AMS 315
Professor Yan Yu
Gani Ates
May 9, 2024

Introduction

We are tasked with finding the multiple regression model that the TA used to generate our data file containing 1 dependent variable and 24 independent variables. The Caspi paper used the same type of model but made a Type I Error. They had three different variables in their model and all were strong associations (P = .02, P = .05, etc) but Bonferonni's correction $\alpha_E \leq m\alpha_i$ – that the sum of the significance tests is the total significance level of the test – shows that their TOTAL alpha was very high once you factor in all the p-values, leading to their demise. The aim is to find specifically G x E interactions, where E = environment and G = genetic variables, if any.

Methods

Data was read into RStudio. We start by analyzing the summary statistics of Y vs Environment variables, showing significance of all the environmental variables (P<.05). However this is not good enough. A quick look at residual plots of the raw linear model including all possible variables shows a heteroscedatic pattern. Box-Cox transformations are applied to fix this, at $\lambda = 94/99$. Stepwise regression t-testing is done in two steps. First is using 'leaps' package, next is using 'knitr' – this searches and helps us find our base model. The fifth step in searching is ideal as it comes 2 steps after a big leap in r-square and BIC without adding too much variance or total alpha. To confirm this model, we search and select the variables with main contributions to the model. This ends up perfectly aligning with our fifth-step model (after splitting E x E interaction): E1 + E2 + E3 + E4:G11+ G8:G19 + G7:G15. Before moving on we make sure this model has a low chance of error and that all contributions are significant enough. According to Bonferronni's condition, the total alpha in this stage is $\alpha = 24 * \Sigma p = .015$ – indicating a good attempt. Experimenting with other values of λ lead to similar results, and lead to the same variables being significant. However, a slightly lower lambda = .925 gives a lower p-value for the y-intercept.

Results

Final Model: $Y^{94/99} = \beta_0 + \beta_1 E_1 + \beta_2 E_2 + \beta_3 E_3 + \beta_4 E_4 G_{11} + \beta_5 G_8 G_{19} + \beta_6 G_7 G_{15} + \epsilon$

Table 1:

```
Call:
lm(formula = I(Y^powah) ~ E1 + E2 + E3 + E4:G11 + G8:G19 + G7:G15,
    data = Dat)

Residuals:
    Min      1Q  Median      3Q     Max
-81.098 -16.620  -0.171  16.174  84.408

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  28.6311     6.2922   4.550 5.83e-06 ***
E1            5.4571     0.4731  11.535  < 2e-16 ***
E2            5.1864     0.4770  10.874  < 2e-16 ***
E3            7.0002     0.4731  14.796  < 2e-16 ***
E4:G11        8.2393     0.1805  45.659  < 2e-16 ***
G8:G19        5.1459     1.4267   3.607 0.000321 ***
G7:G15        5.1071     1.4111   3.619 0.000306 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 25.4 on 1369 degrees of freedom
Multiple R-squared:  0.6531,    Adjusted R-squared:  0.6516
F-statistic: 429.6 on 6 and 1369 DF,  p-value: < 2.2e-16
```

Values for β_i are in Table 1 under section "Estimate", as well as the error. Significant at $\alpha = .015$. There is a 1.5% chance we made a Type I error.

Discussion

Chance of type I error is lowered by getting rid of G x G interactions – leads to a rather trivial adjusted r-square decrease of ~ 0.01. 99% Confidence Interval for y-intercept using n=30 (minimum n = 9 according to link): $\beta_0 \pm t\alpha/2, n-2 * se(\beta_0) = (28.6311 \pm (2.763)(6.2922) = (11.25, 46.02)$ with a risk ratio of 17.4. If we repeat our experiment, 99% of the time we should expect the y-intercept to fall in this CI. Smaller risk ratios are observed for variable coefficients. Our adjusted r-square is 0.6516 indicating a moderate fit to our data. There is no significant evidence for use of another type of model. About 35% of the data varies unpredictably, indicating the below average possibility of a specific configuration not discovered through our methods. We may be limited by our methods in the possible configuration of variables we discover. However unlike Caspi et al. our total chance of Type I error is low, and we can reject the null hypothesis at alpha = .015 that all coefficients in the model equal zero – accept H_1: at least one coefficient is non-zero. We can even perform individual tests to say all of the coefficients are non-zero – but that is for another day.

