How a Wide of Family, Community, School and Individual Effect 8th Grade Students’ Science IRT Score in the United States

HUDM 6026 Final Project:

Luyang Guo (lg2856)

Boyi Xue (bx2138)

Haoyan Zhang (hz2400)

Teachers College, Columbia University

Abstract

This paper mainly focuses on estimating students' science IRT score in the U.S by their family, community, school and individual information and fit models with informative predictors to predict students' science IRT score. By using "The Early Childhood Longitudinal Study, Kindergarten Class of 1998-99 (ECLS-K)" dataset, we choose one dependent variable and 28 independent variables to indicate all four aspects in our research topic. To fit the best model, we calculate error rate using five methods: ordinary least squares (OLS) regression, subset selection, lasso, ridge and random forest. It turns out that all the error rates in five methods are different in a small decimal and error rate for model using stepwise selection is 60.02005, which is the lowest.

Keywords: ECLS-K, subset selection, lasso, ridge, random forest

How a Wide range of Family, Community, School and Individual Effect 8th Grade Students' Science IRT Score in the United States

Education is an important determinant of not only individual quality of life, but through building human capital, which has an impact on economic growth, as well. (Badr, 2010). For more than a decade, educators have emphasized the importance of mathematics and science literacy (Ma & Wilkins, 2002). Especially, considerable investments have been made to investigate crucial factor of influence on student acquisition of knowledge and skills in science, such as the National Assessment of Educational Progress (NAEP) and the Third International Mathematics and Science Study (TIMSS). An interesting research contributed to look at the effects of SES (social economic status) and school resources on 8th grade student performance in Egypt using data from TIMSS 2007. Other researches focused on effects of school racial composition (Michelson, Bottia & Lambert, 2013), parent characteristics, teacher qualification and peer effect on student science achievement (Jiang & Huang, 2012).

However, few dissertations devote to build an effective prediction model to predict students' science achievement with family, community, school, and individual variables. Our project concentrates on how a wide range of family, community, school, and individual variables affect 8th grade student's science IRT score in the United States.

We first use stepwise model selection methods to find a best model with statistical significance. In part two, we use lasso regression, ridge regression and random forest methods and compare their cross-validated performance on mean squared prediction error. By doing so, we get the best prediction model with family, community, school, and individual variables with expected lowest error rate to predict upcoming students' science score performance.

Data Description

As mentioned earlier, we use "The Early Childhood Longitudinal Study, Kindergarten Class of 1998-99 (ECLS-K) dataset in our research. The ECLS-K was developed under the sponsorship of the U.S. Department of Education, Institute of Education Sciences and National Center for Education Statistics. With a total of 21,260 kindergartners throughout the nation participated, this data set contains dependable and comprehensive data that could be used to analyze students' development and achievements in kindergarten and elementary school. Not only including the information of children themselves, this dataset also involves information about children's families, teachers, schools and care providers on their cognitive, social, emotional and physical development, all of which provide representative and diverse variables to measure the research question we are focused.

To analyze how students' science IRT score are influenced by a range of family, community, school and individual situations, 8th grade science IRT score is used as the outcome variable. There are other 25 independent variables representing family circumstance, student information, school and academy information and community circumstance. Meanwhile, we also contain gender, race and socioeconomic status as demographic variables in our model. See Table 1 for details about variables' indicator and their type and domain.

Table 1

Variable Definitions

Dependent Variable	
C7R2SSCL	Science IRT scores in eighth grade as defined in ECLS-K
Independent Variable	
Demographic Variables	

GENDER	Dummy: 1 for male, 2 for female
WKWHITE	Dummy: 1 for white, 2 for others
WKSESL	Social-economics index of occupational status (values range from -3.61 to 2.69)
Family Variables	
P2TVRULE	Dummy, value 1 if one has family rules for TV
P7HELPR	Dummy, value 1 if have someone help with reading homework
P7HFAML	Dummy, family type, value 1 if have 2 parents plus siblings
P7HMEMP	Dummy, current mother employment status, value 1 if work 35 hours or more per week
W8PARED	Dummy, parent highest education level, value 1 if educated 8th grade or below
W8INCCAT	Dummy, family computed income category, value 1 if earn $5,000 or less
P7ATTENP	Dummy, value 1 if parent attend PTA meeting
Student Variables	
C7ANGRY	Dummy, the extend of angry when had trouble learning, value 1 if it is not true at all
C7LIKRD	Dummy, whether like reading or not, value 1 if not like reading at all
C7LONLY	Dummy, whether often feel lonely or not, value 1 if not feel lonely at all
C7FLGOOD	Dummy, whether feel good about yourself, value 1 if strongly disagree
C7ADVIPA	Dummy, if listen to parents advice on important decisions, value 1 if it is true
School and Academy Variables	
S7ADA	Dummy, average percent of daily attendance for year, value 1 if attend

