|  |  |  |
| --- | --- | --- |
| A picture containing text, sign, clipart  Description automatically generated | **BOSTON**  **UNIVERSITY** | **METROPOLITAN COLLEGE** |

**AD 699 DATA MINING FOR BUSINESS ANALYTICS**

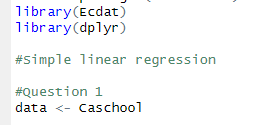
**ASSIGNMENT 2**

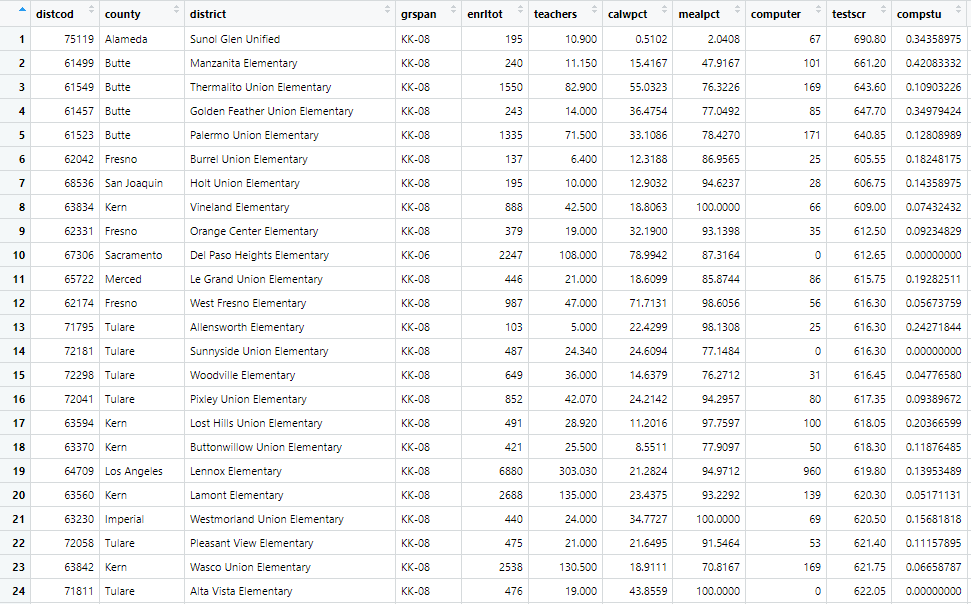
**MARCH 4, 2023**

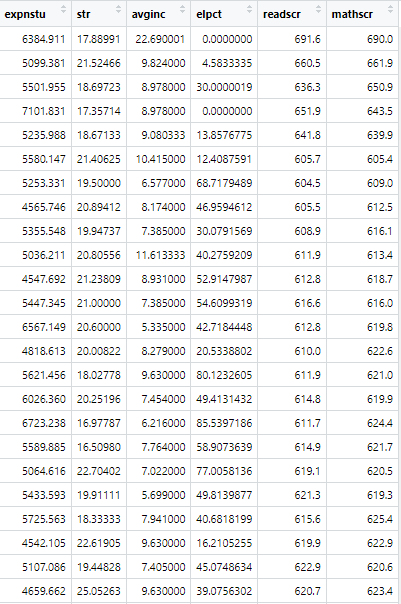
**Aravind Hanumantha Rao**

**BU ID - U55859882**

Bring this dataset into your R environment-





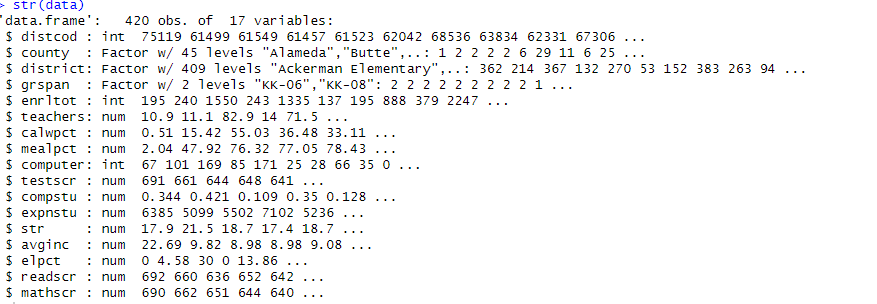


As seen from the codes and the database , I have installed the library from the cran repository and loaded the library of Ecdat . The data consist of Caschool of 17 columns and 420 records.

**2) Use either the str() function or the glimpse() function from dplyr to learn more about**

**this dataset. After taking a look at the dataset description and seeing the results here,**

**which variables in the dataset are numeric, and which are categorical?**



Distcod – numeric

County – categorical

District – categorical

Grspan- categorical

Enrltot – numeric

Teachers – numeric

Calwpct – numeric

Mealpct- numeric

Computer -numeric

Testscr-numeric

Compstu- numeric

Expnstu -numeric

Str-numeric

Avginc- numeric

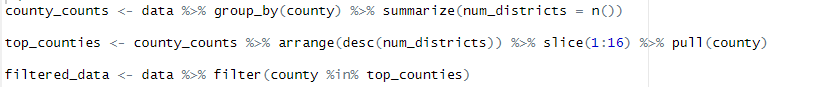
Elpct- numeric

Readscr- numeric

Mathsscr- numeric

**3)Filter the dataset so that only rows from the 16 most common counties remain (this will**

**leave you with just the counties that have 10 or more school districts).**

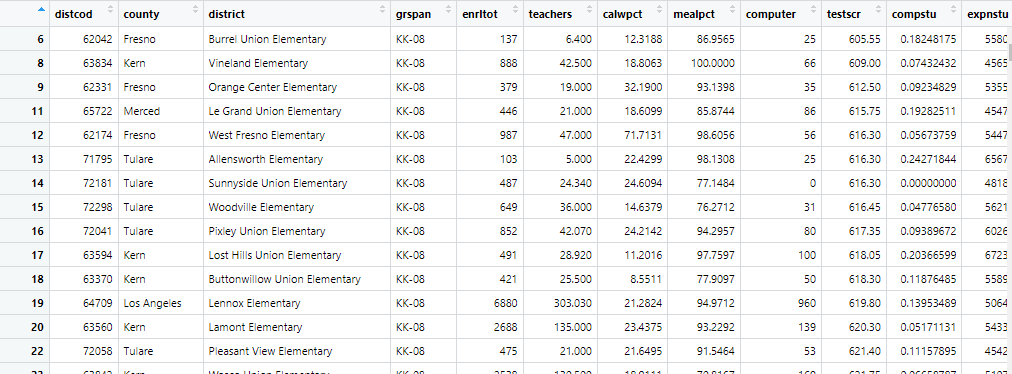


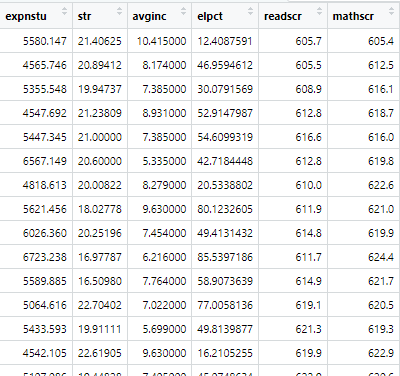
Here is the sample of few rows on the number of districts for each county



1. County\_counts variable – it groups by the county and counts the number of districts in each of the county
2. Top\_counties variable – then sort it by descending order and arranges it by choosing the top 16 countries using the slice function
3. Filtered\_data – shows the results of the top counties

