

# Ecological factors influencing primate vision conformation: a phylogenetic regression workflow for the *mmodely* R-package (Version 0.2.2)

David M. Schruth  
dschruth@anthropoidea.org

November 5, 2022

## 1 Introduction

Traits evolve in the context of historically and ecologically complex arrangements that can present difficulty for researchers attempting to uncover causal relationships [5] or draw inference to larger populations [7]. Primates represent a manageably diverse clade of mammals who exhibit a wide range of behaviors and morphology conducive to revealing evolutionary processes. Ideas on the origins on primates range from predation detection [1] or deterrence [15] to targeting in hunting [2] or locomotion [3]. Here, using the *mmodely* package on primate data for locomotion [14] and vision [4], I demonstrate how the origins of primate cranial morphology can be elucidated via several ecological variables from numerous datasets [12]. Model averaging [MA] [8] and model selection [MS] [9] results primarily highlight arboreal locomotor targeting and trophic security [15] variables (such as stature or group size) as playing key roles in determining convergence of primate orbits. The *mmodely* package enables implementation of a combination of phylogenetic controlled regression [10] and information theoretic [11] (MA and MS) examination to simultaneously compare (weighted) predictor coefficients across the numerous sub-datasets generated during the exploration of all possible model combinations.

## 2 Licensing

The *mmodely* package is licensed under the Apache License v2.0: it is therefore free to use and redistribute, however, we, the copyright holders, wish to maintain primary control over any further development. Please be sure to cite *mmodely* if you use the package in presentations or work leading to publication.

## 3 Installation

This package largely depends upon the *caper* package, but most functions do not require any particular library. It is recommended that you have *caper*, *ape*, and the *caroline* package installed as a minimum.

```
> # wget https://cran.r-project.org/src/contrib/Archive/caroline/caroline_0.8.0.tar.gz
> # wget https://cran.r-project.org/src/contrib/Archive/caper/caper_0.5.tar.gz
> # wget https://cran.r-project.org/src/contrib/Archive/ape/ape_3.0-5.tar.gz
> # R CMD INSTALL caroline_0.8.0.tar.gz
> # R CMD INSTALL caper_0.5.tar.gz
> # R CMD INSTALL ape_3.0-5.tar.gz
```

Building the *mmodely* package from source requires that you have the proper dependency packages, *caroline*, installed from CRAN. This can typically be accomplished via the following commands from within the R command line environment:

```
install.packages(c('caroline', 'ape', 'caper'))
```

After a successful installation the *mmodely* package can be loaded in the normal way: by starting R and invoking the following `library` command:

```
> library(caper)
> library(mmodely)
```

## 4 Reading in Data

Read in the tree [13] and datasets then merge them together.

```
> data.path <- system.file("extdata","primate-example.data.csv", package="mmodely")
> data <- read.csv(data.path, row.names=1)
> data$gn_sp <- rownames(data)
> # merge data sets here if applicable
>
> tree.path <- system.file("extdata","primate-springer.2012.tre", package="mmodely")
> phyl <- ape::read.tree(tree.path)[[5]]
> #5. RAxML phylogram based on the 61199 bp concatenation of 69 nuclear and ten mitochondrial genes.
>
> phyl <- trim.phylo(phylo=phyl, gs.vect=data$gn_sp) # prune unused nodes and branches
> comp <- comp.data(phylo=phyl, df=data)
```

Typically there will be some missing data (species) in certain sources that do not occur in others. A merge of these will result in NA values for some cells. The more missing cells and merges there are, the more sub-datasets will be possible, due to case-wise deletion in the process of combinatorics underlying model iteration, averaging, and selection. The above example has little if any missing data, but the examples below introduce some artificially.

## 5 Basic Reporting

First, for illustration purposes, we perform a simple analysis of a single model using 'pgls' directly from the *caper* package, then show-off the 'pgls.report' functionality of the *mmodely* package. ANOVA, AIC, and one-line model reports can be output via this function.

```
> model <- as.formula('OC ~ mass.Kg + group.size')
> fit <- caper::pgls(formula=model, data=comp)
> summary(fit)
```

Call:

```
caper::pgls(formula = model, data = comp)
```

Residuals:

Min	1Q	Median	3Q	Max
-30.495	-5.996	17.023	34.907	54.965

Branch length transformations:

```
kappa [Fix] : 1.000
lambda [Fix] : 1.000
delta [Fix] : 1.000
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	66.037399	8.409402	7.8528	7.05e-07 ***
mass.Kg	0.153139	0.060894	2.5148	0.02298 *
group.size	-0.030092	0.107433	-0.2801	0.78299

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

Residual standard error: 34.2 on 16 degrees of freedom

Multiple R-squared: 0.3513, Adjusted R-squared: 0.2702

F-statistic: 4.332 on 3 and 16 DF, p-value: 0.02045

```
> pgls.report(comp, f=model, anova=TRUE, QC.plot=TRUE)
```

Call:

```
pgls(formula = f, data = cd, lambda = 1, kappa = k, delta = d,
      bounds = bounds)
```

Residuals:

Min	1Q	Median	3Q	Max
-30.495	-5.996	17.023	34.907	54.965

Branch length transformations:

kappa [Fix] : 1.000  
lambda [Fix] : 1.000  
delta [Fix] : 1.000

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	66.037399	8.409402	7.8528	7.05e-07 ***
mass.Kg	0.153139	0.060894	2.5148	0.02298 *
group.size	-0.030092	0.107433	-0.2801	0.78299

---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 34.2 on 16 degrees of freedom

Multiple R-squared: 0.3513, Adjusted R-squared: 0.2702

F-statistic: 4.332 on 3 and 16 DF, p-value: 0.02045

[1] "AIC = 134.2"

Analysis of Variance Table

Sequential SS for pglS: lambda = 1.00, delta = 1.00, kappa = 1.00

Response: OC

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
mass.Kg	1	10043.9	10043.9	8.5862	0.009806 **
group.size	1	91.8	91.8	0.0785	0.782991
Residuals	16	18716.2	1169.8		

---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
++mass(0.023) | group(0.783)

Call:

pgls(formula = f, data = cd, lambda = 1, kappa = k, delta = d,  
bounds = bounds)

Coefficients:

(Intercept)	mass.Kg	group.size
66.03740	0.15314	-0.03009

## 6 Multivariate Combinatoric Iteration

The *mmodely* package's chief contribution is enabling approaches that utilize multi-model iteration averaging. Using a smaller subset of variables can speed up the (slower) maximum likelihood computation step and still achieve the desired result of fixed tree transformation parameters.

