# CS 504

**Final Report**

# Spring 2021

**Team Hydrogen**

# George Mason University

# **Prediction on Attributes Affecting Salary in the U.S.**

**Team Members:**

Glenn Turner, Kevin Uruena

Ryan Vavra, Mike Walsh

Lucien Wobembong

# **Table of Contents**

Contents

[**Table of Contents** 2](#_Toc71042211)

[**Abstract** 4](#_Toc71042212)

[1. Introduction 5](#_Toc71042213)

[1.1. Background and Rationale 5](#_Toc71042214)

[1.2. Research 9](#_Toc71042215)

[1.3. Project Objectives 10](#_Toc71042216)

[1.4. Problem Space 10](#_Toc71042217)

[1.5. Primary User Stories: 11](#_Toc71042218)

[1.6. Solution Space 11](#_Toc71042219)

[1.7. Product Vision 11](#_Toc71042220)

[Scenario #1 12](#_Toc71042221)

[Scenario #2 12](#_Toc71042222)

[1.8. Definition of Terms: 12](#_Toc71042223)

[2. Data Acquisition 14](#_Toc71042224)

[2.1. Overview: 14](#_Toc71042225)

[2.2. Field Descriptions: 14](#_Toc71042226)

[2.3. Data Context: 16](#_Toc71042227)

[2.4. Data Conditioning 16](#_Toc71042228)

[2.5. Data Quality Assessment: 17](#_Toc71042229)

[2.6. Other Data Sources 18](#_Toc71042230)

[3. Analytics and Algorithms 19](#_Toc71042231)

[3.1 – Correlation Matrix: Correlation Between Attributes 20](#_Toc71042232)

[3.2 – Spearman Correlation: Correlation Between Salary and Education 22](#_Toc71042233)

[Table 3.2 – Risks Associated with the Algorithms 23](#_Toc71042234)

[3.3 Kendall and Spearman Correlation 23](#_Toc71042235)

[3.4 Linear Regression 24](#_Toc71042236)

[3.5 Lasso Regression 24](#_Toc71042237)

[3.6 Random Forest 25](#_Toc71042238)

[3.7 Gradient Boosting 26](#_Toc71042239)

[3.8 Extreme Gradient Boosting 27](#_Toc71042240)

[3.9 Algorithm Comparison and Selection 27](#_Toc71042241)

[3.10 AWS SageMaker and XGBoost Tuning 28](#_Toc71042242)

[4. Visualization 30](#_Toc71042243)

[4.1. Risk Associated with the Visualizations 30](#_Toc71042244)

[4.2. Distribution of Attributes in Dataset with Outliers 30](#_Toc71042245)

[4.3 - Salary Distribution with Outliers 34](#_Toc71042246)

[4.4 - Salary Distribution without Outliers 37](#_Toc71042247)

[4.5 – XGBoost and Sage Maker 39](#_Toc71042248)

[4.6 – Tableau: Salary by Gender 42](#_Toc71042249)

[4.6.1- Distribution of salary by Gender and Industry of Worker 42](#_Toc71042250)

[4.6.2- Distribution of salary by Gender and Educational Attainment 43](#_Toc71042251)

[5. Findings 44](#_Toc71042252)

[5.1. End User Changed 44](#_Toc71042253)

[5.2. Improvements through Additional Attributes 44](#_Toc71042254)

[5.3. Data Conditioning and Outliers Removal 44](#_Toc71042255)

[5.4. Finding the Right Algorithms - XGBoost 45](#_Toc71042256)

[5.5. Best and Important Attributes 45](#_Toc71042257)

[5.6. Attributes and Insights from Data 45](#_Toc71042258)

[6. Summary 47](#_Toc71042259)

[7. Future Work 48](#_Toc71042260)

[8. Appendix: Codes and Interactive Visualization 49](#_Toc71042261)

[9. References 50](#_Toc71042262)

# **Abstract**

Data shows that in the past 40 years, worker salaries has not grown along productivity. These workers do not have a tool they can use to describe a fair salary, nor do they have a method for determining important factors for raising their salary. This study will use data collected by the U.S. Census Bureau to create a model capable of predicting salary and prioritizing certain factors that enable a higher salary. This study will evaluate several analysis methodologies, like linear regression, correlation and Extreme Gradient Boosting (XGBoost) to define an accurate model.

The results of the study show that XGBoost provides the most accurate model (81% accuracy) and that all of the data from the Census should be evaluated without any omissions (e.g., age restrictions). The analysis also shows that Experience and Industry are two of the biggest factors that lead to a higher salary. Other large factors that contribute to higher salary are education and acquired certifications.

These results show that the most effective way for an individual to achieve a higher salary is to change industries or seek higher education. Future iterations of this study should expand and seek additional factors that contribute to a salary of $100,000 or more.

# Introduction

## Background and Rationale

The American dream is based on the belief that through dedication, hard-work and perseverance, anyone from any socio-economic background can prosper in America.

Every day, millions of Americans embark on the journey to archive their version of the American dream through employment. According to Statista, since 1990, the U.S. has maintained an employment level of at least 119 million (Statista, 2021).

To get an accurate count of how many people lived or worked each year in the U.S., the Census Bureau asks that people complete a voluntary survey. The Bureau has been playing its role since its formation in 1902.

The Census Bureau collects data on people’s age, work status, education, marital status, occupation, relationship, race, sex, capital gain/loss, hours worked per week, native country, and hundreds of other attributes. Data collected by the Census Bureau not only allows the federal government to know how to allocate billions in federal funds and how many seats each state gets in the House of Representatives, it also helps citizen, students, financial advisors, businesses, researchers, policy makers, etc. to make important financial decisions.

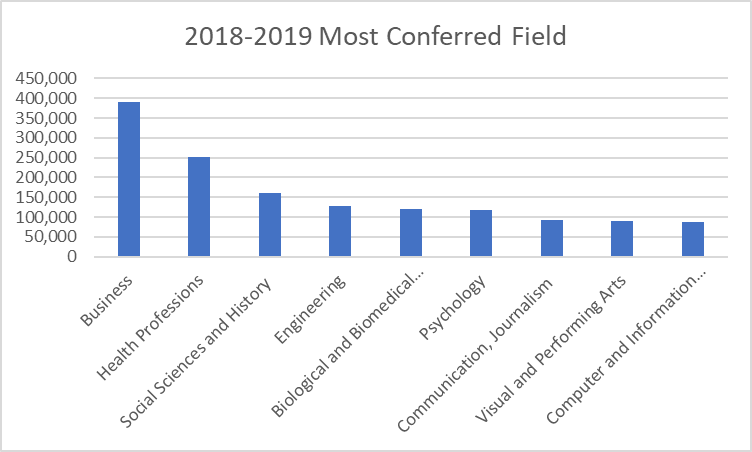
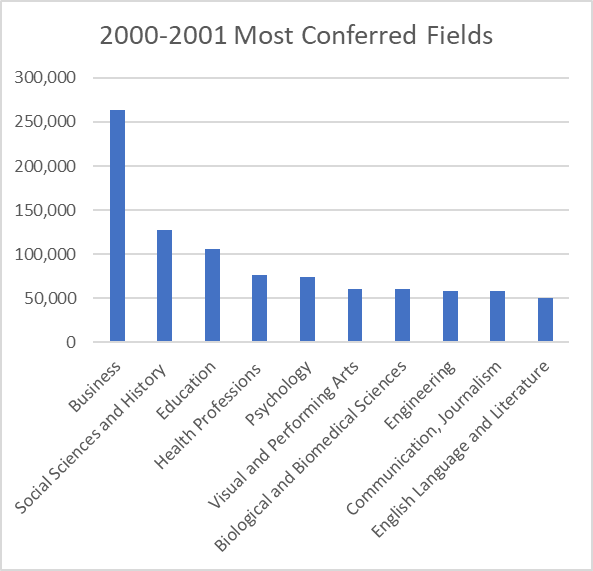
For the year 2020-2021, an estimated 3.7 million students will graduate high school and an estimated 19.7 million (NCES, n.d.) students will be attending college during the same period. As previously stated, the Census Bureau collects data on thousands of attributes, including college degree, major, and salary. These attributes are useful for these college students wanting to ensure their chosen major is not only one they will enjoy, but one which will allow them a comfortable lifestyle. Based on Figure 1, the top 5 most conferred field for 2018-2019 were business, health profession, social sciences and history and engineering. Compared to Figure 2 were the top conferred fields for 2000-2001 were business, social sciences and history, education, health profession and psychology.

Figure 1: 2018-2019 Conferred Field of Studies

Figure 2: 2000-2001 Conferred Field of Studies

Census data utilization is not limited to just high school students choosing their first college major, or businesses choosing their target market. It’s also used by millions of Americans switching careers. When choosing which career to switch to, salary is a crucial factor. Based on data from CNBC (Liu, 2020), which is displayed in figure 3, the 25 highest paying professions in 2020 were as followed:

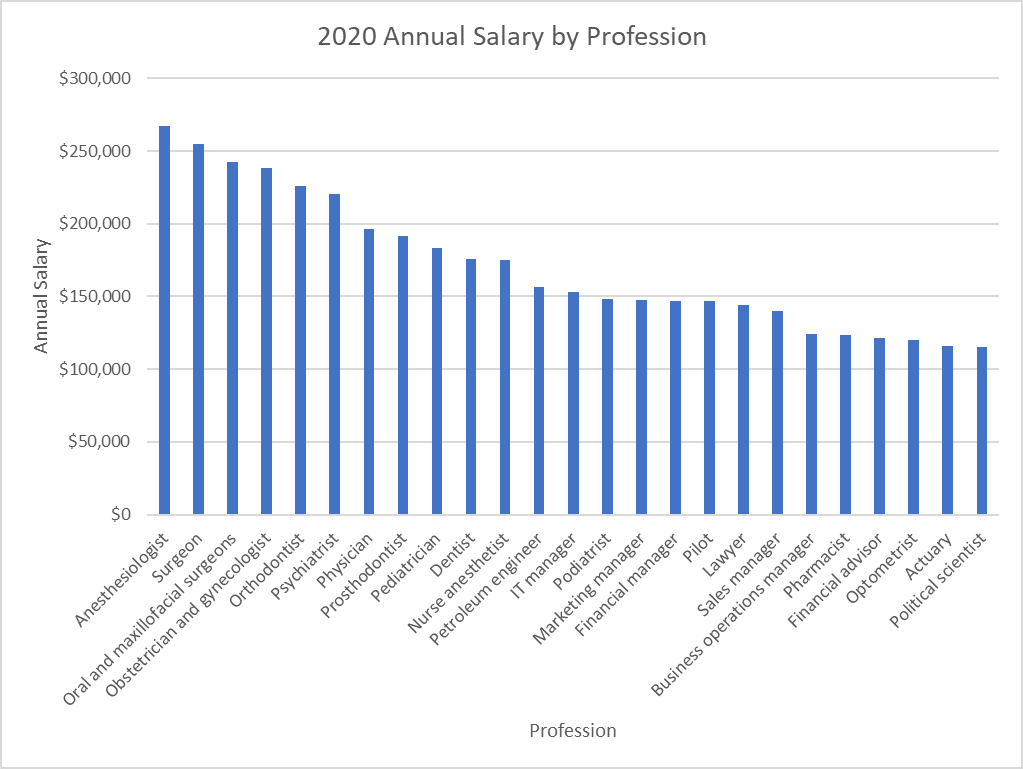


Figure 3: Highest paying specialization and their Annual Salary

Looking at figure 3, healthcare had the highest paying professions, followed by engineering and business. Looking at the data from CNBC, there was a correlation between high earning fields and high levels of education. The top 10 highest paying job in the healthcare needed a doctorate, while a doctorate is not a requirement for business or engineering jobs.

Although salaries and job demand (over 2.6 million jobs were added in 2018) are high today, that has not always been the case. Salary is not always guaranteed to increase over time as job demand fluctuates. This stagnation in salary regardless of job demand overtime is because “*[w]ages are insensitive to current economic conditions [since] they are effectively installment payments on the employer's obligation to transfer a certain amount of wealth to the worker over the duration of the employment arrangement*” (Brookings Institute, 1980).

The Economic Policy Institute (EPI) found that from 1948 to 2018, there has been a disparity between productivity growth and compensation. Looking at figure 4, from 1981 to 2018, there was a 142.4% increase in productivity, while hourly compensation only increased by 28.2% increase. This divergence has received criticism from both lawmakers and economists alike over the years.

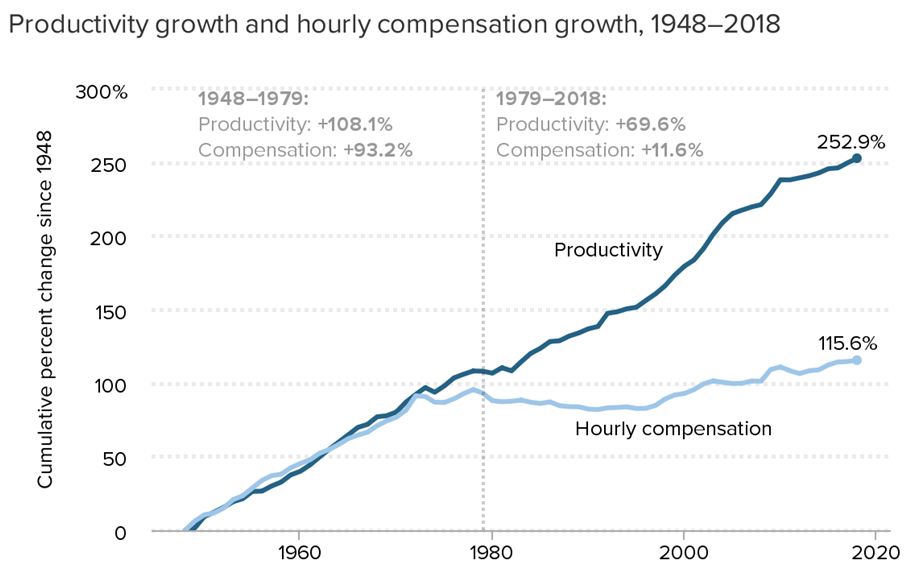


Figure 4: Productivity and Hourly Compensation from 1947 to 2018

Over the past 40 years, there has not only been a disparity between productivity and hourly production, but there has also been a strong growth for the highest wage workers, leading to income inequality. From 2018 – 2019, the 95th percentile of workers had a 4.5% increase, while the 10th percentile saw a decline of 0.7% (Gould, 2020).

Figure 5, shows the growth difference between the 95th, 50th and 10th percentile. Based on the data collected by EPI, there was a sharp increase for the 95th percentile (from 24.9% in 2000 to 63.2% in 2018). 50th percentile saw an 8.6% increase, while the 10th percentile had a slow increase from -6.7% in 2000 to 3.3% in 2019. This clear wage inequality has led to hundreds of protests and demand for change.

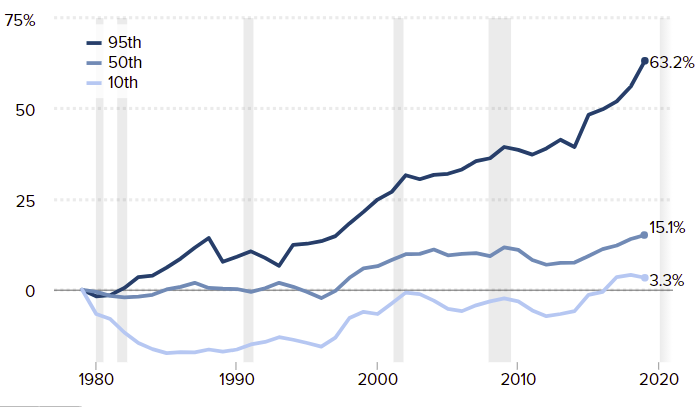


Figure 5: Change in Wages by Wage Percentile

The fight over wage inequality has not been limited between high and low earners. Gender has also been playing a role in increasing the wage gap. While both men and women have experienced wage stagnation, women have experienced it more.

Looking at figure 6, From 2011 to 2019, 95th percentile men had a 29.5% increase in wages, while women saw a 15.3% increase (Gould, 2020). As previous stated, there is a noticeable gap between the various percentile in the men’s wage, but that is not apparent in the women. Based on figure 6, between the various percentile, the wage gap is more compressed for women, leading us to conclude that there is not as much of a wage gap within the women’s category.

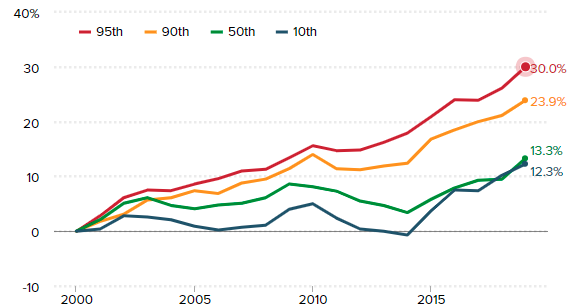
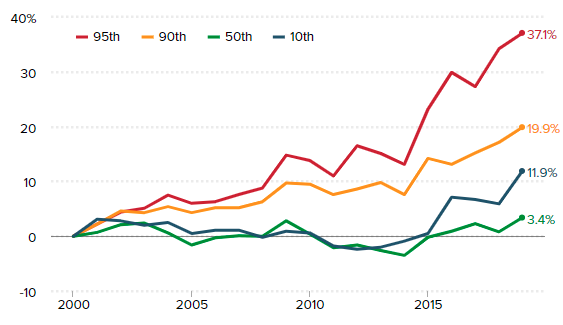


Figure 7: Women Wage Gap

Figure 6: Men Wage Gap



Workers’ educational level also added to the wage stagnation and wage inequality. EPI found that from 2000 to 2019, those with advanced degrees saw the highest increase in wages, compared to those with only a high school degree.

Figure 8 shows that there was a 10.8% difference between those with “advanced degrees” and those with only “some college” in 2019. From 2010 to 2019, those with “advanced degree” saw a 7.9% increase in wages (8.5% in 2010 and 16.4% in 2019), “college” graduate had a 7.4% increase (3.1% in 2010 and 10.5% in 2019), “some college” had a 7.4% increase (-2.0% in 2010 and 0.8% in 2019), “high school” saw 1.2% increase (-0.9% in 2010 and 2.1% in 2019) and those with “less than high school degree” saw a 7.6% increase (1.2% in 2010 and 8.8% in 2019).

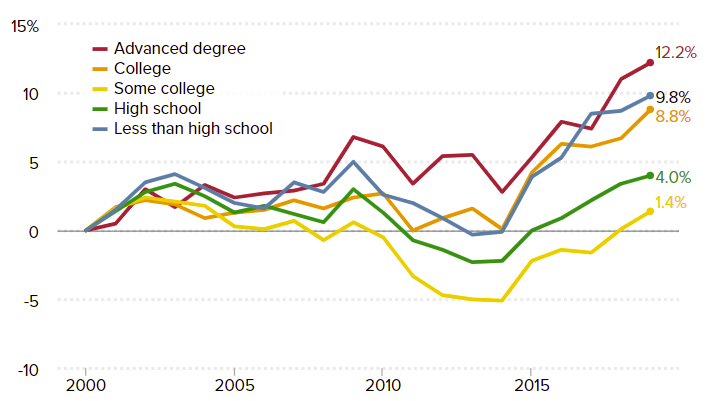


Figure 8: Hourly Wages by Education

After reviewing this correlation between job, salary, educational level and gender, we became curious about what other attributes correlate with salary. Team Hydrogen will be exploring other attributes such as educational level, native country, hours worked per week, occupation, etc. and their effect on salary. We were inspired to select these attributes by the “Adult Data Set”, which had similar options.

