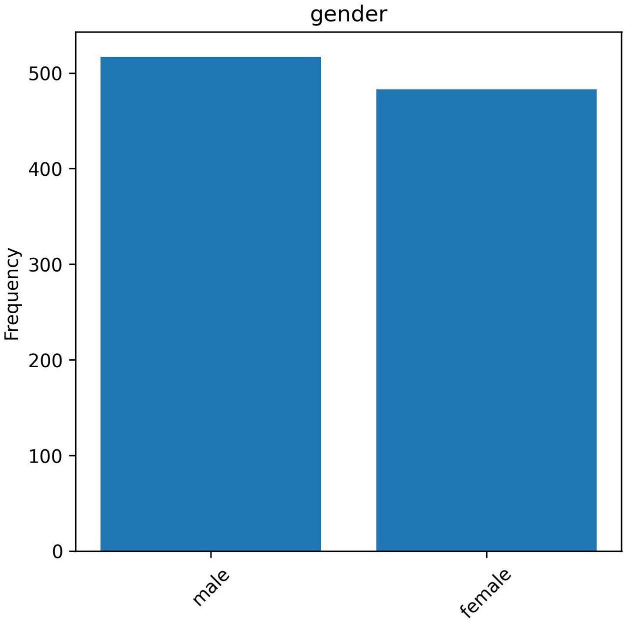
Report: A Comprehensive Analysis of Student Exam Scores and Demographic Factors

Problems to Solve

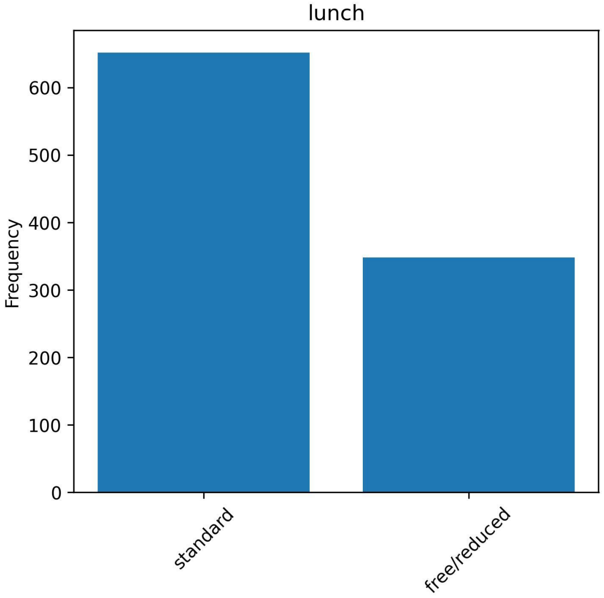
- Which variable has the most influence on test scores
- How effective is the test preparation course?
- What would be the best way to improve student scores on each test?
- What patterns and interactions in the data can you find?

Data Visualization

There are 517 Males and 483 Females in our data



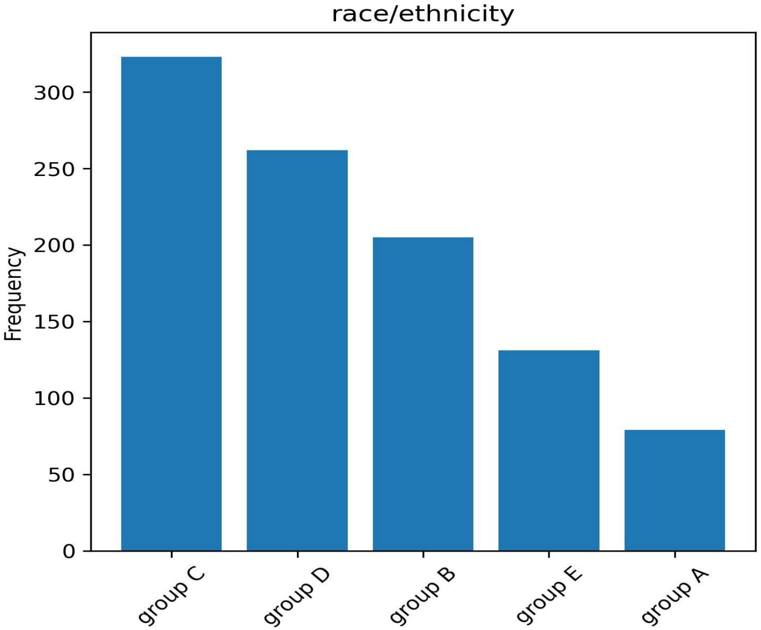
There are 652 students that have standard lunch and 348 students that have free/reduced lunch



