Multiple regression and correlation

Use the full model as a start

$$Y = f(X_1, X_2)$$

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \beta_4 X_1^2 + \beta_5 X_2^2$$

```
attach(trees)
fit = lm(Volume ~ Girth * Height + I(Girth^2) + I(Height^2),
         data=trees)
fit = step(fit)
summary(fit)
```

Model selection

```
library(MuMIn)

head(Cement) # Example from Burnham and Anderson (2002), page 100:
Cement = rbind(Cement, (Cement + rnorm(dim(Cement)[1]*dim(Cement)[2], 0, 1))) # enlarge the dataset
Cement = rbind(Cement, (Cement + rnorm(dim(Cement)[1]*dim(Cement)[2], 0, 1))) # enlarge the dataset

# Full model
fit <- lm(y ~ (X1 + X2 + X3 + X4)^2 + I(X1^2) + I(X2^2) + I(X3^2) + I(X4^2), data = Cement, na.action = na.fail)
fit.all <- dredge(fit) # model comparison
fit.all[1:6, ] # the top 6 models
```

	(Int)	X1	X1^2	X2	X2^2	X3	X3^2	X4	X4^2	X1:X2	X1:X3	X2:X3	df	logLik	AICc	delta	weight
2454	133.1	-1.438		-0.7578		-2.391			-0.00782	0.05582		0.04619	8	-101.331	222	0	0.459
2518	129.6	-1.439		-0.731		-2.406		0.06591	-0.00819	0.05653		0.04724	9	-101.256	224.8	2.79	0.114
2966	132.8	-1.44		-0.7539		-2.381			-0.0078	0.05565	0.002958	0.04597	9	-101.309	224.9	2.89	0.108
2462	129.7	-1.392		-0.6734	-0.00055	-2.35			-0.00747	0.05522		0.04573	9	-101.312	224.9	2.9	0.108
2486	132.5	-1.416		-0.7502		-2.347	-0.00092		-0.00779	0.05548		0.04586	9	-101.328	224.9	2.93	0.106
2456	133.1	-1.438	1.15E-05	-0.7578		-2.391			-0.00782	0.05582		0.04619	9	-101.331	224.9	2.94	0.106

```
#or as a 95% confidence set:
avgmod.95p <- model.avg(fit.all, cumsum(weight) <= .95); confint(avgmod.95p)

# The same result, but re-fitting the models via 'get.models'
confset.95p <- get.models(fit.all, cumsum(weight) <= .95); model.avg(confset.95p)

# Force re-fitting the component models
model.avg(fit.all, cumsum(weight) <= .95, fit = TRUE)
# Models are also fitted if additional arguments are given
model.avg(fit.all, cumsum(weight) <= .95, rank = "AIC")

# using BIC (Schwarz's Bayesian criterion) to rank the models
BIC <- function(x) AIC(x, k = log(length(residuals(x))))
model.avg(confset.95p, rank = BIC)
```

Model average

```
#models with delta.aicc < 4
summary(model.avg(fit.all, subset = delta < 4))
```

Component models:

	df	logLik	AICc	delta	weight
1/3/5/8/9/11	8	-101.33	222.01	0.00	0.46
1/3/5/7/8/9/11	9	-101.26	224.80	2.79	0.11
1/3/5/8/9/10/11	9	-101.31	224.90	2.89	0.11
1/3/4/5/8/9/11	9	-101.31	224.91	2.90	0.11
1/3/5/6/8/9/11	9	-101.33	224.94	2.93	0.11
1/2/3/5/8/9/11	9	-101.33	224.95	2.94	0.11

Term codes:

X1	I(X1^2)	X2	I(X2^2)	X3	I(X3^2)	X4	I(X4^2)	X1:X2	X1:X3	X2:X3
1	2	3	4	5	6	7	8	9	10	11

Model-averaged coefficients:
(full average)

	Estimate	Std. Error	Adjusted SE	z value	Pr(>|z|)
(Intercept)	1.32E+02	1.42E+01	1.46E+01	9.074	< 2e-16
X1	-1.43E+00	5.25E-01	5.40E-01	2.65	0.00805
X2	-7.45E-01	2.59E-01	2.66E-01	2.796	0.00517
X3	-2.38E+00	4.33E-01	4.46E-01	5.349	1.00E-07
I(X4^2)	-7.82E-03	1.42E-03	1.46E-03	5.37	1.00E-07
X1:X2	5.58E-02	1.02E-02	1.05E-02	5.316	1.00E-07
X2:X3	4.62E-02	7.66E-03	7.88E-03	5.867	< 2e-16
X4	7.51E-03	6.56E-02	6.72E-02	0.112	0.91108
X1:X3	3.20E-04	5.12E-03	5.26E-03	0.061	0.95153
I(X2^2)	-5.88E-05	1.00E-03	1.03E-03	0.057	0.95437
I(X3^2)	-9.78E-05	3.84E-03	3.95E-03	0.025	0.98024
I(X1^2)	1.22E-06	2.78E-03	2.86E-03	0	0.99966

(conditional average)

	Estimate	Std. Error	Adjusted SE	z value	Pr(>|z|)
(Intercept)	1.32E+02	1.42E+01	1.46E+01	9.074	< 2e-16
X1	-1.43E+00	5.25E-01	5.40E-01	2.65	0.00805
X2	-7.45E-01	2.59E-01	2.66E-01	2.796	0.00517
X3	-2.38E+00	4.33E-01	4.46E-01	5.349	1.00E-07
I(X4^2)	-7.82E-03	1.42E-03	1.46E-03	5.37	1.00E-07
X1:X2	5.58E-02	1.02E-02	1.05E-02	5.316	1.00E-07
X2:X3	4.62E-02	7.66E-03	7.88E-03	5.867	< 2e-16
X4	6.59E-02	1.84E-01	1.89E-01	0.348	0.72771
X1:X3	2.96E-03	1.53E-02	1.58E-02	0.188	0.85098
I(X2^2)	-5.46E-04	3.01E-03	3.09E-03	0.177	0.85971
I(X3^2)	-9.22E-04	1.18E-02	1.21E-02	0.076	0.93921
I(X1^2)	1.15E-05	8.56E-03	8.81E-03	0.001	0.99896

Effect of a variable in the context of multiple other variables

```
# (Azen and Traxel, 2009)
library(dominanceanalysis)
fit = glm(am ~ (mpg + hp)^2 +
        I(mpg^2) + I(hp^2),
        data=mtcars,
family=binomial)
dapres <- dominanceAnalysis(fit)
DOM = getFits(dapres, "r2.m")[[1]];
DOM
```

	mpg	hp	I(mpg^2)	I(hp^2)	mpg:hp
1	0.314	0.046	0.320	0.011	0.005
mpg	NA	0.242	0.007	0.224	0.249
hp	0.509	NA	0.482	0.214	0.343
I(mpg^2)	0.000	0.209	NA	0.190	0.238
I(hp^2)	0.526	0.249	0.499	NA	0.094
mpg:hp	0.557	0.385	0.553	0.100	NA
mpg+hp	NA	NA	0.017	0.000	0.009
mpg+I(mpg^2)	NA	0.252	NA	0.250	0.242
mpg+I(hp^2)	NA	0.018	0.033	NA	0.031
mpg+mpg:hp	NA	0.002	0.000	0.007	NA
hp+I(mpg^2)	0.044	NA	NA	0.002	0.029
hp+I(hp^2)	0.295	NA	0.270	NA	0.336
hp+mpg:hp	0.175	NA	0.168	0.206	NA
I(mpg^2)+I(hp^2)	0.060	0.021	NA	NA	0.048
I(mpg^2)+mpg:hp	0.005	0.000	NA	0.001	NA
I(hp^2)+mpg:hp	0.464	0.490	0.453	NA	NA
mpg+hp+I(mpg^2)	NA	NA	NA	0.004	0.000
mpg+hp+I(hp^2)	NA	NA	0.021	NA	0.042
mpg+hp+mpg:hp	NA	NA	0.009	0.033	NA
mpg+I(mpg^2)+I(hp^2)	NA	0.006	NA	NA	0.010
mpg+I(mpg^2)+mpg:hp	NA	0.011	NA	0.018	NA
mpg+I(hp^2)+mpg:hp	NA	0.029	0.011	NA	NA
hp+I(mpg^2)+I(hp^2)	0.046	NA	NA	NA	0.066
hp+I(mpg^2)+mpg:hp	0.015	NA	NA	0.039	NA
hp+I(hp^2)+mpg:hp	0.002	NA	0.001	NA	NA
I(mpg^2)+I(hp^2)+mpg:hp	0.021	0.038	NA	NA	NA
mpg+hp+I(mpg^2)+I(hp^2)	NA	NA	NA	NA	0.045
mpg+hp+I(mpg^2)+mpg:hp	NA	NA	NA	0.049	NA
mpg+hp+I(hp^2)+mpg:hp	NA	NA	0.024	NA	NA
mpg+I(mpg^2)+I(hp^2)+mpg:hp	NA	0.042	NA	NA	NA
hp+I(mpg^2)+I(hp^2)+mpg:hp	0.025	NA	NA	NA	NA
mpg+hp+I(mpg^2)+I(hp^2)+mpg:hp	NA	NA	NA	NA	NA

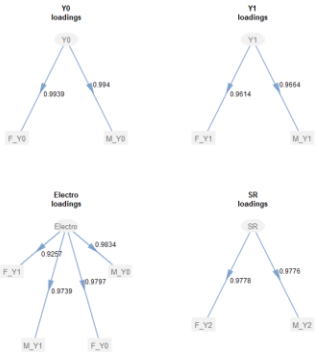
Path analysis

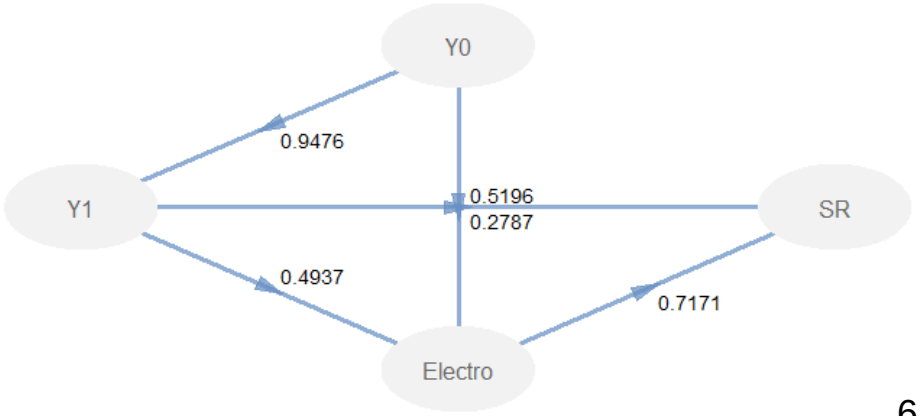
```
library(plspm)

D = read.csv('d:/data.csv', header=T);D
# path matrix (inner model realtionships)
Y0 = c(0, 0, 0, 0)
Y1 = c(1, 0, 0, 0)
Electro = c(1, 1, 0, 0)
SR = c(0,1,1,0)
saker_path = rbind(Y0, Y1, Electro, SR)
# add optional column names
colnames(saker_path) = rownames(saker_path)
# plot the path matrix
innerplot(saker_path)
# list indicating what variables are associated with what latent variables
saker_blocks = list(c(1,2), c(3,4),c(1,2,3,4),c(5,6))
# all latent variables are measured in a reflective way
saker_modes = rep("A", 4)
# run plspm analysis
saker_pls = plspm(D, saker_path, saker_blocks, modes = saker_modes)
# what's in saker_pls?
saker_pls
# path coefficients
saker_pls$path_coefs
# inner model
saker_pls$inner_model
# summarized results
summary(saker_pls)
# plot the results (inner model)
plot(saker_pls)
# plot the loadings of the outer model
plot(saker_pls, what = "loadings", arr.width = 0.2)
# plot the weights of the outer model
plot(saker_pls, what = "weights", arr.width = 0.1)
```

M_Y0	F_Y0	M_Y1	F_Y1	M_Y2	F_Y2
100	100	80	60	70	50
60	55	40	35	30	30
30	35	15	20	10	12
62	60	50	40	38	35
40	45	30	35	25	27
70	65	60	35	50	30
50	45	45	40	32	28
55	60	42	45	31	32

M_Y0 number of males at Year 0

	relationships	direct	indirect	total
1	Y0 -> Y1	0.948	0	0.948
2	Y0 -> Electro	0.52	0.468	0.987
3	Y0 -> SR	0	0.972	0.972
4	Y1 -> Electro	0.494	0	0.494
5	Y1 -> SR	0.279	0.354	0.633
6	Electro -> SR	0.717	0	0.717





Something more about interaction

In multiple regression, the variance of the dependent variable can be explained by a number of explanatory variables, in the form of linear terms, quadratic or other high order terms, and interaction terms (Haase 2011).

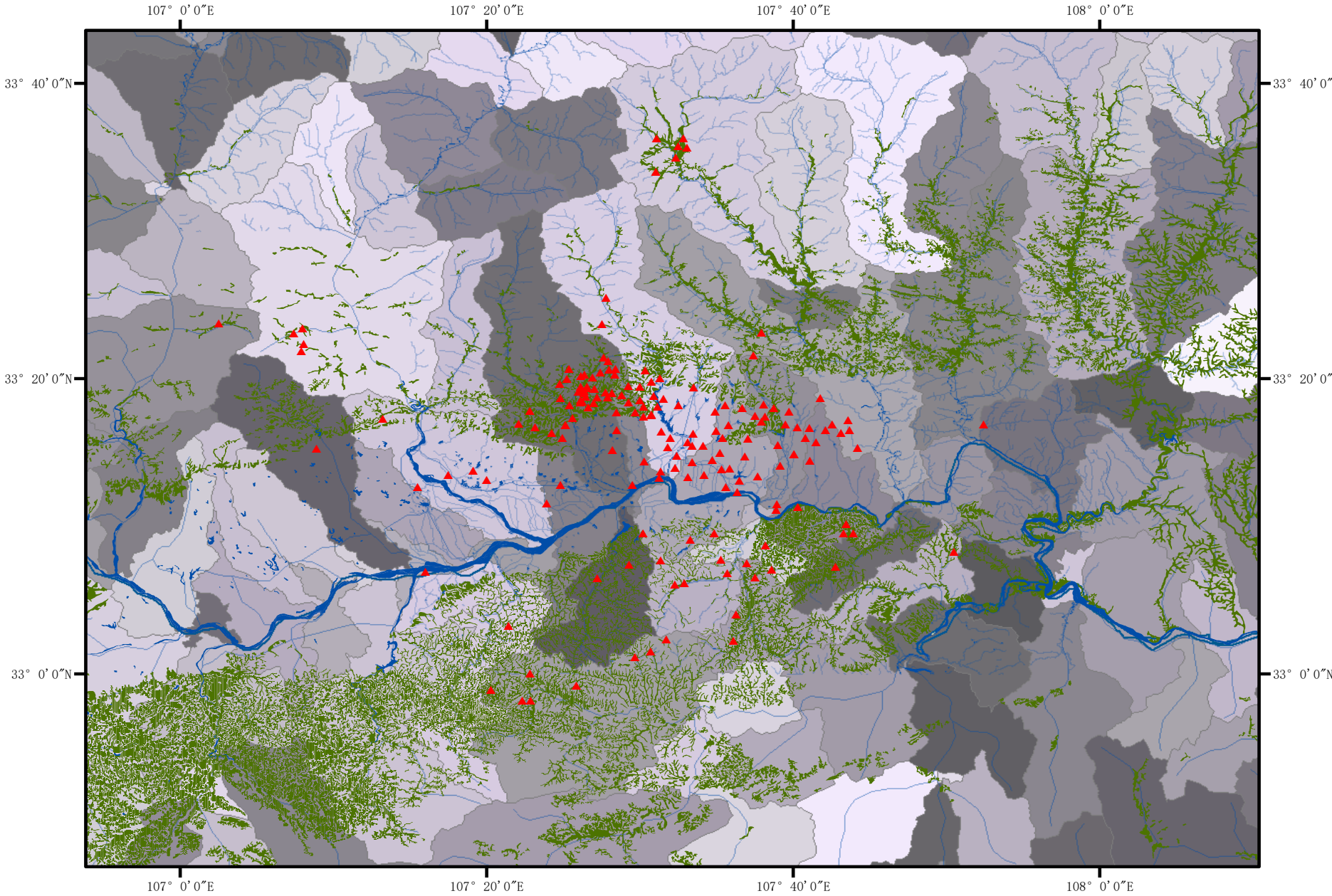
When an interaction term has a significant contribution to the model, it means the effect of one explanatory variable on the dependent variable changes depending on that of another explanatory variable. In other words, the interaction effect indicates the simultaneous influence of two variables on the dependent variable is not additive, and a nonlinear relationship is expected (Li et al. 2020).