```
> pv0 <- c("mass.Kg", "group.size", "arboreal", "nocturnal") #"swing.pct"
> est.mods <- get.model.combos(predictor.vars=pv0, outcome.var='OC', min.q=2)
> ps <- get.phylo.stats(phylo=phyl, data=data, trait.clmn='OC');

$lambda
[1] 0.8133958

$logL
[1] -204.9656

$P
[1] 0

$K
[1] 0.7251836

$P
[1] 0.001

> lambda <- ps$lambda$lambda ; print(lambda)
[1] 0.8133958

> PGLSi <- pgl.iter(models=est.mods, phylo=phyl, df=data, l=lambda, k='ML', d='ML')

1 OC~mass.Kg+group.size+arboreal+nocturnal
2 OC~mass.Kg+group.size+arboreal
3 OC~mass.Kg+group.size+nocturnal
4 OC~mass.Kg+arboreal+nocturnal
5 OC~group.size+arboreal+nocturnal
6 OC~mass.Kg+group.size
7 OC~mass.Kg+arboreal
8 OC~mass.Kg+nocturnal
9 OC~group.size+arboreal
10 OC~group.size+nocturnal
11 OC~arboreal+nocturnal

> pgl.iter.stats(PGLSi) # check run, especially to see how few sub-datasets exist

models: 11
dimensions of sub-datasets:
  q    n  qXn rwGsm
  3    1    3    1
      q  n
Min.  2.000000 63
1st Qu. 2.000000 63
Median  2.000000 63
Mean    2.545455 63
3rd Qu. 3.000000 63
Max.    4.000000 63
tree transformation parameter averages:
      l      k      d
0.8133958 1.4591898 1.2008103
distributions of optimization parameters:
      n      n.1      q      rwGsm      model.no      R2
Min.  :63  Min.  :63  Min.  :2.000  Min.  :117909  Min.  : 1.0  Min.  :0.04241
```

1st Qu.:63	1st Qu.:63	1st Qu.:2.000	1st Qu.:117909	1st Qu.: 3.5	1st Qu.:0.05945
Median :63	Median :63	Median :2.000	Median :117909	Median : 6.0	Median :0.20084
Mean :63	Mean :63	Mean :2.545	Mean :117909	Mean : 6.0	Mean :0.15453
3rd Qu.:63	3rd Qu.:63	3rd Qu.:3.000	3rd Qu.:117909	3rd Qu.: 8.5	3rd Qu.:0.21835
Max. :63	Max. :63	Max. :4.000	Max. :117909	Max. :11.0	Max. :0.23433
R2.adj	AIC	AICc	AICw		
Min. :0.01049	Min. :400.1	Min. :400.5	Min. :0.0003387		
1st Qu.:0.02002	1st Qu.:401.5	1st Qu.:402.1	1st Qu.:0.0006806		
Median :0.16042	Median :402.6	Median :403.7	Median :0.0654038		
Mean :0.11865	Mean :405.8	Mean :406.4	Mean :0.0909091		
3rd Qu.:0.18158	3rd Qu.:412.4	3rd Qu.:412.8	3rd Qu.:0.1460563		
Max. :0.19116	Max. :413.5	Max. :414.2	Max. :0.3197358		

## 7 Tree Transformation Averaging and Re-iteration

After running PGLS on a test-subset of predictor-variable combinations using maximum likelihood, we can average the tree transformation parameters [16] to obtain fixed values going forward. This approach can speed up computations for larger sets of modeling data and variable combinations.

```
> tt.avgs <- apply(PGLSi$params, 2, mean, na.rm=TRUE) # tree transformation averages
> print(tt.avgs)
```

```
      l      k      d
0.8133958 1.4591898 1.2008103
```

```
>
```

Next we use the full set of variables and our tree transform averages. For demonstration, we sprinkle in some missing values to our dataset so as to artificially boost the number of sub-datasets. The subsequent fixed tree parameter iteration run should now generate more diverse output upon which the *mmodely* can demonstrate its unique model averaging and model selection functionality.

```
> pvs <- c("mass.Kg", "group.size", "infant.carry", "arboreal", "DPL.km", "nocturnal")
> all.mods <- get.model.combos(predictor.vars=pvs, outcome.var='OC', min.q=2)
> data <- subset(data, !grepl(rownames(data), pattern='gorilla')) # remove an OC measurement outlier
> # randomly sprinkle in some missing values (for more interesting for model selection)
> missing.value.ct <- 1
> for(pv in pv0){ data[sample(x=1:nrow(data), size=missing.value.ct), pv] <- NA}
> PGLSi <- pglis.iter(models=all.mods, phylo=phyl, df=data, l=lambda, k=tt.avgs['k'], d=tt.avgs['d'])
```

```
1 OC~mass.Kg+group.size+infant.carry+arboreal+DPL.km+nocturnal
2 OC~mass.Kg+group.size+infant.carry+arboreal+DPL.km
3 OC~mass.Kg+group.size+infant.carry+arboreal+nocturnal
4 OC~mass.Kg+group.size+infant.carry+DPL.km+nocturnal
5 OC~mass.Kg+group.size+arboreal+DPL.km+nocturnal
6 OC~mass.Kg+infant.carry+arboreal+DPL.km+nocturnal
7 OC~group.size+infant.carry+arboreal+DPL.km+nocturnal
8 OC~mass.Kg+group.size+infant.carry+arboreal
9 OC~mass.Kg+group.size+infant.carry+DPL.km
10 OC~mass.Kg+group.size+infant.carry+nocturnal
11 OC~mass.Kg+group.size+arboreal+DPL.km
12 OC~mass.Kg+group.size+arboreal+nocturnal
13 OC~mass.Kg+group.size+DPL.km+nocturnal
14 OC~mass.Kg+infant.carry+arboreal+DPL.km
15 OC~mass.Kg+infant.carry+arboreal+nocturnal
16 OC~mass.Kg+infant.carry+DPL.km+nocturnal
17 OC~mass.Kg+arboreal+DPL.km+nocturnal
18 OC~group.size+infant.carry+arboreal+DPL.km
19 OC~group.size+infant.carry+arboreal+nocturnal
20 OC~group.size+infant.carry+DPL.km+nocturnal
21 OC~group.size+arboreal+DPL.km+nocturnal
22 OC~infant.carry+arboreal+DPL.km+nocturnal
23 OC~mass.Kg+group.size+infant.carry
24 OC~mass.Kg+group.size+arboreal
25 OC~mass.Kg+group.size+DPL.km
26 OC~mass.Kg+group.size+nocturnal
27 OC~mass.Kg+infant.carry+arboreal
28 OC~mass.Kg+infant.carry+DPL.km
29 OC~mass.Kg+infant.carry+nocturnal
30 OC~mass.Kg+arboreal+DPL.km
31 OC~mass.Kg+arboreal+nocturnal
32 OC~mass.Kg+DPL.km+nocturnal
33 OC~group.size+infant.carry+arboreal
34 OC~group.size+infant.carry+DPL.km
35 OC~group.size+infant.carry+nocturnal
```