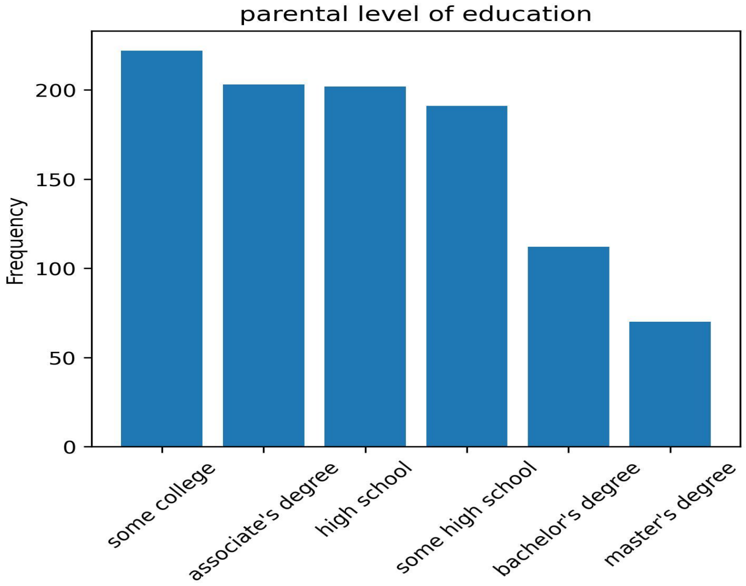
There are 335 students who completed the test preparation course and 665 that did not in our data

Data Visualization pt. 2

There are 323 students in group C, 262 in group D, 205 in group B, 131 in group E, and 79 in group A.



There are 222 students that have parents with education level “some college”, 203 with associate's degree, 202 with high school, 191 some high school, 112 bachelor's degree, and 70 with a master's degree.



Data Mining Tool #1 EDA

- Exploratory Data Analysis (EDA) is a critical initial step in understanding and summarizing a dataset.
  - It provides insights into the types of variables (categorical, numerical), their distributions, ranges, and any peculiarities like missing values or outliers.
  - This understanding is foundational for making informed decisions during analysis.

- Performed several EDA tasks using Python to gain insights into the dataset containing information about 1000 students across various categories
  - Used df.describe() to obtain statistical summaries of the 'total_score' column, providing a quick overview of central tendency, dispersion, and distribution of test scores among students.
  - Generated frequency tables that illustrate relationships between categorical variables.

	total_score
count	1000.000000
mean	203.136000
std	43.542732
min	65.000000
25%	175.750000
50%	202.000000
75%	235.000000
max	300.000000

	race/ethnicity	test preparation course	Frequency
0	group A	completed	32
1	group A	none	47
2	group B	completed	72
3	group B	none	133
4	group C	completed	102
5	group C	none	221
6	group D	completed	84
7	group D	none	178
8	group E	completed	45
9	group E	none	86

Investigating the relationship between 'race/ethnicity' and 'test preparation course' to observe potential correlations or patterns.

Data Mining Tool #1 EDA (continued)

	race/ethnicity	lunch	Frequency
0	group A	free/reduced	26
1	group A	standard	53
2	group B	free/reduced	70
3	group B	standard	135
4	group C	free/reduced	115
5	group C	standard	208
6	group D	free/reduced	78
7	group D	standard	184
8	group E	free/reduced	59
9	group E	standard	72

	race/ethnicity	parental level of education	Frequency
0	group A	associate's degree	11
1	group A	bachelor's degree	14
2	group A	high school	15
3	group A	master's degree	8
4	group A	some college	20
5	group A	some high school	11
6	group B	associate's degree	40
7	group B	bachelor's degree	20
8	group B	high school	39
9	group B	master's degree	19
10	group B	some college	49
11	group B	some high school	38
12	group C	associate's degree	75
13	group C	bachelor's degree	35
14	group C	high school	58
15	group C	master's degree	20
16	group C	some college	69
17	group C	some high school	66
18	group D	associate's degree	50
19	group D	bachelor's degree	29
20	group D	high school	59
21	group D	master's degree	16
22	group D	some college	57
23	group D	some high school	51
24	group E	associate's degree	27
25	group E	bachelor's degree	14
26	group E	high school	31
27	group E	master's degree	7
28	group E	some college	27
29	group E	some high school	25