The presence of an interaction effect is very common, such as the phenotypic differences of wildlife (e.g. body size) in the relationship between taxa and environmental temperature (Engqvist 2005), the interaction of elevation, slope, and aspect in the prediction of forest productivity or species composition (Stage and Salas 2007), the interaction of vegetation density and approaching style of hunter on bobwhite (quail) behavior (McGrath et al. 2018), and the effect on traits of genotype-by-environment interactions (Campbell and Waser 2001).

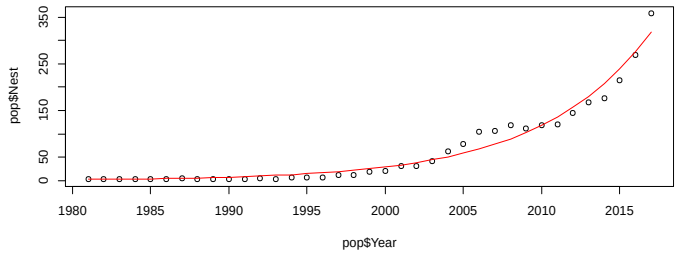
Hence, interaction terms are a fundamental part of multiple regressions (Zar 1999).

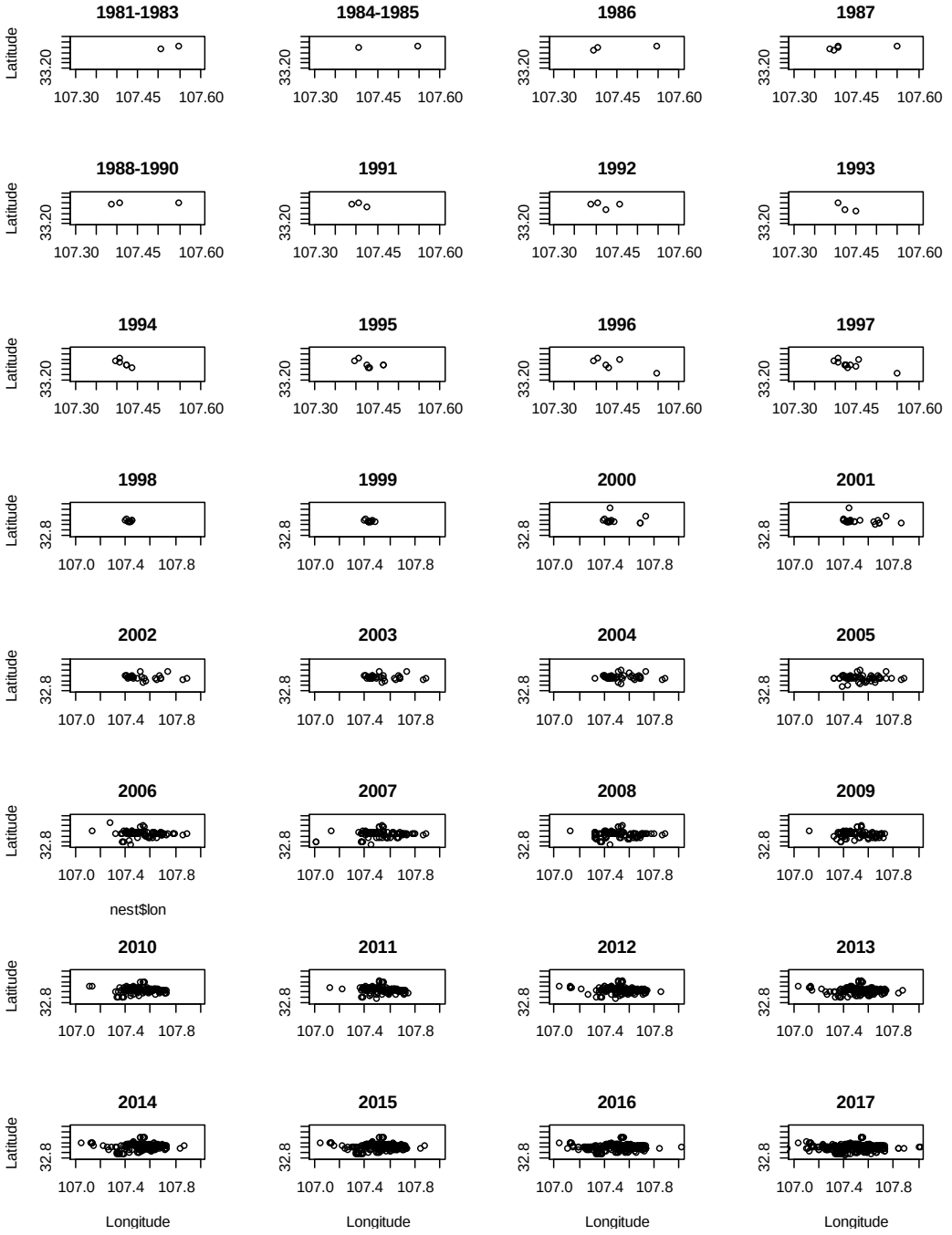
Campbell, D. R., and N. M. Waser. 2001. Genotype-by-environment interaction and the fitness of plant hybrids in the wild. Evolution 55:669.
Engqvist, L. 2005. The mistreatment of covariate interaction terms in linear model analyses of behavioural and evolutionary ecology studies. Animal Behaviour 70:967-971.
Haase, R. F. 2011. Multivariate general linear models. SAGE Publications, Inc.
Li, X., B. Li, G. Wang, X. Zhan, and M. Holyoak. 2020. Deeply digging the interaction effect in multiple linear regressions using a fractional-power interaction term. MethodsX 7:101067.
McGrath, D. J., T. M. Terhune, II, and J. A. Martin. 2018. Vegetation and predator interactions affect northern bobwhite behavior. Journal of Wildlife Management 82:1026-1038.
Stage, A. R., and C. Salas. 2007. Interactions of elevation, aspect, and slope in models of forest species composition and productivity. Forest Science 53:486-492.
Zar, J. H. 1999. Biostatistical Analysis. Fourth edition edition. Pearson.

Nest site selection of the crested ibis



Nests of the crested ibis



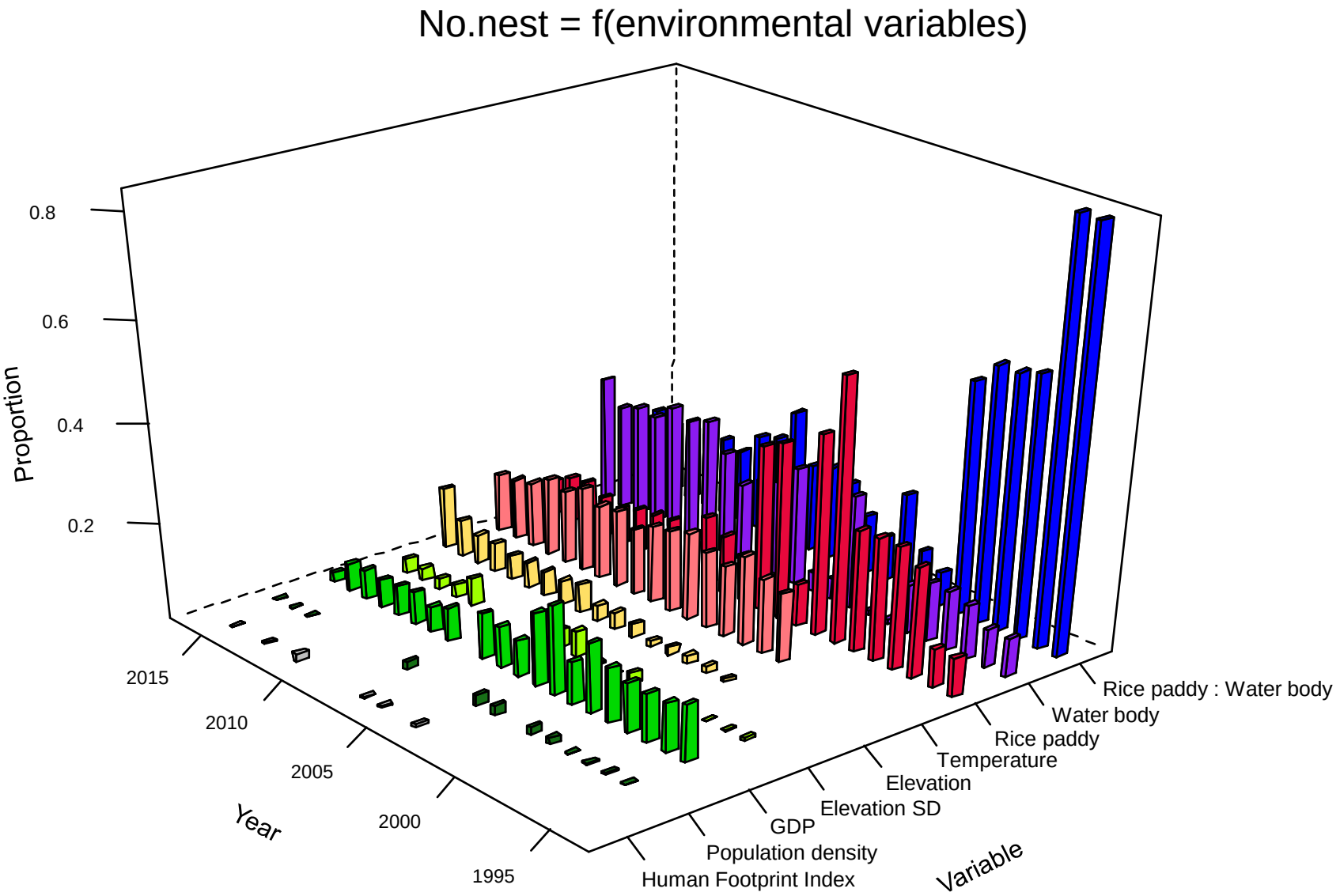


Habitat variables

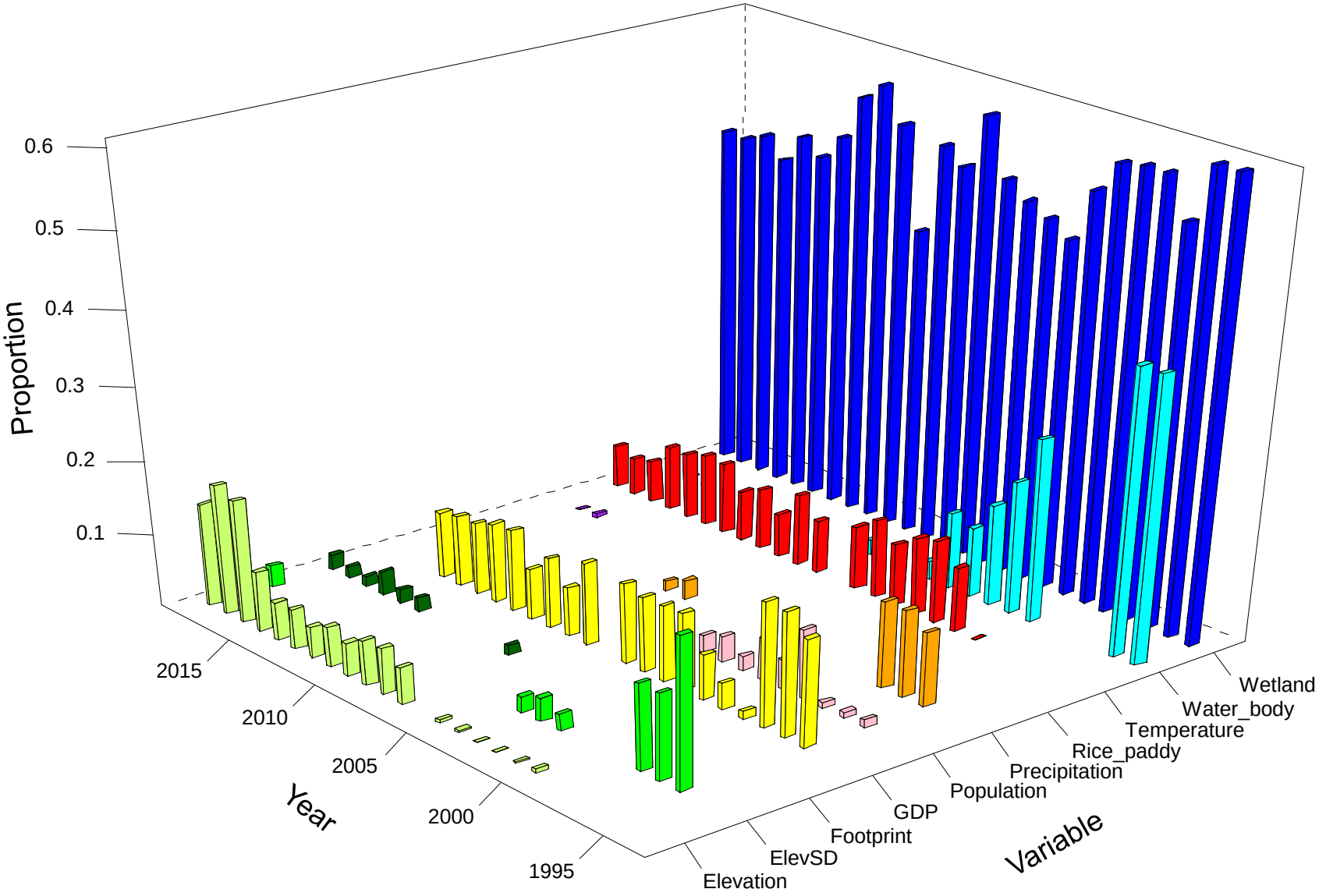
In every watersheds

- **Average elevation**
- **SD of elevation**
- **Area of rice paddy**
- **Area of water body**
- **Human footprint index**
- **Population density**
- **GDP**
- **Temperature**
- **Precipitation**
- **Area of the watershed**

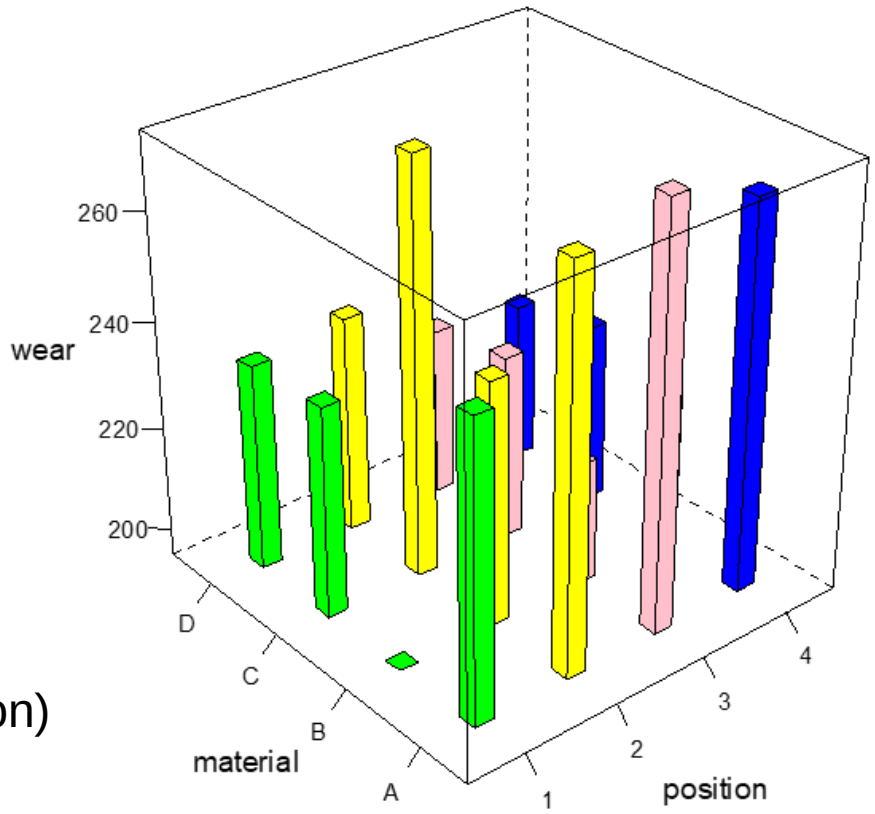
Importance of environmental variables on nest site selection