```

36 OC~group.size+arboreal+DPL.km
37 OC~group.size+arboreal+nocturnal
38 OC~group.size+DPL.km+nocturnal
39 OC~infant.carry+arboreal+DPL.km
40 OC~infant.carry+arboreal+nocturnal
41 OC~infant.carry+DPL.km+nocturnal
42 OC~arboreal+DPL.km+nocturnal
43 OC~mass.Kg+group.size
44 OC~mass.Kg+infant.carry
45 OC~mass.Kg+arboreal
46 OC~mass.Kg+DPL.km
47 OC~mass.Kg+nocturnal
48 OC~group.size+infant.carry
49 OC~group.size+arboreal
50 OC~group.size+DPL.km
51 OC~group.size+nocturnal
52 OC~infant.carry+arboreal
53 OC~infant.carry+DPL.km
54 OC~infant.carry+nocturnal
55 OC~arboreal+DPL.km
56 OC~arboreal+nocturnal
57 OC~DPL.km+nocturnal

```

## 8 Fixed iteration run statistics

We should briefly inspect how this fixed iteration run performed and how many sub-datasets we need to investigate. By default *mmodely* uses 'rwGsm.' This abbreviation stands for 'raw *Genus species* sums.' It represents a sum of the (concatenated) raw character values of all species constituting the underlying dataset (which has all rows with any missing data removed) for a particular combination of model predictor variables. While this default is preferred, the number of species 'n' or number of model variables 'q' can also be used.

```
> ppls.iter.stats(PGLSi)
```

```
models: 57
```

```
dimensions of sub-datasets:
```

q	n	qXn	rwGsm
5	7	18	15

	q	n
Min.	2.000000	54.00000
1st Qu.	2.000000	55.00000
Median	3.000000	56.00000
Mean	3.263158	57.38596
3rd Qu.	4.000000	60.00000
Max.	6.000000	61.00000

```
tree transformation parameter averages:
```

l	k	d
0.8133958	1.4591898	1.2008103

```
distributions of optimization parameters:
```

n	n.1	q	rwGsm	model.no
Min. :54.00	Min. :54.00	Min. :2.000	Min. :100120	Min. : 1
1st Qu.:55.00	1st Qu.:55.00	1st Qu.:2.000	1st Qu.:102151	1st Qu.:15
Median :56.00	Median :56.00	Median :3.000	Median :104160	Median :29
Mean :57.39	Mean :57.39	Mean :3.263	Mean :106975	Mean :29
3rd Qu.:60.00	3rd Qu.:60.00	3rd Qu.:4.000	3rd Qu.:112268	3rd Qu.:43
Max. :61.00	Max. :61.00	Max. :6.000	Max. :114559	Max. :57

R2	R2.adj	AIC	AICc	AICw
Min. :0.003849	Min. :-0.03050	Min. :347.0	Min. :348.2	Min. :0.000e+00
1st Qu.:0.068114	1st Qu.: 0.02437	1st Qu.:355.3	1st Qu.:356.5	1st Qu.:0.000e+00
Median :0.108098	Median : 0.04704	Median :363.6	Median :364.3	Median :9.492e-05
Mean :0.098689	Mean : 0.04301	Mean :369.3	Mean :370.2	Mean :1.754e-02

```
3rd Qu.:0.122207 3rd Qu.: 0.06691 3rd Qu.:384.3 3rd Qu.:384.8 3rd Qu.:4.766e-03
Max. :0.176245 Max. : 0.11025 Max. :394.2 Max. :394.9 Max. :3.040e-01
```

## 9 Model Averaging

Now we can estimate the predictor variable parameters by averaging over all possible fixed PGLS runs, using the AICc differences (from the lowest AICc) as weights. By default this AICw weighted average is performed per sub-dataset using 'rwGsm' as mentioned in the preceding section.

```
> w.means.pds <- average.fit.models(vars=pvs, fits=PGLSi$fits, optims=PGLSi$optim, by='rwGsm')
> #
> apply(w.means.pds, 2, mean, na.rm=T) #average of weighted means over all sub-datasets
```

```
    mass.Kg  group.size infant.carry  arboreal  DPL.km  nocturnal
0.1579875  0.1015038  6.0052188  0.7851088 -0.1542093 -2.5152675
```

```
> w.means.pds # weighted means per sub-dataset
```

```
    mass.Kg  group.size infant.carry  arboreal  DPL.km  nocturnal
100120 0.15624  0.09358  5.59395  2.48506 -0.18932 -2.22657
101869 0.17327  NaN  6.16105  1.45829  0.47025 -2.39168
102129 0.14489  0.08318  6.69876  NaN -0.42297  NaN
102151  NaN  0.10669  4.60359  1.44184 -0.32948 -2.53635
103878 0.16852  NaN  7.31946  NaN  0.26896  NaN
103900  NaN  NaN  5.31089  0.14127  0.47195 -2.76987
104160  NaN  0.10146  5.63057  NaN -0.39529  NaN
105909  NaN  NaN  6.72348  NaN  0.46017  NaN
110519 0.14220  0.09646  NaN  1.07702 -0.65379 -2.37406
112268 0.16442  NaN  NaN  0.13083 -0.05529 -2.58888
112528 0.14282  0.09562  NaN  NaN -0.76905  NaN
112550  NaN  0.11078  NaN  0.33436 -0.57195 -2.49943
114277 0.17154  NaN  NaN  NaN -0.10783  NaN
114299  NaN  NaN  NaN -0.78780  0.13517 -2.73530
114559  NaN  0.12426  NaN  NaN -0.62467  NaN
```



## 10 Model Selection

We can select the best model by sorting each subset (e.g. by AICc) or by using visualization methods.