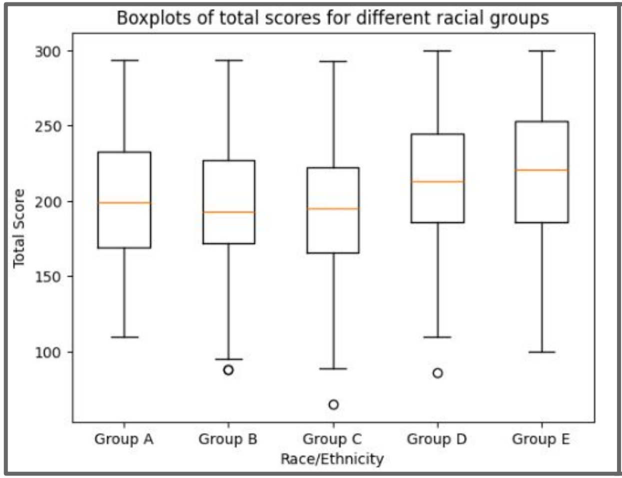
- Through frequency tables and comparative analysis, it is possible to evaluate if there's a correlation between various categories (such as parental level of education) and improved scores.

Data Mining Tool #1 EDA (continued)

Benefits of EDA:

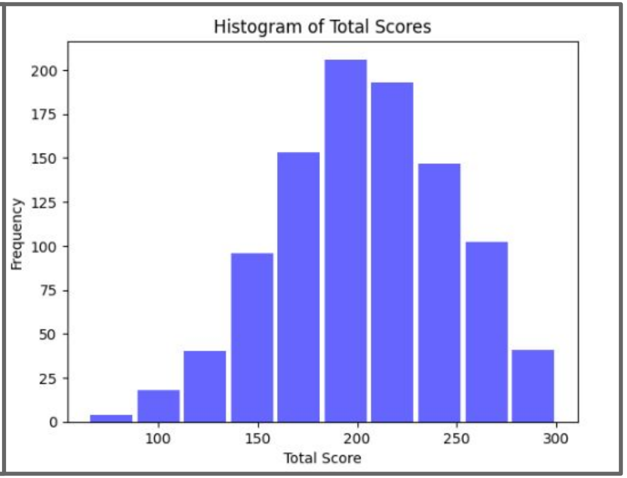
- The EDA algorithm visually and statistically explores relationships between variables, potentially revealing which factors (such as 'race/ethnicity', 'parental level of education', etc.) might have the most influence on test scores.

- The visual representations helps understand the data trends and patterns.
  - This in turn can guide recommendations for enhancing student scores, such as providing additional resources for specific demographic groups or targeting support for those not taking the test preparation course.



Box Plots:

By generating box plots for 'total_score' across different racial/ethnic groups, it visually compares the distribution of scores among these groups, identifying potential variations or disparities.



Histogram:

Creating a histogram for the 'total_score' column gives a visual representation of the distribution of scores, allowing to observe patterns, skewness, or outliers.

Data Mining Tool #2 KNN

Benefits of KNN:

- ❖ K-Nearest Neighbors (KNN) is a versatile and intuitive algorithm used in both classification and regression tasks in machine learning.
  - ➢ KNN operates on the principle that similar things are close to each other. It classifies data points based on their similarity to neighboring points.
- ❖ It helps understand which students are similar based on their characteristics (gender, race/ethnicity, parental education) and how these similarities relate to test score classifications.
- ❖ KNN can help discern which variables contribute more to the classification of 'Average/Excellent/Fail' based on test scores.

```python
# Function to convert total_score to letter grade
def score_to_grade(score):
    if score >= 250:
        return 'Excellent'
    elif score >= 170:
        return 'Average'
    else:
        return 'Fail'
```

```python
# Apply the function to update the total score from number to a letter grade
df['total_score_letter'] = df['total_score'].apply(score_to_grade)
```

