High-dimensional Empirical Analysis: Whether liking classical music

Zhou Zhou 15220182202882

2021/5/16

Abstract

There are probably more variables than we can imagine to be justifiable to affect people's love of **classical music-symphony & chamber** on the social network or in realistic life.

Based on many variables from GSS(U.S. General Social Survey) in 1993, I'll try to predict whether a person love classical music-symphony & chamber by performing regularized linear with a regularizer that combines L1 and L2 (Elastic Net) in this article.

Background

To perform least squares linear regression, we use the model:

$$f_{\hat{\theta}}(x) = \hat{\theta} \cdot x$$

We fit the model by minimizing the mean squared error cost function:

$$L(\hat{\theta}, X, y) = \frac{1}{n} \sum_{i}^{n} (y_i - f_{\hat{\theta}}(X_i))^2$$

In the above definitions, X represents the $n \times p$ data matrix, x represents a row of X, y represents the observed outcomes, and $\hat{\theta}$ represents the model weights.

L2 Regularization: Ridge Regression

L_2 regularization, a method of penalizing large weights in our cost function to lower model variance.

To add L_2 regularization to the model, we modify the cost function above:

$$L(\hat{\theta}, X, y) = \frac{1}{n} \sum_{i}^{n} (y_i - f_{\hat{\theta}}(X_i))^2 + \lambda \sum_{j=1}^{p} \hat{\theta}_j^2$$

Notice that the cost function above is the same as before with the addition of the L_2 regularization $\lambda \sum_{j=1}^{p} \hat{\theta}_j^2$ term. The summation in this term sums the square of each model weight $\hat{\theta}_1, \hat{\theta}_2, ..., \hat{\theta}_p$. The term also introduces a new scalar model parameter λ that adjusts the regularization penalty.

The regularization term causes the cost to increase if the values in $\hat{\theta}$ are further away from 0. With the addition of regularization, the optimal model weights minimize the combination of loss and regularization penalty rather than the loss alone. Since the resulting model weights tend to be smaller in absolute value, the model has lower variance and higher bias.

Using L_2 regularization with a linear model and the mean squared error cost function is also known more commonly as **ridge regression**.

Using L_2 regularization allows us to tune model bias and variance by penalizing large model weights. L_2 regularization for least squares linear regression is also known by the more common name ridge regression. Using regularization adds an additional model parameter λ that we adjust using cross-validation.

L1 Regularization: Lasso Regression

L_1 regularization, another regularization technique that is useful for feature selection.

We fit the model by minimizing the mean squared error cost function with an additional regularization term:

$$L(\hat{\theta}, X, y) = \frac{1}{n} \sum_{i}^{n} (y_i - f_{\hat{\theta}}(X_i))^2 + \lambda \sum_{j=1}^{p} \hat{\theta}_j^2$$

In the above definitions, X represents the $n \times p$ data matrix, x represents a row of X, y represents the observed outcomes, and $\hat{\theta}$ represents the model weights, and λ represents the regularization parameter.

To add L_1 regularization to the model, we modify the cost function above:

$$L(\hat{\theta}, X, y) = \frac{1}{n} \sum_{i}^{n} (y_i - f_{\hat{\theta}}(X_i))^2 + \lambda \sum_{j=1}^{p} |\hat{\theta}_j|$$

Observe that the two cost functions only differ in their regularization term. L_1 regularization penalizes the sum of the absolute weight values instead of the sum of squared values.

Using L_1 regularization with a linear model and the mean squared error cost function is also known more commonly as **lasso regression**. (Lasso stands for Least Absolute Shrinkage and Selection Operator.)

Lasso regression performs *feature selection*—it discards a subset of the original features when fitting model parameters. This is particularly useful when working with high-dimensional data with many features. A model that only uses a few features to make a prediction will run much faster than a model that requires many calculations. Since unneeded features tend to increase model variance without decreasing bias, we can sometimes increase the accuracy of other models by using lasso regression to select a subset of features to use.