## Research

At the outset of this project, we initially came across the Adult Data Set hosted in the UC Irvine repository (Kohavi, 2019). This initial dataset was extracted from the 1994 Census database with the following set of parameters: (AAGE>16) && (AGI>100) && (AFNLWGT>1) && (HRSWK>0).

While further reviewing this dataset to verify its applicability in our project, we found that there were fifty-two other papers that had cited this dataset. We reviewed the context in which these papers referenced the Adult Data Set to ascertain whether we would be replicating, or closely mirroring the work of others.

Upon further exploration, these papers were only using the dataset for testing and validating their models for classification algorithms and not a regression problem. Given the usefulness of the dataset in testing and validating these models, we determined that the dataset would be a viable option for our needs.

Due to the age of the Adult Data Set, we concluded it would be best to find more recent data that would suit our needs so that any recommendations made by the model would be more current. This led us to the possibility of extracting our own data from the Census database. Not only would this provide us with more recent data, but it would allow us to make our own selections with regards to what attributes we found to be most salient, as well as provide us with a salary figure that the Adult Data Set was lacking.

With the goal of more recent data in mind, we found two possibilities from the Census website that could accomplish our goals. The first option was the “Annual and Social Economic Supplements” dataset. This dataset would not only provide us with existing data, but also time-series data as the data ranged from 1998 – 2020. The risk that comes along with using this dataset is that there are hundreds of attributes, which would require some detailed data cleaning to make it functional for our analysis.

The second option involves scraping data from the Census database using an API. We found that the census does allow for API scraping, but the user must apply and receive a key. With this information, we know we can pull as many tailored datasets as our project evolves and requires, a capability not available with the Adult Data Set. Like the risk associated with the “Annual and Social Economic Supplements” dataset, this option could require additional time cleaning the data that we wouldn’t necessarily have to spend with the Adult Data Set.

The team decided to use 2019 and 2020 data from the U.S. Census Bureau as it contained more than enough factors and entries to analyze for predicting salary.

## Project Objectives

The objectives of the project are multifaceted, but center around the key goal of being able to project the salary of an individual based on a variety of attributes that are specific to the individual. Utilizing the US Census Data, we are seeking to answer a key sub-problem to our goal of predicting salary.

This sub-problem is simply determining which attributes are most important in predicting salary. Unfortunately, this is not a simple process as we must first limit our dataset of 843 attributes to a smaller set that we feel is most relevant.

After limiting the attributes of the dataset, we can then determine whether we can predict the salary with a variety of initial analyses. While the goal of having a core set of factors that determine salary, we do not want to inject our biases into the model and are willing to let the data take us in ways we may not have predicted.

After determining these factors, we can then answer a variety of sub-problems we are seeking to answer. These sub-problems are ever-changing as the data evolves, but are currently:

* Are negative factors more important than positives?
* How do the hours work per week factor into salary?
* Does occupation and educational level determine level of income?
* Can an employer use the presented model to adjust employees’ salaries based on their earning potential to reduce employee turnover and retain talent?

The overarching end goal is to have a model that can predict salary and can be used by a variety of users to determine the expected salary based upon the current, or future, attributes of any given individual.

## Problem Space

The team will use the Census Bureau dataset to explore factors, relationships and their impact on salary. This project will show how impactful these individual and groups of factors are and recommend which of them an individual seeking a higher salary should prioritize to meet that goal. This tool will also be used by business owners to offer a competitive salary based on factors such as experience and education level. The team will accomplish these goals using analytics tools, and methodologies and show our results in the most intuitive format.

We will also use this data to evaluate other topics as they arise throughout our analysis, given time constraints.

## Primary User Stories:

The stories that will be used to articulate how our software will provide value to the user are:

* As an employer, I would like to know the factors impacting salary potential so that I know the correct amount to offer an employee to retain them.
* As an employer, I would like to have a tool that allows me to offer a competitive salary to a potential employee so that I have a better probability of hiring them.
* As an individual seeking employment, I would like to know how much an employer should offer such that I am compensated fairly.
* As an individual seeking a higher salary, I need to know which controllable factors will lead to greater earning potential so that I can focus on those factors to enable greater earning.
* As an individual seeking a higher salary, I need to prioritize factors that will lead to greater earning potential so that I can effectively use my time to pursue greater earning potential.

These stories will guide the team to evaluate factors impacting earning potential and aid us in drawing conclusions.

## Solution Space

Our prototype delivers value to its users when it accurately predicts and informs users of which factors impact salary most. Users derive value from these predictions and factors when they gain understanding about their likely financial prospects. This is not only important for future planning related to financial decisions, but also a possible guide for changes they can make in their lives to achieve these goals.

From an individual perspective, a user can use our model to make career path decisions, whether that is at the start of their journey, or later when deciding if their current position has a competitive compensation. From an enterprise perspective, using this model to calculate the potential earning power of your workforce can help with budget and forecast costs. This is vital to ensure sustainable growth for the company.

In addition, this prototype can generate potential insights into possible predictions for employee turnover. A high employee turnover can result in higher expenses, delays in projects, and even more planning since employee replacements are time consuming and costly. According to Mercer.com, he leading cause of voluntary workforce turnover in 2019 in the United States were better job opportunity (81% response), and better base salary at 39% (Mercer, 2020). Since these reasons indicate salary compensation is an important reason for an employee to stay in a company, our model can help an employer to identify potential improvements to compensation packages. An improved package makes the company more competitive in the market and allows it to retain top talent.

## Product Vision

* This product could have value for a wide array of customers in the public and private sectors. It could be used by individuals who are seeking to better understand their finances. There are a number of similar products available from financial institutions like Fidelity Investments, but our product will provide deep insights into their financial position and will be strongest when given large sets of data, at little to no cost to them. This tool could also be produced for organizations to help them make financial or wage decisions by leveraging their own data in combination with public data.

### Scenario #1

* An 18-year-old student has just started her first job, and she’s starting to forecast her future financial situation. She’s a little dismayed to see that her salary is quite low, so she uses our tool to see how her current information (attributes) might affect her future earnings. She uses the tool to see if there are factors that she could leverage to boost her future earnings.

### Scenario #2

* An employee has been at the same job for a number of years and would like to earn a higher salary. This employee would use our tool to get an idea for things they should target to earn that higher salary.

## Definition of Terms:

|  |  |
| --- | --- |
| **WORD** | **DFINITION** |
| Algorithm | “The list of instructions and rules that a computer needs to do to complete a task.” (ThinkAutomation, n.d.) |
| Application programming interface (API) | Application programming interface (API) allows applications to speak with each other |
| Attributes | Properties of an entity |
| AWS SageMaker | Amazon tool that allows for the creating, training and deployment of data in the cloud (Amazon, n.d) |
| Collinear | Variables which can be accurately predicted by each other using linear regression techniques |
| Conferred | Awarded degree by an institution |
| Correlation | Relationship/connections between attributes |
| Dependent Variable | This variable is the output of the process |
| EC2 Instance | Amazon that provides scalable computing capabilities |
| Economic Policy Institute | Non-profit think tank |
| Exploratory Data Analysis | Is a technique “used by data scientists to analyze and investigate data sets and summarize their main characteristics, often employing data visualization methods” (IBM n.d.) |
| Factors | Columns in the dataset (ex, occupation, education) |
| F-Score | Measurement of a model’s accuracy on a dataset (Wood, 2019) |
| Gradient Boosting (GB) | Machine learning boosting method that minimize prediction errors by combining previous models with the best possible model |
| Hyperparameter | Setting values that can control how an algorithm operates |
| Independent Variable | These are the variables, or features that are used as the input |
| Interquartile Method | This is a method for finding outliers involving finding the interquartile range (IQR) and then setting an upper and lower bound (e.g., multiplying by a constant) to find outliers. (Taylor, 2018) The standard formula for this method is sets the lower bound at Q1 - (1.5\*IQR), and the upper bound at Q3 + (1.5\*IQR) |
| Jupyter Notebook | Allows for the writing, editing and executing of codes in any programing language |
| Model | File trained to recognize certain patterns (Microsoft, n.d.) |
| Outlier | “observation that lies an abnormal distance from other values in a random sample from a population” (NIST.Gov, n.d) |
| Overfitting | Modeling error that occurs when a function is too closely aligned to a limited ste of data points (Twin, 2020) |
| Pearson Correlation | “Measures the degree of the relationship between linearly related variables” (Statistics solutions n.d.) |
| Percentile | The value below which a percentage of data falls |
| Predictors | Independent variables used to predict the dependent (target) variable |
| Regression | Statistical model that shows relationships between variables |
| RMSE | Measures error in prediction by model (Moody, 2019) |
| S3 Bucket | Amazon public cloud storage (Amazon, n.d.) |
| Salary | Payment to employee for providing a service |
| Shrinkage | The reduction of regression coefficients to simplify models and remove predictor variables |
| Sklearn Package | Open-source machine learning tool used for predictive data analysis. It’s built on NumPy, SciPy, and matplotlib (Scikit-learn, n.d.) |
| Spearman Correlation | “Is a non-parametric test that is used to measure the degree of association between two variables” (Statistics solutions, n.d.) |
| Survey | Method of gathering information through questioning |
| Tableau | Data analysis and visualization tool owned by Salesforce |
| The Census Bureau | Government entity that specializes in keeping track of population in the U.S. through surveys |
| Trends | Direction in which an event or change is moving |
| Tuning | The process of adjusting hyperparameters of algorithms with the intention of improving performance |
| XGBOOST | Gradient Boosting designed for speed and performance |

# Data Acquisition

## Overview:

The inspiration for investigating factors that contribute to an individual’s salary was the Adult Data Set found on the UCI Irvine website. This dataset had enough records to explore different algorithms, and the attributes were clearly labeled. In addition, the data was extracted from the US Census Bureau, and used in several published papers.

However, it was outdated data as it was published in 1994. It only provided categorical data for whether a person made more or less than fifty thousand a year, instead of actual range for salary. With this in mind, we decided to retrieve recent data from the Census. Through research, we found the Annual Social and Economic Supplement (ASEC) dataset, which had record types of Person, and Household.

ASEC 2020 for record type *Person*, originally had 157,960 entries, with about 840 attributes related to a person’s income. As for the ASEC 2020 for record type *Household*, the original dataset had 91,501 entries, with about 134 attributes.

We decided to combine both the *Person* and *Household* dataset into one to better correlate person’s salary attributes with the location of that person’s household. Both datasets were combined using the household sequence number attribute (PH\_SEQ in Person, H\_SEQ in Household). The result set has the same number of entries as *Person*, 157,960 entries, with 843 attributes. The same procedure was conducted with the ASEC 2019 dataset, which was eventually combined with ASEC 2020 in an effort to acquire more data entries through a longer period of time. This resulted in a final data set of 338,060 entries. Since the original *Adult Data Set* had attributes that were interesting to us, we decided to find the corollary to them in this new data set, as well as other attributes that were related to a person’s job. This resulted in 40 attributes we would be focusing on

## Field Descriptions:

The initial data set, as pulled from the Annual Social and Economic Supplement repository, had 843 attributes available. After extensive review, we whittled down the number of attributes to those listed below based upon our view of their relevance to our goal of salary prediction and modelling the data to predict salary.

|  |  |  |
| --- | --- | --- |
|  | **FIELD** | **DESCRIPTION** |
| 1 | A\_AGE | Age |
| 2 | A\_CLSWKR | Class of Worker (Private, Federal, etc.) |
| 3 | A\_DTIND | Industry (2 Digit Recode - Detailed Groups) |
| 4 | A\_DTOCC | Occupation (2 Digit Recode - Detailed Groups) |
| 5 | A\_FTPT | School Status (Enrolled as FT/PT Student) |
| 6 | A\_HGA | Educational Attainment |
| 7 | A\_HRS1 | Hours worked last week at all jobs |
| 8 | A\_MARITL | Marital Status |
| 9 | A\_MJIND | Industry (2 Digit Recode - Major Groups) |
| 10 | A\_MJOCC | Occupation (2 Digit Recode - Major Groups) |
| 11 | A\_SEX | Sex |
| 12 | A\_UNCOV | Is job covered by a union or employee association contract? |
| 13 | A\_UNMEM | Is person a member of a labor union or association similar? |
| 14 | A\_USLHRS | Hours per week at job |
| 15 | A\_WKSTAT | Work Status (Full-time/Part-time, etc.) |
| 16 | CAP\_VAL | Value of capital gains received |
| 17 | CAP\_YN | Yes/No - Did receive capital gains? |
| 18 | CLWK | Longest job class of worker (last year) |
| 19 | DIV\_VAL | Value of dividends received |
| 20 | DIV\_YN | Yes/No - Did receive dividends? |
| 21 | GESTFIPS | State FIPS (Federal Information Processing Standards) Code |
| 22 | INDUSTRY | Industry of longest job last year (4 Digit Code - Most Specific) |
| 23 | LJCW | Longest job class of worker (last year; more detailed) |
| 24 | OCCUP | Occupation of Longest Job Last Year (4 Digit Code - Most Specific) |
| 25 | PEARNVAL | Total Person’s Earnings (Dependent Variable) |
| 26 | PECERT1 | Currently have active Professional/State/Industry License? |
| 27 | PECERT2 | Is Certificate/License issued by the Government? |
| 28 | PECERT3 | Is Certificate/License required for your job? |
| 29 | PEHRUSLT | Hours usually worked last week |
| 30 | PEIOIND | Industry (4 Digit Code - Most Specific) |
| 31 | PEIOOCC | Occupation (4 Digit Code - Most Specific) |
| 32 | PENATVTY | Country of Origin |
| 33 | POCCU2 | Occupation of Longest Job Last Year (2 Digit Recode - Detailed Groups) |
| 34 | PRCITSHP | Citizenship Status |
| 35 | PRDTRACE | Race |
| 36 | WECLW | Longest job class of worker (last year; most detailed) |
| 37 | WEIND | Industry of longest job last year (2 Digit Recode - Detailed Groups) |
| 38 | WEMIND | Industry of longest job last year (2 Digit Recode - Major Groups) |
| 39 | WEMOCG | Occupation of Longest Job Last Year (2 Digit Recode - Major Groups) |
| 40 | WEWKRS | Weeks worked last year |
| 41 | WEXP | Weeks worked (bucketed categories) last year |

## Data Context:

The project relies on a dataset from the US Census Bureau. The US Census Bureau updates its Current Population Survey (CPS) every month with data collected from a representative sample of U.S. households, families, and persons [sic.]. This monthly data is then compiled and integrated with additional data in the Annual Social and Economic Supplement (ASEC).

It is among the most thorough and recent datasets of the US population which are publicly available. As our project focusses primarily on individual attributes and salary, most of our data comes from the “person” dataset within the ASEC. Our final dataset combined distributions from March of 2019 and 2020. Of note with respect to the timing of this data, there was a large economic downturn starting in February of 2020.This was caused by the onset of the global coronavirus pandemic. This almost certainly had some effect on our dataset, but its precise effect is difficult to quantify. For example, one of the primary effects of the pandemic was a lower response rate among contacted households in the March 2020 ASEC.[[1]](#footnote-2)

## Data Conditioning

We are confident that this dataset is accurate and has valid data due to the data being collected and published by the United States Census Bureau. The published data set has 800 plus fields and the first activity was to analyze those fields and select those germane to our analysis. Please see section 2.2 for a list of the selected fields.

Once the fields were selected, the team ran initial data cleansing activities on the resultant dataset. Using R, we ran the NA.OMIT function on the dataset to remove any records with NA values in them. There was no change in the dataset (I.e., no NA values) validating our assumption of clean data published by the US Census. The team continued cleansing activities, searching for blank values using R:

formattedSalaryData[formattedSalaryData ==""] <- NA #this replaces blanks with NA

cleanFSD<-na.omit(formattedSalaryData) #this removes all NA rows with no change in the dataset.

After completing these cleansing activities, the team used R to further analyze the data. Using graphing functions like scatterplots and the pair function, the team agreed on values to omit from the analysis. We saw that Armed Services members were not accurately represented and further analysis of the documentation showed that Armed Services members are only included if they live with a civilian (U.S. Census Bureau, 2020). This led the team to omit Armed Services members, as the salaries included in the dataset were not representative of that class of worker. Using these same mechanisms, the team concluded that workers working below 20 hours a week should be omitted and children (under 18) should also be omitted. Another omission was workers reporting negative earnings.

The team analyzed this smaller dataset using several algorithms such as correlations, XG Boost and linear regression. Initial results using these algorithms were not promising, leading the team to use all the rows in the dataset and repeating these analyses again to achieve better results. Age considerations, hours worked, and wage limits were removed and the resulting analysis was stronger and much more promising towards developing an accurate model for predicting salary. The team also added more factors to the dataset to improve the analysis results.

Using all the attributes for analysis presented some issues. One specific issue was the presence of “–9999” in the salary column. This skewed some visualizations initially but was easily corrected with the “scipen=50” option in R. This option removed scientific notation and used integers for salaries in the Y axis, which corrected the minor visualization issue.

The team also added in 2019 data into the analysis. The addition of the factors and the 2019 data improved the analysis results considerably and led to a much stronger model for predicting salary.

## Data Quality Assessment:

Before any dataset can be used, it should be checked for completeness, uniqueness, accuracy, conformity and overall quality. The reason why these four steps are essential is because they expose inconsistencies and technical issues with the dataset. It also ensures the data fits the need of the organization or issue(s) you’re looking to solve.

* **Completeness**: Our first dataset was originally collected in 1994 and was cleaned and donated to University of California Irvine in 1996. Since we wanted our recommendation to be based on modern data, we pulled new dataset from the Census Bureau ASEC report for the years 2019 and 2020. The new dataset originally had too many attributes (840), attributes needed for our research were in different places and there were null values. We were able to combine the *Person* and *Household* dataset, trim the number of attributes down and after removing the null values, the dataset decreased to 338,060 records. After discussing with the team, we agreed that our data cleaning was complete and met our standards.
* **Uniqueness**: The Census Bureau collects data on hundreds of unique attributes ranging from salary levels to number of family members. The dataset we pulled from their website was unique and the attributes being used in our research were also unique.
* **Accuracy**: Being a government collected, and cleaned dataset, we are certain the data we pulled are accurate. Data released to the public is not just checked for accuracy by a handful of people. Accuracy is check by hundreds of enumerators, informants, and supervisors. The Census Bureau has a “Census Quality Indicators Task Force,” and their job is to produce scientific indicators to check for quality and accuracy. Over the past 230 years, the Census has deployed numerous operations to ensure their data is complete, accurate and free of errors. Based on the level of rigor the Census puts in data collection and cleaning, we are certain our dataset meets the highest level of accuracy.
* **Conformity**: We decided to use the data as provided and not make any modifications. The fields used for analysis did not require reformatting or modification.
* **Overall Quality**: Team Hydrogen can say with high confidence that the dataset pulled from the Census Bureau has a high overall quality. Data collected by the Bureau is highly regarded in all industries.

## Other Data Sources

While searching for data sources that would work for our goals, we came across *UC Irvine’s Adult Data Set*. This data source was the impetus to our question regarding salary prediction and became a reference for us as we dug deeper into what we would need to predict salary.

While this data set would possibly work for us, we ultimately decided against using it for a couple of reasons. Our first concern was that the salary attribute was not a specific salary amount but instead it was a binary category. Either the individual made over $50,000 or they made $50,000 or under. While this could be helpful if our goal was to simply predict that an individual would be in one of those two categories, it did not quite meet the needs of our project’s goal and we wanted to pursue other data sources that might meet these needs.