Importance of environmental variables on nest site selection in generalized linear models (negative binomial distribution)



3-D bar-plot




```
library(latticeExtra)
library(faraway); data(abrasion)
abrasion$position = as.factor(abrasion$position)

cloud(wear ~ position + material, abrasion,
      panel.3d.cloud = panel.3dbars,
      col.facet = colorRampPalette(c('green','yellow','pink','blue'))(4)[abrasion$position],
      xbase=0.2, ybase=0.2, scales=list(arrows=F, col=1),
      par.settings = list(axis.line = list(col = "transparent")))
```

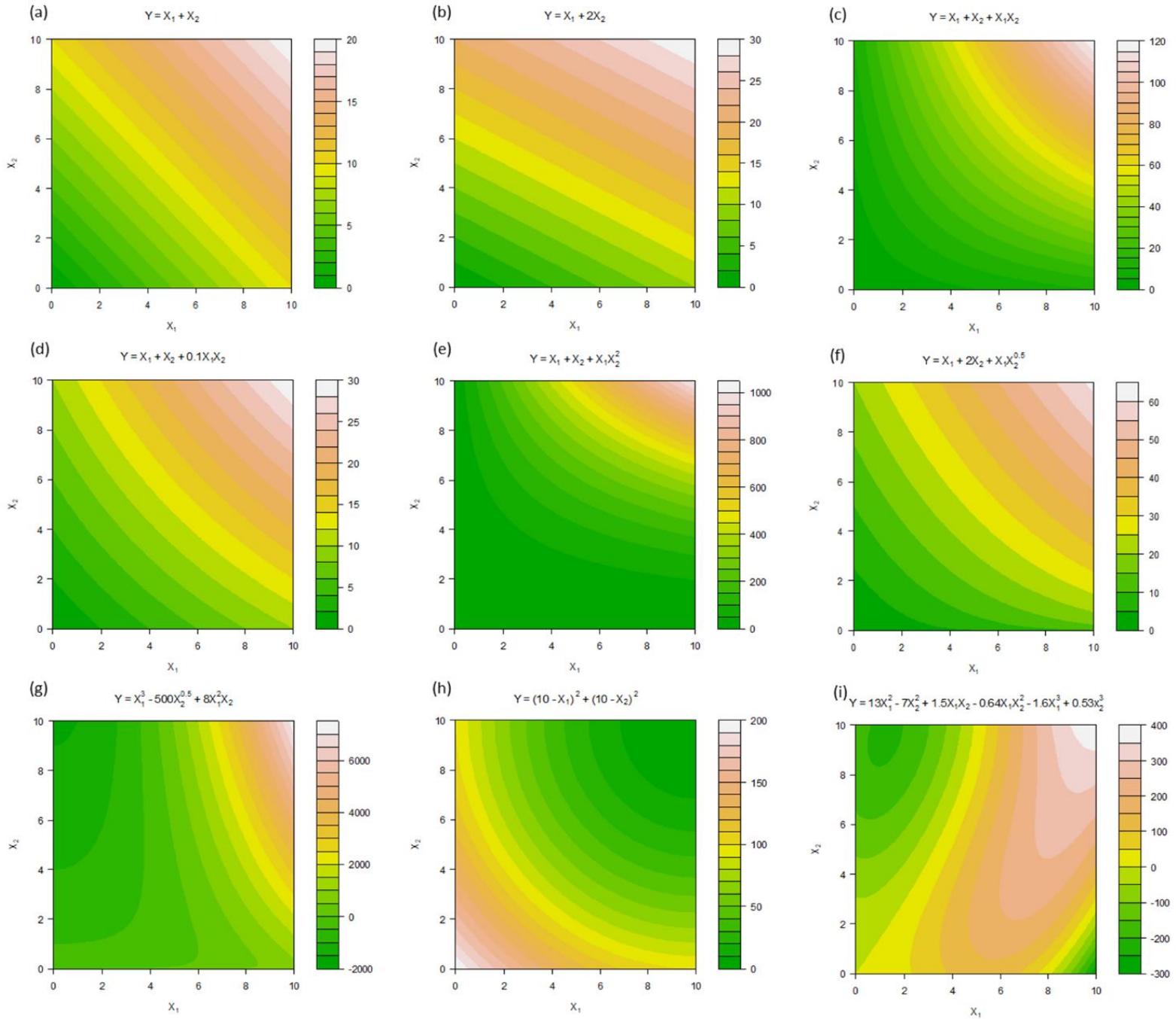
Interaction: $\beta X_1 X_2$

In most techniques developed for regressions, the interaction effect is quantified as the product of two associated explanatory variables, in the form of $\beta X_1 X_2$, where β is the coefficient, X_1 and X_2 are explanatory variables (Roisman et al. 2012, Kohler and Krzyżak 2017).

Kohler, M., and A. Krzyżak. 2017. Nonparametric regression based on hierarchical interaction models. IEEE Transactions on Information Theory 63:1620-1630.

Roisman, G. I., D. A. Newman, R. C. Fraley, J. D. Haltigan, A. M. Groh, and K. C. Haydon. 2012. Distinguishing differential susceptibility from diathesis-stress: Recommendations for evaluating interaction effects. Development and Psychopathology 24:389-409.

Contour plots for linear models



Interaction: Polynomial regression

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \beta_4 X_1^2 + \beta_5 X_2^2 + \beta_6 X_1^2 X_2 + \beta_7 X_1 X_2^2 + \beta_8 X_1^3 + \beta_9 X_2^3$$

High order terms in of the explanatory variables in polynomial regression can also be used to represent the nonlinear patterns (Cheng and Schneeweiss 1998).

However, polynomial regression can only represent a limited collection of curve shapes when lower order of polynomials (e.g. 2, 3) are used.

Additionally, higher order polynomial regression usually failed to predict Y-values outside of the observed range (Royston and Altman 1994).

Cheng, C. L., and H. Schneeweiss. 1998. Polynomial regression with errors in the variables. Journal of the Royal Statistical Society Series B-Statistical Methodology 60:189-199.

Royston, P., and D. G. Altman. 1994. Regression using fractional polynomials of continuous covariates: Parsimonious parametric modelling. Applied Statistics 43:429.

Interaction: Fractional polynomial regression ($\beta X_1^M X_2^N$)

Fractional polynomial regression (FPR) extends ordinary polynomial regressions beyond a few integer exponents (Royston and Altman 1994, Royston and Altman 1997).

Royston and Sauerbrei (2004) further invented multivariable fractional polynomials interaction (MFPI), which can handle interactions of continuous predictors in the form of fractional polynomials as $\beta X_1^M X_2^N$.

The algorithm of MFPI is available in Stata (Royston 2012), which gives a limited options (i.e. −2, − 1, − 0.5, 0, co0.5, 1, 2, 3) for the powers of a predictor. The package for *R*, mfp (R Core Team 2019), was designed to run MFPI, yet the function for treating interaction terms is still absent (Original by Gareth Ambler and modified by Axel Benner 2015).

Original by Gareth Ambler and modified by Axel Benner. 2015. mfp: Multivariable Fractional Polynomials. R package version 1.5.2. https://CRAN.R-project.org/package=mfp.

R Core Team. 2019. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

Royston, P. 2012. MFPIGEN: Stata module for modelling and displaying interactions between continuous predictors. Boston College Department of Economics, revised 31 Oct 2012.

Royston, P., and D. G. Altman. 1994. Regression using fractional polynomials of continuous covariates: Parsimonious parametric modelling. Applied Statistics 43:429.

Royston, P., and D. G. Altman. 1997. Approximating statistical functions by using fractional polynomial regression. Statistician 46:411-422.

Royston, P., and W. Sauerbrei. 2004. A new measure of prognostic separation in survival data. Statistics in Medicine 23:723-748.

MethodsX 7 (2020) 101067

Contents lists available at ScienceDirect

MethodsX

journal homepage: www.elsevier.com/locate/mex

Method Article

Deeply digging the interaction effect in multiple linear regressions using a fractional-power interaction term

Xinhai Li[a,b,*], Baidu Li[c], Guiming Wang[d], Xiangjiang Zhan[a,e,*], Marcel Holyoak[f]

[a] Key Laboratory of Animal Ecology and Conservation Biology, Institute of Zoology, Chinese Academy of Sciences, Beichen West Road, Beijing 100101, China
[b] University of Chinese Academy of Sciences, Yuquan Road, Beijing 100049, China
[c] York University, 4700 Keele Street, Toronto, Ontario M3J 1P3, Canada
[d] Department of Wildlife, Fisheries and Aquaculture, Mississippi State University, Mississippi State, MS 39762-9690, USA
[e] CAS Center for Excellence in Animal Evolution and Genetics, Chinese Academy of Sciences, Kunming 650223, China
[f] Department of Environmental Science and Policy, University of California, 1 Shields Ave., Davis, CA 95616, USA

A B S T R A C T

In multiple regression $Y \sim \beta_0 + \beta_1X_1 + \beta_2X_2 + \beta_3X_1X_2 + \varepsilon$., the interaction term is quantified as the product of X_1 and X_2. We developed fractional-power interaction regression (FPIR), using $\beta X_1^M X_2^N$ as the interaction term. The rationale of FPIR is that the slopes of Y-X_1 regression along the X_2 gradient are modeled using the nonlinear function (Slope $= \beta_1 + \beta_3 M X_1^{M-1} X_2^N$), instead of the linear function (Slope $= \beta_1 + \beta_3X_2$) that regular regressions normally implement. The ranges of *M* and *N* are from -56 to 56 with 550 candidate values, respectively. We applied FPIR using a well-studied dataset, nest sites of the crested ibis (*Nipponia nippon*).We further tested FPIR by other 4692 regression models. FPIRs have lower AIC values (-302 ± 5003.5) than regular regressions (-168.4 ± 4561.6), and the effect size of AIC values between FPIR and regular regression is 0.07 (95% CI: 0.04–0.10). We also compared FPIR with complex models such as polynomial regression, generalized additive model, and random forest. FPIR is flexible and interpretable, using a minimum number of degrees of freedom to maximize variance explained. We have provided a new R package, interactionFPIR, to estimate the values of *M* and *N*, and suggest using FPIR whenever the interaction term is likely to be significant.

- Introduced fractional-power interaction regression (FPIR) as $Y \sim \beta_0 + \beta_1X_1 + \beta_2X_2 + \beta_3X_1^M X_2^N + \varepsilon$ to replace the current regression model $Y \sim \beta_0 + \beta_1X_1 + \beta_2X_2 + \beta_3X_1X_2 + \varepsilon$;
- Clarified the rationale of FPIR, and compared it with regular regression model, polynomial regression, generalized additive model, and random forest using regression models for 4692 species;
- Provided an R package, interactionFPIR, to calculate the values of *M* and *N*, and other model parameters.

* Corresponding author.
E-mail addresses: lixh@ioz.ac.cn (X. Li), zhanxj@ioz.ac.cn (X. Zhan).

https://doi.org/10.1016/j.mex.2020.101067
2215-0161/

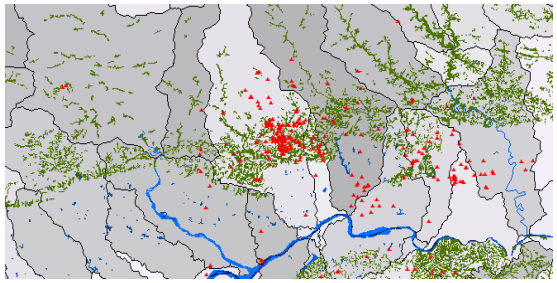
Fractional-power interaction regression (FPIR)

We developed and tested a new method, which we call fractional-power interaction regression (FPIR), to estimate the values of M and N (550 candidate values from -56 to 56) in the model

$$Y \sim \beta_0 + \beta_1X_1 + \beta_2X_2 + \beta_3X_1^M X_2^N + \varepsilon$$

```
# The R package interactionFPIR can be installed from GitHub
library(remotes)
install_github("Xinhai-Li/interaction")
library(interactionFPIR)

attach(trees)
results = FPIR1twoway(trees$Volume, trees$Girth, trees$Height)
results2 = FPIR1twowaytune(trees$Volume, trees$Girth,
                                    trees$Height, 1.35, 0.3)
#1.175  0.2
```

Li, X., B. Li, G. Wang, X. Zhan, and M. Holyoak. 2020. Deeply digging the interaction effect in multiple linear regressions using a fractional-power interaction term. MethodsX 7:101067.



Y: number of nests (red points) in 95 watersheds , with rice paddies (green areas) and water bodies (blue areas)

=

X_1: log(areas of rice paddies +1)

x

X_2: log(areas of water bodies + 1)

Regular regression model:

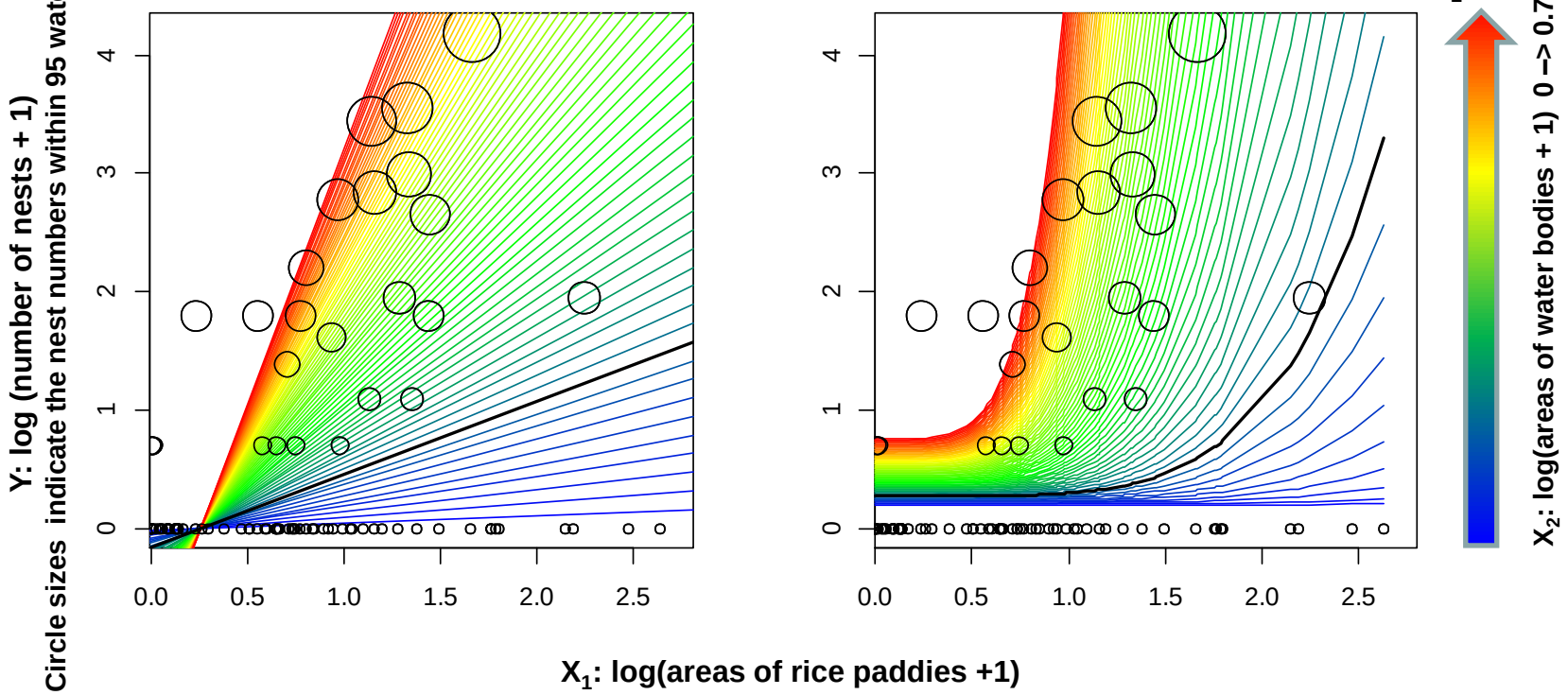
$Y \sim 4.74X_1X_2$

The slopes of Y-X_1 regression are $4.74X_2$

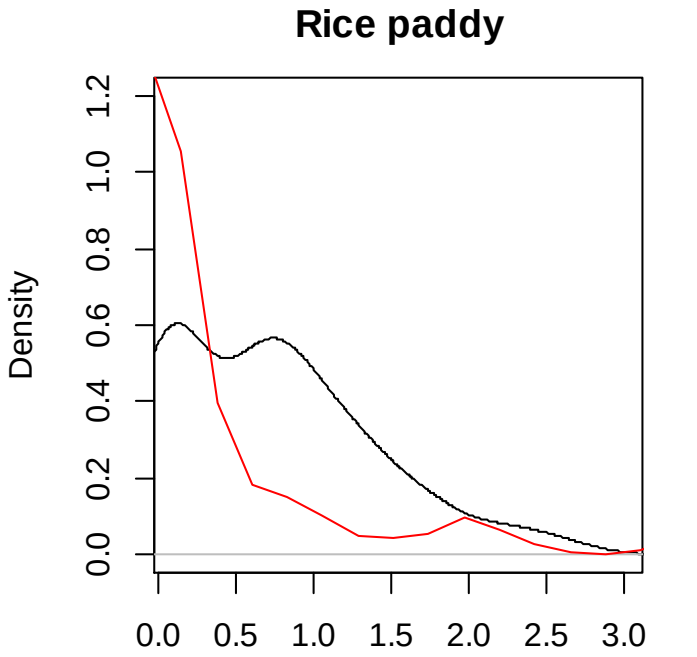
Fractional-power interaction regression:

$Y \sim 0.25 + 11.4X_1^{4.9} X_2^{2.6}$

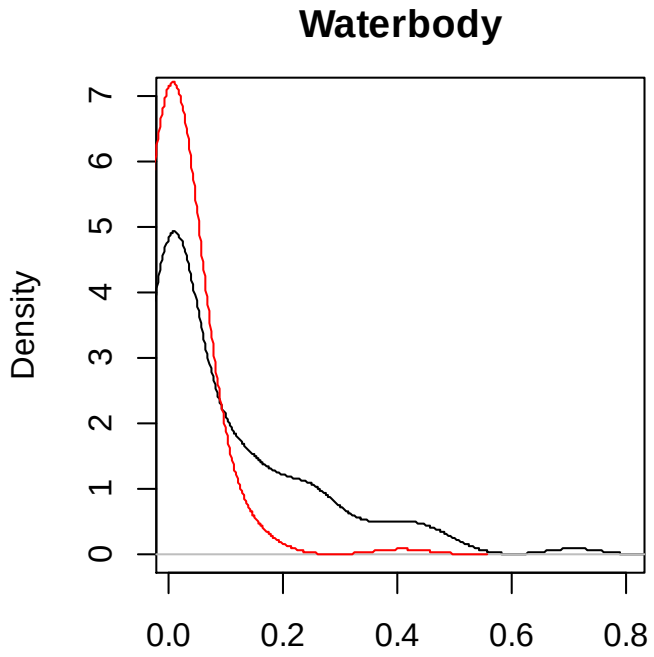
The slopes of Y-X_1 regression are $55.86\ X_1^{3.9} X_2^{2.6}$



The distribution of areas of rice paddies and waterbodies (black lines) in watersheds that the crested ibis occur, and distribution of power transformed values (red lines)

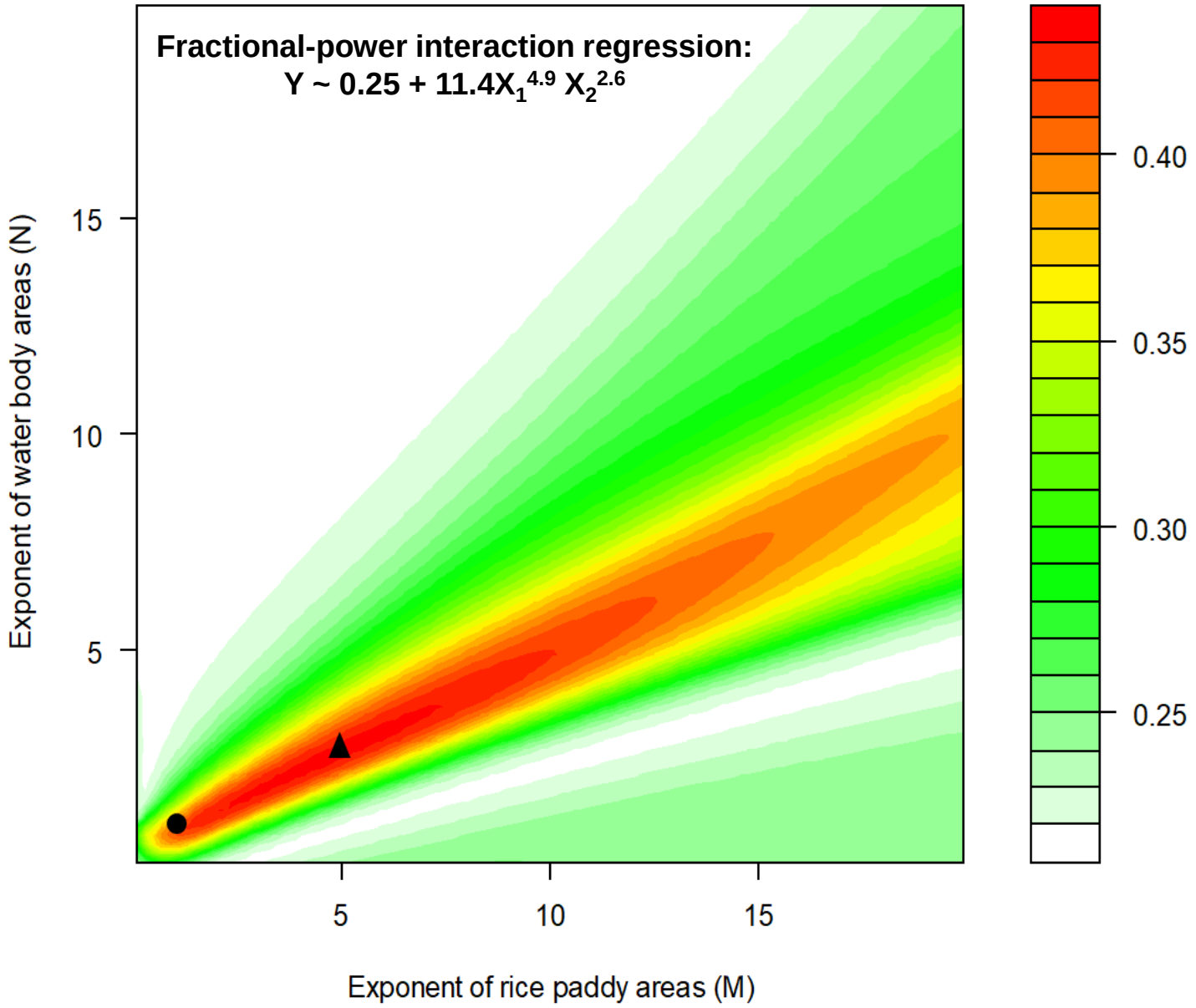


Black line: log(rice paddy areas + 1)
Red line: (log(rice paddy areas + 1))^4.9



Black line: log(waterbody areas + 1)
Red line: (log(waterbody areas + 1))^2.6

R square values using series powers of X_1 and X_2 in $Y \sim X_1 + X_2 + X_1^N X_2^M$ (N, M = 0.1, 0.2, …, 19.9, 20)

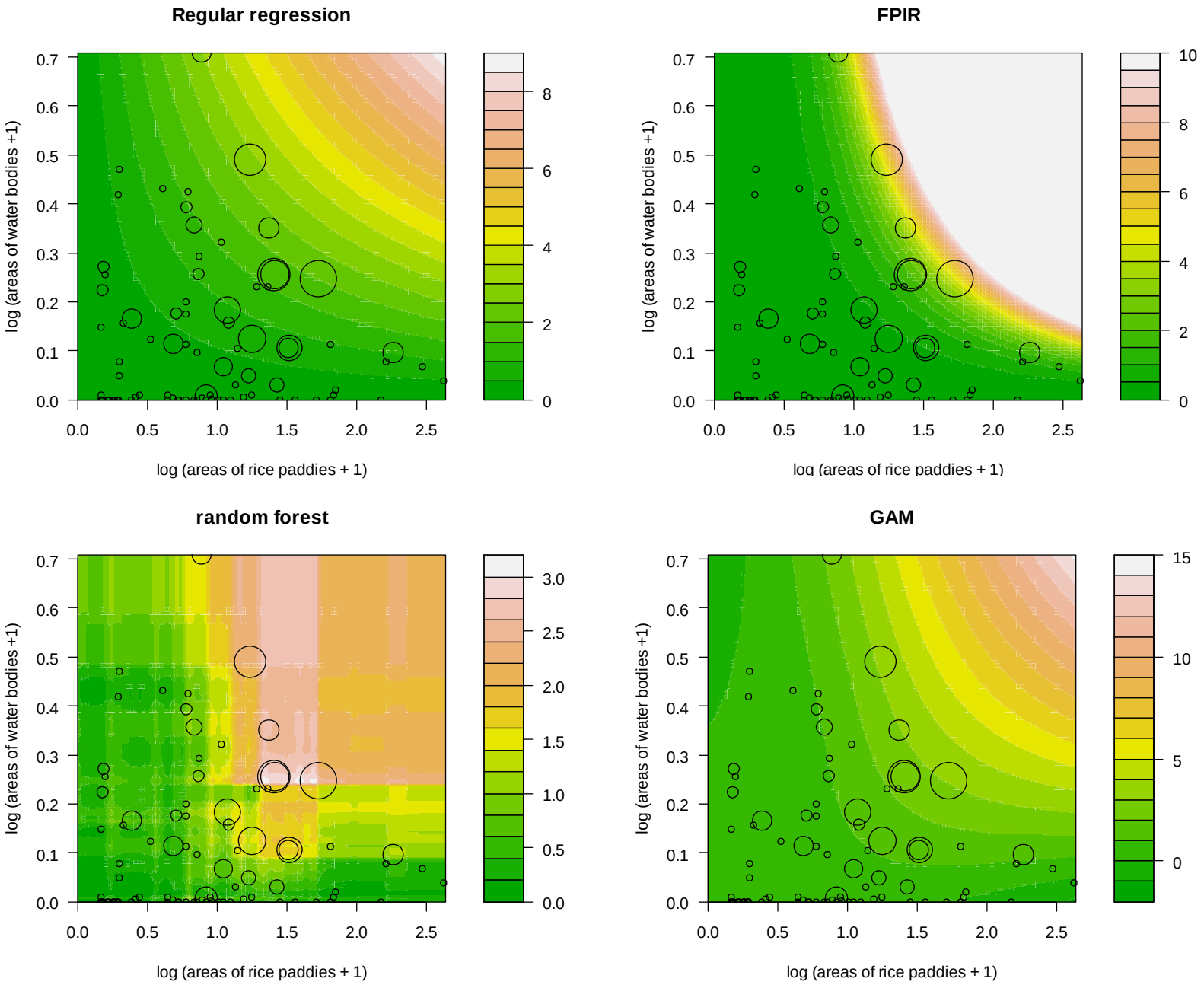


Nests <- b0 + 0.11 x rice_paddy + 0.90 x waterbody +
9.83 x rice_paddy$^{4.9}$ x 0.90 x waterbody$^{2.6}$

The best and worst six models

	Power1	Power2	Rsq_T	Rsq_I	coef_x1	coef_x2	coef_I
2345	4.5	2.4	0.43422	0.20989	0.09198	0.81234	9.03537
2651	5.1	2.7	0.43425	0.20992	0.12032	0.93755	10.25636
2344	4.4	2.4	0.43427	0.20993	0.09823	0.76034	9.36295
2446	4.6	2.5	0.43427	0.20994	0.10802	0.80806	9.76517
2447	4.7	2.5	0.43430	0.20996	0.10196	0.85683	9.42684
2549	**4.9**	**2.6**	**0.43431**	**0.20997**	**0.11139**	**0.89846**	**9.83374**
	Power1	Power2	Rsq_T	Rsq_I	coef_x1	coef_x2	coef_I
5914	1.4	6	0.21742	0.00000	0.41767	2.60707	-0.00038
7323	2.3	7.4	0.21742	0.00000	0.41767	2.60710	-0.00343
3406	0.6	3.5	0.21742	0.00000	0.41767	2.60717	-0.00092
8432	3.2	8.5	0.21742	0.00000	0.41767	2.60719	-0.01762
9340	4	9.4	0.21742	0.00000	0.41767	2.60725	-0.04105
5713	1.3	5.8	0.21742	0.00000	0.41767	2.60728	-0.00644

Model comparison



Testing FPIR using occurrences of 4692 species

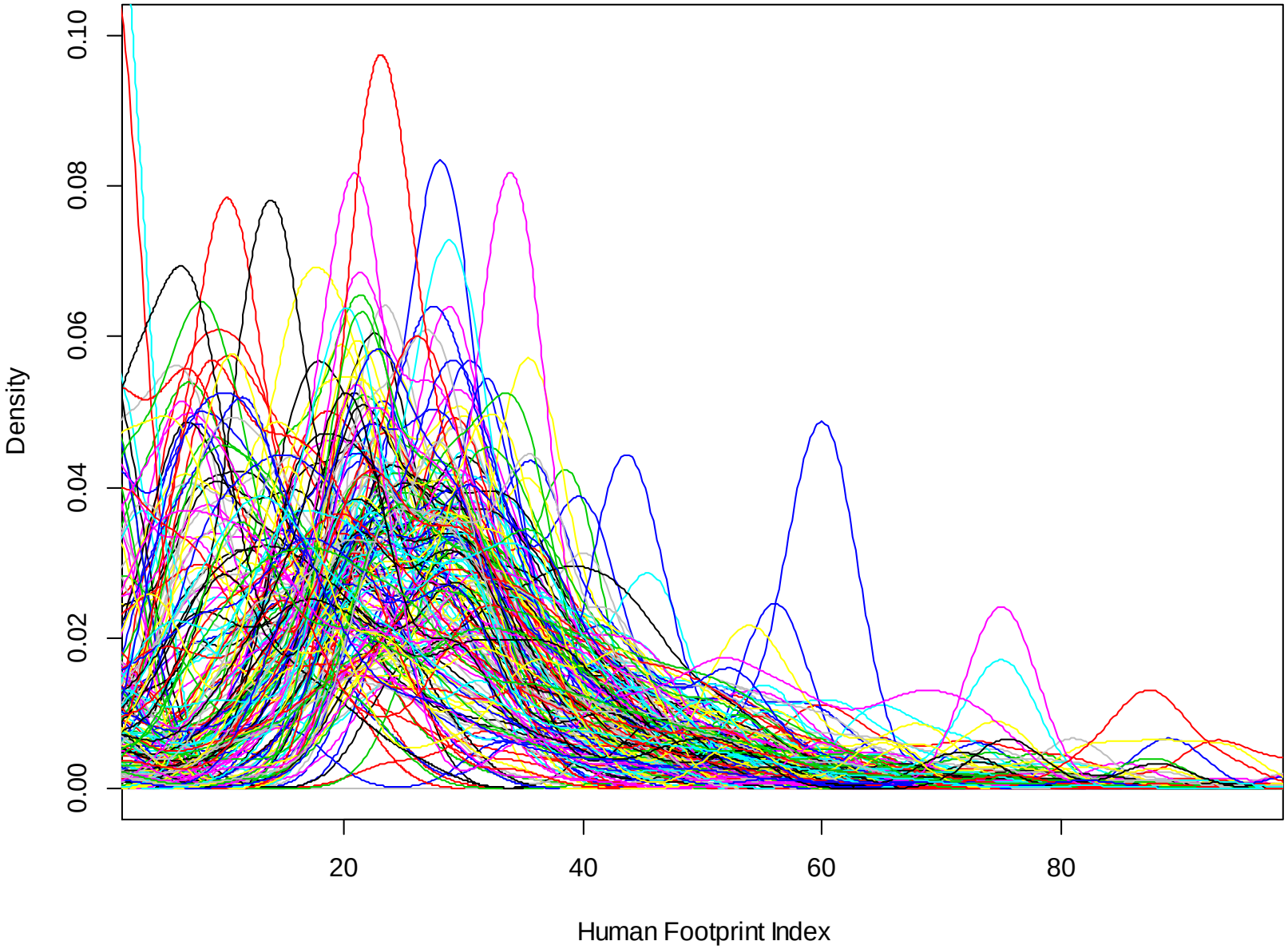
$$\mathbf{HFI \sim \beta_0 + \beta_1 E + \beta_2 P + \beta_3 E^M \times P^N + \varepsilon}$$

where *HFI* is human footprint index at the species occurrences, representing human impacts. *E* is elevation (m), *P* is annual total precipitation (mm/year), and *βs* are coefficients. To make the regression coefficients comparable, we standardized the three variables to the scale of 0-1.

