David Guo DSC 424 Final Report

Non-Technical Summary:

The data I'm using is data about student performance on the subject of Portuguese of students in two secondary schools in Portugal. There are three different grades corresponding to three different periods of the school year.

The data contains 33 variables and 649 instances, so there are 649 students surveyed. The data was collected by using school reports and questionnaires. There are three numerical variables from 1-20 that measures grades for three quarters of the school year. The data also contain different types of variables like categorical variables such as Mjob (mother's job) and reason (reason the student chose the school). There are ordinal variables such as Medu (mother's education) and studytime (weekly study time on a scale from 1-5). There are also numeric variables like age and absences.

The first technique I used to analyze the data is principal component analysis in order to look at the ordinal variables in order to see how the data is related to each other. principal component analysis works by taking a matrix containing the data and breaking it down into different components. I also used other types of similar analysis for variables, such as principal factor analysis and common factor analysis in order to analyze the variables.

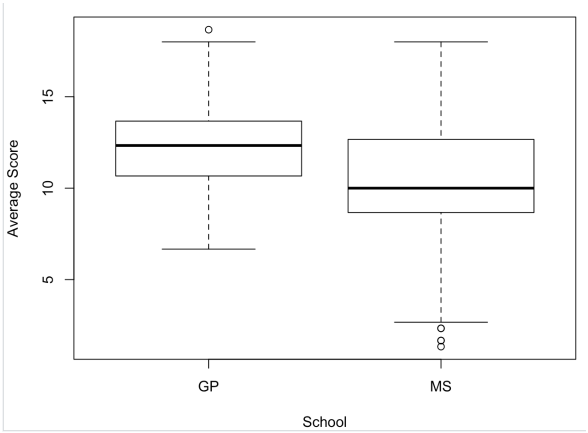
In the factor analysis for the variables, I noticed positive associations with Dalc, Walc (workday and weekend alcohol consumption) and goout (going out with friends), and these three variables have a negative association with famrel (family relationship) There is also positive relationship with Medu and Fedu (mother's and father's education). Also, famrel, freetime, and goout all have positive associations with each other (a student with more freetime will go out with their friends more and have a good family relationship).

The second technique I used is with different types of regression models. The first regression model is ordinary least squares (OLS). The goal of this model to minimize the sum of the square of the errors. Since there are 33 variables, I'm not going to use all of them. I used a selection algorithm called forwards selection to select variables that will best fit the data and created a model with those variables. I also created another OLS model where I replaced the ordinal variables in my OLS model with the components that I obtained from running common factor analysis on the data.

Another regression technique is called regularized regression, which tries to minimize the coefficients of the model, such as ridge regression, lasso regression and elastic net regression.

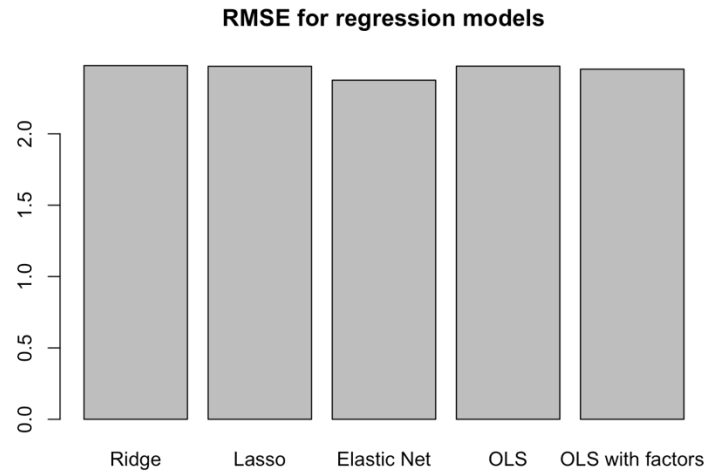
Ridge regression included all the variables in the model, while lasso and elastic net regression removed some variables. Elastic net regression had less variables than in the lasso regression, which had less variables than in the ridge regression.

For the variable selection models, some significant variables in all three models are the school (it looks like one school has lower scores on average than another school), the number of failures, whether or not the student has extra educational support, and the quality of family relationships.



(The GP school looks like it has significantly higher grades on average than the MS school).

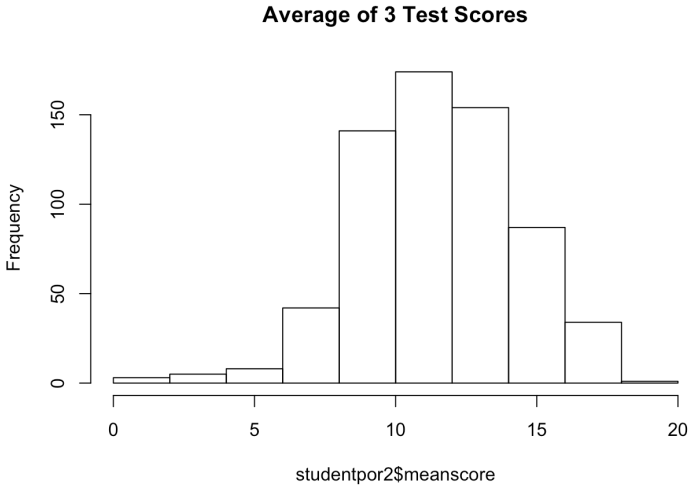
I split the data into a test and training set. I ran the models on the training set, and I predicted the data on the test set, and measured the errors of the predictions. For the models, the elastic net one had the lowest error (RMSE), though the errors were all pretty similar. Overall, the fitted models weren't too close to the actual data, in all 5 models, only around 40% of the variance in the grades can be explained by the model.