The data looks like this –





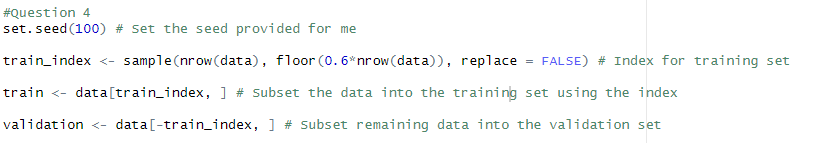
Question 4

**Using your assigned seed value, create a data partition. Assign approximately 60% of**

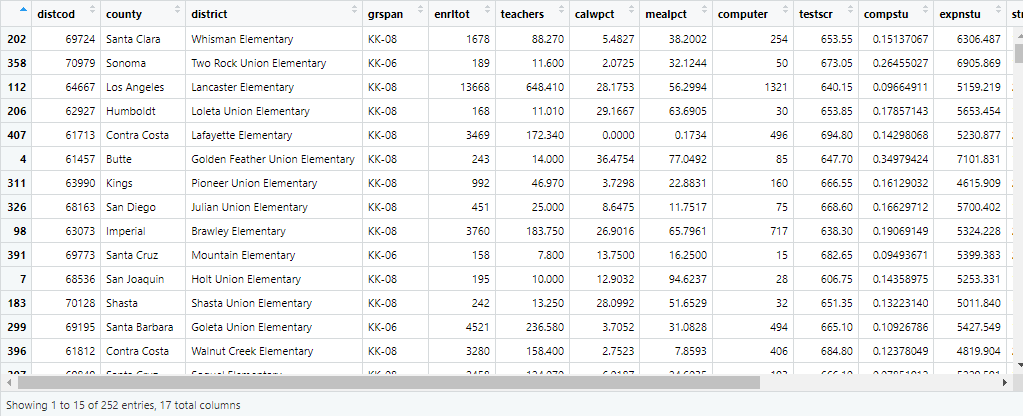
**the records to your training set, and the other 40% to your validation set.**

**a. Why is it important to partition the data before doing any sort of in-depth**

**analysis of the variables?**

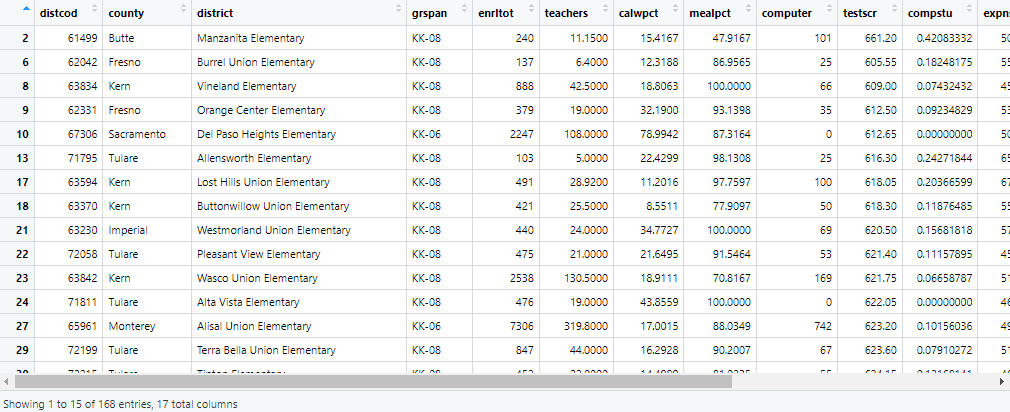


Training set – 60 percent ( 252 entries)



Random sample of observation from the 400 entries where 60 percent is used into the training set.

Validation set – 40 percent ( 168 entries)



Random sample of observation from the 400 entries where 40 percent is used into the training

As seen from the code , the set.seed produces the seed value. This line of code randomly selects a sample of rows from the data object to be included in the training set. The number of rows selected is determined by taking 60% of the total number of rows in the data object and rounding down to the nearest integer using the floor() function. The resulting sample is stored in the train\_index variable. The replace = FALSE argument ensures that each selected row is unique and not repeated in the sample.

Partitioning the data is important because it allows for more accurate modeling and testing of the data. By splitting the data into training and validation sets, one can train the model on a subset of the data and then test it on another subset to evaluate its performance. This helps to prevent overfitting of the model to the training data and allows for a more reliable assessment of its ability to predict outcomes on new data.

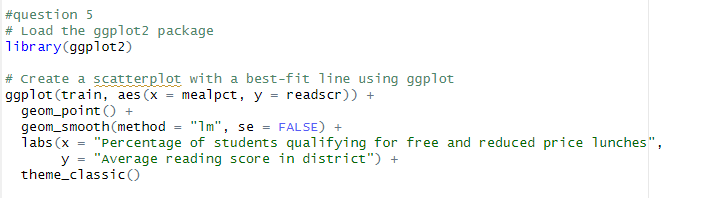
**5)Let’s explore the relationship between readscr (average reading score in a district) and**

**mealpct (the percentage of students in the district who qualify for free and reduced**

**price lunches, based on low family incomes). Using ggplot, create a scatterplot that**

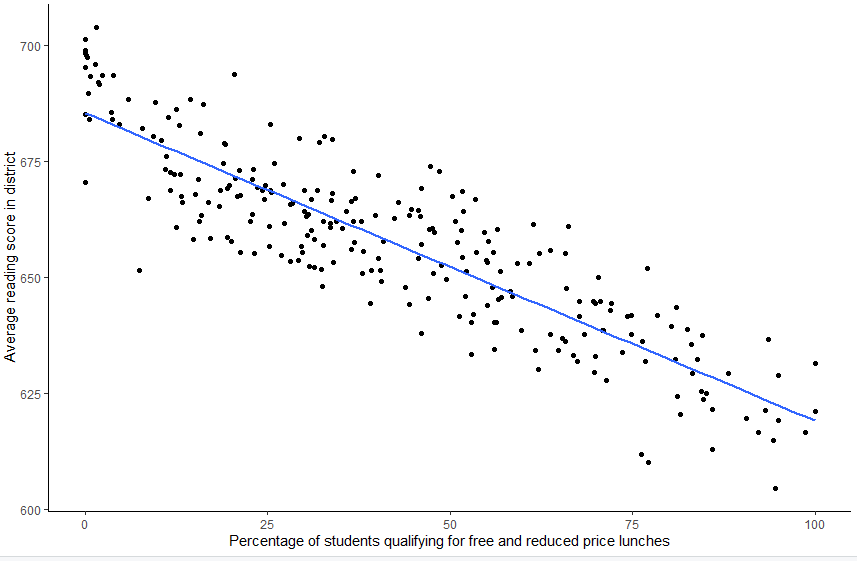
**depicts readscr on the y-axis and mealpct on the x-axis. Add a best-ﬁt line to this**

**scatterplot. Use only your training set data to build this plot?**

****

Output –

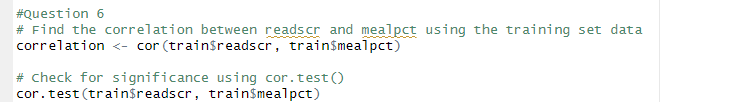
This code creates a scatterplot using the ggplot2 package in R. The plot displays the relationship between two variables: the percentage of students in a district who qualify for free and reduced price lunches (x-axis) and the average reading score in the district (y-axis). The plot also includes a best-fit line generated by a linear regression model using the geom\_smooth() function with the method set to "lm". The labs() function is used to set the axis labels, and the theme\_classic() function is used to adjust the plot's appearance. Overall, this code is used to visualize the relationship between mealpct and readscr in the training set.



**What does this plot suggest about the relationship between these variables?Does this make intuitive sense to you? Why or why not?**

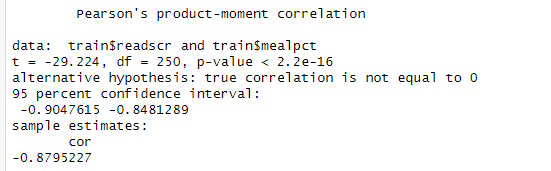
The scatterplot with a best-fit line indicates that readscr and mealpct have an inverse relationship, meaning that as the percentage of students in a district who qualify for free and reduced-price lunches increases, the district's average reading score tends to decrease. This relationship aligns with common sense because students from low-income households may encounter various obstacles that can make academic achievement more difficult, such as lack of access to educational resources like books and technology. In addition, low-income students may have a higher risk of food insecurity, which can affect their cognitive development and lead to poor health outcomes.

