Factors Influence Asian American Educational Attainment

https://github.com/carrieeehuang/MACSS30100_Final_Project.git

Carrie Huang yhuang11@uchicago.edu

2025-03-12

Table of contents

Introduction

Educational attainment is one of the strongest predictors of social mobility, impacting life trajectories such as economic opportunities and health status. Model Minority myth is a long-held assumption that Asian Americans are homogeneously high-achieving group. While some Asian American subgroups achieve high levels of education, other face barriers linked to socioeconomic status, citizenship, and gender (Covarrubias and Liou 2014). These disparities remain underexplored, particularly in the context of predictive modeling. This study seeks to analyze what factors influence educational attainment among Asians in the United States. The objective of the study is to create a model that can precisely anticipate the highest educational level of an Asian American by categorizing them into six groups: "Less than High School", "High School Diploma", "Some College", "Bachelor's Degree", "Master's Degree", and "Doctorate". To achieve this, the study employs a supervised machine learning approach, which the educational attainment problem characterized as a ordinal classification. By applying a machine learning approach to the Annual Social and Economic Supplement (ASEC) dataset, this study aims to provide insights into the educational inequality within the Asian community. Understanding these disparities is crucial, as a more nuanced perspective can address structural barriers rather than reinforcing stereotypes.

This project want to answer the following research questions: Which model performs best in predicting educational attainment? Which features stand out as most influential? How do an individual's socioeconomic and demographic circumstances shape the odds of attaining higher education among Asians in the United States?

Literature Review

Machine Learning Models in Educational Research

Prior research has applied computational methods to analyze educational issue using multiple machine learning algorithms. Random Forest and Decsion Tree are one of the popular models used by researchers to study educational attainment. Masci, Johnes, and Agasisti (2018) study on the relationship between demographics and educational performance by applying machine learning techniques to analyze cross-national school performance data from the OECD's PISA assessments. Using Regression Trees and Gradient Boosting, they identified how student background, school characteristics, and socioeconomic factors interact to shape academic outcomes. Their findings concluded that parental education and income levels were the strongest determinants of educational success, while immigration status had varying effects depending on national context. This study supports my research in two ways: first, it validates the use of Decision Trees for educational prediction, and second, it demonstrates that socioeconomic and demographic features significantly influence academic trajectories. Rizvi, Rienties, and Khoja (2019) also used Decision Trees to analyze demographic predictors impacting online learning performance. Their study applied Random Forest and Decision Tree models to classify students based on engagement and success rates, using a dataset from online education platforms. Their Gini impurity-based feature selection identified income, gender, and age as key determinants of student outcomes, supporting the idea that socioeconomic background significantly influences educational achievement. Additionally, their findings reinforce that income and gender disparities persist across different educational settings, strengthening the argument for my focus on Asian American educational inequalities. Akmeşe, Kör, and Erbay (2021) also applied Random Forest to predict student performance in higher education settings. Using a dataset of socio-demographic variables, such as age, gender, family income, and parental education, they found that Random Forest outperformed all other models, achieving an accuracy rate of 89%.

Other studies related to education modeling used Logistic Regression for analysis. Chang (2006) compared Logistics Regression, Classification and Regression, and Neural Networks to explore the college admission result. The study finds that the key predictors include region, communication frequency, academic performance and major choice. Classification and Regression and Neural Networks outperform Logistic Regression, having prediction accuracies of around 75%. Antons and Maltz (2006), studying the similar topic, utilized logistic regression, decision trees, and neural networks, and their key features include financial aid data, student demographics, and academic performace to enhence prediction accuracy. Since their final work aimed to develop a model for Admissions Office, a logistic regression was chosen due to the interpretability by college staff. In addition, Decision Tree and Neutral Networks were used for feature selection and refine logistics regression inputs. Basu et al. (2019) applied seven supervised machine learning algorithms to predict college enrollments. For their study, logistic regression was the best performing model, and the most important predictors were GPA, campus visit indicator, high school class size, reader academic rating, and gender. Martinez et al.(2024) examined the higher education drop out rate by at-risk students in the United States. They test various machine learning models, including Support Vector Machines (SVM), Naive Bayes, K-nearest neighbors (KNN), Decision Trees, Logistic Regression, and Random Forest, to predict which students are likely to drop out or underperform. Socioeconomic and demographic features—along with academic history—are key predictors. While all the models perform well with more than 78% accuracy, there are slight variations in the prediction accuracy. The top model was Naive Bayes, generating an outcome with more than 89% accuracy (Martinez, Sood, and Mahto 2024).

In two recent studies, researchers have employed XGBoost to address education-related questions. Cheng et al. classify and predict the students' performance by examining and comparing the machine learning and artificial neural network assessments (2024). They applied five models: Random Forest Classifier, the Decision Tree Classifier, the K Neighbors Classifier, the MLP Classifier, and the XG-Boost Classifier. Their results shows that XG-Boost represented the best performance (Cheng, Liu, and Jia 2024). Asselman et al. (2023) propose a new PFA approach based on different models, Random Forest, AdaBoost, and XGBoost, in order to increase the predictive accuracy of student performance. The experimental results show that the scalable XGBoost has outperformed the other evaluated models and substantially improved the performance prediction compared to the original PFA algorithm (Amal Asselman and Aammou 2023). Taken together, these studies establish a strong foundation for applying machine learning to educational prediction. Building on previous researches, this study adopted an ordinal classification approach to model the highest educational attainment among Asian Americans (Cirelli et al. 2018).

Asian American Educational Attainment

Public discourses on Asian American educational attainment often cited structural factors as evidence of the model minority myth. Socioeconomic conditions, immigration policies, and community resources all interact to create disparities within Asian population in the United States and account for the racial gap in educational success. Structural factors challenge the oversimplified "model" portrayal in public conversations by addressing systematic factors. However, these narratives still fail to address the complexities and diversity of Asian communities.

Policies on Asian immigration have been a critical structural factor shaping the public narrative about students. The Immigration Act of 1965 resulted in a large wave of Asian immigration, and the policy favored highly educated and skilled professionals (Lee and Zhou 2015). Lee and Zhou's research examines the strikingly high educational attainment of Asian Americans and argues that this phenomenon is deeply shaped by structural factors, particularly US immigration policies (2015). They introduce hyper-selectivity, which refers to the way US immigration policies favor highly educated individuals from certain Asian countries, such as China, India, and South Korea (Lee and Zhou 2015). Many post-1965 Asian immigrants came with higher levels of education than native-born Americans and non-migrating peers in their countries of origin (Lee and Zhou 2015). One of their central arguments is that hyper-selectivity fosters the narrative of the "success frame" (Lee and Zhou 2015). It is a rigid definition of achievement that frames the educational and career aspirations of second-generation Asian Americans. Parents, peers, and ethnic networks continuously reinforce this success frame. Not meeting the standard can bring a source of disappointment and even shame. Beyond the "success frame," another critical consequence of hyper-selectivity is "ethnic capital," referring to the establishment of community institutions, such as tutoring programs and networks (Lee and Zhou 2015). Thus, ethnic capital benefits the children of highly educated-Asian immigrants and children from less-educated families. The social infrastructure helps maintain high academic expectations, encourages competitive educational environments, and reinforces the success frame. Summing up, Lee and Zhou argue that the hyper-selectivity of Asian immigrants leads families to take on the "success frame," which is supported by public resources at school and "ethnic capital" in the community.

Furthermore, the ethnic community's social structure plays a pivotal role in educational outcomes. Through interviewing Chinese and Korean immigrant communities, Zhou and Kim find that education-related information circulates in these immigrant families through ethnic institutions, including Korean churches and community centers (2006). Hence, Chinese and

Korean immigrant families have significant social capital generated by the social structure. These ethnic communities' supplementary education for Chinese and Korean American children and help them "gain entrance to prestigious colleges in disproportionately large numbers" (Zhou and Kim 2006).

Prior studies have examined how Asian American parents make deliberate choices regarding schooling, extracurricular activities, and career plans to maximize their children's opportunities for social mobility. These strategies reflect a thoughtful response to the structural constraints Asian Americans face in the labor market. Dhingra investigates the trend of extracurricular academic enrichment among middle- and upper-middle-class Asian American families (2020). He delves into the growth of after-school learning centers, participation in spelling bees, and math competitions, which he termed "hyper education" (2020). The research reveals that both Asian American and White families engage in these practices to secure a competitive edge in the educational landscape (Dhingra 2020). This competitive aspect of education stems from Asian parents' belief that their children will be primarily compared with other high-achieving Asian students during college admissions instead of assessed in comparison with all peers across racial groups. This phenomenon reflects a broader societal emphasis on high-achievement education, where standard schooling is perceived as insufficient for ensuring academic and professional success.

However, the educational success of these communities can mask the structural barriers faced by less-resourced families. For example, middle-class Chinese and Korean communities benefit significantly from community networks and corresponding educational resources, while Vietnamese refugees have little economic or social capital to have community resources to support additional education for children (Zhou and Bankston 1998). These refugee families quickly established "tight-knit" communities, and this community solidarity contributes to children's high academic performance and ability to maneuver in American society (Zhou and Bankston 1998). This uneven access to community resources reflects the disparity in how structural factors impact different subgroups of Asian Americans. Structural factors, such as selectivity and community networks, further complicate the picture. Structural realities provide critical reasoning for success while perpetuating disparities among Asian subgroups.

Hypothesis

Prior research has demonstrated that family background, income, and immigrant status play a crucial role in shaping academic trajectories (Covarrubias and Liou 2014; Masci, Johnes, and Agasisti 2018; Rizvi, Rienties, and Khoja 2019; Akmeşe, Kör, and Erbay 2021; Antons and Maltz 2006). This study aims to develop a predictive model for educational attainment by applying an ordinal classification approach to the Annual Social and Economic Supplement (ASEC) dataset.

Main Hypothesis

- Null Hypothesis: There is no significant relationship between socioeconomic and demographic factors and the highest educational attainment of an Asian American individual.
- Alternative Hypothesis: Socioeconomic and demographic factors significantly influence the likelihood of attaining a higher educational level among Asian Americans, following an order from less than high school to doctorate.

Specific Factor Hypothesis

To further illustrate the role of individual factors, the following sub-hypotheses will be tested:

Personal Income

- Null Hypothesis:Income has no significant impact on educational attainment
- Alternative Hypothesis: Higher income is associated with a greater likelihood of obtaining a bachelor's degree or higher.

Immigrant Status

- Null Hypothesis: There is no significant difference in educational attainment between first-generation and later-generation Asian Americans.
- Alternative Hypothesis: Asian Americans who are foreign born and become US citizen by naturalization exhibit different educational attainment patterns compared to US born citizens, potentially facing structural barriers or cultural expectations influencing their education trajectory.

Country of Origin

- Null Hypothesis: There is no significant difference in educational attainment between Asian Americans from different country of origins.
- Alternative Hypothesis: Asian Americans immigrated from different Asian countries exhibit different educational attainment patterns compared, potentially influenced by the differences between hyper-selectiviy characteristics and refugee status.

Methodology

Data Sources

The Annual Social and Economic Supplement (ASEC), also called the March Supplement of the Current Population Survey, is an extensive census with more than 829 variables. In 2023, ASEC was given to around 70,000 household and the result data set has approximately 142,000 records. For this research's purposes, the ASEC provides educational attainment data for all major racial populations in the United States. Thus, I obtained 9,875 observations filtered for Asian individuals, ensuring a sufficiently large sample for training a machine learning model. Some of the data will be used to test the accuracy of the model, and some of it will be used as training data.

Workflow of this project

First, EDA is conducted to gain insights into the dataset and prepare the data for modeling. I will check for missing values, data types, and distributions. Descriptive statistics and visualizations will be used to examine the key variables across different educational attainment levels. Additionally, a correlation matrix will be generated.

Second, a feature selection process is applied to retain the most relevant predictors and remove recoded or low-impact features. I will first do hierarchical clustering on the correlation matrix to group highly correlated features to avoid redundancy. Then I will use Decision Tree to feature VIP and identify key variables influencing educational attainment. Finally, Lasso regression will be used to further reduce the irrelevant predictors to zero.