Using L_1 regularization, like L_2 regularization, allows us to tune model bias and variance by penalizing large model weights. L_1 regularization for least squares linear regression is also known by the more common name lasso regression. Lasso regression may also be used to perform feature selection since it discards insignificant features.

Lasso vs. Ridge In Practice

If our goal is merely to achieve the highest prediction accuracy, we can try both types of regularization and use cross-validation to select between the two types.

Sometimes we prefer one type of regularization over the other because it maps more closely to the domain we are working with. For example, if know that the phenomenon we are trying to model results from many small factors, we might prefer ridge regression because it won't discard these factors. On the other hand, some outcomes result from a few highly influential features. We prefer lasso regression in these situations because it will discard unneeded features.

Elastic Net

Like ridge and lasso, we again attempt to minimize the residual sum of squares plus some penalty term:

$$L(\hat{\theta}, X, y) = \frac{1}{n}\sum_{i}^{n}(y_i - f_{\hat{\theta}}(X_i))^2 + \lambda[(1-\alpha)\frac{||\beta||_2^2}{2} + \alpha||\beta||_1]$$

Here, $||\beta||_1$ is called the L_1 norm.

$$||\beta||_1 = \sum_{j=1}^{p}|\hat{\theta}_j|$$

Similarly, $||\beta||_2$ is called the L_2, or Euclidean norm.

$$||\beta||_2 = \sqrt{\sum_{j=1}^{p}\hat{\theta}_j^2}$$

These both quantify how "large" the coefficients are. Like lasso and ridge, the intercept is not penalized and can use glment takes care of standardization internally. Also reported coefficients are on the original scale.

The new penalty is $\frac{\lambda\cdot(1-\alpha)}{2}$ times the ridge penalty plus $ $ times the lasso lasso penalty. (Dividing the ridge penalty by 2 is a mathematical convenience for optimization.) Essentially, with the correct choice of λ and α these two "penalty coefficients" can be any positive numbers.

Often it is more useful to simply think of α as controlling the mixing between the two penalties and λ controlling the amount of penalization. α takes values between 0 and 1. Using $\alpha = 1$ gives the lasso that we have seen before. Similarly, $\alpha = 0$ gives ridge.

Empirical Analysis

Data Description

Questions from GSS(U.S. General Social Survey) associated with liking classical music:

I'm going to read you a list of some types of music. Can you tell me which of the statements on this card comes closest to your feeling about each type of music (HAND CARD CA TO RESPONDENT.) Let's start with big band music. Do you like it very much(1), like it(2), have mixed feelings(3), dislike it(4), dislike it very much(5), or is this a type of music that you don't know much about?

F. Classical music-symphony & chamber

To capture what factors effect people's preference to classical music most, I choose 62 variables that might correlate with it from GSS and run both Lasso and elastic net method to regularize the regression model. The variables are listed below:

Variable name	Variable Label
popular	To be well liked or popular
workhard	To work hard

Variable name	Variable Label
hrs2	Number of hours usually work a week
drunk	Ever drink too much?
smoke	Does r smoke
grass	Should marijuana be made legal
pawrkslf	Father self-emp. or worked for somebody
jobsec	No danger of being fired
postlife	Belief in life after death
sibs	Number of brothers and sisters
childs	Number of children
age	Age of respondent
degree	Rs highest degree
padeg	Fathers highest degree
madeg	Mothers highest degree
spdeg	Spouses highest degree
mapa	Contrast between mother and father
sex	Respondents sex
race	Race of respondent
absingle	Not married
hompop	Number of persons in household
income	Total family income
rincome	Respondents income
bigband	Like or dislike bigband music
blugrass	Like or dislike bluegrass music
country	Like or dislike country western music
blues	Like or dislike blues or r and b music
musicals	Like or dislike broadway musicals
classicl	Like or dislike classical music
income91	Total family income
folk	Like or dislike folk music
gospel	Like or dislike gospel music
jazz	Like or dislike jazz
latin	Like or dislike latin music
moodeasy	Like or dislike easy listening music
newage	Like or dislike new age music
premarsx	Sex before marriage
opera	Like or dislike opera
anrights	Animals have rights too
rap	Like or dislike rap music
reggae	Like or dislike reggae music
conrock	Like or dislike contemporary rock music
oldies	Like or dislike oldies rock music
hvymetal	Like or dislike heavy metal music
tvshows	How often r watches tv drama or sitcoms
hitage	Beaten as child or adult
excelart	Artistic excellence found in folk art
fear	Afraid to walk at night in neighborhood
owngun	Have gun in home
happy	General happiness
hapmar	Happiness of marriage
health	Condition of health
natenvir	Improving & protecting environment
life	Is life exciting or dull