- ❖ The 'total_score_letter' column is derived from the 'total_score' column using a function that categorizes the total score into three classes: 'Average,' 'Excellent,' and 'Fail' based on score ranges:
  - ➢ 'Excellent' includes scores greater than or equal to 250.
    'Average' includes scores between 170 and 249.
    'Fail' includes scores below 170.
  - ➢ Moving forward, these 'Average/Excellent/Fail' classifications serve as target variables in future algorithms.

Data Mining Tool #2 KNN (continued)

- Each categorical column ('Gender', 'Race/Ethnicity', 'Test Preparation Course',) is encoded separately into numeric representations.
  - For example, 'Gender' is encoded to '0' for female and '1' for male.
- The dataset is divided into two parts:
  - All the features except the target variables and the other
  - The target variable for classification.
- The code then iterates over different values of 'k' (number of neighbors) - specifically, for k values of 3, 5, and 10.

```python
# Initialize the LabelEncoder
label_encoder = LabelEncoder()
# Encode the 'Gender' column --> 0 for female, 1 for Male
df['gender'] = label_encoder.fit_transform(df['gender'])
df['race/ethnicity'] = label_encoder.fit_transform(df['race/ethnicity'])
df['parental level of education'] = label_encoder.fit_transform(df['parental level of education'])
df['test preparation course'] = label_encoder.fit_transform(df['test preparation course'])
df['lunch'] = label_encoder.fit_transform(df['lunch'])
```

```python
# Splitting the dataset
attr = df.drop(columns = ['total_score_letter','total_score'])  # features
target = df['total_score_letter']  # target variable
```

```python
# Splitting dataset into 30% test and 70% training data with random_state as 0
attr_train, attr_test, target_train,target_test  = train_test_split(attr, target,test_size = 0.3,train_size=0.7, random_state = 0, shuffle = True)
```

```python
# Training the knn models using k = 3,5,10 values
k_values = [3, 5, 10]

for k in k_values:
    knn = KNeighborsClassifier(n_neighbors = k)
    knn.fit(attr_train, target_train)
    target_pred = knn.predict(attr_test)
    accuracy = accuracy_score(target_test,target_pred )
    cm = confusion_matrix(target_test, target_pred)
    cr = classification_report(target_test, target_pred)
    print(f'Accuracy of model with k = {k}: {accuracy}')
    print('')
    print(f'Classification Report of model with k = {k}:\n {cr}\n')
```

Data Mining Tool #2 KNN (continued)

Results

Accuracy of model with k = 3: 0.5966666666666667

Classification Report of model with k = 3:

	precision	recall	f1-score	support
Average	0.68	0.77	0.72	197
Excellent	0.27	0.15	0.19	41
Fail	0.39	0.35	0.37	62
accuracy			0.60	300
macro avg	0.45	0.42	0.43	300
weighted avg	0.57	0.60	0.58	300

Accuracy of model with k = 5: 0.6333333333333333

Classification Report of model with k = 5:

	precision	recall	f1-score	support
Average	0.69	0.85	0.76	197
Excellent	0.29	0.15	0.19	41
Fail	0.47	0.27	0.35	62
accuracy			0.63	300
macro avg	0.48	0.42	0.43	300
weighted avg	0.59	0.63	0.60	300

Accuracy of model with k = 10: 0.63

Classification Report of model with k = 10:

	precision	recall	f1-score	support
Average	0.66	0.93	0.77	197
Excellent	0.40	0.10	0.16	41
Fail	0.18	0.03	0.05	62
accuracy			0.63	300
macro avg	0.41	0.35	0.33	300
weighted avg	0.52	0.63	0.54	300

Data Mining Tool #3 Naive Bayes Model

- Assumes Independence among the different features
  - Can still perform well without this because in real-world situations it rarely holds
- Based on Bayes' theorem, which describes the probability of an event based on prior knowledge of conditions that might be related to the event
- An NB model is easy to build and particularly useful for very large data sets (due to speed and efficiency)
- Can still perform well for sophisticated data sets despite the ease of the model