**6)Now, again using training set data only, ﬁnd the correlation between Readscr and mealpct. Then, use cor.test() to see whether this correlation is signiﬁcant ?**



Output –





This code finds the correlation between the reading scores (readscr) and the percentage of students who qualify for free and reduced price lunches (mealpct) using the training set data. The function cor() is used to calculate the correlation coefficient. The result is stored in the variable correlation. Then, the cor.test() function is used to check for the significance of the correlation. This function returns the correlation coefficient, the p-value, and a confidence interval for the correlation.

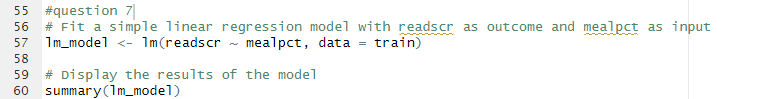
**What is this correlation? Is it a strong one?Is the correlation signiﬁcant?**

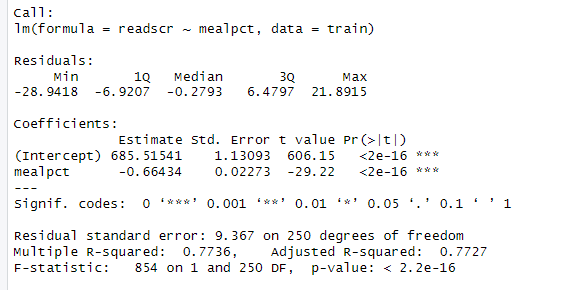
The output shows the result of the correlation test between train$readscr and train$mealpct. The value of t-statistic is -29.224 with a degrees of freedom of 250, and the p-value is less than 2.2e-16, which is extremely small. This indicates that the correlation between the two variables is significant, and we can reject the null hypothesis that the true correlation is zero. The 95% confidence interval is -0.9047615 to -0.8481289, which indicates that we are 95% confident that the true correlation lies within this range. The correlation coefficient is -0.8795227, which indicates a strong negative correlation between train$readscr and train$mealpct.

**7)Using your training set, create a simple linear regression model, with readscr as your**

**outcome variable and mealpct as your input variable. Use the summary() function to**

**display the results of your model?**

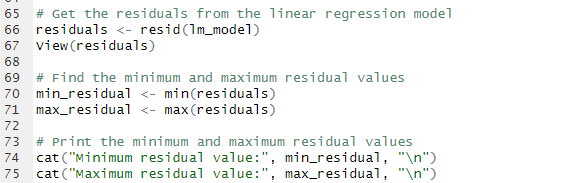




This code fits a simple linear regression model using the lm() function in R. The dependent variable or the outcome variable is readscr, and the independent variable or input variable is mealpct. The data argument specifies that the model is to be fit on the training data.

The summary() function is then called to display the results of the model. This provides a summary of the key statistics and parameters of the regression model, including the coefficients, standard errors, t-values, p-values, and the R-squared value. These statistics provide information on the strength and significance of the relationship between the input and output variables.

**What are the minimum and maximum residual values in this model?**

****

****

This code fits a simple linear regression model to the training data with readscr as the outcome variable and mealpct as the predictor variable. Then, it obtains the residuals from the linear regression model using the resid() function and stores them in a variable named residuals. The View() function is used to view the residuals in a separate window.

Next, the code finds the minimum and maximum residual values in the residuals vector using the min() and max() functions, respectively, and stores them in variables named min\_residual and max\_residual.

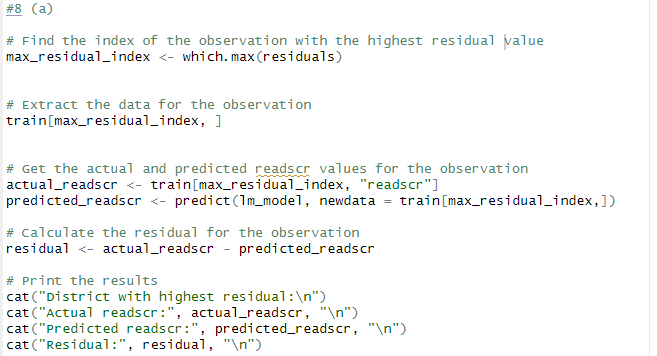
**a.**

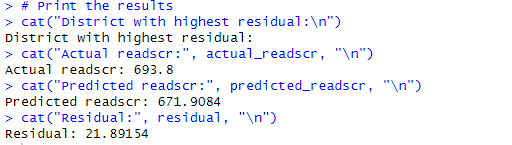
**Find the observation whose rating generated the highest residual value in your**

**model. What was the district’s actual average reading score? What did the**

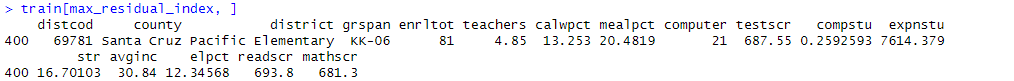
**model predict that it would be? How is the residual calculated from the two**

**numbers that you just found?**

****

****

This code finds the index of the observation in the training data that has the highest residual value, extracts the data for that observation, gets the actual and predicted reading scores for that observation using the linear regression model fitted earlier, and calculates the residual for that observation. The residual is the difference between the actual reading score and the reading score predicted by the linear regression model for that observation.



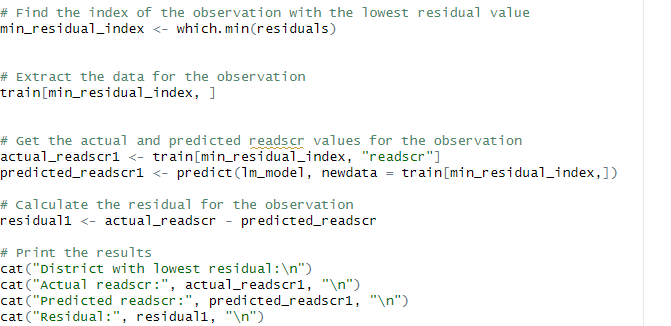
The extracted value from the observation

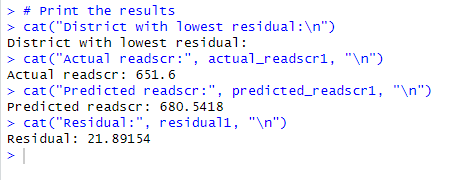
**b) Find the observation whose rating generated the lowest residual value. What**

**was the district’s actual average reading score? What did the model predict that**

**it would be? How is the residual calculated from the two numbers that you just**

**found?**

****

****

This code finds the index of the observation with the lowest residual value, extracts the data for that observation, gets the actual and predicted readscr values for the observation using the linear regression model, and calculates the residual for the observation.

**It looks like there are some cases where this model is quite a bit “o the mark.”**

**Write a few sentences with your thoughts about why mealpct may not perfectly**

**predict reading scores?**

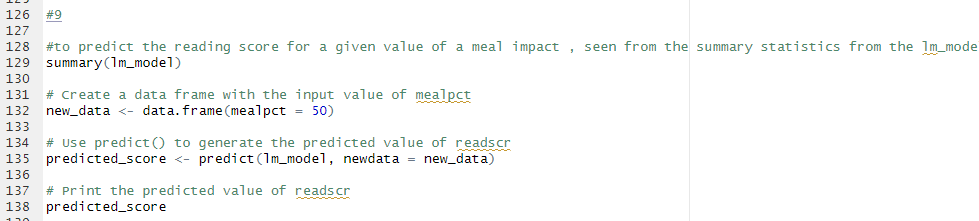
There are multiple reasons why the variable mealpct may not be a perfect predictor of reading scores. These include the presence of confounding variables that are linked to both mealpct and reading scores, which could result in a false association between the two variables. Additionally, measurement error in the data used to measure both mealpct and reading scores may lead to inaccurate estimates of their relationship.

