Gene-Environment Interactions in Regression

Vincent Yabor
AMS 578

May 6, 2021

Abstract

The goal of this project is to determine a model that was used for a given dataset by using rigorous statistical methods. Further, I must clean the dataset of missing data without deletions or mean imputations. To solve these problems, various methods were implemented throughout this study such as the expectation-maximization algorithm, multiple linear regression, and stepwise selection. This paper begins with some general introduction into the context behind my study as well as some facts about my data. After, the paper discusses some theory behind my methods. Then I explain my methodologies behind building and analyzing the models. Finally, I provide the results and come to a conclusion on which model is best for the data.

1 Introduction

In the paper *Influence of Life Stress on Depression: Moderation by a Polymorphism in the 5-HTT Gene* by Caspi et al., the authors looked to model mental illness, namely depression, with genetics and stressful situations in a person's life [1]. The authors found gene by environment (GxE) interactions to be significant in their multiple regression models. Other researchers such as Risch et al. in the paper *Interaction Between the Serotonin Transporter Gene (5-HTTLPR), Stressful Life Events, and Risk of Depression* used larger datasets to assess Caspi et al.'s findings [2]. Risch et al. determined that Caspi et al. produced a type I error in their results. That is, no other researcher was able to find a well fitting model with GxE interactions. In particular, Risch et al. found that stressful events in one's life were associated with depression whereas there was no association found between the 5-HTTLPR gene and depression.

I was provided with three datasets. These are IDE, IDG, and IDY. IDE contains six continuous environmental variables, E_1 through E_6, as well as a patient identifier for all 2090 rows. IDG contains 25 binary variables R_1 through R_{25} as well as a patient identifier. Finally, IDY contains a continuous variable, Y, and a patient identifier. The patient identifiers were in random orders in the datasets. Of the 66,880 entries in the merged dataset, there were 470 missing values in 420 unique rows. In other words, I was missing 0.7% of the total data and 20% of the rows had at least one missing value. I used the programming language R to merge and clean the data, perform multiple regression and stepwise selection, and ultimately determine the best model for the data. Thus, giving me an equation relating the dependent variable Y to the independent variables.

2 Methods

2.1 Merging Datasets and Dealing with Missing Data

Since all three datasets contained a common ID variable, I used the *merge()* function in R, to combine the data based on common ID values and sort them by ID. Rather than deleting the NA values from my dataset, I explored some more sophisticated options. After careful consideration, I decided to use an R package called Amelia to impute my data. Amelia utilizes the

expectation-maximization (EM) algorithm with bootstrapping [3]. This algorithm obtains maximum likelihood estimates of parameters when there is missing data. The first step is to compute the expected value of the complete-data log likelihood. The next step is to maximize the expectation previously computed and update the missing value. These steps are then repeated until convergence [4]. Consider a dataset $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$ where only $\mathcal{X}$ is observed. It follows that the log likelihood is $l(\theta; \mathcal{X}, \mathcal{Y})$ where θ is an unknown parameter vector. The goal is to find the MLE of θ. The expectation portion of the algorithm is conducted as follows. First, define

$$Q(\theta; \theta_{old}) := E[l(\theta; \mathcal{X}, \mathcal{Y})|\mathcal{X}, \theta_{old}] = \int l(\theta; \mathcal{X}, y)p(y|\mathcal{X}, \theta_{old})dy$$

where p is the conditional density of $\mathcal{Y}$ given $\mathcal{X}$. Further, we must assume $\theta = \theta_{old}$. The maximization step involves maximizing the above expectation over θ.

$$\theta_{new} := \max_\theta Q(\theta; \theta_{old})$$

Then let $\theta_{old} = \theta_{new}$. These steps are then repeated until convergence [3]. The Amelia package in R does this process automatically and imputes the dataset with these new values.

2.2 Background on Selecting Variables

When testing my models, I paid close attention to adjusted R^2, Akaike information criterion (AIC), Bayesian information criterion (BIC), and statistically significant variables with p-value less than 0.001. It is not feasible to base variable selection on one criterion alone. Instead, one must make judgments based on many factors.