Variable name	Variable Label
natheal	Improving & protecting nations health
natfare	Welfare
confinan	Confid in banks & financial institutions
natsoc	Social security
sexfreq	Frequency of sex during last year
coneduc	Confidence in education
confed	Confid. in exec branch of fed govt
conpress	Confidence in press

Note that all the data are limited in year of 1993.

Preparation of Dataset

```
getwd()
```

```
## [1] "C:/Users/84057/Desktop"
```

```
library(foreign)
library(AER)
```

```
## Loading required package: car
```

```
## Loading required package: carData
```

```
## Loading required package: lmtest
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: sandwich
```

```
## Loading required package: survival
```

```
library(haven)
GSS <- read_dta("C:/Users/84057/Desktop/GSS1993.dta")
attach(GSS)
dataset <- data.frame(popular, workhard, hrs2, drunk, smoke, grass, pawrkslf, jobsec, postlife, sibs, cl
detach(GSS)

library(glmnet)
```

```
## Loading required package: Matrix

## Loaded glmnet 4.1-1
```

```
summary(dataset$classicl)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##    1.00    2.00    3.00    2.67    4.00    5.00      77
```

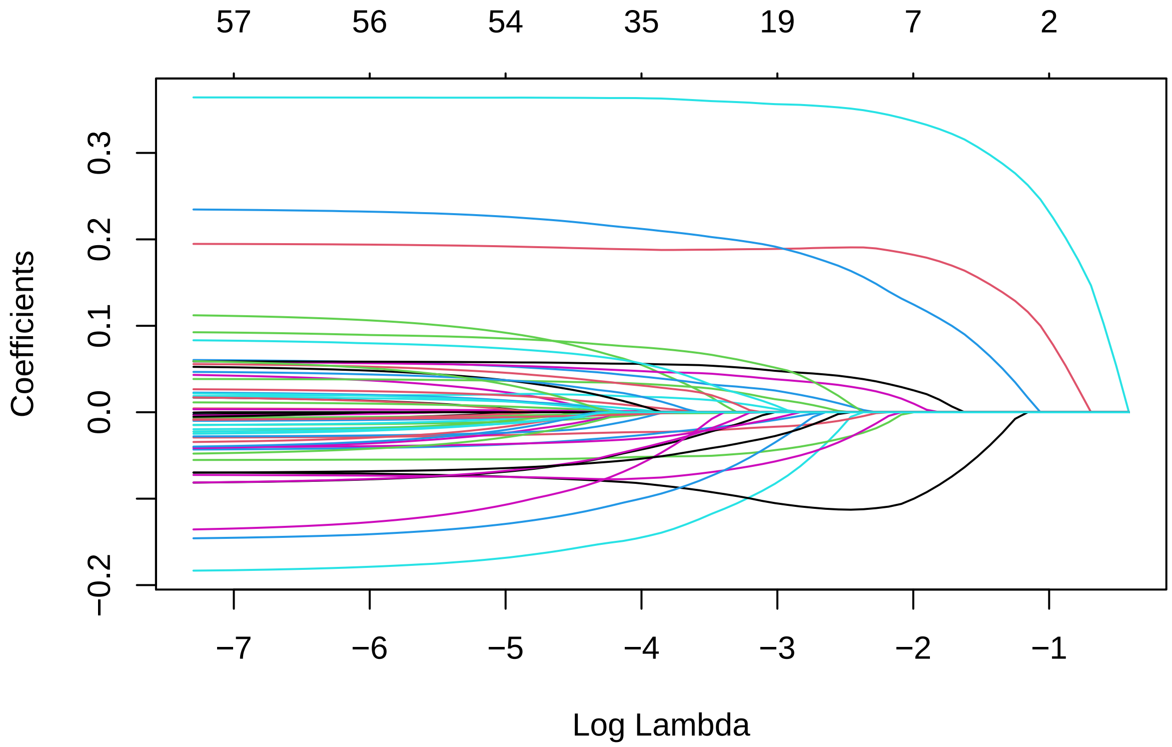
Data Processing

```
dataset <- replace(dataset, TRUE, lapply(dataset, na.aggregate))
logit<-glm(classicl~.,dataset,family='gaussian',control=list(maxit=100))
coeftest(logit)
```