Data

Load and inspect data

```r
data = read.csv('pppub23.csv')
# head(data)
```

To obtain a dataset with Asian records, I want to only filter out the Asian data set, using the PRDTRACE column and select rows = 4 (Asian). I obtained 9,875 observations filtered for Asian individuals, ensuring a sufficiently large sample for training a machine learning model. Some of the data will be used to test the accuracy of the model, and some of it will be used as training data.

```r
df_asian = data[data$PRDTRACE == 4, ]
```

```
[1] 9875  829
```

In this dataset, there are many columns which are record identifiers, allocation flags, or survey administration columns. Therefore, I want to remove these columns from the dataset so that they do not intervene my models. In sum, about 300 columns are removed based on the naming rules in the codebook.

```r
cols_record_id = grep("(SEQ|IDNUM|RECORD|FILEDATE)", names(df_asian), ignore.case = TRUE, val
cols_alloc_flag = grep("^I_", names(df_asian), value = TRUE)
cols_alloc_flag2 = grep("^PX", names(df_asian), value = TRUE)
cols_id2 = grep("_ID$", names(df_asian), value = TRUE)
id = c("H_IDNUM", "TAX_ID", "SPM_ID")
job = grep("^WE", names(df_asian), value = TRUE)
job2 = grep("^WK", names(df_asian), value = TRUE)
cols_to_remove = unique(c(cols_record_id, cols_alloc_flag, cols_alloc_flag2, id, job, job2))
# print(cols_to_remove)
```

```r
df_asian_clean = df_asian[, !(names(df_asian) %in% cols_to_remove)]
```

```r
dim(df_asian_clean)
```

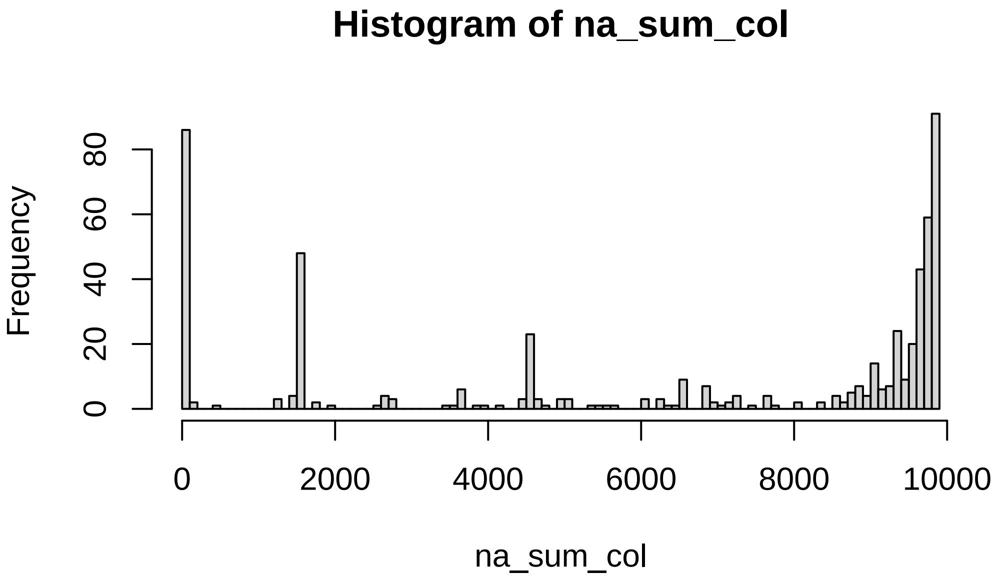
```
[1] 9875  544
```

From the codebook, I know that negative values and zero means not in the scope or missing values. Thus, I will convert the negatives and zeros to NA.

```r
df_asian_clean = df_asian_clean |> mutate(across(
.cols = where(is.numeric),
.fns = ~ ifelse(. <= 0, NA, .) ))
```

After converting the negative and zero values, I check for the number of NA in columns and rows, and removed columns with more than 80% missing values.

```r
nobs = ncol(df_asian_clean)
```



```
cols_with_many_missing = na_sum_col > (nobs *0.2)
print(table(cols_with_many_missing))
```

```
cols_with_many_missing
FALSE  TRUE 
   86   458 
```

```
# cols_with_many_missing
# df_asian_clean = df_asian_clean[!cols_with_many_missing]
```

During this process, some important columns have been dropped. For instance, the educational attainment column (A_HGA) was dropped due to missingness, probability because zero means children in the dataset. Also, the income column (PEARNVAL) was dropped. Since they are important features in my project, I am adding them back.

```
df_asian_clean$A_HGA = df_asian$A_HGA
df_asian_clean$PEARNVAL = df_asian$PEARNVAL
```

```
na_sum_row = df_asian_clean |> is.na() |> rowSums()
# hist(na_sum_row)
```

```
dim(df_asian_clean)
```

```
[1] 9875  544
```

After cleaning the dataset, we have 9,875 Asian records and 90 columns of features.

Explore Important Features

Educational Attainment

The dependent variable for this study is highest educational attainment. I will operationalize this variable by using the "A_HGA" variable in ASEC with 16 choices. I will a recode and group these categories into six groups: Less than High School, High School Diploma, Some College, Bachelor's Degree, Master's Degree, and Doctorate.

```r
# df_asian_clean$A_HGA
```

```r
unique(df_asian_clean$A_HGA)
```

```
 [1] 40 39 46 44 43 45  0 42 36 34 35 31 33 38 37 41 32
```

```r
df_asian_clean = df_asian_clean |>
  filter(A_HGA != 0) |>
  mutate(edu = case_when(
    A_HGA < 39 ~ "Less than high school",
    A_HGA == 39 ~ "High school diploma",
    A_HGA == 40 ~ "Some college but no degree",
    A_HGA %in% c(41, 42, 43) ~ "Bachelor's degree or Associate degree",
    A_HGA %in% c(44, 45) ~ "Master's degree or Professional school degree",
    A_HGA == 46 ~ "Doctorate degree",
  ))
```

I added this functin to fix the order of the education category.

```r
df_asian_clean = df_asian_clean |>
  mutate(edu = factor(edu,
                      levels = c("Less than high school",
                                 "High school diploma",
                                 "Some college but no degree",
                                 "Bachelor's degree or Associate degree",
                                 "Master's degree or Professional school degree",
                                 "Doctorate degree"),
                      ordered = TRUE))
```

```r
df_asian_clean |> filter(!is.na(edu)) |> ggplot(aes(x = edu)) +
  geom_bar() +
  labs(title = "Figure 1: Bar Plot of Educational Attainment")+
  coord_flip()
```

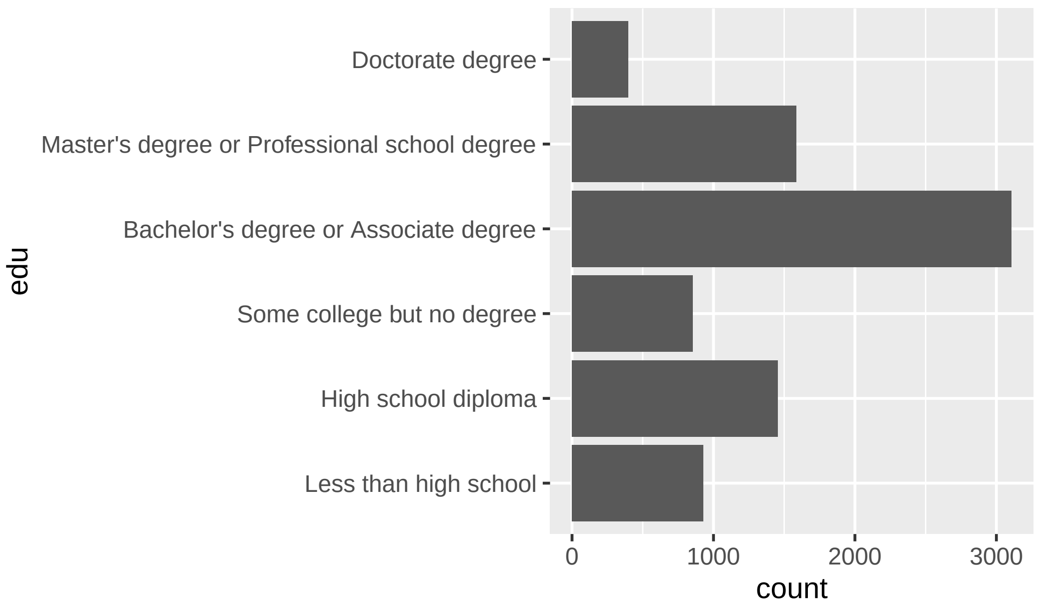


Figure 1: Bar Plot of Educational

```r
cat = table(df_asian_clean$edu)
```

```r
pie(cat,
    main = "Figure 2: Pie Chart of Educational Attainment",
    col = hcl.colors(length(cat), "BluYl"),)
```

Figure 2: Pie Chart of Educational Attainment

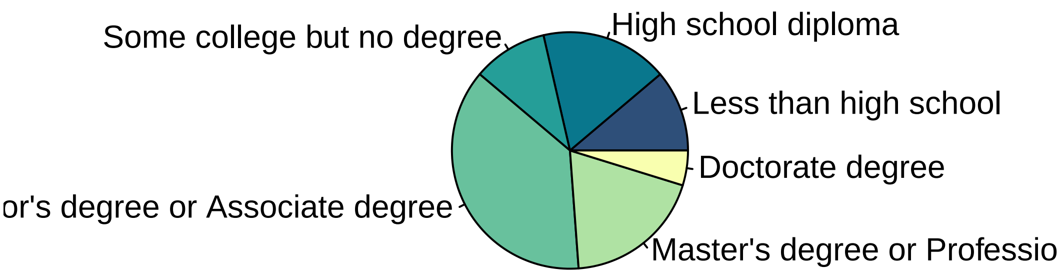


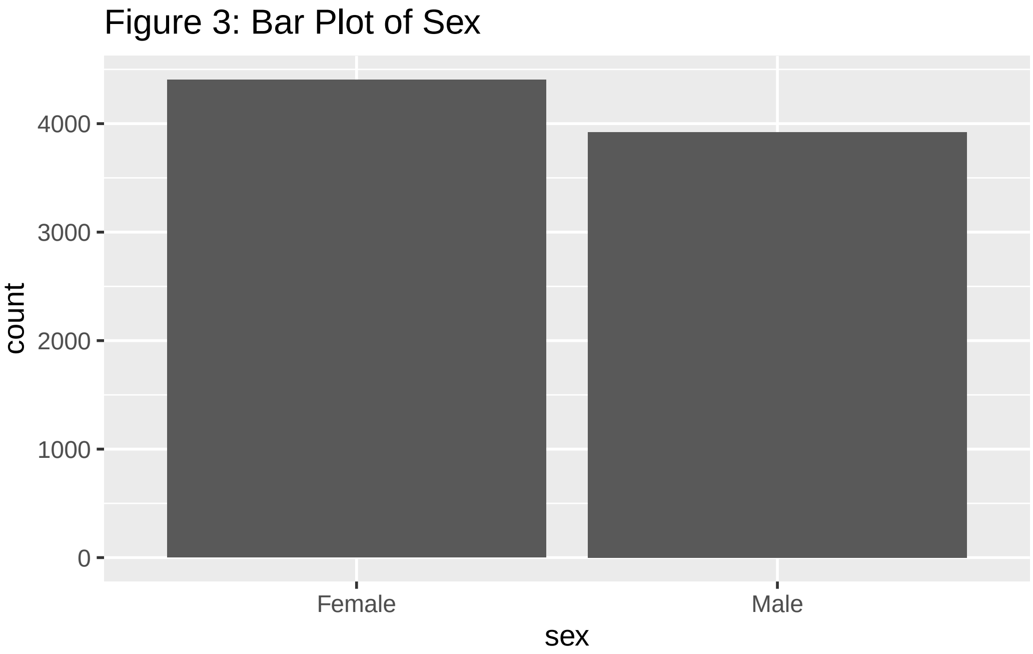
Figure 1 is the distribution of Asian American's educational attainment and Figure 2 is a pie chart, which displays the proportion of each educational level. From Figure 1 and Figure 2, most Asian Americans have a college diploma. Interestingly, more people are master graduate than whom do not go to college (see Figure 1). This result aligns with current literature which analyzes Asian Americans' high education attainment.