Conditional probability: Bayes' Theorem

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

Data Mining Tool #3 Naive Bayes Model (continued)

- Accuracy score could be low because we assume independence but there might be dependence among some features
  - Ex: Lunch vs Test Preparation Course
- Still has the highest accuracy score compared to all the other algorithms used on this dataset
- The range in precision among the different categories is the lowest compared to the other categories
  - Reflects a stable and uniform predictive capability of the model

```
Accuracy: 0.6533333333333333

[[181   6  10]
 [ 33   8   0]
 [ 54   1   7]]

Classification Report
              precision    recall  f1-score   support

     Average       0.68      0.92      0.78       197
   Excellent       0.53      0.20      0.29        41
        Fail       0.41      0.11      0.18        62

    accuracy                           0.65       300
   macro avg       0.54      0.41      0.41       300
weighted avg       0.60      0.65      0.59       300
```

Data Mining Tool #4 CART Algorithm

- Advantages
  - Interpretability: The resulting decision tree can be easily visualized and understood, providing interpretable rules for predictions.
  - Versatility: It uses any combination of continuous/ discrete variables.
- Disadvantages
  - The tree structure may be unstable → Small variations in the data can lead to different trees, making the model less stable

```
CART Tree Rules:
|--- lunch <= 0.50
|   |--- test preparation course <= 0.50
|   |   |--- parental level of education <= 4.50
|   |   |   |--- gender <= 0.50
|   |   |   |   |--- race/ethnicity <= 2.50
|   |   |   |   |   |--- race/ethnicity <= 0.50
|   |   |   |   |   |   |--- class: Average
|   |   |   |   |   |--- race/ethnicity >  0.50
|   |   |   |   |   |   |--- parental level of education <= 0.50
|   |   |   |   |   |   |   |--- race/ethnicity <= 1.50
|   |   |   |   |   |   |   |   |--- class: Average
|   |   |   |   |   |   |   |--- race/ethnicity >  1.50
|   |   |   |   |   |   |   |   |--- class: Excellent
|   |   |   |   |   |   |--- parental level of education >  0.50
|   |   |   |   |   |   |   |--- parental level of education <= 1.50
|   |   |   |   |   |   |   |   |--- class: Average
|   |   |   |   |   |   |   |--- parental level of education >  1.50
|   |   |   |   |   |   |   |   |--- parental level of education <= 3.00
|   |   |   |   |   |   |   |   |   |--- race/ethnicity <= 1.50
|   |   |   |   |   |   |   |   |   |   |--- class: Average
|   |   |   |   |   |   |   |   |   |--- race/ethnicity >  1.50
|   |   |   |   |   |   |   |   |   |   |--- class: Fail
|   |   |   |   |   |   |   |   |--- parental level of education >  3.00
|   |   |   |   |   |   |   |   |   |--- race/ethnicity <= 1.50
|   |   |   |   |   |   |   |   |   |   |--- class: Average
|   |   |   |   |   |   |   |   |   |--- race/ethnicity >  1.50
|   |   |   |   |   |   |   |   |   |   |--- class: Average
```

Snippet of the Regression Tree

Data Mining Tool #4 CART Algorithm (continued)

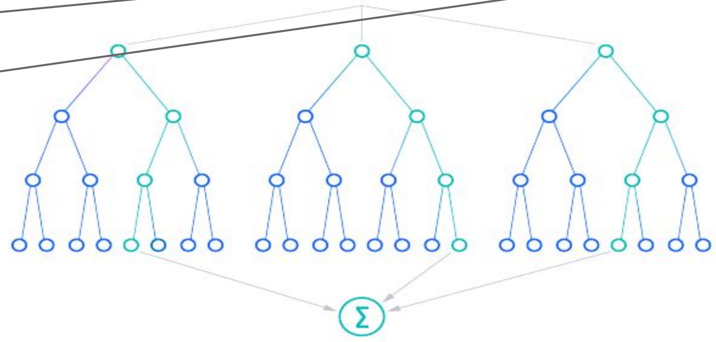
- ❖ Accuracy score is one of the lowest among the different algorithms used on this dataset
- ❖ Problem could be: overfitting
  - ➢ CART trees can be prone to overfitting (especially if the tree is allowed to grow excessively)
- ❖ After altering the max_depth size to 5 the accuracy score boosted up to 62%

```
Accuracy: 0.5933333333333334

[[161  15  21]
 [ 34   7   0]
 [ 47   5  10]]

Classification Report
              precision    recall  f1-score   support

     Average       0.67      0.82      0.73       197
   Excellent       0.26      0.17      0.21        41
        Fail       0.32      0.16      0.22        62

    accuracy                           0.59       300
   macro avg       0.42      0.38      0.38       300
weighted avg       0.54      0.59      0.55       300
```

