MARS Assignment

Dheeraj Gadhiraju

Student ID: 7325590

STAT 451/551

April 21st, 2015

I have taken the final data from the midterm project.

The following data consists of 10 predictor variables, some of which were left alone and some of which were changed into factor variables.

> loandat <- read.csv(file.choose())

> assign=runif(length(loandat[,1]))

> group=assign\*NA

> group=ifelse(assign>.6,1,2)

> table(group)

group

1 2

16854 25652

> loandat=cbind(loandat,group)

> train=subset(loandat,group==2)

> validate=subset(loandat,group==1)

> earthmod <- earth(Response~.-Loan\_ID-Member\_ID, data=train,glm=list(family=binomial),degree=1)

> summary(earthmod, style='max')

Call: earth(formula=Response~.-Loan\_ID-Member\_ID, data=train,

glm=list(family=binomial), degree=1)

GLM Response =

-2.086553

+ 0.6312227 \* GradeB

+ 0.9092047 \* GradeC

+ 1.213706 \* GradeD

+ 1.383563 \* GradeE

+ 1.689658 \* GradeF

+ 1.850817 \* GradeG

+ 0.7993166 \* IncFactLow

+ 0.3799913 \* IncFactMed

+ 1.633426e-05 \* max(0, Loan\_Amnt - 6000)

- 0.002455945 \* max(0, FICO\_Low - 675)

- 0.1349297 \* max(0, 8 - Inq\_6mnths)

Earth selected 12 of 15 terms, and 11 of 20 predictors

Termination condition: RSq changed by less than 0.001 at 15 terms

Importance: Inq\_6mnths, IncFactLow, FICO\_Low, Loan\_Amnt, GradeF, GradeC, GradeD, GradeE, GradeG, IncFactMed, GradeB, DTI-unused, ...

Number of terms at each degree of interaction: 1 11 (additive model)

Earth GCV 0.1197019 RSS 3065.09 GRSq 0.0519175 RSq 0.05354308

GLM null.deviance 21527.13 (25651 dof) deviance 20163.2 (25640 dof) iters 5

This number is the logarithm of the odds of the individual defaulting on a loan. To get the raw probability, you take the inverse logit of the number. This gives us a predicted probability of

A For loop is used to calculate the predicted probabilities using R:

> mypred=c() #create a nempty list to fill

|  |
| --- |
| > for (i in 1:nrow(train)){mypred[i]=ilogit(sum(earthmod$bx[i,]\*earthmod$glm.coef))}  > rpreds <- predict(earthmod, type='response')  > preds <- cbind(mypred, rpreds)  > head(preds)  mypred Response  [1,] 0.20833911 0.20833911  [2,] 0.09940128 0.09940128  [3,] 0.10939251 0.10939251  [4,] 0.39959430 0.39959430  [5,] 0.16139481 0.16139481  [6,] 0.20432251 0.20432251  By utilizing the matrix of basis functions and the glm coefficients, we can accurately obtain any desired predictions.  Now we want to obtain predictions from the validation set, so again the following is an example.  For example if you had requested a loan amount of 5000 and has had one inquiry in the last six months. Therefore,  the predicted probability of this person defaulting on a loan is:  We can still use the predict() command to get the desired probabilities: |
|  |
| |  | | --- | | > | |

> rvalpreds <- predict(earthmod,type='response',validate)

> head(rvalpreds)

Response

[1,] 0.1483151

[2,] 0.2804043

[3,] 0.1527276

[4,] 0.1464939

[5,] 0.2436860

[6,] 0.1349074

The above predictions are closely matching the example used.

After installing pROC from the library we can obtain the ROC curves for both the training and validation sets, using the roc() command

> trainroc <- roc(train$Response, rpreds)

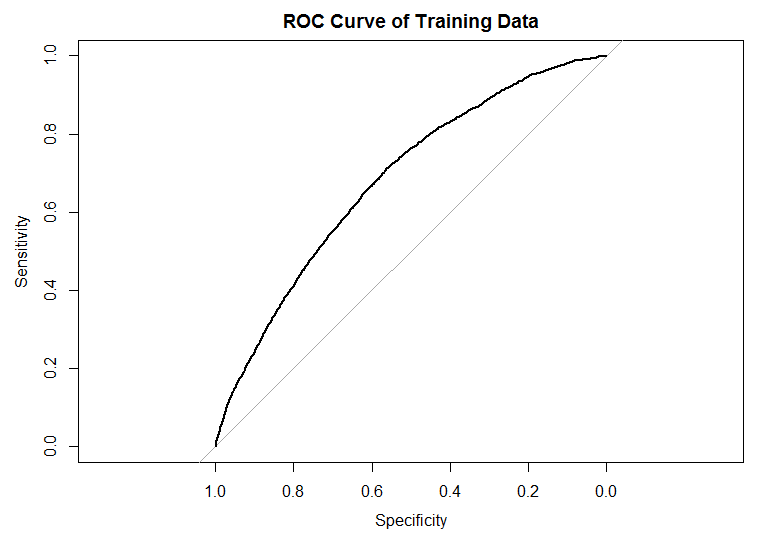
> plot(trainroc, type='l', main="ROC Curve of Training Data")

Call:

roc.default(response = train$Response, predictor = rpreds)

Data: rpreds in 21850 controls (train$Response 0) < 3802 cases (train$Response 1).

