Testing the Current Success of a Test Preparation Course and Giving Insights for Future Success

Noah Alexander, Nathan Acosta, Joshua Duazo, Ali Zain Charolia, Sahran Prasla, and Toan Nguyen

2022-12-01

Introduction

The dataset that we have chose to work with is labeled as "Students Performance in Exams" and provided from Kaggle. This dataset contains the socioeconomic and demographic data of 1000 students from a certain district as well as whether a student completed a specific test preparation course or not. The demographic data of each student includes variables such as gender, race, parental level of education, and lunch type.

`Description of Variables`

- Column 1: Gender; Male or Female
- Column 2: Race/Ethnicity; Represented as group letters for fairness
- Column 3: Parental Level of Education; some high school, high school, some college, bachelor's degree, or master's degree
- Column 4: Lunch; Standard or Free/Reduced
- Column 5: Test Preparation Course; None or Completed
- Column 6: Math Score; numerical value between 0-100
- Column 7: Reading Score; numerical value between 0-100
- Column 8: Writing Score; numerical value between 0-100

With this data we are going to answer the given questions using different data modeling techniques:

(1) How successful is the test preparation course?
(2) What groups of students should the course be marketed to?

How Successful is the Test Preparation Course?

We will be looking at two different models and the predictors which are statistically significant to them. The first is a logistic regression model which will predict whether or not the student took the preparation course. The second model is a decision tree which will also predict whether or not the student took the preparation course. By using different models, we hope to affirm that certain predictors are heavily correlated. Specifically, we would like to see that higher test scores are correlated to students taking the test preparation course. We will now construct the models and draw conclusions from them in an effort to ascertain how successful the test preparation course really is.

1 Logistic Regression Model

1.1 The Model Equation and Background

We will construct a logistic regression model with the aim of predicting whether or not a student has completed the test preparation course using all predictor variables. The equation for this is

$$log(p(x)/(1-p(x))) = B_0+B_1 \ast gender+B_2 \ast race.ethnicity.B+B_3 \ast race.ethnicity.C+B_4 \ast race.ethnicity.D+ B_5 \ast race.ethnicity.E+B_6 \ast parental.level.of.education.bachelors+B_7 \ast parental.level.of.education.highschool+ B_8 \ast parental.level.of.education.masters + B_9 \ast parental.level.of.education.some.college + B_1 0 \ast parental.level.of.education.some.highschool + B_1 1 \ast lunch + B_1 2 \ast math.score + B_1 3 \ast reading.score + B_1 4 \ast writing.score$$

Where $B_1 = \{0 \text{ if female, } 1 \text{ if male}\}$,
$B_{11} = \{0 \text{ if reduced, } 1 \text{ if standard}\}$, and
$B_2, B_3, ..., B_{10} = \{0 \text{ if not, } 1 \text{ if so}\}$

For the equation above, the output is the likelihood of the student taking the test preparation course. From this model, we will look at which predictors are the most statistically significant and which are not and therefore need to be dropped from the model. We chose to use a logistic regression model because our response is a binary qualitative output. Another reason we are using this model is because this model has good interpretability as opposed to other, more complex models. One of the downsides; however, is that we cannot tell exactly how one variable influences the response. One of the drawbacks of this model is that we cannot see the incremental increase in likelihood given a unit increase in any of the predictors. Rather, we can only say that one predictor makes the outcome more or less likely as it grows. One thing we will want to be aware of is overfitting our data. However, by using an 80% training and 20% testing split on the dataset we can avoid this.

1.2 Building the Logistic Regression Model

When we build the model, we should not automatically assume every predictor in the dataset is statistically significant. We will use the step function, which begins with all the predictors in the model and systematically takes away and/or adds predictors to the model to make it better. Specifically, it looks to minimize the AIC and maximize the adjusted Rˆ2. The AIC is an estimator of the relative quality of statistical models for a given set of data given a collection of models (here is the same model with different predictors).

```
exams.glm = glm(test.preparation.course~ .,
                data = exams,
                family = "binomial")

summary(exams.glm)
```

```
##
## Call:
## glm(formula = test.preparation.course ~ ., family = "binomial",
##     data = exams)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8380  -0.7812   0.3920   0.7567   2.2468
##
## Coefficients:
##                                            Estimate Std. Error z value
## (Intercept)                                 5.36024    0.60393   8.876
## gendermale                                 -2.57485    0.26542  -9.701
## race.ethnicitygroup B                       0.26902    0.32285   0.833
## race.ethnicitygroup C                       0.48543    0.30734   1.579
## race.ethnicitygroup D                       1.19350    0.31993   3.731
## race.ethnicitygroup E                      -0.01737    0.35973  -0.048
## parental.level.of.educationbachelor’s degree 0.54470    0.29614   1.839
## parental.level.of.educationhigh school     -0.18849    0.25893  -0.728
## parental.level.of.educationmaster’s degree  1.12011    0.36008   3.111
## parental.level.of.educationsome college    -0.19055    0.24121  -0.790
## parental.level.of.educationsome high school -0.47114    0.26063  -1.808
## lunchstandard                               0.14947    0.18958   0.788
## math.score                                  0.13846    0.01497   9.251
## reading.score                               0.11389    0.02045   5.570
## writing.score                              -0.30538    0.02482 -12.305
##                                            Pr(>|z|)
## (Intercept)                                 < 2e-16 ***
## gendermale                                  < 2e-16 ***
## race.ethnicitygroup B                      0.404698
## race.ethnicitygroup C                      0.114234
## race.ethnicitygroup D                        0.000191 ***
## race.ethnicitygroup E                        0.961478
## parental.level.of.educationbachelor’s degree 0.065871 .
## parental.level.of.educationhigh school       0.466632
## parental.level.of.educationmaster’s degree   0.001866 **
## parental.level.of.educationsome college      0.429549
## parental.level.of.educationsome high school  0.070656 .
## lunchstandard                                0.430462
## math.score                                    < 2e-16 ***
## reading.score                                2.55e-08 ***
## writing.score                                 < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1275.33  on 999  degrees of freedom
## Residual deviance:  959.37  on 985  degrees of freedom
## AIC: 989.37
##
## Number of Fisher Scoring iterations: 5
```

```
table = step(exams.glm)
```

```
## Start:  AIC=989.37
## test.preparation.course ~ gender + race.ethnicity + parental.level.of.education +
##     lunch + math.score + reading.score + writing.score
##
##                               Df Deviance     AIC
## - lunch                        1   959.99  987.99
## <none>                             959.37  989.37
## - parental.level.of.education  5   985.16 1005.16
## - race.ethnicity               4   988.26 1010.26
## - reading.score                1   992.26 1020.26
## - math.score                   1  1059.16 1087.16
## - gender                       1  1073.30 1101.30
## - writing.score                1  1169.54 1197.54
##
## Step:  AIC=987.99
## test.preparation.course ~ gender + race.ethnicity + parental.level.of.education +
##     math.score + reading.score + writing.score
##
##                               Df Deviance     AIC
## <none>                             959.99  987.99
## - parental.level.of.education  5   985.32 1003.32
## - race.ethnicity               4   989.59 1009.59
## - reading.score                1   992.27 1018.27
## - math.score                   1  1073.15 1099.15
## - gender                       1  1078.01 1104.01
## - writing.score                1  1169.58 1195.58
```