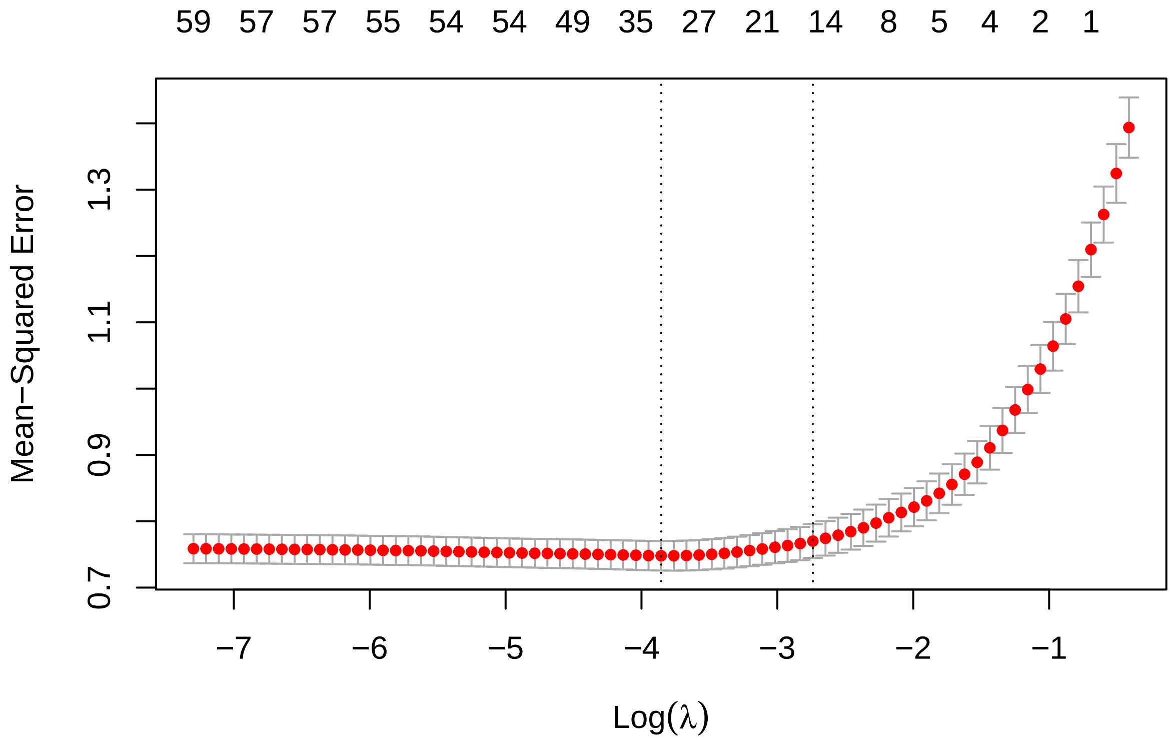
```
## 
## z test of coefficients:
## 
##                Estimate  Std. Error z value  Pr(>|z|)    
## (Intercept)  1.32409939  0.76899398  1.7219 0.0850950 .  
## popular     -0.00872954  0.03629334 -0.2405 0.8099216    
## workhard     0.00471406  0.02892670  0.1630 0.8705455    
## hrs2        -0.01499614  0.01488584 -1.0074 0.3137378    
## drunk       -0.00331427  0.07030418 -0.0471 0.9624002    
## smoke       -0.18467350  0.06309630 -2.9269 0.0034241 ** 
## grass        0.04629511  0.06837624  0.6771 0.4983651    
## pawrkslf     0.01833852  0.05227764  0.3508 0.7257452    
## jobsec      -0.00941169  0.02243727 -0.4195 0.6748751    
## postlife     0.11457480  0.07257145  1.5788 0.1143851    
## sibs        -0.01035434  0.00801618 -1.2917 0.1964678    
## childs       0.01853615  0.01624720  1.1409 0.2539187    
## age          0.00351822  0.00215439  1.6330 0.1024591    
## degree      -0.06901757  0.02480874 -2.7820 0.0054027 ** 
## padeg       -0.02938070  0.02473266 -1.1879 0.2348605    
## madeg       -0.05556892  0.02941625 -1.8891 0.0588844 .  
## spdeg       -0.04324195  0.02915862 -1.4830 0.1380770    
## mapa         0.01831558  0.01719068  1.0654 0.2866783    
## sex          0.00068924  0.04959430  0.0139 0.9889117    
## race        -0.04366907  0.04711509 -0.9269 0.3539994    
## absingle    -0.00319659  0.05779375 -0.0553 0.9558913    
## hompop       0.01919504  0.02087246  0.9196 0.3577635    
## income       0.00222896  0.02188989  0.1018 0.9188949    
## rincome      0.01197476  0.00999314  1.1983 0.2308006    
## bigband      0.06135840  0.02682322  2.2875 0.0221660 *  
## blugrass    -0.02046897  0.02829177 -0.7235 0.4693755    
## country     -0.07219658  0.02390594 -3.0200 0.0025275 ** 
## blues       -0.00412144  0.02688441 -0.1533 0.8781599    
## musicals     0.19498842  0.02703837  7.2115 5.532e-13 ***
## income91    -0.00756424  0.01202927 -0.6288 