2.2.1 Adjusted R^2

The adjusted R^2 is an adjustment on R^2. It has the form

$$R^2_{adj} = 1 - (\tfrac{(1-R^2)(n-1)}{n-k-1})$$

where N is the number of points in the data sample, and k is the number of variables in the model. R^2 represents how close to the fitted regression line the data is. It increases with the number of significant variables in the model. The formula for R^2 is

$$1 - \frac{SS_{res}}{SS_{tot}}$$

where SS_{tot} is the total sum of squares and SS_{res} is the sum of squares due to residuals. The adjusted R^2 is a way to compare models which have differing numbers of variables. A general rule of thumb is that a model is better than a previous one if the adjusted R^2 increases.

2.2.2 AIC and BIC

AIC uses the maximum likelihood estimate as well as the number of parameters in a model to estimate how much information is lost. A lower value indicates less information being lost, which means a better model. AIC can be found with the formula:

$$AIC = -2L + 2K$$

where L is the log likelihood and K is the number of estimated parameters in the model.

BIC is quite similar to AIC. It introduces a penalty term for the number of parameters. It has the form

$$BIC = K \ln n - 2L$$

where K and L are the same as above while n is the number of data points. Like the AIC, a lower value for BIC means you have a better model. AIC and BIC are valuable criteria to compare models and reduce overfitting.

2.3 Building the Right Model

To get an idea of where to begin in terms of which variables may be relevant in the final model, I took a look at the correlation matrix for my imputed data. In particular, I observed the correlations of the dependent variable Y on the independent variables.

	Y
E1	0.0059
E2	0.4223
E3	0.0279
E4	0.03
E5	0.5967
E6	0.4787
R1	-0.0012
R2	0.026
R3	0.0201
R4	0.0125
R5	-0.0442
R6	6.00E-04
R7	-0.0148
R8	0.0115
R9	-0.0208
R10	-0.0111
R11	-0.0489
R12	-0.0239
R13	-0.0448
R14	-0.0012
R15	-0.0143
R16	-0.0104
R17	-0.0058
R18	0.0534
R19	0.0167
R20	0.0017
R21	-0.0107
R22	0.0082
R23	0.0452
R24	-0.0062
R25	-0.0269
Y	1

Table 1: Correlation Matrix

The only relatively high correlations between Y and any of the independent variables involved E_2, E_5, and E_6. Thus, I should expect these variables to show up in my final model. I also checked the correlations between all of the independent variables with each other and found they all had very low values, reducing the possibility of multicollinearity. Nevertheless, I decided to test out several models until I achieved one with a good fit. The best model will have a relatively higher R^2 value and lower AIC and BIC values.

I began with a linear model of Y on all genetic variables as well as the environmental variables up to the fourth power. The following model gives an adjusted R^2 value of 0.7491 with highly significant p-values, i.e. less than 0.001, on the variables E_2, E_5, and E_6.

```
ff1 <- Y~(R1+R2+R3+R4+R5+R6+R7+R8+R9+R10+R11+R12+R13+R14+R15+R16
          +R17+R18+R19+R20+R21+R22+R23+R24+R25)+poly(E1,4)+poly(E2,4)+
          poly(E3,4)+poly(E4,4)+poly(E5,4)+poly(E6,4)
fit1 <- lm(ff1,data)
```

This indicates that more attention should be paid to these variables than the others when fitting further models. Further, the AIC and BIC values are 97577.36 and 97865.26, respectively.

Next, I studied a model with quadratic terms. The following model involved the association between Y and the independent variables used in the previous model as well as the cross product between the independent variables.

```
ff2 <- Y~(poly(E1,4)+poly(E2,4)+poly(E3,4)+poly(E4,4)+poly(E5,4)+poly(E6,4)+R1
          +R2+R3+R4+R5+R6+R7+R8+R9+R10+R11+R12+R13+R14+R15+R16+R17
          +R18+R19+R20+R21+R22+R23+R24+R25)^2
fit2 <- lm(ff2,data)
```

This resulted in an adjusted R^2 of 0.7865 with statistically significant variables E_2, E_5, E_6, $E_1 * R_{20}$, $E_2 * E_5$, and $E_2 * E_6$. The AIC and BIC are 97809.9 and 104533. Comparatively, this model may or may not be as good as the one with only linear terms. Although the adjusted R^2 is higher in the second model, it has higher AIC and BIC values.