Sex

I looked at sex because literature has shown gender places a critical role in shaping educational attainment (Covarrubias and Liou 2014). In addition, gender norms may socialize people into different educational journey.

```
df_asian_clean$sex = case_match(df_asian_clean$'A_SEX',
1 ~ "Male",
2 ~ "Female"
)
```

```
df_asian_clean |> filter(!is.na(sex)) |> ggplot(aes(x = sex)) +
geom_bar() +
labs(title = "Figure 3: Bar Plot of Sex")
```



In this dataset, we have slightly more female Asian records than male (see Figure 3).

```
ggplot(df_asian_clean, aes(x = sex, fill = edu)) +
  geom_bar(position = "fill") +
  labs(title = "Figure 4: Education Distribution by Citizenship Status",
       x = "Sex",
       y = "Proportion",
       fill = "Education Level") +
  theme_minimal()
```

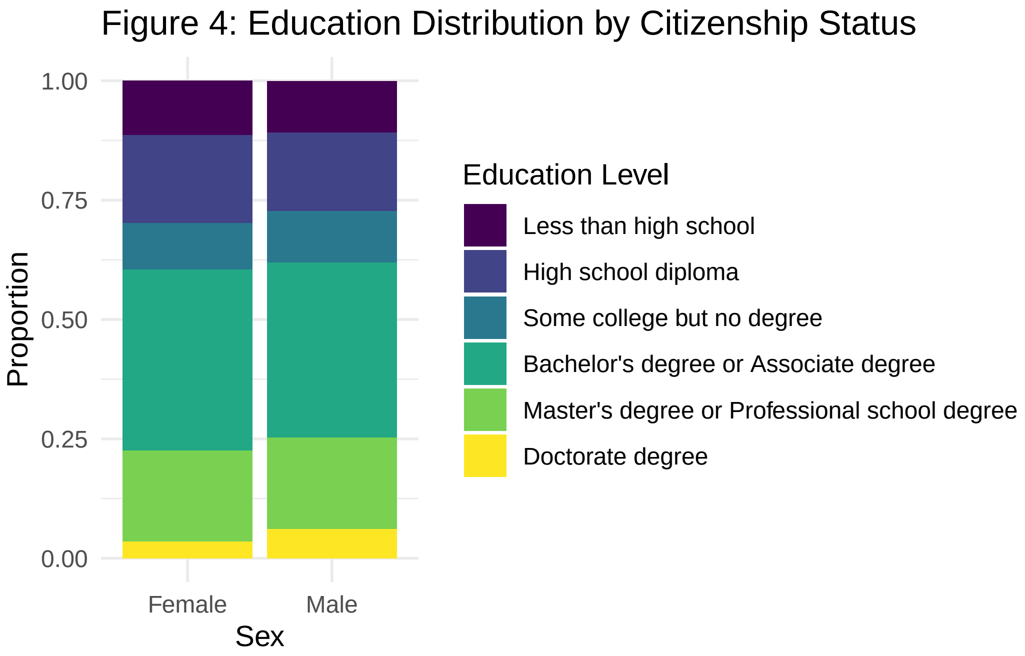


Figure 4 is a breakdown of educational attainment by sex. Men are more likely to pursue post-graduate education than women, and more women do not go to college after high school. A possible explanation is the gender socialization and biases, where girls are less likely to continue education and boys are seen with more potentials.

Citizenship status

In the dataset, the citizenship group have 5 categories, which are Native, born in US, Native, born in PR or US outlying area, Native, born abroad of US parent(s), Foreign born, US cit by naturalization, and Foreign born, not a US citizen. Since Asian American educational attainment is largely shaped by the structural factors such as immigration policies and hyper-selectivity, I grouped the five categories into four, representing the generational differences. Past literature has shown how hyper-selected Asian immigrants has high educational attainment comparing with their non-immigrant peers at home, and native US population (Lee and Zhou 2015). I hope the “Not US Citizen” and “First-Gen immigrant” can capture this trend. In addition, Asian parents who follow the “success frame” are more likely to invest in children’s education, ensuring education success for their second-gen children (Dhingra 2020).

```
df_asian_clean$citizenship = case_match(df_asian_clean$'PRCITSHP',
1 ~ "Native",
2 ~ "Native",
3 ~ "Second-Gen Immigrant",
4 ~ "First-Gen Immigrant",
5 ~ "Not US Citizen"
)
```

```
df_asian_clean |> filter(!is.na(citizenship)) |> ggplot(aes(x = citizenship)) +
geom_bar() +
labs(title = "Figure 5: Bar Plot of Citizenship Status")
```

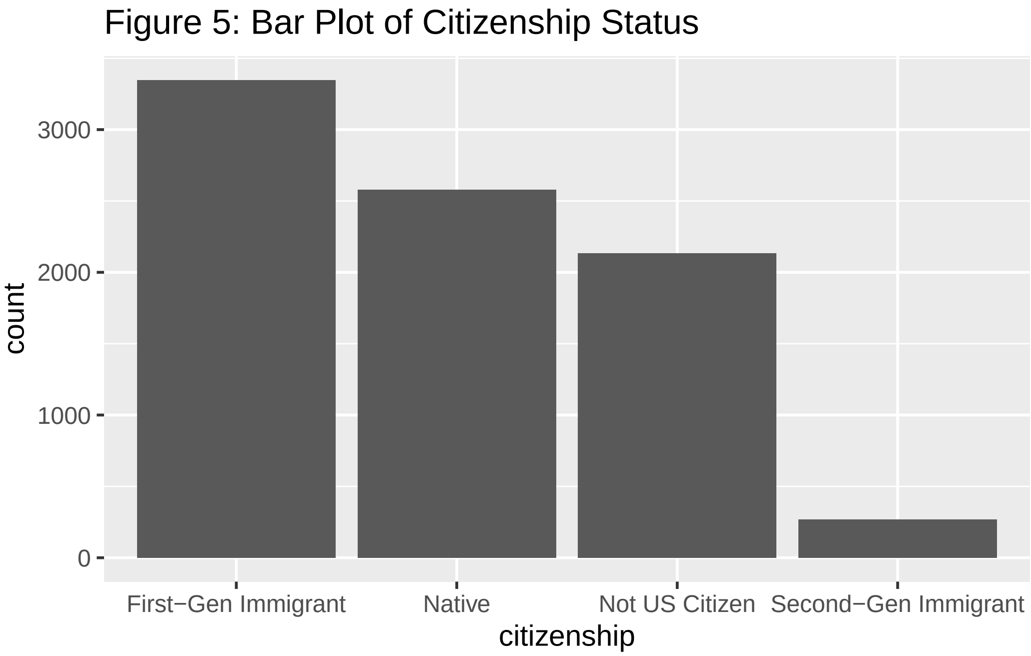


Figure 5 shows that in this dataset, most records are first-gen Asian immigrants, followed by Asian Americans who are over second-generation living in US, and Asian immigrants who are not US citizens. There is less records for second-gen immigrants in this dataset.

```
ggplot(df_asian_clean, aes(x = citizenship, fill = edu)) +
  geom_bar(position = "fill") +
  labs(title = "Figure 6: Education Distribution by Citizenship Status",
       x = "Citizenship Status",
       y = "Proportion",
       fill = "Education Level") +
  theme_minimal()
```

Figure 6: Education Distribution by Citizenship Status

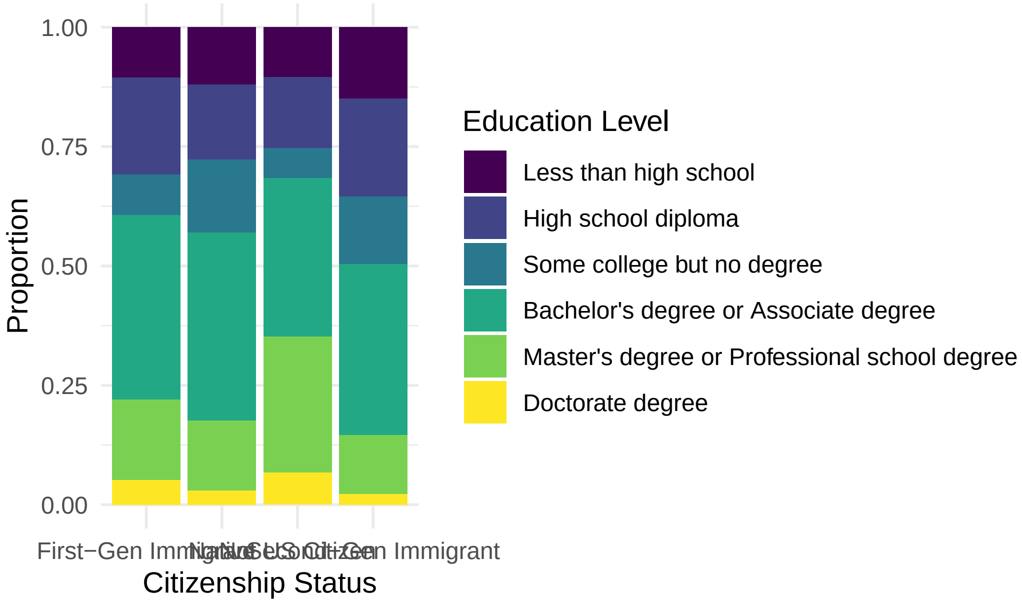


Figure 6 is a breakdown of educational attiainment by immigration status. This graph is interesting that non-US citizens are more likely to get advanced degree, which resonates with the hyper-selectivity of immigrant policies in the US. Following that is the first-gen immigrants, which may have the same hyper-selectivity characteristics. Second-gen immigrants are more least likely to go to college or drop out, which is probably due to assimilation.

Personal Income

The dataset does not have a class marker, so I use income to serve as a proxy. I categorize it into five relatable groupings: Lower, lower-middle, middle, upper-middle, upper class, using a conventional scale of socioeconomic status. Income is a important predictor for education due to reasons including family investment in extracurricular activities and housing selections, which explicitly link to access to good public schools and college admissions. Social and cultural capital is also linked to socioeconomic status, which infer more explore to resources.

```
[1]     0  5700     0 60000 32000 40000
```

```
df_asian_clean$income = cut(df_asian_clean$'PEARNVAL',
                            breaks = c(0, 30000, 58000, 94000, 153000, Inf),
                            labels = c("Low", "Lower-Middle", "Middle", "Upper-Middle", "Hig
                            right = TRUE)
```

```
[1] <NA>         Low          Middle       Lower-Middle High
[6] Upper-Middle
Levels: Low Lower-Middle Middle Upper-Middle High
```

```r
df_asian_clean = df_asian_clean |> drop_na(income)
```

```r
df_asian_clean = df_asian_clean |>
  mutate(income = factor(income,
                         levels = c("Low", "Lower-Middle", "Middle", "Upper-Middle", "High"),
                         ordered = TRUE))
```

```r
df_asian_clean |> filter(!is.na(income)) |> ggplot(aes(x = income)) +
geom_bar() +
labs(title = "Figure 7: Bar Plot of Class Distribution")
```

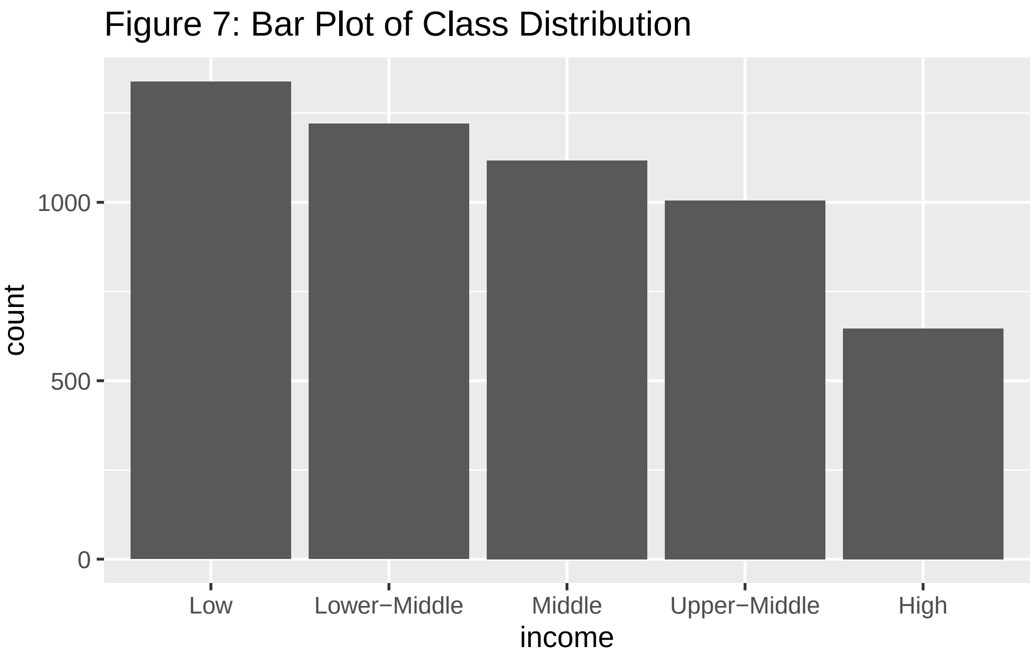


Figure 7 illustrates the overall SES distribution of Asians in this dataset. The number of people decreases as the income level, a proxy for social class, increases.