```
> select.best.models(PGLSi, using='AICc')
```

	n	n.1	q	qXn	rwGsm	model.no	R2	R2.adj	AIC	AICc	AICw
100120	54	54	4	4X54	100120	8	0.16233580	0.09395505	346.9518	348.2018	3.040450e-01
102129	55	55	3	3X55	102129	23	0.15968271	0.11025228	352.8184	353.6184	2.026458e-02
101869	55	55	3	3X55	101869	29	0.11874418	0.06690560	353.4706	354.2706	1.462538e-02
102151	55	55	3	3X55	102151	35	0.11341035	0.06125801	353.5510	354.3510	1.404920e-02
103878	56	56	2	2X56	103878	44	0.12193307	0.08879847	359.0199	359.4814	1.080433e-03
104160	56	56	2	2X56	104160	48	0.11054034	0.07697583	359.4910	359.9525	8.536716e-04
103900	56	56	2	2X56	103900	54	0.05899456	0.02348492	360.6637	361.1252	4.749530e-04
105909	57	57	2	2X57	105909	53	0.05921751	0.02437371	368.4775	368.9304	9.589089e-06
110519	59	59	3	3X59	110519	26	0.11935284	0.07131754	376.4176	377.1583	1.567101e-07
112550	60	60	2	2X60	112550	51	0.08326281	0.05109659	382.3204	382.7490	9.574358e-09
112268	60	60	2	2X60	112268	47	0.07558052	0.04314475	383.2639	383.6924	5.973593e-09
112528	60	60	2	2X60	112528	43	0.10326789	0.07180361	383.2902	383.7188	5.895510e-09
114559	61	61	2	2X61	114559	50	0.07279483	0.04082224	390.8490	391.2700	1.351415e-10
114299	61	61	2	2X61	114299	56	0.02300252	-0.01068704	392.1753	392.5963	6.962799e-11
114277	61	61	2	2X61	114277	46	0.05812524	0.02564680	392.2457	392.6667	6.721946e-11

Plotting the coefficients of determination versus the AIC values allows selection of high-performing models for inspection and reporting.

```
> plot.pglis.iters(PGLSi)
```

```
> sdevs.objs <- get.pglis.coefs(PGLSi$fits, est='t value')
```

```
> coefs.objs <- get.pglis.coefs(PGLSi$fits, est='Estimate')
```

```
> report.vect <- sapply(1:length(PGLSi$fits), function(i) fit.1ln.rprt(PGLSi$fits[[i]], rtrn.line=FALSE, mn=i))
```