Questions:

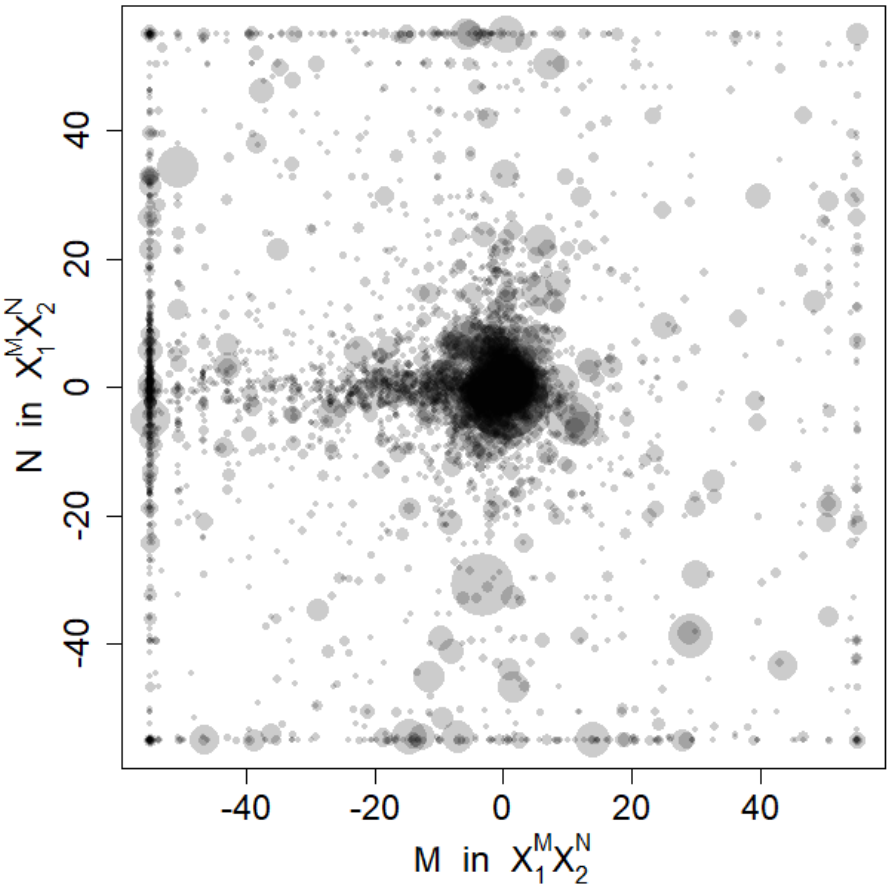
- Individuals occurring at lower-elevation areas are more tolerant of human impacts.
- Individuals occurring in wetter areas are more tolerant of human impacts.
- In lower-elevation areas, human impact and precipitation are positively correlated at the occurrence sites, whereas in higher areas the relationship is reversed. That is there is an interactive effect.
- The above interaction is a function of both elevation and precipitation in the form of $\beta E^M P^N$ (where E is elevation and P is annual total precipitation).

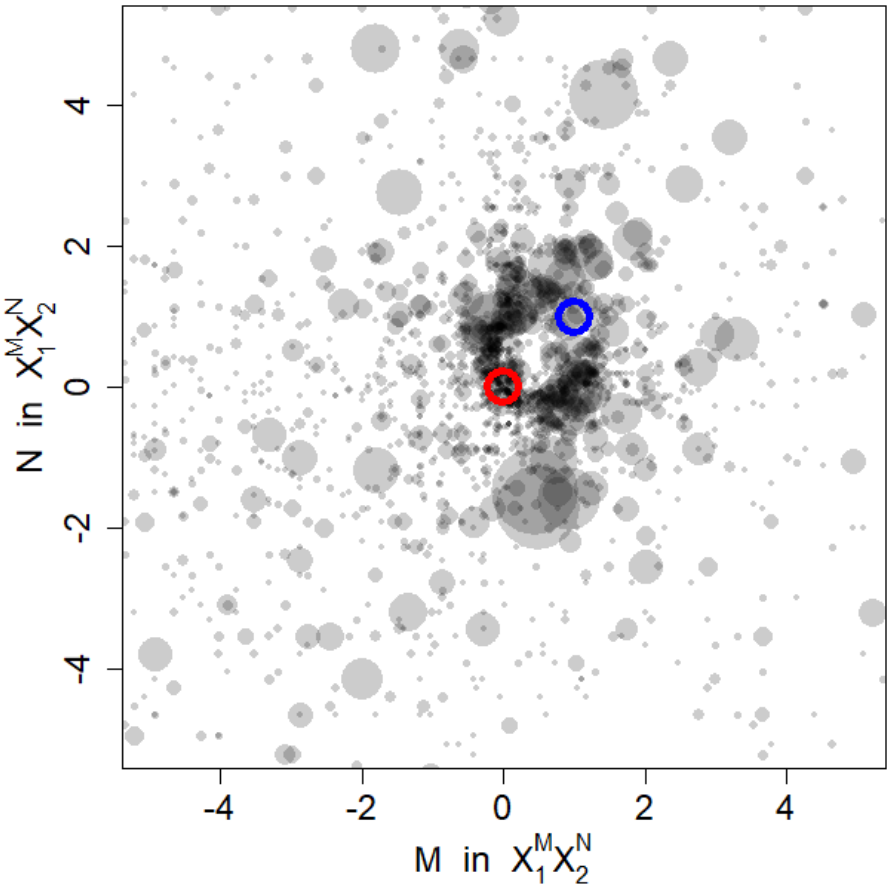
Distributions of human footprint index for 4692 species



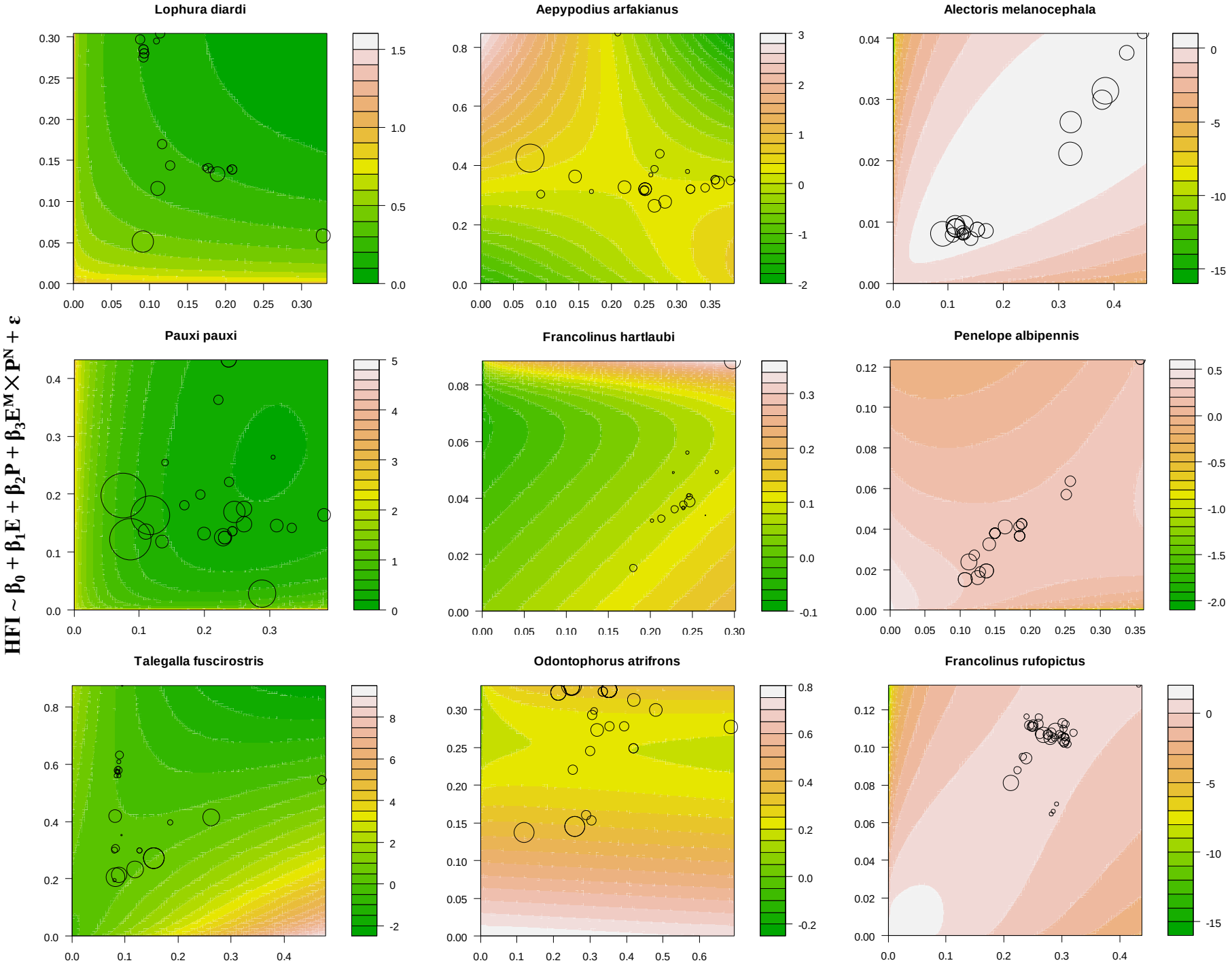
The occurrence data of taxa downloaded from GBIF website (4692 species)

Class	Order	Family	Species	Occurrences	GBIF DOI
Clitellata	/	/	334	175435	https://doi.org/10.15468/dl.4vlmaw
Insecta	Hymenoptera	Formicidae	2153	290125	https://doi.org/10.15468/dl.c9o5mh
Insecta	Hemiptera	Cicadidae	174	14585	https://doi.org/10.15468/dl.mqaniq
Arachnida	Araneae	Salticidae	281	48792	https://doi.org/10.15468/dl.383zw0
Amphibia	Anura	Hylidae	348	193922	https://doi.org/10.15468/dl.qjwkh1
Reptilia	Squamata	Colubridae	295	128290	https://doi.org/10.15468/dl.okmmxx
Reptilia	Squamata	Scincidae	595	244326	https://doi.org/10.15468/dl.nnyj0o
Aves	Galliformes	/	256	1151250	https://doi.org/10.15468/dl.lwji3z
Mammalia	Lagomorpha	/	50	198132	https://doi.org/10.15468/dl.oqcwcl
Mammalia	Artiodactyla	/	206	283468	https://doi.org/10.15468/dl.mj88eh



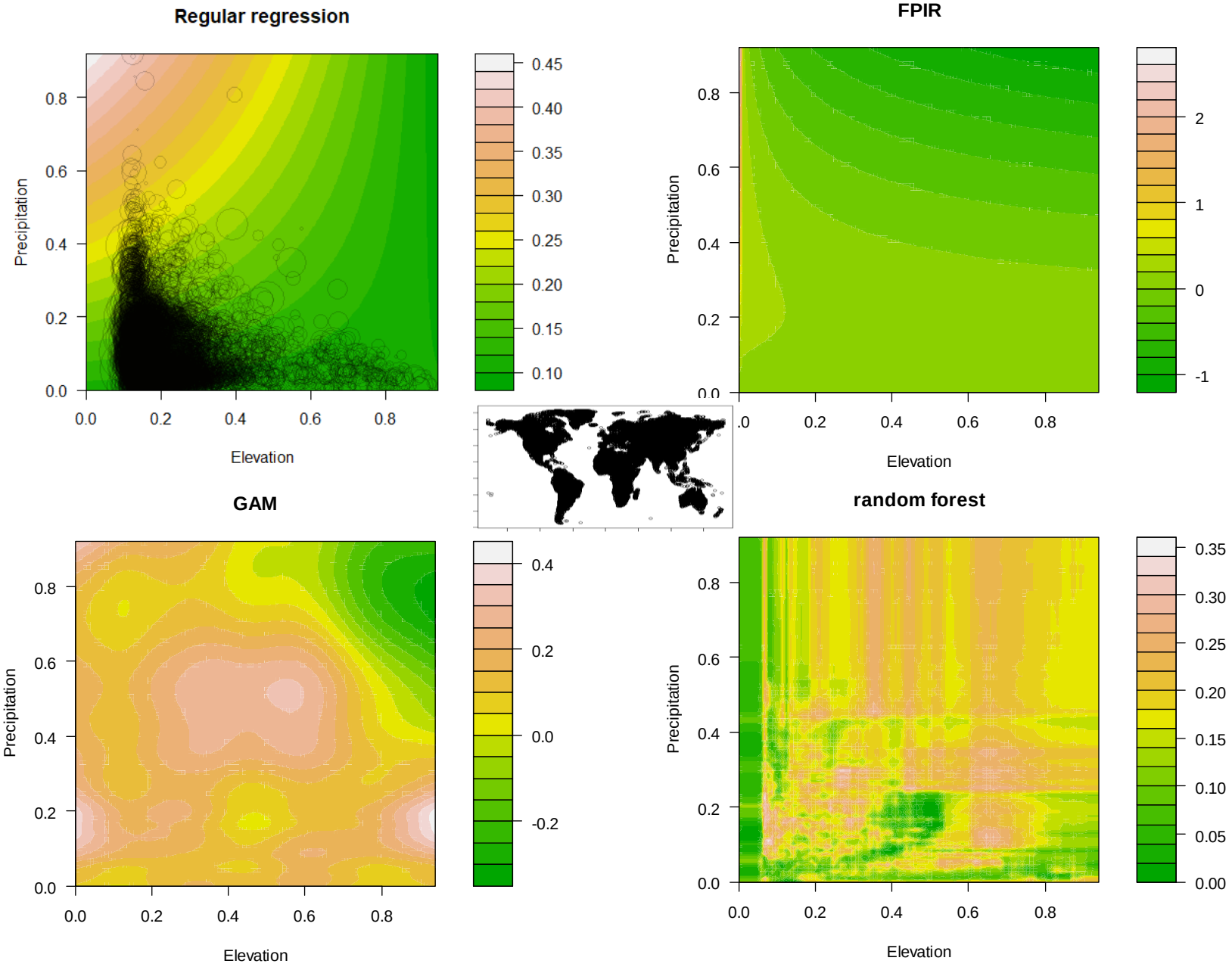


The values of parameters M and N in FPIR (HFI ~ E + P + $E^M \times P^N$ + ε) for the 4692 species within the range (-56 to 56, left panel) and the range (-5 to 5, right panel). The sizes of circles indicate the proportion of variance explained by the interaction term. At the right panel, the red circle shows the value zero for M and N (no interaction effect), and blue circle shows value one for M and N (traditional interaction effect) in regular regressions.