Ideally, next I would have fit a cubic and quartic model with all variables. This would give me all of the two, three, and four way interactions between the independent variables. The reason I could not do this is because it would have produced more predictors than observations. In that case, a valid model cannot be possibly produced. Instead, I shifted my focus towards the more relevant variables from my study thus far. Namely, E_2, E_5, E_6, E_1, and R_{20}.

I made a dataset consisting of only the above variables, and Y on which I would perform stepwise selection. This algorithm is based on the AIC and involves having variables enter and leave a model based on AIC until convergence. The model I perform stepwise selection on is the following.

```
fit3 <- lm(Y~(R20+E1+E2+E5+E6)^4,data=data.fit)
stp <- step(fit3,data=data.fit,direction = 'both')
```

That is, all two, three, and four way interactions between the five independent variables will be considered. The algorithm determined the best model, shown below. Note that the adjusted R^2 for *fit3* is 0.7811 with AIC and BIC of 97273.41 and 97454.05.

$$\begin{gathered} Y = R_{20} + E_1 + E_2 + E_5 + E_6 + R_{20} * E_1 + R_{20} * E_2 \\ + R_{20} * E_5 + E_1 * E_2 + E_2 * E_5 + E_2 * E_6 + E_5 * E_6 \\ + R_{20} * E_1 * E_2 + R_{20} * E_2 * E_5 + E_2 * E_5 * E_6 \end{gathered}$$

This model had an adjusted R^2 of 0.7821 with AIC and BIC of 97248.84 and 97344.80. This is a large improvement in AIC and BIC compared to *fit2* and a small improvement compared to *fit3*.

I continued by removing the variables which have p-values greater than 0.001 from the model. This left me with just $E_2 * E_5 * E_6$. Fitting that to a model against Y is shown below.

```
fit4 <- lm(Y~E2:E5:E6,data=data.fit)
```

The adjusted R^2, AIC, and BIC, are 0.7815, 97237.32, and 97254.26, respectively. There is not enough of a difference between these values for this model and the previous one to draw a conclusion about which is better. The model, *fit4*, is a simpler model. Thus, it is the model I will consider

best. It should be noted that the intercept for this model has a p-value of 0.929. Rather than eliminating the intercept, I ran stepwise selection on this model to see if I should drop the intercept or leave it in. The resulting best model was the one that included an intercept.

3 Results

The final model for gene-environment interactions in depression is

$$Y = \alpha + \beta * E_2 * E_5 * E_6$$

where $\alpha = 1.616 * 10^7$ and $\beta = 37.07$. Further, it can be seen that the regression assumptions are all met for this model. The Residuals vs Fitted plot

