC(model_script)

##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## AICs    -9722.301 -9721.371 -9720.852 -9720.339 -9720.331 -9719.674
## models  88       99       58       92       97       76
## probs   0.1948772 0.1224324 0.09441271 0.07306166 0.07277994 0.05239229
##
## m88  glm(formula = R ~ A * B + A * D + B * D + C, family = gaussian,
##        data = data)
## m99  glm(formula = R ~ A * B + A * D + B * D + C * D, family = gaussian,
##        data = data)
## m58  glm(formula = R ~ A * B + B * D + C, family = gaussian, data = data)
## m92  glm(formula = R ~ A * B + A * C + A * D + B * D, family = gaussian,
##        data = data)
## m97  glm(formula = R ~ A * B + A * D + B * C + B * D, family = gaussian,
##        data = data)
## m76  glm(formula = R ~ A * B + B * D + C * D, family = gaussian, data = data)

anova(m88, m99, test="Chisq") # adding C*D does not improve fit
anova(m58, m88, test="Chisq") # Adding A*D marginally improves fit
anova(m58, m76, test="Chisq") # Adding C*D does not improve fit
anova(m34, m58, test="Chisq") # Adding B*D improves fit

## Analysis of Deviance Table
##
## Model 1: R ~ A * B + A * D + B * D + C
## Model 2: R ~ A * B + A * D + B * D + C * D
##   Resid. Df Resid. Dev Df   Deviance Pr(>Chi)
## 1         1895      0.66692
## 2         1894      0.66655  1 0.00037502  0.3019
## Analysis of Deviance Table
##
## Model 1: R ~ A * B + B * D + C
## Model 2: R ~ A * B + A * D + B * D + C
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1         1896      0.66813
## 2         1895      0.66692  1  0.00121  0.06371 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Analysis of Deviance Table
##
## Model 1: R ~ A * B + B * D + C
## Model 2: R ~ A * B + B * D + C * D
##   Resid. Df Resid. Dev Df   Deviance Pr(>Chi)
## 1         1896      0.66813

```

```
## 2      1895      0.66784  1 0.0002886   0.3655
## Analysis of Deviance Table
##
## Model 1: R ~ A * B + C + D
## Model 2: R ~ A * B + B * D + C
##   Resid. Df Resid. Dev Df   Deviance Pr(>Chi)
## 1      1897      0.67063
## 2      1896      0.66813  1 0.0024994  0.00774 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The best fit model is m58. It did not have the minimum AIC value or maximum Akaike weight, but the addition of A*D (sex_binom*months_since_start_c) was not detected as a significant improvement in model fit, according to the ANOVA test.

Best Fit

```
M3 = glm(wing2body_c ~ sex_binom*pophost_binom + pophost_binom*months_since_start_c
        + month_of_year_c, data=data_long, family=gaussian)
summary(M3)
```

```
##
## Call:
## glm(formula = wing2body_c ~ sex_binom * pophost_binom + pophost_binom *
##   months_since_start_c + month_of_year_c, family = gaussian,
##   data = data_long)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.070837 -0.010794 -0.000093  0.010596  0.113993
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -4.542e-04  4.601e-04  -0.987  0.32368
## sex_binom     -1.787e-03  4.467e-04  -4.001 6.55e-05 ***
## pophost_binom  4.289e-03  4.613e-04   9.297 < 2e-16 ***
## months_since_start_c -1.727e-05  2.225e-05  -0.776  0.43763
## month_of_year_c    7.155e-04  1.379e-04   5.188 2.35e-07 ***
## sex_binom:pophost_binom  1.804e-03  4.466e-04   4.038 5.60e-05 ***
## pophost_binom:months_since_start_c  5.904e-05  2.217e-05   2.663 0.00781 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.0003523901)
##
##      Null deviance: 0.72538  on 1902  degrees of freedom
## Residual deviance: 0.66813  on 1896  degrees of freedom
## AIC: -9720.9
##
## Number of Fisher Scoring iterations: 2
```

All single effects except months_since_start and all interactions are significant in the best fit model for predicting wing-to-body ratio. It is noticeable that month and year effect sizes are relatively small.