Does higher income lead to more education attainment?

```r
ggplot(df_asian_clean, aes(x = income, fill = edu)) +
  geom_bar(position = "fill") +
  labs(title = "Figure 8: Education Distribution by Income Category",
       x = "Income Category",
       y = "Proportion",
       fill = "Education Level") +
  theme_minimal()
```

Figure 8: Education Distribution by Income Category

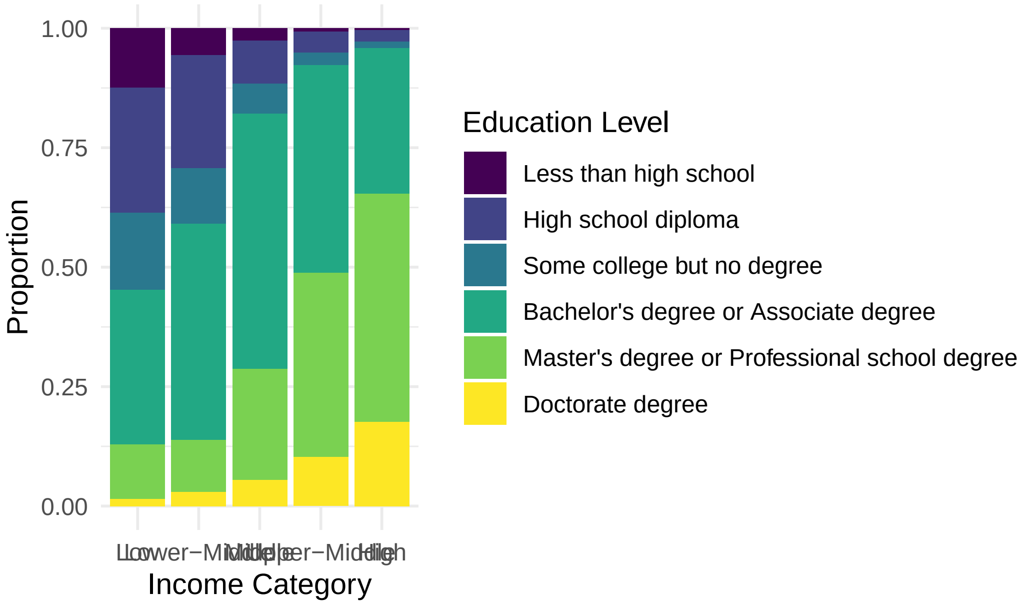


Figure 8 shows the educational attainment by SES. We can see that with income increases, less people only have high school degree, and more people with advanced degrees.

Country of Origin

```
df_asian_clean$country = case_match(df_asian_clean$'PRDASIAN',
1 ~ "Indian",
2 ~ "Chinese",
3 ~ "Filipino",
4 ~ "Japanese",
5 ~ "Korean",
6 ~ "Vietnamese",
7 ~ "Other Asian"
)
```

```
df_asian_clean |> filter(!is.na(country)) |> ggplot(aes(x = country)) +
geom_bar() +
labs(title = "Figure 9: Bar Plot of Asian Subgroup")
```

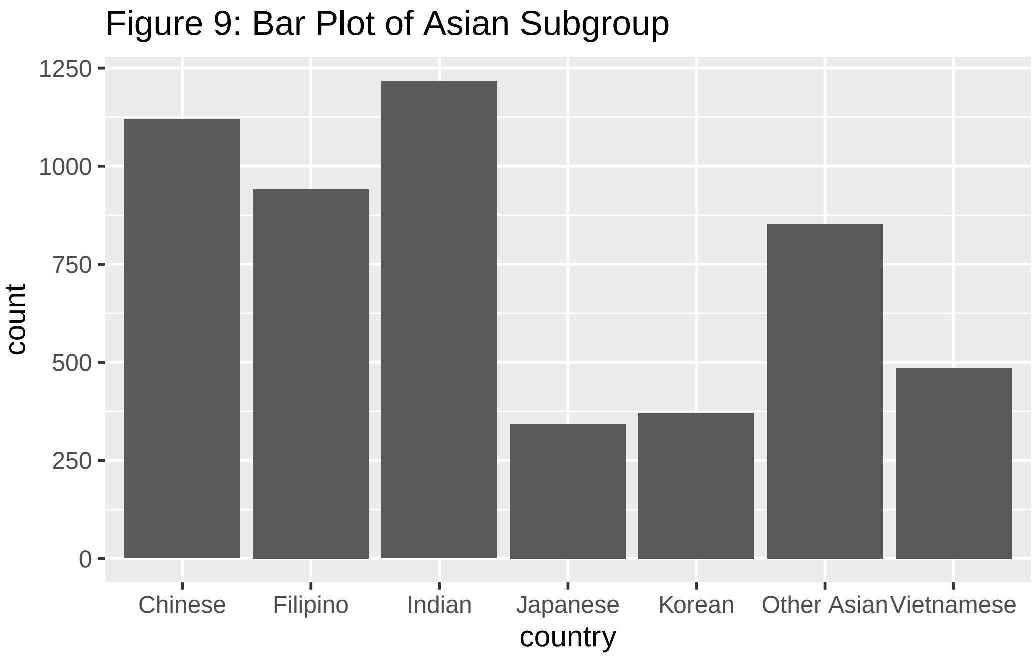


Figure 9 shows distribution of Asian home country in this dataset. Indians, Chinese, and Fillipinos are most represented in this dataset, which follows a similar population distribution in the US.

```
ggplot(df_asian_clean, aes(x = country, fill = edu)) +
  geom_bar(position = "fill") +
  labs(title = "Figure 10: Education Distribution by Country of Origin",
       x = "Country of Origin",
       y = "Proportion",
       fill = "Education Level") +
  theme_minimal() +
  coord_flip()
```

Figure 10: Education Distribution by Country of Origin

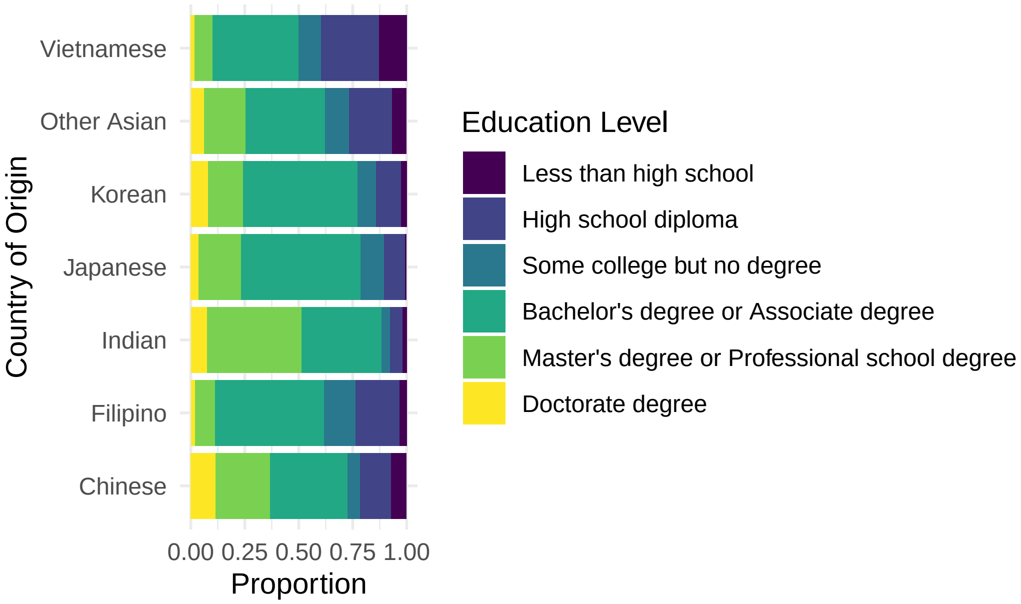


Figure 10 shows that Indians are the most highly educated. More than 50% of Indian Americans have Master's degree or equivalent, and around 80% have a bachelor's degree. Also, More than 50% of Vietnamese Americans did not go to college. And only a small portion went to graduate school and beyond. Overall, this result reflects the high educational attainment in Asian community.

Feature Selection

I started by grouping together features that were very similar using hierarchical clustering on their correlation matrix, so I wouldn't keep multiple versions of the same information. Then, I ran LASSO within each group to zero out unhelpful features and keep only the ones that mattered. Finally, I added back in some important categorical variables (such as country and citizenship) that I specifically wanted in the model, converting them to dummy columns. This way, I ended up with a smaller, more focused set of predictors that still included the key factors I cared about.

Hierarchical Clustering

I will need to convert the categorical variables in my hypothesis, the ones that I just cleaned and transformed in EDA, into dummy variables.

```
[1] "sexFemale"                      "sexMale"
[3] "citizenshipNative"              "citizenshipNot US Citizen"
```

```
 [5] "citizenshipSecond-Gen Immigrant" "countryFilipino"
 [7] "countryIndian"                   "countryJapanese"
 [9] "countryKorean"                   "countryOther Asian"
[11] "countryVietnamese"
```

```
df_asian_clean$income_num = as.numeric(df_asian_clean$income)
df_asian_clean$edu_num = as.numeric(df_asian_clean$edu)
```

```
excluded_cols = c("edu_num", "A_SEX", "PRCITSHP", "PEARNVAL", "PRDASIAN", "sex", "citizenshi
```

I will first do the correlation matrix for numeric variables.

I will first scale the data. Then, I want to exclude original columns that I just transformed for clustering to avoid redundancy.

```
numeric_cols = names(df_asian_clean)[sapply(df_asian_clean, is.numeric)]
cols_to_keep = setdiff(numeric_cols, excluded_cols)
df_for_corr = df_asian_clean[, cols_to_keep]
```

Scale the data

```
df_for_corr = scale(df_for_corr)
df_for_corr = as.data.frame(df_for_corr)
```

```
df_for_corr = cbind(df_for_corr, encoded_cats)
```

Remove near zero variance column.

```
nzv_cols = nearZeroVar(df_for_corr, freqCut = 95/5)
df_filtered = df_for_corr[, -nzv_cols]
```

I am checking if there are too many missing values, since it may affect correlation.

```
threshold = 0.2 * nrow(df_for_corr)  # 20% missing
df_for_corr = df_for_corr[, colSums(is.na(df_for_corr)) < threshold]
```

There are many colms with some missingness, so my clustering has errors . So I need to fix that first:

```
cor_matrix = cor(df_for_corr, use = "pairwise.complete.obs")

valid_columns = colnames(cor_matrix)[!is.na(colSums(cor_matrix))]
cor_matrix = cor_matrix[valid_columns, valid_columns]
# dim(cor_matrix)
```

```
cor_matrix = cor_matrix[, colSums(is.na(cor_matrix)) < nrow(cor_matrix) * 0.2]
# dim(cor_matrix)
```

Replace NA with column means:

Since using hierarchical clustering here is for feature selection, my goal is to group highly correlated features and remove redundancy. I will use Complete Linkage, which produce well-seperated clusters and help identify strongly correlated variables.

Now we need to decide the number of feature clusters.

```
inconsistency_values = cutree(hc.complete,
                              h = quantile(hc.complete$height, 0.8))
# table(inconsistency_values)
```

I experiment with different cutting height, cut at 75th percentile gives 22 clusters, and many clusters have only 1-2 features, so I need to find larger clusters. Finally, I settled at 80th percentile. From the cutree result, I got 20 clusters, which is a reasonable range and we will do more feature selections with decision tree and lasso.

```
plot(hc.complete, main = "Figure 11: Dendrogram of Features",
     xlab = "", sub = "", cex = 0.6)
rect.hclust(hc.complete, k = 20, border = "red")
```

Figure 11: Dendrogram of Features

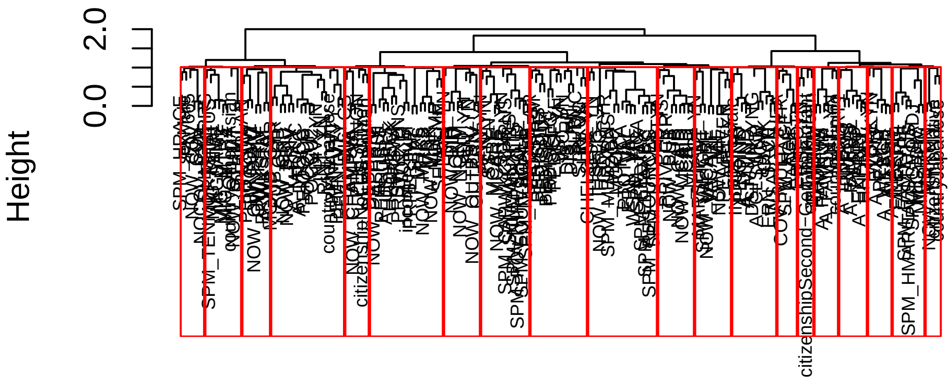


Figure 11 is an illustration of the process of cutree in hierarchical clustering. Each of these red boxes, signaling each cluster I have formed from 80th percentile cutree. Now having defined each cluster, I will run LASSO to reduce dimension to set unhelpful features to zero and keep only the ones that mattered.