Area under the curve: 0.6818



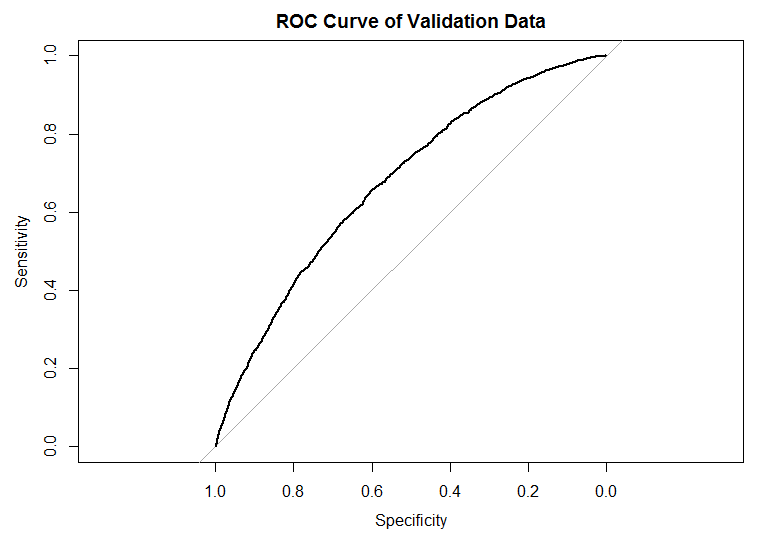
plot(validroc, type='l', main='ROC Curve of Validation Data')

Call:

roc.default(response = validate$Response, predictor = rvalpreds)

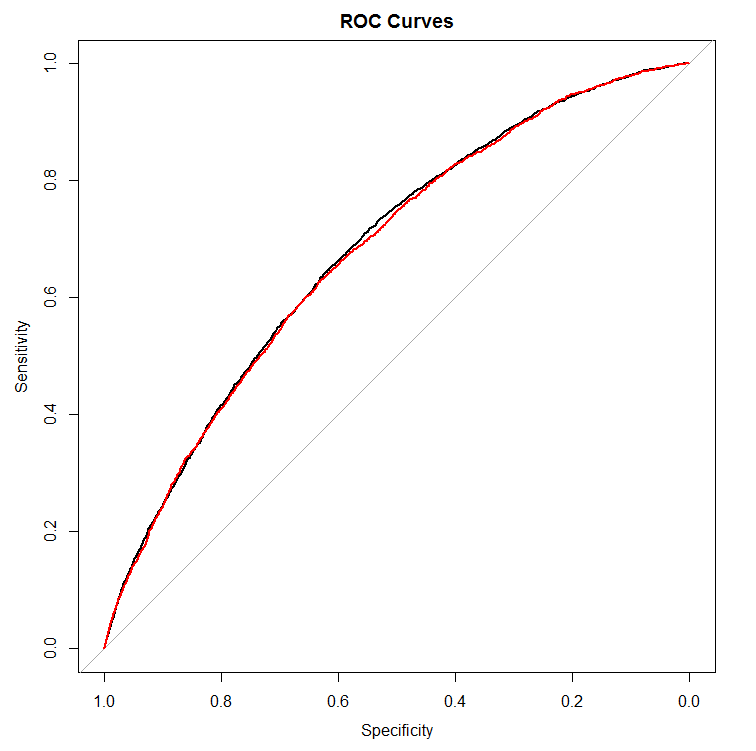
Data: rvalpreds in 14362 controls (validate$Response 0) < 2492 cases (validate$Response 1).

Area under the curve: 0.6755



> plot(trainroc, type='l', main='ROC Curves')

> plot(validroc, type='l', add=TRUE, col=2)



Between the two models there isn’t much difference and we can conclude that the model is good

Degree 2 model.