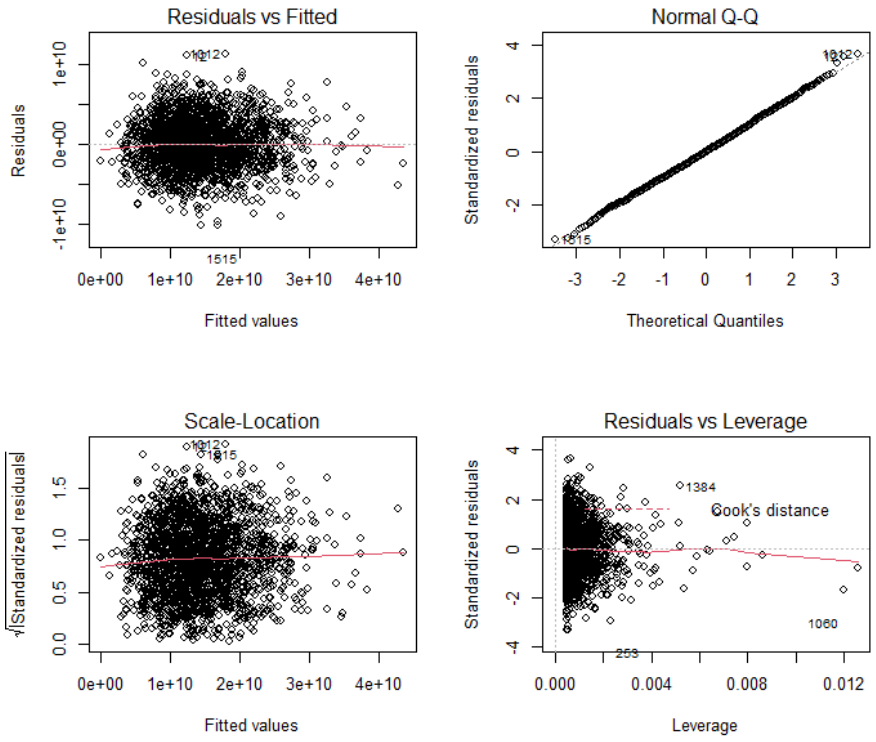

Figure 1: Regression Plots

indicates that there is no clear nonlinear relation between the residuals and

fitted values. Further, the Normal Q-Q plot shows that the residuals follow a normal distribution. The Scale-Location plot shows equal variance among the residuals, called homoscedasticity. Finally, the Residuals vs Leverage plot indicates there are no significant influential cases or outliers in the model.

The ANOVA table of the final model is the following.

	Estimate	Std. Error	t value	Pr
(Intercept)	16164950.98	180219241.6	0.089696033	0.928537371
E2:E5:E6	37.07377125	0.42889022	86.44116736	0

Table 2: ANOVA Table for Final Model

4 Conclusions and Discussion

The best model to represent the data shows that the only variables which have a statistically significant impact on Y are environmental variables, namely $E_2 * E_5 * E_6$. There were other possible candidates for good models since the adjusted R^2, AIC, and BIC did not seem to differ by a lot between some of the models I fit. It is important to investigate several aspects of a model before coming to a conclusion. From the beginning, I suspected E_2, E_5, and E_6 would be included in my final model due to the high correlation with Y. The R^2, AIC, BIC, p-values, and stepwise selection all seemed to back this claim.

References

[1] A. Caspi. "Influence of Life Stress on Depression: Moderation by a Polymorphism in the 5-HTT Gene". In: *Science* 301.5631 (2003), pp. 386–389. DOI: 10.1126/science.1083968.

[2] Neil Risch et al. "Interaction Between the Serotonin Transporter Gene (5-HTTLPR), Stressful Life Events, and Risk of Depression". In: *JAMA* 301.23 (2009), pp. 2462–2471. DOI: 10.1001/jama.2009.878.

[3] Martin Haugh. *The EM Algorithm*. 2015. URL: http://www.columbia.edu/~mh2078/MachineLearningORFE/EM_Algorithm.pdf.

[4] Matthew Blackwell, Gary King, and James Honaker. *Package 'Amelia'*. Nov. 2019. URL: https://cran.r-project.org/web/packages/Amelia/Amelia.pdf.