After finding more recent data that met the needs of the salary attribute, we ultimately decided against the *Adult Data Set* due to the age of the data set. The data was extracted from the 1994 Census database, meaning that this data was over two decades old. If we wanted to provide modern answers, then we needed to use current data.

The other option we contemplated was utilizing the Census website’s API capabilities to pull our own data. This would allow us to select which attributes we wanted, get more recent data, and have a better understanding of the dataset since we pulled it together.

We decided against this option for one key reason: potential difficulties with learning to scrape the data. Doing it incorrectly could lead to improper data or more work with cleaning up the data than we necessarily wanted to take on. The risks of that simply outweighed the pros given the benefits of the data source we ultimately landed on.

We initially started some exploratory data analysis on the 2020 Census ASEC dataset with only 20 predictor variables. The conditioning included only people ages 18 to 65, hours worked per week higher than 18 hours, and only positive values for our dependent variable salary.

After that initial exploratory data analysis, we ran into issues trying to find correlations between salary and the other attributes and we our initial practice models yielded weak results. As a result of these less than promising finds we decided to revisit the ASEC data and the attributes selected. In the end, we end up with 40 attributes we considered interesting from the same ASEC data set and we brought in 2019 data. With this larger, more diverse dataset, no conditioning was set as to maintain as much data variation as possible. The additional attributes proved to be very useful in boosting both correlation strength and modeling accuracy.

# Analytics and Algorithms

Data exploration is an essential and necessary step for any data analysis, implementation, and publication. Before deciding which algorithms would be best for our dataset, each analyst was encouraged to process the dataset using the algorithms and methods of their choice. Hydrogen ended up exploring the dataset using the following algorithms and modeling: Pearson Correlation, Linear Regression, Lasso Regression, Random Forest, Gradient Bosting (GB), and Extreme Gradient Boosting (XGBoost).

Before we go into the specifics of each algorithm, it is important to notice the distribution of our target variable salary (PEARNVAL) for our dataset with 338,060 entries shown in figure 9. We can clearly see that salary does not have a normal distribution, and that on the contrary it is quite right skewed. Around 64% of the data (217,148 entries) reported an income of 0 to 25,000. After around 150,000, we can see how the count is minimal. Based on this observation, the team decided to explore and identify the presence of outliers in the data.



Figure 9: Salary Distribution for the 2019 and 2020 ASEC dataset.

In order to identify outliers in our target category, the interquartile (IQR) method was used to identify the upper and lower bounds of salary. In addition, the boxplot method in the seaborn package in Python was used graphically to show the outliers. Figure 10 shows the box plot of salary.

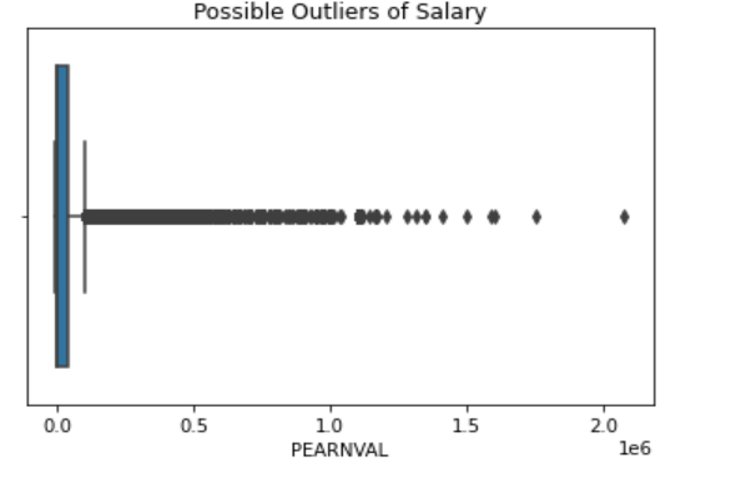


Figure 10: Boxplot for target variable salary (PEARNVAL) to identify outliers in dataset.

As shown in figure 10, we can clearly observe the presence of outliers. Starting at around 100,000, there are several consecutive data points outside the upper bound limit. Two data points around 1.6M and 2.1 M, are both out of the limit. Using the results from IQR (Q1 = 0, Q3 = 40,000, IQR = 40,000), we can obtain a range between –60,000 and 100,000 for salary. With this range in mind, the team decided to explore 2 different versions of the dataset, one with outliers and another without the outliers. The version without outliers had a total of 318,872 entries, which represents around 5.6% data loss in comparison with the dataset that includes outliers. For each algorithm explored, modeling was performed for both datasets, with and without outliers.

## 3.1 – Correlation Matrix: Correlation Between Attributes

Pearson correlation is one algorithm the team used to analyze how related two factors are to each other. This is an assessment of how closely two variables grow together. A positive correlation of 1 shows they grow together perfectly, a negative correlation of 1 shows they grow the exact opposite of each other. The team is looking for correlations such that -.5>Correlation > .5 to show a good relationship between 2 variables.

As part of exploratory data analysis, the team used these correlation matrices to ascertain relationships between the independent variables or features of the dataset and the dependent variable of salary (PEARNVAL). Figure 11 shows this correlation matrix using the total dataset, including outliers.

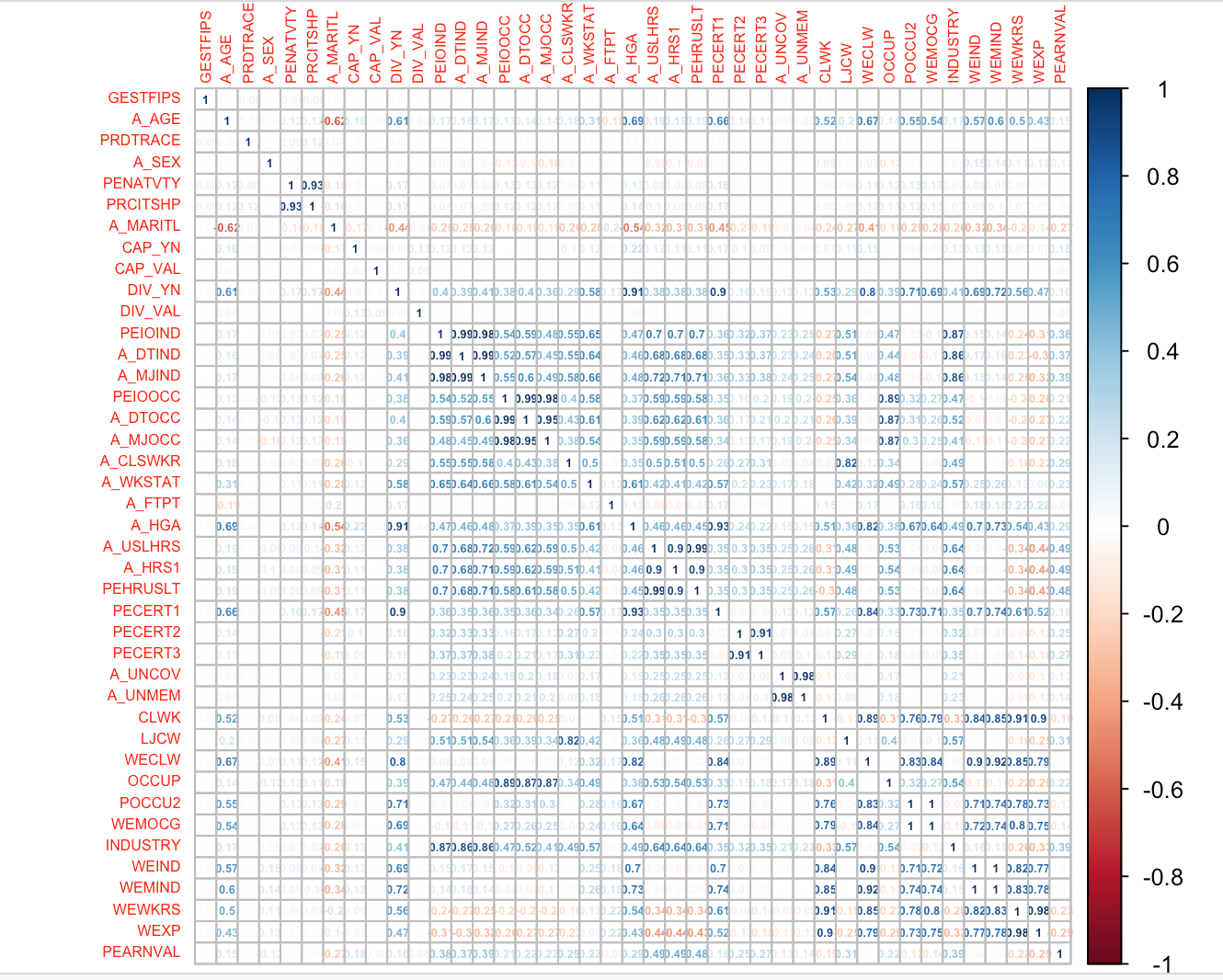


Figure 11: Correlation Matrix with Outliers

Looking at figure 11, the most correlated attributes to a salary are: A\_HRS1 (how many hours worked last week at all jobs), A\_USLHRS (how many hours a week does one normally work at this job) and PEHRUSLT (hours worked last week). It is important to note that these fields all calculated as .49, .49 and .48, respectively. These numbers do not meet the .5 standard for correlation that the team is looking for.

Due to the lack of strong correlation using the dataset with outliers, we decided to check correlation using the dataset without outliers. The dataset generated the correlation matrix seen in figure 12. The correlations got stronger with the outliers removed. For the previous 3 factors, the correlation scores increased to .74, .73 and .73 respectively. The next strongest correlation is INDUSTRY, calculating to .62. That factor is related to the longest job worked in the previous year. This is an indicator that experience may factor into earning a higher salary. During our initial research into salaries and jobs in the U.S., we found that the highest paying jobs were in the healthcare industry and salary breakdown by profession is depicted in figure 3. Industry having a strong correlation is not surprising because those in the healthcare industry earned the highest because they must complete years of internships, residencies and other experience to find work.

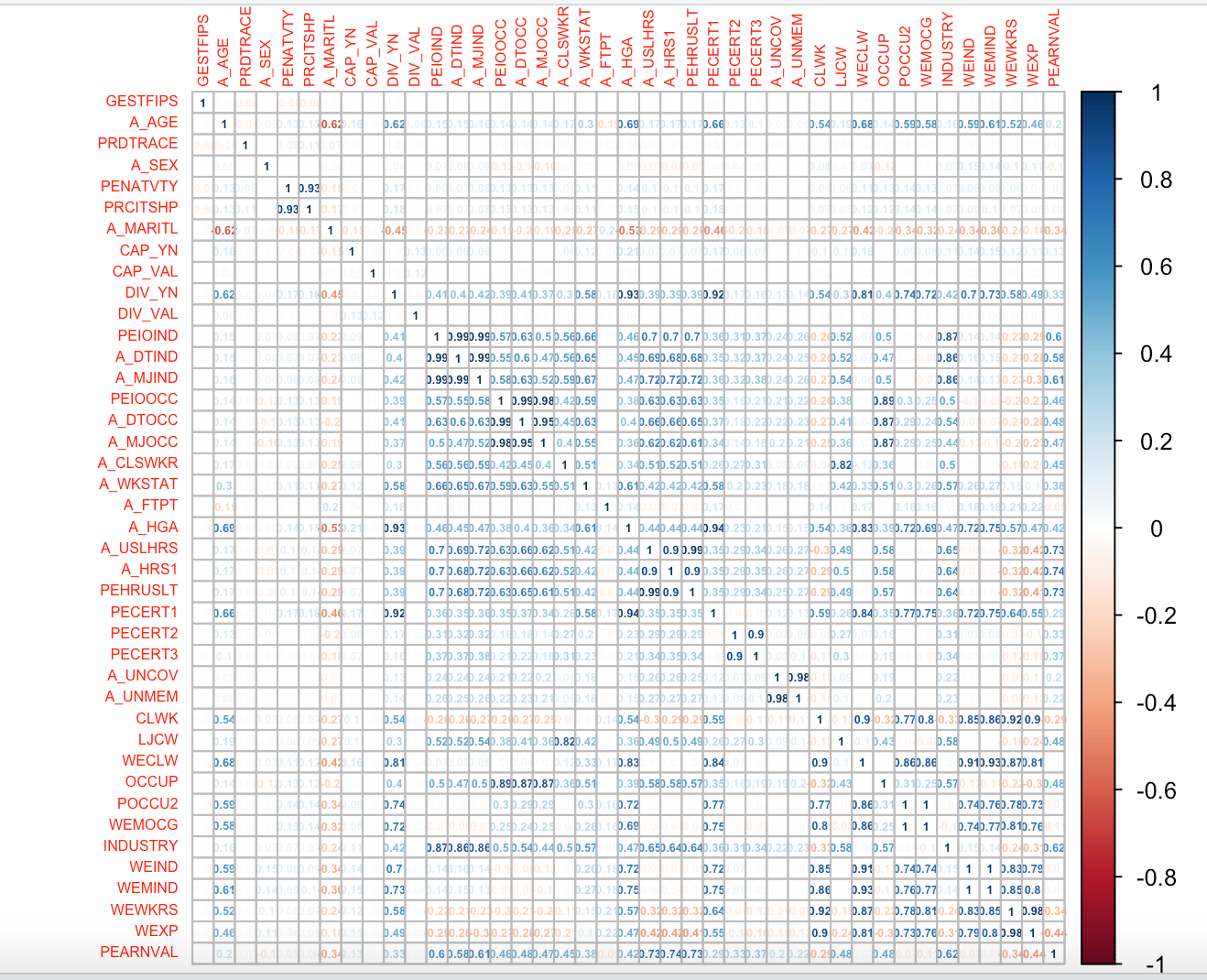


Figure 12: Correlation Matrix without Outliers

## 3.2 – Spearman Correlation: Correlation Between Salary and Education

It is interesting to note that using Pearson correlation does not show education (A\_HGA) to play a significant role in earning a higher salary (measures at .42). Afterall, every student is constantly reminded that education is the best way to high earning jobs. This led the team to research other correlation calculations outside of Pearson. Pearson correlation works well for variables that are normally distributed, like weight and height. Looking more into the attribute A\_HGA (education), it is measured from 31 (less than first grade) to 46 (PhD or equivalent). This scales differently from a salary scale and may benefit from a different correlation calculation. Spearman correlation shows education to correlate with Salary at .53 as seen in figure 13. This is a stronger correlation than Pearson but not as strong as one would think.

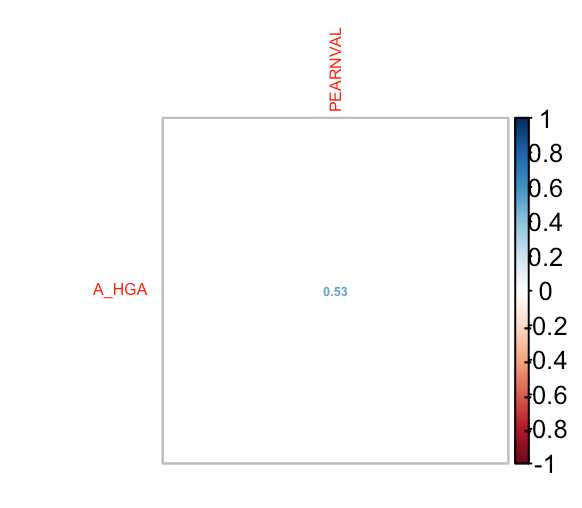


Figure 13: Spearman Correlation

## Table 3.2 – Risks Associated with the Algorithms

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | **Description** | **Probability of Occurrence** | **Impact** | **Mitigation** | **Status** |
| 1 | If we incorrectly calculate or interpret Pearson’s correlation, then we will have inaccurate conclusions. | 1 | 3 | Review each other’s code. | Current |
| 2 | If we do not find suitable variables to use to predict salary with linear regression, then we will need to adjust our analysis goals. | 1 | 2 | Data set has been initially analyzed and there appear to be good variables. | Current |
| 3 | If the accuracy of the generated model is low, then we will not have the ability to extrapolate information from the data. | 2 | 3 | We plan on running the model multiple times | Current |
| 4 | If the accuracy of the generated model is low, then there is a higher risk of overfitting the test data. | 1 | 3 | We have adjusted the model parameters and it is over 80% accurate | Closed |

## 3.3 Kendall and Spearman Correlation

The first algorithm tool we wanted to explore would allow us to see the correlation between the attributes. According to the Statistical Tools for High-throughput Data Analysis, a correlation test “measures linear dependence between 2 variables.” In laymen’s term, this is an assessment of how closely 2 variables grow together. A positive correlation of 1 shows they grow together perfectly, a negative correlation of 1 shows they grow the exact opposite of each other. We are looking for correlations such that -.5>Correlation > .5.

We will apply scikit-learn (a free python module for machine learning), Kendall and Spearman Correlations across the dataset to evaluate correlations. The reason for this exploration is to find factors with high correlation to salary.

Our initial visualization between salary and the other attributes are depicted in figure 14. The result not only revealed what we already knew and suspected, but it also revealed new insights we never have seen without the visualization. Based on the data on figure 14, the top attributes with high correlation to PEARNVAL (salary) were: A\_HRS1 (Hours worked last week at all jobs), A\_USLHRS (Hours per week at job), PEHRUSLT (Hours usually worked last week), INDUSTRY (Industry of longest job last year (4 Digit Code - Most Specific)) and A\_MJIND (Industry (2 Digit Recode - Major Groups)).

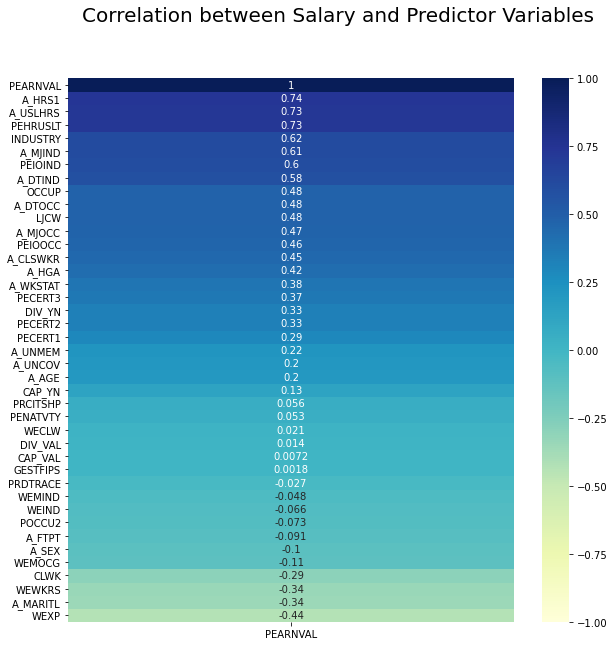


Figure 14: Correlation between Salary and features

## 3.4 Linear Regression

After conducting some exploratory data analysis with several types of correlation, we sought to explore our data further information using Linear Regression. Linear Regression is a simple model designed to tell us the relationship between variables by fitting a line to the data (Yale University, n.d.).

In this case we attempted to determine the relationship of our variables to the PEARNVAL (Total Earnings) of the individuals in our dataset. The LinearRegression class from the sklearn package in Python was used to evaluate a multiple linear regression models. When running linear regression with our dataset, with outliers, we received and R-squared value of 0.37 and an RMSE of 48,000. We found that the fields we focused on were statistically significant. However, linear regression simply did not fit the data we had and there was no way forward with linear regression despite tinkering with the various attributes and how they interacted with each other. As for the dataset without outliers, our R-squared and RMSE values improved significantly to values of 0.74 and 13,400, respectively. This makes sense as extreme values such as outliers could have a significant contribution to the sum of squared residuals, and therefore after removing them, the model obtained better performance metrics.