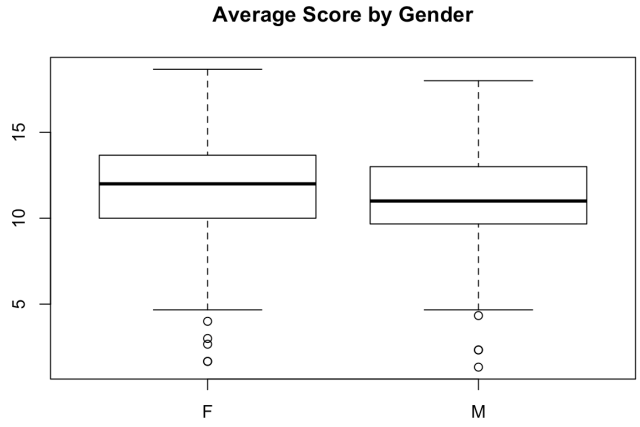
Exploratory Analysis:

I looked at the data, and I felt like most of the variables could be important to analysis. I tried to look at the three different types of variables and check to see if there is a relationship between grades and the variable. I also converted some of the factor variables into dummy binary variables, so I ended up with 39 variables in the new dataset. There weren't any missing values or data that otherwise looked off, so I didn't feel the need to clean the data. The grades were obviously the main data, and I

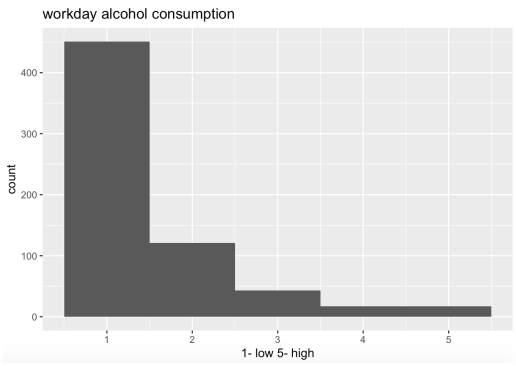
averaged the three grades into an aggregate grade. The grades were mostly normally distributed. Maybe a little left skewed due to the average being at 11.6.

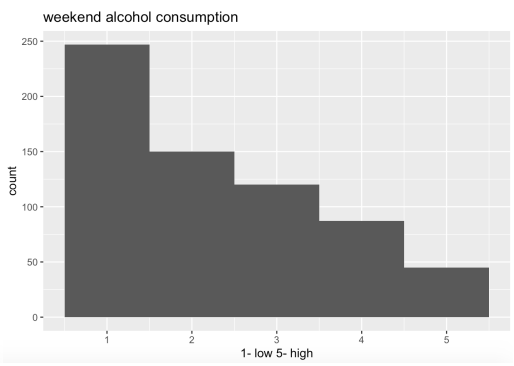


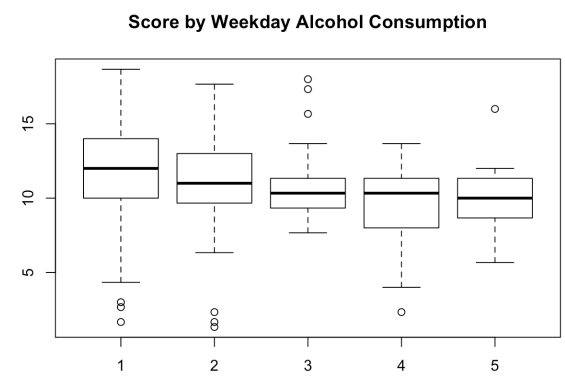
In regards to gender, there are more female than male students, and female students performed slightly better on average.

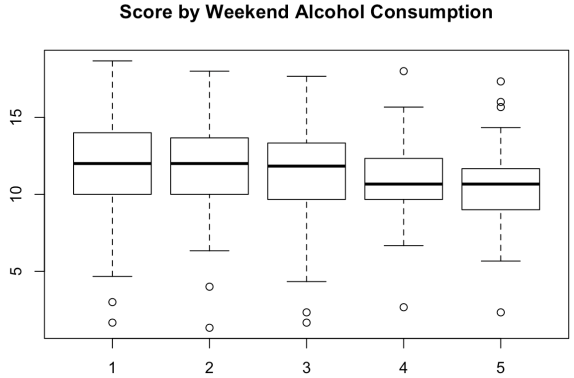


In Portugal there are two drinking ages for alcohol, 16 years for wine and beer and 18 for all types of alcoholic beverages. From the plots, most people consume little to no alcohol on the workdays and more people consume alcohol on the weekends. People who also drink more, especially on the weekdays, tend to get lower grades.

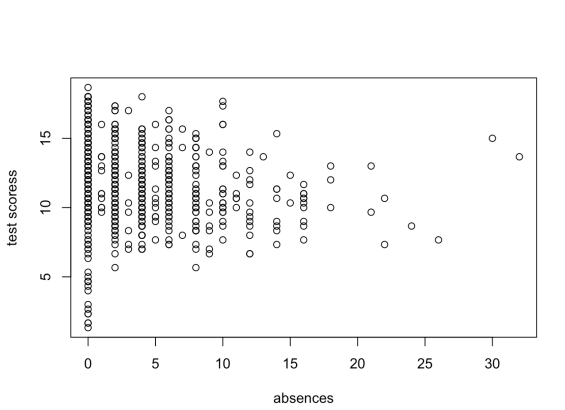


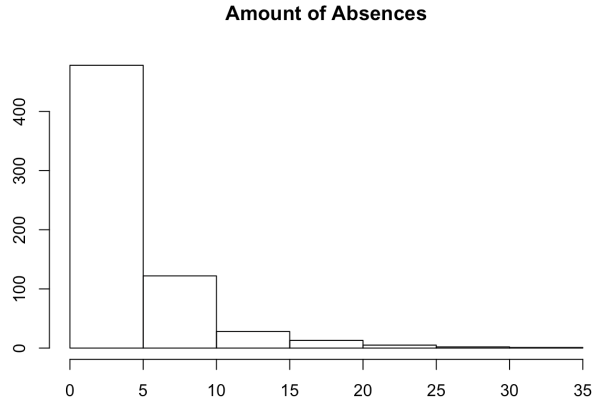






There appears to be a negative correlation between absences and grades, and the data of absences shows a right skew. I looked at the other numeric variables and I didn't see nonlinearities in the data, especially since the numeric variables in the data are all discrete.

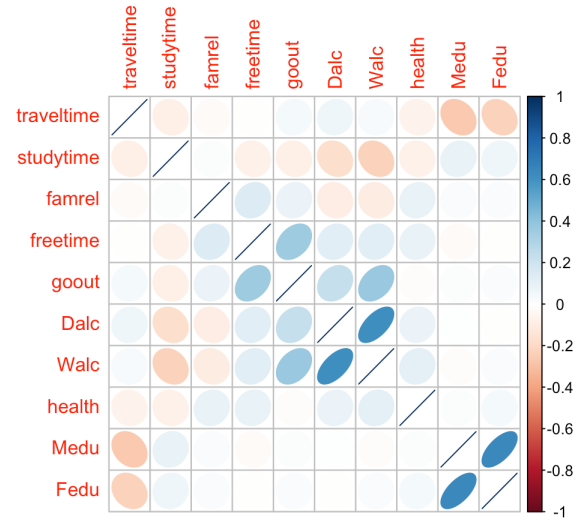




Since it looks like a potentially a lot of variables affect the grades data and they could be correlated to each other, I decided to use regularized regression, especially lasso and elastic net regression, to see what variables could be included in the model and what could be excluded. I also decided to use principal component analysis to look at the ordinal variables in the data, as there are a good number of ordinal variables in the data, and it's kind of hard to see their relationships based on a plot of the data.