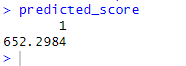
**9)What is the regression equation generated by your model? Make up a hypothetical**

**input value and explain what it would predict as an outcome. To show the predicted**

**outcome value, you can either use a function in R, or just explain what the predicted**

**outcome would be, based on the regression equation and some simple math?**

****

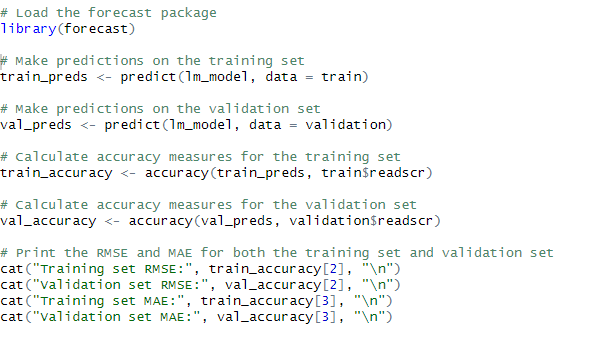
****

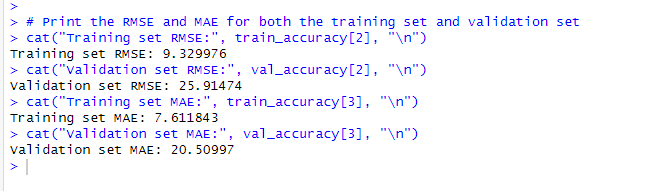
This code predicts the reading score for a given value of meal impact, based on the simple linear regression model created earlier. It first displays the summary statistics of the model using the summary() function. Then, a new data frame is created with a specified value of mealpct. I have kept the specified value to be 50.The predict() function is then used to generate the predicted value of readscr for the given value of mealpct. Finally, the predicted value is printed to the console using cat().

**10)Using the accuracy() function from the forecast package, assess the accuracy of your**

**model against both the training set and the validation set. What is the purpose of**

**making this comparison? Focus on RMSE and MAE here in particular.**

****

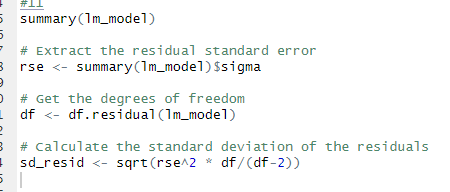
****

The purpose of this code is to compare the accuracy of the linear regression model on both the training and validation datasets. The code first loads the forecast package, then uses the predict function to generate predicted values for the readscr variable using the linear regression model (lm\_model) for both the training and validation datasets.

Next, the accuracy function from the forecast package is used to calculate the Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) between the predicted values and the actual readscr values for both datasets.

RMSE is a measure of how much the predictions differ from the actual values on average, while MAE measures the average absolute difference between the predicted and actual values. Lower values of RMSE and MAE indicate better model performance.

The purpose of comparing the RMSE and MAE for both the training set and validation set is to evaluate the performance of the model on both datasets and to determine if the model is overfitting or underfitting. A lower RMSE and MAE indicate better performance of the model. In this case, the RMSE and MAE are lower for the training set compared to the validation set, suggesting that the model may be overfitting the training data and not generalizing well to new data.

’

This code calculates the standard deviation of the residuals in the linear regression model lm\_model.

First, it extracts the residual standard error (RSE) from the summary of the lm\_model. The RSE is an estimate of the standard deviation of the errors that are not explained by the model.

Then, it gets the degrees of freedom (df) for the model using the df.residual function.

Finally, it calculates the standard deviation of the residuals (sd\_resid) using the formula sqrt(rse^2 \* df/(df-2)). This formula uses the RSE and df to estimate the standard deviation of the errors that are not explained by the model. The purpose of calculating the standard deviation of the residuals is to assess how well the model fits the data and to check if the assumptions of linear regression are met.

Model rsme is 9.329976 and the standard deviation is –



The model's RMSE value is 9.329976 and the standard deviation of reading scores in the training set is 9.404. Comparing both values indicates that the model is making reasonably accurate predictions. However, it also suggests that there is still some error in the predictions, as the RMSE is not significantly smaller than the standard deviation. In summary, this comparison teaches us that the model is useful but not perfect, and there is still room for improvement to further decrease the prediction error. This could be done by exploring other variables or using more complex models.

**Multiple linear regression**

1.

Before we go any further, let’s clean things up a bit here by getting rid of some

variables. For anything you remove, take it out of both your training set and your

validation set.

a. The outcome variable that we used in the ﬁrst part of this assignment is one of

three total test score variables in this dataset. We’ll re-use the reading score

again here, but get rid of the other two test score variables now -- that will save

us from possible problems later on.

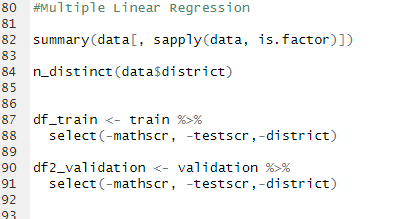
b. Next -- if there are any categorical variables that have as many, or nearly as

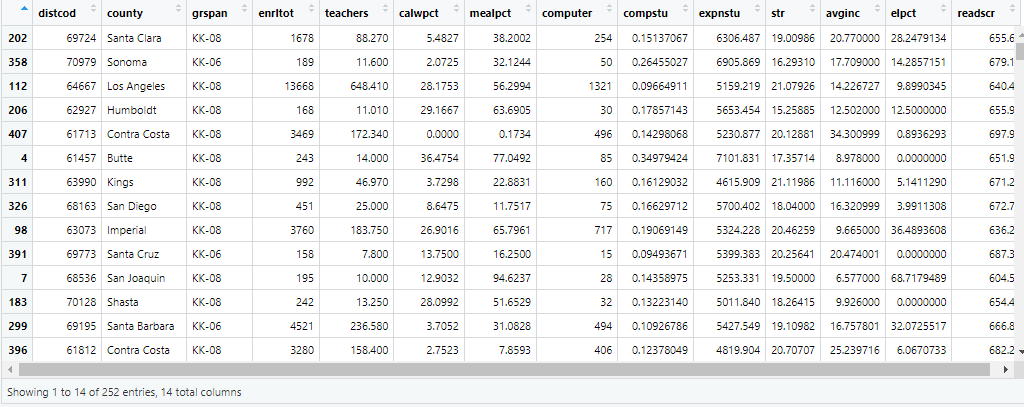
many, unique values as there are records in the dataset, get rid of them, too. Be

careful here. To know whether a variable is categorical or numeric, you will

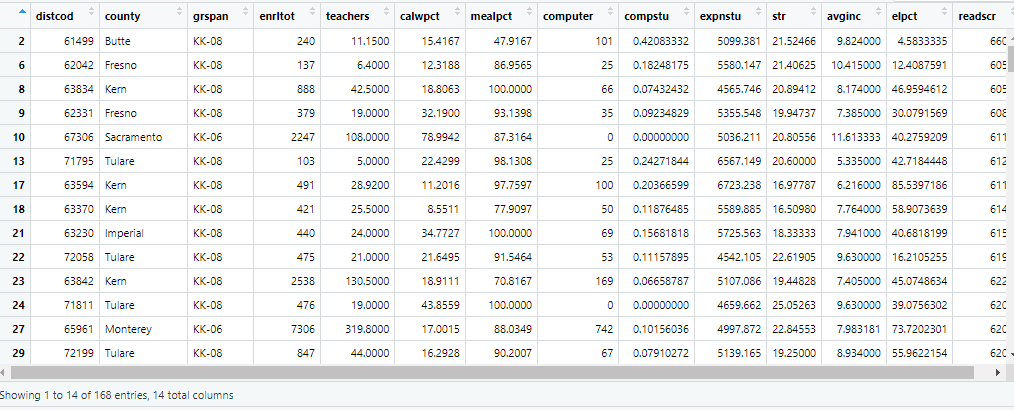
sometimes have to read the dataset description. Lazily just using the str()

results without thinking about variables’ meanings could cause problems here.