## 3.5 Lasso Regression

Lasso regression (Least Absolute Shrinkage and Selection Operator) is a regression tool designed to expand upon multiple linear regression. Within R, this tool is accessible with the package “glmnet.” In addition to performing linear regression, the primary feature of Lasso regression is “shrinkage.” Shrinkage is the process of reducing regression coefficients until they reach zero, at which point the variables with a coefficient of zero are excluded from the model. This can be extremely useful in analyzing large, complex datasets with possible collinearity. The main benefit of the Lasso regression is to reduce the number of predictor variables and simplify the resulting model to allow for faster and less complex modeling. (Stephanie, 2021).

Our team used the Lasso model as an intermediary step between linear regression and decision tree analytics. The Lasso model had potential to improve upon multiple linear regression and offer valuable insights into which variables from our dataset were worth preserving. After running the Lasso model again multiple linear regression, we were able to improve our RMSE score from about 48,000 to 47,000. We were also able to increase the R-squared value from 0.37 to 0.38. When running Lasso with the dataset without outliers, the RMSE and R-squared values improved to about 19,000 and 0.7, respectively.

In cross-validation, the Lasso regression also gave our team important insights into our selected variables. Figure 15 shows that all 40 predictors were selected by the regression to achieve the lowest RMSE value. As a result of this process, we learned that all our predictors were useful and should continue to be used in the analysis. We also learned that there was no significant improvement to be made with a model based on linear regression, so our team began to explore more advanced models.

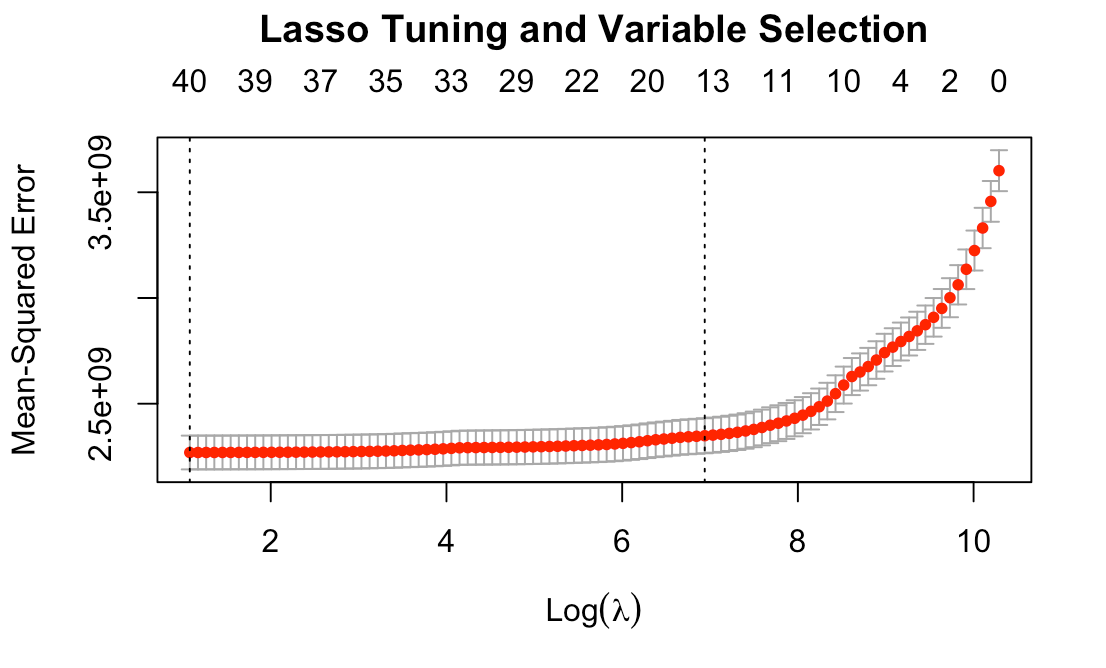


Figure 15: Lasso Model

## 3.6 Random Forest

Random Forest is a supervised learning algorithm based on decision trees. The first process of Random Forest is the random selection of rows of data from the dataset. The selected rows are referred to as “in-bag.” The algorithm then builds a decision tree using the in-bag data. This tree is evaluated by passing the data which was not randomly selected to make the tree (out-of-bag data). The process of random sampling can be repeated many times, and functions as a built-in cross-validation system. As the algorithm builds more trees into the “forest,” modeling accuracy improves until it reaches a point of diminishing returns where further trees do not significantly contribute to the forest. One of the primary advantages to regression in this way is that overfitting is an extremely low risk. The out-of-bag validation process makes overfitting unlikely. (Random forest)

Our team used Random Forest as a first step into decision tree modeling. The RandomForestRegressor class from the sklearn.ensemble package in Python was used to model our dataset using this algorithm. Immediately, Random Forest produced much better results than linear regression techniques. Although Random Forest was not our final regression technique, its initial success was proof that decision-tree techniques were a good fit for the dataset. With default parameters set for regression, Random Forest was able to create a moderately accurate model using the dataset with outliers.

One of the reasons that Random Forest was not used as the final modeling algorithm was the lack of fine-tuning parameters. Compared to later algorithms that our team used; Random Forest’s default regression represented an extremely high portion of the possible modeling strength with this technique. The default Random Forest model scored an RMSE of 45,000 and an R-squared value of 0.45, significantly better than any previous technique. As for the RMSE and R-squared values using the dataset without outliers, 11,700 and 0.8 were obtained, respectively.

In addition to creating a stronger model, Random Forest was also able to give us valuable insight into the strength of predictors with the “Variable Importance Plot,” which lists the most significant predictor variables in descending order. Figure 16 shows the Variable Importance Plot with the top ten variables displayed.

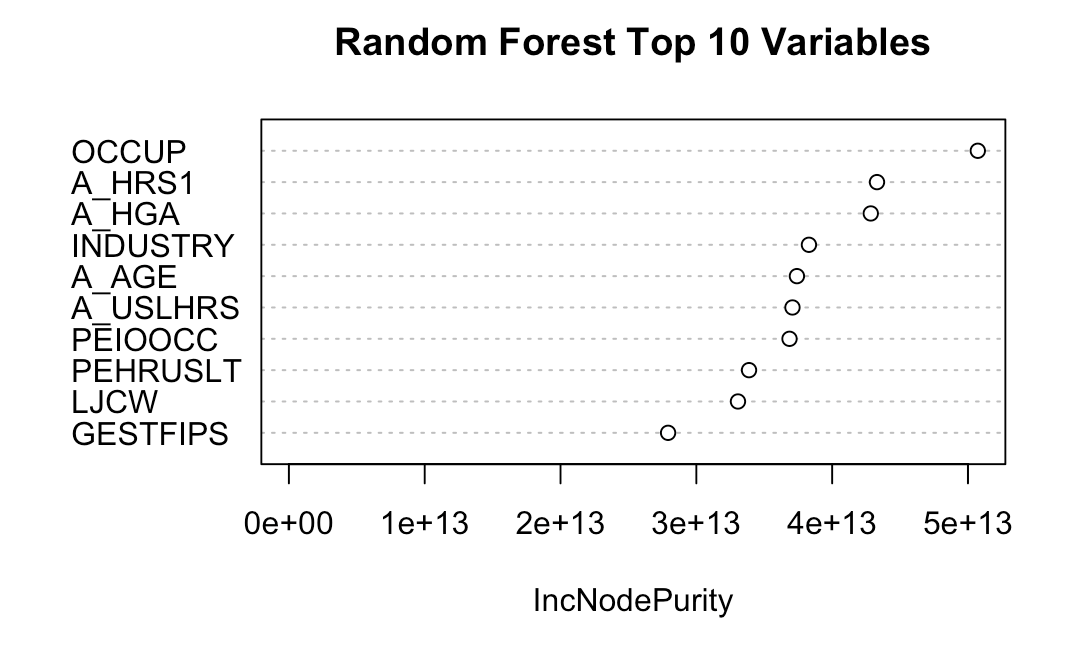


Figure 16: Random Forest

## 3.7 Gradient Boosting

Ensemble learning is the process of combining multiple machine learning models to solve a particular problem. This approach provides more accurate predictions than a single individual model, and it also reduces the chances of choosing a single poor model. There are different types of ensemble learning, with Sequential Ensemble Learning (Boosting) being one of them (Lyashenko, n.d.). The idea behind boosting algorithms is to build an ensemble by training each new base model, also defined as a weak learner, to emphasize on the errors the previous model made. This means each consecutive model is learning from the mistakes of the previous one and therefore boosting the learning. A weak learner is a classifier that can have a weak correlation with the actual value, therefore combining weak learner’s predictions can result in a strong learner.

Weak learners are generated using simple base learning algorithms which could be classified between Tree and Linear base learners (Malik, 2020). Most boosting algorithms work by default with tree-based learners. A decision tree is a supervised machine learning algorithm used to predict a dependent variable based on several independent variables. There are 2 types of decision trees which are Classification Trees, and Regression Trees (Malik, 2020). In Classification Trees, the target variable is a fixed or categorical variable, so the algorithm attempts to predict which category the target should belong to. In Regression Trees, the target variable is continuous and as a result, the tree attempts to predict the value of the target variable. For both Gradient Boosting and XGboost, Regression Trees were used as the base learners considering they are superior for regression problems (Muller, 2018).

Gradient boosting is a supervised machine learning, boosting algorithm in which errors are minimized by a gradient descent algorithm. It is also an ensemble algorithm. It adjusts weights using a gradient, which is a direction in the loss function. The gradient is adjusted by optimizing the loss of the model by updating weights during each iteration. Loss is the difference between the predicted and actual value, so the gradient descent algorithm decides which weight to use for each subsequent model based on the errors from the previous model.

We used the GradientBoostingRegressor class from the sklearn.ensemble package from Python to model our dataset. Gradient boosting has several hyperparameters. Hyperparameters are set before training of the data and are defined as different variables that can be tuned to improve the performance of the algorithm (Koehrsen, 2018). GradientBoostingRegressor comes out of the box with default values, however we decided to adjust some of them to improve the model. We used a learning\_rate of 0.01, which controls how fast the model learns (usually a lower value is preferred, meaning the model is learning at a slower pace), a max\_depth of 8, which determines the depth of each tree, and a n\_estimators of 500, which is the number of trees in the model (Dash, 2020). With these hyperparameters, gradient boosting yielded a R-squared value of 0.47 and a RMSE value of 43,800 for the data with outliers. As for the dataset without outliers, a R-squared value of 0.8 and a RMSE of 11,700 were obtained. As with the previous discussed algorithms, we saw a significant improvement of the model using the dataset without outliers.

## 3.8 Extreme Gradient Boosting

Extreme Gradient Boosting, or XGboost, is another boosting algorithm, with the already described gradient boosting framework at its core. However, XGboost has been optimized to improve performance and speed, as well as the ability to train large datasets. Some additional enhancements include Tree Pruning, which helps to prevent overfitting of the training data by replacing nodes that do not contribute to improving the classification on leaves (Malik, 2020), and making the core algorithm parallelizable. This means XGboost can fully use multi-core computers, and across distributed networks which is beneficial when dealing with large datasets (Pathak, 2019). It is no surprise that XGboost is one of the most popular machine learning models today, having won many Kaggle competitions because of its speed and accuracy when dealing with both regression and classification problems.

The XGboost package was used in Python as well as R to run the XGboost algorithm. As with the case of gradient boosting, we decided to use different values than the default ones for a few of the algorithm hyperparameters. We decided on a learning\_rate of 0.03, a max\_depth of 8, and n\_estimators of 500. Using the dataset with outliers, XGboost scored a value of 0.49 for R-squared, and 43,300 for RMSE. As for the dataset without outliers, XGboost scored a value of 0.81 for R-squared, and 11,467 for RMSE.

## 3.9 Algorithm Comparison and Selection

The results from each algorithm with little or no tuning are presented below in table 3.3. The analysis of these statistical metrics guided selection of our final algorithm and tuning process.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Using Final\_Data\_With\_Outliers** | | **Using Final\_Data\_No\_Outliers** | |
| **Algorithm** | **RMSE** | **R-squared** | **RMSE** | **R-squared** |
| LinReg | 48069 | 0.37 | 13442 | 0.74 |
| Lasso | 46928 | 0.38 | 19238 | 0.7 |
| RandomF | 44995 | 0.45 | 11774 | 0.8 |
| GB | 43863 | 0.47 | 11764 | 0.8 |
| XGB | 43339 | 0.49 | 11467 | 0.81 |

*Table 3.3: RMSE and R-squared values for each algorithm using the datasets with and without outliers.*

While creating the different models using the different algorithms we have presented, the datasets (with outliers and without outliers) were portioned into two sets, one for training containing 80% of the total data, and 20% for testing.

Overall, XGboost was the best performing algorithm for both datasets. Gradient boosting and Random Forest also performed well, with regular Linear and Lasso regression performing the worst in terms of R-squared and RMSE values. In addition, there is a significant improvement in performance for all algorithms using the dataset without outliers. R-squared values increased significantly, and even doubled for linear regression. For the RMSE metric we observed a similar result. Using the dataset without outliers reduced RMSE values more than half for all algorithms. The best result was obtained with the XGboost algorithm on the dataset without outliers. This model had the highest R-squared value at 0.81, and the lowest RMSE value at 11,467. These results can be seen in figure 17 for R-squared values, and figure 18 for RMSE values.

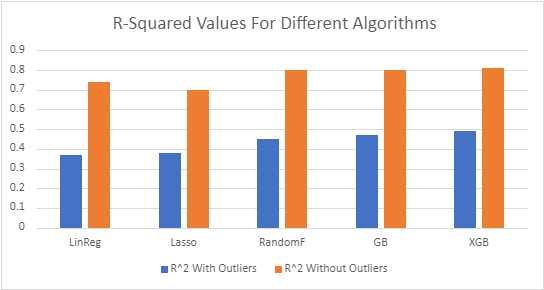


Figure 18: Bar graph comparing the RMSE values obtained for different algorithms using the dataset with and without outliers.

Figure 17: Bar graph comparing the R-squared values obtained for different algorithms using the dataset with and without outliers.

## 3.10 AWS SageMaker and XGBoost Tuning

AWS SageMaker is a fully managed machine learning service within the AWS ecosystem. The goal is to allow data scientists and developers to quickly build, train, and deploy machine learning models without having to worry about the technicalities of installing software (What Is Amazon SageMaker? - Amazon SageMaker, n.d.) It provides an integrated Jupyter notebook instance service, as well as a fully integrated machine learning environment called SageMaker Studio. One of the nice features of SageMaker studio is the ability to run models on autopilot mode. Once a dataset is provided, and a target variable is specified, autopilot will run a series of experiments to determine the best performing algorithm. It will also run additional jobs to determine the best combination of hyperparameter values. The result is a series of jobs in which one is marked as the ‘best’ job. This job contains a model that is stored in an S3 bucket, and that can be deployed into a specified EC2 instance and an endpoint configuration to access the model from an outside source.

Based on the results we obtained using the different algorithms, it was clear that XGboost, and the dataset without outliers provided the best results. We decided to run an autopilot experiment using the dataset without outliers. After 2 hours and 250 jobs, SageMaker autopilot also selected XGboost as the best performing algorithm. It did try to use other built-in algorithms such as linear learner and mlp, but XGboost provided the best results. The job also provided us with the exact values for some of the hyperparameters. We used these values in our local machines to run the model again, and we obtained similar results. From SageMaker, we were able to identify additional values for other hyperparameters we had not previously explored. These included the objective, which specifies the function to optimize during model training, in our case to minimize R-squared, gamma, defined as “minimum loss reduction required to make a further partition on a leaf node of the tree” (XGBoost Hyperparameters - Amazon SageMaker, n.d.), and colsample\_bytree, defined as a subsample ratio of columns when constructing each tree.

With the combination of SageMaker’s suggested tuning parameters, and custom-built hyperparameter grid-searching code in R, we were able to improve our baseline XGBoost results to an R-squared value of .82 and an RMSE of 11295. The resulting XGBoost model with tuned hyperparameters was used in the final analysis of our dataset, which can be seen in section 4.5 below.

# Visualization

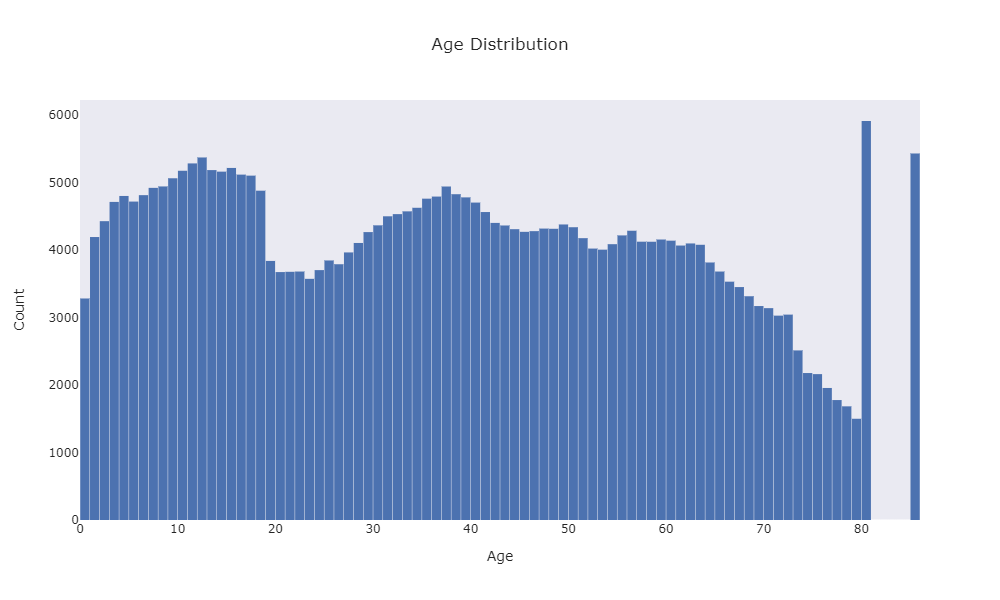
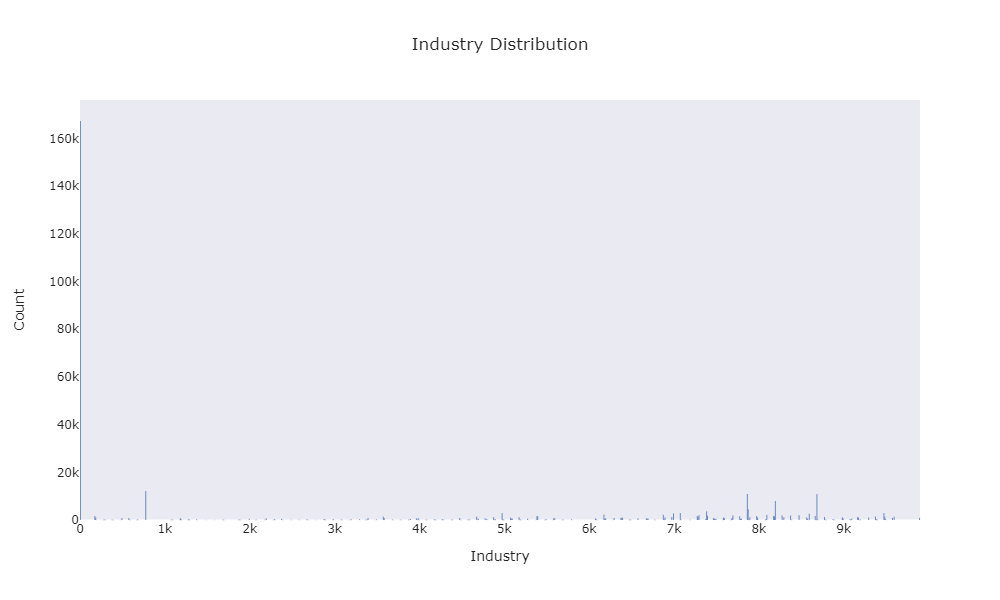
Before diving deeper into the exploration of our dataset, the Hydrogen team wanted to first explore if there were any disparities between salary and education, gender, location, occupation and race. We wanted to complete our initial visualization on these attributes since we had mentioned them in the background and rational. This initial exploration will also allow us to compare our findings to those we found during our initial research.