	less than 90%
S7REGSKL	Dummy, value 1 if it is a regular public school
C7R4RSCL	Reading IRT scores for 8th grade students
C7R4MSCL	Math IRT scores for 8th grade students
A4OFTSCI	Dummy, the frequency of science study, value 1 if never study science
A4TXSCI	Dummy, time for science study, value 1 if study 1-30 minutes one day
S2AFTSCH	Dummy, value 1 if school offers after school care
Community Variables	
S2CMNITY	Dummy, the location of community school, value 1 if in rural, farming community or Indian reservation
S2TRFFIC	Dummy, the extend of problem with heavy traffic, value 1 if it is a big problem
S2VLENCE	Dummy, the extend of problem with violent crime, value 1 if it is a big problem
S2SIGNIN	Dummy, value 1 if all visitors must sign in
S2TNSION	Dummy, the extend of tension from differences, value 1 if it is a big problem
S2SECURT	Dummy, value 1 if employ security guards

Method

Ordinary least squares (OLS) regression works well in many real-world applications. In OLS, we fit a linear model of the form as below with the lowest squared error.

$$Y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \ldots + \beta_p x_{pi} + \varepsilon_i$$

When there are amount of predictors and some of them may not informative, OLS will estimate small coefficients for the variables, but will not justify them as a waste which will results in overfitting noise and correspondingly make weak prediction to the new cases. Thus, when we do not have confirmative experience or knowledge about the research question, we may contain many non-formative predictors in our model. In these situations, we could deal with these non-informative variables using selection methods. In our research, three methods for models selection are performed: Subset Selection containing Stepwise Selection and Best Subset Selection according to AIC or BIC; Regularization/Shrinkage (i.e. Lasso and Ridge) and Random Forest according to 10-cross validation error rate.

Subset Selection

Best subset selection. Best subset selection involves fitting the OLS model for every possible combination of the predictors and picking the one that is best according to AIC or BIC criteria. This method is computationally demanding to fit all possible models.

Forward, Backward and Stepwise subset selection. Instead, forward and backward stepwise approaches to model selection approximate the best subset solution by working with a more restricted set of models with the same criteria AIC or BIC as best subset selection. Specifically, forward stepwise selection begins with a null model and moves forward one predictor at a time to get the best AIC or BIC, backward stepwise selection begins with the full model (i.e., containing all p predictors) and then removes the least useful predictors according to AIC or BIC one at a time.

AIC and BIC. The AIC (Akaike's Information Criterion) and BIC (Bayesian Information Criterion) are measures of relative fit of statistical models. The AIC and BIC are both based on negative two times the maximized value of the likelihood (-2LL). They are defined as follow

$$AIC = 2 * \mathrm{k} - 2LL$$

$$BIC = \ln(n) * k - 2LL ,$$

where k is the number of parameters and LL is the log-likelihood value at the MLE. AIC and BIC are similarly constructed, with the essential difference being the term multiplied by k, the number of parameters estimated. The BIC tends to penalize model complexity more heavily than the AIC. Comparatively speaking, the AIC is somewhat more permissive of model complexity than the BIC, it may be preferred for prediction. On the other hand, the BIC tends to create a simpler model which is more suitable for explanation. In this research, we choose AIC as our selection criteria.

The Limitations of Stepwise Selection. Forward and backward stepwise approaches to model selection approximate the best subset solution by working with a more restricted set of models, so it may have some drawbacks. Firstly, inflation of Type I error rate when testing the significance of predictors. It is, therefore, not appropriate to interpret the statistical significance of regression coefficients selected by stepwise selection routines at face value. For the purpose of making good predictions, on the other hand, stepwise selection methods are very useful because they can eliminate non-informative variables. In addition, functional form assumptions are strong. If you do not specify any squared terms or interactions in the "upper" model, the assumption is that all predictors are linearly related to the outcome. However, we are not sure about where there is an interaction or not.

With more than 20 variables in our initial model, it is hard to identify where there should be an interaction or whether there are any squared terms in the “upper” model. So, in fact, we are not so confident with our functional form. In spite of those limitations, we use AIC forward and backward subset selection method to identify the best predictor model for student science score.

Regularize/Shrinkage method

Shrinkage method is another way to mitigate the noise predictors from a full model by regularizing or shrinkage the coefficient of these predictors toward zero. The two best-known techniques to do shrinkage are Ridge and Lasso.

In OLS, we estimate betas by let the RSS (the residual sum of squares)

$$\mathrm{RSS} = \sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2$$

equal to zero. In Ridge and Lasso, the estimation is based on RSS but additional penalties for big beta are considered.

Ridge. In Ridge, the penalty is the multiple of l2 norm and the tuning parameter λ. The l2 norm of x is defined as $\|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$. The estimation of coefficients are the betas that minimize

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2 = RSS + \lambda\,\|\beta\|_2$$

where λ is the tuning parameter which is positive. The disadvantage of ridge is that it can only shrink the coefficients but cannot force them to be exactly zero.