0.5294670    
## folk         0.23583031  0.02641615  8.9275 < 2.2e-16 ***
## gospel      -0.01564650  0.02367628 -0.6609 0.5087077    
## jazz         0.05985737  0.02619104  2.2854 0.0222886 *
## latin        0.05902998  0.02600806  2.2697 0.0232270 *
## moodeasy    -0.00966430  0.02538746 -0.3807 0.7034464
## newage       0.03866255  0.02479296  1.5594 0.1188979
## premarsx     0.02351079  0.02403591  0.9782 0.3279988
## opera        0.36427556  0.02348522 15.5108 < 2.2e-16 ***
## anrights    -0.00443954  0.02112522 -0.2102 0.8335478
## rap         -0.08292261  0.02426088 -3.4180 0.0006309 ***
## reggae       0.02760982  0.02525677  1.0932 0.2743212
## conrock     -0.02398601  0.02652749 -0.9042 0.3658923
## oldies      -0.02823783  0.02621371 -1.0772 0.2813837
## hvymetal     0.02293318  0.02423517  0.9463 0.3440074
## tvshows     -0.04083748  0.01903011 -2.1459 0.0318778 *
## hitage       0.05400701  0.05837687  0.9251 0.3548910
## excelart    -0.03634363  0.05395033 -0.6736 0.5005339
## fear        -0.04997586  0.05622823 -0.8888 0.3741086
## owngun      -0.14791347  0.05508116 -2.6854 0.0072449 **
## happy        0.08470740  0.04019938  2.1072 0.0351018 *
## hapmar      -0.13894516  0.05789053 -2.4001 0.0163890 *
## health      -0.00716520  0.03472548 -0.2063 0.8365265
## natenvir     0.05686686  0.05005517  1.1361 0.2559216
## life         0.09350263  0.04866448  1.9214 0.0546846 .
## natheal      0.04749352  0.05341951  0.8891 0.3739670
## natfare     -0.02623870  0.04256334 -0.6165 0.5375894
## confinan    -0.08283345  0.04529367 -1.8288 0.0674283 .
## natsoc      -0.07035513  0.03843385 -1.8306 0.0671675 .
## sexfreq     -0.00849623  0.01397173 -0.6081 0.5431201
## coneduc      0.06217979  0.04554583  1.3652 0.1721860
## confed      -0.04196739  0.04631819 -0.9061 0.3649004
## conpress    -0.02326697  0.04480942 -0.5192 0.6035915
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Then I regularize it with Lasso approach:

```
# Regularized by LASSO
x<-model.matrix(classicl~.,dataset)
y<-dataset$classicl
lassofit.all<-glmnet(x,y,alpha=1,family="gaussian")
plot(lassofit.all,xvar="lambda")
```



```
# Cross Validation
cv.lasso <- cv.glmnet(x,y,alpha=1,family="gaussian")
plot(cv.lasso)
```

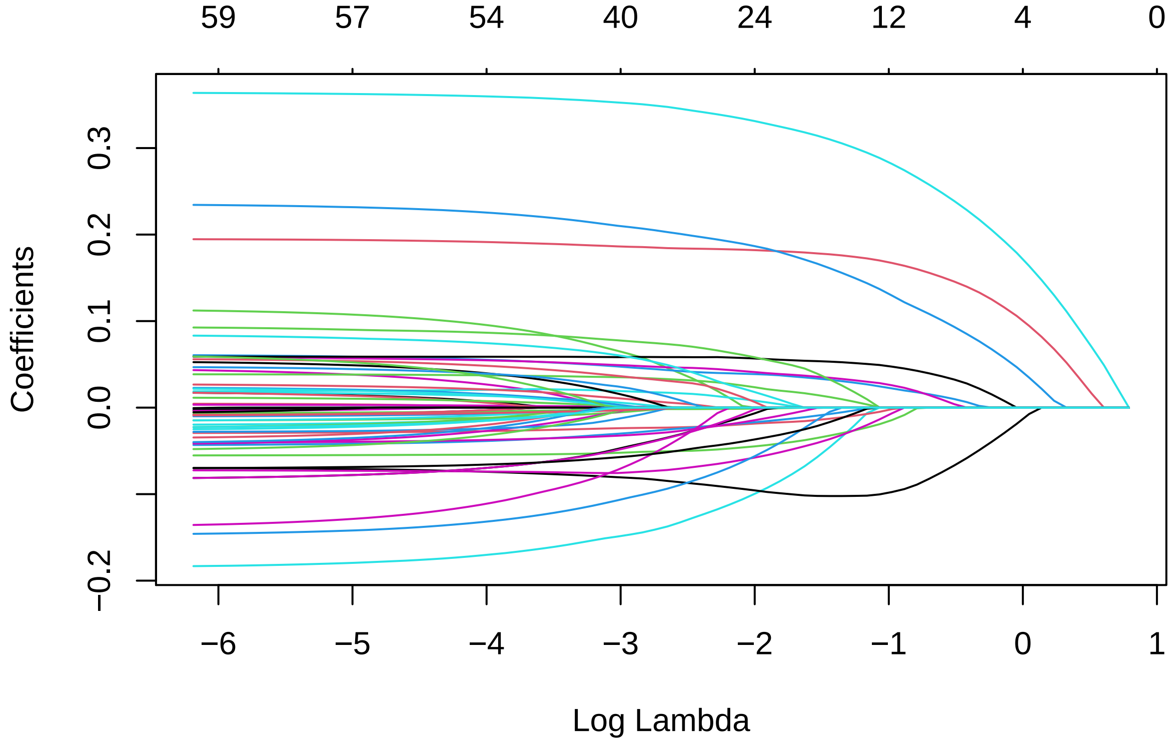