According to Tableau, data visualization should “…[grab] our interest and [keep] our eyes on the message.” To make our data easy to understand and follow, we will be using scatterplots, bar graphs, Tableau, pie charts, and histograms.

## Risk Associated with the Visualizations

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | **Description** | **Probability of Occurrence** | **Impact** | **Mitigation** | **Status** |
| 1 | If we incorrectly select visualization algorithms for our data, then we will misrepresent our findings. | 1 | 3 | Research and verify visualizations | Current |
| 2 | If the visualizations do not show strong correlations between the data or an accurate model, then we need to adjust our calculations and/or find other data | 1 | 2 | Exploratory data analysis has shown good correlation and accurate models | Current |
| 3 | If the visualizations we select show values or trends contrary to our analysis, then we will need to verify those visualizations and potentially adjust our study direction. | 1 | 3 | Visualizations have been consistent with findings thus far | Current |

## Distribution of Attributes in Dataset with Outliers



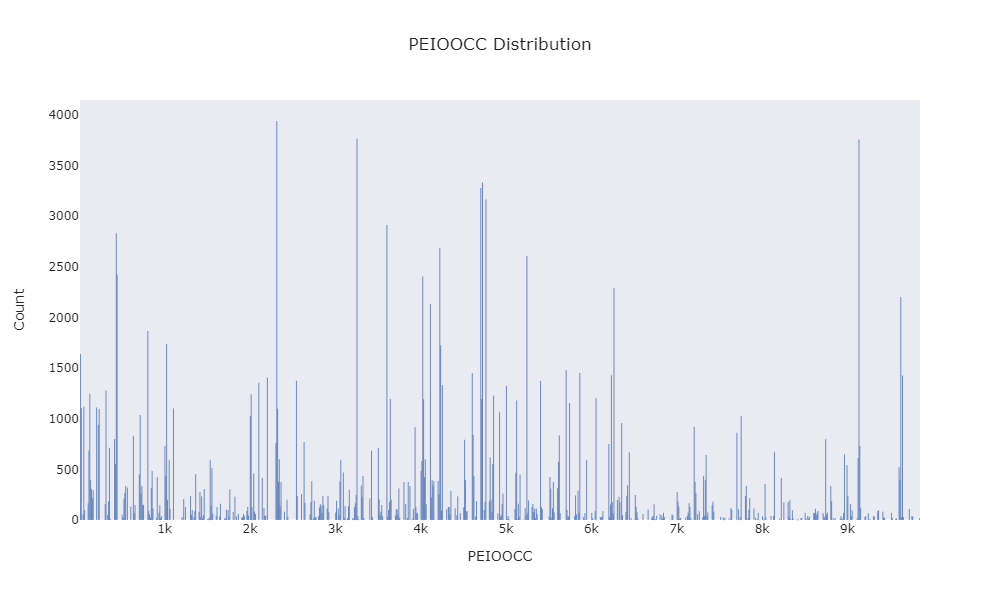


Figure 19: Distribution of different variables using histograms.

In figure 19, we can observe the population distribution of the industry (INDUSTRY), age (A\_AGE), and occupation code using a 4-digit code that is most specific (PEIOOCC). Both industry and PEIOOCC consist of 4-digit codes that represent different industries. These codes range from 0 to 10,000. It was challenging to visualize the distribution of the industry codes as not all consecutive values refer to similar industry types. For the industry plot, the mode is at the zero category, which corresponds to not in universe. The second highest peak manifested at 770, which corresponds to the construction industry. For PEIOOCC, there are peaks at the 2310 value which corresponds to Elementary and middle school teachers, between 3250 and 3259 which include veterinarians and nurses, and at 9130 which corresponds to driver and sales workers. In the age distribution we can observe values from 0 to 85. The mean was approximately 38 years. It is important to notice that a significant amount of the data corresponded to individuals under the age of 20.

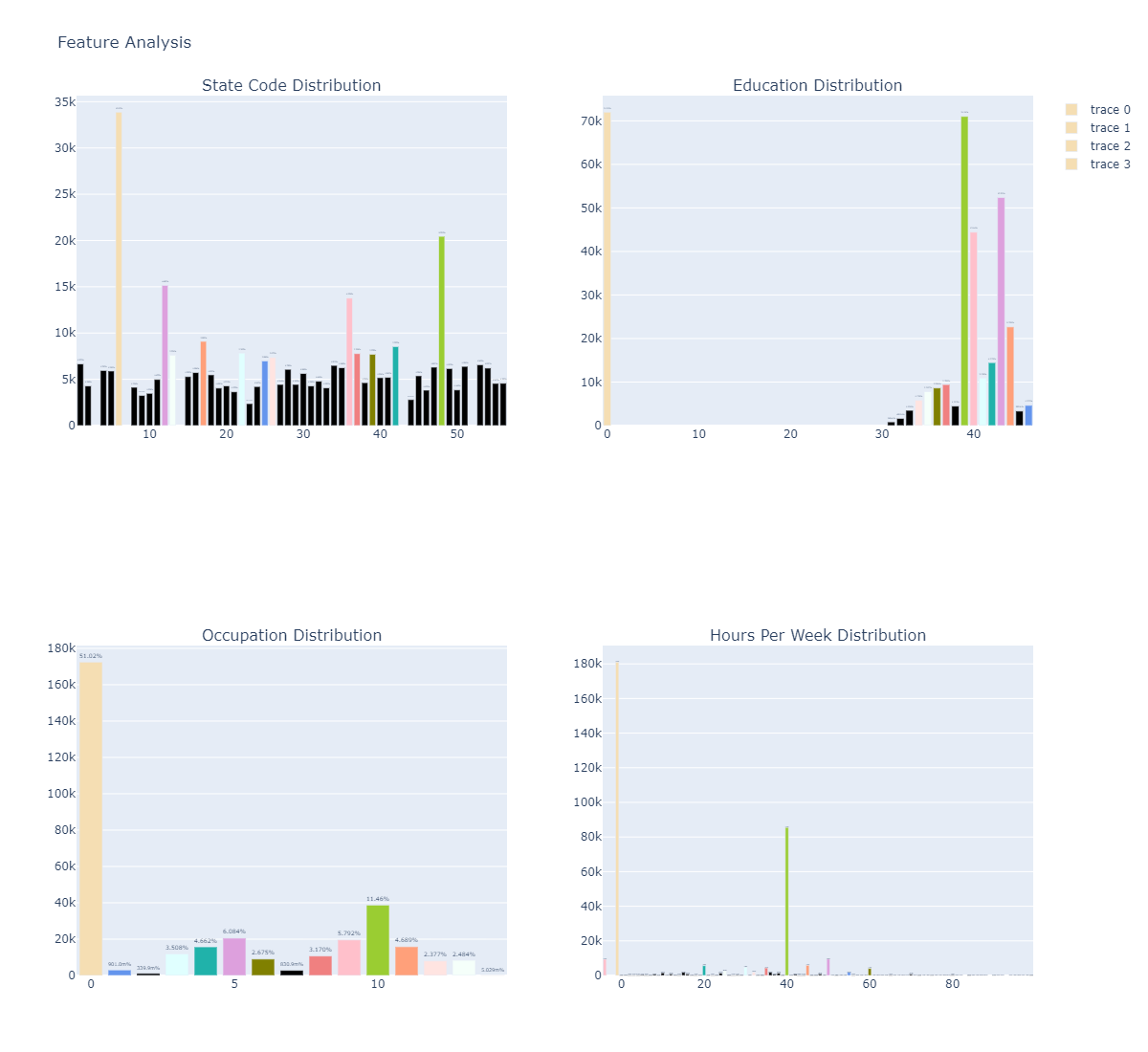
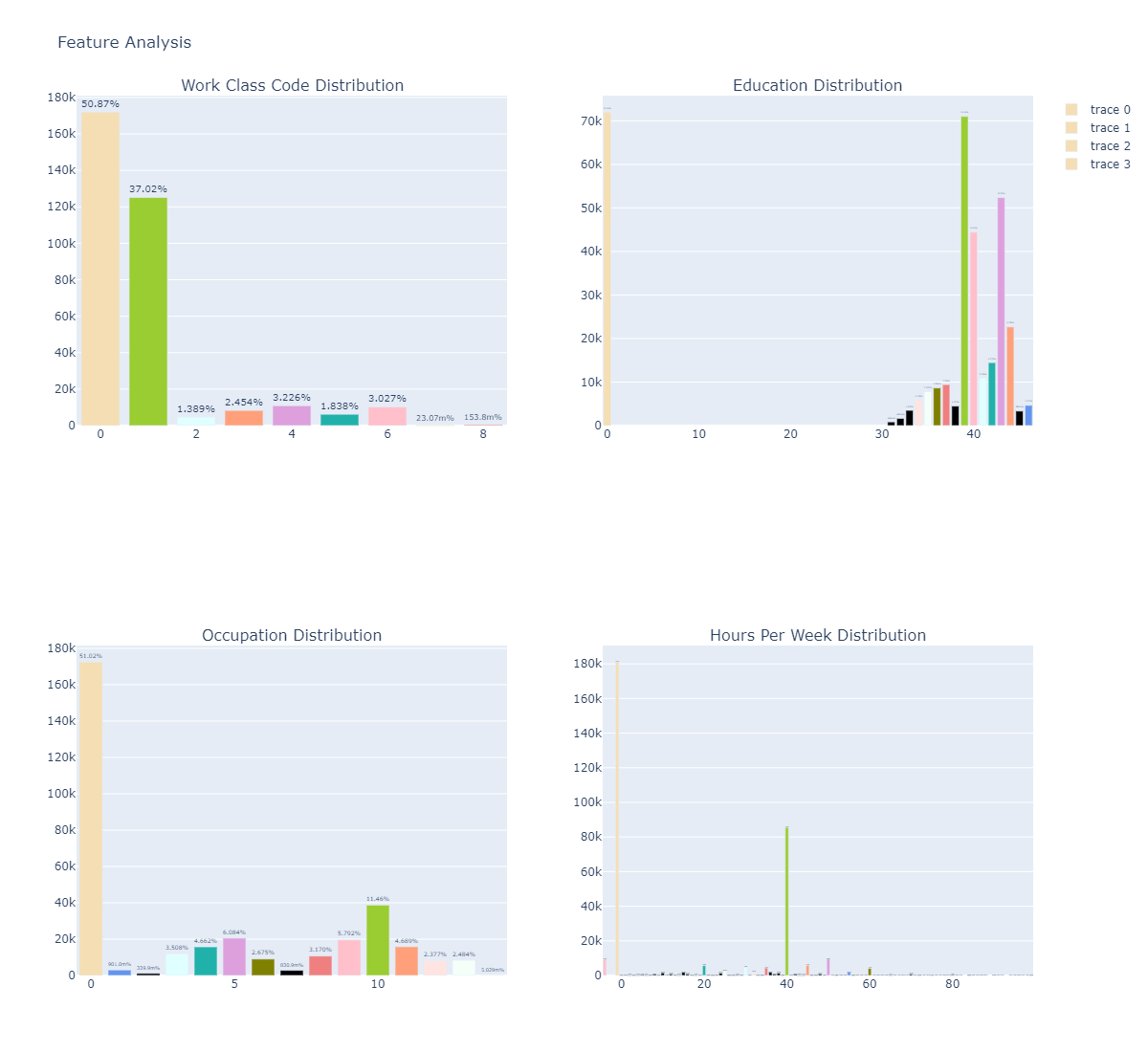
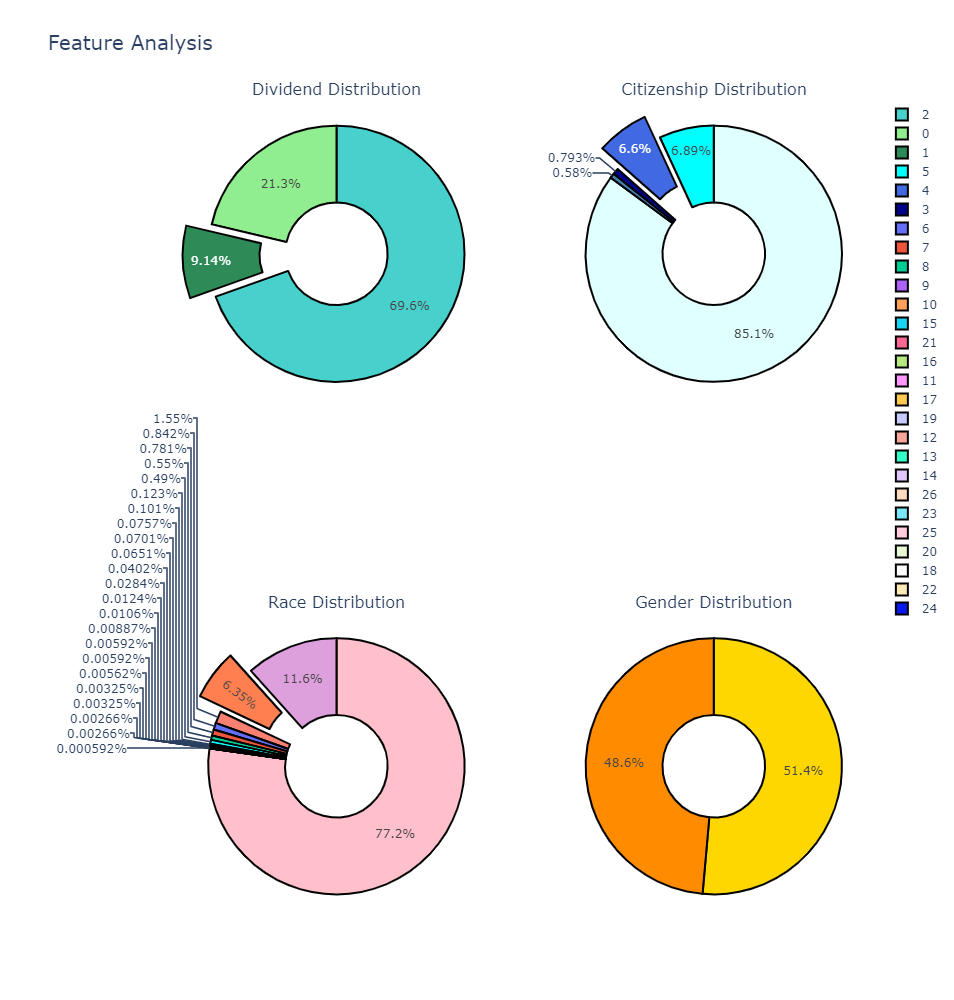


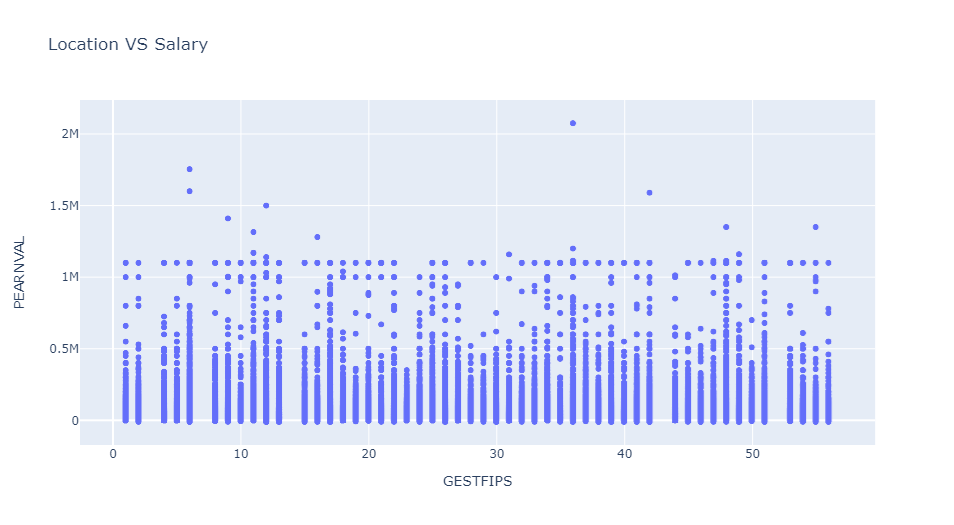
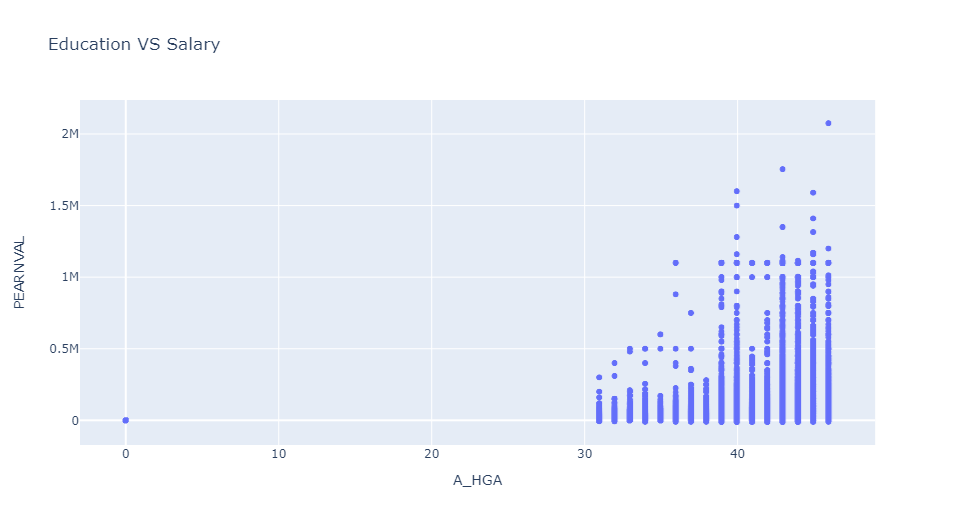
Figure 20: distribution of different variables using bar graphs.