Lasso. The penalty function used in lasso is called l1 norm, which is defined as $\|x\|_1 = \sqrt{\sum_{i=1}^{n} x_i}$. The estimations are the betas that satisfy minimizing the following equation

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j = RSS + \lambda\,\|\beta\|_1$$

where λ is the tuning parameter which is positive.

When performing shrinkage methods, we need to decide a best λ to get the minimum predict error rate, which both tunes the model to contain less predictors that are informative and keeps adequate predictors to explain response variable. Cross Validation is a method to get the test error of fitted model on a set of sample.

Validation set & 10-Cross Validation. The idea behind Validation set is that it randomly divided the data into two parts, a training set and a validation set; and uses the training set to fit the model and the fitted model is used to predict response variable using test set and calculate the MSE as test error rate. Cross validation is an extension of validation set considering the variation of MSEs. 10-Cross validation is randomly divided the data into ten folds and each time use one of them as the test set and average ten MSEs as the model's error rate. In our research, we use R package glmnet and retrieve the test error rate from function cv.glmnet.

Random Forest

Random forest is a notion of random decision forests that could improve over bagged trees. In tree bagging, we build a number of decision trees on bootstrapped training samples. For random forest, the training algorithm is the same as in tree bagging and bootstrap. Particularly, given a training set $\mathrm{X} = x_1 \cdots x_n$ with responses $\mathrm{Y} = y_1 \cdots y_n$ bagging repeatedly selects a random sample with replacement of the training set and fits trees to these samples:

For $\mathrm{b} = 1, \cdots, B$: 1.Take a random sample n from X, Y with replacement (i.e., a bootstrap sample) of cases and call these X_b, Y_b; 2.Train a decision or regression tree f_b on X_b, Y_b.

After training, predictions for unseen samples $\mathrm{x}^{'}$ can be made by averaging the predictions from all the individual regression trees on $\mathrm{x}^{'}$:

$$\hat{f} = \frac{1}{B}\sum_{b=1}^{B} \widehat{f_b}\,(x')$$

Otherwise, taking the majority vote in the case of decision trees will also works. During the process, a fresh sample of m predictors is taken at each split, and by default, the number of predictors considered at each split is approximately equal to the square root of the total number of predictors.

Out-of-bag Error Rate. In order to estimate the test error of random forest, as well as in a bagged model, we use the out-of-bag (OOB) outputs. Without performing cross-validation or the validation set approach, this way's focus is that trees are repeatedly fit to bootstrapped subsets of the observations. On average, each bagged trees will use two-thirds of the observations. And the remaining out-third will call out-of-bag (OOB) observations. And we can predict B/3 predictions by using each of the trees in the observation of OOB. Then, by averaging all these predicted responses, we could conclude to one single answer. Thus, the OOB MSE or classification error could also be computed. As the response for each observations is predicted using only the trees that were not fit using that observation, OOB error is a very convenient and valid estimate. That is why we use OOB error rate to estimate in random forest.

Results

Backward Stepwise Selection

Backward stepwise selection begins with the full model (i.e., containing all p predictors) and then removes the least useful predictors one at a time. The model selected by AIC backward stepwise selection keeps 15 variables in the model. The statistic table is showed in Table 2.

Table 2

Backward stepwise selection model with AIC

	Estimate	**Std. Error**	**t value**	**Pr(>|t|)**	
(Intercept)	10.455574	1.812973	5.767	8.6e-09	***
GENDER	-3.297211	0.240232	-13.725	< 2e-16	***
WKWHITE	-2.750159	0.306225	-8.981	< 2e-16	***
WKSESL	0.630419	0.205831	3.063	0.002206	**
W8INCCAT	0.096152	0.055655	1.728	0.084119	.
C7ANGRY	-0.203620	0.130709	-1.558	0.119348	
C7LIKRD	0.405679	0.120354	3.371	0.000756	***
C7LONLY	0.276523	0.172553	1.603	0.109106	
C7FLGOOD	-0.488206	0.199595	-2.446	0.014483	*
S7REGSKL	-0.580780	0.252042	-2.304	0.021251	*
C7R4RSCL	0.237435	0.007256	32.721	< 2e-16	***
C7R4MSCL	0.298681	0.008382	35.632	< 2e-16	***
S2AFTSCH	-0.644901	0.242882	-2.655	0.007954	**
S2CMNITY	-0.127907	0.050836	-2.516	0.011902	*
S2TRFFIC	0.357823	0.184788	1.936	0.052881	.
S2TNSION	0.421538	0.245626	1.716	0.086198	.