Data Mining Tool #5 Random Forest

- Random Forest combines the output of multiple decision trees to reach a single result.
- N_estimators is one of the parameters you can modify, it controls how many trees will be used.
- Used to control the randomness of algorithms.
- Benefits
  - **Flexible**; can handle both regression and classification with high accuracy
  - **Easy to determine feature importance**; Gini importance and mean decrease in impurity (MDI) are used to measure the model's accuracy decreases when a given variable is excluded.
- Challenges
  - **Slow**; RF handle large data sets
  - **Requires more resources**; Works with large data sets

```python
# creating the model
from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=100,random_state=8)
```



`RandomForestClassifier`(*n_estimators=100, *, criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False, class_weight=None, ccp_alpha=0.0, max_samples=None*)

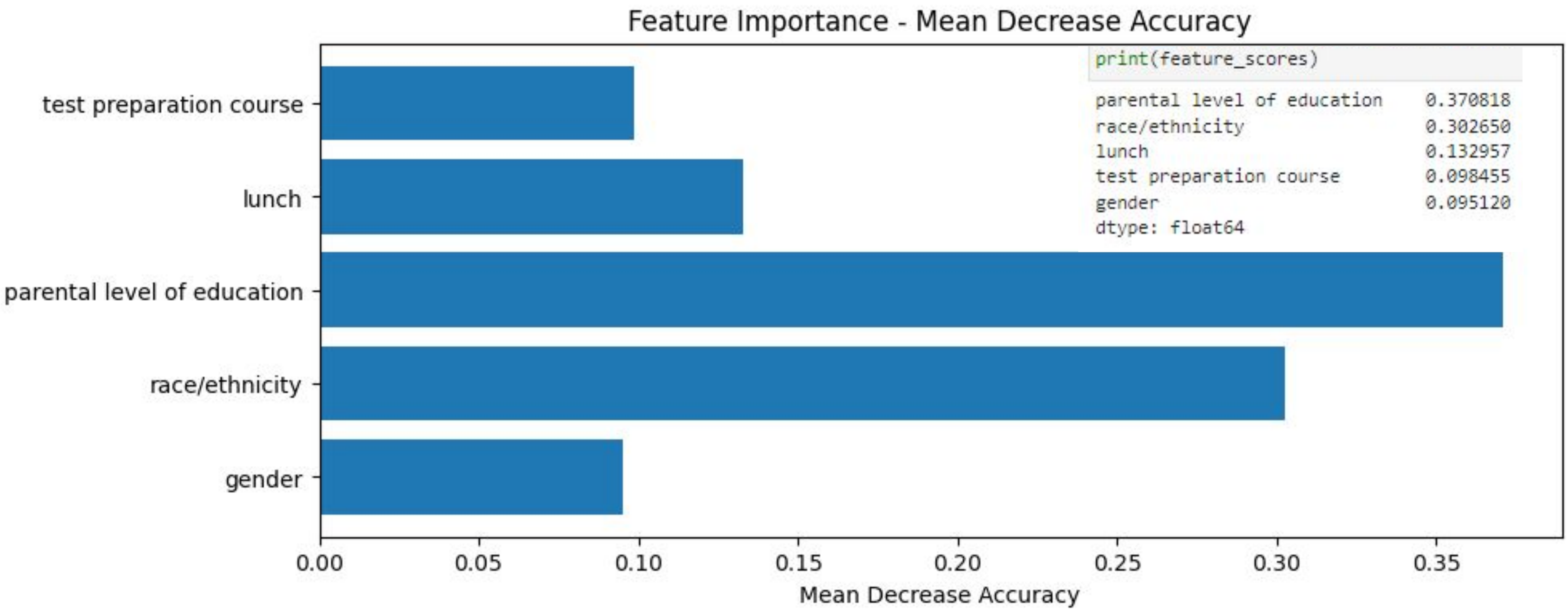
Data Mining Tool #5 Random Forest (our Results)