We see that there are several predictors being used that are not important to the model. However, there are some predictors we can not get rid of. For example, we see that race.ethnicitygroup E is not statistically significant, but this is a dummy variable for the overall predictors of race.ethnicitygroup which does have dummy variables that are statistically significant comprising it. There *is* one predictor that is not statistically significant to the model though. We can see that lunchstandard is not statistically significant because it has a p-value of .4305. Furthermore, using the step function we see that the decision to drop that predictor from the model is backed up. Therefore, we will not include lunch as a predictor variable in our logistic regression model.

With these changes in mind, let us perform cross validation through the 80% training and 20% testing sets we mentioned earlier.

```
set.seed(234)
test_error = matrix(nrow=10, ncol=1)
for (i in 1:10) {
  sample = sample.int(n = nrow(exams),
                    size = floor(.8*nrow(exams)),
                    replace = FALSE)
  train = exams[sample,]
  test = exams[-sample,]
  exams.glm3 = glm(test.preparation.course~ .-lunch,
                data = train,
                family = "binomial")
  exams.pred = predict.glm(exams.glm3,newdata = test,type = "response")
  yhat = ifelse(exams.pred < 0.5, 'completed', 'none')
  conf.test = table(test$test.preparation.course,yhat)
  test_error[i] = (conf.test[1,2] + conf.test[2,1])/200
}
mean(test_error)
```

```
## [1] 0.2515
```

We get an average test error rate of 25.15% for the logistic regression model. However, since our focus is on inference based on predictor significance, we do not need to worry about this relatively high error rate.

1.3 Results

```
exams.glm2 = glm(test.preparation.course~ .-lunch,
                data = exams,
                family = "binomial")
summary(exams.glm2)
```

```
##
## Call:
## glm(formula = test.preparation.course ~ . - lunch, family = "binomial",
##     data = exams)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8565  -0.7845   0.3961   0.7658   2.2058
##
## Coefficients:
##                                             Estimate Std. Error z value
## (Intercept)                                  5.34937    0.60359   8.863
## gendermale                                  -2.59589    0.26399  -9.833
## race.ethnicitygroup B                        0.27075    0.32302   0.838
## race.ethnicitygroup C                        0.48747    0.30755   1.585
## race.ethnicitygroup D                        1.18910    0.32016   3.714
## race.ethnicitygroup E                       -0.05916    0.35580  -0.166
## parental.level.of.educationbachelor’s degree  0.54890    0.29642   1.852
## parental.level.of.educationhigh school      -0.17594    0.25830  -0.681
## parental.level.of.educationmaster’s degree   1.11140    0.36002   3.087
## parental.level.of.educationsome college     -0.19025    0.24101  -0.789
## parental.level.of.educationsome high school -0.45575    0.25980  -1.754
## math.score                                   0.14137    0.01452   9.739
## reading.score                                0.11140    0.02018   5.521
## writing.score                               -0.30396    0.02470 -12.304
##                                             Pr(>|z|)
## (Intercept)                                  < 2e-16 ***
## gendermale                                   < 2e-16 ***
## race.ethnicitygroup B                       0.401921
## race.ethnicitygroup C                       0.112959
## race.ethnicitygroup D                       0.000204 ***
## race.ethnicitygroup E                       0.867952
## parental.level.of.educationbachelor’s degree 0.064058 .
## parental.level.of.educationhigh school      0.495790
## parental.level.of.educationmaster’s degree  0.002021 **
## parental.level.of.educationsome college     0.429893
## parental.level.of.educationsome high school 0.079391 .
## math.score                                   < 2e-16 ***
## reading.score                               3.38e-08 ***
## writing.score                                < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1275.33  on 999  degrees of freedom
## Residual deviance:  959.99  on 986  degrees of freedom
## AIC: 987.99
##
## Number of Fisher Scoring iterations: 5
```

Now we must answer the question: Was the course successful in raising scores for those who took it? From looking at the summary of the model, we can see that as math and reading scores increase so does the likelihood of the student taking the test preparation course. What is really interesting is that students with a higher writing score are less likely to take the test preparation course. For one, we can say that there is a correlation between higher test scores in math and reading and students who take the test preparation course; however, we cannot say the same is true for writing. Is this a shortcoming of the course? I do not think we can tell simply from what we have collected, because we do not have enough data to prove causation. What if this just shows that students who are more likely to have lower writing scores are more likely to seek test prep for lack of confidence? We now know that the test preparation course helps to improve math and reading scores, but we should seek to improve the course to help improve writing scores.