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.734 on 4535 degrees of freedom

Multiple R-squared: 0.6981, Adjusted R-squared: 0.6971

F-statistic: 699.2 on 15 and 4535 DF, p-value: < 2.2e-16

The F-test of the model is significant, as well as the majority of variables in this best AIC model. However, there are two exceptions: "C7LONLY" (student often feels lonely) and "C7ANGRY" (angry when have trouble learning). We may also eliminate these variables in following data analysis.

From statistic table for the best AIC model, we can find some interesting results. In demographic variables, female and the white perform not as good as male and other races in science IRT score. The coefficient for SES is 0.6, by which means one unite increase in SES will lead science IRT score to increase 0.6 points, holding other variables constant. Such kind of positive effect also showed in family variables - Household Income.

For individual variables, like Reading has positive effect on science achievement, whereas, "Feel Good about Self" have negative effects. Like reading means good learning habit and over-confidence about self may lead to a failure.

For academic variables, the coefficient of Attend Regular Public School is -0.6, by which means science education quality of public school is not as competitive as private schools in the U.S. Reading IRT score and Math IRT score have positive relation with students' science score. It makes sense because good student may have higher IQ or have good learning strategy for every subject.

For community variables, economic factors pose essential impact on student science acquisition. If the community school located in rural, farming community or Indian reservation, it yields 0.12 reduction on science IRT score on average. On the other hand, heavy traffic problem is positive related to student science performance, which contributed to better education quality in big cities. In addition, Tension from Difference also has positive influence on science education. Sometimes, tension can benefit individuals in learning competence.

Forward Stepwise Selection

Forward stepwise selection start with the null model (i.e. no predictors) and moves forward one predictor at a time. The best AIC model selected with forward direction is shown in table 3.

Table 3

Forward stepwise selection model with AIC

	Estimate	**Std. Error**	**t value**	**Pr(>|t|)**	
(Intercept)	11.771654	1.668342	7.056	1.97e-12	***
C7R4MSCL	0.298105	0.008366	35.634	< 2e-16	***
C7R4RSCL	0.238072	0.007237	32.898	< 2e-16	***
GENDER	-3.317863	0.239807	-13.836	< 2e-16	***
WKWHITE	-2.785728	0.301987	-9.225	< 2e-16	***
WKSESL	0.750657	0.179603	4.180	2.98e-05	***
C7LIKRD	0.419519	0.119565	3.509	0.000455	***
C7FLGOOD	-0.564489	0.185213	-3.048	0.002319	**
S2TRFFIC	0.366863	0.184742	1.986	0.047114	*
S7REGSKL	-0.562554	0.252051	-2.232	0.025670	*
S2AFTSCH	-0.658075	0.242545	-2.713	0.006689	**
S2CMNITY	-0.122585	0.050676	-2.419	0.015602	*
S2TNSION	0.438216	0.245499	1.785	0.074328	.
P7HFAMIL	-0.217490	0.128566	-1.692	0.090780	.

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.736 on 4537 degrees of freedom

Multiple R-squared: 0.6979, Adjusted R-squared: 0.697

```
F-statistic: 806.1 on 13 and 4537 DF,  p-value: < 2.2e-16
```

With forward direction, the model keeps 13 variables and all of them have statistical significance. Compared with backward selection, forward stepwise selection omit family household income variable and select family type as predictor in prediction model. The coefficient of "P7HFAMIL" is -0.2, by which means the children living in single parent family may have lower science IRT score on average.

Ten cross-validation of linear models

Construct the linear regression of the full model, null model and best backward subset selection model using function "glm". Then use "cv.glm" to get the ten cross validation test error rate for these three models. We get their test error rate are 60.37471, 197.4805 and 60.02005.

Lasso and Ridge

When doing the lasso and ridge, first we define lambda as a sequence of 100 numbers in the range of 10e-2 to 10e+10 to guarantee that both the null and full model are included in these models. We can get the change of l2 norm and coefficients according to the change of log lambda. As shown in Figure 1, we can see when log lambda is -5, there is no change from the full model and the penalty l2 norm is really big; and when log lambda is increasing to 10, all coefficients are approaching zero and the l2 norm is zero. Thus we adjust lambda to be in the range of 10e-2 to 10e5 to do the ten cross validation. Calculate the ten cross validation error rate for all lambdas. We get the best lambda 0.1353048 with the lowest test error rate 60.2911 (see Figure 2).

Figure 1.