```
1 +mass(0.092) +group(0.123) +infan(0.225) arbore(0.355) | DPL(0.974) noctur(0.38) R2adj: 0.071 AICc: 352.48
2 +mass(0.083) +group(0.125) +infan(0.179) arbore(0.341) DPL(0.931) | R2adj: 0.075 AICc: 350.73
3 +mass(0.087) +group(0.096) +infan(0.177) +arbore(0.283) | noctur(0.373) R2adj: 0.09 AICc: 349.84
4 +mass(0.128) +group(0.139) +infan(0.301) | DPL(0.596) noctur(0.365) R2adj: 0.074 AICc: 350.83
5 +group(0.076) +mass(0.12) arbore(0.763) | DPL(0.535) noctur(0.324) R2adj: 0.049 AICc: 381.23
6 +mass(0.065) +infan(0.133) arbore(0.44) DPL(0.583) | noctur(0.385) R2adj: 0.041 AICc: 358.52
7 +group(0.076) +infan(0.268) arbore(0.615) | DPL(0.846) -noctu(0.312) R2adj: 0.033 AICc: 358.75
8 +mass(0.08) +group(0.089) +infan(0.151) +arbore(0.298) | R2adj: 0.094 AICc: 348.2
9 +mass(0.115) +infan(0.134) +group(0.148) | DPL(0.686) R2adj: 0.095 AICc: 355.86
10 +mass(0.133) +group(0.16) +infan(0.194) | noctur(0.397) R2adj: 0.087 AICc: 348.61
11 +group(0.075) +mass(0.111) arbore(0.771) | DPL(0.59) R2adj: 0.049 AICc: 379.84
12 +group(0.094) +mass(0.12) arbore(0.562) | noctur(0.346) R2adj: 0.06 AICc: 379.18
13 +group(0.078) +mass(0.123) | DPL(0.426) -noctu(0.322) R2adj: 0.065 AICc: 378.85
14 +mass(0.059) +infan(0.102) arbore(0.425) DPL(0.499) | R2adj: 0.046 AICc: 356.85
15 +mass(0.063) +infan(0.157) arbore(0.569) | noctur(0.338) R2adj: 0.054 AICc: 356.33
16 +mass(0.084) +infan(0.172) DPL(0.874) | noctur(0.371) R2adj: 0.049 AICc: 356.67
17 +mass(0.084) arbore(0.953) | DPL(0.943) -noctu(0.313) R2adj: 0.009 AICc: 388.36
18 +group(0.076) +infan(0.236) arbore(0.612) | DPL(0.933) R2adj: 0.032 AICc: 357.39
19 +group(0.065) +infan(0.203) arbore(0.5) | -noctu(0.315) R2adj: 0.051 AICc: 356.27
20 +group(0.081) +infan(0.301) | DPL(0.627) -noctu(0.307) R2adj: 0.047 AICc: 356.51
21 ++grou(0.048) arbore(0.999) | DPL(0.552) -noctu(0.304) R2adj: 0.024 AICc: 386.97
22 +infan(0.154) DPL(0.647) arbore(0.789) | -noctu(0.313) R2adj: -0.011 AICc: 365.63
23 +infan(0.08) +mass(0.118) +group(0.151) | R2adj: 0.11 AICc: 353.62
24 +group(0.086) +mass(0.111) arbore(0.591) | R2adj: 0.062 AICc: 377.77
25 +group(0.07) +mass(0.108) | DPL(0.403) R2adj: 0.067 AICc: 385.26
26 +group(0.112) +mass(0.139) | noctur(0.356) R2adj: 0.071 AICc: 377.16
27 +mass(0.057) +infan(0.132) arbore(0.604) | R2adj: 0.056 AICc: 354.93
28 +infan(0.067) +mass(0.076) DPL(0.773) | R2adj: 0.073 AICc: 361.71
```

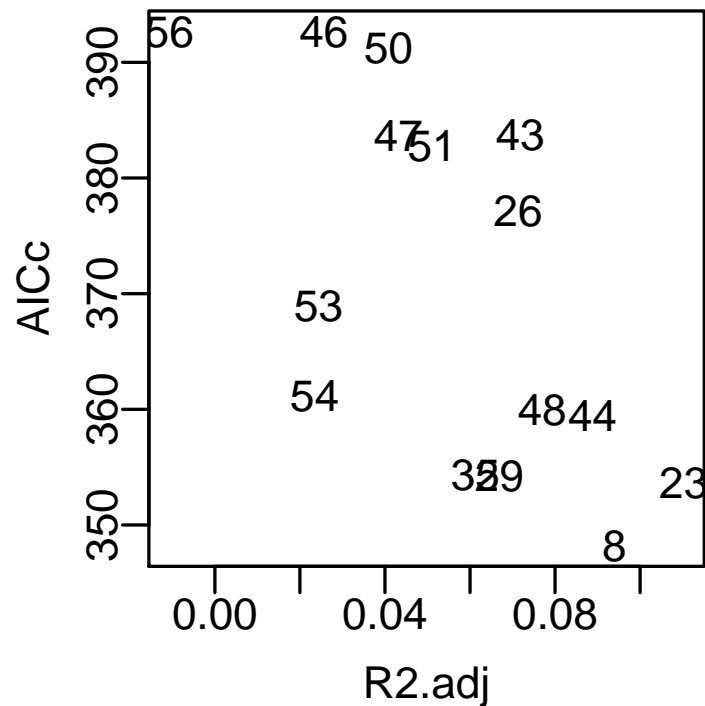
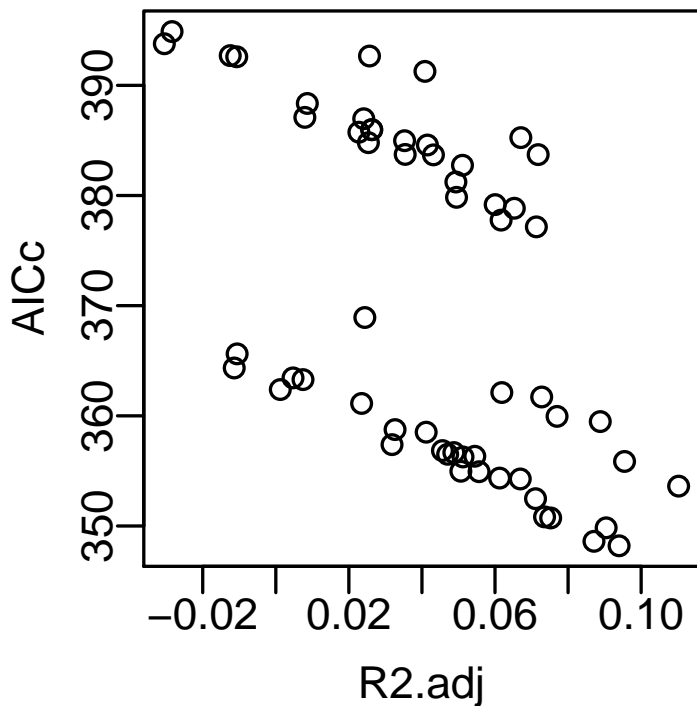
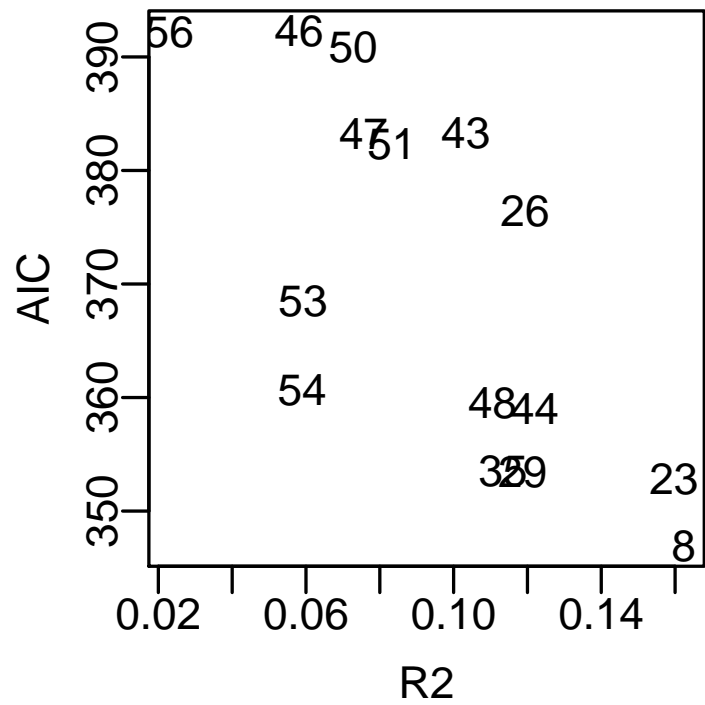
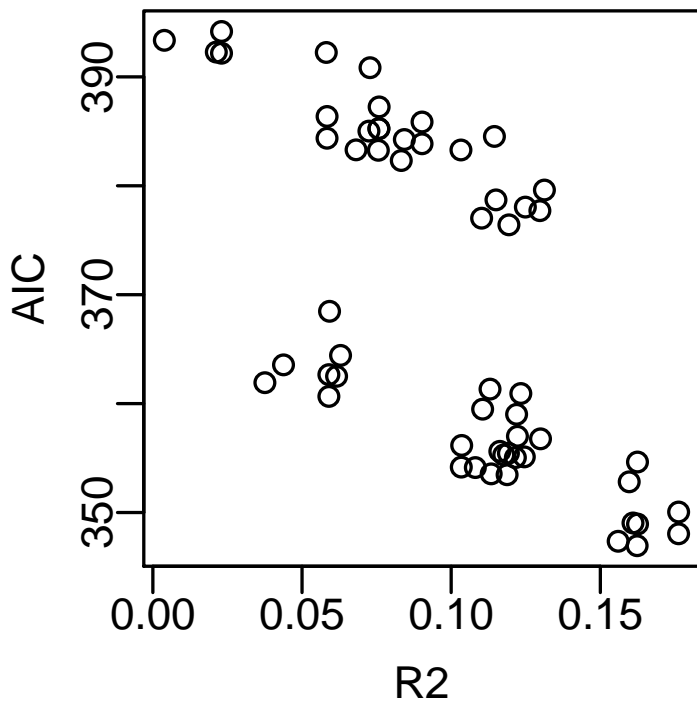


Figure 1: All possible model combinations appear as individual points above. As there is a generally negative association between AIC and the coefficient of determination, the points tend to follow a negative sloping streak to the lower right. The "best" models typically appear in the lower right of each streak. Therefore, minimizing AIC tends to also maximize the coefficient of determination, but not necessarily. This four panel plot looks at correct and adjusted versions of each model assessment measure.