Wing-to-Body Ratio Variance

We then tested how sex, host plant, month, or year effects the wing-to-body ratio variance of long-winged soapberry bugs.

First, the mean wing-to-body ratio was computed using `aggregate()` to group the wing-to-body ratio recordings in `data_long` according to sex, host plant, month, and year. The subsequent subset data created was `w2b_table` (n=36). Then, summary statistics were applied to the data subset and variance (`sd`) was modeled.

```
w2b_table = aggregate(wing2body ~ sex_binom*pophost_binom*month_of_year*months_since_start,
                      data=data_long, FUN=mean)
w2b_table$sd = aggregate(wing2body ~ sex_binom*pophost_binom*month_of_year*months_since_start,
                        data=data_long, FUN=sd)$wing2body
w2b_table$se = aggregate(wing2body ~ sex_binom*pophost_binom*month_of_year*months_since_start,
                        data=data_long, FUN=SE)$wing2body
```

```
data = w2b_table
data = data.frame(R=data$sd,
                  A=data$sex_binom,
                  B=data$pophost_binom,
                  C=(data$month_of_year),
                  D=data$months_since_start)

model_script = paste0(source_path,"generic models-gaussian glm 4-FF.R")
model_comparisonsAIC(model_script)
```

```
##           [,1]      [,2]      [,3]
## AICs    -280.1872 -279.8713 -279.4758
## models  8          19          2
## probs   0.1198675 0.1023577 0.08398967
##
## m8      glm(formula = R ~ B + C, family = gaussian, data = data)
## m19     glm(formula = R ~ B * C, family = gaussian, data = data)
## m2      glm(formula = R ~ B, family = gaussian, data = data)
```

```
anova(m8, m19, test="Chisq") # Adding B*C does not improve fit
anova(m2, m8, test="Chisq")  # Adding C does not improve fit
anova(m0, m2, test="Chisq")  # Adding B improves fit
```

```
## Analysis of Deviance Table
##
## Model 1: R ~ B + C
## Model 2: R ~ B * C
##   Resid. Df Resid. Dev Df   Deviance Pr(>Chi)
## 1          33 0.00070342
## 2          32 0.00067127  1 3.215e-05   0.2157
## Analysis of Deviance Table
##
## Model 1: R ~ B
## Model 2: R ~ B + C
##   Resid. Df Resid. Dev Df   Deviance Pr(>Chi)
## 1          34 0.00075844
```

```
## 2          33 0.00070342  1 5.5025e-05   0.1081
## Analysis of Deviance Table
##
## Model 1: R ~ 1
## Model 2: R ~ B
##   Resid. Df Resid. Dev Df    Deviance Pr(>Chi)
## 1          35 0.00087733
## 2          34 0.00075844  1 0.00011888  0.02097 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The best fit model is m2. It did not have the minimum AIC value or maximum Akaike weight, but the addition of B (pophost_binom) to the null model lead to a significant improvement in model fit detected by the ANOVA test.

Best Fit

```
M4 = glm(sd ~ pophost_binom, data=w2b_table, family=gaussian)
summary(M4)
```

```
##
## Call:
## glm(formula = sd ~ pophost_binom, family = gaussian, data = w2b_table)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.0059374 -0.0033018 -0.0006274  0.0022332  0.0147212
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0165999  0.0007872  21.088  <2e-16 ***
## pophost_binom 0.0018172  0.0007872   2.309   0.0272 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2.230719e-05)
##
##      Null deviance: 0.00087733  on 35  degrees of freedom
## Residual deviance: 0.00075844  on 34  degrees of freedom
## AIC: -279.48
##
## Number of Fisher Scoring iterations: 2
```

Host plant (*K. elegans* = 1, *C. corindum* = -1) is significant in predicting wing-to-body ratio variance. Soapberry bugs collected from *K. elegans*, goldenrain tree, experience more variance in wing-to-body ratio than those collected from *C. corindum*, balloon vine.

LOESS & Linear Regression Plots

Wing Morph Frequency

Group significant elements