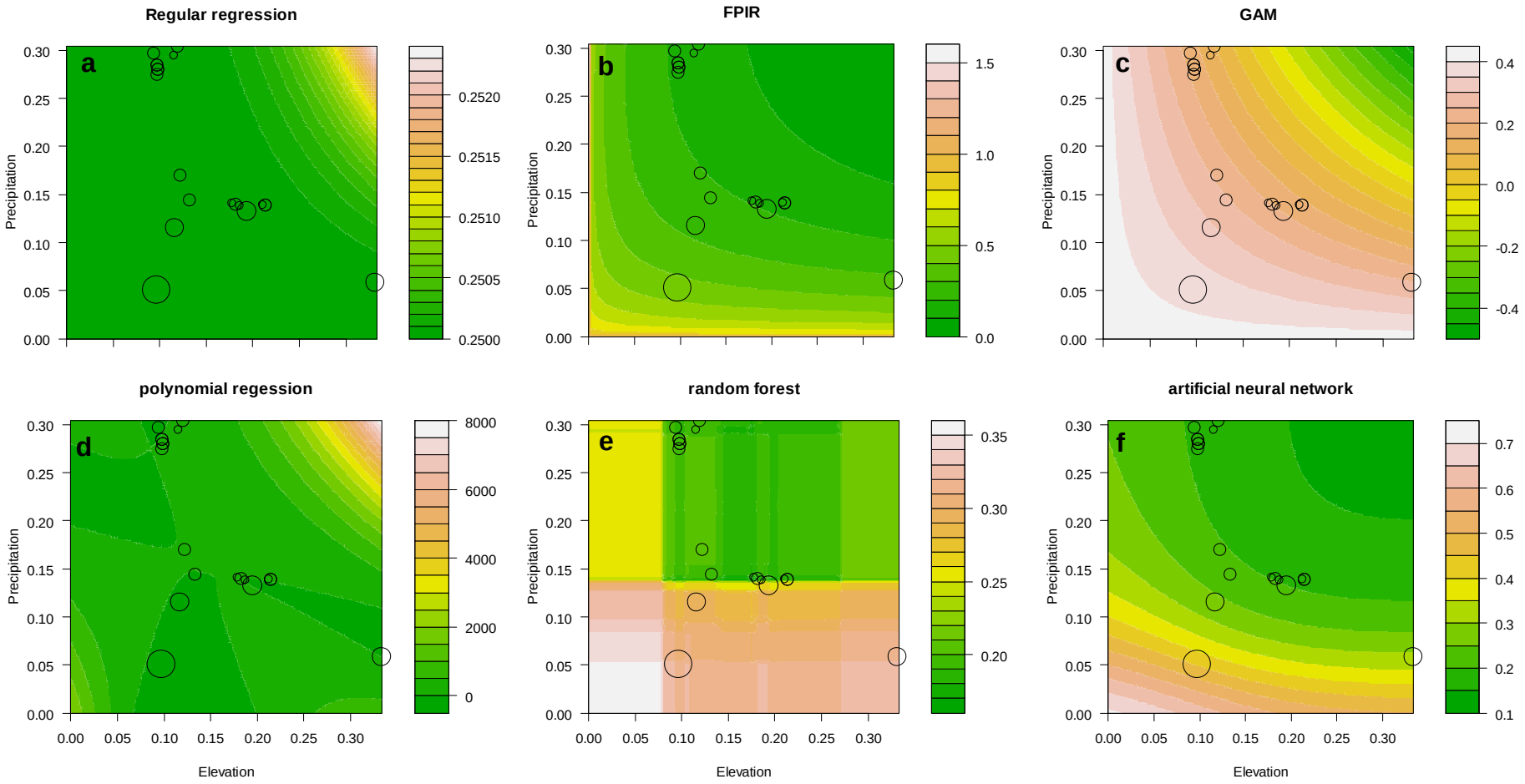

Lophura diardi
Aepypodius arfakianus
Alectoris melanocephala
Pauxi pauxi
Francolinus hartlaubi
Penelope albipennis
Talegalla fuscirostris
Odontophorus atrifrons
Francolinus rufopictus
$HFI \sim \beta_0 + \beta_1 E + \beta_2 P + \beta_3 E^M \times P^N + \varepsilon$

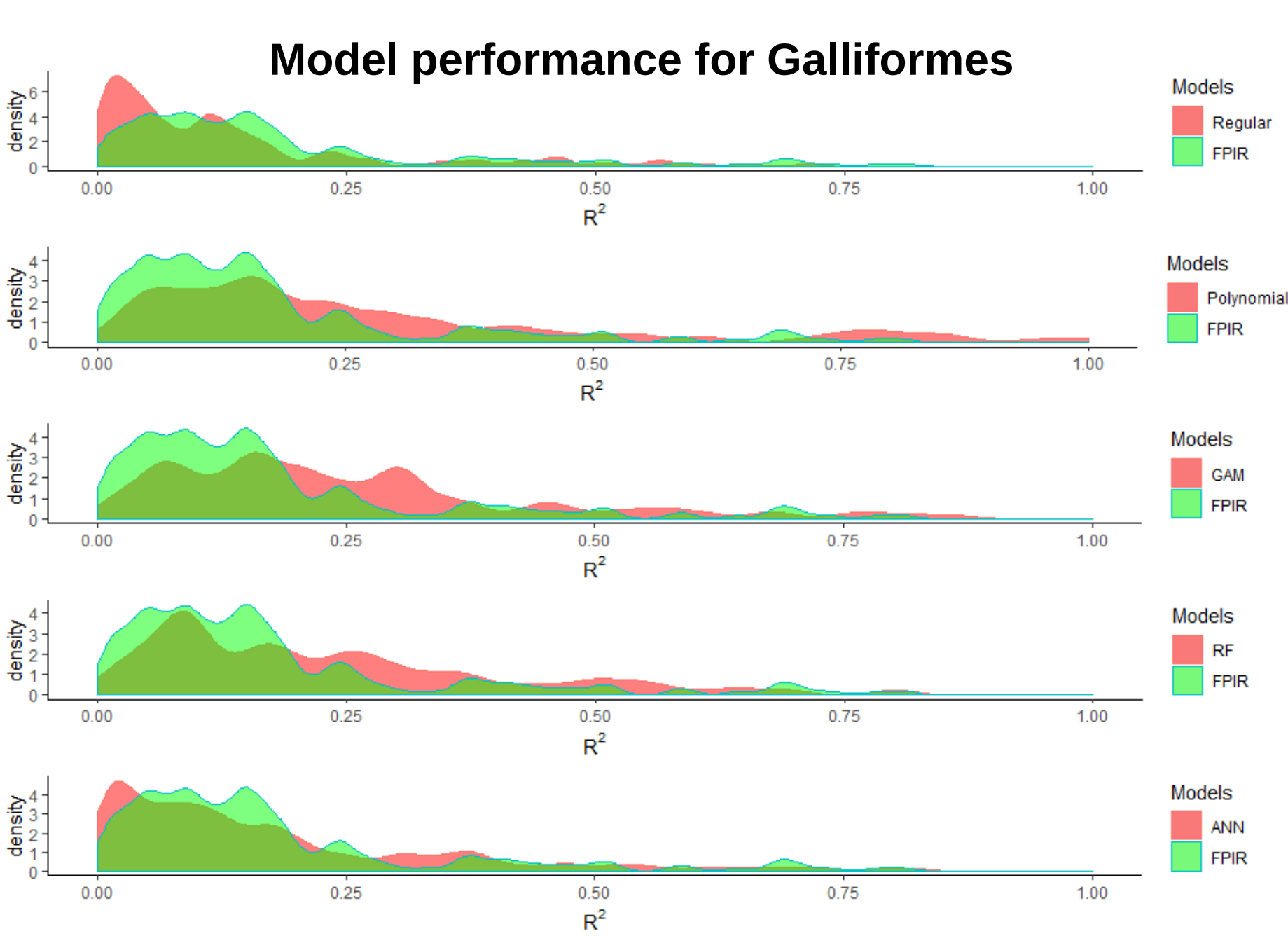
Background N=15677

SS: elevation, 0.0029; precipitation, 0.0427; interaction, 0.0006; residual, 0.9536



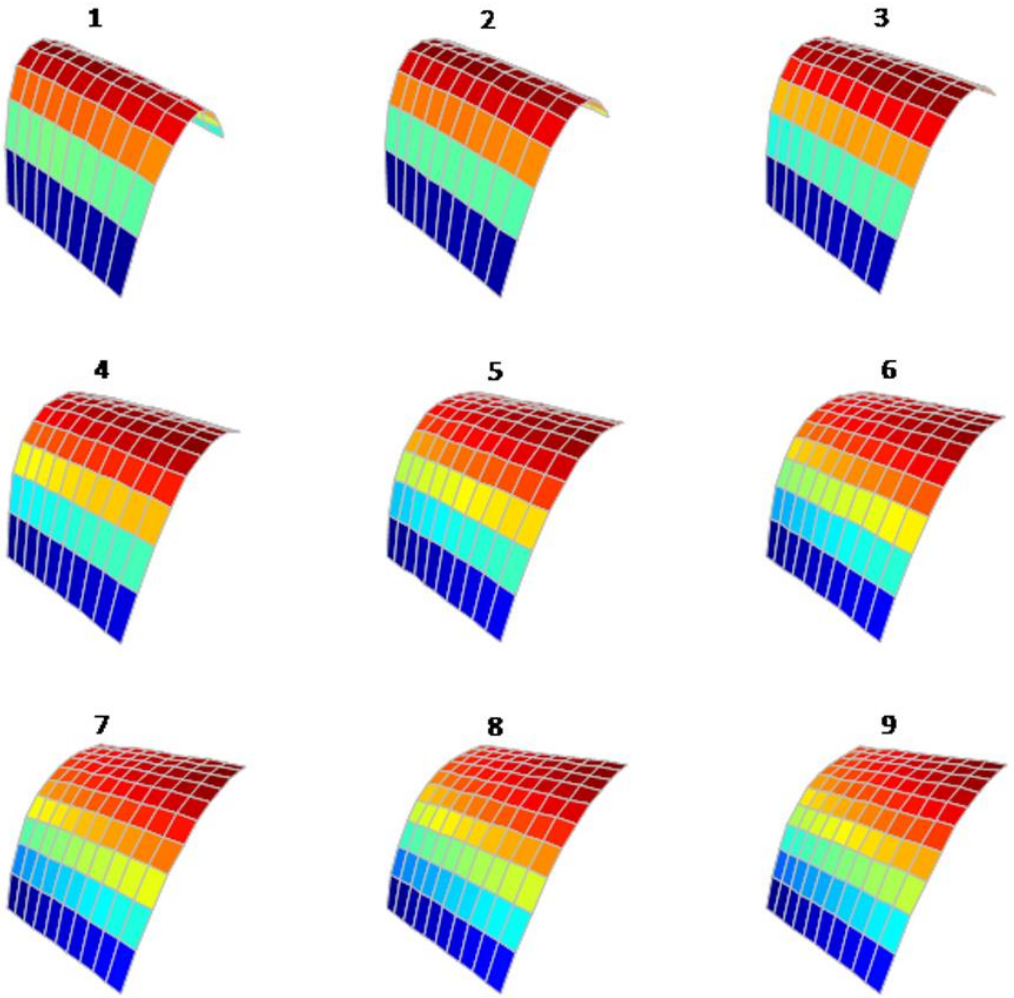
Lophura diardi 戴氏鹇




Model performance for Galliformes
density
R^2
0.00
0.25
0.50
0.75
1.00
Models
Regular
FPIR
Polynomial
GAM
RF
ANN

$$Y = X_1 X_2^{10} X_3^{0.1}$$

library(interactionFPIR)
FPIR1threeway (y, x1, x2, x3)



The R^2 values of the regression model $Y \sim X_1 + X_2 + X_3 + X_1^L X_2^M X_3^N$ (L, M, N ⊂ 0.1, 0.2, …, 9.9, 10) processed by fractional-power interaction regression (FPIR) using the simulated data $Y = X_1 X_2^{10} X_3^{0.1} + \varepsilon$. The perpendicular axis denotes the R^2 values, and the other two axes are gradients of L (0.5-1.5) and M (9.5-10.5). The nine panels correspond to the nine levels of N from 0.1 to 0.9.

FPIR sample code

```r
# Fractional-power interaction regression (FPIR)
# y = b0 + b1x1 + b2x2 + b3x1^b4 x2^b5
## FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
library(remotes)
install_github("Xinhai-Li/interaction", force = TRUE)
library(interactionFPIR)

attach(trees)
results = FPIR1twoway(trees$Volume, trees$Girth, trees$Height) # 1.35, 0.3

results[[1]] # parameters for 10000 models
results[[2]] # parameters for the best model
results[[3]] # regression coefficient
results[[4]] # adjusted R square#'
results2 = FPIR1twowaytune(trees$Volume, trees$Girth, trees$Height, 1.35, 0.3)
# tuned parameters: b4=1.175, b5=0.2

# simulated data
x1 = runif(100); x2 = runif(100);
y = x1 * x2^10; y = y + rnorm(length(y), 0, 0.01) # generate a dataset
results = FPIR1twoway (y, x1, x2)
Exp1 = results[[2]]$Ex1; Exp2 = results[[2]]$Ex2
results.2 = FPIR1twowaytune (y, x1, x2, Exp1, Exp2)
results.2[[1]] # parameters for 10000 models
results.2[[2]] # parameters for the best model
results.2[[3]] # regression coefficient
results.2[[4]] # adjusted R square


# Bayesian method
data(trees)
results = interaction_bayes(trees$Volume, trees$Girth, trees$Height, WinBUGS = "d:/softwares/WinBUGS14/", digits=2)
results
```

library(remotes)
install_github("Xinhai-Li/interaction", force = TRUE)
library(interactionFPIR)

attach(trees)
results = FPIR1twoway(trees$Volume, trees$Girth, trees$Height) # 1.35, 0.3

results[[1]] # parameters for 10000 models
results[[2]] # parameters for the best model
results[[3]] # regression coefficient
results[[4]] # adjusted R square#'
results2 = FPIR1twowaytune(trees$Volume, trees$Girth, trees$Height, 1.35, 0.3)
# tuned parameters: b4=1.175, b5=0.2

# simulated data
x1 = runif(100); x2 = runif(100);
y = x1 * x2^10; y = y + rnorm(length(y), 0, 0.01) # generate a dataset
results = FPIR1twoway (y, x1, x2)
Exp1 = results[[2]]$Ex1; Exp2 = results[[2]]$Ex2
results.2 = FPIR1twowaytune (y, x1, x2, Exp1, Exp2)
results.2[[1]] # parameters for 10000 models
results.2[[2]] # parameters for the best model
results.2[[3]] # regression coefficient
results.2[[4]] # adjusted R square

# Bayesian method
data(trees)
results = interaction_bayes(trees$Volume, trees$Girth, trees$Height, WinBUGS = "d:/softwares/WinBUGS14/", digits=2)
results

```
> results = interaction_bayes(trees$Volume, trees$Girth, trees$Height, WinBUGS = "d:/softwares/WinBUGS14/", digits=2)
> results
$`Summary table for model coefficients`
              mean         sd         2.5%          25%        50%        75%       97.5%     Rhat n.eff
b0       8.1438668 9.61732177 -11.02300000   1.76400000   8.139000  14.570000  27.9302500 0.999231  1000
b1      -6.1173109 1.96318598 -10.36150000  -7.37400000  -6.063000  -4.781500  -2.3842750 1.059456    58
b2      -0.4913433 0.15186226  -0.81022500  -0.59077500  -0.491050  -0.389675  -0.2020900 1.035948   100
b3       1.1233895 0.42050767   0.53899460   0.81135000   1.019499   1.319250   2.1710000 1.279454    16
b4       1.0136395 0.05082505   0.91037499   0.98117500   1.016000   1.043000   1.1240500 1.127224    47
b5       0.5193047 0.02994139   0.44910000   0.50510000   0.519700   0.539875   0.5686050 1.523735    11
tau      0.1118244 0.03252483   0.05805474   0.08968749   0.108400   0.131125   0.1840793 0.999002  1000
deviance 156.9628000 4.14221367 150.19749919 154.10000000 156.500000 159.400000 167.0024993 1.014906   210

$Formula
[1] "y = 8.14 - 6.12x1 - 0.49x2 + 1.12x1^1.01x2^0.52"
```

results = interaction_bayes(trees$Volume, trees$Girth, trees$Height, WinBUGS = "d:/softwares/WinBUGS14/", digits=2)

Log

stats(*)

Node statistics

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
b0	7.261	9.848	0.4196	-11.91	7.499	26.17	24	1000
b1	-6.088	2.499	0.2118	-11.22	-5.899	-1.549	24	1000
b2	-0.5015	0.2311	0.02288	-0.9935	-0.4626	-0.1333	24	1000
b3	1.283	0.9106	0.1054	0.2298	1.022	3.381	24	1000
b4	1.024	0.1162	0.01411	0.8502	1.033	1.271	24	1000
b5	0.5271	0.07775	0.009447	0.435	0.5046	0.7273	24	1000
deviance	157.0	5.636	0.5414	146.9	156.9	168.4	24	1000
tau	0.113	0.03517	0.002216	0.05463	0.1094	0.1888	24	1000

dic.stats()