Ridge L2 norm and coefficients change with log(lambda)

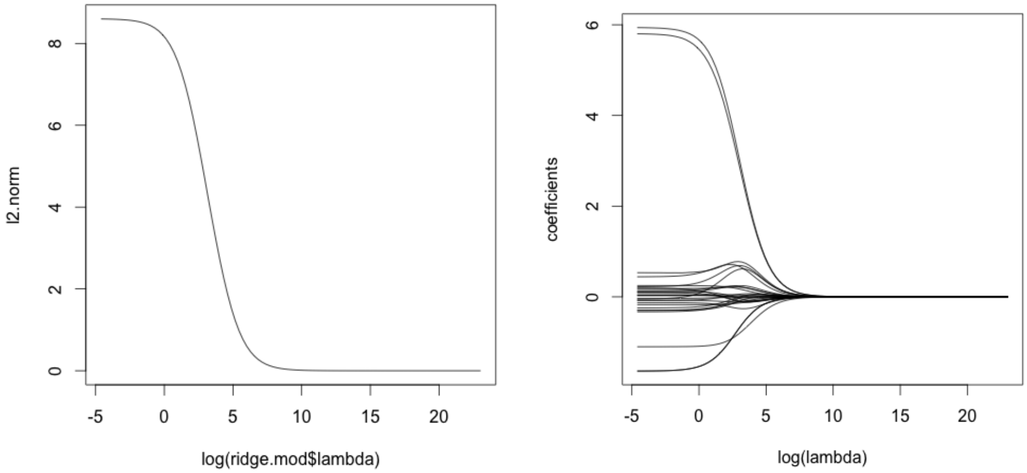
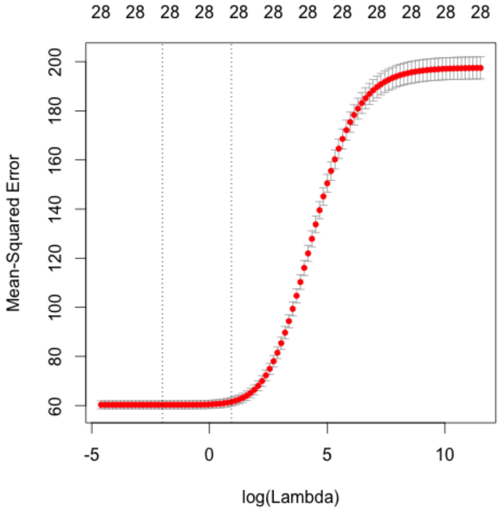


Figure 2.

Ridge MSE according to different tuning parameter lambda



Doing the same procedure for lasso. As seen in figure 3, the l1 norm is huge at first and drop to zero when log lambda is approaching 3, which is equal to 10e3; and correspondingly, all coefficients shrink to exactly zero when log lambda is at around 3. Finally, doing cross validation to the models with different lambdas the lowest test error rate from ten-cross validation is 60.30253 with lambda of 0.05722368.

Figure 3.

Lasso L1 norm and coefficients change with log(lambda)

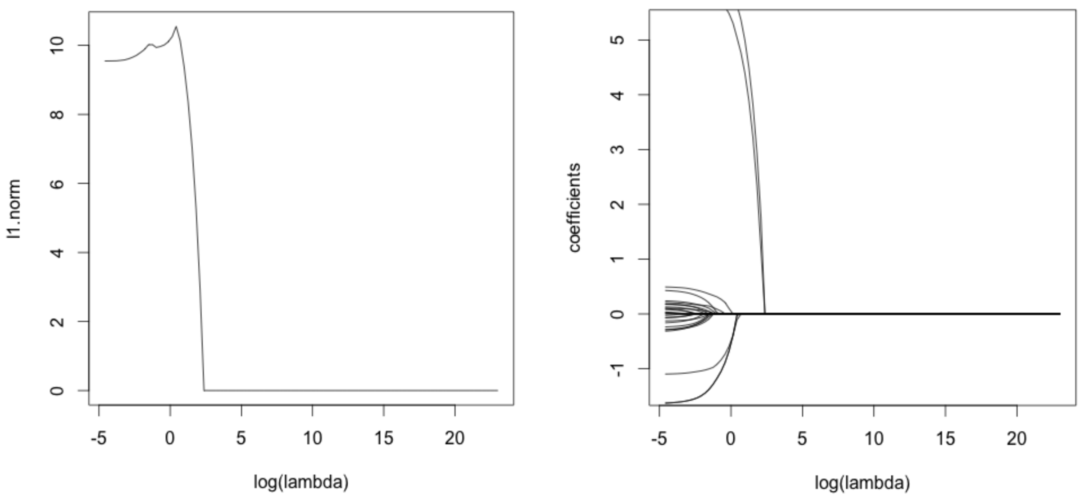
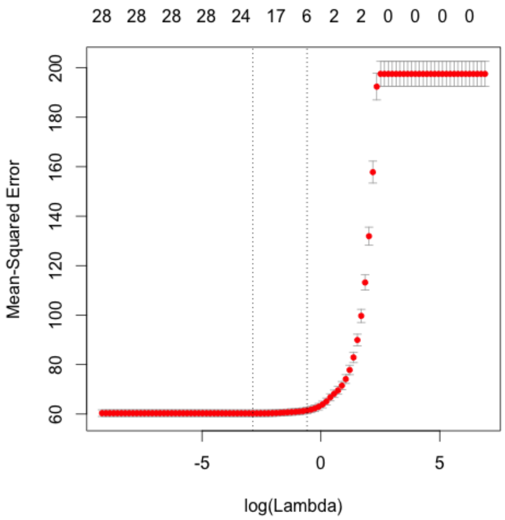


Figure 4.