2 Decision Tree Model

2.1 The Model Equation and Background

We will construct a decision tree model with the aim of predicting whether or not a student has completed the test preparation course using all predictor variables:

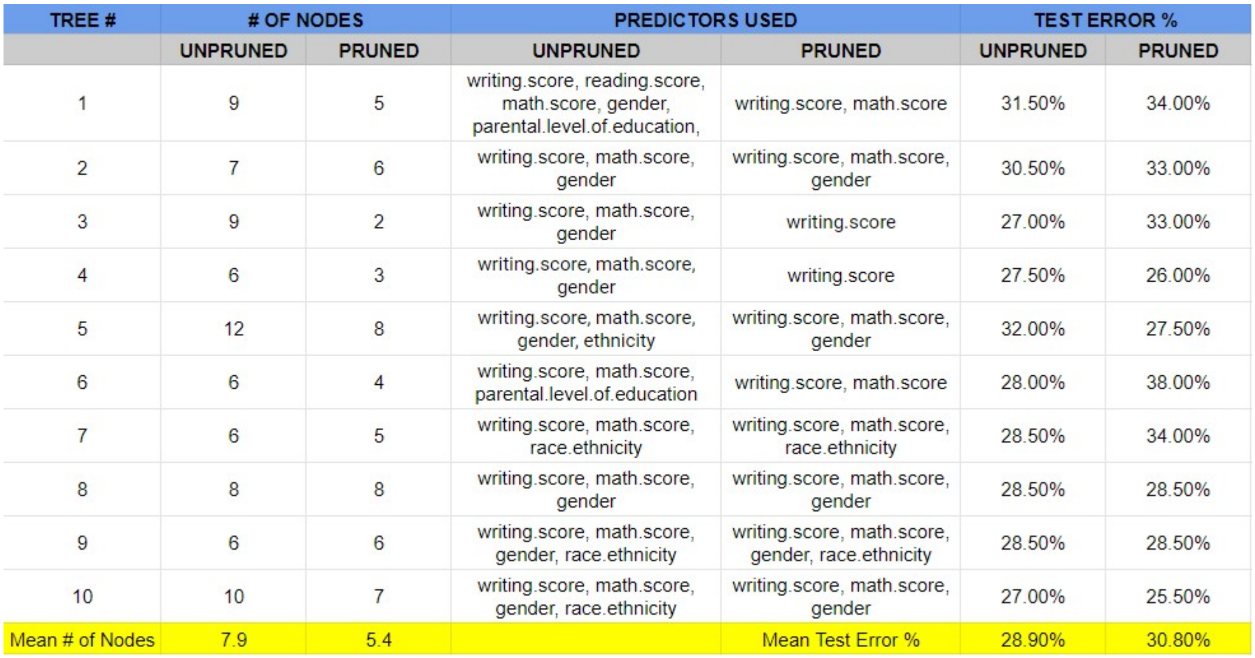
$prep\ \ gender + race.ethnicity + parental.level.of.education + lunch + math.score + reading.score + writing.score$

From this model, we will make inferences on what predictors are the most important to model accuracy and therefore are the most important in predicting who has taken the test preparation course. This is the same objective we had with the logistic regression model, and now we will see if a decision tree model can give similar or more insightful results. Decision tree models have high interpretability at the cost of high variability, a risk of overfitting, and do not possess as high of a prediction accuracy as with other models. However, via pruning we may be able to reduce the drawback of high variance and overfitting within the tree models. Using decision trees will allow us to better understand the relationship between the response and predictor variables. The code below was used to create 10 decision tree models, each of which were pruned in accordance with their CV plots. For simplicity, a table was created which lists the number of terminal nodes, predictors, and test error rate for each of the pruned and unpruned models generated below.

2.2 Building and Pruning the Decision Trees

```r
set.seed(234)
test_accuracy = c()

for (i in c(1:10)) {

  sample = sample.int(n=nrow(exams),
                    size = floor(0.8*nrow(exams)),
                    replace = FALSE)

  train = exams[sample,]
  test = exams[-sample,]

  tree.prep = tree(test.preparation.course~ ., data = train)

  cv.prep = cv.tree(tree.prep)
  min_cv = min(cv.prep$dev)
  dev_index = match(min_cv, cv.prep$dev)
  size = cv.prep$size[dev_index]
  prune.prep = prune.tree(tree.prep, best = size)

  tree.pred = predict(prune.prep, test, type = "class")
  confusion_matrix = as.data.frame(table(tree.pred,test$test.preparation.course))
  total = sum(confusion_matrix$Freq)
  correct = sum(confusion_matrix[c(1,4),]$Freq)
  accuracy = correct/total

  test_accuracy = append(test_accuracy, accuracy)
}

x = c(1:10)
average_accuracy = mean(test_accuracy)
```

We begin the process by splitting the dataset into 80% training and 20% testing splits randomly and using the training subset to train our tree model. We then get the test error rate of the unpruned tree (and would graph it to look at the model using the plot() and text() functions). Then we prune the tree by looking for where the cv has a local depression (for all of these models, finding the minimum is sufficient). This local depression is the number of terminal nodes we need in our pruned tree, and using this we prune the original tree. Then we calculate the test error rate as we did before.