```

29 +mass(0.074) +infan(0.165) | noctur(0.349) R2adj: 0.067 AICc: 354.27
30 +mass(0.077) arbore(0.963) DPL(0.983) | R2adj: 0.008 AICc: 387.1
31 +mass(0.081) arbore(0.919) | -noctu(0.309) R2adj: 0.026 AICc: 385.98
32 +mass(0.077) | DPL(0.912) -noctu(0.309) R2adj: 0.026 AICc: 385.98
33 +group(0.059) +infan(0.19) arbore(0.533) | R2adj: 0.051 AICc: 354.97
34 +group(0.084) +infan(0.156) | DPL(0.704) R2adj: 0.062 AICc: 362.12
35 +group(0.082) +infan(0.212) | noctur(0.328) R2adj: 0.061 AICc: 354.35
36 ++grou(0.047) | arbore(0.987) DPL(0.612) R2adj: 0.023 AICc: 385.75
37 +group(0.057) arbore(0.802) | noctur(0.324) R2adj: 0.035 AICc: 384.98
38 ++grou(0.044) | DPL(0.515) -noctu(0.3) R2adj: 0.041 AICc: 384.59
39 +infan(0.132) DPL(0.563) arbore(0.786) | R2adj: -0.011 AICc: 364.35
40 +infan(0.172) arbore(0.985) | -noctu(0.281) R2adj: 0.005 AICc: 363.45
41 +infan(0.156) DPL(0.706) | -noctu(0.308) R2adj: 0.007 AICc: 363.29
42 DPL(0.975) | arbore(0.75) -noctu(0.295) R2adj: -0.028 AICc: 394.89
43 +group(0.104) +mass(0.124) | R2adj: 0.072 AICc: 383.72
44 +mass(0.062) +infan(0.066) | R2adj: 0.089 AICc: 359.48
45 +mass(0.073) arbore(0.967) | R2adj: 0.025 AICc: 384.8
46 +mass(0.065) | DPL(0.9) R2adj: 0.026 AICc: 392.67
47 +mass(0.073) | -noctu(0.307) R2adj: 0.043 AICc: 383.69
48 +group(0.075) +infan(0.104) | R2adj: 0.077 AICc: 359.95
49 +group(0.052) arbore(0.843) | R2adj: 0.035 AICc: 383.73
50 ++grou(0.038) | DPL(0.497) R2adj: 0.041 AICc: 391.27
51 +group(0.052) | noctur(0.326) R2adj: 0.051 AICc: 382.75
52 +infan(0.159) | arbore(0.953) R2adj: 0.001 AICc: 362.39
53 +infan(0.071) DPL(0.619) | R2adj: 0.024 AICc: 368.93
54 +infan(0.168) | -noctu(0.276) R2adj: 0.023 AICc: 361.13
55 DPL(0.895) | arbore(0.736) R2adj: -0.031 AICc: 393.78
56 | arbore(0.705) -noctu(0.287) R2adj: -0.011 AICc: 392.6
57 DPL(0.839) | -noctu(0.288) R2adj: -0.012 AICc: 392.7

```

```

> par(mar=c(5,5,3,3))
> plot.pgls.R2AIC(PGLSi$optim)

```

## 11 Coefficient Plotting

Finally, the resulting model fits from the PGLS runs can be plotted out horizontally as distributions so the influence of each ecological predictor variable can be compared.