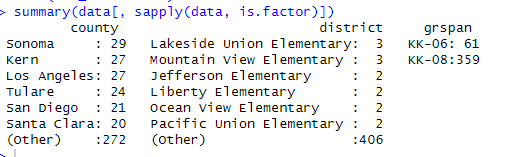
Df\_train- the result of 252 entries excluding the columns of mathsscr , testscr and district



Df\_validation – the result of 168 entries excluding mathsscr, testscr and district

The question for part will be only using the reading score and get rid of the other scores . seen from the 2 pictures the changes have been made

The second part of the question , to check for unique values ( categorical variables 0 . There are 3 categorical variables as per the data.





The command "n\_distinct(data$district)" to determine the number of distinct values (i.e., unique values) in the "district" variable within the "data" data frame. Seen there are 409 distinct values for district and if the unique values are closer to the total data sets then its better to get rid . Therefore , district values were removed and the counties were more than enough for the dataset.

**2. Build a correlation table in R that depicts the correlations among all of the numerical**

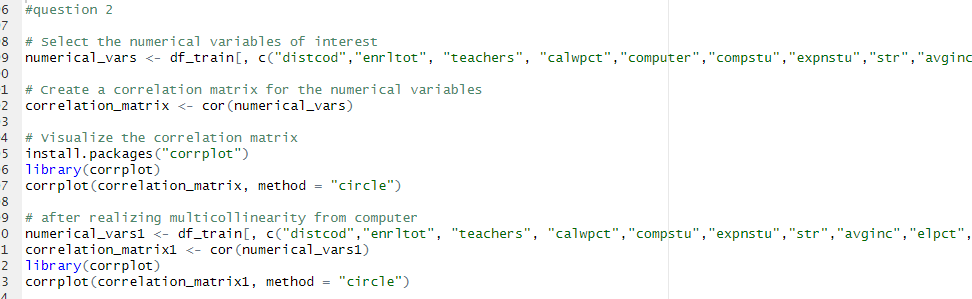
**variables that you might use as predictors (use your training set to build this). Are there any**

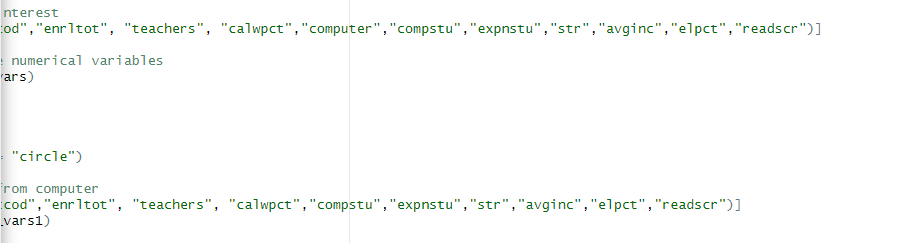
**variable relationships (.80 or higher) that suggest that multicollinearity could be an issue**

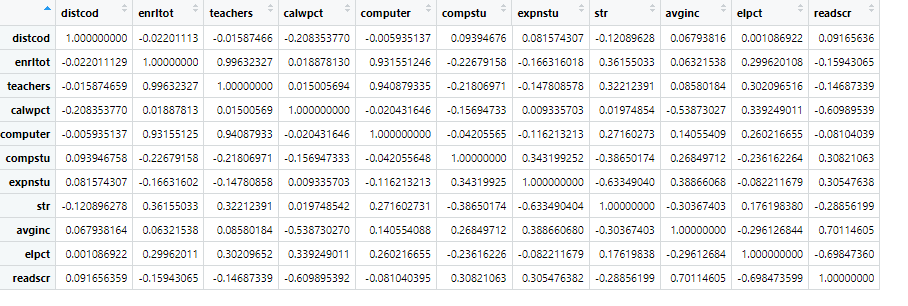
**here? If so, for any strongly correlated variable pair, remove any variables that should be**

**taken out of the model. If you removed any, how did you decide which ones to remove? If**

**not, why did you keep the ones that you have left?**



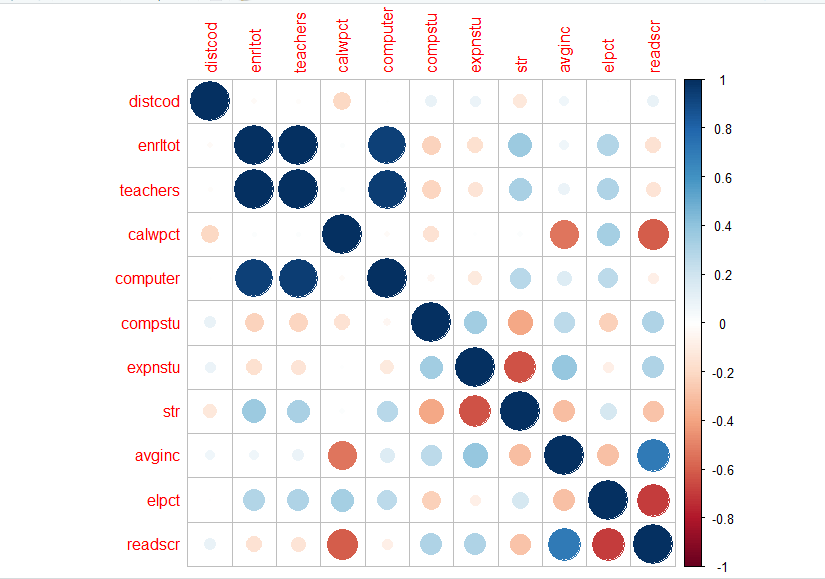




Here is the output for all the correlations among the numerical variable relationship

From the corrplot function -In summary, this code generates a circular correlation plot to visualize the pairwise correlation coefficients between variables in the dataset. The resulting plot can help to identify patterns on whether there is any multicollinearity between the variables seen above .

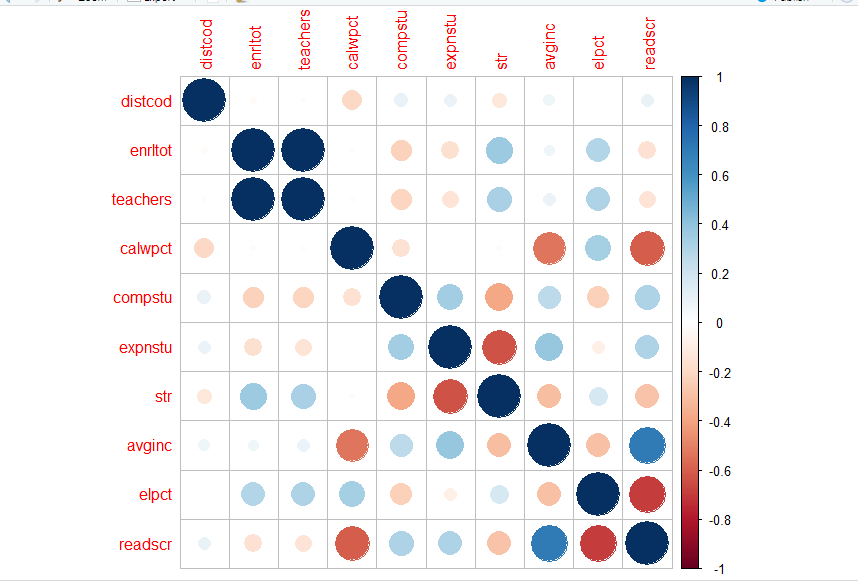
Correlation matrix: Calculate the correlation matrix between all pairs of independent variables in the dataset. If any variables have a correlation coefficient that is greater than 0.8 or less than -0.8, this indicates a strong correlation and potential multicollinearity.



Here is the corrplot function and can see the correlation where the values lie between -1 to 1 . the circles(different colours) indicate the different correlation values and helps us easily identifying whether there is multicollinearity .

Seen from the data-

Var1( enrltot) and computer(var 2) is 0.8 and higher . Va3(teachers) and Va2(computer) are 0.8 and higher . Seen computer variable is showing multicollinearity. Therefore it needs to be removed from the data .



The new corrplot shows this data after computers are removed , the multicollinearity isn’t there . the code for this is seen above.