Application of Techniques: Principal Component Analysis

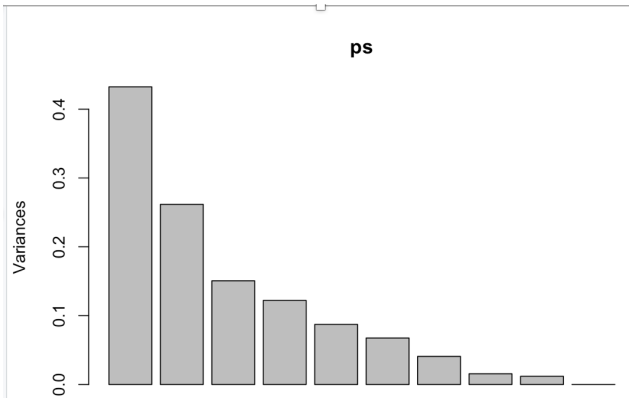
I decided to look into the ordinal variables in the dataset and conduct principal component and principal factor analysis on the data to see how the data is related to each other. Before analyzing the data, I created a data matrix of the Spearman correlations of the data. I think using Spearman makes sense because we are looking at students and students often have a class rank.




```
Importance of components:
                          PC1    PC2    PC3    PC4     PC5    PC6     PC7    PC8     PC9
Standard deviation     0.6575 0.5115 0.3881 0.3496 0.29542 0.2600 0.20191 0.1249 0.10926
Proportion of Variance 0.3634 0.2199 0.1265 0.1027 0.07334 0.0568 0.03426 0.0131 0.01003
Cumulative Proportion  0.3634 0.5832 0.7098 0.8125 0.88581 0.9426 0.97687 0.9900 1.00000
```

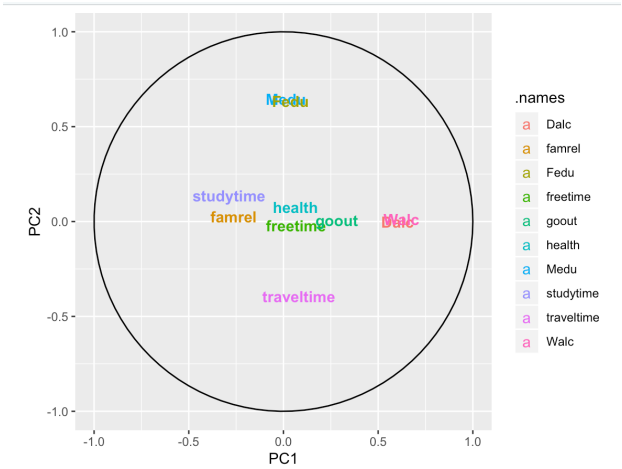
```
Rotation (n x k) = (10 x 10):
                   PC1         PC2          PC3
traveltime -0.26914435 -0.39225084  0.389488438
studytime   0.33371871 -0.25189792  0.242230919
famrel      0.10312789 -0.27430525 -0.511702509
freetime   -0.18381501 -0.02281028 -0.573638550
goout      -0.28318619  0.15124823 -0.338456726
Dalc       -0.40083819  0.36192077  0.201803914
Walc       -0.42621840  0.40991486  0.109014290
health     -0.03033286  0.01890222 -0.177494072
Medu        0.43158538  0.43696181  0.001009060
Fedu        0.40256180  0.44015594 -0.007432412
```

I conducted a principal component analysis based on the Spearman correlations of the data. Results of the PCA shows that the 1st component measures 33.6% of the variance in the data, 2nd measures 21%, and so on. In the first two components, Medu and Fedu (father's and mother's education), has the biggest loadings, along with Walc and Dalc (weekday and weekend alcohol consumption).



Looking at the scree plot, there is a knee at three components, so I will use at least 3 components for the principal factor analysis.

```
Loadings:
           RC1    RC2    RC3    RC4    RC5
Dalc        0.822
Walc        0.852
Medu               0.894
Fedu               0.892
famrel     -0.421         0.576
freetime                  0.767
goout       0.414         0.686
health                           0.901
traveltime                             -0.507
studytime                               0.858

                 RC1   RC2   RC3   RC4   RC5
SS loadings    1.806 1.734 1.418 1.080 1.051
Proportion Var 0.181 0.173 0.142 0.108 0.105
Cumulative Var 0.181 0.354 0.496 0.604 0.709
```



For principal factor analysis, I had to use 5 factors so all the variables could meet the 0.4 cutoff in at least one of the factors. The five factors ended up explaining 71% of the total variance. From the loadings, we can see that for the first component, there are positive associations with Dalc, Walc (workday and weekend alcohol consumption), and goout (going out with friends), and a negative association with famrel (family relationship). This makes sense given that if you drink alcohol on the weekdays, you'll drink more alcohol on the weekends, and you'll go out with friends to drink, and maybe if you don't have a good relationship with your family, you'll drink more. This first component mainly measures alcohol consumption. The second component shows that there is a positive association with Medu and Fedu (mother's and father's education), so this component mainly measures education. This makes sense as the more educated your mother is, the more educated your father is. For the third component, famrel, freetime, and goout all have positive associations (a student with more freetime will go out with their friends more and have a good family relationship). The fourth component has health as its own component with a positive association, but it's not really significantly correlated with the other variables in the analysis, so it's left by itself. The fourth component has a negative association with traveltime and studytime, which makes sense, as if you take more time commuting to school, you'll have less time to study.

