p01 Discovery and Prep

Code

Introduction

Along with data science, I am extremely interested in the field of machine learning. From my basic understanding of what's going on in industry, I am witnessing an intertwining of the skills from both fields in demand. I wanted to choose a dataset that would provide me with the opportunity to learn some statistical modeling along with machine learning. I wanted to find something that could provide me with a challenge, but not too much of one for someone at my skill level, but also give me some versatility in what models I can build from it. I ended up choosing the 1994 Census Income dataset from the UCI Machine Learning Repository.

The typical way this dataset is used is to predict whether an individual's income exceeds 50,000 dollars using the variables within the dataset. We can use statistical modeling techniques like logistic regression along with some machine learning algorithms like neural networks, classification, random forest, support vector machines, and possibly XGBoost.

Data Prep and Discovery

First we load the necessary packages.

Hide

```
#First, we must load the necessary packages.
library(ggplot2)
library(plyr)
library(gridExtra)
library(gmodels)
library(grid)
library(vcd)
library(scales)
library(ggthemes)
library(knitr)
```

Then we must download the data which comes in the form of a test and training set. In my DSProject directory, I created a working directory in which to do this in order to keep my raw data separate for organizational purposes within its own folder named CensusData. For the purposes of this particular assignment, we will go ahead and download the data directly.

Hide

```
#Import the training data.
adult_train <- read.table("https://archive.ics.uci.edu/ml/machine-learning-databases/adu
lt/adult.data", sep = ",", header = FALSE)
adult_test <- read.table("https://archive.ics.uci.edu/ml/machine-learning-databases/adul
t/adult.test", sep = ",", header = FALSE,
                         skip = 1, na.strings = " ?")
```

Let's take a preliminary look at the training data. We note that the number of observations and variables respectively are:

Hide

```
(dim(adult_train))
```

```
[1] 32561    15
```

The column names are such that they're labeled ambiguously as "V1, V2,…".We get the true names from the attributes list available at https://archive.ics.uci.edu/ml/datasets/Census+Income (https://archive.ics.uci.edu/ml/datasets/Census+Income) .

Hide

```
colnames(adult_train) <- c("age", "workclass", "fnlwgt", "education", "education_num", 
"marital_status", "occupation", "relationship",
                           "race", "sex", "capital_gain", "capital_loss", "hours_per_wee
k", "native_country", "income")
```

Now we will take a look at the first few observations of the dataset and its structure as well.

Hide

```
head(adult_train)
```

	…	workclass	fnlwgt	education	education_num	marital_status	occupation
	<int>	<fctr>	<int>	<fctr>	<int>	<fctr>	<fctr>
1	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical
2	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-manageria
3	38	Private	215646	HS-grad	9	Divorced	Handlers-cleane
4	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleane
5	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty
6	37	Private	284582	Masters	14	Married-civ-spouse	Exec-manageria

6 rows | 1-8 of 15 columns

Hide

```
str(adult_train)
```

```
'data.frame':	32561 obs. of  15 variables:
 $ age           : int  39 50 38 53 28 37 49 52 31 42 ...
 $ workclass     : Factor w/ 9 levels " ?"," Federal-gov",..: 8 7 5 5 5 5 5 7 5 5 ...
 $ fnlwgt        : int  77516 83311 215646 234721 338409 284582 160187 209642 45781 1594
49 ...
 $ education     : Factor w/ 16 levels " 10th"," 11th",..: 10 10 12 2 10 13 7 12 13 10
...
 $ education_num : int  13 13 9 7 13 14 5 9 14 13 ...
 $ marital_status: Factor w/ 7 levels " Divorced"," Married-AF-spouse",..: 5 3 1 3 3 3 4
3 5 3 ...
 $ occupation    : Factor w/ 15 levels " ?"," Adm-clerical",..: 2 5 7 7 11 5 9 5 11 5
...
 $ relationship  : Factor w/ 6 levels " Husband"," Not-in-family",..: 2 1 2 1 6 6 2 1 2
1 ...
 $ race          : Factor w/ 5 levels " Amer-Indian-Eskimo",..: 5 5 5 3 3 5 3 5 5 5 ...
 $ sex           : Factor w/ 2 levels " Female"," Male": 2 2 2 2 1 1 1 2 1 2 ...
 $ capital_gain  : int  2174 0 0 0 0 0 0 0 14084 5178 ...
 $ capital_loss  : int  0 0 0 0 0 0 0 0 0 0 ...
 $ hours_per_week: int  40 13 40 40 40 40 16 45 50 40 ...
 $ native_country: Factor w/ 42 levels " ?"," Cambodia",..: 40 40 40 40 6 40 24 40 40 40
...
 $ income        : Factor w/ 2 levels " <=50K"," >50K": 1 1 1 1 1 1 1 2 2 2 ...
```

Variables

We see that the variables age, fnlwgt, education_num, capital_gain, capital_loss, and hours_per_week are of type integer. The other variables are factors with differing levels. To see what levels of each factor we have, we provide a function called get_factor_levels() which takes a dataframe as an argument, identifies the factor variables, and outputs the levels of each factor variable it finds.