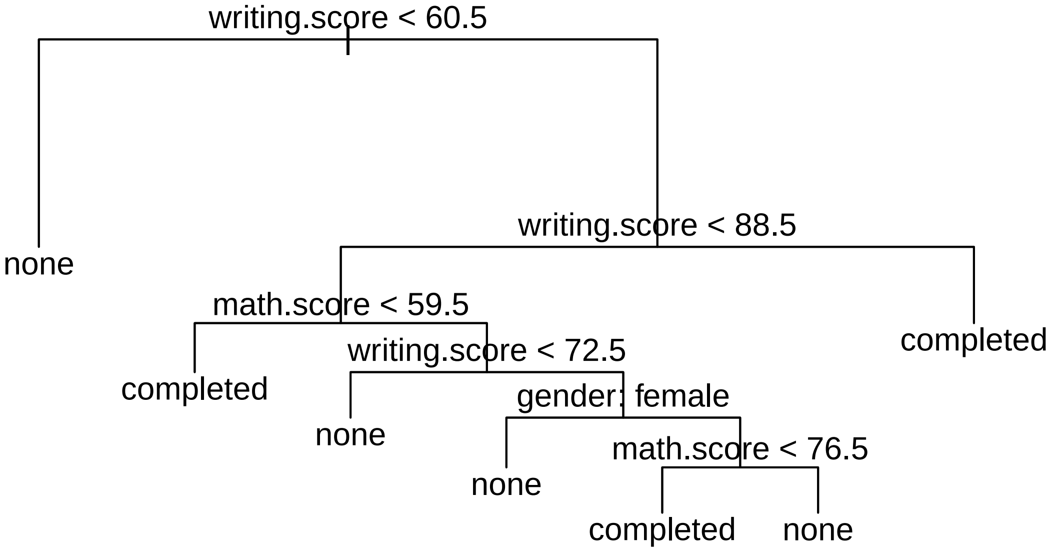


TREE #	# OF NODES		PREDICTORS USED		TEST ERROR %	
	UNPRUNED	PRUNED	UNPRUNED	PRUNED	UNPRUNED	PRUNED
1	9	5	writing.score, reading.score, math.score, gender, parental.level.of.education,	writing.score, math.score	31.50%	34.00%
2	7	6	writing.score, math.score, gender	writing.score, math.score, gender	30.50%	33.00%
3	9	2	writing.score, math.score, gender	writing.score	27.00%	33.00%
4	6	3	writing.score, math.score, gender	writing.score	27.50%	26.00%
5	12	8	writing.score, math.score, gender, ethnicity	writing.score, math.score, gender	32.00%	27.50%
6	6	4	writing.score, math.score, parental.level.of.education	writing.score, math.score	28.00%	38.00%
7	6	5	writing.score, math.score, race.ethnicity	writing.score, math.score, race.ethnicity	28.50%	34.00%
8	8	8	writing.score, math.score, gender	writing.score, math.score, gender	28.50%	28.50%
9	6	6	writing.score, math.score, gender, race.ethnicity	writing.score, math.score, gender, race.ethnicity	28.50%	28.50%
10	10	7	writing.score, math.score, gender, race.ethnicity	writing.score, math.score, gender	27.00%	25.50%
Mean # of Nodes	7.9	5.4		Mean Test Error %	28.90%	30.80%

From the table, we can see that the average number of terminal nodes for the pruned trees is 5.4, with a mean test error rate of 30.8%. This is much worse than the logistic regression model. Note that in the pruned models the most common predictors to be used are writing score, math score, and occasionally gender. This is an important difference from the logistic and linear regression models. For the other models, we decided to keep the gender and parental level of education predictors included because there was significance. However, the tree models used writing scores, math scores, and gender and excluded the parental level of education and lunch for the most part.

2.3 Results:

```
tree.exams = tree(test.preparation.course~ ., data = exams)
prune.exams = prune.tree(tree.exams, best = 7) #predetermined
plot(prune.exams)
text(prune.exams, pretty=0)
```



Does this model show that the test course is successful? This conclusion is harder to draw with a decision tree. From this specific model (remember, high variance means these can be subject to change based on what data we train on), we can see that students who have a writing score above 88.5 are predicted to have taken the test preparation course. This is conflicting with what the logistic regression model says. We also see that this model predicts students with writing scores between 60.5 and 88.5 and a math score less than 59.5 to have taken the prep course. This model does not give specific relationships as to how writing and math scores affect the likelihood of a student taking the course. However, we can conclude that writing and math scores *do* affect the likelihood of a student taking the course.

What Groups of Students Should the Course be Marketed To?

The purpose of this subsection of models is to answer the question: What groups of students should the course be marketed to? To answer this question, we will perform linear regression using the data given. We will split the entire process into three simple similar steps. This will help us create three different models, each predicting a different score. The following are the different scores being predicted:

1. Mathematics Score
2. Reading Score
3. Writing Score

Once we develop, build, and run the models, we will analyse the outputs and accordingly derive the conclusion. The following are the models developed to explore this question in the order given above:

1 Mathematics Scores Prediction Linear Regression Model

1.1 Linear Model [siginificant predictors, best subset selection]

Since our response variable (math.score) is quantitative, we will use Linear Regression as our model of choice to fetch the desired output. Once the response, data set, and predictors are set up, we can create the Linear Model as follows:

$$math.score = B_0 + B_1 * gender + B_2 * race.ethnicity.B + B_3 * race.ethnicity.C + B_4 * race.ethnicity.D + B_5 * race.ethnicity.E + B_6 * parental.level.of.education.bachelors + B_7 * parental.level.of.education.highschool + B_8 * parental.level.of.education.masters + B_9 * parental.level.of.education.some.college + B_10 * parental.level.of.education.some.highschool + B_11 * lunch + B_12 * test.preparation.course$$

Where $B_1 = \{0 \text{ if female, } 1 \text{ if male}\}$,
$B_{11} = \{0 \text{ if reduced, } 1 \text{ if standard}\}$,
$B_{12} = \{0 \text{ if completed, } 1 \text{ if not}\}$, and
$B_2, B_3, ..., B_{10} = \{0 \text{ if not, } 1 \text{ if so}\}$

```r
# create linear model and load it into a exams.lm variable
exams.lm <- lm(math.score ~ gender + race.ethnicity + parental.level.of.education
              + lunch + test.preparation.course, data = exams)
```