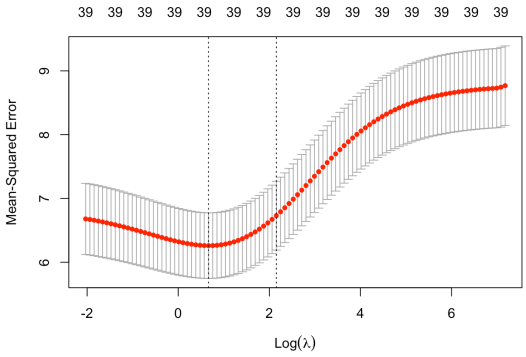
From the factor plot, we can see that Walc and Dalc are grouped together, Medu and Fedu, are grouped together, traveltime is by itself, and studytime, health, goout, famrel, and freetime are fairly closely grouped together, though not as closely grouped as Medu and Fedu, and Walc and Dalc are.

```
Call:
factanal(x = studentpor3, factors = 4)

Uniquenesses:
traveltime  studytime     famrel   freetime      goout       Dalc       Walc     health
     0.902      0.937      0.886      0.731      0.407      0.617      0.005      0.725
      Medu       Fedu
     0.157      0.496

Loadings:
           Factor1 Factor2 Factor3 Factor4
traveltime         -0.295
studytime  -0.213
famrel     -0.121           0.232   0.213
freetime                    0.499   0.122
goout       0.325           0.683  -0.145
Dalc        0.614
Walc        0.992
health      0.105                   0.510
Medu                0.914
Fedu                0.709

               Factor1 Factor2 Factor3 Factor4
SS loadings      1.548   1.438   0.792   0.362
Proportion Var   0.155   0.144   0.079   0.036
Cumulative Var   0.155   0.299   0.378   0.414

Test of the hypothesis that 4 factors are sufficient.
The chi square statistic is 8.6 on 11 degrees of freedom.
The p-value is 0.659
```

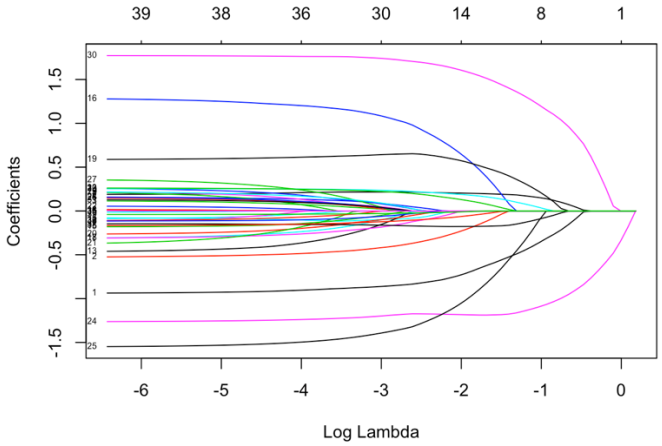
I also tried common factor analysis on the ordinal variables. Looking at the chi-squared score, it looks like 4 components gives a failure to reject H0, it's enough to capture the variance in the data, though there are three variables (traveltime, studytime, and famrel) that don't meet the 0.4 cutoff (5 factors don't meet the cutoff either). The four factors end up explaining 41.4% of the actual data. The common factor analysis reveals similar associations between the variables like in the principal factor analysis such as with alcohol consumption, parents' education, free time and going out, and health by itself.

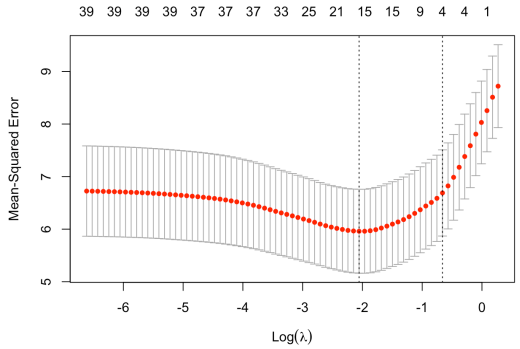
Application of Techniques: Regression

Before running any models, I split the data into a training and test set. I ran ridge regression on the data with all the predictors in the data. I got a lambda min 1.61, and looking at the graph, the mean squared error is minimized at the log lambda of 0.48 and the R squared was at 0.4, so around 40% of the variability in the dependent variable can be explained by the data.

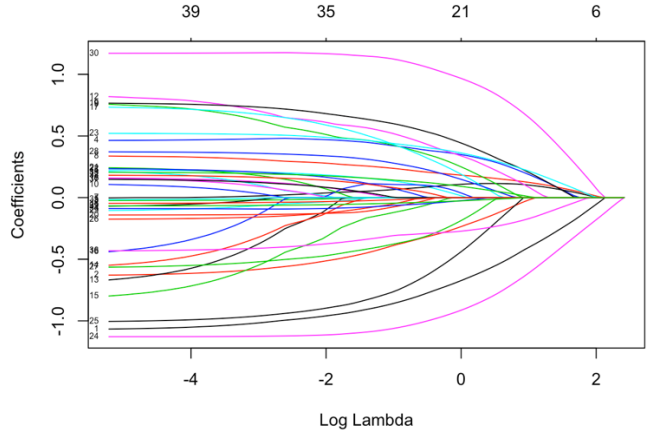


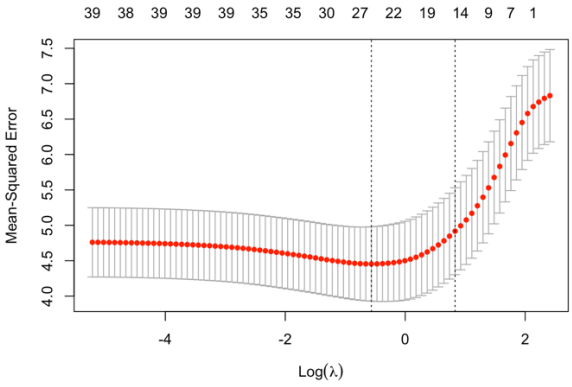
Running the lasso regression, I got a lambda min of 0.128, and looking at the graph, the mean squared error is minimized at the log lambda of around -2.05 The R squared was at 0.39, which is slightly smaller than the r squared of the ridge regression. The model at the log lambda of 0.39 removed reduced the coefficients of most of the variables to 0, removing 25 variables.





I then ran elastic net regression, and found that the alpha value of 0.1 produced the smallest MSE. The lambda min for model with an alpha 0.1 was 0.569, and the R squared value was 0.404. The elastic model at the log lambda of -0.564 removed 13 variables, 12 less than the lasso model. Some variables left in both lasso and elastic net models were: schoolMS (the school), sex, Medu (mother's education), studytime, and failures.