Lasso MSE according to different tuning parameter lambda

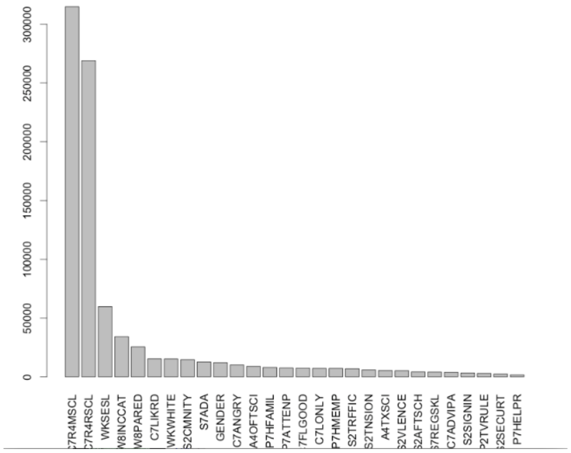


Random Forest

Here, we also include all variables to run random forest model fit. As seen in figure 5, variables "C7R4MSCL", "C7R4RSCL", "WKSESL", "W8INCCAT" and "W8PARED" fit model much better than the other variables. This means numerical variables like reading IRT score, math IRT score, social-economic status index and family income are highly correlated with our science IRT score. Besides, parent highest education level is also a very appropriate variable to interpret science IRT score.

Figure 5.

Histogram of selected variables in descending order in random forest



After that, variables “C7LIKERD”, “WKWHITE”, “S2CMNITY”, “S7APA”, “GENDER”, “S7ANGRY” and “A4OFTSCI” also reveals some correlations with science IRT score. This time, aspects in student, school and community in our research topic are included to show some relationships. In student circumstance, answers “whether like reading or not” and “the extend of angry when had trouble learning” shows relationship to our dependent variable to some extent. In school environment, variables like “average percent of daily attendance” and “the frequency of science study” are the variables that we should keep in our final model. In community environment, the location of community school turns out to influence the IRT science score to some extent. Besides, gender and race status are also suitable demographic variables to interpret.

By calculating out-of-bag (OOB) error rate in random forest, we get the error rate of 61.82561.

Discussion

Summary of the ten-cross validation error rate for five methods could be concluded in table 4. Error rates from null model to full model is from 197.4805 to 60.37471. The error rate for the third linear regression on predictors selected by stepwise selection has the minimum predicted error rate of 60.02005. The error rates of Ridge (60.2911) and Lasso (60.30253) are both smaller than that of full model (60.37471), whereas the error rate of random forest (61.82561) is higher than the full model. Among the last three methods, Ridge and Lasso performed similarly and approximately better in estimating error rate than random forest.

Table 4.

Summary of the ten-cross validation error rate for five methods

Method	**Mean MSE**
Full model	60.37471
Null model	197.4805
Model selected by stepwise selection	60.02005
Lasso	60.30253
Ridge	60.2911
Random forest	61. 82561

Conclusion

In our research, we research on the influence of students' family, community, school, and individual variables on their science achievement and predict the IRT science score using predictors in these four aspects. In part one, stepwise selection is performed to get the smallest AIC. In backward stepwise selection, keeping the significant variables, the outcome is positive related to SES, reading score and negative related to feel good about self, regular public school, community school located in rural, farming community or Indian reservation. In model selected by forward selection, there is only 13 predictors and children living in single parent family may have negative relationship with science IRT score on average.

In part two, our focus is on the prediction rather than interpretation using Lasso, Ridge and Random Forest. In the result, Ridge have the lowest error rate which slightly smaller than Lasso which may result from the feature that ridge will not force predictors to exactly zero and make the model considered all variables even the coefficients are tiny. Random Forest did not perform well in this research may result from the set of number of bagging and the number of trees. It is an improved tree method using bootstrap to reduce the variance of each different initial data set. When we can choose the parameters of random forest widely, it would be an accuracy method.

References

Angelo Canty and Brian Ripley (2016). boot: Bootstrap R (S-Plus) Functions. R package version 1.3-18.

A. Liaw and M. Wiener (2002). Classification and Regression by randomForest. R News 2(3), 18--22.

Badr, M. (2010). School effects on students test scores in Egypt. The University Of Nottingham, 2-3.

Davison, A. C. & Hinkley, D. V. (1997) Bootstrap Methods and Their Applications. Cambridge University Press, Cambridge. ISBN 0-521-57391-2

Douglas Bates and Martin Maechler (2016). Matrix: Sparse and Dense Matrix Classes and Methods. R package version 1.2-4. https://CRAN.R-project.org/package=Matrix

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning: With applications in R.