- Obtained a 60% accuracy using Random Forest on our data set
- Using model.feature_importances_ it enables to see which feature is the most important in higher test scores.
  - A higher mean decrease accuracy value indicates that the variable is more important
- For example, in Mean Decrease Accuracy, if we remove test Preparation course, the models accuracy decreases by about 9%, vs if we remove parental level of education, it loses about 37%

```
Accuracy= 0.5966666666666667
[[156  16  25]
 [ 31   9   1]
 [ 42   6  14]]
              precision    recall  f1-score   support

     Average       0.68      0.79      0.73       197
   Excellent       0.29      0.22      0.25        41
        Fail       0.35      0.23      0.27        62

    accuracy                           0.60       300
   macro avg       0.44      0.41      0.42       300
weighted avg       0.56      0.60      0.57       300
```

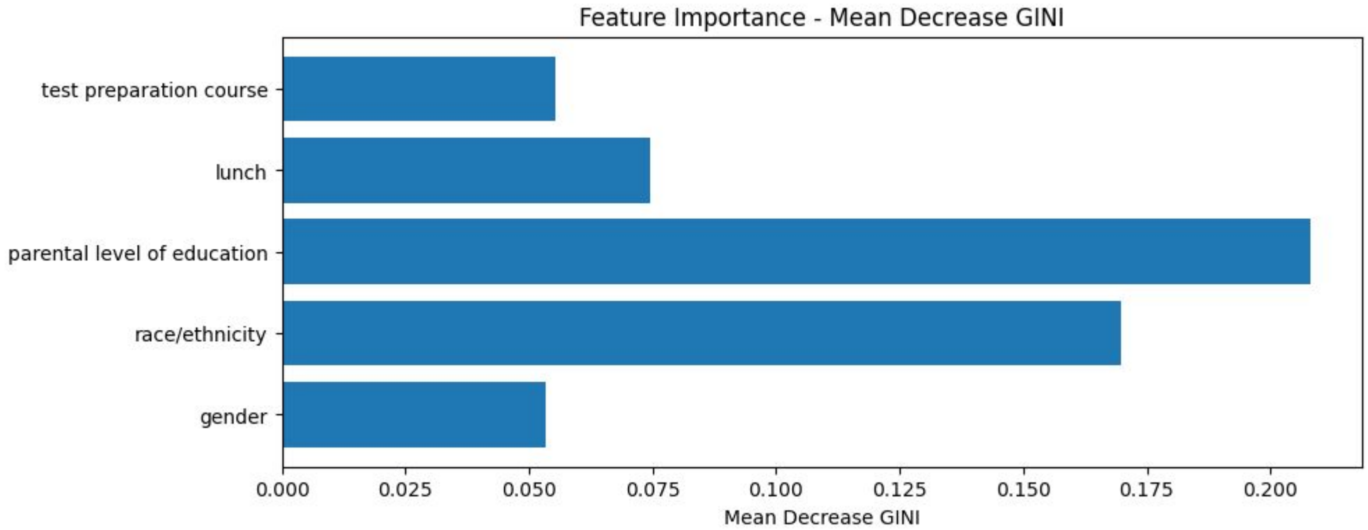
```
print(feature_scores)

parental level of education    0.370818
race/ethnicity                 0.302650
lunch                          0.132957
test preparation course        0.098455
gender                         0.095120
dtype: float64
```

Data Mining Tool #5 Random Forest (our Results)



Data Mining Tool #5 Random Forest (continued)

- With the Mean Decrease GINI, it tracks variable importance. EX. It measured how much lunch contributed to each leaf in the trees of the forest.
- Formula to calculate: total decrease of node impurity averaged over all trees
- Higher values mean higher importance