The summary of the best fitting degree 2 model is as follows:

> earthmod2 <- earth(Response~.-Loan\_ID-Member\_ID, data=train,glm=list(family=binomial),degree=2)

> summary(earthmod2, style='max')

Call: earth(formula=Response~.-Loan\_ID-Member\_ID, data=train,

glm=list(family=binomial), degree=2)

GLM Response =

-1.095864

+ 0.3324365 \* GradeD

+ 0.9870391 \* GradeG

+ 0.8101456 \* IncFactLow

+ 0.3840974 \* IncFactMed

+ 2.001168e-05 \* max(0, Loan\_Amnt - 6000)

- 0.009965121 \* max(0, FICO\_Low - 675)

- 0.1634832 \* max(0, 8 - Inq\_6mnths)

+ 0.04617126 \* max(0, Inq\_6mnths - 8)

+ 0.01112722 \* GradeC \* max(0, FICO\_Low - 675)

+ 0.01074906 \* GradeD \* max(0, FICO\_Low - 675)

+ 0.09373678 \* GradeE \* max(0, 8 - Inq\_6mnths)

+ 0.1349823 \* GradeF \* max(0, 8 - Inq\_6mnths)

Earth selected 13 of 15 terms, and 10 of 20 predictors

Termination condition: RSq changed by less than 0.001 at 15 terms

Importance: FICO\_Low, Inq\_6mnths, IncFactLow, GradeF, Loan\_Amnt, GradeD, GradeE, GradeG, IncFactMed, GradeC, GradeB-unused, ...

Number of terms at each degree of interaction: 1 8 4

Earth GCV 0.1198034 RSS 3065.775 GRSq 0.05111333 RSq 0.05333156

GLM null.deviance 21527.13 (25651 dof) deviance 20178.12 (25639 dof) iters 5

For example consider a FICO\_Low of 740, a requested loan amount of 2500, a IncFact value of Low, and five inquiries in the last six months.

Thus, we can get a predicted probability of:

The above result might slightly vary from the exact prediction of 0.2478.

A for loop is used to calculate the exact training set predictions similar to the degree1 model.

> mypred2=c()

> for (i in 1:nrow(train)){mypred2[i]=ilogit(sum(earthmod2$bx[i,]\*earthmod2$glm.coef))}

The predict() command could be used to get all the predictions and would compare to our obtained predictions.

> rpreds2 <- predict(earthmod2, type='response')

> preds2 <- cbind(mypred2, rpreds2)

> head(preds2

+ )

mypred2 Response

[1,] 0.23202867 0.23202867

[2,] 0.09806301 0.09806301

[3,] 0.16094469 0.16094469

[4,] 0.38776329 0.38776329

[5,] 0.15984016 0.15984016

[6,] 0.20925993 0.2092599

> rvalpreds2 <- predict(earthmod2, type='response', newdata=validate)

After installing pROC from the library we can obtain the ROC curves for both the training and validation sets, using the roc() command

> trainroc2 <- roc(train$Response, rpreds2)

> validroc2 <- roc(validate$Response, rvalpreds2)

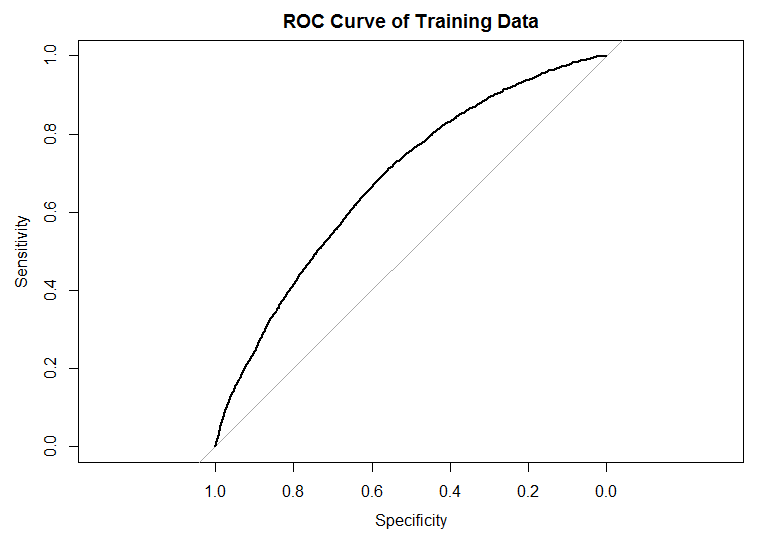
> plot(trainroc2, type='l', main="ROC Curve of Training Data")

Call:

roc.default(response = train$Response, predictor = rpreds2)

Data: rpreds2 in 21850 controls (train$Response 0) < 3802 cases (train$Response 1).