I also applied linear regression with forwards and backwards selection to the dataset, with the lower bound of the model being the empty model, and the upper bound being the model with all the variables in the data aside from the grade. For the linear model, both forwards and backwards selection produced the same model. I narrowed down the model selection to 12 variables. This model produced an R squared of 0.379.

```
Call:
lm(formula = scoretrainy ~ failures + higheryes + schoolMS +
    schoolsupyes + health + Medu + sexM + romanticyes + Fjobteacher +
    reasonreputation + internetyes + Dalc, data = scoretrainframe)

Residuals:
     Min       1Q   Median       3Q      Max
-10.6816  -1.2953  -0.1257   1.4771   6.9777

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     11.10405    0.71130  15.611  < 2e-16 ***
failures        -1.49165    0.23702  -6.293 1.06e-09 ***
higheryes        1.83363    0.45239   4.053 6.39e-05 ***
schoolMS        -0.96949    0.31246  -3.103 0.002093 **
schoolsupyes    -1.44596    0.42496  -3.403 0.000755 ***
health          -0.20714    0.09048  -2.289 0.022728 *
Medu             0.26823    0.13341   2.011 0.045225 *
sexM            -0.65308    0.28730  -2.273 0.023697 *
romanticyes     -0.55085    0.27510  -2.002 0.046116 *
Fjobteacher      1.03235    0.51622   2.000 0.046388 *
reasonreputation 0.52024    0.32258   1.613 0.107809
internetyes      0.61751    0.35377   1.745 0.081889 .
Dalc            -0.24399    0.14637  -1.667 0.096521 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.325 on 311 degrees of freedom
Multiple R-squared:  0.4019,    Adjusted R-squared:  0.3788
F-statistic: 17.42 on 12 and 311 DF,  p-value: < 2.2e-16
```

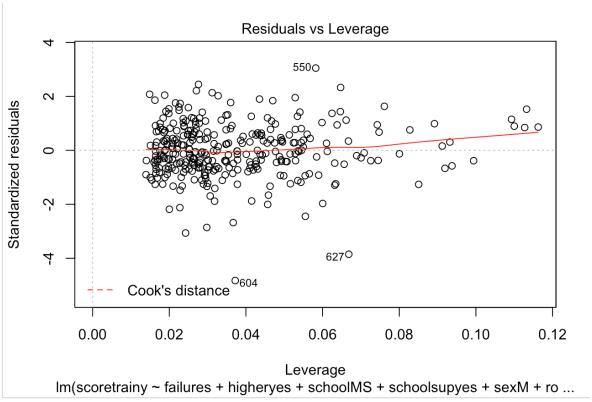
```
Call:
lm(formula = scoretrainy ~ failures + higheryes + schoolMS +
    schoolsupyes + sexM + romanticyes + Fjobteacher + reasonreputation +
    internetyes + alcohol + pedu, data = scoretrainframe2)

Residuals:
     Min       1Q   Median       3Q      Max
-11.0826  -1.3851  -0.1012   1.4834   6.9208

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      10.6511     0.5944  17.920  < 2e-16 ***
failures         -1.4730     0.2391  -6.159 2.25e-09 ***
higheryes         1.8392     0.4548   4.044 6.64e-05 ***
schoolMS         -0.9356     0.3149  -2.971 0.003198 **
schoolsupyes     -1.5516     0.4260  -3.642 0.000316 ***
sexM             -0.6808     0.2924  -2.329 0.020518 *
romanticyes      -0.5431     0.2771  -1.960 0.050891 .
Fjobteacher       0.9368     0.5253   1.783 0.075498 .
reasonreputation  0.6219     0.3217   1.933 0.054113 .
internetyes       0.6745     0.3551   1.899 0.058427 .
alcohol           0.3031     0.1650   1.837 0.067212 .
pedu             -0.2863     0.1578  -1.814 0.070682 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.34 on 312 degrees of freedom
Multiple R-squared:  0.3921,    Adjusted R-squared:  0.3707
F-statistic:  18.3 on 11 and 312 DF,  p-value: < 2.2e-16
```

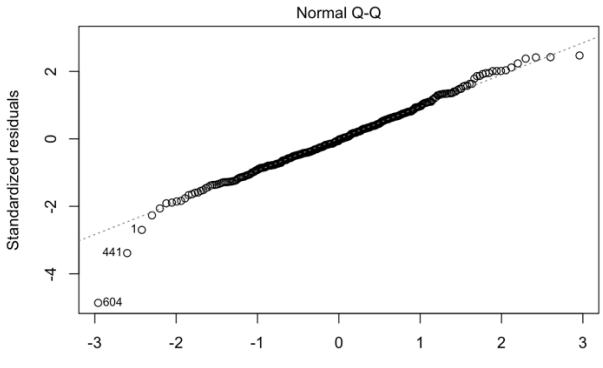
(OLS model with and without the CFA factor scores)

I also created another model where I added the factors from the common factor analysis on to the linear model created by backwards selection, and I removed the three ordinal variables that were originally included in the model. I only included the first two factors in the linear model (alcohol consumption, alcohol, and parent's education, pedu), as when I ran the model with 4 factors, the other 2 factors had high p values. The model has a R squared of 0.371.

(Vifs)

```
    failures       higheryes        schoolMS    schoolsupyes           sexM
    1.183690        1.242418        1.229964        1.080581        1.215080
romanticyes    Fjobteacher reasonreputation     internetyes         alcohol
    1.065589        1.164819        1.047460        1.147783        1.159872
       pedu
    1.404281
```





I checked for multicollinearity of the data in the linear regression, and it didn't look like there was any multicollinearity, as the vif values for all the variables were well below 10. The corrplot from earlier doesn't show much correlation between most of the variables. Looking also at the graph of

the residuals, with the Cook's distance to check for outliers, it doesn't look like there are any outliers. The residuals also look like they have equal variance and look mostly normally distributed.

I cross validated all the models by applying my models on the training set to predict the grade on the test data. I then calculated the RMSE for the predictions. The RMSE for the ridge regression model was at 2.480, RMSE for lasso at 2.471, RMSE for elastic was at 2.381, and RMSE for OLS was at 2.474, and the RMSE for the OLS with factors was at 2.453. The RMSE values were very similar, with the RMSE for the elastic net regression model being the smallest.

Conclusion

Some conclusions that I took away from the analysis are that: elastic net regression seems to be the best model in predicting grade. It has the highest R squared, lowest MSE, also uses model selection, so it's more parsimonious than a typical linear regression model and avoids multicollinearity.