LASSO

```r
lasso_results = list()
lasso_coef = data.frame()
```

```r
  # I add this helper function because x has missing values,
  # and glmnet requires a matrix with not missingness.
impute_mean_safe = function(x) {
  if(all(is.na(x))) return(rep(0, length(x)))
  if(anyNA(x)) x[is.na(x)] = mean(x, na.rm = TRUE)
  return(x)
}
```

```r
# Apply LASSO to each cluster
for (i in seq_along(cluster_list)) {
  features = cluster_list[[i]]

  # if cluster is empty, skip
  if(length(features) == 0) next

  # Create matrix from features in this cluster
  x_cluster = df_for_corr[, features, drop = FALSE]

  # Handle missing values using the helper function
  x_cluster = as.matrix(apply(x_cluster, 2, impute_mean_safe))

  # Skip if all features are zero or NA
  if(ncol(x_cluster) == 0 || all(is.na(x_cluster))) next

  # Apply LASSO
  tryCatch({
    cv_fit = cv.glmnet(x_cluster, y, alpha = 1, family = "gaussian")

    # Extract coefficients
    coef_cv = coef(cv_fit, s = "lambda.1se")
    coef_dense = as.matrix(coef_cv)
    # Create coefficient dataframe
    coef_df = data.frame(
      Coefficient = coef_dense[, 1],
      Feature = rownames(coef_dense),
      stringsAsFactors = FALSE
    )

    # Remove intercept and zero coefficients
    coef_df = coef_df[coef_df$Feature != "(Intercept)", ]
    coef_df = coef_df[coef_df$Coefficient != 0, ]

    # Add to results if not empty
    if (nrow(coef_df) > 0) {
      coef_df$Cluster = paste("Cluster", i, sep = "_")
      lasso_coef = rbind(lasso_coef, coef_df)
    }
  }, error = function(e) {
    # Skip this cluster if error occurs
    message("Error in cluster ", i, ": ", e$message)
  })
}
```

I added tryCatch and if-else states, because I had ran into a lot of error and warning during this process.

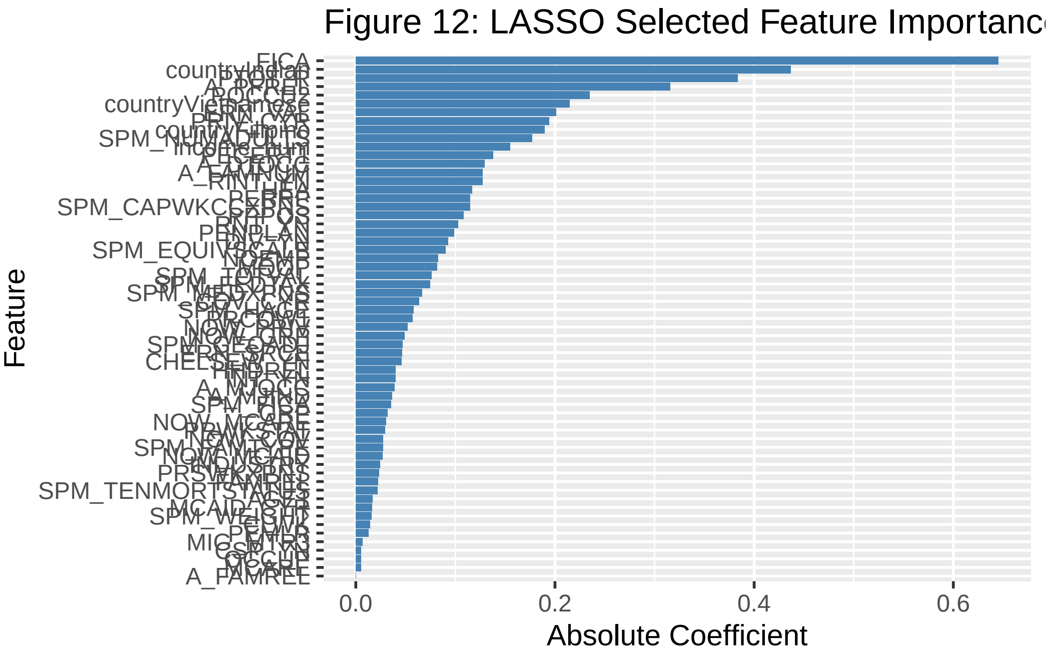
```
lasso_coef = lasso_coef[order(abs(lasso_coef$Coefficient), decreasing = TRUE), ]
print(lasso_coef)
```

```
                   Coefficient           Feature    Cluster
FICA                0.6453519622              FICA Cluster_15
countryIndian       0.4366621233     countryIndian  Cluster_2
PTOT_R              0.3835811530            PTOT_R  Cluster_5
A_PFREL            -0.3156644506           A_PFREL Cluster_12
POCCU2             -0.2350425849            POCCU2  Cluster_4
countryVietnamese  -0.2145910537 countryVietnamese  Cluster_4
ERN_VAL            -0.2011104692           ERN_VAL Cluster_15
PRIV_CYR            0.1939085088          PRIV_CYR Cluster_14
countryFilipino    -0.1894507448   countryFilipino Cluster_11
SPM_NUMADULTS      -0.1770701737     SPM_NUMADULTS Cluster_13
income_num          0.1550922553        income_num  Cluster_5
PECERT1            -0.1377757719           PECERT1  Cluster_1
```

```
A_DTOCC           -0.1294232548           A_DTOCC  Cluster_4
A_FAMNUM          -0.1272110700          A_FAMNUM Cluster_11
RINT_YN           -0.1271353024           RINT_YN  Cluster_4
HEA               -0.1167862831               HEA  Cluster_9
PERRP             -0.1148668853             PERRP  Cluster_1
SPM_CAPWKCCXPNS    0.1148062616   SPM_CAPWKCCXPNS Cluster_13
PPPOS             -0.1083559521             PPPOS  Cluster_1
RNT_YN            -0.1029233229            RNT_YN Cluster_11
PENPLAN           -0.0986014161           PENPLAN  Cluster_7
DIV_YN            -0.0926833293            DIV_YN  Cluster_4
SPM_EQUIVSCALE    -0.0902426691    SPM_EQUIVSCALE Cluster_13
NOEMP              0.0827023110             NOEMP  Cluster_5
MOOP               0.0814519742              MOOP Cluster_15
SPM_TOTVAL         0.0761323363        SPM_TOTVAL Cluster_15
SPM_FEDTAX        -0.0744217651        SPM_FEDTAX Cluster_15
SPM_MEDXPNS       -0.0665151569       SPM_MEDXPNS Cluster_15
COV_CYR            0.0633453685           COV_CYR Cluster_15
SPM_HAGE          -0.0580106547          SPM_HAGE  Cluster_9
PRCOW1            -0.0569060885            PRCOW1  Cluster_7
NOW_PRIV          -0.0521515848          NOW_PRIV  Cluster_4
NOW_GRP           -0.0491445991           NOW_GRP  Cluster_4
SPM_GEOADJ         0.0471736403        SPM_GEOADJ Cluster_12
ERN_SRCE          -0.0462449480          ERN_SRCE Cluster_10
CHELSEW_YN         0.0461988182        CHELSEW_YN Cluster_15
HHDREL            -0.0399129925            HHDREL  Cluster_1
INT_YN            -0.0397595133            INT_YN  Cluster_4
A_MJOCC           -0.0387995294           A_MJOCC  Cluster_4
A_MJIND            0.0363659638           A_MJIND Cluster_10
SPM_FICA           0.0353895040          SPM_FICA Cluster_15
GRP               -0.0319158528               GRP  Cluster_4
NOW_MCARE          0.0305920533         NOW_MCARE Cluster_13
PRWKSTAT          -0.0291884910          PRWKSTAT  Cluster_7
NOW_COV           -0.0273874411           NOW_COV Cluster_16
SPM_FAMTYPE        0.0271686116       SPM_FAMTYPE Cluster_12
NOW_MCAID          0.0268019758         NOW_MCAID Cluster_14
INDUSTRY           0.0241162332          INDUSTRY Cluster_10
PRSWKXPNS         -0.0235351064         PRSWKXPNS  Cluster_5
FAMREL            -0.0221734961            FAMREL Cluster_11
SPM_TENMORTSTATUS -0.0216996943 SPM_TENMORTSTATUS Cluster_19
AGE1               0.0169504967              AGE1  Cluster_9
MCAID_CYR         -0.0164092311         MCAID_CYR  Cluster_4
SPM_WEIGHT         0.0157804290        SPM_WEIGHT  Cluster_3
CLWK               0.0142856851              CLWK Cluster_10
```

```
PEMLR               -0.0129811647          PEMLR  Cluster_6
MIG_MTR3             0.0070121472       MIG_MTR3 Cluster_19
CSP_YN              -0.0052867600         CSP_YN Cluster_15
OCCUP               -0.0052687820          OCCUP  Cluster_4
MCARE                0.0051875905          MCARE Cluster_13
A_FAMREL            -0.0002267708       A_FAMREL  Cluster_1
```

```
ggplot(lasso_coef, aes(x = reorder(Feature, abs(Coefficient)), y = abs(Coefficient))) +
  geom_bar(stat = "identity", fill = "steelblue") +
  coord_flip() +
  labs(title = "Figure 12: LASSO Selected Feature Importance",
       x = "Feature",
       y = "Absolute Coefficient")
```



```
# Extract selected features
lasso_features = unique(lasso_coef$Feature)
```

Positive coefficients mean that higher values of that predictor raise the predicted education level on my 1–6 scale (from less than high school to PhD). Negative coefficients mean that higher values of that predictor lower the predicted education level.

Lasso gives us 58 features, which includes the dummy variables I created for country of origin. But only immigration status is not salient (see Figure 12). Here is a detailed interpretation of the features:

1. income_num A higher scaled income corresponds to higher predicted educational levels, consistent with the idea that greater economic resources facilitate advanced schooling.
2. countryIndian Indian American is strongly associated with higher education, possibly reflecting selective immigration patterns and cultural emphasis on higher degrees.
3. countryFilipino Filipino American is negatively linked to higher educational attainment. May reflect nuances in immigration history or socioeconomic disparities within this subgroup.
4. countryVietnamese Vietnamese American shows a small negative coefficient, possibly reflecting socioeconomic barriers or neighborhood patterns that lower average educational attainment.
5. PERRP The relationship to the household reference person (PERRP) has a negative sign, suggesting that certain roles (e.g., child, extended family) predict lower average schooling. I did not clean this category and transform it into numeric variable because it is not ordinal. Thus, this result might not be convincing.
6. FILESTAT A negative coefficient for filing status (a tax indicator) indicates that individuals with certain filing categories slightly lower in predicted education levels.
7. SPM_FAMTYPE Under the Supplemental Poverty Measure, certain family types (e.g., single-parent households) appear to correlate with higher educational outcomes.
8. NOW_GRP Current group coverage (like employer group insurance) is weakly negative, perhaps reflecting that not all group plans are linked to higher-skill positions. This is contradictory to the private coverage PRV_CYR.
9. SPM_NUMADULT The number of people in the household is negative, suggesting that a larger family might slightly reduce individual educational attainment.
10. GRP Another indicator of group coverage also shows a small negative effect. Implying group coverage alone doesn't guarantee advanced schooling levels.
11. FICA and PTOT These two are also index about income, one is a direct number of income (numerical) and the other is payroll.

```
lasso_features = unique(lasso_coef$Feature)
```

Model Comparison

This study applied three selected machine learning algorithms, Random Forest, and XGBoost, to a preprocessed data set from ASEC 2023.