**3)What are dummy variables? In a couple of sentences, describe what they are and explain**

**their purpose.**

Dummy variables are a form of categorical variable utilized in statistical analysis and econometrics to represent groups or categories in a quantifiable way. These variables are assigned the values of either 0 or 1 to indicate the absence or presence of a specific group or category. For instance, a researcher examining the correlation between income and education may establish a dummy variable to indicate if an individual has a college degree. This variable will take on a value of 1 if the person has a college degree and 0 if they do not. Regression analysis frequently employs dummy variables to model the impact of categorical variables on continuous outcome variables or to manage the effects of categorical variables.

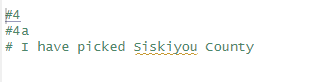
4. Let’s try building a model with just a categorical input. (Note: Question 4 is unrelated to

any other step -- you can think of it as being like its own “island”).

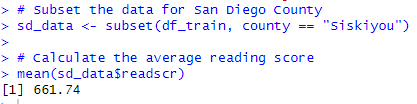
a.Pick any county in California -- it doesn’t matter which one. You can pick one

that you’ve visited, lived in, heard of...or perhaps one whose name you happen

to like. Which one did you pick?



b.Find the average (mean) reading score for your chosen county.



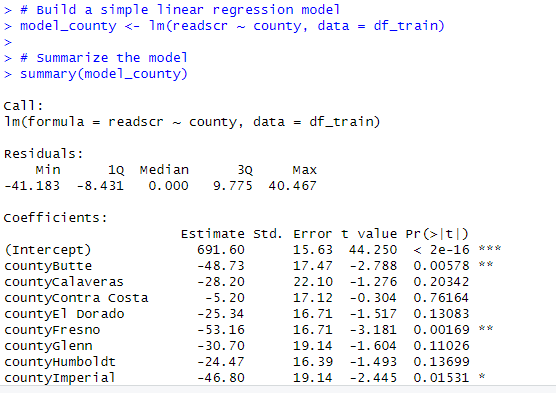
This code subsets the df\_train dataframe to create a new dataframe called sd\_data that contains only the observations where the county variable is equal to "Siskiyou".

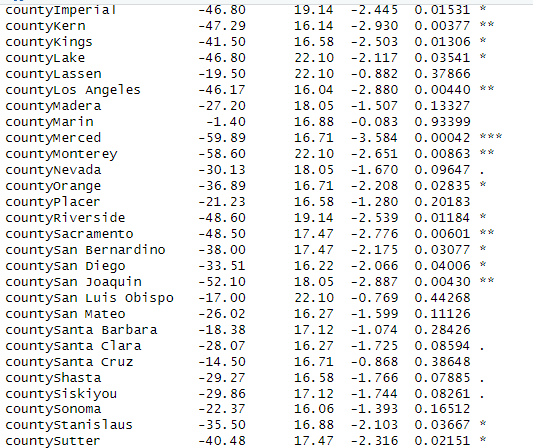
Then, the code calculates the average reading score of the subsetted data by taking the mean of the readscr variable in the sd\_data dataframe using the mean() function.

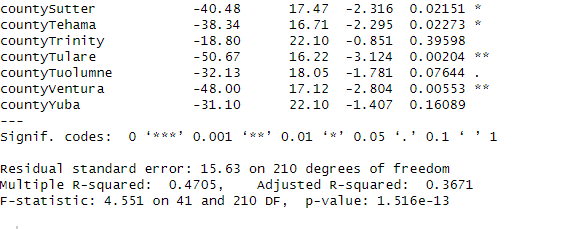
In summary, this code is calculating the average reading score for all observations in the df\_train dataframe that belong to Siskiyou county, and storing that value in the variable mean(sd\_data$readscr).

**c.Now, build a simple linear regression model, with readscr as the outcome, and**

**county as the input.**





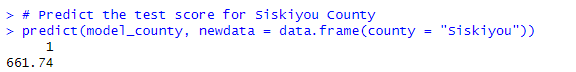


This code builds a simple linear regression model in R. The response variable is readscr and the predictor variable is county, using the lm() function.

The model is fit using the training data stored in the df\_train dataframe. The data argument specifies the data frame that contains the variables in the model formula.

The summary() function is then used to provide a summary of the linear regression model, including information such as the estimated coefficients, standard errors, t-statistics, p-values, and the R-squared value. This summary can help to evaluate the significance and strength of the relationship between the predictor and response variables.

**d.What does this model predict as the test score for your county?**



**e.What is the relationship between your answer to 4b and your answer to 4d? In a**

**sentence or two, why does this make sense? (Note: No detailed statistical**

The relationship between the answer to 4b and 4d is that the predicted test score for the chosen county would be close to the average reading score for that county. This makes sense because the simple linear regression model uses the county variable as the predictor, and the average reading score for that county is a reflection of the overall performance of the students in that county.

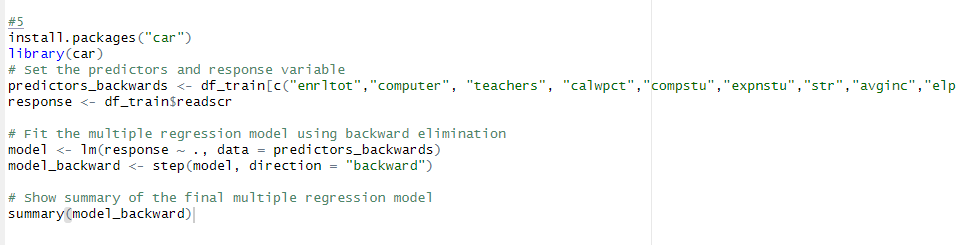
**5. Using backward elimination, build a multiple regression model with the data in your**

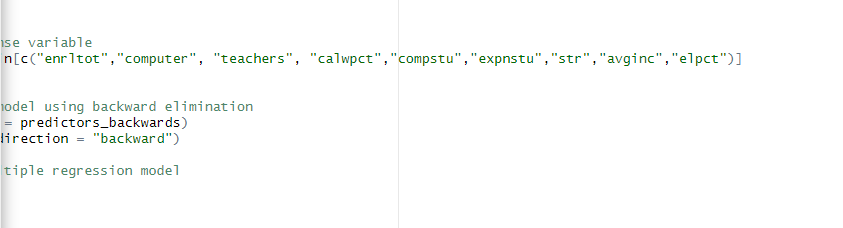
**training set, with the goal of predicting the readscr variable. Start with all of the potential**

**predictors that you have left (if you eliminated any in Step 1 or Step 2, don’t bring them**

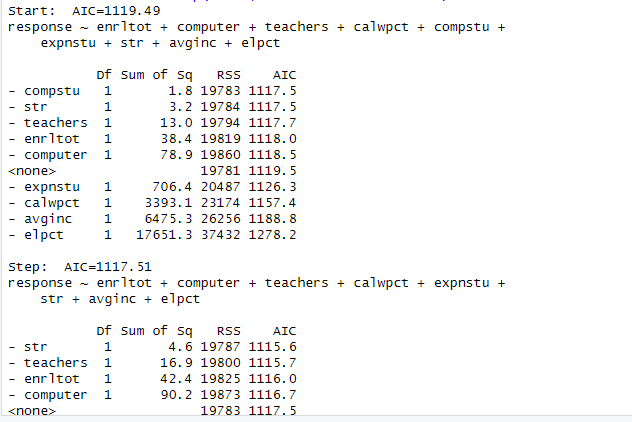
**back...they’re gone!)**

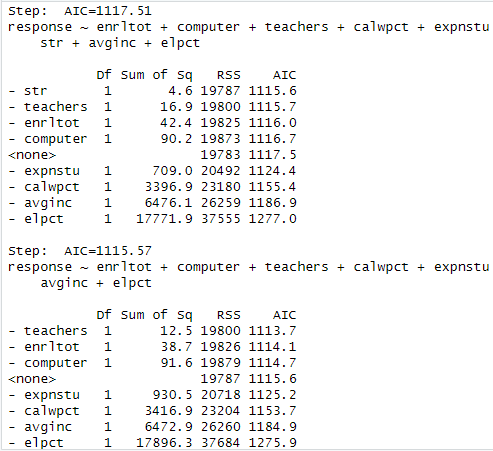
**a. Show a summary of your resulting multiple linear regression model.**