Jiang, H. & Huang, S. (2012). Research in Education Assessment and Learning (REAL). Iperc.org. Retrieved 9 May 2016, from http://iperc.org/REAL

Jerome Friedman, Trevor Hastie, Robert Tibshirani (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. Journal of Statistical Software, 33(1), 1-22. URL http://www.jstatsoft.org/v33/i01/.

Mickelson, R., Bottia, M., & Lambert, R. (2013). Effects of School Racial Composition on K-12 Mathematics Outcomes: A Metaregression Analysis. Review Of Educational Research, 83(1), 121-158. http://dx.doi.org/10.3102/0034654312475322

Ma, X. & Wilkins, J. (2002). The Development of Science Achievement in Middle and High Schoolr: Individual Differences and School Effects. Evaluation Review, 26(4), 395-417. http://dx.doi.org/10.1177/0193841x02026004003

Revolution Analytics and Steve Weston (2015). foreach: Provides Foreach Looping Construct for R. R package version 1.4.3. https://CRAN.R-project.org/package=foreach

R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Venables, W. N. & Ripley, B. D. (2002) Modern Applied Statistics with S. Fourth Edition. Springer, New York. ISBN 0-387-95457-0

Appendix

```
###Part 1
### Get data
data_o<-read.csv("HUDM6026.csv")
data<-data_o[,!names(data_o) %in% c("X...CHILDID","filter_.")]

#explore the structure of data frame
str(data)
head(data)

#build full model with all variables
ful_mod <- lm(C7R2SSCL ~ ., data = data)
summary(ful_mod)

# build null model containing only response variable
null_mod <- lm(C7R2SSCL ~ 1, data = data)

## use Stepwise Selection to select best fit model

library(MASS)
### AIC model selection
bestAIC <- stepAIC(ful_mod,direction = "backward") #start with full model, backward stepwise
selection
summary(bestAIC)


bestAIC2 <- stepAIC(null_mod,
            scope = list(upper=ful_mod, lower=null_mod),
            direction = "forward") # start with null model, forward stepwise selection
summary(bestAIC2)

bestAIC3 <- stepAIC(null_mod,scope = list(upper=ful_mod, lower=null_mod),
            direction = "both") # from both direction, the result is the
summary(bestAIC3)                    # forward or backward conditioning on initial model

# BIC model selection
n <- nrow(data)
bestBIC  <- stepAIC(ful_mod, direction = "both",  k = log(n))
summary(bestBIC)

############################################
### Part 2

######################################################################
################    Ols Model    #####################################
######################################################################
```

```
### Fit ols by glm
library(boot)
glm1 <- glm(C7R2SSCL ~ ., data = data, family = gaussian)  # Full model
glm2 <- glm(C7R2SSCL ~ 1, data = data, family = gaussian)  # Null model
glm3 <-
glm(C7R2SSCL~GENDER+WKWHITE+WKSESL+W8INCCAT+C7ANGRY+C7LIKRD+C7
LONLY+C7FLGOOD+S7REGSKL
          +C7R4RSCL+C7R4MSCL+S2AFTSCH+S2CMNITY+S2TRFFIC+S2TNSION,data =
data, family = gaussian) # Stepwise selection
summary(glm1)
### The default cost function is mean squared error, which works for continuous outcome
set.seed(1)
cv.err <- cv.glm(data = data, glmfit = glm1, K = 10)
cv.err1 <- cv.glm(data = data, glmfit = glm2, K = 10)
cv.err2 <- cv.glm(data = data, glmfit = glm3, K = 10)

### Get 10-fold CV error estimate
cv.err$delta[1]
cv.err1$delta[1]
cv.err2$delta[1]

######################################################################
################ Lasso and Ridge #####################################
######################################################################

### Scale data before using lasso or ridge
data<-data[,!names(data) %in% c("C7R2SSCL")]
sds <- apply (data, 2, sd)
matsds <- matrix(rep(sds, times = nrow(data)), nrow(data), byrow = TRUE)
data <- data/matsds
data<-cbind(data, C7R2SSCL = data_o$C7R2SSCL)

### Load relative packages
library(Matrix)
library(foreach)
library(glmnet)

### X and Y matrices
x=model.matrix(C7R2SSCL~.,data)[,-1]  # For use glmnet, construct x design matrix
y=data$C7R2SSCL                        # it can transform qualitative variables into dummy