Next, we want to interpret which predictors (gender, race.ethnicity, parental.level.of.education, lunch, and test.preparation.course) are significant to predict the Mathematics Exam scores for a student. Based on the step() function below, we will be able to derive that easily to understand which predictors are significant. We find out that all predictors are significant in predicting Math scores for a given student. This is also true for the next two models. *For the sake of concision, the calculation will not be included in later models.*

```r
# use step() to show we shouldn't get rid of anything
table = step(exams.lm)
```

```
## Start:  AIC=5096.66
## math.score ~ gender + race.ethnicity + parental.level.of.education +
##     lunch + test.preparation.course
##
##                               Df Sum of Sq    RSS    AIC
## <none>                                     159279 5096.7
## - test.preparation.course      1      4442 163722 5122.2
## - gender                       1      8273 167553 5145.3
## - parental.level.of.education  5     12515 171794 5162.3
## - race.ethnicity               4     17430 176709 5192.5
## - lunch                        1     34383 193662 5290.1
```

1.2 Perform Cross Validation

Next, we perform cross validation on our model to estimate the performance of our Machine Learning Linear Regression Model. For this, we create a matrix and store it in the mse.cv variable. We then create a for loop that loops 10 times creating different models every iteration. We fetch MSE of each model that is built in every iteration of the loop, and finally we calculate the average MSE by averaging/calculating mean of the columns of means. We use an 80/20 training/test split for the cross validation.

```r
mse.cv = matrix(nrow = 10, ncol = 1)
set.seed(12)
# for loop to create multiple models
for (i in 1:10) {
  sample = sample.int(n = nrow(exams), size = floor(.8*nrow(exams)), replace = FALSE)
  train = exams[sample,]
  test = exams[-sample,]

  # Framing data for easier reusability
  exams = data.frame(exams, exams$math.score)
  # use lm() to create multiple models
  exam_data.lm <- lm(math.score ~ gender + race.ethnicity + parental.level.of.education
                    + lunch + test.preparation.course, data = train)

  # get the MSE for each model
  mse.cv[i,1] = mean((test$math.score - predict(exam_data.lm, test))^2)
}

# use colMeans() to find means of all mse.cv values
colMeans(mse.cv)
```

```
## [1] 165.0638
```

In this case, we get an MSE value of 165.0638. Since we are looking to make inferences based on the model, a low MSE is not vital to the analysis.

1.3 Results

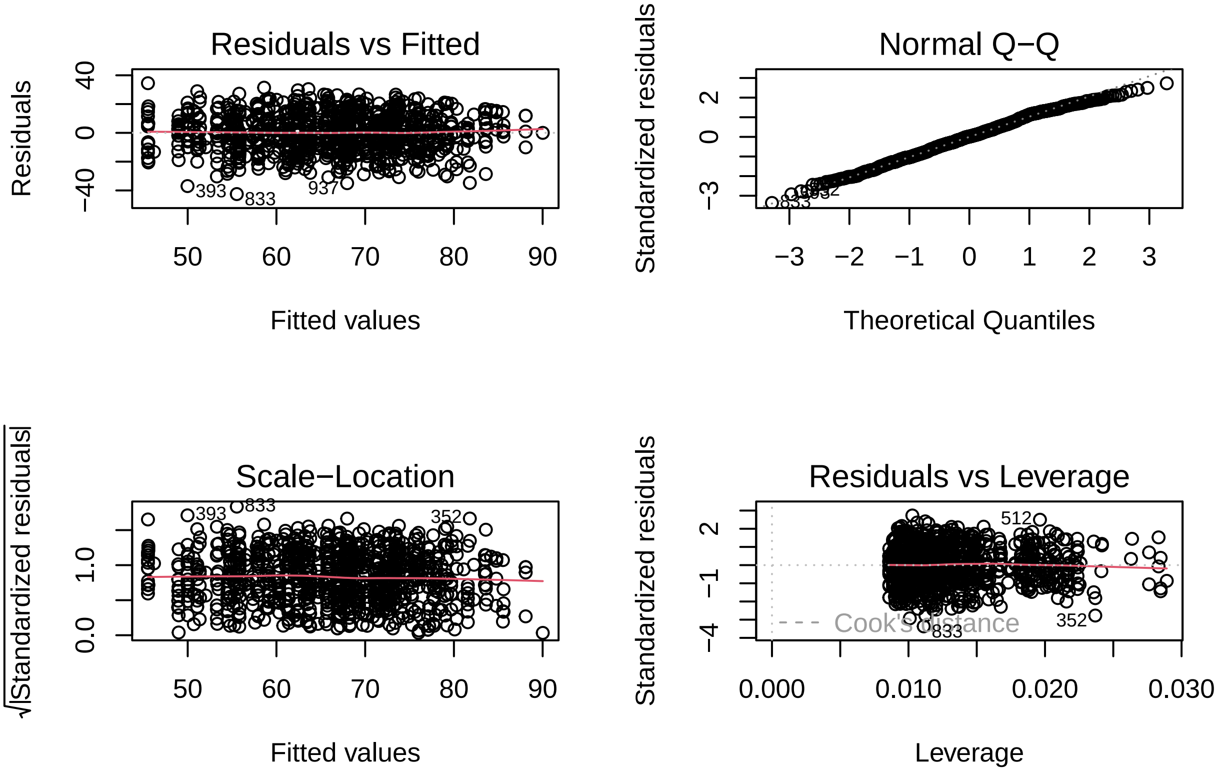
```
# finally, use summary to find which predictors are significant in predicting the
# response variable, math.scores
summary(exams.lm)
```

```
##
## Call:
## lm(formula = math.score ~ gender + race.ethnicity + parental.level.of.education +
##     lunch + test.preparation.course, data = exams)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -42.533  -8.809   0.240   8.770  34.478
##
## Coefficients:
##                                              Estimate Std. Error t value
## (Intercept)                                   58.4573     1.9069  30.656
## gendermale                                     5.8301     0.8142   7.160
## race.ethnicitygroup B                         -0.6369     1.6887  -0.377
## race.ethnicitygroup C                         -0.6802     1.6036  -0.424
## race.ethnicitygroup D                          4.8710     1.6377   2.974
## race.ethnicitygroup E                         11.3919     1.8198   6.260
## parental.level.of.educationbachelor’s degree   1.9270     1.5036   1.282
## parental.level.of.educationhigh school        -4.3507     1.2686  -3.430
## parental.level.of.educationmaster’s degree     4.0278     1.7749   2.269
## parental.level.of.educationsome college       -3.3067     1.2371  -2.673
## parental.level.of.educationsome high school   -7.7663     1.2884  -6.028
## lunchstandard                                 12.4076     0.8500  14.596
## test.preparation.coursenone                   -4.4887     0.8555  -5.247
##                                              Pr(>|t|)
## (Intercept)                                   < 2e-16 ***
## gendermale                                   1.57e-12 ***
## race.ethnicitygroup B                        0.706157
## race.ethnicitygroup C                        0.671554
## race.ethnicitygroup D                        0.003008 **
## race.ethnicitygroup E                        5.73e-10 ***
## parental.level.of.educationbachelor’s degree 0.200306
## parental.level.of.educationhigh school       0.000629 ***
## parental.level.of.educationmaster’s degree   0.023464 *
## parental.level.of.educationsome college      0.007643 **
## parental.level.of.educationsome high school  2.34e-09 ***
## lunchstandard                                 < 2e-16 ***
## test.preparation.coursenone                  1.89e-07 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 12.7 on 987 degrees of freedom
## Multiple R-squared:  0.328,  Adjusted R-squared:  0.3198
## F-statistic: 40.14 on 12 and 987 DF,  p-value: < 2.2e-16
```