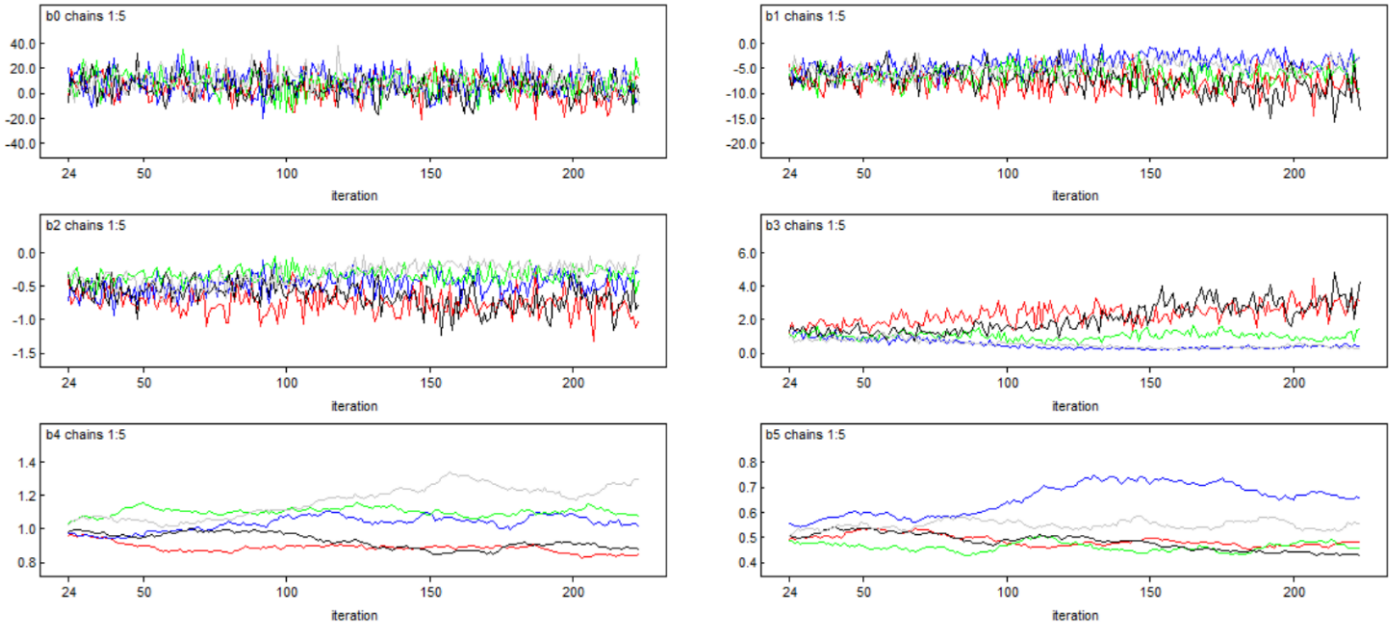
DIC

Dbar = post.mean of -2logL; Dhat = -2LogL at post.mean of stochastic nodes

	Dbar	Dhat	pD	DIC
y	157.049	5166.080	-5009.030	-4851.980
total	157.049	5166.080	-5009.030	-4851.980

history(*,C:/Users/ADMINI~1/AppData/Local/Temp/Rtmpua8nHd/history.odc)

History



results = interaction_bayes(DNo_nests, DRice_paddy, D$Water_body, WinBUGS = "d:/softwares/WinBUGS14/", digits=3)

Log

stats(*)

Node statistics

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
b0	1.095	0.7519	0.031	-0.3432	1.078	2.588	51	1000
b1	-0.08977	0.2857	0.0185	-0.7069	-0.07501	0.4312	51	1000
b2	2.586	3.253	0.1915	-4.253	2.676	8.499	51	1000
b3	1.063	1.508	0.1784	0.04488	0.609	6.015	51	1000
b4	10.43	2.796	0.3377	4.782	10.11	15.85	51	1000
b5	8.288	2.168	0.2615	3.85	7.995	12.51	51	1000
deviance	581.6	8.961	1.01	567.8	581.0	603.2	51	1000
tau	0.03803	0.006608	4.426E-4	0.02639	0.03751	0.05221	51	1000

dic.stats()

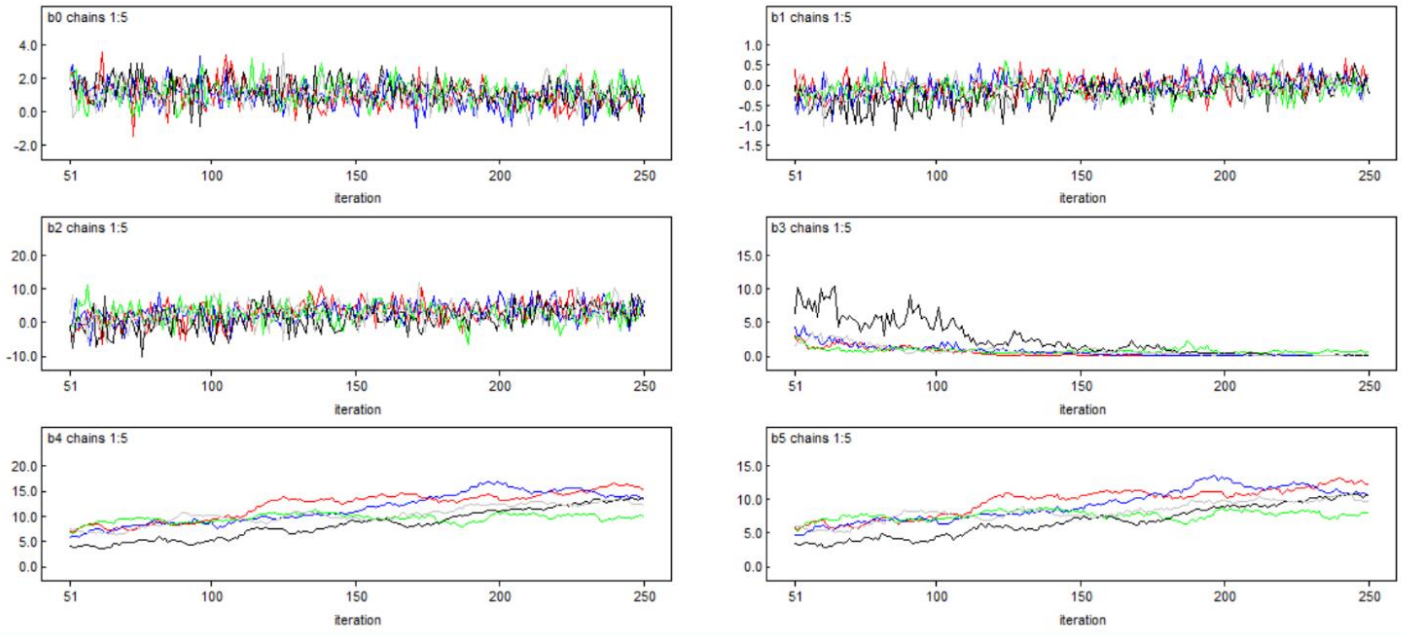
DIC

Dbar = post.mean of -2logL; Dhat = -2LogL at post.mean of stochastic nodes

	Dbar	Dhat	pD	DIC
y	581.599	746.713	-165.115	416.484
total	581.599	746.713	-165.115	416.484

history(*,C:/Users/ADMINI~1/AppData/Local/Temp/Rtmpua8nHd/history.odc)

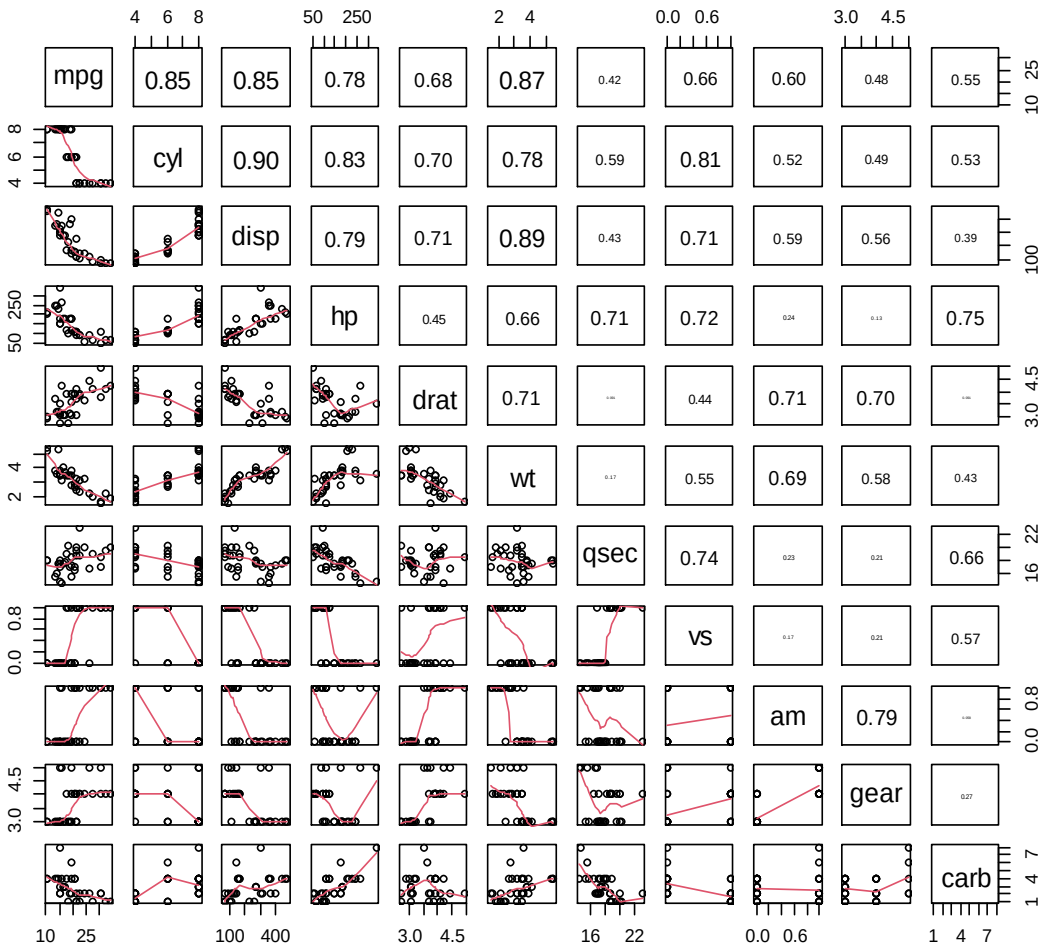
History



Pairwise correlation

R code – correlation plot

```
## put (absolute) correlations on the upper panels,
## with size proportional to the correlations.
panel.cor <- function(x, y, digits = 2, prefix = "", cex.cor, ...)
{
  usr <- par("usr"); on.exit(par(usr))
  par(usr = c(0, 1, 0, 1)) # ranges for x-axis and y-axis in plots
  r <- abs(cor(x, y))
  txt <- format(c(r, 0.123456789), digits = digits)[1]
  txt <- paste(prefix, txt, sep = "")
  if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)
  text(0.5, 0.5, txt, cex = cex.cor * r)
}

pairs(mtcars, lower.panel = panel.smooth,
        upper.panel = panel.cor)
```



Multiple correlation

Multiple correlation coefficient

- Correlation coefficient in the context of multiple regression
- *R* can be defined as the correlation between the criterion (*Y*) and the best linear combination of the predictors

$$R = r_{Y\hat{Y}}$$

$$\hat{Y} = b_0 + b_1 X_1 + b_2 X_2 + \cdots + b_p X_p$$

Partitioning of variance

- Sum of squares that is related to ***regression*** (*SSR*):

$$\sum(\hat{Y}_i - \bar{Y})^2 = SS_{\hat{Y}}$$

- Sum of squares that is related to ***residual*** (error; *SSE*):

$$\sum(Y_i - \hat{Y}_i)^2 = SS_{\text{residual}}$$

- Sum of squares related to ***total*** deviation (*SST*):

$$\sum(Y_i - \bar{Y})^2 = SS_Y$$

Multiple correlation coefficient (squared)

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

- R^2 can be interpreted in terms of percentage of accountable variation
- R^2 = 0.755: we can say that 75.5% of the variation in *Y* can be predicted on the basis of the *X*s

Partial regression coefficients

- Coefficients in multiple regression are called ***partial regression coefficients***

$$\hat{Y} = b_0 + b_1 X_1 + b_2 X_2 + \cdots + b_p X_p$$

- Example: b_1 is coefficient for regression of Y on X_1 when we ***partial out*** the effect of X_2, …, X_p
  - When other variables are held constant
- Common mistake: equate b_1 ***in the context of*** the other X_i with the simple regression coefficient when ***ignoring*** X_i.

Partial correlation

Partial correlation measures the degree of association between two random variables, with the effect of a set of controlling random variables (e.g., x_2) removed for both variables (e.g., y and x_1)

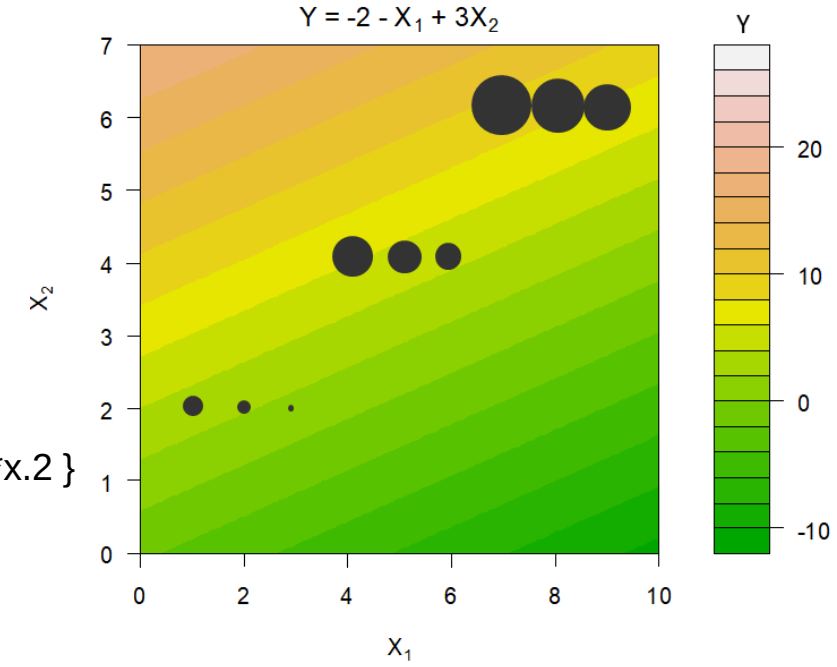
$y = x_1 + x_2$

$$Partial_{y1} = \frac{r_{y1} - (r_{y2})(r_{12})}{\sqrt{1 - r_{y2}^2}\sqrt{1 - (r_{12})^2}}$$

$$Semi - Partial_{y1} = \frac{r_{y1} - (r_{y2})(r_{12})}{\sqrt{1 - (r_{12})^2}}$$

Partial correlation example

```
y   <- c(3, 2, 1, 6, 5, 4, 9, 8, 7)
x1 <- c(1, 2, 3, 4, 5, 6, 7, 8, 9)
x2 <- c(2, 2, 2, 4, 4, 4, 6, 6, 6)
# multiple regression
fit = summary(lm(y ~ x1 + x2))
interception  = fit[[4]][1,1]
coef_x1       = fit[[4]][2,1]
coef_x2       = fit[[4]][3,1]
x.1 <- seq(min(x1) -1, max(x1) + 1, length= 100)
x.2 <- seq(min(x2) -2, max(x2) + 1, length= 100)
f <- function(x.1, x.2) { r <- interception + coef_x1*x.1 + coef_x2*x.2 }
y.pred <- outer(x.1, x.2, f)
filled.contour(x.1, x.2, y.pred, main="", color = terrain.colors,
                  xlab=expression(paste(X[1])),
                  ylab=expression(paste(X[2])),
                  ylim=c(1, 7))
points(x1/1.3, x2, pch=16, cex=y) # you may need adjust 1.3 to other values

library(ggm)# partial correlation
D = cbind(y, x1, x2)
D = jitter(D, factor = .01)
pcor(c("y", "x1"), var(D)) # 0.8
pcor(c("y", "x1","x2"), var(D)) # -0.9999
```



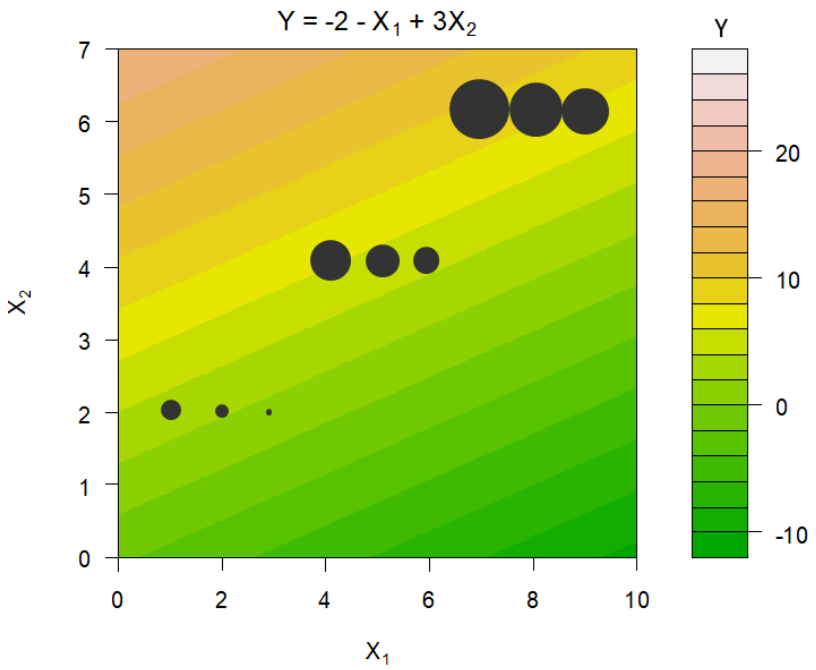
R code - partial correlation

```
# partial correlation
library(ggm)

## The marginal correlation between analysis and statistics
pcor(c("footprint", "GDP"), var(ibis.pre))
cor(ibis.pre$footprint, ibis.pre$GDP)
0.528

## The correlation between footprint and GDP given elevation
pcor(c("footprint", "GDP","elevation"), var(ibis.pre))
0.507

## The correlation between footprint and GDP given elevation and latitude
pcor(c("footprint", "GDP","elevation",“latitude"), var(ibis.pre))
0.5
```

Partial correlation example

- For any ***fixed*** value of X_2, slope of the regression line of Y on X_1 is negative (in fact $b_{01.2} = -0.99$)
- However, regression of Y on X_1 when ***ignoring*** X_2 is positive ($b_{01} = 0.8$)

Partialling out and ignoring are very different!