Data Split

I will use the set of features selected from LASSO to train my models. By removing irrelevant features and simplifying the model, LASSO helps to reduce the risk of overfitting on the training data. In addition, this results in a simpler model that is easier to understand and explain, it is crucial for the interpretability of my project. On top of the LASSO selected features, I also include the features in my hypothesis for modeling, in order to testify my hypothesis. Categorical features, such as citizenship, country, ans sex, are transformed into one-hot encoding in previous steps.

```r
df_asian_clean = df_asian_clean |>
  bind_cols(encoded_cats)
```

```r
hypothesis_features = colnames(encoded_cats)
final_features = unique(c(hypothesis_features, lasso_features))

# Prepare final dataset
df_final = df_asian_clean |>
  dplyr::select(edu, all_of(final_features)) |>
  select_if(~ !any(is.na(.)))
```

Out of the 3217 Asian American records in the data set, 80% are used for training and 20% are used to test the four models' performances. Since the output variable is a factor for classification, I used caret::createDataPartition() to split the data in a stratified manner, ensuring the to ensure the educational attainment distribution is preserved in both training and testing data. As a result, 2,576 records are in the training data set, with 219 less than high school, 640 high school graduate, 372 did not finish college, 576 with bachelor's degrees, 501 with master's degrees, and 268 with PhD. Having the same proportion, among the 641 records in testing data set, 54 less than high school, 160 high school graduate, 92 did not finish college, 144 with bachelor's degrees, 125 with master's degrees, and 66 with PhD.

```r
set.seed(123)
index = createDataPartition(df_final$edu, p = 0.8, list = FALSE)
train_data = df_final[index, ]
test_data = df_final[-index, ]
```

First, the original data was split into two data sets (training and testing) based on the holdout method. Then, I applied 10-fold cross-validation on the training set of 2,576 observations. This approach randomly splits the training data into 10 roughly equal subsets, ensuring each one has a similar proportion of each educational level. In each iteration, one fold is held out as a validation subset while the remaining nine folds are used to train the model. This repeats 10 times so that each fold serves as a validation subset exactly once. By preserving the proportions of educational categories in each fold, it minimize sampling bias and maintain consistent class distributions for training and validation.

```r
folds = vfold_cv(train_data, v = 10, strata = edu)
```

Selected four machine learning models were trained based on nine training subsets, and the remaining validation subset was rotated k times. After evaluating the four models' prediction performances, the best prediction performance model was selected based on comparative analysis. Then, the test dataset was applied to the chosen model and its prediction performance was evaluated and utilized.

Once the train, validation, and test datasets were created from the original data, four machine learning models were compiled using train and validation data with stratified 10-fold cross-validation applied. Then, I evaluated each model's classification prediction performance in terms of accuracy, sensitivity, specificity, precision, score and AUC values.

Workflow

I decided to try two different models for predicting educational attainment, Random Forest and XGBoost, hoping they would do better on the classification side. I set up tidymodels workflows that included a recipe (to handle things like dummy encoding, zero-variance checks, etc.) and a model specification (where we declared the parameters we wanted to tune).

```r
basic_rec = recipe(edu ~ ., data = train_data) |>
  step_dummy(all_nominal_predictors()) |>
  step_zv(all_predictors()) |>
  step_normalize(all_predictors())
```

Random Forest

Random Forest is another popular model for educational outcome classification problem (Masci, Johnes, and Agasisti 2018; Rizvi, Rienties, and Khoja 2019; Akmeşe, Kör, and Erbay 2021). Random Forest can handle large dataset and missing values well. However, the primary disadvantage is that it is prone to overfitting, particularly when working with small datasets. Another potential issue is that it is difficult to understand the logic underlying each prediction.

I started off by defining a random forest model where I let the key parameters, mtry, trees, and min_n, be tuned. This way, I can let the tuning process figure out the best combination for my data. I used the ranger engine because it's fast and gives me a nice way to check variable importance with the impurity measure. Then I set up a workflow that combines this model spec with a recipe that handles things like one-hot encoding for categorical variables, removing predictors with zero variance, and normalizing all the predictors. This setup ensures my data is prepped correctly before it goes into the model.

For the tuning part, I built both a regular and a smaller random grid to explore the hyperparameters without being too computational exhaustive. I set up control parameters to save predictions, work in parallel, and keep everything organized, which makes the whole process smoother. After running cross-validation with these grids, I picked the best model based on accuracy, finalized the workflow, and fit the model on my training data. Finally, I generated predictions on a test set, plotted performance metrics and a confusion matrix to see how well my model was doing, and even extracted feature importance to understand which variables were driving the results.

```r
rf_spec = rand_forest(
  mtry = tune(),
  trees = tune(),
  min_n = tune()
) |>
  set_engine("ranger", importance = "impurity") |>
  set_mode("classification")

# Create workflow
rf_wf = workflow() |>
  add_recipe(basic_rec) |>
  add_model(rf_spec)

## Parameter tuning grid
# Random Forest
rf_params = parameters(rf_spec) |>
  update(
    mtry = mtry(range = c(3, 10)),
    trees = trees(range = c(100, 1000)),
    min_n = min_n(range = c(2, 20))
  )
```

```
Warning: `parameters.model_spec()` was deprecated in tune 0.1.6.9003.
i Please use `hardhat::extract_parameter_set_dials()` instead.
```

```r
rf_grid = grid_regular(rf_params, levels = 5)

classification_metrics = metric_set(accuracy, sens, spec, yardstick::precision)
```

To figure out the best hyperparameters for Decision Tree, I used a grid search combined with 10-fold cross-validation. Essentially, this process tried different values for hyperparameters such as tree depth, then I trained and validated on multiple folds of the data. We collected metrics such as accuracy, sensitivity, specificity, and precision, and I compared them across all those parameter combinations. Once we found the best settings for each model, we finalized them—meaning we took those tuned hyperparameters and fit the models again on the entire training set.

```
> A | warning: While computing multiclass `precision()`, some levels had no predicted events
               (i.e. `true_positive + false_positive = 0`).
               Precision is undefined in this case, and those levels will be removed from the
               averaged result.
               Note that the following number of true events actually occurred for each
               problematic event level:
               'Less than high school': 19

There were issues with some computations   A: x1
There were issues with some computations   A: x1
```

```
rf_metrics
```

```
# A tibble: 4 x 9
   mtry trees min_n .metric   .estimator  mean     n std_err .config
  <int> <int> <int> <chr>     <chr>      <dbl> <int>   <dbl> <chr>
1    10   748     9 accuracy  multiclass 0.540    10 0.00657 Preprocessor1_Mode~
2    10   748     9 precision macro      0.552    10 0.0180  Preprocessor1_Mode~
3    10   748     9 sens      macro      0.383    10 0.0111  Preprocessor1_Mode~
4    10   748     9 spec      macro      0.885    10 0.00142 Preprocessor1_Mode~
```

```
basic_rec = recipe(edu ~ ., data = train_data) |>
  step_dummy(all_nominal_predictors(), one_hot = TRUE) |>
  step_zv(all_predictors()) |>
  step_normalize(all_predictors())

rf_wf = workflow() |>
  add_recipe(basic_rec) |>
  add_model(rf_spec)
```

```r
# Finalize best model
final_rf = finalize_workflow(rf_wf, rf_best) |>
  fit(data = train_data)

# Predictions on test set
rf_pred = predict(final_rf, test_data, type = "class")

# Confusion matrix
all_preds = data.frame(
  actual = test_data$edu,
  rf = rf_pred$.pred_class
)

rf_cm = conf_mat(all_preds, truth = actual, estimate = rf)


# Calculate test set metrics
rf_metrics_test = accuracy(all_preds, truth = actual, estimate = rf)

# Extract feature importance
rf_importance = extract_fit_parsnip(final_rf) |>
  vip(num_features = 15)

print("Figure 12: VIP Plot of Random Forest")
```

```
[1] "Figure 12: VIP Plot of Random Forest"
```

```r
print(rf_importance)
```

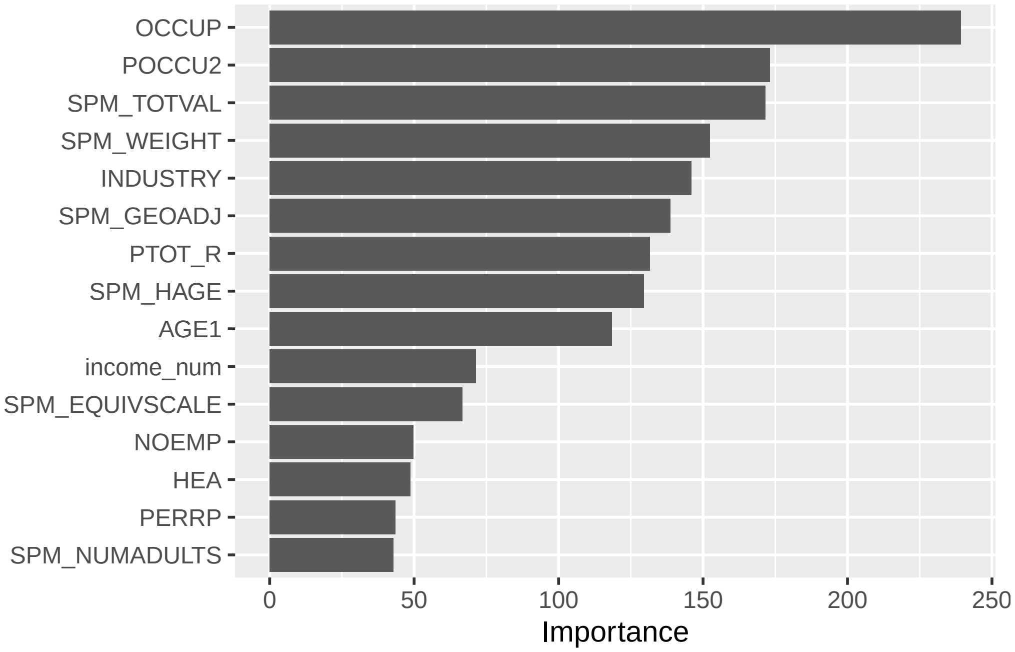
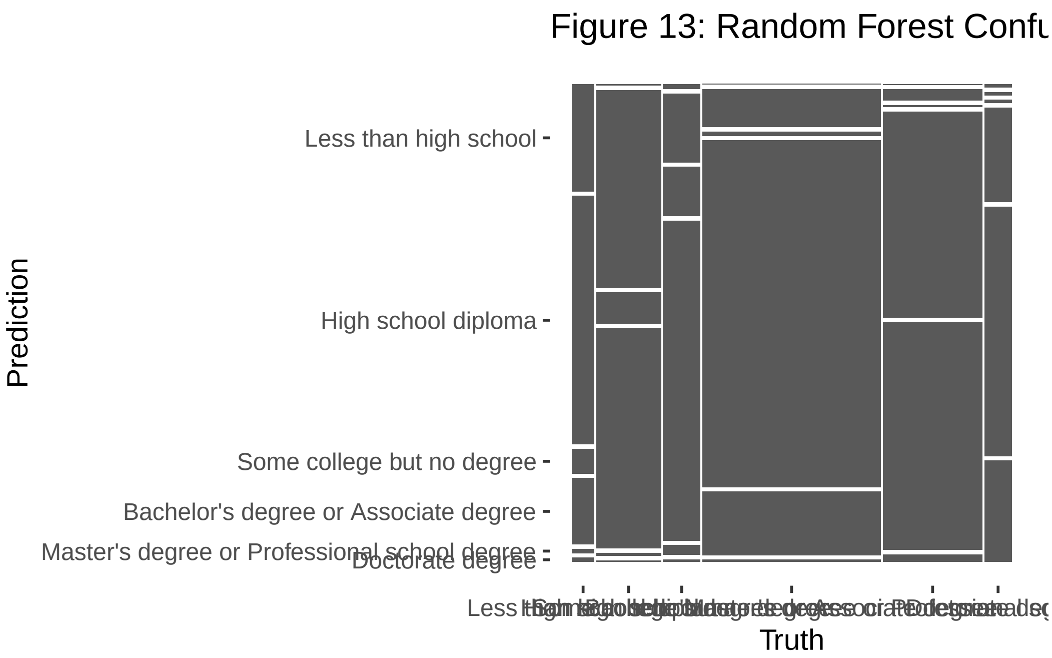


Figure 12 is a VIP plot from the best performing random forest model.