```

> par.old <- par(mar=c(5,8,1,4),mfrow=c(2,1))
> distro.dots.modsel(sdevs.objs, R2x=7, xlab='t value')
> distro.dots.modsel(coefs.objs, R2x=7, xlab='Estimate')

```

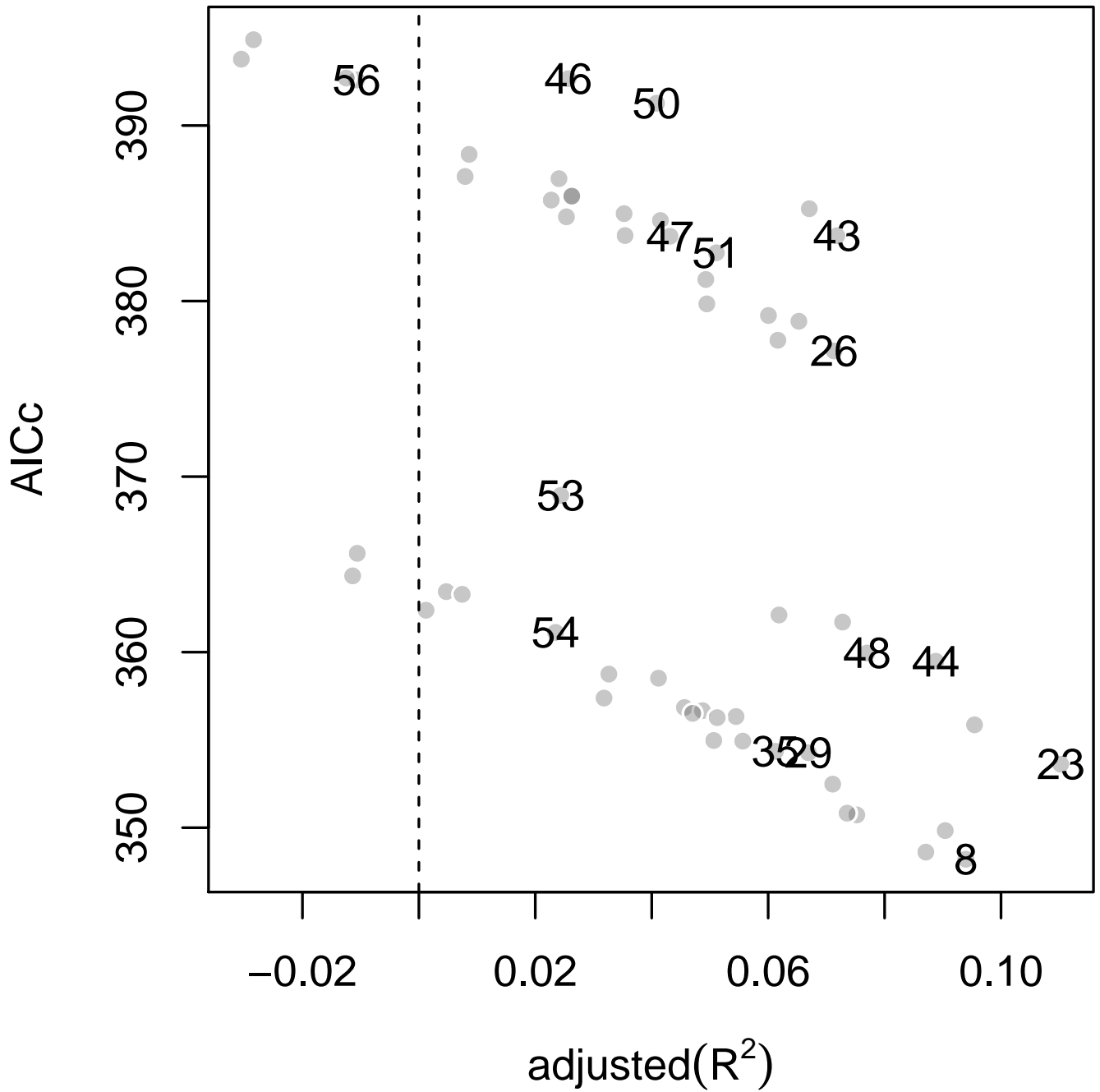


Figure 2: This is a one panel version of the previous model selection plot. The numbered points in the lower right corner of each streak of possible models represent the best model within a sub-dataset. Since these AICc values should not strictly be compared, it is recommended that all "best" models selected from each sub-dataset should be inspected or reported somehow, such as in the form of the distro dots plot below.

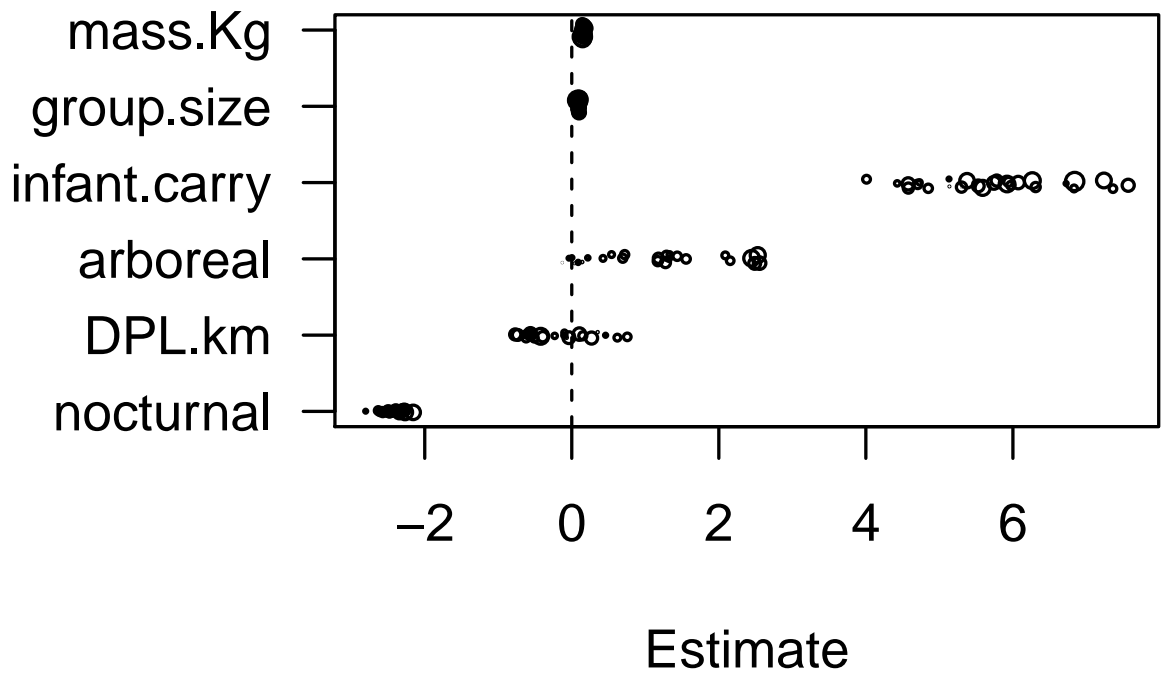
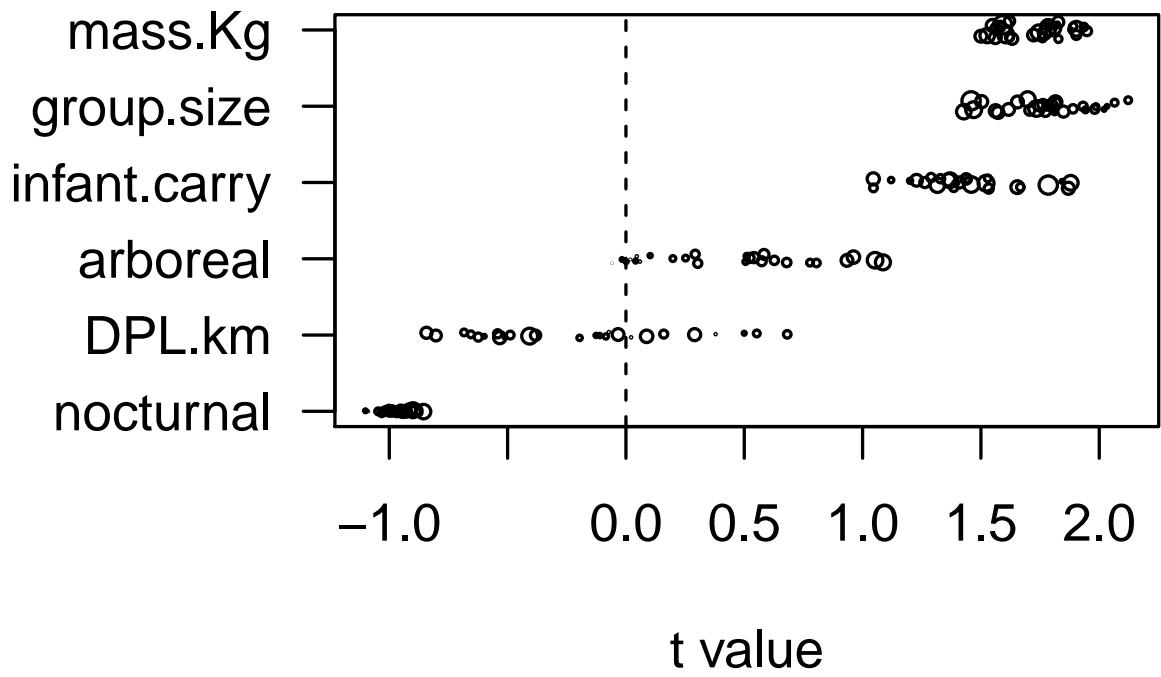


Figure 3: These horizontal parameter distribution dots plots demonstrate how the (t-values of) coefficients from all models can be simultaneously plot in order to verify consistency of estimates across the various (often missing-data driven) sub-datasets. Note that arboreal locomotor targeting and staturally protective (e.g. infant fur-cling carrying) factors play key roles driving orbital convergence.

## References

- [1] Isbell, L. (2006) ‘Snakes as agents of evolutionary change in primate brains’ *Journal of Human Evolution*, 51(1) pp. 1–35.
- [2] Cartmill, M. (1972) ‘Arboreal Adaptations and the Origin of the Order Primates’ In *The Function and Evolutionary Biology of Primates*, pp. 97–122, Aldine-Atherton.
- [3] Szalay, Frederick S., and Marian Dagosto. ‘Evolution of Hallucial Grasping in the Primates’ *Journal of Human Evolution*, 17(1–2) pp. 1–33. [https://doi.org/10.1016/0047-2484\(88\)90047-4](https://doi.org/10.1016/0047-2484(88)90047-4).
- [4] Wheeler, B.C., B.J. Bradely, and J.M. Kamilar. (2011) ‘Predictors of Orbital Convergence in Primates: A Test of the Snake Detection Hypothesis of Primate Evolution’ *Journal of Human Evolution*, 61, pp. 233–42.
- [5] Pagel, M. (1999) ‘Inferring the historical patterns of biological evolution’ *Nature*, 401, pp. 877–884. Available at: <https://doi.org/10.1038/44766>.
- [6] Grueber, C.E., Nakagawa, .R.J., and Jamieson, I.G. (2011) ‘Multimodel inference in ecology and evolution: challenges and solutions’ *Journal of Evolutionary Biology*, 24, pp. 699–711. Available at: <https://doi.org/10.1111/j.1420-9101.2010.02210.x>