When we print the summary of the linear model, we discover the following:

- Having Standard Lunch has the greatest positive impact on the Math Score of a student (Estimate: 12.4076).
- Being a student from race or ethnic group E (Estimate: 11.3919) also has a very great positive impact on the Math Score of a student.
- The greatest negative impact on Math Score of a student is caused by their parents’ education level being “some high school” (Estimate: -7.7663).
- A student with gender male, belonging to ethnic group D or E, parents having a bachelor’s degree, or master’s degree, and/or standard lunch will have an increased Mathematics score than other students as those predictors impact Math scores positively.
- A student belonging to ethnic group B or C, parents having high school education, or some college, or some high school education and/or not taking the test preparation course will have a decreased/lower Mathematics score as these predictors impact Math scores negatively.

```
# use par() to set parameters for multiple plots (2,2)
par(mfrow=c(2,2))

# use plot() to plot different graphs for the linear model [exams.lm]
plot(exams.lm)
```



Lastly, we plot out different diagnostic plots to analyze our Linear Model better and derive conclusions through it more visually. Through this, we find out that the model is Linear, Independent, and has a normal distribution. This is also true for the next two models. *For the sake of concision, this result will also not be calculated in later models.*

2 Reading Scores Prediction Linear Regression Model

2.1 Linear Model [siginificant predictors, best subset selection]

Since our response variable (reading.score) is quantitative, we will use Linear Regression as our model of choice to fetch the desired output. Once the response, data set, and predictors are set up, we can create the Linear Model as follows:

$reading.score = B_0 + B_1 * gender + B_2 * race.ethnicity.B + B_3 * race.ethnicity.C + B_4 * race.ethnicity.D + B_5 * race.ethnicity.E + B_6 * parental.level.of.education.bachelors + B_7 * parental.level.of.education.highschool + B_8 * parental.level.of.education.masters + B_9 * parental.level.of.education.some.college + B_10 * parental.level.of.education.some.highschool + B_11 * lunch + B_12 * test.preparation.course$

Where $B_1 = \{0 \text{ if female, } 1 \text{ if male}\}$,
$B_{11} = \{0 \text{ if reduced, } 1 \text{ if standard}\}$,
$B_{12} = \{0 \text{ if completed, } 1 \text{ if not}\}$, and
$B_2, B_3, ..., B_{10} = \{0 \text{ if not, } 1 \text{ if so}\}$

```
# create linear model and load it into a exams.lm variable
exams.lm <- lm(reading.score ~ gender + race.ethnicity + parental.level.of.education + lunch + test.preparation.course, data = exams)
```

Next, we want to interpret which predictors - gender, race.ethnicity, parental.level.of.education, lunch, and test.preparation.course are significant to predict the Reading Exam scores for a student. Based on the step()

function, we find out that all predictors are significant in predicting Reading scores for a given student. This was calculated in section 1.1.

2.2 Perform Cross Validation

Next, we perform cross validation on our model to estimate the performance of our Machine Learning Linear Regression Model. For this, we create a matrix and store it in the mse.cv variable. We then create a for loop that loops 10 times creating different models every iteration. We fetch MSE of each model that is built in every iteration of the loop, and finally we calculate the average MSE by averaging/calculating mean of the columns of means. We use an 80/20 training/test split for the cross validation.

```
mse.cv = matrix(nrow = 10, ncol = 1)
set.seed(22)
# for loop to create multiple models
for (i in 1:10) {
  sample = sample.int(n = nrow(exams), size = floor(.8*nrow(exams)), replace = FALSE)
  train = exams[sample,]
  test = exams[-sample,]

  # Framing data for easier reusability
  exams = data.frame(exams, exams$math.score)
  # use lm() to create multiple models
  exam_data.lm <- lm(reading.score ~ gender + race.ethnicity + parental.level.of.education
                     + lunch + test.preparation.course, data = train)

  # get the MSE for each model
  mse.cv[i,1] = mean((test$reading.score - predict(exam_data.lm, test))^2)
}

# use colMeans() to find means of all mse.cv values
colMeans(mse.cv)
```

```
## [1] 170.5502
```

In this case, we get an MSE value of 165.3882. Since we are looking to make inferences based on the model, a low MSE is not vital to the analysis.