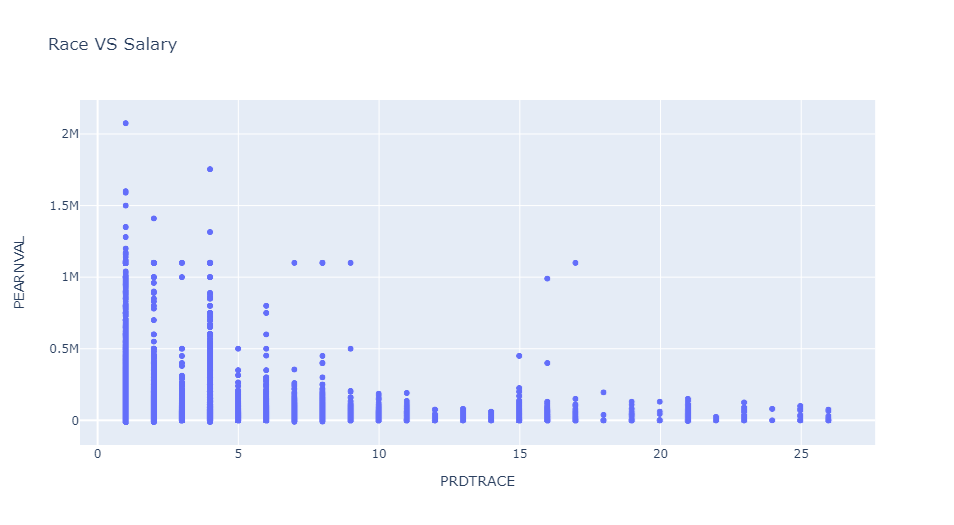
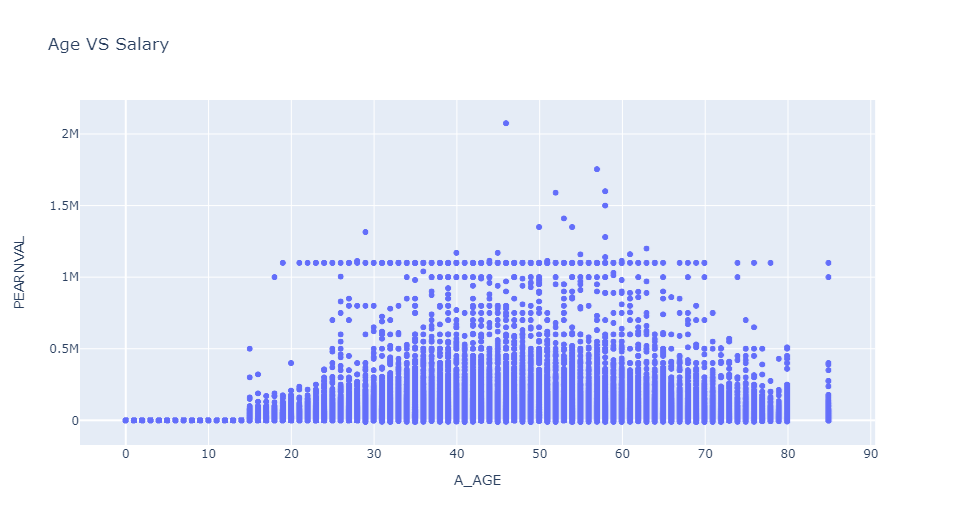
Figure 20 shows the distribution of data for 5 different predictor variables. These include work class (A\_CLSWKR), education (A\_HGA), occupation group code (A\_MJIND), number of hours worked per week (A\_USLHRS), and the unique state code (GESTFIPS). From all the graphs we can observe that not all the variables are normally distributed. For the work class variable, almost 80% of the entire data corresponded to the 0 and 1 codes. The zero code is the not in universe category, most likely children, while the 1 code corresponded to private workers. For the education variable, there are 2 modes with values of 0 and 39. These values correspond to the children and high school graduate categories, respectively. For the occupation distribution, the mode is at the zero category, with a second largest count at the category with value of 10 (around 11.46% of the data). The zero category correspond to children and the 10 value is for educational and health services. For the hours per week variable, 53% reported zero of hours, and 25% percent reported 40 hours, the second largest after zero hours. As for the state code, around 10% of the data (the largest) corresponded to the category 6, which is the code for California, while 6% (second largest) corresponded to the category 48, which is Texas.

Figure 21: Pie charts showing the distribution of different predictor variables.



In figure 21, we used pie charts to demonstrate the distribution of dividend received (DIV\_YN), citizenship status (PRCITSHP), race (PRDTRACE), gender (A\_SEX), weeks worked the previous year (WEXP), and the longest job class of worker from the previous year (CLWK). As seen in the pie chart, almost 70% of the data in the dividend distribution correspond to the category with value of 2, which is no dividend. Only 9% did receive a dividend. 85% of the people in our dataset were in the category of value 1 for citizenship status, which corresponds to native, born in the US. For race, a significant majority were in the category with value of 1, which corresponds to White only. It is followed by Black only at 11.6%, and Asian only at 6.35%. For gender, 51.4% were females, and 48.6% were males. WEXP had a 35.6% of the data in the category with value of 1, which corresponds to 50 to 52 weeks or full time in the previous year. The second highest percentage in WEXP corresponded to a nonworker. For CLWK, the category of value 1, which corresponds to a private class of worker, was the highest at 41%, followed by never worked at 28%.





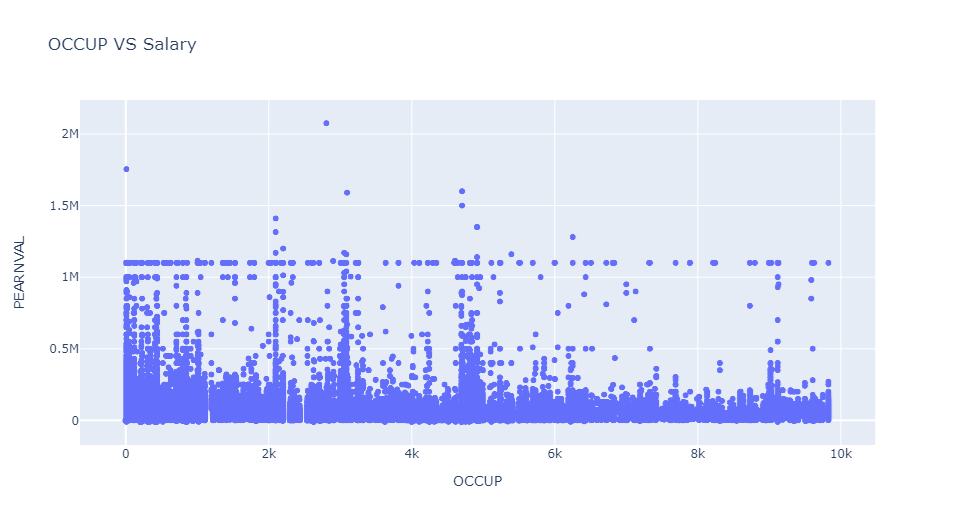
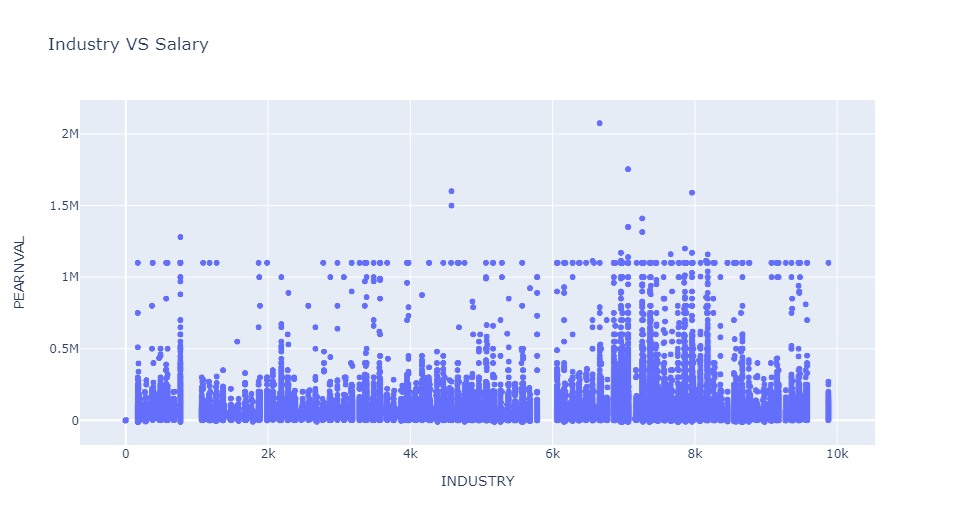


Figure 22: Relationship between salary (PEARNVAL) and different predictor variables using scatter plots.

Figure 22 shows the relationship between the predictor variables education (A\_HGA), location (GESTFIPS), age (A\_AGE), race (PRDTRACE), industry code (INDUSTRY), and occupation of longest job in the previous year (OCCUP), and the target value of salary (PEARNVAL). From the education plot, we can observe a positive correlation. Higher values for the age codes correspond to higher education attainments as well. The highest salary value in the dataset (an outlier) at 2.1M, is for a person in the 46 categories, which is a doctorate degree. For location, we can see a very even distribution, with all states reporting high salary values, with New York being the one with the highest reported salary. For age, there is a higher concentration of high salaries after age 45, with the highest at 46. However, it is to notice that even relatively low ages also reported high salaries, with people at 19 and early 20’s reporting over 1M. For race, the shape seems to decrease rapidly after the value of 3. The majority of high salaries belong to the categories 1, 2, and 4, or White, Black, and Asian, with White being the highest. Both graphs for industry and OCCUP are share similarities. For example, a wide variety of both industry and OCCUP codes reported salaries over 1M. The highest salary corresponded to the 6670-industry code, which is radio and television broadcasting and cable, and the, and the OCCUP code of 2810, which corresponds to news analysts, reporters, and journalists.

## 4.3 - Salary Distribution with Outliers

We can see from figure 23 below that there are outliers when salary is distributed across the industries present in this census data.

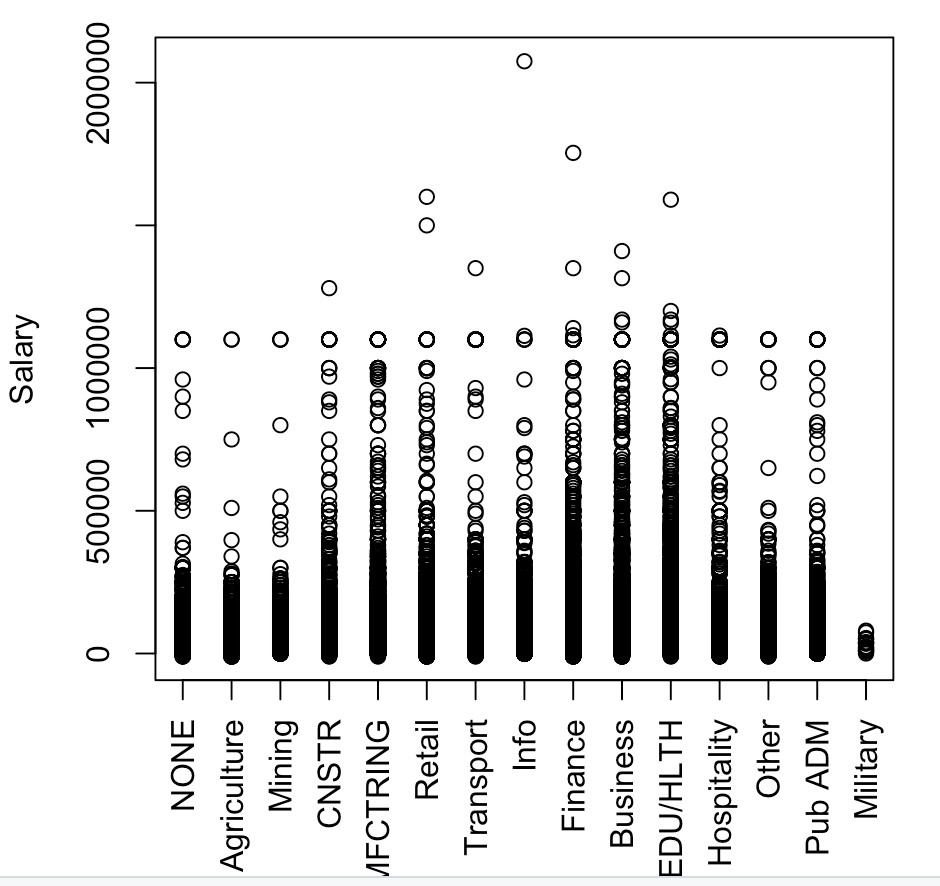
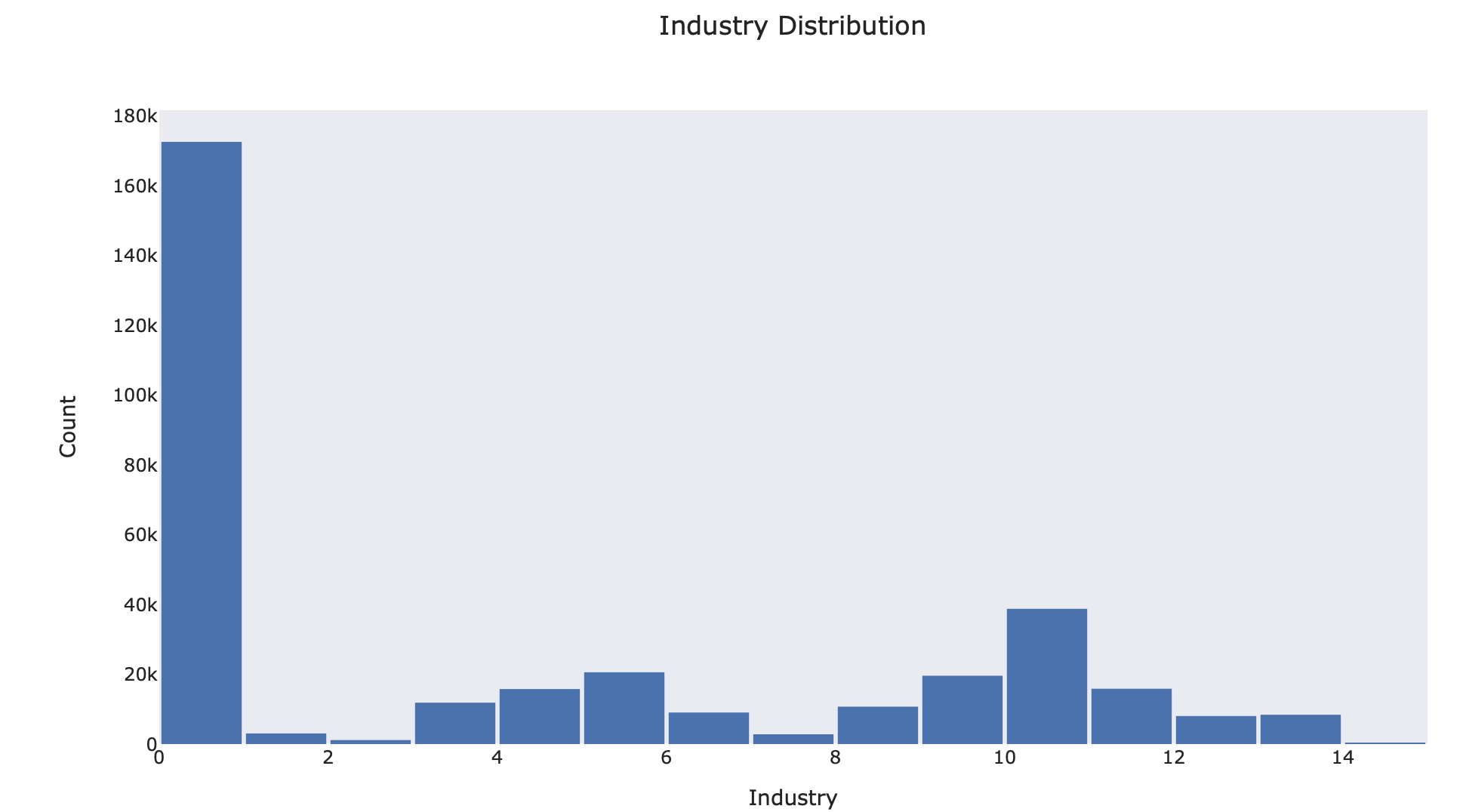


Figure 23: Salary Distribution Across Industries with Outliers

Figure 24 shows the distribution of industries as represented across the census dataset. This is the distribution with outliers included.



*Figure 24: Industry Representation with outliers*

It is interesting to note in the above 2 pictures that the industry of ‘None” seems to be very well represented in the data set and workers within that industry appear to make a fair amount of money.

When distributing salaries across the worker class, the outliers are verified as seen in figure 25:

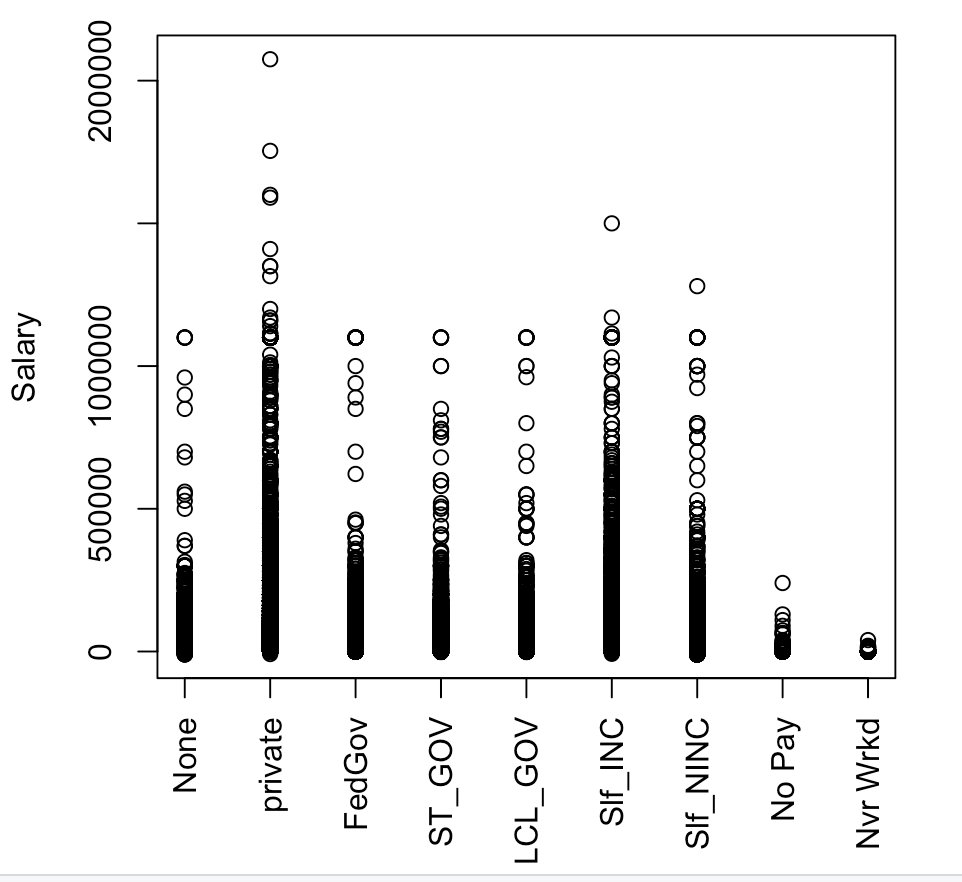


Figure 25: Salary Distribution Across Classes of Worker with Outliers as shown in R

Figure 26 below shows the class of workers, with outliers, as distributed across the dataset:

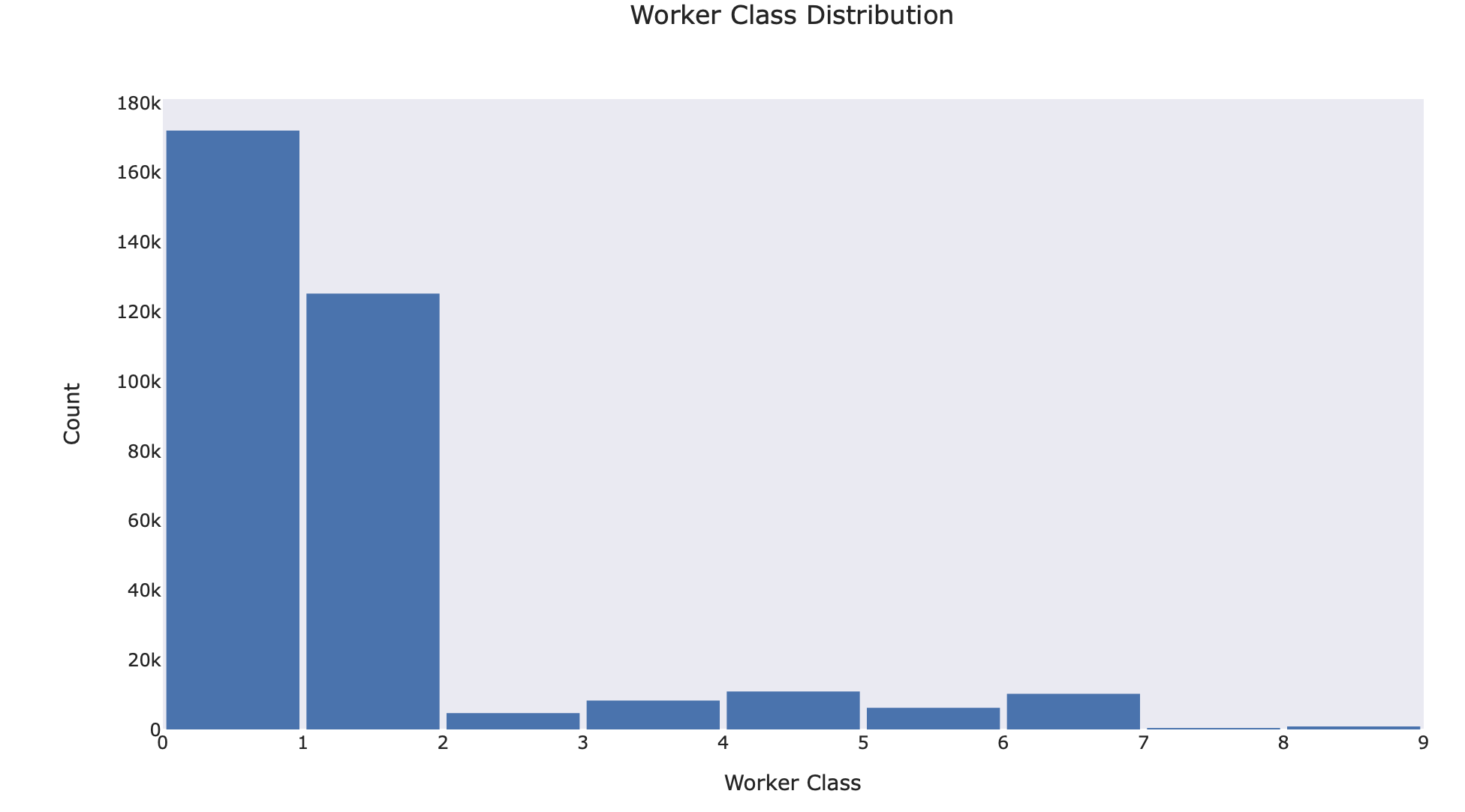


Figure 26: Worker Class distribution with Outliers included, as seen in Python

The worker class of “None” is well represented again in the pictures above. There seems to be high representation of workers of the class “None” within the dataset and those workers seem to make a fair amount of money.