1. POCCU is an indicator of occupation.
2. SPM_TOTVAL is a numeric value of family income. This suggests that family income is a great influencer on educational attainment.
3. SPM_GEOADJ is geographic food, shelter, clothing and utility in the area, which may reflect the resources in the place growing up. High score can signal a more carefree childhood, which might increase educational investment.
4. PTOT is a numeric record of family income, which is an almost identical measure of SPM_TOTVAL and income_num.
5. SPM_NUMADULTS is the number of family member in the household, which may influece the childhood and adolesence development.
6. CountryIndian is one of the hot-coded variable for country of origin. This aligns with my hypothesis that there are divergent outcomes from different Asian origins.

```r
rf_pred = predict(final_rf, test_data, type = "class")

all_preds = data.frame(
  actual = test_data$edu,
  rf = rf_pred$.pred_class
)


rf_cm = conf_mat(all_preds, truth = actual, estimate = rf)
```

```
# print(rf_cm)

autoplot(rf_cm) +
  labs(title = "Figure 13: Random Forest Confusion Matrix")
```



```
rf_metrics_test = accuracy(all_preds, truth = actual, estimate = rf)

rf_f1 = f_meas(all_preds, truth = actual, estimate = rf)

# Print the F1 score
print(rf_f1)
```

```
# A tibble: 1 x 3
  .metric .estimator .estimate
  <chr>   <chr>          <dbl>
1 f_meas  macro          0.416
```

```
accuracy_by_level = all_preds |>
  group_by(actual) |>
  summarize(
    correct = sum(as.character(rf) == as.character(actual)),
    total = n(),
    accuracy = correct / total
  )

print(accuracy_by_level)
```

```
# A tibble: 6 x 4
  actual                                          correct total accuracy
  <ord>                                             <int> <int>    <dbl>
1 Less than high school                                13    54    0.241
2 High school diploma                                  70   160    0.438
3 Some college but no degree                           10    92    0.109
4 Bachelor's degree or Associate degree               338   443    0.763
5 Master's degree or Professional school degree       124   247    0.502
6 Doctorate degree                                     15    66    0.227
```

```
ggplot(accuracy_by_level, aes(x = actual, y = accuracy, fill = actual)) +
  geom_col() +
  labs(title = "Figure 14: Accuracy by Education Level - Random Forest",
       x = "Education Level",
       y = "Accuracy") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

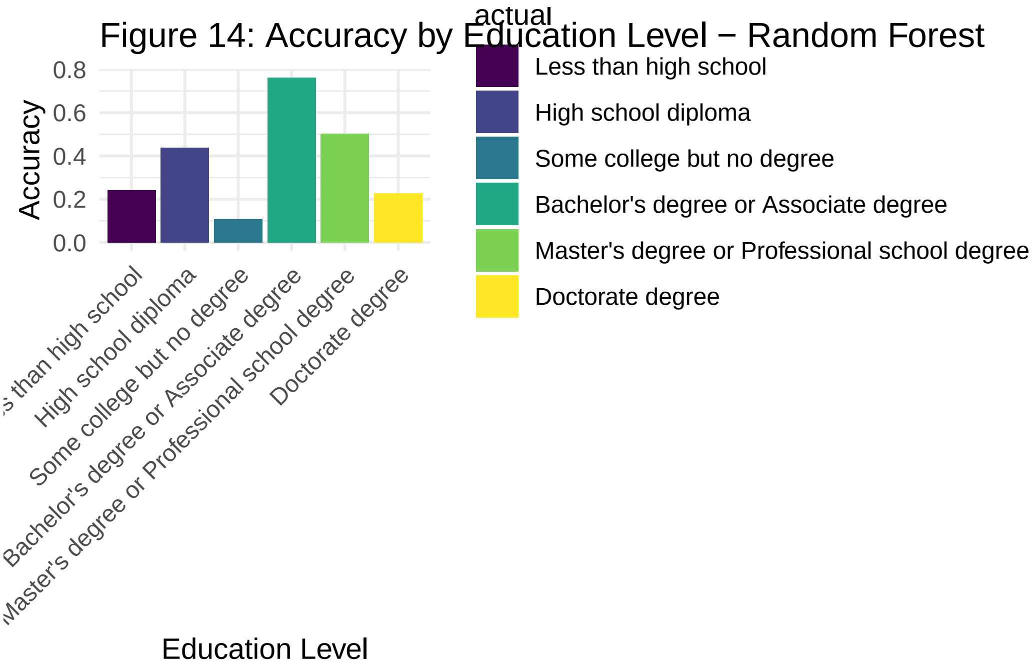


Figure 13 is a confusion matrix. Certain education levels, especially "Bachelor's degree or Associate degree", has wide columns, meaning they occur frequently in the data. The size of the tiles on the diagonal for these columns indicates that the model is getting many of those observations correct. However, some off-diagonal tiles are still visible—especially between closely related categories (for example, "High school diploma" and "Some college but no degree") suggesting misclassification there. Narrower columns (like "Less than high school" or "Doctoral degree") reflect fewer data points in those categories, which can make it harder for the model to correctly predict them.

The model correctly classified 555 out of 1,062 cases, which gives an overall accuracy of approximately 53%. This indicates that slightly more than half of the predictions match the actual outcomes.

Figure 14 is a accuracy by education level plot. The model struggles with classifying less than high school, with only 27% accuracy. The accuracy for high school diploma is also low (40%), indicating potential difficulty distinguishing this group from adjacent categories. Dropping out college is the lowest accuracy among all categories (14%). The model might be confusing this group with other similar education levels or the features may not be strong predictors for this intermediate category. The model performs best for bachelor's degree (76%). This could be due to a larger sample size or more distinct feature patterns associated with this group, leading to more reliable predictions. Moderate performance for Master's degree (48%) suggests the model captures some aspects of this group but still leaves considerable room for improvement. Similar to the lower education categories, the model has difficulty accurately predicting PhD (23%), possibly due to fewer samples or overlapping characteristics with other advanced degree groups.

The model has a f1 score of 0.42, which is on the lower side for most classification tasks. There are several possible reason for the score. Some of the six classes are underrepresented, it becomes harder to predict them correctly, often dragging down the overall F1 score. Education levels that are easily confused, the model might struggle to differentiate them, which lowers F1.

XGBoost

XGBoost is a popular choice for recent compuational research in education subfield, especailly in understanding students academic performance, and has yield high accuracy (Cheng, Liu, and Jia 2024; Amal Asselman and Aammou 2023). XGBoost is a strong choice for this research because it excels at handling complex, high-dimensional datasets and can effectively capture non-linear relationships among predictors. It can improve accuracy and robust performance, even with diverse feature sets such as socioeconomic indicators, demographic attributes, and occupation categories. In addition, XGBoost offers built-in regularization, helping to control overfitting, and is relatively efficient compared to many other ensemble algorithms. However, it does have drawbacks: parameter tuning can be time-consuming, and the resulting models may be less interpretable than simpler methods.

```r
folds = vfold_cv(train_data, v = 10, strata = edu)

basic_rec = recipe(edu ~ ., data = train_data) |>
  step_dummy(all_nominal_predictors()) |>
  step_zv(all_predictors()) |>
  step_normalize(all_predictors())
```

Similar to Decision Tree, I also used tuning grid on hyperpameters including number of trees, minium number of children, depth of tree, learning rate, etc.

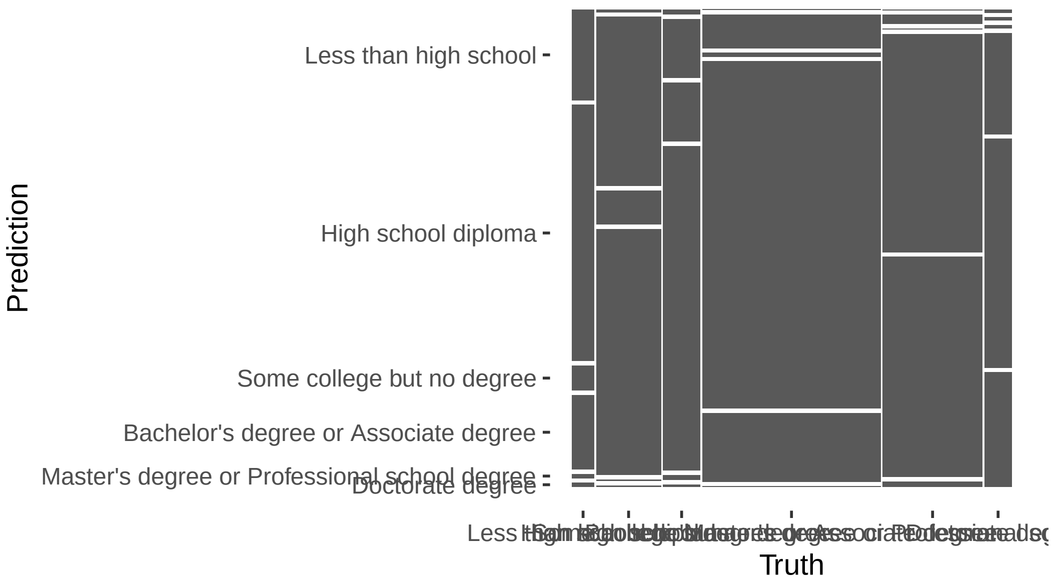
```r
xgb_spec = boost_tree(
  mtry = tune(),
  trees = tune(),
  min_n = tune(),
  tree_depth = tune(),
  learn_rate = tune(),
  loss_reduction = tune()
) |>
  set_engine("xgboost") |>
  set_mode("classification")

# Create workflow
xgb_wf = workflow() |>
  add_recipe(basic_rec) |>
  add_model(xgb_spec)

## Parameter tuning grid
# XGBoost
xgb_params = parameters(xgb_spec) |>
  update(
    trees = trees(range = c(100, 500)),
    tree_depth = tree_depth(range = c(3, 8)),
    mtry = mtry(range = c(3, 8)),
    min_n = min_n(range = c(5, 15)),
    learn_rate = learn_rate(range = c(-3, -1)),
    loss_reduction = loss_reduction(range = c(-1, 1))
  )
xgb_grid = grid_regular(xgb_params, levels = 3)

classification_metrics = metric_set(accuracy, sens, spec, yardstick::precision)
```

```
xgb_metrics
```

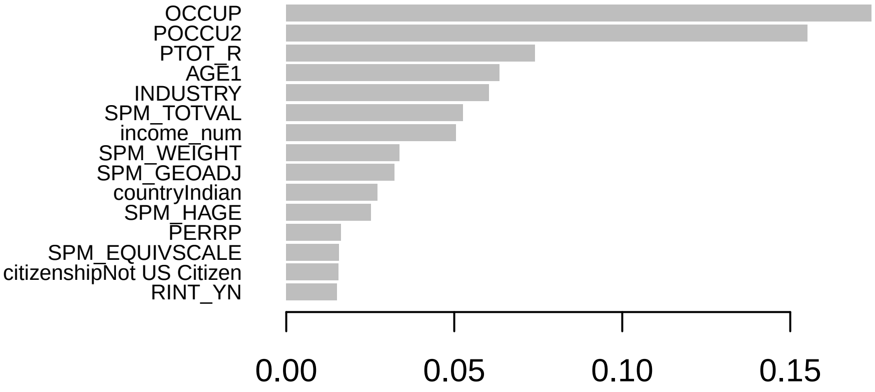
```
# A tibble: 4 x 12
   mtry trees min_n tree_depth learn_rate loss_reduction .metric   .estimator
  <int> <int> <int>      <int>      <dbl>          <dbl> <chr>     <chr>
1     8   317    11          8    0.00880          0.231 accuracy  multiclass
2     8   317    11          8    0.00880          0.231 precision macro
3     8   317    11          8    0.00880          0.231 sens      macro
4     8   317    11          8    0.00880          0.231 spec      macro
# i 4 more variables: mean <dbl>, n <int>, std_err <dbl>, .config <chr>
```

```
# A tibble: 1 x 3
  .metric .estimator .estimate
  <chr>   <chr>          <dbl>
1 f_meas  macro          0.416
```

Figure 15: XGBoost Confusion M



```
[1] "Figure 16: VIP Plot - XGBoost"
```



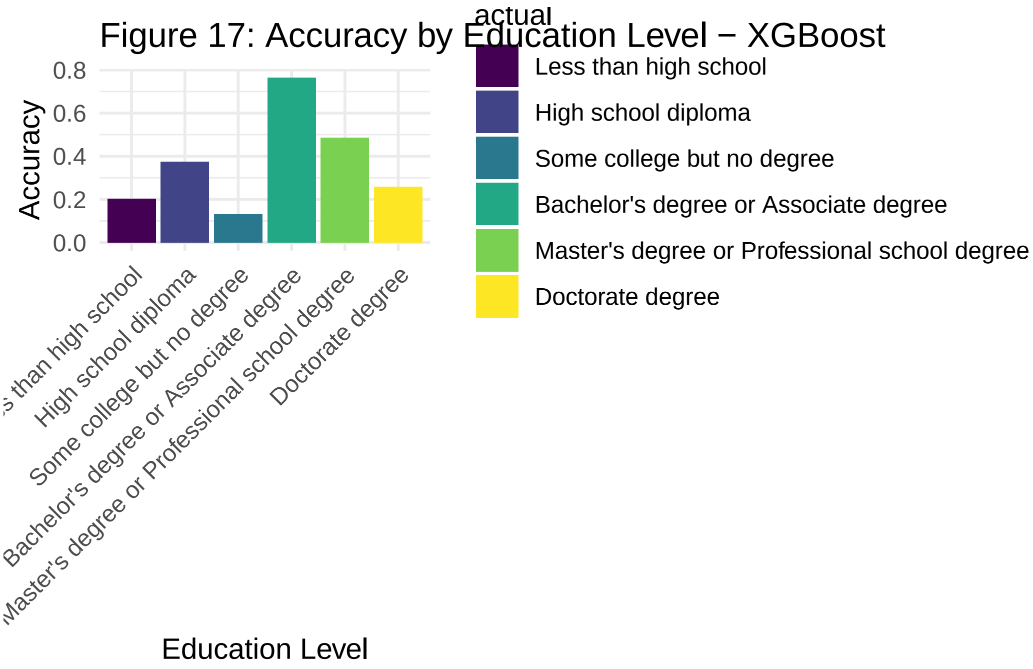


Figure 17: Accuracy by Education Level − XGBoost

Figure 15 is confusion matrix for XGBoost model, we can see that the columns for “High school diploma” and “Bachelor’s degree or Associate degree” remain wide, indicating these classes are common in this dataset. The diagonal tiles for those columns are substantial, suggesting the model correctly identifies many of those observations. However, there are still visible off-diagonal tiles, showing misclassifications, especially around “High school diploma” vs. “Some college but no degree,” and among the more advanced degrees (e.g., “Master’s degree or Professional school degree” vs. “Bachelor’s degree or Associate degree”). Narrower columns for “Less than high school” and “Doctoral degree” highlight that these classes are less frequent, which can make them harder to predict accurately.

Figure 16 is the VIP plot from XGBoost model. POCCU remains the most important feature in both models. Similarly, SPM_TOTVAL and PTOT, two income indicators, are also ranked high in both models. Number of adults and SPM_GEOADJ are also ranked in both models.

Excitingly, Not US Citizen is ranked in the XGBoost VIP, while Random Forest does not have any citizenship variable. XGBoost gives a higher citizenshipNot US Citizen as an important feature, which is hypothesized based on literature.