2.3 Results

```
# finally, use summary to find which predictors are significant in predicting the
# response variable, math.scores
summary(exams.lm)
```

```
##
## Call:
## lm(formula = reading.score ~ gender + race.ethnicity + parental.level.of.education +
##     lunch + test.preparation.course, data = exams)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -39.331  -9.046   0.262   9.100  30.528
##
## Coefficients:
##                                              Estimate Std. Error t value
## (Intercept)                                   72.1475     1.9109  37.756
## gendermale                                    -5.8814     0.8159  -7.208
## race.ethnicitygroup B                         -0.9794     1.6922  -0.579
## race.ethnicitygroup C                         -0.7088     1.6070  -0.441
## race.ethnicitygroup D                          4.5592     1.6411   2.778
## race.ethnicitygroup E                          6.6393     1.8236   3.641
## parental.level.of.educationbachelor’s degree   1.8772     1.5068   1.246
## parental.level.of.educationhigh school        -3.9272     1.2712  -3.089
## parental.level.of.educationmaster’s degree     4.2452     1.7786   2.387
## parental.level.of.educationsome college       -2.8373     1.2397  -2.289
## parental.level.of.educationsome high school   -7.3478     1.2911  -5.691
## lunchstandard                                  8.4732     0.8518   9.947
## test.preparation.coursenone                   -7.4356     0.8573  -8.673
##                                              Pr(>|t|)
## (Intercept)                                   < 2e-16 ***
## gendermale                                   1.13e-12 ***
## race.ethnicitygroup B                        0.562895
## race.ethnicitygroup C                        0.659265
## race.ethnicitygroup D                        0.005571 **
## race.ethnicitygroup E                        0.000286 ***
## parental.level.of.educationbachelor’s degree 0.213113
## parental.level.of.educationhigh school       0.002062 **
## parental.level.of.educationmaster’s degree   0.017180 *
## parental.level.of.educationsome college      0.022308 *
## parental.level.of.educationsome high school  1.66e-08 ***
## lunchstandard                                 < 2e-16 ***
## test.preparation.coursenone                   < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 12.73 on 987 degrees of freedom
## Multiple R-squared:  0.2628, Adjusted R-squared:  0.2539
## F-statistic: 29.32 on 12 and 987 DF,  p-value: < 2.2e-16
```

When we print the summary of the linear model, we discover the following:

- Having Standard Lunch has the greatest positive impact on the Reading Score of a student (Estimate: 8.4732).
- Being a student from race or ethnic group E (Estimate: 6.6393) also has a great positive impact on the Reading Score of a student.
- The greatest negative impact on Writing Score of a student is caused by not taking the Test Preparation Course (Estimate: -7.4356).
- A student belonging to ethnic group D or E, parents having a bachelor's degree, or master's degree, and/or standard lunch will have an increased Reading score than other students as those predictors impact Reading scores positively.
- A student with gender male, belonging to ethnic group B or C, parents having high school education, or some college, or some high school education and/or not taking the test preparation course will have a decreased/lower Reading score as these predictors impact Reading scores negatively.

3 Writing Scores Prediction Linear Regression Model

3.1 Linear Model [siginificant predictors, best subset selection]

Since our response variable (writing.score) is quantitative, we will use Linear Regression as our model of choice to fetch the desired output. Once the response, data set, and predictors are set up, we can create the Linear Model as follows:

$$writing.score = B_0 + B_1 * gender + B_2 * race.ethnicity.B + B_3 * race.ethnicity.C + B_4 * race.ethnicity.D + B_5 * race.ethnicity.E + B_6 * parental.level.of.education.bachelors + B_7 * parental.level.of.education.highschool + B_8 * parental.level.of.education.masters + B_9 * parental.level.of.education.some.college + B_10 * parental.level.of.education.some.highschool + B_11 * lunch + B_12 * test.preparation.course$$

Where $B_1 = \{0 \text{ if female, } 1 \text{ if male}\}$,
$B_{11} = \{0 \text{ if reduced, } 1 \text{ if standard}\}$,
$B_{12} = \{0 \text{ if completed, } 1 \text{ if not}\}$, and
$B_2, B_3, ..., B_{10} = \{0 \text{ if not, } 1 \text{ if so}\}$

```r
# create linear model and load it into a exams.lm variable
exams.lm <- lm(writing.score ~ gender + race.ethnicity + parental.level.of.education + lunch + test.preparation.course, data = exams)
```

Next, we want to interpret which predictors - gender, race.ethnicity, parental.level.of.education, lunch, and test.preparation.course are significant to predict the Writing Exam scores for a student. Based on the step() function, we find out that all predictors are significant in predicting Reading scores for a given student. This was calculated in section 1.1.