```
# function to calculate 95% confidence interval (CI).
CI_95 = function(x){qnorm(0.975)*sd(x)/sqrt(length(x))}
CI_95_binom_upper = function(y) {binom.confint(x=sum(y, na.rm=TRUE), n=length(y[!is.na(y)]),
                                                conf.level=0.95, methods='exact')$upper}
CI_95_binom_lower = function(y) {binom.confint(x=sum(y, na.rm=TRUE), n=length(y[!is.na(y)]),
                                                conf.level=0.95, methods='exact')$lower}

# aggregate the full data
w_morph_summary = aggregate(wing_morph_binom ~ sex*pophost*month_of_year*months_since_start,
                             data=raw_data, FUN=mean)

# compute standard error (SE), upper and lower CI, & sample size (n)
w_morph_summary$se = aggregate(wing_morph_binom ~
                               sex*pophost*month_of_year*months_since_start,
                               data=raw_data, FUN=SE)$wing_morph_binom
w_morph_summary$upper = aggregate(wing_morph_binom ~
                                  sex*pophost*month_of_year*months_since_start,
                                  data=raw_data, FUN=CI_95_binom_upper)$wing_morph_binom
w_morph_summary$lower = aggregate(wing_morph_binom ~
                                  sex*pophost*month_of_year*months_since_start,
                                  data=raw_data, FUN=CI_95_binom_lower)$wing_morph_binom
w_morph_summary$n = aggregate(wing_morph_binom ~
                              sex*pophost*month_of_year*months_since_start,
                              data=raw_data, FUN=length)$wing_morph_binom

# jitter points slightly
jitter = runif(n=nrow(w_morph_summary), min=-0.1, max=0.1)
w_morph_summary$dates = w_morph_summary$month_of_year + jitter

dd = w_morph_summary
```

Check for LOESS Residuals

```
plot_lowess_residuals = function(lfit, x, y) {
  lfun = approxfun(lfit)
  fitted = lfun(x)
  resid = y-fitted
  plot(fitted,resid)
  abline(h=0,col=8)
}

l1 = lowess(dd$dates, dd$wing_morph_binom, f=0.4) # f = alpha, the smoother span

par(mfrow=c(1,2))
plot(dd$dates, dd$wing_morph_binom)
lines(l1, type = "l")
plot_lowess_residuals(l1, dd$dates, dd$wing_morph_binom)
```

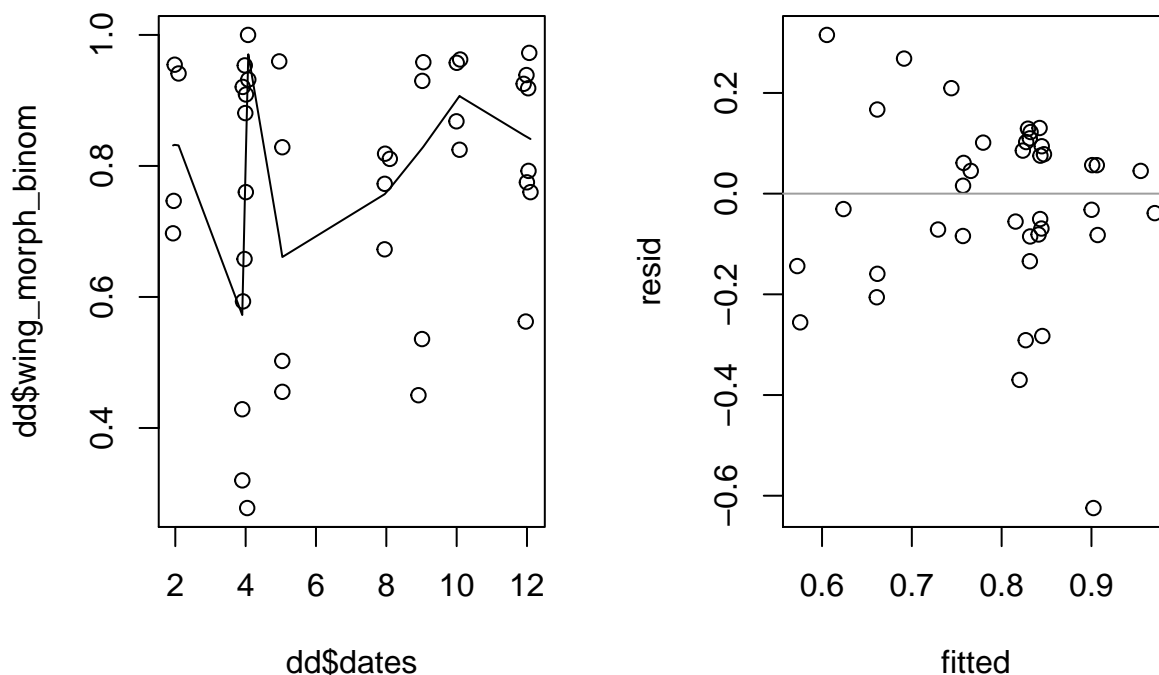


Figure: Panels A, B, C, D (long-wing morph freq with month) & E (long-wing morph freq with year)

Note: Single-variate models or simpler models substituted best fit slope calculations below in order to avoid multiple interaction terms found in the best fit model. This led to cleaner glm line plotting.

Panels A and B Regression Computations:

```
# single-variate model of month predicting wing morph
fit = glm(wing_morph_binom ~ month_of_year, family="binomial", data=raw_data)
xmonth = seq(2,12, 0.01)
wing_probs = predict(fit, list(month_of_year=xmonth), type="response")

# extract pvalue from best fit regression model
fit_pvalue = round(summary(M1)$coeff[, "Pr(>|z|)"][5], 5)
pvalue = paste0("italic(p)[glm]==", fit_pvalue)
```

Panels C and D Regression Computations:

```
# multi-variate model with month, sex, and host plant predicting wing morph
fit2 = glm(wing_morph_binom ~ sex_binom * pophost_binom +
           pophost_binom * month_of_year, family = "binomial", data = raw_data)
xmon = seq(2,12, 0.01)
set.seed(194842)
bsex = sample(c(-1,1), replace=TRUE, size=length(xmon))
bhost = sample(c(-1,1), replace=TRUE, size=length(xmon))
wprobs = predict(fit2, list(sex_binom = bsex,
                           pophost_binom = bhost,
                           month_of_year = xmon), type="response")

pred = cbind(xmon, bsex, bhost, wprobs)
pred = as.data.frame(pred)

predFK = pred[pred$bhost==1 & pred$bsex==1,]
predFC = pred[pred$bhost==1 & pred$bsex==1,]
```



```

predMK = pred[pred$bhost==1 & pred$bsex==1,]
predMC = pred[pred$bhost==1 & pred$bsex==1,]

# extract pvalue from best fit regression model
fit_pvalue = round(summary(M1)$coeff[, "Pr(>|z|)"][6], 4)
pvalue = paste0("italic(p)[glm]==", fit_pvalue)

```

Panel F Regression Computations:

```

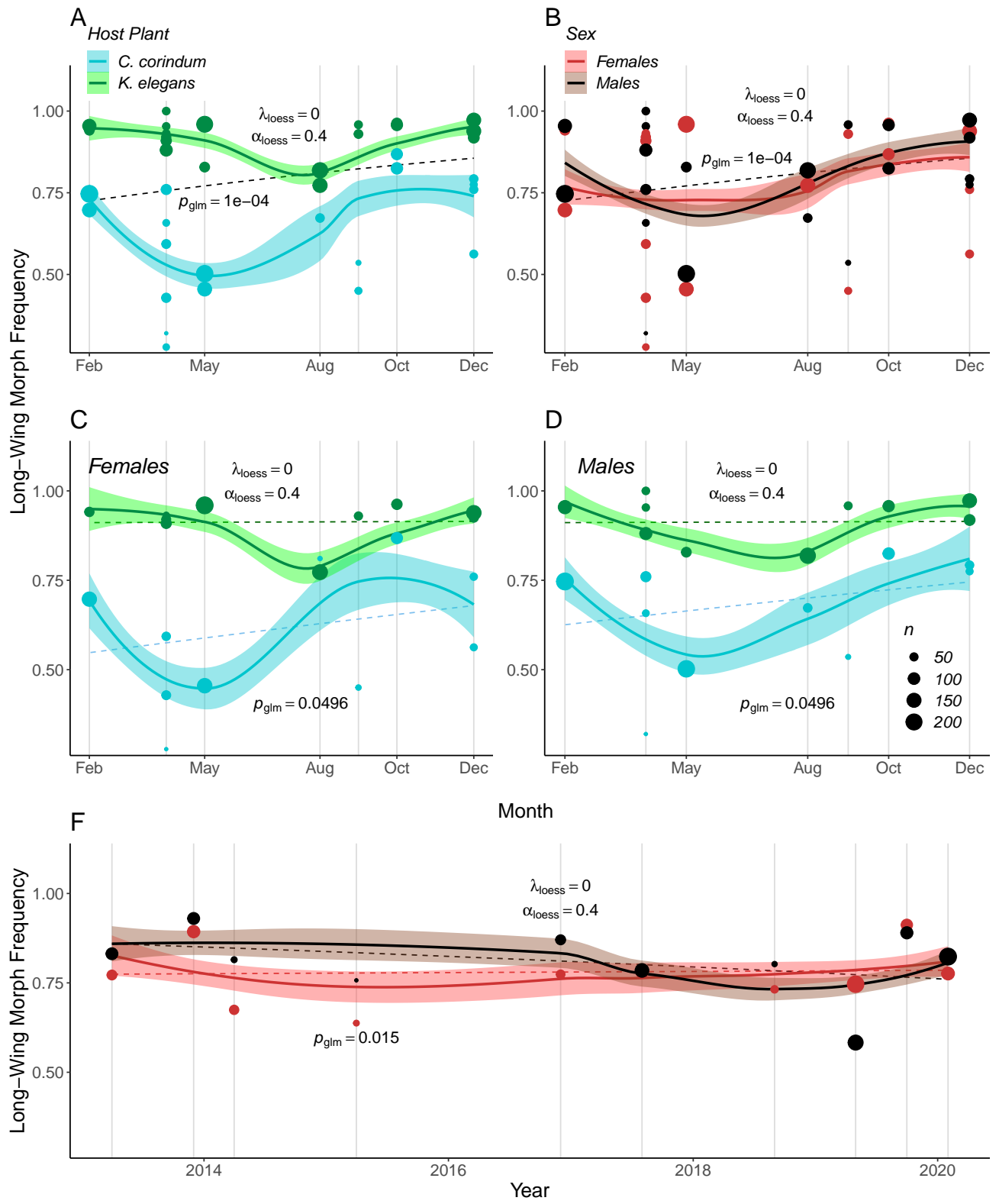
# multi-variate model with year, sex, and host plant predicting wing morph
fit3 = glm(wing_morph_binom ~ sex_binom * dates, family = "binomial", data = raw_data)
xyr = seq(sort(unique(dd$dates))[1], sort(unique(dd$dates))[10], 1)
set.seed(194842)
bsex = sample(c(-1, 1), replace=TRUE, size=length(xyr))
bhost = sample(c(-1, 1), replace=TRUE, size=length(xyr))
wprobs = predict(fit3, list(sex_binom = bsex,
                           pophost_binom = bhost,
                           dates = xyr), type="response")

pred = cbind(xyr, bsex, bhost, wprobs)
pred = as.data.frame(pred)
pred$xyr = as.Date.numeric(pred$xyr)

predF = pred[pred$bsex==1,]
predM = pred[pred$bsex==1,]

# extract pvalue from best fit regression model
fit_pvalue = round(summary(M1)$coeff[, "Pr(>|z|)"][7], 3)
pvalue = paste0("italic(p)[glm]==", fit_pvalue)

```



Wing-to-Body Ratio

Group significant elements

```
w2b_summary = aggregate(wing2body~sex*pophost*dates*month_of_year, data=data_long, FUN=mean)
w2b_summary$se = aggregate(wing2body~sex*pophost*dates, data=data_long,
                           FUN=SE)$wing2body
w2b_summary$n = aggregate(wing2body~sex*pophost*dates, data=data_long,
                          FUN=length)$wing2body

jitter = runif(n=nrow(w2b_summary), min=-0.1, max=0.1) # jitter slightly
w2b_summary$dates = w2b_summary$dates + jitter
d = w2b_summary
```

Check for LOESS Residuals