Looking at the models overall, it seems like it's hard to predict student grades, as the R squared for all the models are pretty low. Although there are a lot of variables in the dataset, the data is mostly self-reported, so we don't know the student personally. We can't see what their daily lives are like and what their study habits are, so we can't necessarily get a good idea of how good of a student someone is.

For the individual variables in the factor analysis, I noticed that the goout variable is in both factor 1 and 3. I think goout could imply two different things. For the first factor, people who have a bad relationship go out with their friends to drink and avoid their domestic problems. In the third component, people who go out with their friends because they have more freetime. More freetime allows them to spend time with their family and so they have a good relationship with their family as well.

Looking at the factor plot, I think that the schools are in a good neighborhood. On the PCA plot, parent's education and traveltime are on the opposite sides of the x axis, so students whose parents have better education can travel less time to school. Better educated people tend to make more money, so I can assume that the people who live closer to the school make more money.

Even though the regression models weren't super accurate, we can still identify which variables are significant in predicting student performance and how they are related to each other. A benefit of this analysis is that teachers who see studies like this can better understand their students' personal situations and can try to tailor their teaching style depending on the student in order to improve their performances. They can do a better job in identifying what their students need help in and how they need help and can help offer the right resources to help their students improve their grades.

Appendix

Ridge regression model coefficients:

```
> coef(scoreridge, s="lambda.min")
40 x 1 sparse Matrix of class "dgCMatrix"
                             1
(Intercept)       9.917268019
schoolMS         -0.607421721
sexM             -0.336796364
age               0.010817924
addressU          0.216567652
famsizeLE3       -0.037299014
PstatusT         -0.006488482
Medu              0.150855036
Fedu              0.084286980
Mjobhealth        0.173438029
Mjobother         0.156597852
Mjobservices     -0.086837390
Mjobteacher       0.286538207
Fjobhealth       -0.359283606
Fjobother         0.046744908
Fjobservices     -0.103228641
Fjobteacher       0.805048896
reasonhome       -0.140547926
reasonother      -0.305154287
reasonreputation  0.537295588
guardianmother   -0.131917170
guardianother    -0.248418085
traveltime        0.002032649
studytime         0.224136009
failures         -0.854604282
schoolsupyes     -0.932716020
famsupyes         0.126149990
paidyes           0.115929867
activitiesyes     0.143546661
nurseryyes        0.172646626
higheryes         1.286148559
internetyes       0.160311530
romanticyes      -0.086234704
famrel            0.197074278
freetime         -0.089565684
goout            -0.008253973
Dalc             -0.086613659
Walc             -0.120248470
health           -0.114985601
absences         -0.025985587
```

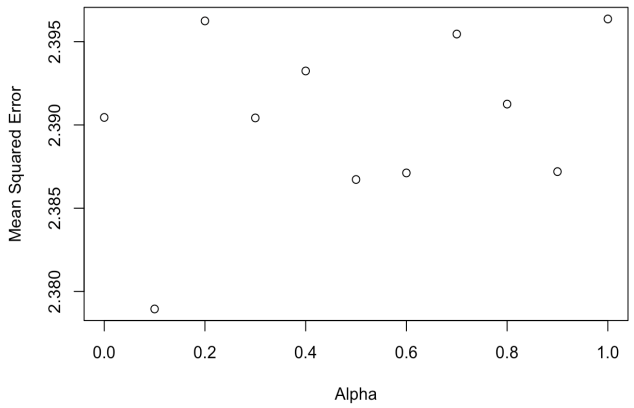
Lasso regression model coefficients:

```
40 x 1 sparse Matrix of class "dgCMatrix"
                            1
(Intercept)      10.13187766
schoolMS         -0.74278097
sexM             -0.24465558
age               .
addressU          .
famsizeLE3        .
PstatusT          .
Medu              0.20447847
Fedu              .
Mjobhealth        .
Mjobother         .
Mjobservices      .
Mjobteacher       .
Fjobhealth        .
Fjobother         .
Fjobservices      .
Fjobteacher       0.68449211
reasonhome        .
reasonother      -0.01408619
reasonreputation  0.58420448
guardianmother    .
guardianother     .
traveltime        .
studytime         0.19166691
failures         -1.18146009
schoolsupyes     -1.08350920
famsupyes         .
paidyes           .
activitiesyes     .
nurseryyes        .
higheryes         1.61897387
internetyes       .
romanticyes       .
famrel            0.13729237
freetime          .
goout             .
Dalc              .
Walc             -0.17749986
health           -0.06750712
absences         -0.01170248
```

Elastic net model coefficients:

```
> coef(fitElastic, s="lambda.min")
40 x 1 sparse Matrix of class "dgCMatrix"
                             1
(Intercept)       9.452725548
schoolMS         -0.786675567
sexM             -0.326056255
age               .
addressU          0.388286691
famsizeLE3        0.095654878
PstatusT          .
Medu              0.083127980
Fedu              0.209922619
Mjobhealth        0.360312083
Mjobother         .
Mjobservices      .
Mjobteacher       0.443931424
Fjobhealth        .
Fjobother         .
Fjobservices     -0.068068965
Fjobteacher       0.101095733
reasonhome        0.335548659
reasonother       .
reasonreputation  0.542550436
guardianmother   -0.027674239
guardianother     .
traveltime        0.009314332
studytime         0.409152595
failures         -1.009816150
schoolsupyes     -0.647356776
famsupyes         .
paidyes          -0.321729675
activitiesyes     0.234158314
nurseryyes        .
higheryes         1.067663073
internetyes       .
romanticyes       0.045574909
famrel            0.124805556
freetime         -0.050761217
goout             .
Dalc             -0.292481126
Walc              .
health           -0.007720745
absences         -0.038349719
```

Plot of alphas with elastic net errors:



Spearman Correlations:

```
> cS
            traveltime   studytime      famrel      freetime       goout         Dalc
traveltime  1.000000000 -0.08938687 -0.02564921 -0.0010490721  0.04071371  0.068463048
studytime  -0.089386875  1.00000000  0.01936969 -0.0764963990 -0.08231785 -0.171309395
famrel     -0.025649208  0.01936969  1.00000000  0.1441234575  0.08777530 -0.097529297
freetime   -0.001049072 -0.07649640  0.14412346  1.0000000000  0.35434528  0.127171556
goout       0.040713715 -0.08231785  0.08777530  0.3543452808  1.00000000  0.233976581
Dalc        0.068463048 -0.17130940 -0.09752930  0.1271715555  0.23397658  1.000000000
Walc        0.031516653 -0.22208798 -0.10203267  0.1201479753  0.37245468  0.613056109
health     -0.063843537 -0.07673230  0.09254227  0.0951053627 -0.01212210  0.084946472
Medu       -0.263288942  0.09841537  0.02508650 -0.0278945937  0.01020522  0.001961144
Fedu       -0.222034367  0.06908043  0.02128390 -0.0001513686  0.02878660 -0.004897390
                Walc      health         Medu          Fedu
traveltime  0.03151665 -0.06384354 -0.263288942 -0.2220343667
studytime  -0.22208798 -0.07673230  0.098415369  0.0690804254
famrel     -0.10203267  0.09254227  0.025086504  0.0212839012
freetime    0.12014798  0.09510536 -0.027894594 -0.0001513686
goout       0.37245468 -0.01212210  0.010205217  0.0287865971
Dalc        0.61305611  0.08494647  0.001961144 -0.0048973897
Walc        1.00000000  0.11428202 -0.018234051  0.0297257057
health      0.11428202  1.00000000  0.016111582  0.0463507580
Medu       -0.01823405  0.01611158  1.000000000  0.6471941646
Fedu        0.02972571  0.04635076  0.647194165  1.0000000000
```

Correlation Significances for Spearman:

```
Probability values (Entries above the diagonal are adjusted for multiple tests.)
           traveltime studytime famrel freetime goout Dalc Walc health Medu Fedu
traveltime       0.00      0.51   0.72     0.88  0.98 0.83 0.92   0.60 0.05 0.07
studytime        0.51      0.00   0.99     0.36  0.25 0.11 0.07   0.48 0.64 0.75
famrel           0.72      0.99   0.00     0.76  0.83 0.23 0.24   0.93 0.80 0.76
freetime         0.88      0.36   0.76     0.00  0.15 0.80 0.74   0.99 0.41 0.45
goout            0.98      0.25   0.83     0.15  0.00 0.31 0.13   0.58 0.44 0.48
Dalc             0.83      0.11   0.23     0.80  0.31 0.00 0.00   0.98 0.46 0.47
Walc             0.92      0.07   0.24     0.74  0.13 0.00 0.00   0.91 0.45 0.53
health           0.60      0.48   0.93     0.99  0.58 0.98 0.91   0.00 0.75 0.81
Medu             0.05      0.64   0.80     0.41  0.44 0.46 0.45   0.75 0.00 0.00
Fedu             0.07      0.75   0.76     0.45  0.48 0.47 0.53   0.81 0.00 0.00
```

Common factor analysis with 5 factors

```
Loadings:
           Factor1 Factor2 Factor3 Factor4 Factor5
Medu        0.916
Fedu        0.706
Walc                0.946
Dalc                        0.915
goout                               0.767
traveltime
studytime
famrel
freetime                            0.461
health                                      0.462

               Factor1 Factor2 Factor3 Factor4 Factor5
SS loadings      1.439   1.185   0.929   0.885   0.344
Proportion Var   0.144   0.118   0.093   0.089   0.034
Cumulative Var   0.144   0.262   0.355   0.444   0.478
```

OLS with 4 Factors from CFA

```
Call:
lm(formula = scoretrainy ~ failures + higheryes + schoolMS +
    schoolsupyes + sexM + romanticyes + Fjobteacher + reasonreputation +
    internetyes + alcohol + pedu + Factor3 + Factor4, data = scoretrainframe2)

Residuals:
     Min       1Q   Median       3Q      Max
-10.5917  -1.4468  -0.0788   1.4087   5.2841

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      10.9632     0.5796  18.917  < 2e-16 ***
failures         -1.4742     0.2294  -6.425 4.95e-10 ***
higheryes         1.7887     0.4810   3.719 0.000237 ***
schoolMS         -0.9729     0.2885  -3.372 0.000842 ***
schoolsupyes     -1.3701     0.4083  -3.356 0.000891 ***
sexM             -0.4895     0.2835  -1.727 0.085199 .
romanticyes      -0.4714     0.2668  -1.767 0.078195 .
Fjobteacher       1.0556     0.5368   1.966 0.050141 .
reasonreputation  0.2172     0.3135   0.693 0.488949
internetyes       0.2570     0.3184   0.807 0.420216
alcohol          -0.3498     0.1427  -2.452 0.014759 *
pedu              0.3391     0.1639   2.068 0.039457 *
Factor3          -0.1025     0.1281  -0.800 0.424131
Factor4          -0.2803     0.2178  -1.287 0.198931
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.248 on 310 degrees of freedom
Multiple R-squared:  0.3773,    Adjusted R-squared:  0.3612
F-statistic: 14.45 on 13 and 310 DF,  p-value: < 2.2e-16
```