Appendix: R Code

```r
set.seed(123)
# Reading in the data
ide <- read.csv('IDEgroup756962.csv',header=T)[,-1]
idg <- read.csv('IDGgroup756962.csv',header=T)[,-1]
idy <- read.csv('IDYgroup756962.csv',header=T)[,-1]
```

```r
# Merging by ID
data1 <- merge(ide,idg,by='ID')
dat <- merge(data1,idy,by='ID')
```

```r
# Summary stats and correlation matrix (excluding ID)
summary(dat)[,-1]
apply(dat,2,length)
apply(dat,2,sd,na.rm=T)[-1]

cor(na.omit(dat[,-1]))
```

```r
# Dealing with missing data
library(Amelia)

## Loading required package:  Rcpp
## ##
## ## Amelia II: Multiple Imputation
## ## (Version 1.7.6, built:  2019-11-24)
## ## Copyright (C) 2005-2021 James Honaker, Gary King and Matthew
Blackwell
## ## Refer to http://gking.harvard.edu/amelia/ for more information
## ##

a.out <- amelia(dat,noms=c('R1','R2','R3','R4','R5','R6','R7','R8','R9','R10'
                           ,'R11','R12','R13','R14','R15','R16','R17'
                           ,'R18','R19','R20','R21','R22','R23','R24'
                           ,'R25'),idvars = 'ID',m=1)

## -- Imputation 1 --
##
##   1  2  3  4
```

```
summary(a.out)
a.out$imputations$imp1
```

```
data <- a.out$imputations$imp1[,-1]
```

```
# summary stats and correlation matrix for the imputation model (excluding ID)
summary(a.out$imputations$imp1)[,-1]
cor(a.out$imputations$imp1)[-1,-1]
apply(a.out$imputations$imp1,2,sd)[-1]
```

```
corr <- round(cor(data$Y,data[,]),4)

ff1 <- Y~(R1+R2+R3+R4+R5+R6+R7+R8+R9+R10+R11+R12+R13+R14+R15+R16
          +R17+R18+R19+R20+R21+R22+R23+R24+R25)+poly(E1,4)+poly(E2,4)+
          poly(E3,4)+poly(E4,4)+poly(E5,4)+poly(E6,4)
ff2 <- Y~(poly(E1,4)+poly(E2,4)+poly(E3,4)+poly(E4,4)+poly(E5,4)+poly(E6,4)
          +R1+R2+R3+R4+R5+R6+R7+R8+R9+R10+R11+R12+R13+R14+R15+R16+R17
          +R18+R19+R20+R21+R22+R23+R24+R25)^2
```

```
fit1 <- lm(ff1,data)
summary(fit1)$coefficients[,4][summary(fit1)$coefficients[,4]<0.001]

##   (Intercept)  poly(E2, 4)1  poly(E5, 4)1  poly(E6, 4)1
## 2.028237e-253 4.155715e-226  0.000000e+00 1.076088e-289

summary(fit1)$adj.r.squared

## [1] 0.7491001

# E2+E5+E6
```

```
fit2 <- lm(ff2,data)
summary(fit2)$coefficients[,4][summary(fit2)$coefficients[,4]<0.001]

##               (Intercept)              poly(E2, 4)1              poly(E5, 4)1
##              7.114434e-12              9.690286e-07              4.461753e-07
##          poly(E1, 4)1:R20 poly(E2, 4)1:poly(E5, 4)1 poly(E2, 4)1:poly(E6, 4)1
##              4.842716e-04              6.737660e-04              1.153177e-05

summary(fit2)$adj.r.squared

## [1] 0.7864832

# E2 + E5 + E6 + E1:R20 E2:E5 + E2:E6

data.fit <- data.frame(R20=data$R20,E1=data$E1,E2=data$E2,
                       E5=data$E5,E6=data$E6,Y=data$Y)

fit3 <- lm(Y~(R20+E1+E2+E5+E6)^4,data=data.fit)
summary(fit3)$adj.r.squared

## [1] 0.7811182
```

```
stp <- step(fit3,data=data.fit,direction = 'both',trace=0)
```

```
summary(stp)
# R20 + E1 + E2 + E5 + E6 + R20:E1 + R20:E2
# + R20:E5 + E1:E2 + E2:E5 + E2:E6 + E5:E6
# + R20:E1:E2 + R20:E2:E5 + E2:E5:E6
```

```
summary(stp)$coefficients[,4][summary(stp)$coefficients[,4]<0.001]

##     E2:E5:E6
## 0.0003659455

# E2:E5:E6
summary(stp)$adj.r.squared
```

```
## [1] 0.7821361

fit4 <- lm(Y~E2:E5:E6,data=data.fit)
summary(fit4)$coefficients

##                 Estimate   Std. Error     t value  Pr(>|t|)
## (Intercept) 1.616495e+07 1.802192e+08  0.08969603 0.9285374
## E2:E5:E6    3.707377e+01 4.288902e-01 86.44116736 0.0000000

summary(fit4)$adj.r.squared

## [1] 0.7814871

a1 <- AIC(fit1)
b1 <- BIC(fit1)

a2 <- AIC(fit2)
b2 <- BIC(fit2)

a3 <- AIC(fit3)
b3 <- BIC(fit3)

a4 <- AIC(fit4)
b4 <- BIC(fit4)

data.frame(c(a1,b1),c(a2,b2),c(a3,b3),c(a4,b4))

##   c.a1..b1. c.a2..b2. c.a3..b3. c.a4..b4.
## 1  97577.36   97809.9  97273.41  97241.11
## 2  97865.26  104533.0  97454.05  97258.05
```

```
par(mfrow=c(2,2))
plot(fit5)
```