```
lambda.star <- cv.lasso$lambda.min
lassofit.star <- glmnet(x,y,alpha=1,lambda=lambda.star,family="gaussian")
coef(lassofit.star)
```

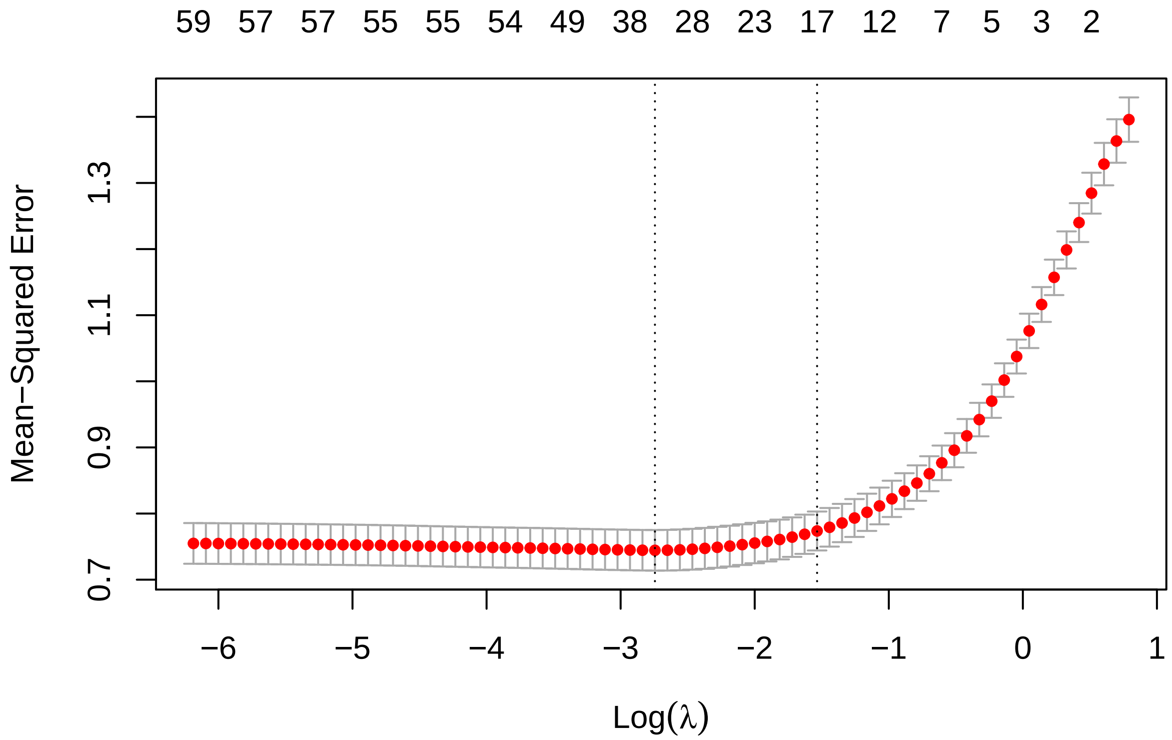
```
## 63 x 1 sparse Matrix of class "dgCMatrix"
##                        s0
## (Intercept)  0.7160758553
## (Intercept)  .
## popular      .
## workhard     .
## hrs2        -0.0008321552
## drunk        .
## smoke       -0.1395656077
## grass        .
## pawrkslf     .
## jobsec       .
## postlife     0.0448863897
## sibs         .
## childs       0.0172230655
## age          0.0007408586
## degree      -0.0849067695
## padeg       -0.0227471455
## madeg       -0.0512958011
## spdeg       -0.0240018235
## mapa         0.0002057680
## sex          .
## race         .
## absingle     .
## hompop       .
## income       .
## rincome      .
## bigband      0.0389441483
## blugrass     .
## country     -0.0754814484
## blues        .
## musicals     0.1877412929
## income91    -0.0011804588
## folk         0.2096515251
## gospel       .
## jazz         0.0463443811
## latin        0.0553574926
## moodeasy     .
## newage       0.0316525786
## premarsx     .
## opera        0.3629254304
## anrights     .
## rap         -0.0380297507
## reggae       0.0045459700
## conrock      .
## oldies      -0.0008675874
## hvymetal     .
## tvshows     -0.0285132375
## hitage       0.0003007528
## excelart     .
## fear         .
## owngun      -0.0941728301
## happy        0.0512144342
## hapmar      -0.0469302236
## health       .
## natenvir     0.0283530205
## life         0.0732825962
## natheal      0.0123706216
## natfare      .
## confinan    -0.0357111370
## natsoc      -0.0509539773
## sexfreq     -0.0012527560
## coneduc      .
## confed       .
## conpress     .
```

Furthermore, I regularize it via elatic net approach:

```r
# Regularized by Elastic Net
x<-model.matrix(classicl~.,dataset)
y<-dataset$classicl
elastic<-glmnet(x,y,alpha=0.3,family="gaussian")
plot(elastic,xvar="lambda")
```



```
# Cross Validation
cv.elastic<-cv.glmnet(x,y,alpha=0.3,family="gaussian")
plot(cv.elastic)
```



```
lambda.star_elastic<-cv.elastic$lambda.min
elastic.star<-glmnet(x,y,alpha=0.3,lambda=lambda.star_elastic,family="gaussian")
coef(elastic.star)
```

```
## 63 x 1 sparse Matrix of class "dgCMatrix"
##                         s0
## (Intercept)   0.7981393237
## (Intercept)   .
## popular       .
## workhard      .
## hrs2         -0.0021849814
## drunk         .
## smoke        -0.1409582692
## grass         .
## pawrkslf      .
## jobsec       -0.0010953337
## postlife      0.0520433433
## sibs          .
## childs        0.0176428486
## age           0.0006345212
## degree       -0.0832536287
## padeg        -0.0232451875
## madeg        -0.0508195372
## spdeg        -0.0267119807
## mapa          0.0020356986
## sex           .
## race         .
## absingle     .
## hompop       .
## income       .
## rincome      .
## bigband      0.0444093980
## blugrass     .
## country     -0.0730687390
## blues        .
## musicals     0.1849983433
## income91    -0.0016893079
## folk         0.2049862971
## gospel       .
## jazz         0.0473080314
## latin        0.0584799477
## moodeasy     .
## newage       0.0336206842
## premarsx     .
## opera        0.3491636771
## anrights     .
## rap         -0.0381120622
## reggae       0.0072180957
## conrock      .
## oldies      -0.0043001544
## hvymetal     .
## tvshows     -0.0297355880
## hitage       0.0050870193
## excelart     .
## fear         .
## owngun      -0.0973201765
## happy        0.0526762598
## hapmar      -0.0519717071
## health       .
## natenvir     0.0324032598
## life         0.0745287490
## natheal      0.0162263586
## natfare      .
## confinan    -0.0388144926
## natsoc      -0.0532384181
## sexfreq     -0.0015701232
## coneduc      .
## confed       .
## conpress     .
```

Conclusion

From the above results, we can use these two methods to explain the person who loves classical music more: the less smoking, the higher education, the more fond of folk, opera and rap, and the person who opposes holding guns. In this process, the income of the respondents and their families is not significant, which is different from our previous expectations.

Reference

[1] Model Selection and Regularization, Jiaming Mao

[2] GSS Data Explorer