Area under the curve: 0.6807



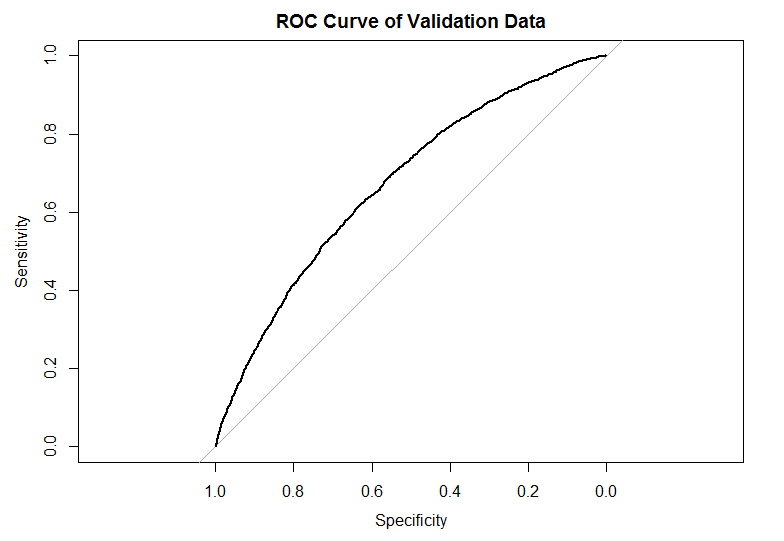
plot(validroc2, type='l', main='ROC Curve of Validation Data')

Call:

roc.default(response = validate$Response, predictor = rvalpreds2)

Data: rvalpreds2 in 14362 controls (validate$Response 0) < 2492 cases (validate$Response 1).

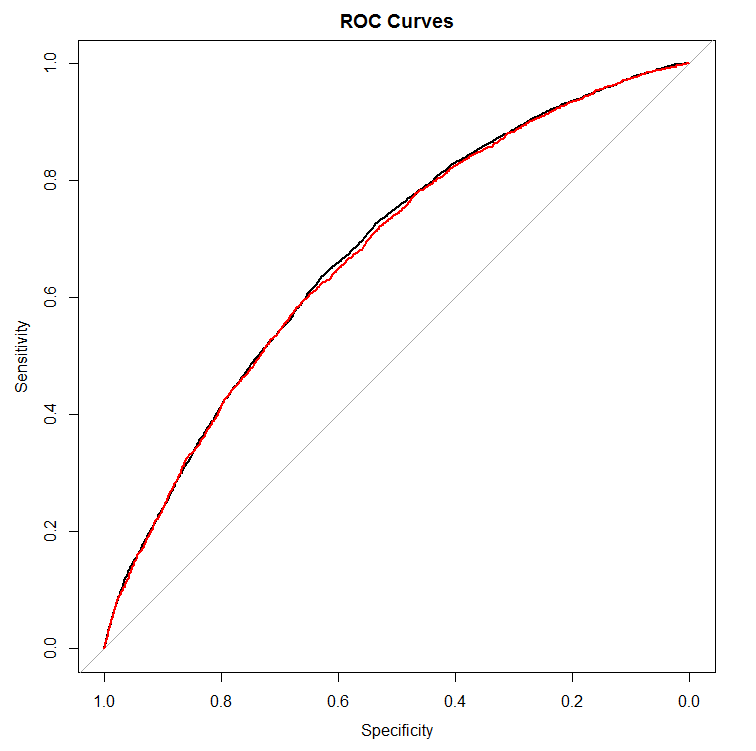
Area under the curve: 0.6707



Looking at the same graph, the sample ROC is the black curve, and the validation ROC is the red one:

> plot(trainroc2, type='l', main='ROC Curves')

> plot(validroc2, type='l', add=TRUE, col=2)



The model is consistent in the new data since the curves aren’t significantly different.

By comparing both the models we can determine whether the addition of the interaction terms was significant or not.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **R-Sq** | **RSS** | **GCV** | **GRSq** |
| No Interaction | 0.053 | 3065.09 | 0.1197 | 0.0519 |
| Interaction | 0.053 | 3065.775 | 0.1198 | 0.0511 |

R-Sq: The R-Sq is the coefficient of determination and it measures the goodness-of-fit of a model. It has a scale from 0 to 1, with 0 being almost no fit and meaning a perfect fit on all data points. Both models give the same very small value of 0.053, so these models perform the same in this regard, but they do not perform very well.

RSS: We do prefer a smaller RSS, hence the no interaction model is preferred, with a smaller margin.

GCV: The Generalized Cross Validation is a statistic that uses the RSS and penalizes it based upon the number of parameters and number of observations. It seems to form a similar function as the AIC/BIC statistics. The smaller values are better, but both of our models perform the same.

GRSq: This stands for Generalized R-Squared. This, according to R documentation, is “an estimate of the predictive power of the model.” This, as the name suggests, is very similar to the R-Sq statistics, therefore larger values are preferred, in which case the no interactions model is preferred.

We can conclude that the model with no interactions would be preferred. Both the models are similar in each statistic including the ROC plots. Hence we could choose the simpler model.