## 4.4 - Salary Distribution without Outliers

When the outliers are removed using the IQR method previously mentioned, we see the distribution of salaries across industries as shown in figure 27.

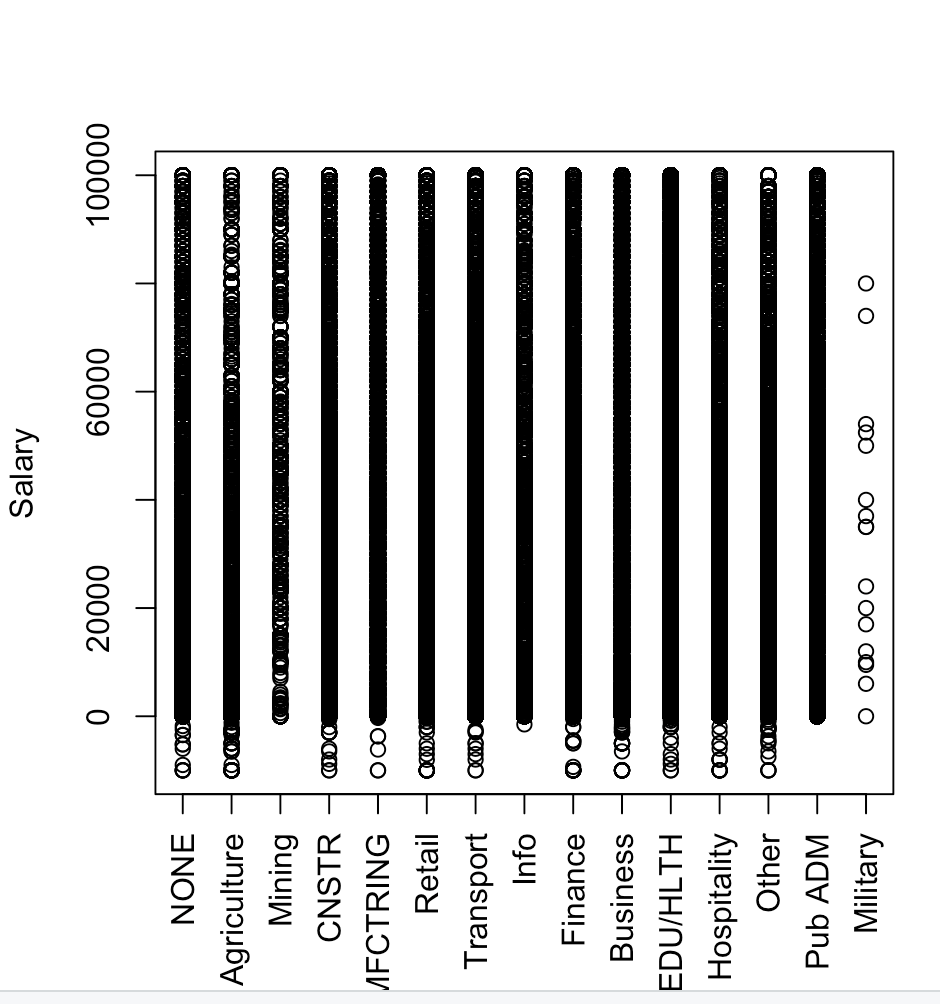


Figure 27: Salary Distribution Across Industries with No Outliers as seen in R

Again, the industry of “None” is well represented and distributed similar to other industries. The 14 industries are very similar with outliers removed with the exception of industry 14, which is the Armed Forces. Armed forces members are uniquely surveyed by the census in that they participate only if they live with a civilian. (U.S. Census Bureau, 2020).

Figure 28 shows the distribution of industry across the dataset without outliers:



Figure 28: Industry distribution across the dataset without outliers, as seen in Python

We see (figure 29) the salary distribution across the class of workers with the outliers removed.

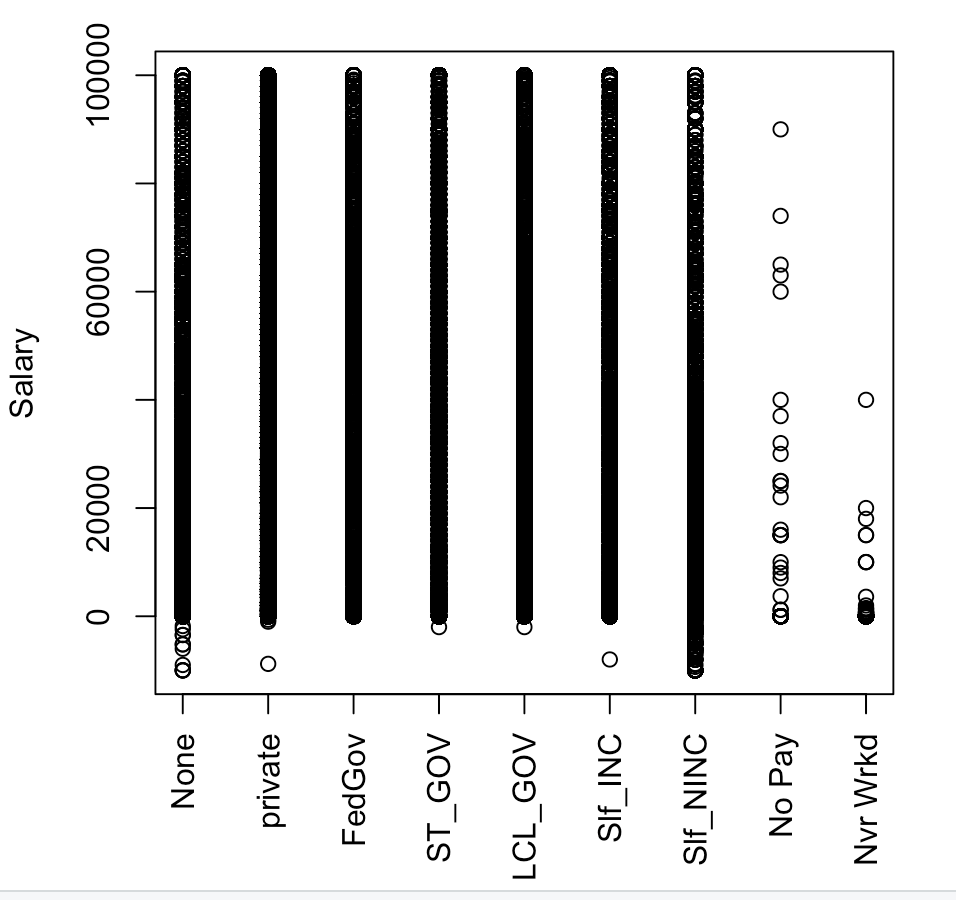


Figure 29: Salary Distribution Across Classes of Worker, No Outliers as seen in R

and in figure 30, we see the distribution of the class of workers across the dataset, with outliers removed:



Figure 30: Distribution of the class of workers across the dataset, no outliers, as seen in Python

Based on the figures above, removing the outliers has a salary upper limit of around 100,000. An interesting result from removing the outliers can be seen in the scale of the Y axis. When outliers are included the salary scales to 200,000 for the visualizations. When outliers are removed the y axis scales to 100,000.

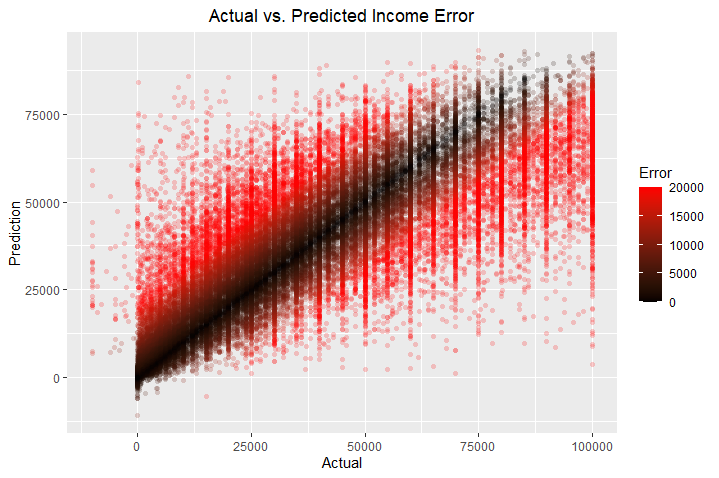
## 4.5 – XGBoost and Sage Maker

The above graphs display the variability that came from involving outliers in our data. It was no surprise that our XGBoost model (figure 31) worked best with outliers removed as well. In conjunction with Amazon SageMaker Studio, we found the best parameters for this dataset. Below you will see the various shrinkage values, or learning rates, that we tested across a variety of iterations. We found that the best learning rate was 0.015 with 2000 iterations. Without hardware limitations we could pursue further iterations and different learning rates, but we felt that the returns with these adjustments would be minimal.



Figure 31: Shrinkage rates across boosting iterations and corresponding RMSE

After determining the best parameters with XGBoost tuning, we felt we achieved our best possible model given the hardware constraints. Below (figure 32), you will find our Actual vs. Predicted outcome using our final model.

Figure 32: Actual salary vs. Predicted Salary

Additionally, you will find a distribution of our modeling error in figure 33.

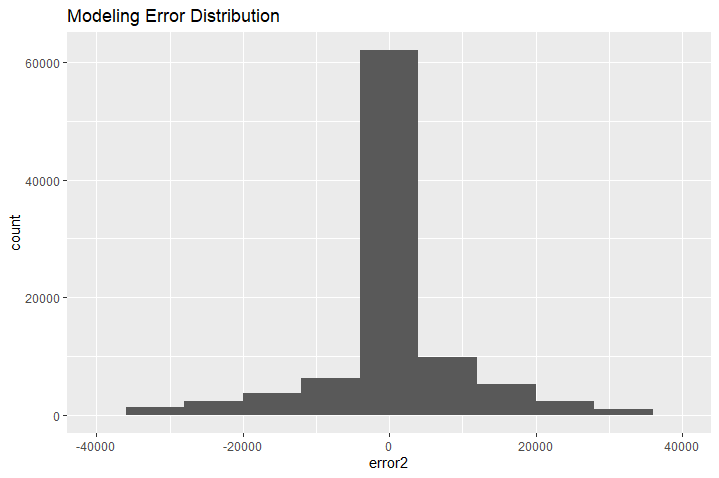


Figure 33: Error Distribution

The xgboost package in Python also provides with a function to visualize the importance of each attribute in the model. Figure 34 shows the variables in order of F-score. For our model, age (A\_AGE) had the highest contribution with a score of 10,963, followed closely by state code (GESTFIPS) with 10,115. In contrast, A\_MJIND, the generic industry code, had the lowest contribution out of all 40 variables.

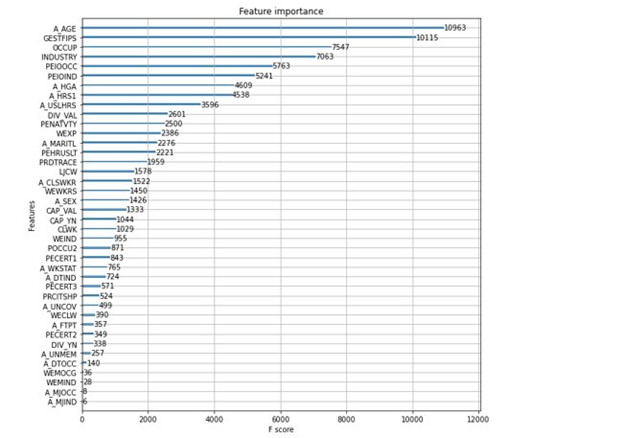


Figure 34: Feature importance based on F-score as computed by our model using the xgboost package in Python.



Figure 35: SageMaker job description of best performing algorithm, including feature importance.

Figure 35 shows the results we obtained from running experiments in autopilot mode using AWS SageMaker. From the figure we can observe how XGBoost was picked as the best performing algorithm. SageMaker also provides feature importance classification. It uses SHAP values to rank the importance of each feature. WEXP was selected as the most important, followed by CLWK. Variables like age (A\_AGE), education (A\_HGA), state code (GESTFIPS), industry (INDUSTRY) and OCCUP codes proved to be significant in both the xgboost test from Python, and the SHAP values in SageMaker.

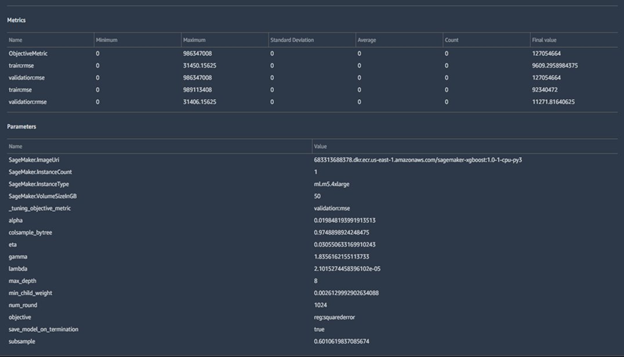


Figure 36: Additional information of the best performing job in SageMaker. Hyperparemeter values for the job are shown, along with the score for performance metrics.

In figure 36, additional information of the best performing AWS SageMaker job is shown. The complete values for each hyperparameter are stated, as well as the values for RMSE for both the training and testing dataset. The training data set did have a lower RMSE value at 9,609, in comparison to the validation data at 11,271. The difference between both values could be attributed to overfitting of the training dataset.

## 4.6 – Tableau: Salary by Gender

One of our scenarios was; An 18-year-old student has just started her first job, and she’s starting to think about where her financial situation might go in the future. She’s a little dismayed to see that her salary is quite low, so she uses our group’s tool to see how her current information might affect her future earnings. She also uses the tool to see if there are any factors that she might have some control over to help boost her future earnings potential. We believe visualizing how gender affects her earning potentials would help her. We visualized the earning potentials of both genders according to industry, education, and class of workers.

We believe an interactive dashboard would allow for a simplified exploration and explanation of our dataset because it removes the need for the reader to know or understand R and Python. An interactive dashboard also allows those without knowledge of Sagemaker, XGB or algorithms in general to filter and tailor the dataset to fit their goals.

## 4.6.1- Distribution of salary by Gender and Industry of Worker

Tableau is regarded as one of the best and simplest way to visualize data, so we wanted to utilize it on our dataset. In our initial findings outlined in the introduction, we found that education, occupation and gender, location all played a role in salary/earnings potential. Using Tableau, we were able visualize median earning by gender and state. This distribution is depicted in figure 37.

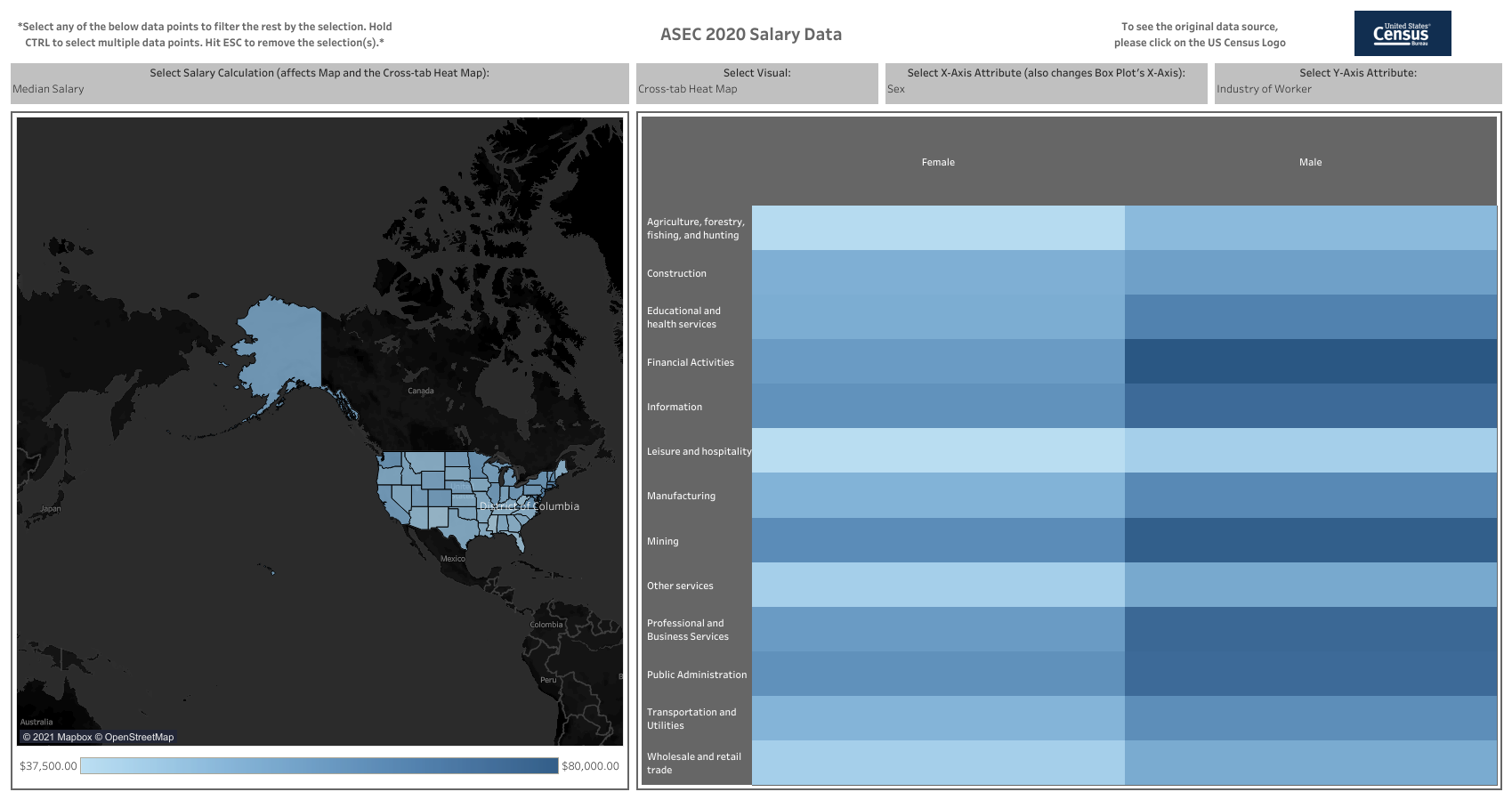


Figure 37: Salary Distribution by Gender and industry (Right) and State (Left)

Looking at the heat map on the right, the first observation that clear is how men have a wide spread of high paying industries, while women have a wider spread of lower paying industries. The highest median paying industry for men was Finance, with a salary of $78,000. For the same industry, women made $50,000. The highest paying industry for women was mining, with a median salary of $55,900. For the same industry, men were paid $75,000. This visualization and our dataset were able to prove the income disparities discussed in-deeper in the introduction. Our findings were in line with the Economic Policy Institutes’ conclusion that “[t]he largest gender wage gap occurs among the highest-paid workers, with higher-earning women facing a 29.2% pay penalty.”