R code:

```
#reading data
library(glmnet)
studentpor <- read.csv("~/Documents/19-20/CSC 424/student/student-por.csv", sep=";")
library(psych)
#finding mean score
studentpor$meanscore = (studentpor$G1 + studentpor$G2 + studentpor$G3 )/3
studentpor2 =studentpor[c(-31, -32,-33)]

set.seed(123)
library(glmnet)
#split data
scoremmatrix = model.matrix(~., data=studentpor2)
scoremmatrix = scoremmatrix[,c(-1)]

n = nrow(scoremmatrix)
s = sample(n, n/2)
scoretrainx = scoremmatrix[s, ]
scoretrainy = scoretrainx[,c(40)]
scoretrainx = scoretrainx[,c(-40)]

scoretestx = scoremmatrix[-s, ]
scoretesty = scoretestx[,c(40)]
scoretestx = scoretestx[,c(-40)]

#ridge regression
scoreridge = cv.glmnet(scoretrainx, scoretrainy, alpha=0)
plot(scoreridge$cvm)
scoreridge$lambda.min
plot(scoreridge)

ridgepredscoretest = predict(scoreridge, scoretestx, s="lambda.min")
rmsesscoretest = sqrt(mean((ridgepredscoretest - scoretesty)^2))
rmsesscoretest

#lasso
scorelasso = cv.glmnet(scoretrainx, scoretrainy, alpha = 1)
plot(scorelasso)
print(scorelasso$lambda.min)
log(scorelasso$lambda.min)
coef(scorelasso, s="lambda.min")
glmnet(scoretrainx, scoretrainy, alpha=1,  lambda=scorelasso$lambda.min)
```

```
plot(scorelasso$glmnet.fit, "lambda", label = TRUE)

lassopredscoretest = predict(scorelasso, scoretestx, s="lambda.min")
rmsesscoretestlasso = sqrt(mean((lassopredscoretest - scoretesty)^2))
rmsesscoretestlasso

#elastic net
alphaBest = 0
bestError = 9999999    # Start out with a huge error
alph = seq(0, 1, .1)
mylist = c()
for (alpha in seq(0, 1, .1))
{
  meanError = 0
  for (i in 1:100)
  {
    # Grab test and training sets
    n = nrow(scoremmatrix)
    s = sample(n, n/2)
    scoreTrain = scoremmatrix[s, ]
    scoreTest = scoremmatrix[-s, ]

    xTrain = as.matrix(scoreTrain[, -40])
    yTrain = as.matrix(scoreTrain[, 40])
    xTest = as.matrix(scoreTest[, -40])
    yTest = as.matrix(scoreTest[, 40])

    fitElastic = cv.glmnet(xTrain, yTrain, alpha=alpha, nfolds=7)
    elasticPred = predict(fitElastic, xTest, s="lambda.min")
    meanError = meanError + sqrt(mean((elasticPred - yTest)^2))
  }
  meanError = meanError / 100

  print(meanError)

  mylist = c(mylist, meanError)
  if (meanError < bestError)
  {
    alphaBest = alpha
    bestError = meanError
  }
}
```

```
alphaBest
glmnet(scoretrainx, scoretrainy, alpha=0.6,  lambda=fitElastic$lambda.min)
fitElastic$lambda.min
log(fitElastic$lambda.min)
fitElastic = cv.glmnet(scoretrainx, scoretrainy, alpha=0.6, nfolds=7)
coef(fitElastic, s="lambda.min")
elasticpred = predict(fitElastic, scoretestx, s="lambda.min")
rmseselastic = sqrt(mean((elasticpred - scoretesty)^2))
rmseselastic
plot(fitElastic$glmnet.fit, "lambda", label = TRUE)

#ols
scoretrainframe = as.data.frame(scoretrainx)
scoretestframe = as.data.frame(scoretestx)

scorelm = lm(scoretrainy ~. , data = scoretrainframe)
summary(scorelm)

scorelm2 <- lm(scoretrainy~ 1, data = scoretrainframe)

scoreforward <- step(scorelm2, scope = list(upper = scorelm, lower = scorelm2), direction =
"forward", trace = F  )
scoreforward
summary(scoreforward)

scorelm = lm(scoretrainy ~. , data = scoretrainframe)
scorebackward <- step(scorelm, direction = "backward", trace = F)
summary(scorebackward)

scoreselect = lm(scoretrainy ~  failures + studytime + Medu + higheryes +
   schoolMS + schoolsupyes + Fjobteacher + freetime + absences +
   sexM + romanticyes + age, data = scoretrainframe)

summary(scoreselect)

olspred = predict(scoreforward, scoretestframe)
rmseols = sqrt(mean((olspred - scoretesty)^2))
rmseols

vif(scorelm)

#PCA
```

```
studentpor3 =studentpor[c("traveltime", "studytime", "famrel", "freetime", "goout", "Dalc", "Walc",
"health", "Medu", "Fedu")]

library(corrplot)

cS = cor(studentpor3, method="spearman")
corrplot(cS, method="ellipse")
ps = prcomp(cS)
summary(ps)
screeplot(ps)
studenttraine = scoreTrain[c("schoolMS", " addresssU", "famsizeLE3", "Fedu", "Mjobhealth",
                    "Mjobteacher","Fjobteacher", "reasonreputation", "studytime",
                    "failures", "schoolsupyes", "nurseryyes", "higheryes",
                    "romaticyes", "freetime", "Walc,", "absences")]

studentteste = scoreTest[c("schoolMS", " addresssU", "famsizeLE3", "Fedu", "Mjobhealth",
                    "Mjobteacher","Fjobteacher", "reasonreputation", "studytime",
                    "failures", "schoolsupyes", "nurseryyes", "higheryes",
                    "romaticyes", "freetime", "Walc,", "absences")]

straino = scoretrainx[,c("traveltime", "studytime", "famrel", "freetime", "goout", "Dalc", "Walc",
"health", "Medu", "Fedu")]

stesto = scoretestx[,c("traveltime", "studytime", "famrel", "freetime", "goout", "Dalc", "Walc",
"health", "Medu", "Fedu")]

sfactr <- factanal(straino, factors=4, rotation="varimax", scores="regression")
print(sfactr$loadings, cutoff = 0.4, sort = TRUE)
sfactr
sfactr

scores = as.data.frame(sfactr$scores)

sfacte <- factanal(stesto, factors=4, rotation="varimax", scores="regression")
print(sfacte$loadings, cutoff = 0.4, sort = TRUE)
sfactr
scoreste = as.data.frame(sfacte$scores)

corrTests = corr.test(cS, adjust = "none")
corrTests

source("/Users/davidguo/Documents/19-20/CSC 424/HW3/PCA_Plot.R")
```

```
pf = principal(cS, rotate="varimax", nfactors=5)
print(pf$loadings, cutoff = 0.4, sort = TRUE)
pf
PCA_Plot_Psyc(pf)
pf$values
pf$communality
pf$rot.mat
head(pf$scores)
summary(pf)

sfact <- factanal(studentpor3, factors=4)
print(sfact$loadings, cutoff = 0.4, sort = TRUE)
sfact

scores = as.data.frame(sfact$scores)
scores$grade = studentpor$meanscore

fitpca = lm(grade ~ ., data=scores)
summary(fitpca)

factanal(cS,  factors = 5)

scoretrainframe2 = cbind(scoretrainframe, scores)

scoretrainframe2$alcohol = scoretrainframe2$Factor1
scoretrainframe2$pedu = scoretrainframe2$Factor2
scoretestframe2$alcohol = scoretestframe2$Factor1
scoretestframe2$pedu = scoretestframe2$Factor2

#regression for factor analysis
lmfact = lm(scoretrainy ~ failures + higheryes + schoolMS +
   schoolsupyes + sexM + romanticyes + Fjobteacher +
   reasonreputation + internetyes + alcohol + pedu , data = scoretrainframe2)
plotres(lmfact)
summary(lmfact)
plotres(lmfact)
scoretestframe2 = cbind(scoretestframe, scoreste)

factpred = predict(lmfact, scoretestframe2)
rmsefact = sqrt(mean((factpred - scoretesty)^2))
rmsefact
```