```
l1 = lowess(d$dates, d$wing2body, f=0.4)

par(mfrow=c(1,2))
plot(d$dates, d$wing2body)
lines(l1, type = "l")
plot_lowess_residuais(l1, d$dates, d$wing2body)
```

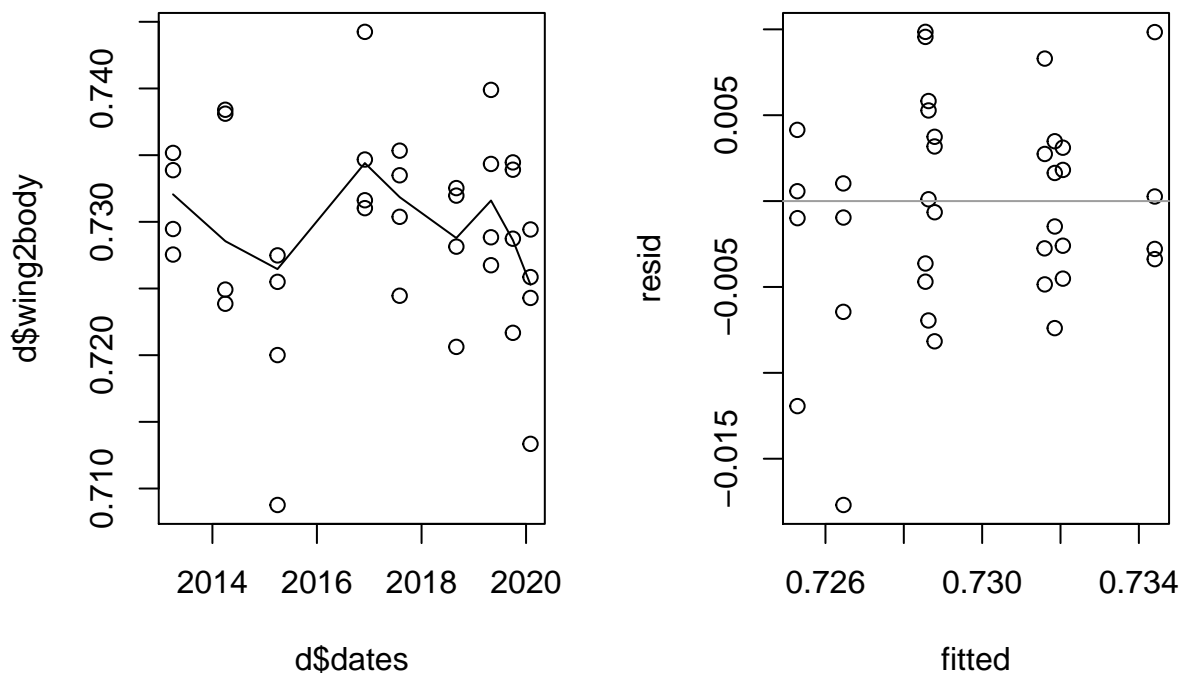


Figure: Panels A, B, (wing-to-body ratio with month) & C (wing-to-body ratio with year)

Panel C Regression Computations:

```
# multi-variate model with year, sex, and host plant predicting wing2body ratio
fit4 = glm(wing2body ~ pophost_binom * dates, data = data_long)
xyr = seq(sort(unique(dd$dates))[1], sort(unique(dd$dates))[10], 1)
set.seed(194842)
bhost = sample(c(-1,1), replace=TRUE, size=length(xyrs))
wprobs = predict(fit4, list(pophost_binom = bhost,
                           dates = xyrs), type="response")
```

```

pred = cbind(xyr, bhost, wprobs)
pred = as.data.frame(pred)
pred$xyr = as.Date.numeric(pred$xyr)

predK = pred[pred$bhost==1,]
predC = pred[pred$bhost==2,]

# extract pvalue from best fit regression model
fit_pvalue = round(summary(M3)$coeff[, "Pr(>|t|)"][7], 3)
pvalue = paste0("italic(p)[glm]==", fit_pvalue)

```