Code Appendix (with explanations)

```
install.packages('mice')
library(mice)
#Read in data

setwd('/Users/gania/Documents/dataset')
getwd();
Dat <- read.csv('P2_095768.csv', header=TRUE)

#Form multiple regression model and residual plot of Y vs Ei
M_E <- lm(Y ~ E1+E2+E3+E4, data=Dat)
summary(M_E)
M_raw <- lm(Y ~
(E1+E2+E3+E4+G1+G2+G3+G4+G5+G6+G7+G8+G9+G10+G11+G12+G13+G14+G15+G16+G17+G18
+G19+G20)^2, data=Dat)
plot(resid(M_raw) ~ fitted(M_raw), main='Residual Plot')

#Analyze regression with box-cox function
powah = 94/99
library(MASS)
b <- boxcox(M_raw)
lambda <- b$x[which.max(b$y)]
# Transform entire equation with lambda = .9494949 = 94/99 according to
Box-Cox
# Reference on extracting lambda:
https://r-coder.com/box-cox-transformation-r/#Extracting_the_exact_lambda
M_trans <- lm( I(Y^(powah)) ~
(E1+E2+E3+E4+G1+G2+G3+G4+G5+G6+G7+G8+G9+G10+G11+G12+G13+G14+G15+G16+G17+G18
+G19+G20)^2, data=Dat )
summary(M_raw)$adj.r.square
summary(M_trans)$adj.r.square

# New transformed residual plot
plot(resid(M_trans) ~ fitted(M_trans), main='New Residual Plot')

# Stepwise Regression of our model. We set nvmax to 24 when
# wanting to see the full list of possible interactions.
install.packages("leaps")
library(leaps)
M <- regsubsets( model.matrix(M_trans)[,-1], I((Dat$Y)^powah),
                 nbest = 1 , nvmax=5,
```

```
                  method = 'forward', intercept = TRUE )
temp <- summary(M)
# Now produce the model. We see the 3rd model has the highest jump in adj
r^2.
# includes E2 + E1:E3 + E4:G11
install.packages("knitr")
library(knitr)
Var <- colnames(model.matrix(M_trans))
M_select <- apply(temp$which, 1,
                  function(x) paste0(Var[x], collapse='+'))
kable(data.frame(cbind( model = M_select, adjR2 = temp$adjr2, BIC =
temp$bic)),
      caption='Model Summary')

#Account for main contributions in model: E2 + E1:E3 + E4:G11
# Split environment to get E1 + E2 + E3 + E4:G11
M_main <- lm( I(Y^powah) ~
E1+E2+E3+E4+G1+G2+G3+G4+G5+G6+G7+G8+G9+G10+G11+G12+G13+G14+G15+G16+G17+G18+
G19+G20, data=Dat)
temp1 <- summary(M_main)
kable(temp1$coefficients[ abs(temp1$coefficients[,4]) <= 0.001, ],
caption='Sig Coefficients')
# 2nd Stage = Checking whether main variables are significant and how so
M_2stage <- lm( I(Y^(powah)) ~ E1+E2+E3+E4:G11+G8:G19+G7:G15, data=Dat)
temp2 <- summary(M_2stage)
kable(temp2$coefficients[ abs(temp2$coefficients[,3]) >= 4, ])
#High adj R^2 of .6516 when using ~.95 BoxCox transformation with
E1+E2+E3+E4:G11+G8:G19+G7:G15
#Also trying other transformations: Note use of no transformation (^1) does
not
#change which variables and interactions end up being significant. However
#using a slightly lower exponent (.93 range) gives a lower p-value for the
y-intercept.
exp <- .925
M_experiment <- lm(I(Y^exp) ~ E1+E2+E3+E4:G11+G8:G19+G7:G15, data=Dat)
summary(M_experiment)
#(4) If results given by different transformations are similar (they are)
#you could try using all of the variables and combine terms
#selected from all of these models. Highest adj R^2 at .6635 achieved
below.
#However the Bonferonni inequality states the sum of alphas creates the
total alpha
#so if we multiply each p value by 24 we would be at alpha = 4.0!!! So this
```

```
#tells us less is better and this model is useless.
M_exp <- lm(I(Y^powah) ~
E1+E2+E3+E1:E3+E2:E4+E2:G11+E4:G11+G1:G7+G1:G16+G1:G17+G2:G14+G2:G20+G4:G10
+G4:G14+G5:G16+G7:G13+G7:G15+G8:G19+G10:G11+G11:G13, data=Dat)
summary(M_exp)
#Final model, sticking to what worked before.
M_att <- lm(I(Y^powah) ~ E1+E2+E3+E4:G11+G8:G19+G7:G15, data=Dat)
summary(M_att)
# Final model. Adj R^2 = .6516, F = 429.6, p-value < 2.2e-16. Significant
at
# alpha = 24 * sum of p values = .015. There is a 1.5% chance we made a
Type I error.
#
M_final <- lm(I(Y^powah) ~ E1 + E2 + E3 + E4:G11 + G8:G19 + G7:G15, data =
Dat)
summary(M_final)
#Self-note put risk ratio AKA width of confidence interval
predict(M_final, Dat, interval="confidence")
```

End of Report