## 4.6.2- Distribution of salary by Gender and Educational Attainment

We believe education is an important attribute when looking at salary. This assumption is supported by the “Random Forest top 10 Variables,” where education was third on the list. Looking at the tableau dashboard depicted in figure 38, the highest earners from both genders had professional school degrees (MD, DSS), but the lowest earners varied for women, the highest earners had a professional school degree with a median income of $98,000, and the lowest earners had completed less than 1st grade, resulting in a median salary of $18,000. For men, a professional school degree had the highest salary, with a median salary of $138,3000 and the lowest earner had only completed up to 11th grade and the median salary was $30,0000.



Figure 38: Salary Distribution by Gender and Education Attainment

# Findings

## End User Changed

Our initial thoughts about the use case for our model indicated that it could be used by a variety of users. After progressing through our analysis, we have realized that predicting salary is far more difficult than initially anticipated and resulted in us limiting the scope of our predictions, thus limiting the use cases of our model. The team decided to focus on salary predictions from the perspective of an employee and not from the perspective of a business owner.

## Improvements through Additional Attributes

Through the course of our exploratory data analysis and our model development, we found that some of our initial decisions regarding our dataset may have limited our ability to predict salary. Taking inspiration from the UC Irvine Adult Data Set, we narrowed down the 2020 ASEC data from 840 attributes to 20 attributes that were either specifically listed in the Adult Data Set or we felt would be relevant to the problem at hand. Initially, we also brought in a Cost-of-Living Index from the State of Missouri with the belief that we would have to normalize the salary data to accommodate for the location of the survey respondent (Missouri Economic Research and Information Center, 2020).

We believed this initial dataset, with over 157,960 instances and 20 attributes, to be more than comprehensive enough to be able to predict the salary of an individual. These beliefs and assumptions were unfounded as our exploratory data analysis yielded less than satisfying results and the cost-of-living index proved to be meaningless. Using gradient boosting on this dataset, our model capped at a 0.4 R-squared value. These disappointments led to a re-evaluation of our attribute and data selection. In the end, we decided to add an additional year of data, 2019, and focus on only attributes that were present in the ASEC data. This led to a final set of data that was comprised of 338,060 instances and 40 attributes.

Exploratory data analysis on this dataset yielded slightly more promising results, but still disappointing in the context of our goal of predicting salary. Our best model with this dataset, XGBoost, yielded an accuracy of 0.49 (see table 3.3). With our algorithms still yielding poor results, the team moved on to data conditioning methods.

## Data Conditioning and Outliers Removal

While taking a closer look at the distribution curve of our target variable in our data set, we noticed it was quite skewed to the right, with the curve starting to flatten at around $120,000. Utilizing the inter-quartile method, we found the upper and lower bounds of salary, which were defined as -$60,000 to $100,000. This new dataset that involved removing the outliers resulted in a 5.6% reduction in the initial dataset down to 318,872 instances.

One of the main reasons our initial models struggled to handle the dataset when outliers were present could be the result of how much of our data was composed of younger people, especially children. More than 50% of the data were for children and younger people who did not work and had no earnings. This affected the salary distribution and made higher salaries more difficult to be accurately predicted.

While we still pursued modeling the dataset with outliers, it was very clear that the new dataset without outliers was producing better results. The correlation table, shown in Figure 11, clearly shows a much higher correlation between the attributes and salary than any previous correlations displayed with the prior datasets. We found the five most correlated attributes to be A\_HRS1 (0.74), A\_USLHRS (0.73), PEHRUSLT (0.73), INDUSTRY (0.62), and A\_MJIND (0.61) (Please refer to the data dictionary at the GitHub link below for attribute details). These correlation results appeared to be verification that our decision to subset the data was a good decision with regards to our end goal of salary prediction. The outliers, as displayed in Figure 10, were simply too difficult to predict given the small number of instances they comprised.

The average median salary in the United States in 2019 was about $47,000. When looking at the distribution of our dataset, and identifying outliers after $100,000, it does demonstrate how it is accurate with the real world. Individuals making over $100,000 are also way above the reported median.

## Finding the Right Algorithms - XGBoost

Using our final dataset without outliers, we proceeded to test various modeling techniques, while also testing these modeling techniques on the dataset with outliers for comparison and validation of our decision to subset the data. Figures 17 and 18 display these comparisons. For both datasets we confirmed that XGBoost provided the highest level of accuracy. For the dataset with outliers, we achieved an R-squared of 0.49 compared to an R-squared of 0.81 without outliers. All of these initial algorithms were run with minimal tuning. Guided by the results of these initial modeling results, as well as results from AWS “autopilot” algorithm selection tool, our team selected XGBoost as the most promising algorithm. After tuning this algorithm’s hyperparameters with a custom grid-searching tool, we created an XGBoost model with an R-squared value of .82.

It is not surprising we obtained the best results using XGboost for our model algorithm. XGboost is one of the most used algorithms for machine learning across many companies, and it is sometimes preferred over more complicated algorithms such as neural networks.

One of the more interesting findings was that the Random Forest and Gradient Boosting models improved dramatically. Both models, with the new dataset, had an accuracy of 0.80 which was only 2% worse than our best model.

## Best and Important Attributes

When looking into which variables influenced salary the most, it is important to note that the answer to this question depends on which metric to look at. Since we used a variety of algorithms, we can choose from highly correlated values from our Correlation plots, F-Scores from our importance plot in Xgboost, and SHAP values from SageMaker. However, there were certain attributes that were relevant in all these graphs. Variables like age (A\_AGE), education (A\_HGA), state code (GESTFIPS), industry (INDUSTRY), number of hours worked (A\_USLHRS), and occupation (OCCUP) codes proved to be significant, regardless of the test and tool. These variables represent common assumptions people make when thinking about what factors can affect the potential salary of an individual. Our analysis demonstrated these assumptions to be true. As mentioned above, work experience and education tend to have a high positive correlation with salary, which indicates a person can focused on advancing their studies and getting more work experience if the goal is to potentially earn a higher salary.

## Attributes and Insights from Data

One of the top factors to achieve a high salary is based on hours worked. This indicates internships for early career employees or potentially getting a second, part-time job if an individual wanted to switch industries. Industry is the next highest correlation to salary. Based on the dataset with outliers, individuals in the health care, finance and information industries have the highest salaries. Individuals in the “Private” class of worker also have the highest salaries based on the data with outliers included.

For an individual to prioritize what to do to increase their salary; switching industries to health care, finance or information would be a good option based on the data and analysis. If this is not an option then education is the next highest correlated factor after hours worked, industry and worker class factors.

It is interesting to note how after removing outliers, different people across different industries, ages, and state reported high earnings. This means there is potential for a variety of people to obtain a high salary, although not the norm based on our findings.

# Summary

In this study we proved that industry and experience are the greatest indicators of a higher salary. While those factors seem intuitive, this study also showed that education does not play as high a role in earning potential as those factors, depending on the analysis methodology you are using. This study also proved that more data and more factors strengthen a model. Initially, the team was using approximately 20 factors and 150,000 records and not getting strong results. When the dataset was increased to more than 300,000 records and 40 factors the models almost doubled in accuracy.

# Future Work

**None Class Categories and Outliers**

We can see a lot of workers in the 0 (None) categories of industry and worker class. The workers in that 0 class make a fair amount of money. An interesting follow-on study would be to survey those individuals that answered “none” on the survey and see what industry or worker class they are in. Those answers would strengthen the existing data or lead to new categories being added for future studies.

Another interesting study would be to focus on the outliers. Studying the existing factors, and potentially adding more, could indicate what factors lead to earning a high income (over 100,000).

**Salary, Gender and Education**

One interesting discrepancy revealed by our algorithms and Tableau visualization was how education attainment resulted in higher earnings for women, but that was not always the case for men. The lowest earner for men completed 11th grade or less, while the lowest earners for women completed 1st grade or less. It would be interesting to find out why that is.

**Deploying Model and User Feedback**

A potential follow up could be to deploy our model into a testing environment for people to use, using AWS or another cloud provider. Within AWS, the model could be deployed into an EC2 instance with an enabled endpoint, making it open to the internet and other services to trigger it. As a result, a web-based application could invoke our model through an API using AWS API Gateway, and AWS Lambda to internally trigger a prediction using our model and the data passed through the request from the user, through the website. Having actual users could allow us to further test our model and adjust when needed to maintain a high level of accuracy.

# Appendix: Codes and Interactive Visualization

**Code:** all the code including R, Tableau, and Python files can be found in the following GitHub repository. The datasets in addition to the data dictionary can also be found in the same repository. Sample code from Kaggle projects were used during the development of some of these files (Adult Census Income Analysis 82% accuracy score, n.d.).

<https://github.com/uruenak/CS504>

**Tableau Interactive Dashboard**: a fully interactive dashboard for exploratory data analysis in Tableau of our dataset can be found in the below link.

[Census Salary Data - Ryan | Tableau Public](https://public.tableau.com/profile/ryan6250#!/vizhome/CensusSalaryData/CensusSalaryData)

# References

2020 Census Quality Indicators: A Report from the American Statistical Association. (2020). Retrieved April 02, 2021, from <https://www.amstat.org/asa/files/pdfs/POL-2020CensusQualityIndicators.pdf>

*Adult Census Income Analysis 82% accuracy score*. (n.d.). Kaggle.com. Retrieved May 2, 2021, from <https://www.kaggle.com/aqsasadaf/adult-census-income-analysis-82-accuracy-score>

Amazon EC2. (1987). Retrieved May 03, 2021, from <https://aws.amazon.com/ec2/?ec2-whats-new.sort-by=item.additionalFields.postDateTime&ec2-whats-new.sort-order=desc>

Amazon SageMaker. (2020). Retrieved May 02, 2021, from <https://aws.amazon.com/sagemaker/>

Becker, B. and Kohavi, R. (1996). UCI Machine Learning Repository [<https://archive.ics.uci.edu/ml/datasets/adult>]. Irvine, CA: University of California, School of

Information and Computer Science

Brownlee, J. (2021, February 17). A Gentle Introduction to XGBoost for Applied Machine Learning. Retrieved April 30, 2021, from <https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/>

Correlation Test Between Two Variables in R. (n.d.). Retrieved April 23, 2021, from <http://www.sthda.com/english/wiki/correlation-test-between-two-variables-in-r>

Current Population Survey 2020 Annual Social and Economic (ASEC) Supplement [PDF]. (2020). Washington, DC: U.S. Census Bureau.

DailySmarty.com (2017). *Difference Between Independent and Dependent Variables in Machine Learning.* Retrieved April 28, 2021, from <https://www.dailysmarty.com/posts/difference-between-independent-and-dependent-variables-in-machine-learning>

Dash, S. (2020, October 21). *Gradient Boosting - A Concise Introduction from Scratch*. ML+. <https://www.machinelearningplus.com/machine-learning/gradient-boosting/#:~:text=The%20Learning%20rate%20and%20n_estimators%20are%20two%20critical>

Employment level of the United States from 1990 to 2020. (2021, January). Retrieved March 26, 2021,

from <https://www.statista.com/statistics/192384/number-of-employees-in-the-us-since-1990/>

Gould, E. (2020, February 20). State of Working America Wages 2019. Retrieved April 02, 2021, from <https://www.epi.org/publication/swa-wages-2019/>

Hall, R. E. (1980). Employment Fluctuations and Wage Rigidity. Retrieved April 01, 2021, from [https://www.brookings.edu/wp-content/uploads/ 1980/01/1980a\_bpea\_hall\_ baily\_](https://www.brookings.edu/wp-content/uploads/1980/01/1980a_bpea_hall_baily_sumers_duesenberry_modigliani_nordhaus_tobin_gordon_kenen_juster_marris.pdf)  sumers\_duesenberry\_modigliani\_nordhaus\_tobin\_ gordon\_kenen\_juster\_marris.pdf

Hoare, J. (n.d.). Gradient Boosting Explained – The Coolest Kid on The Machine Learning Block. Retrieved April 30, 2021, from <https://www.displayr.com/gradient-boosting-the-coolest-kid-on-the-machine-learning-block/>

IBM. (n.d.). *Exploratory Data Analysis.* Retrieved April 28, 2021.

<https://www.ibm.com/cloud/learn/exploratory-data-analysis>

Koehrsen, W. (2018, January 10). *Hyperparameter Tuning the Random Forest in Python*. Medium; Towards Data Science. <https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74>

Kohavi, R. (2019). *UCI Machine Learning Repository: Adult Data Set*. Uci.edu. <https://archive.ics.uci.edu/ml/datasets/Adult>

Liu, J. (2020, January 07). The best-paying job of 2020 pays $267,020—here are the other 24. Retrieved

March 26, 2021, from <https://www.cnbc.com/2020/01/06/the-best-paying-jobs-of-2020-from->

[https://www.mathsisfun.com/data/percentiles.html us-news-and-world-report.html](https://www.mathsisfun.com/data/percentiles.html%20us-news-and-world-report.html)

Lyashenko, V. (n.d.). *Random Forest Regression - The Definitive Guide | cnvrg.io*. Cnvrg.io. <https://cnvrg.io/random-forest-regression/>

Malik, S. (2020, March 23). *XGBoost: A Deep Dive Into Boosting - DZone AI*. Dzone.com. <https://dzone.com/articles/xgboost-a-deep-dive-into-boosting>

Missouri Economic Research and Information Center. (2020). *Cost of Living Data Series*. Cost of Living Data Series | Missouri Economic Research and Information Center. <https://meric.mo.gov/data/cost-living-data-series>.

Moody, J. (2019, September 05). What does RMSE really mean? Retrieved May 02, 2021, from

<https://towardsdatascience.com/what-does-rmse-really-mean-806b65f2e48e>

Muller, F. (2018, January 12). *XGBoost Tree vs. Linear*. STATWORX. <https://www.statworx.com/de/blog/xgboost-tree-vs-linear/>

*North American employee turnover: trends and effects*. (2020, January 29). Imercer. <https://www.imercer.com/articleinsights/North-American-Employee-Turnover-Trends-and-Effects>

Pathak, M. (2019, November 8). *Using XGBoost in Python*. Datacamp.com. <https://www.datacamp.com/community/tutorials/xgboost-in-python>

Random forest approach for regression in r programming. (2020, July 10). Retrieved April 30, 2021, from <https://www.geeksforgeeks.org/random-forest-approach-for-regression-in-r-programming/>

Scikit-learn. (n.d.). Retrieved May 02, 2021, from <https://scikit-learn.org/stable>

Singh, H. (2018, November 03). Understanding Gradient Boosting Machines. Retrieved April 30, 2021, from <https://towardsdatascience.com/understanding-gradient-boosting-machines-9be756fe76ab>

Statistics solutions.(n.d.). *Correlation (Pearson, Kendall, Spearman).* Retrieved April 21,2021 from

<https://www.statisticssolutions.com/correlation-pearson-kendall-spearman/>

Stephanie. (2021, April 27). Lasso regression: Simple definition. Retrieved April 30, 2021, from <https://www.statisticshowto.com/lasso-regression/#:~:text=Lasso%20regression%20is%20a%20type,i.e.%20models%20with%20fewer%20parameters>

Taylor, C. (2018, April 27). *What is the Interquartile Rang Rule?* ThoughtCO.

<https://www.thoughtco.com/what-is-the-interquartile-range-rule-3126244>

The NCES fast Facts tool provides quick answers to many education QUESTIONS (National Center for EDUCATION STATISTICS). (n.d.). Retrieved March 28, 2021, from <https://nces.ed.gov/fastfacts/display.asp?id=372#:~:text=About%203.7%20million%20students%20are,from%20private%20schools%20(source>).

ThinkAutomation. (n.d.) *What is an Algorithm? An ‘in a nutshell’ Explanation.* Retrieved May 3, 2021. https://www.thinkautomation.com/eli5/what-is-an-algorithm-an-in-a-nutshell-explanation/

Twin, A. (2021, April 21). Overfitting. Retrieved May 03, 2021, from <https://www.investopedia.com/terms/o/overfitting.asp#:~:text=Overfitting%20is%20a%20modeling%20error,to%20any%20other%20data%20sets>.

U.S. Census Bureau. (2020). *2020 Annual Social and Economic (ASEC) Supplement.* Retrieved April 5, 2020. https://www2.census.gov/programs-surveys/cps/techdocs/cpsmar20.pdf

What are outliers in the data? (n.d.). Retrieved May 02, 2021, from <https://www.itl.nist.gov/div898/handbook/prc/section1/prc16.htm>

*What Is Amazon SageMaker? - Amazon SageMaker*. (n.d.). Docs.aws.amazon.com; Amazon Web Services. <https://docs.aws.amazon.com/sagemaker/latest/dg/whatis.html>

What is a machine learning model? (n.d.). Retrieved May 02, 2021, from

<https://docs.microsoft.com/en-us/windows/ai/windows-ml/what-is-a-machine-learning-model#:~:text=A%20machine%20learning%20model%20is,and%20learn%20from%20those%20data.&text=Windows%20Machine%20Learning%20uses%20the,ONNX)%20format%20for%20its%20models>.

Wood, T. (2019, May 17). What is the F-score? Retrieved May 02, 2021, from <https://deepai.org/machine-learning-glossary-and-terms/f-score>

*XGBoost Hyperparameters - Amazon SageMaker*. (n.d.). Docs.aws.amazon.com; AWS Web Services. Retrieved May 2, 2021, from <https://docs.aws.amazon.com/sagemaker/latest/dg/xgboost_hyperparameters.html>

Yale University. (n.d.). *Linear Regression.* Retrieved April 15, 2021. http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm

1. Greater discussion of the effect of COVID-19 on census data can be found on page 358 of ASEC Technical Documentation [↑](#footnote-ref-2)