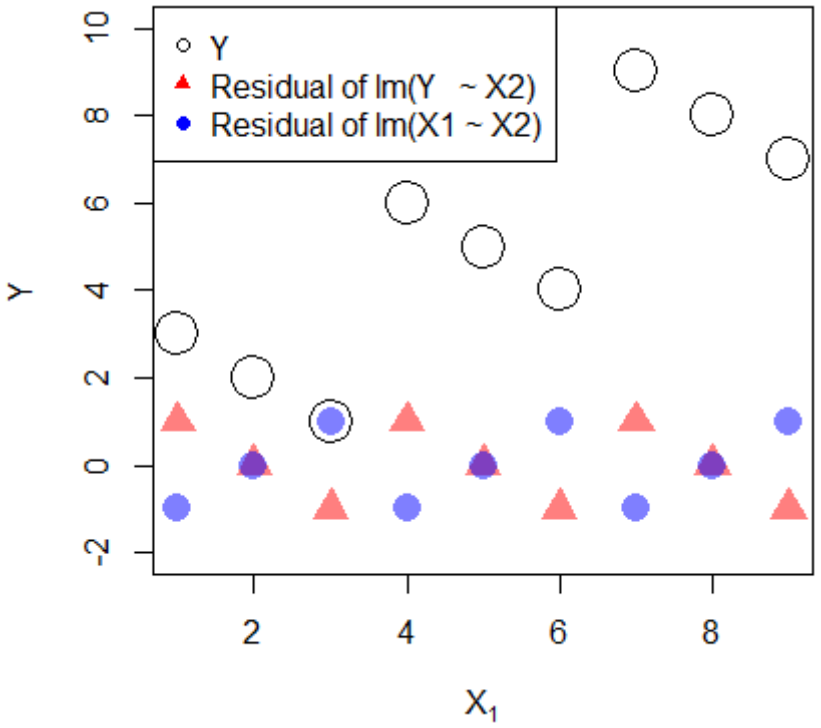
- When X_1 and X_2 are ***independent***, regression coefficients will be equal in both cases

Remove the effect of X_2

- Suppose we regress *Y* on X_2 and obtain the residual values $Y_r = Y_i - \hat{Y}_i$
- Residual values represent part of *Y* that cannot be predicted by X_2: ***independent*** of X_2
- Now regress X_1 on X_2 generating $X_{1r} = X_{1i} - \hat{X}_{1i}$
- Again, residual values represent part of X_1 that is ***independent*** of X_2
- We now have two sets of residuals: part of *Y* and part of X_1 that are ***independent*** of X_2
  - Partialled X_2 out of *Y* and out of X_1

Remove the effect of X_2

- Now regress Y_r on X_{1r}: regression coefficient will be the partial coefficient b
- Correlation between Y_r and X_{1r} is the ***partial correlation*** of Y and X_1, with X_2 partialled out: $r_{01.2}$



$$b_{Y_r X_{1r}} = \frac{\mathrm{cov}_{Y_r X_{1r}}}{s^2_{X_{1r}}} = -1$$

```
plot(x1, y, ylim=c(-2,10), pch=1, cex=3,
     xlab=expression(paste(X[1])), ylab="Y")
points(x1, resid.y, col=adjustcolor( "red", alpha.f = 0.5),
       pch=17, cex=2)
points(x1, resid.x1, col=adjustcolor( "blue", alpha.f = 0.5),
       pch=16, cex=2)
legend("topleft",
       c("Y", "Residual of lm(Y   ~ X2)", "Residual of lm(X1 ~ X2)"),
       pch=c(1, 17, 16), col=c(1,2,4))
```

Contribution, fraction, partial R^2

- Contribution of a variable x_j to the explanation of the variation of a dependent variable y.
- Fraction [a] in variation partitioning.
- Partial R^2 (partial determination coefficient) between an x_j and a y variable.

Contribution

Scherrer (1984) called the quantity $a_j * r_{yxj}$ the "contribution" of the *j*-th variable to the explanation of the variance of *y;*

– a_j is the standardized regression coefficient of the *j*-th explanatory variable,

– r_{yxj} is the simple correlation coefficient (Pearson *r*) between *y* and x_j.

Fraction [a] in variation partitioning

semipartial correlation squared

- This fraction measures the proportion of the variance of *y* explained by the explanatory variable x_1 (for example) when the other explanatory variables (x_2, x_3...) are held constant **with respect to x_1 only** (and not with respect to *y*).

- Thus, one obtains fraction [a] by examining the r^2 obtained by regressing *y* on the residuals of a regression of x_1 on x_2, x_3...

Partial R^2

- The **partial *R*** measures the mutual relationship between two variables *y* and x_j when other variables (x_1, x_2, x_3...) are held constant **with respect to the two variables involved y and x_j**

- The **partial R^2** is the square of the partial *R* above. For y = x_1+x_2, it measures the proportion of the variance of the residuals of *y* with respect to x_2 that is explained by the residuals of x_1 with respect to x_2.

R code for contribution, fraction, partial R^2 and variance partitioning

```
mtcars; mtcars.st = scale(mtcars); apply(mtcars.st, 2, var)
mtcars.st = as.data.frame(mtcars.st)

fit = lm(mpg ~ cyl + disp + hp + drat + wt + qsec + vs + am + gear + carb, data=mtcars.st)

# Contribution
Contribution = coef(fit) * cor(mtcars.st)[1,]
fit2 = step(fit)
Contribution = coef(fit2)[-1] * cor(mtcars.st)[1, c(6,7,9)]

# Fraction
f.wt = lm(wt ~ qsec + am, data=mtcars.st)
res.wt = resid(f.wt)
Fraction.a = summary(lm(mtcars.st$mpg ~ res.wt))$r.squared

# Partial.R2
f.mpg = lm(mpg ~ qsec + am, data=mtcars.st)
res.mpg = resid(f.mpg)
Partial.R2 = summary(lm(res.mpg ~ res.wt))$r.squared

# Variance partition
fit = lm(mpg ~ wt + qsec + am, data=mtcars.st)
anova(fit)[[2]] / sum(anova(fit)[[2]])
```

	wt	qsec	am
Contribution	0.5516	0.1521	0.1458
Fraction [a]	0.1628	0.0968	
Partial R^2	0.5199	0.1175	
Variance	0.7528	0.0735	0.0232

'wt' (car weight measured in tons)
'qsec' (the number of second a car takes to reach .25 miles)
'am' (transmission type)

Variance partitioning

```
mpg = mtcars$mpg
env = mtcars[,-1]
hier.part(mpg, env, fam = "gaussian", gof = "Rsqu")

lm.1=lm(mpg ~.,mtcars)
library(car)
Anova(lm.1, type=3)

library(relaimpo)
calc.relimp(lm.1)

library(dominanceanalysis)
da<-dominanceAnalysis(lm.1)
summary(da)

library(rdacca.hp)
rdacca.hp(mpg, env, method = "RDA", type = "R2")
```

	I	J	Total	ind.exp.var
cyl	0.110	0.616	0.726	14.593
disp	0.107	0.611	0.718	14.114
hp	0.097	0.506	0.602	12.767
drat	0.063	0.401	0.464	8.287
wt	0.134	0.619	0.753	17.722
qsec	0.030	0.146	0.175	3.900
vs	0.059	0.382	0.441	7.824
am	0.058	0.302	0.360	7.618
gear	0.032	0.198	0.231	4.290
carb	0.067	0.236	0.304	8.884

	Unique	Average .share	Individual	I.Perc (%)
cyl	0.0001	0.1212	0.1213	13.96
disp	0.0035	0.1167	0.1202	13.83
hp	0.0061	0.1009	0.107	12.31
drat	0.0014	0.0719	0.0733	8.43
wt	0.024	0.1343	0.1583	18.22
qsec	0.0079	0.0304	0.0383	4.41
vs	0.0001	0.0658	0.0659	7.58
am	0.0094	0.064	0.0734	8.45
gear	0.0012	0.0429	0.0441	5.07
carb	0.0004	0.0669	0.0673	7.74

Canonical correlation analysis

- In CCA, there can be multiple response variables.
- Canonical correlations are the maximum correlation between a linear combination of the responses and a linear combination of the predictor variables.

Given a linear combination of X variables:

$$F = f_1X_1 + f_2X_2 + ... + f_pX_p$$

and a linear combination of Y variables:

$$G = g_1Y_1 + g_2Y_2 + ... + g_qY_q$$

The **first canonical correlation** is:

Maximum correlation coefficient between F and G,

for all F and G

$F_1=\{f_{11},f_{12},...,f_{1p}\}$ and $G_1=\{g_{11},g_{12},...,g_{1q}\}$
are corresponding **canonical variates**

One example of CCA

```
# http://www.ats.ucla.edu/stat/r/dae/canonical.htm

require(ggplot2)
require(GGally)
require(CCA)

# Example 1. A researcher has collected data on three psychological variables, four academic variables (standardized test scores)
# and gender for 600 college freshman. She is interested in how the set of psychological variables relates to the academic
# variables and gender.  In particular, the researcher is interested in how many dimensions (canonical variables) are necessary to
# understand the association between the two sets of variables.

mm <- read.csv("http://www.ats.ucla.edu/stat/data/mmreg.csv")
colnames(mm) <- c("Control", "Concept", "Motivation", "Read", "Write", "Math",  "Science", "Sex")
summary(mm); head(mm)

psych <- mm[, 1:3]
acad <- mm[, 4:8]

ggpairs(psych)
ggpairs(acad)

# correlations within and between the two sets of variables
matcor(psych, acad)  #CCA package
```

Control	Concept	Motivation	Read	Write	Math	Science	Sex
-0.84	-0.24	1	54.8	64.5	44.5	52.6	1
-0.38	-0.47	0.67	62.7	43.7	44.7	52.6	1
0.89	0.59	0.67	60.6	56.7	70.5	58	0
0.71	0.28	0.67	62.7	56.7	54.7	58	0
-0.64	0.03	1	41.6	46.3	38.4	36.3	1
1.11	0.9	0.33	62.7	64.5	61.4	58	1

One example of CCA

```
# Canonical Correlation Analysis
cc1 <- cc(psych, acad)
summary(cc1)
```

```
##             Length  Class   Mode
## cor         3       -none-  numeric
## names       3       -none-  list
## xcoef       9       -none-  numeric
## ycoef       15      -none-  numeric
## scores      6       -none-  list
```

```
# display the canonical correlations
cc1$cor
```

```
## [1] 0.4641  0.1675  0.1040
```

```
# raw canonical coefficients
cc1[3:4]
```

```
## $xcoef
##                [,1]      [,2]      [,3]
## Control    -1.2538   -0.6215   -0.6617
## Concept     0.3513   -1.1877    0.8267
## Motivation -1.2624    2.0273    2.0002
##
## $ycoef
##                [,1]      [,2]      [,3]
## Read       -0.044621 -0.004910  0.021381
## Write      -0.035877  0.042071  0.091307
## Math       -0.023417  0.004229  0.009398
## Science    -0.005025 -0.085162 -0.109835
## Sex        -0.632119  1.084642 -1.794647
```

http://www.ats.ucla.edu/stat/r/dae/canonical.htm

```
# compute canonical loadings
cc2 <- comput(psych, acad, cc1)
```

```
# display canonical loadings
cc2[3:6]
```

```
## $corr.X.xscores
##                [,1]      [,2]      [,3]
## Control    -0.90405  -0.3897  -0.1756
## Concept    -0.02084  -0.7087   0.7052
## Motivation -0.56715   0.3509   0.7451
##
## $corr.Y.xscores
##                [,1]      [,2]      [,3]
## Read       -0.3900   -0.06011   0.01408
## Write      -0.4068    0.01086   0.02647
## Math       -0.3545   -0.04991   0.01537
## Science    -0.3056   -0.11337  -0.02395
## Sex        -0.1690    0.12646  -0.05651
##
## $corr.X.yscores
##                [,1]      [,2]      [,3]
## Control    -0.419555 -0.06528  -0.01826
## Concept    -0.009673 -0.11872   0.07333
## Motivation -0.263207  0.05878   0.07749
##
## $corr.Y.yscores
##                [,1]      [,2]      [,3]
## Read       -0.8404   -0.35883   0.1354
## Write      -0.8765    0.06484   0.2546
## Math       -0.7639   -0.29795   0.1478
## Science    -0.6584   -0.67680  -0.2304
## Sex        -0.3641    0.75493  -0.5434
```

One example of CCA

In general, the number of canonical dimensions is equal to the number of variables in the smaller set, however, the number of significant dimensions may be even smaller.

Canonical dimensions, also known as canonical variates, are latent variables that are analogous to factors obtained in factor analysis.

For this particular model there are three canonical dimensions of which only the first two are statistically significant.

```
# tests of canonical dimensions
ev <- (1 - cc1$cor^2)

n <- dim(psych)[1]
p <- length(psych)
q <- length(acad)
k <- min(p, q)
m <- n - 3/2 - (p + q)/2
w <- rev(cumprod(rev(ev)))

# initialize
d1 <- d2 <- f <- vector("numeric", k)

for (i in 1:k) {
  s <- sqrt((p^2 * q^2 - 4)/(p^2 + q^2 - 5))
  si <- 1/s
  d1[i] <- p * q
  d2[i] <- m * s - p * q/2 + 1
  r <- (1 - w[i]^si)/w[i]^si
  f[i] <- r * d2[i]/d1[i]
  p <- p - 1
  q <- q - 1
}

pv <- pf(f, d1, d2, lower.tail = FALSE)
(dmat <- cbind(WilksL = w, F = f, df1 = d1, df2 = d2, p = pv))
```

http://www.ats.ucla.edu/stat/r/dae/canonical.htm

Assignment

General objectives: learn about multiple linear regression.

- Make a dataset ready, including at least three continuous variables Y, X1 and X2 (X3 and X4 are suggested to be included).
- Check multicollinearity (column relationship) and independence (row relationship).
- Start from the full model, including all quadratic terms and interaction terms

```
fit = lm(Y ~ X1 + X2 + I(X1^2) + I(X2^2) + X1:X2, data=mydata).
```

- Run model selection and remove insignificant variables and terms.
- Report R^2, significance of each variables and terms, homogeneous of residuals.
- Briefly interpret the results.

R code – correlation plot

```
## put (absolute) correlations on the upper panels,
## with size proportional to the correlations.
panel.cor <- function(x, y, digits = 2, prefix = "", cex.cor, ...)
{
  usr <- par("usr"); on.exit(par(usr))
  par(usr = c(0, 1, 0, 1)) # ranges for x-axis and y-axis in plots
  r <- abs(cor(x, y))
  txt <- format(c(r, 0.123456789), digits = digits)[1]
  txt <- paste(prefix, txt, sep = "")
  if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)
  text(0.5, 0.5, txt, cex = cex.cor * r)
}

pairs(ibis.pre, lower.panel = panel.smooth,
        upper.panel = panel.cor)
```