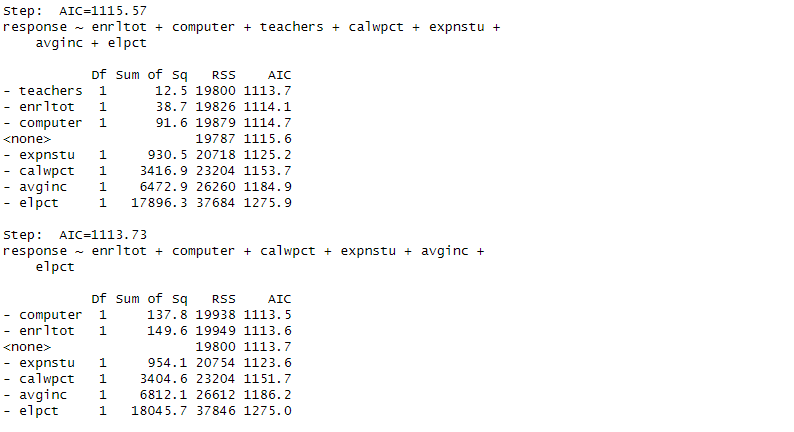


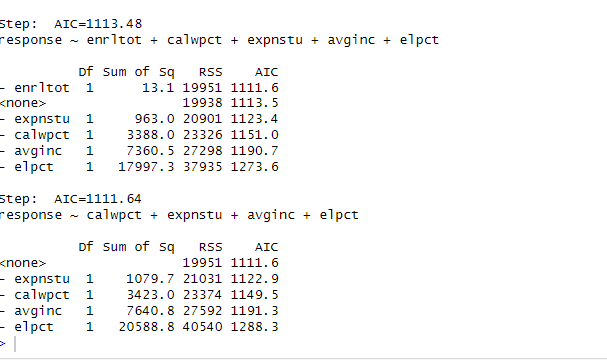


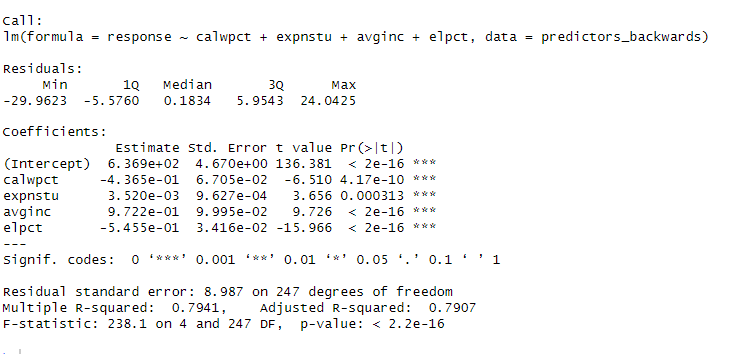
Output for model\_backward











The result indicates that –

The linear regression model shows that the variables calwpct, expnstu, avginc, and elpct are all significant predictors of the test score (response variable). The coefficients suggest that as the percentage of students qualifying for CalWORKS (calwpct) and the percentage of English learners (elpct) increase, the test score decreases. On the other hand, as expenditure per student (expnstu) and average district income (avginc) increase, the test score increases. The adjusted R-squared value of 0.7907 suggests that the model explains around 79% of the variance in the test score.

**6.**

**Model metrics**

**a. What is the total sum of squares for your model? (SST). This can be found by**

**summing all of the squared diferences from the mean for your outcome variable.**

**b. What is the total sum of squares due to regression for your model? (SSR). This can**

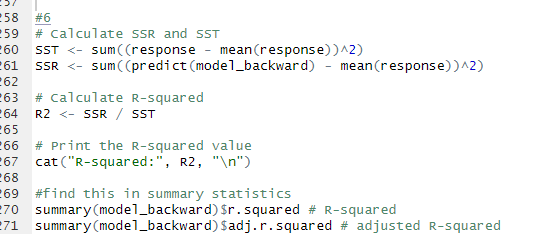
**be found by summing all the squared diferences between the ﬁtted values and the**

**mean for your outcome variable. Do not use any other SSR deﬁnition, besides the one**

**listed here in the previous sentence.**

**c. What is your SSR / SST?**

**Where can you also see this value in the summary of your regression model?**

****

****

****

**SSR/SST -**This is the r squared value and it can be found in the summary statistics and seen from the code

**7)Getting from a t-value to a p-value. Choose one of the predictors from your model (it**

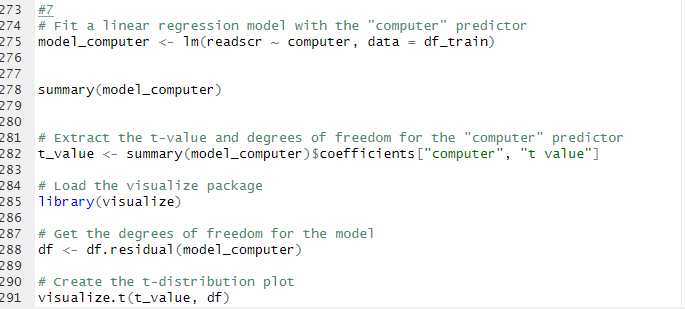
**could be a numeric input variable or a single level from a categorical input). What is the**

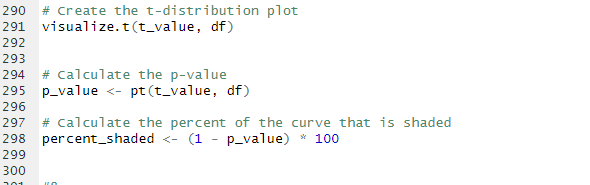
**t-value for that predictor? Using the visualize.t() function from the visualize package,**

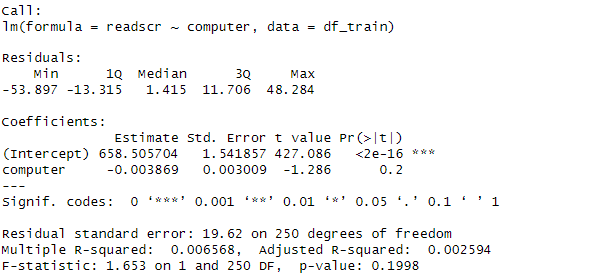
**create a plot of the t-distribution that shows the distribution for that t-value and the**

**number of degrees of freedom in your model. What percent of the curve is shaded? How**

**does this relate to the p-value for that predictor?**

****

****

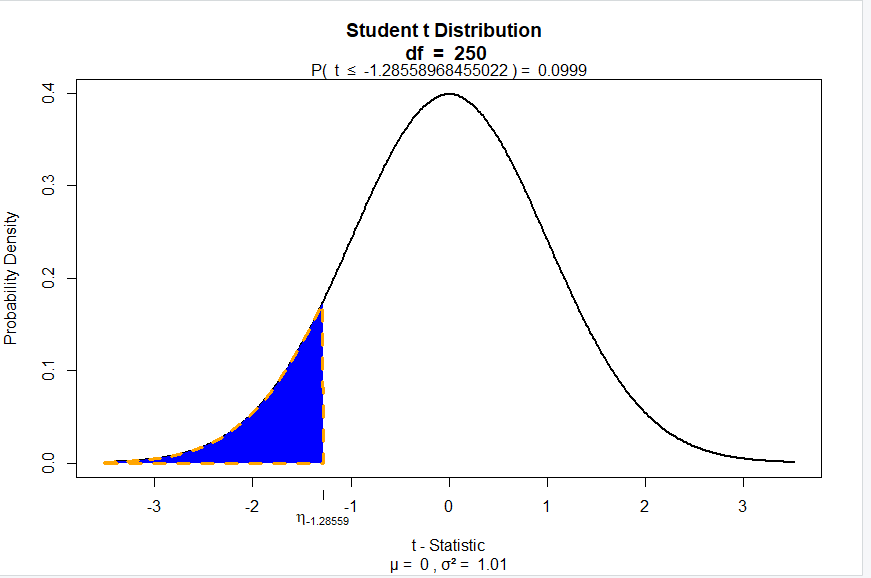
****

This is a linear regression model with readscr as the response variable and computer as the predictor variable. The model estimates the relationship between the number of computers and the average reading score. The model shows that there is no significant relationship between the number of computers and the average reading score (p-value: 0.1998), as the coefficient of the predictor variable computer is not significant (-0.003869). The model's R-squared value is very low (0.006568), indicating that the predictor variable explains only a small portion of the variance in the response variable.