3.2 Perform Cross Validation

Next, we perform cross validation on our model to estimate the performance of our Machine Learning Linear Regression Model. For this, we create a matrix and store it in the mse.cv variable. We then create a for loop that loops 10 times creating different models every iteration. We fetch MSE of each model that is built in every iteration of the loop, and finally we calculate the average MSE by averaging/calculating mean of the columns of means. We use an 80/20 training/test split for the cross validation.

```r
mse.cv = matrix(nrow = 10, ncol = 1)
set.seed(32)
# for loop to create multiple models
for (i in 1:10) {
  sample = sample.int(n = nrow(exams), size = floor(.8*nrow(exams)), replace = FALSE)
  train = exams[sample,]
  test = exams[-sample,]

  # Framing data for easier reusability
  exams = data.frame(exams, exams$math.score)
  # use lm() to create multiple models
  exam_data.lm <- lm(reading.score ~ gender + race.ethnicity + parental.level.of.education
                    + lunch + test.preparation.course, data = train)

  # get the MSE for each model
  mse.cv[i,1] = mean((test$reading.score - predict(exam_data.lm, test))^2)
}

# use colMeans() to find means of all mse.cv values
colMeans(mse.cv)
```

```
## [1] 165.6883
```

In this case, we get an MSE value of 165.6883. Since we are looking to make inferences based on the model, a low MSE is not vital to the analysis.

3.3 Results

```
# finally, use summary to find which predictors are significant in predicting the
# response variable, math.scores
summary(exams.lm)
```

```
##
## Call:
## lm(formula = writing.score ~ gender + race.ethnicity + parental.level.of.education +
##     lunch + test.preparation.course, data = exams)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -37.382  -8.717   0.275   9.145  28.298
##
## Coefficients:
##                                            Estimate Std. Error t value
## (Intercept)                                 72.7883     1.8660  39.007
## gendermale                                  -7.9286     0.7968  -9.951
## race.ethnicitygroup B                       -0.6381     1.6525  -0.386
## race.ethnicitygroup C                       -0.5657     1.5693  -0.361
## race.ethnicitygroup D                        6.7070     1.6026   4.185
## race.ethnicitygroup E                        6.6207     1.7808   3.718
## parental.level.of.educationbachelor's degree   2.8744     1.4714   1.953
## parental.level.of.educationhigh school        -5.6195     1.2414  -4.527
## parental.level.of.educationmaster's degree     5.4470     1.7368   3.136
## parental.level.of.educationsome college       -3.3463     1.2106  -2.764
## parental.level.of.educationsome high school   -8.6177     1.2608  -6.835
## lunchstandard                                9.7013     0.8318  11.663
## test.preparation.coursenone                -10.1754     0.8372 -12.154
##                                            Pr(>|t|)
## (Intercept)                                 < 2e-16 ***
## gendermale                                  < 2e-16 ***
## race.ethnicitygroup B                      0.699465
## race.ethnicitygroup C                      0.718548
## race.ethnicitygroup D                      3.10e-05 ***
## race.ethnicitygroup E                      0.000212 ***
## parental.level.of.educationbachelor's degree 0.051045 .
## parental.level.of.educationhigh school       6.72e-06 ***
## parental.level.of.educationmaster's degree   0.001762 **
## parental.level.of.educationsome college      0.005813 **
## parental.level.of.educationsome high school  1.43e-11 ***
## lunchstandard                               < 2e-16 ***
## test.preparation.coursenone                 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.43 on 987 degrees of freedom
## Multiple R-squared:  0.3727, Adjusted R-squared:  0.3651
## F-statistic: 48.87 on 12 and 987 DF,  p-value: < 2.2e-16
```

When we print the summary of the linear model, we discover the following:

- Having Standard Lunch has the greatest positive impact on the Writing Score of a student (Estimate: 9.7013).
- Being a student from race or ethnic group D (Estimate: 6.7070) also has a great positive impact on the Writing Score of a student.
- The greatest negative impact on Writing Score of a student is caused by not taking the Test Preparation Course (Estimate: -10.1754).
- A student belonging to ethnic group D or E, parents having a bachelor's degree, or master's degree, and/or standard lunch will have an increased Writing score than other students as those predictors impact Writing scores positively.
- A student with gender male, belonging to ethnic group B or C, parents having high school education, or some college, or some high school education and/or not taking the test preparation course will have a decreased/lower Writing score as these predictors impact Writing scores negatively.

Conclusion

In this report, we examined both the success of the test preparation course and who it should be marketed to in the future. We will focus on the overall results of the latter first. From examining the three model outputs in part 2, we see that scores decrease when the student does not take the course and that that predictor is statistically significant. What we are looking for is what predictors are negatively impacting grades and how can we encourage students with these qualities to take the test preparation course in order to increase their grades. For each of these models, not having free or reduced lunch translates to having the largest increase in predicted score for the student. Translation: students with an economic disadvantage are more likely to underperform. All of these models show that students with parents that do not have a college degree are predicted to have lower scores as well. To increase test scores for the district and test preparation course participation, our solution would be to offer scholarships to students from economically disadvantaged backgrounds and/or students whose parents do not have a college degree.

We deemed that the writing and math scores were relevant to predicting test preparation course participation and that higher scores in reading and math were associated with participation in answering the first question. When analyzing the success of the decision tree and the linear regression model, it is easy to see that for this question the logistic regression model was much more useful in providing interpretable results for our business question than the decision tree was. In part, this is because of the high variability of the decision tree. While a decision tree can be good for giving overall insights on the data and how it relates to the predictor, it cannot outperform a regression model (logistic or linear) when looking for relationships. In conclusion, we have found that for identifying and making inferences about relationships a logistic regression model is better than a decision tree.

Endnotes

Analysis for the logistic regression model was performed by Noah Alexander, Joshua Duazo, and Toan Nguyen. The decision tree model was performed by Nathan Acosta. The linear regression models were performed by Ali Zain Charolia and Sahran Prasla. The report was contributed to by everyone, with compilation, formatting, editing, and conclusions written by Noah Alexander.

References

James, Gareth, et al. An Introduction to Statistical Learning: With Applications in R (Springer Texts in Statistics). 2nd ed. 2021, Springer, 2022.

“Students Performance in Exams.” Students Performance in Exams | Kaggle, /datasets/whenamancodes/students-performance-in-exams. Accessed 1 Dec. 2022.

R Core Team (2022). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.