variables

### Ridge
grid =10^ seq (-2,10, length =100)    # Conclude null and full model
ridge.mod = glmnet(x,y,alpha =0,lambda =grid )
```

```
dim(coef(ridge.mod)) #29*100 coef for 100 models
l2.norm<-sqrt(apply(coef(ridge.mod)[-1,]^2,2,sum))
plot(log(ridge.mod$lambda),l2.norm,type="l")
plot(log(ridge.mod$lambda),coef(ridge.mod)[2,],type="l",ylim=range(coef(ridge.mod)[-1,]),xlab
= "log(lambda)",ylab = "coefficients")
for (i in 1:27){points(log(ridge.mod$lambda),coef(ridge.mod)[i+1,],type="l")}
## there is almost no change after log(lambda)=10 which is
## in magnitude of 10e+4-10e+5
grid = 10^ seq (-2,5, length =100)
ridge.mod = cv.glmnet(x,y,alpha =0,lambda =grid)
min(ridge.mod$cvm)
ridge.mod$lambda.min
plot(ridge.mod)
cbind("ridge"=coef(ridge.mod, s = "lambda.min"), "full model"=coef(glm1))

### Lasso
grid =10^ seq (-2,10, length =100)
lasso.mod = glmnet(x,y,alpha =1,lambda =grid)    ## to find the lambda range containing
l1.norm<-apply(coef(lasso.mod)[-1,],2,sum)         ## both null and full model
plot(log(lasso.mod$lambda),l1.norm,type="l",xlab="log(lambda)")
plot(log(lasso.mod$lambda),coef(lasso.mod)[2,],type="l",ylim=range(coef(ridge.mod)[-
1,]),xlab="log(lambda)",ylab = "coefficients")
for (i in 1:27){points(log(lasso.mod$lambda),coef(lasso.mod)[i+1,],type="l")}

grid =10^ seq (-4,3, length =100)
lasso.mod = cv.glmnet(x,y,alpha =1,lambda =grid)
min(lasso.mod$cvm)
lasso.mod$lambda.min
plot(lasso.mod)




######################################
# ANOTHER TEST CODED BY OURSELVES ####
######################################

### Generate index for test and train
kfcv.sizes = function(n, k=10) {
  # generate sample sizes for k-fold cross validation on a data set of
  # size n

  # author: Matthias C. M. Troffaes
  # date: 22 Nov 2010
  # license: GPLv3
  sizes = c()
  for (i in 1:k) {
    first = 1 + (((i - 1) * n) %/% k)
    last = ((i * n) %/% k)
    sizes = append(sizes, last - first + 1)
  }
  sizes
}

kfcv.testing = function(n, k=10) {
  # generate testing sample indices for k-fold cross validation on a
  # data set of size n

  # author: Matthias C. M. Troffaes
  # date: 22 Nov 2010
  # license: GPLv3

  indices = list()
  sizes = kfcv.sizes(n, k=k)
  values = 1:n
  for (i in 1:k) {
    # take a random sample of given size
    s = sample(values, sizes[i])
    # append random sample to list of indices
    indices[[i]] = s
    # remove sample from values
    values = setdiff(values, s)
  }
  indices
}

### Generate train and test sets
set.seed(1)
test_index<-kfcv.testing(4551)
testset = list()
trainset = list()
for (i in 1:10){
  testset[[i]]<-data[test_index[[i]],]
  train_index <- setdiff(seq(nrow(data)),test_index[[i]])
  trainset[[i]]<-data[train_index,]
}

### Calculate error rate for OLS model
full.mse<-vector()
null.mse<-vector()
```

```
best.mse<-vector()
for (i in 1:10){

  # fit models with trainset
  olsfull.mod<-lm(C7R2SSCL~.,data = trainset[[i]])
  olsnull.mod<-lm(C7R2SSCL~1,data = trainset[[i]])

  # predict models use testset
  new = testset [[i]]
  new.x = new[,-29]
  full.pred = predict(olsfull.mod,newdata=new.x)
  null.pred = predict(olsnull.mod,newdata=new.x)

  # error
  full<-mean((full.pred-new[,29])^2)
  full.mse<-c(full.mse,full)
  null<-mean((null.pred-new[,29])^2)
  null.mse<-c(null.mse,null)

}
cbind("Full MSE"=mean(full.mse),"Null MSE"=mean(null.mse))

###Part 3: Random Forest
install.packages("randomForest")
library(randomForest)
set.seed(1314)
rf1 <- randomForest(C7R2SSCL ~ ., data = data)
imp <- data.frame(importance(rf1))
impnames <- rownames(imp)
impvals <- as.numeric(unlist(imp))
impnames2 <- impnames[order(impvals, decreasing = TRUE)]
impvals2 <- impvals[order(impvals, decreasing = TRUE)]
barplot(impvals2, names.arg = impnames2, las = 3)
print(rf1)

### Estimate estimate OOB error rate for the RF fit
### First get the predicted values based on the OOB units
preds <- rf1$predicted
### Then calculate the OOB MSPE
Y <- data$C7R2SSCL
(MSPE <- mean((preds - Y)^2))
```