\

250 degrees of freedom



P -value –

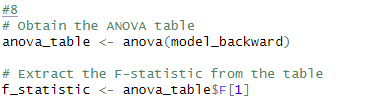
****

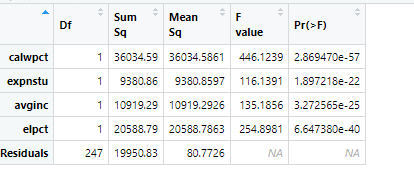
The percentage of the curve that will be shaded from the p value –



**What is your model’s F-statistic? What does the F-Statistic measure? Using R,**

**demonstrate where the F-Statistic comes from?**







The F-statistic is a measure of the overall significance of a linear regression model. It assesses whether the model as a whole explains a significant amount of variance in the outcome variable beyond what would be expected by chance.

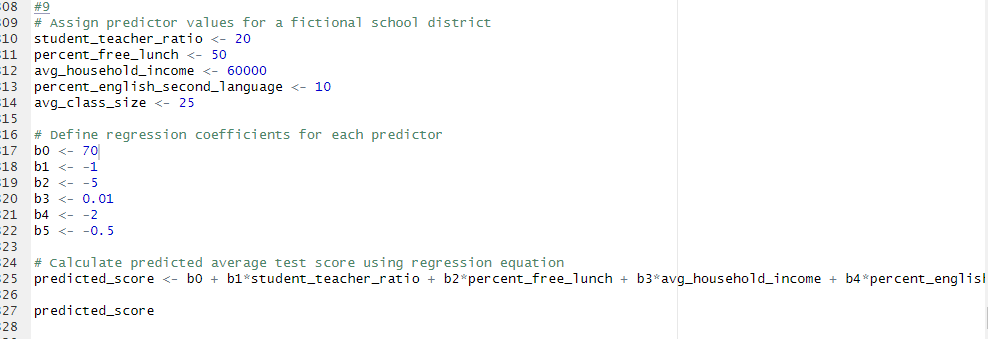
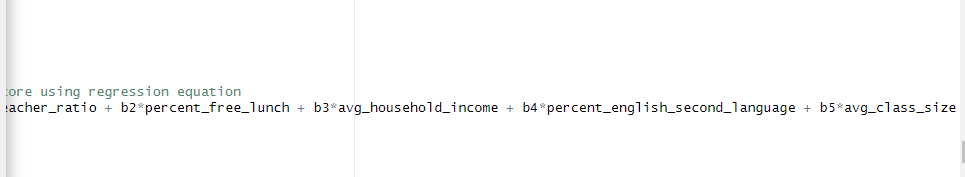
In R, the F-statistic can be found in the output of a linear regression model. It is calculated as the ratio of two measures of variability: the explained variability (or the sum of squared differences between the predicted values and the mean of the outcome variable) and the unexplained variability (or the sum of squared differences between the actual values and the predicted values). The F-statistic is then calculated as the ratio of the explained variability to the unexplained variability, divided by the degrees of freedom for each.

**9)Make up a ﬁctional school district, and assign attributes to it for each of the predictors in**

**your model. What does your model predict that this district’s average test scores will be?**

**To answer this, you can use a function in R or just explain it using the equation and some**

**simple math?**

**** ****

Explanation of the code –

This code assigns values to five predictors for a fictional school district: student-teacher ratio, percent of students receiving free lunch, average household income, percent of students who speak English as a second language, and average class size.

Then, the code defines six regression coefficients (including the intercept term) for each predictor.

Finally, the code calculates the predicted average test score for the school district using the multiple linear regression equation: predicted\_score = b0 + b1 \* student\_teacher\_ratio + b2 \* percent\_free\_lunch + b3 \* avg\_household\_income + b4 \* percent\_english\_second\_language + b5 \* avg\_class\_size, where b0-b5 are the regression coefficients and represent the expected change in the dependent variable (test scores) for a one-unit change in each predictor, holding all other predictors constant.

****

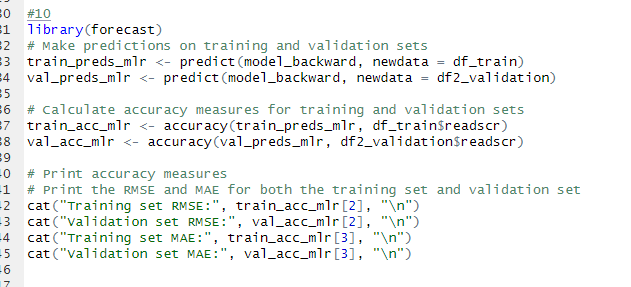
**10)Using the accuracy() function from the forecast package, assess the accuracy of your**

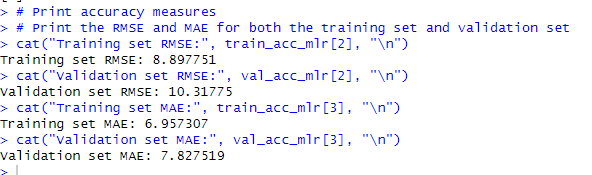
**model against both the training set and the validation set. What do you notice about these**

**results? Describe your ﬁndings in a couple of sentences. In this section, you should talk**

**about the overﬁtting risk and also about the way your MLR model differed from your SLR**

**model in terms of accuracy.**

****

****

This code is evaluating the accuracy of a multiple linear regression model on a training set and a validation set.

The first two lines use the predict function to make predictions on the training set and validation set using the multiple linear regression model stored in model\_backward. The newdata argument specifies the dataset to make predictions on.

The next two lines use the accuracy function to calculate the root mean squared error (RMSE) and mean absolute error (MAE) for the training set and validation set predictions. The accuracy function takes two arguments: the predicted values and the actual values.

The root mean squared error (RMSE) is a measure of the difference between the actual and predicted values. The lower the RMSE, the better the model fits the data. In this case, the training set RMSE of 8.897751 means that, on average, the model's predictions for the training set are off by approximately 8.9 points on a scale of 0-100. Similarly, the validation set RMSE of 10.31775 means that, on average, the model's predictions for the validation set are off by approximately 10.3 points on a scale of 0-100.

Comparing the values, we can see that the latest model (with the higher RMSE and MAE on the validation set) is likely overfitting to the training data. Overfitting occurs when a model becomes too complex and starts to memorize the noise in the training data instead of learning the underlying patterns. This often results in a model that performs very well on the training data but poorly on new, unseen data.

In this case, the increase in RMSE and MAE on the validation set suggests that the SLR model is performing worse than the MLR model, which may be an indication of overfitting. This can happen when a model has too many parameters or when the training data is not representative of the population as a whole.

To avoid overfitting, it is important to use techniques such as regularization, cross-validation, and early stopping. These techniques can help to prevent the model from becoming too complex and memorizing the noise in the training data.

Training set RMSE: 8.897751 (MLR)

Training set RMSE: 9.329976 (SLR)

Validation set RMSE: 10.31775 (MLR)

Validation set RMSE: 25.91474 (SLR)

Training set MAE: 6.957307 (MLR)

Training set MAE: 7.611843 (SLR)

Validation set MAE: 7.827519 (MLR)

Validation set MAE: 20.50997 (SLR)