**PPM OUTPUT EXPLAINED**

Given the following model…

* dim(spxyz) = 28 13
* dim(ppmxyz) = 200028 14
* names(ppmxyz) = "X" "Y" "landuse" "bio1" "bio12" "bio15" "bulkdens" "pawc" "soilcarb" "srtm" "slope" "aspect" "Pres" "wt"
* # covariates = 12 (landuse, bio1, bio12, bio15, bulkdens, pawc, soilcarb, srtm, slope, aspect) + (X, Y)
* # interaction terms = X:Y
* ppmform = ~poly(X, Y, degree = 2, raw = FALSE) + poly(landuse, degree = 2, raw = FALSE)
* mod <- try(ppmlasso(formula = ppmform, data = ppmxyz, n.fits = n.fits, criterion = "bic",standardise = FALSE), silent=TRUE)

The output is…

> str(mod)

List of 27

$ betas : num [1:8, 1:51] -5.29 0 0 0 0 ...

$ lambdas : num [1:51] 0.1215 0.1034 0.088 0.0749 0.0638 ...

$ likelihoods : num [1:51] -244 -243 -226 -210 -201 ...

$ pen.likelihoods : num [1:51] -244 -244 -243 -238 -233 ...

$ lambda : num 4.54e-05

$ beta : num [1:8] -7.39 412.62 551.34 -612.07 0 ...

$ mu : num [1:200028, 1] 0.108 0.108 0.108 0.108 0.108 ...

..- attr(\*, "dimnames")=List of 2

.. ..$ : chr [1:200028] "687" "688" "689" "690" ...

.. ..$ : NULL

$ likelihood : num -158

$ criterion : chr "bic"

$ family : chr "poisson"

$ gamma : num 0

$ alpha : num 1

$ init.coef : logi NA

$ criterion.matrix:'data.frame': 51 obs. of 5 variables:

..$ AIC : num [1:51] 490 491 459 429 410 ...

..$ BIC : num [1:51] 491 493 465 434 415 ...

..$ HQC : num [1:51] 490 491 461 431 411 ...

..$ GCV : num [1:51] 11.38 11.33 10.2 9.25 8.66 ...

..$ NLGCV: num [1:51] 10.58 10.53 9.27 8.18 7.5 ...

$ data : num [1:200028, 1:8] 1 1 1 1 1 1 1 1 1 1 ...

..- attr(\*, "dimnames")=List of 2

.. ..$ : chr [1:200028] "687" "688" "689" "690" ...

.. ..$ : chr [1:8] "(Intercept)" "poly(X, Y, degree = 2, raw = FALSE)1.0" "poly(X, Y, degree = 2, raw = FALSE)2.0" "poly(X, Y, degree = 2, raw = FALSE)0.1" ...

$ pt.interactions : logi NA

$ wt : num [1:200028] 1e-06 1e-06 1e-06 1e-06 1e-06 1e-06 1e-06 1e-06 1e-06 1e-06 ...

$ pres : num [1:200028] 1 1 1 1 1 1 1 1 1 1 ...

$ x : num [1:200028] 146 146 146 146 146 ...

$ y : num [1:200028] -17.1 -17.1 -17.1 -17.1 -17.1 ...

$ r : logi NA

$ call : language ppmlasso(formula = ppmform, data = ppmxyz, n.fits = n.fits, criterion = "bic", standardise = FALSE)

$ formula :Class 'formula' language ~poly(X, Y, degree = 2, raw = FALSE) + poly(landuse, degree = 2, raw = FALSE)

.. ..- attr(\*, ".Environment")=<environment: R\_GlobalEnv>

$ s.means : NULL

$ s.sds : NULL

$ cv.group : num [1:200028] 0 0 0 0 0 0 0 0 0 0 ...

$ n.blocks : logi NA

- attr(\*, "class")= chr [1:2] "ppmlasso" "list"

**NUMBER OF COEFFICIENTS (mod$beta)**

* If no categorical covariates (i.e. landuse): Number of covariates \* number of degrees of freedom + + 2 spatial covariates (X & Y) + 1 interaction term (only between X & Y)
* If categorical covariate is used (i.e. landuse): (Number of covariates – Number of categorical covariates) \* number of degrees of freedom + + 2 spatial covariates (X & Y) + 1 interaction term (only between X & Y) + sum over number of categorical covariates (number of classes in categorical covariate)

**Example: ~ poly(X, Y, degree = 2, raw = FALSE) + poly(landuse, degree = 2, raw = FALSE)**

1. (Intercept)
2. X poly(X, Y, degree = 2, raw = FALSE)1.0
3. X^2 poly(X, Y, degree = 2, raw = FALSE)2.0
4. Y poly(X, Y, degree = 2, raw = FALSE)0.1
5. Y^2 poly(X, Y, degree = 2, raw = FALSE)0.2
6. X:Y poly(X, Y, degree = 2, raw = FALSE)1.1
7. landuse poly(landuse, degree = 2, raw = FALSE)1
8. laduse^2 poly(landuse, degree = 2, raw = FALSE)2"

**Example: *~poly(X, Y, srtm, degree = 2, raw = FALSE) + poly(bio1, degree = 2, raw = FALSE)***

1. (Intercept)
2. X poly(X, Y, srtm, degree = 2, raw = FALSE)1.0.0
3. X^2 poly(X, Y, srtm, degree = 2, raw = FALSE)2.0.0
4. Y poly(X, Y, srtm, degree = 2, raw = FALSE)0.1.0
5. Y^2 poly(X, Y, srtm, degree = 2, raw = FALSE)0.2.0
6. srtm poly(X, Y, srtm, degree = 2, raw = FALSE)0.0.1
7. srtm^2 poly(X, Y, srtm, degree = 2, raw = FALSE)0.0.2
8. X:Y:srtm poly(X, Y, srtm, degree = 2, raw = FALSE)1.1.0
9. poly(X, Y, srtm, degree = 2, raw = FALSE)1.0.1
10. poly(X, Y, srtm, degree = 2, raw = FALSE)0.1.1
11. bio1 poly(bio1, degree = 2, raw = FALSE)1
12. bio1^2 poly(bio1, degree = 2, raw = FALSE)2

Error in models

Class 'try-error' atomic [1:1] Error in glm.fit(x = structure(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, :

NA/NaN/Inf in 'x'

..- attr(\*, "condition")=List of 2

.. ..$ message: chr "NA/NaN/Inf in 'x'"

.. ..$ call : language glm.fit(x = c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, | \_\_truncated\_\_ ...

.. ..- attr(\*, "class")= chr [1:3] "simpleError" "error" "condition"

Warnings after model fitting

Warning messages:

1: step size truncated due to divergence

2: glm.fit: algorithm did not converge

3: glm.fit: algorithm stopped at boundary value

4: glm.fit: fitted rates numerically 0 occurred

5: glm.fit: algorithm did not converge

6: glm.fit: fitted rates numerically 0 occurred

7: step size truncated due to divergence

8: glm.fit: fitted rates numerically 0 occurred

9: step size truncated due to divergence

10: glm.fit: algorithm did not converge