- [7] Stephens, P.A., Buskirk, S.W., and del Rio, C.M. (2007) ‘Inference in ecology and evolution’ *Trends in Ecology and Evolution*, 22(4). Available at: <https://doi.org/10.1016/j.tree.2006.12.003>.
- [8] Symonds, M.R.E., and Moussalli, A. (2011) ‘A brief guide to model selection, multimodel inference and model averaging in behavioral ecology using Akaike’s information criterion’ *Behavioral Ecology and Sociobiology*, 65, pp. 13–21. Available at: <https://doi.org/10.1007/s00265-010-1037-6>.
- [9] Johnson, J.B., Omland, K.S. (2004) ‘Model selection in ecology and evolution’ *Trends in Ecology and Evolution*, 19(2). Available at: <https://doi.org/10.1016/j.tree.2003.10.013>.
- [10] Revell, L.J. (2014) phytools: An R package for phylogenetic comparative biology (and other things). [CRAN]. Available at: <http://cran.r-project.org/package=phytools>.
- [11] Garamszegi L.Z. (2011) ‘Information-theoretic approaches to statistical analysis in behavioral ecology: an introduction’ *Behavioral Ecology and Sociobiology*, 65, pp. 1–11. Available at: <https://doi.org/10.1007/s00265-010-1028-7>.
- [12] Rowe, N. and Meyers, M. (2017) *All the World’s Primates*. Charlestown, RI: Pogonias Press.
- [13] Springer, M.S., et.al. (2012) ‘Re-evaluating primate monogamy’ *PLoS ONE* 7(11) p. e49521 <http://doi.org/10.1371/journal.pone.0049521>.
- [14] Schruth, D.M. (2019) ‘Primate Locomotor Activity’ *The Center for Open Science*. Available at: <https://osf.io/cd68q/>.
- [15] Schruth, D.M. (2021a) Arboreal locomotion and trophic security at the dawn of Euprimate vision. *EcoEvoRxiv*. Available at: <http://doi.org/10.32942/osf.io/d6wk2>.
- [16] Schruth, D.M. (2021b) ‘A global variable-permutation based approach for estimating tree transformation parameters used in phylogenetically controlled multivariate regression’ *Protocols.io*. Available at: <http://doi.org/10.17504/protocols.io.bzdhp236>.
- [17] Ross, C. (2001) ‘Park or Ride? Evolution of Infant Carrying in Primates’ *International Journal of Primatology*, 22(5), pp. 749–771. Available at: <https://doi.org/10.1023/A:1012065332758>.