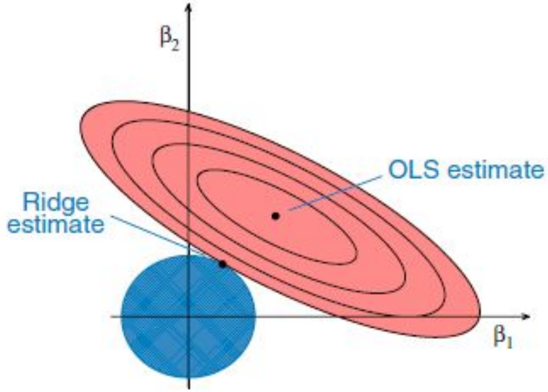


Data Mining Tool #6 Ridge Regression

- This is a model that is used when the data you are studying suffers from multicollinearity.
- Multicollinearity is a phenomenon that occurs when some independent variables in a model are correlated.
- Linear least squares with l2 regularization.
- Regularization improves the conditioning of the problem and reduces the variance of the estimates. Larger values specify stronger regularization
- Benefits
  - Good to use when dataset has too many predictors
- Challenges
  - Limited use

Ridge(*alpha=1.0*, ***, *fit_intercept=True*, *copy_X=True*, *max_iter=None*, *tol=0.0001*, *solver='auto'*, *positive=False*, *random_state=None*)

Geometric Interpretation of Ridge Regression:



Heatmap


gender
race/ethnicity
parental level of education
lunch
test preparation course
math score
reading score
writing score
total_score
1 0.063 -0.079 -0.034 -0.0077 0.2 -0.19 -0.25 -0.081
0.063 1 -0.011 -0.033 0.03 0.21 0.13 0.13 0.17
-0.079 -0.011 1 -0.034 0.0089 -0.19 -0.15 -0.16 -0.17
-0.034 -0.033 -0.034 1 -0.029 0.37 0.29 0.32 0.34
-0.0077 0.03 0.0089 -0.029 1 -0.15 -0.25 -0.32 -0.25
0.2 0.21 -0.19 0.37 -0.15 1 0.82 0.81 0.92
-0.19 0.13 -0.15 0.29 -0.25 0.82 1 0.95 0.97
-0.25 0.13 -0.16 0.32 -0.32 0.81 0.95 1 0.97
-0.081 0.17 -0.17 0.34 -0.25 0.92 0.97 0.97 1
1.0
0.8
0.6
0.4
0.2
0.0
−0.2

Data Mining Tool #6 Ridge Regression (our Results)

Data	Score
Removing Parental level of education	10.36
No scores	13.81
Just Math	92.74
Just Writing	96.09
Math+Reading	99.09
Reading+Writing	98.06
Math+Reading+Writing	99.999999

Suggestion for future work

- ❖ Recommend to change the current preparation course.
- ❖ Add student age to see if that is a factor
- ❖ Test more students from different locations and see if location has anything to do with test scores.

Results

- Which variable has the most influence on test scores
  - Individual test scores have greatest influence on total test score, Parental level of education and race/ethnicity has the next greatest impact on test scores.
- How effective is the test preparation course?
  - Very little, infact, it hindered the Students test scores.
- What would be the best way to improve student scores on each test?
  - Restructure the test preparation course.
- What patterns and interactions in the data can you find?
  - If a student has a high score in one subject, it is likely they will have high scores in the other subjects as well.
  - Gender has pretty much no effect to your test scores

References

- EDA
  - https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15
  - https://www.geeksforgeeks.org/what-is-exploratory-data-analysis/
- KNN
  - https://www.ibm.com/topics/knn
  - https://www.geeksforgeeks.org/k-nearest-neighbours/
- NB
  - https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/
- CART
  - https://maxtech4u.com/cart-algorithm-applications-advantages-disadvantages/
- RF
  - https://www.ibm.com/topics/random-forest
  - https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
- Ridge
  - https://online.stat.psu.edu/stat857/node/155/
  - https://www.investopedia.com/terms/m/multicollinearity.asp#:~:text=Multicollinearity%20is%20a%20statistical%20concept,in%20less%20reliable%20statistical%20inferences.
  - https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html