```
# A tibble: 1 x 3
  .metric  .estimator .estimate
  <chr>    <chr>          <dbl>
1 accuracy multiclass     0.526
```

```
# A tibble: 6 x 4
  actual                                        correct total accuracy
  <ord>                                           <int> <int>    <dbl>
1 Less than high school                              11    54    0.204
2 High school diploma                                60   160    0.375
3 Some college but no degree                         12    92    0.130
4 Bachelor's degree or Associate degree             339   443    0.765
5 Master's degree or Professional school degree     120   247    0.486
6 Doctorate degree                                   17    66    0.258
```

Based on Figure 17, class-by-class results from XGBoost model, “Bachelor’s degree or Associate degree” has the highest accuracy (71%), indicating that XGBoost is most effective at correctly identifying individuals in that category. “Master’s degree or Professional school degree” also shows a moderately strong performance (55%). However, the model struggles with the other categories: 41% accuracy for “High school diploma,” 26%–27% for “Less than high school” and “Doctorate degree,” and is particularly low (about (16%) for “Some college but no degree.” The overall trend aligns with the Random Forest results. Both model has difficulty distinguishing among those categories—possibly due to overlapping features or insufficient representation of those classes in the training data. Overall, while the model performs well for the most populous categories, it has notable confusion in distinguishing the more granular or less frequent education levels.

Comparing Random Forest and XGBoost Performance

I compared two different machine learning algorithms, Random Forest and XGBoost, on a multi-class classification task involving six education-level categories. Both models had hyperparameter tuning and were evaluated using several performance metrics, including accuracy, precision (macro-averaged), sensitivity (macro-averaged), specificity (macro-averaged), and F1 score. Macro-averaged metrics treat each class equally, which is particularly useful in imbalanced or multi-class scenarios where some categories (such as bachelor degree) are more frequent than others.

In terms of overall accuracy, the Random Forest model (0.535) has a slight edge over XGBoost (0.531). The Random Forest also demonstrates a higher macro-averaged precision (0.545) compared to XGBoost (0.553), suggesting that when Random Forest predicts a particular class, it is more likely to be correct for that class. XGBoost exhibits a slightly lower macro-averaged sensitivity (0.37) than Random Forest (0.38), indicating that XGBoost catches more of each class's positive cases on average. Both models achieve similarly high specificity (0.884 for XGBoost vs. 0.882 for Random Forest), meaning they are both quite good at correctly identifying negative cases across all classes.

Random Forest and XGBoost have early identical F1 score 0.42, which indicates a balance between precision and recall in a multi-class setting. An F1 score for both is considered low. This level may reflect the inherent difficulty of distinguishing among six similar or overlapping education categories, as well as any imbalances in class representation.

Discussion

This project is trying to answer my research question: 1. Which model performs best? 2. What are the most influencial factors? By implementing and evaluating Random Forest and XGBoost, my answer to the RQ1 is that they have similar performance. From my analysis, both Random Forest and XGBoost achieve similar performance in predicting educational attainment, but Random Forest holds a slight edge in overall accuracy and macro-averaged precision. However, it's important to note that both models leave considerable room for improvement—neither achieves particularly high scores on metrics like F1, suggesting that neither model is especially robust at capturing all nuances of educational outcomes among the different Asian subgroups. If my primary goal were to maximize sensitivity (i.e., reduce missed cases), XGBoost might be preferable due to its marginally higher macro-averaged sensitivity. But from a practical standpoint, I must conclude that both models, as currently trained, are not especially strong for this multi-class task.

In response to the second research question, both models' variable importance plots highlight POCCU (occupation indicator) as a key predictor, along with SPM_TOTVAL and PTOT (income-related measures) that underscore the central role of economic resources in educational attainment. This result shows that the hypothesis on income and educational attainment is tested true. This finding aligns with literature's emphasize on household income, more specifically, since Asian American parents invest sustantially into children's education, comparing with White counterparts (Lee and Zhou 2015), more economic capital can bring higher chance of investing in good public-school district, more extracurricular activities, including SAT prep and tutoring classes.

SPM_GEOADJ (cost-of-living adjustments) and SPM_NUMADULTS (household size) also rank highly, suggesting that geographic and familial contexts affect educational trajectories. There is few literature on the direct impact of household size and geographic location on education. However, I can offer several possible reasons. The cost of living adjustments reflects the socioeconomic standing of the neighborhood, which ties closely with school access and networking. Living in the neighborhood with high income is more likely to access to good health, food, public schools, and better family background, which contribute to a more "carefree" childhood. In addition, Zhou and Lee's concept of "ethnic capital" can be used to explain our findings (Lee and Zhou 2015). In Asian American neighborhoods, family will exchange educational news and resources, establishing local tutoring centers, and socialize children into the "success frame" which rigidly defines educational success, and create mental burden to children who fail to success academically.

Interestingly, country of origin variables—such as CountryIndian in Random Forest and CountryFilipino in XGBoost—point to diverging outcomes within the broader "Asian" category, reflecting the diversity of experiences among different subgroups. This result testes my initial hypothesis that there are educational disparities among Asian subgroups. Indians are the most highly educated: 79% of Indian immigrants in the United States hold a BA or higher compared with only 8% of India's population, indicating that US Indian immigrants are 10 times more likely to be college educated than their nonmigrant counterparts (Lee and Zhou 2015). Among the five largest US Asian groups—Chinese, Indians, Filipinos, Vietnamese, and Koreans—all but Vietnamese (the only refugee group) are hyperselected. These top five Asian groups account for 83% of the US Asian population. As a result, it makes a lot of sense to have CountryIndian and CountryFilipino as important features. However, this project unable to intepretate the features with more detail because Random Forest and XGBoost are "black boxes."

However, there is no evidence to rejct my null hypothesis on immigration status, which expects non-US citizens and first-gen immigrants will have higher-level of educational attainment.

Taken together, these findings indicate that an individual's socioeconomic status (income, occupation) and demographic background (household size, country of origin, and local resource availability) significantly affect the odds of attaining higher levels of education among Asians in the United States. Higher family income and a more stable environment (as implied by higher SPM_GEOADJ) can create conditions that enable educational success. Conversely, certain subpopulations may face additional challenges or exhibit different patterns of success, as evidenced by the inclusion of country-specific variables. The fact that both models converge on similar sets of top predictors reinforces the idea that income, occupational factors, and demographic circumstances collectively shape educational outcomes in this population.

There are several limitations in this project. First, I might be missing important variables, such as parental education levels, cultural factors, or language barriers, that could better explain variation in educational attainment. A more extensive dataset could help capture the complex pathways leading to higher education. In addition, some features in this dataset is not fully cleaned, for instance, there are several features on income, which is almost identical in the codebook. Second, although Random Forest and XGBoost are powerful, they can be somewhat opaque. More interpretable models or techniques might offer clearer insights into how exactly these factors shape educational outcomes. Given more time, data, or computational power, I would closely explore more all of the 900 features in this dataset, making sure I would not include anything that is not related to the project into models. I would also enhance the interpretability using additional methods like SHAP. Third, with more time and computational power, I would like to try ensemble methods combining multiple algorithms, which has by scholarly to increase performance Amal Asselman and Aammou (2023).

References

Akmeşe, Ömer Faruk, Hakan Kör, and Hasan Erbay. 2021. "Use of Machine Learning Techniques for the Forecast of Student Achievement in Higher Education." *Information Technologies and Learning Tools* 82 (2): 297–311.

Amal Asselman, Mohamed Khaldi, and Souhaib Aammou. 2023. "Enhancing the Prediction of Student Performance Based on the Machine Learning XGBoost Algorithm." *Interactive Learning Environments* 31 (6): 3360–79. https://doi.org/10.1080/10494820.2021.1928235.

Antons, Christopher M., and Elliot N. Maltz. 2006. "Predictive Models of Student College Commitment Decisions Using Machine Learning." *New Directions for Institutional Research* 2006 (131): 69–81.

Basu, Kanadpriya, Treena Basu, Ron Buckmire, and Nishu Lal. 2019. "Predictive Models of Student College Commitment Decisions Using Machine Learning." *Data* 4 (2): 1–18.

Chang, Lin. 2006. "Applying Data Mining to Predict College Admissions Yield: A Case Study." *New Directions for Institutional Research* 2006 (131): 53–69.

Cheng, Biqian, Yuping Liu, and Yunjian Jia. 2024. "Evaluation of Students' Performance During the Academic Period Using the XG-Boost Classifier-Enhanced AEO Hybrid Model." *Expert Systems with Applications* 238: 122136. https://doi.org/https://doi.org/10.1016/j.eswa.2023.122136.

Cirelli, Jared, Andrea M. Konkol, Faisal Aqlan, and Joshua C. Nwokeji. 2018. "Predictive Analytics Models for Student Admission and Enrollment." *Proceedings of the International Conference on Industrial Engineering and Operations Management* 2018 (SEP): 1395–1403.

Covarrubias, Alejandro, and Daniel D. Liou. 2014. "Asian American Education and Income Attainment in the Era of Post-Racial America." *Teachers College Record: The Voice of Scholarship in Education* 116 (6): 1–38.

Dhingra, Pawan. 2020. *Why Good Schools, Good Grades, and Good Behavior Are Not Enough*. New York, USA: New York University Press. https://doi.org/doi:10.18574/nyu/9781479882250.001.0001.

Lee, Jennifer, and Min Zhou. 2015. "Front Matter." In *The Asian American Achievement Paradox*, i–vi. Russell Sage Foundation. http://www.jstor.org/stable/10.7758/9781610448505.1.

Martinez, Azucena L. Jimenez, Kanika Sood, and Rakeshkumar Mahto. 2024. "Early Detection of at-Risk Students Using Machine Learning." https://arxiv.org/abs/2412.09483.

Masci, Chiara, Geraint Johnes, and Tommaso Agasisti. 2018. "Student and School Performance Across Countries: A Machine Learning Approach." *European Journal of Operational Research* 269 (3): 1072–85.

Rizvi, Saman, Bart Rienties, and Shakeel Ahmend Khoja. 2019. “The Role of Demographics in Online Learning; a Decision Tree Based Approach.” *Computers and Education* 137: 34–47.

Zhou, Min, and Carl L. Bankston. 1998. *Growing up American: How Vietnamese Children Adapt to Life in the United States*. Russell Sage Foundation. http://www.jstor.org/stable/10.7758/9781610445689.

Zhou, Min, and Susan Kim. 2006. “Community Forces, Social Capital, and Educational Achievement: The Case of Supplementary Education in the Chinese and Korean Immigrant Communities.” *Http://Lst-Iiep.iiep-Unesco.org/Cgi-Bin/Wwwi32.exe/[in=epidoc1.in]/?t2000=023327/(100* 76 (April). https://doi.org/10.17763/haer.76.1.u08t548554882477.