Hide

```
get_factor_levels <- function(mydata){
  col_names <- names(mydata)
  for (i in 1:length(col_names)){
    if (is.factor(mydata[, col_names[i]])){
      message(noquote(paste("Covariate ", "*",
                            col_names[i], "*",
                            " with factor levels: ",
                            sep = "")))
      print(levels(mydata[, col_names[i]]))
    }
  }
}

get_factor_levels(adult_train)
```

```
Covariate *workclass* with factor levels:
```

```
[1] " ?"                " Federal-gov"      " Local-gov"        " Never-worked"     " Pr
ivate"          " Self-emp-inc"
[7] " Self-emp-not-inc" " State-gov"        " Without-pay"
```

```
Covariate *education* with factor levels:
```

```
 [1] " 10th"         " 11th"         " 12th"         " 1st-4th"      " 5th-6th"      " 7
th-8th"      " 9th"
 [8] " Assoc-acdm"   " Assoc-voc"    " Bachelors"    " Doctorate"    " HS-grad"      " M
asters"      " Preschool"
[15] " Prof-school"  " Some-college"
```

```
Covariate *marital_status* with factor levels:
```

```
[1] " Divorced"              " Married-AF-spouse"     " Married-civ-spouse"    " Married
-spouse-absent" " Never-married"
[6] " Separated"             " Widowed"
```

```
Covariate *occupation* with factor levels:
```

```
 [1] " ?"                 " Adm-clerical"      " Armed-Forces"      " Craft-repair"
" Exec-managerial"
 [6] " Farming-fishing"   " Handlers-cleaners" " Machine-op-inspct" " Other-service"
" Priv-house-serv"
[11] " Prof-specialty"    " Protective-serv"   " Sales"             " Tech-support"
" Transport-moving"
```

```
Covariate *relationship* with factor levels:
```

```
[1] " Husband"        " Not-in-family"  " Other-relative" " Own-child"      " Unmarried"
" Wife"
```

```
Covariate *race* with factor levels:
```

```
[1] " Amer-Indian-Eskimo" " Asian-Pac-Islander" " Black"              " Other"
" White"
```

```
Covariate *sex* with factor levels:
```

```
[1] " Female" " Male"
```

```
Covariate *native_country* with factor levels:
```

```
 [1] " ?"                            " Cambodia"                     " Canada"                      
" China"                        
 [5] " Columbia"                     " Cuba"                         " Dominican-Republic"          
" Ecuador"                      
 [9] " El-Salvador"                  " England"                      " France"                      
" Germany"                      
[13] " Greece"                       " Guatemala"                    " Haiti"                       
" Holand-Netherlands"           
[17] " Honduras"                     " Hong"                         " Hungary"                     
" India"                        
[21] " Iran"                         " Ireland"                      " Italy"                       
" Jamaica"                      
[25] " Japan"                        " Laos"                         " Mexico"                      
" Nicaragua"                    
[29] " Outlying-US(Guam-USVI-etc)"   " Peru"                         " Philippines"                 
" Poland"                       
[33] " Portugal"                     " Puerto-Rico"                  " Scotland"                    
" South"                        
[37] " Taiwan"                       " Thailand"                     " Trinadad&Tobago"             
" United-States"                
[41] " Vietnam"                      " Yugoslavia"                  
```

```
Covariate *income* with factor levels:
```

```
[1] " <=50K" " >50K" 
```

The output above indicates that some of the factor variables have a level denoted by " ?“. Those are missing values according to the documentation provided for the census data. We must get rid of the missing values before we can proceed with any exploratory and predictive analysis. We read in the data again, but with the additional specification na.strings =” ?".

Hide

```
adult_train <- read.table("https://archive.ics.uci.edu/ml/machine-learning-databases/adu
lt/adult.data", sep = ",", header = FALSE, na.strings = " ?")

#Don't forget to rename the columns.
colnames(adult_train) <- c("age", "workclass", "fnlwgt", "education", "education_num",
"marital_status", "occupation", "relationship",
                          "race", "sex", "capital_gain", "capital_loss", "hours_per_wee
k", "native_country", "income")
```

#Since those previous ?’s are now NA’s, we may sweep them out with na.omit().

Hide

```
adult_train <- na.omit(adult_train)
```

We’ll also enumerate the rows of the data.

Hide

```
row.names(adult_train) <- 1:nrow(adult_train)
```

From a boxplot and summary of the variable hours_per_week, we see that the mean number of working hours per week is 41, and at least 50% of the people taking part of the survey work between 40 and 45 hours per week.

Hide

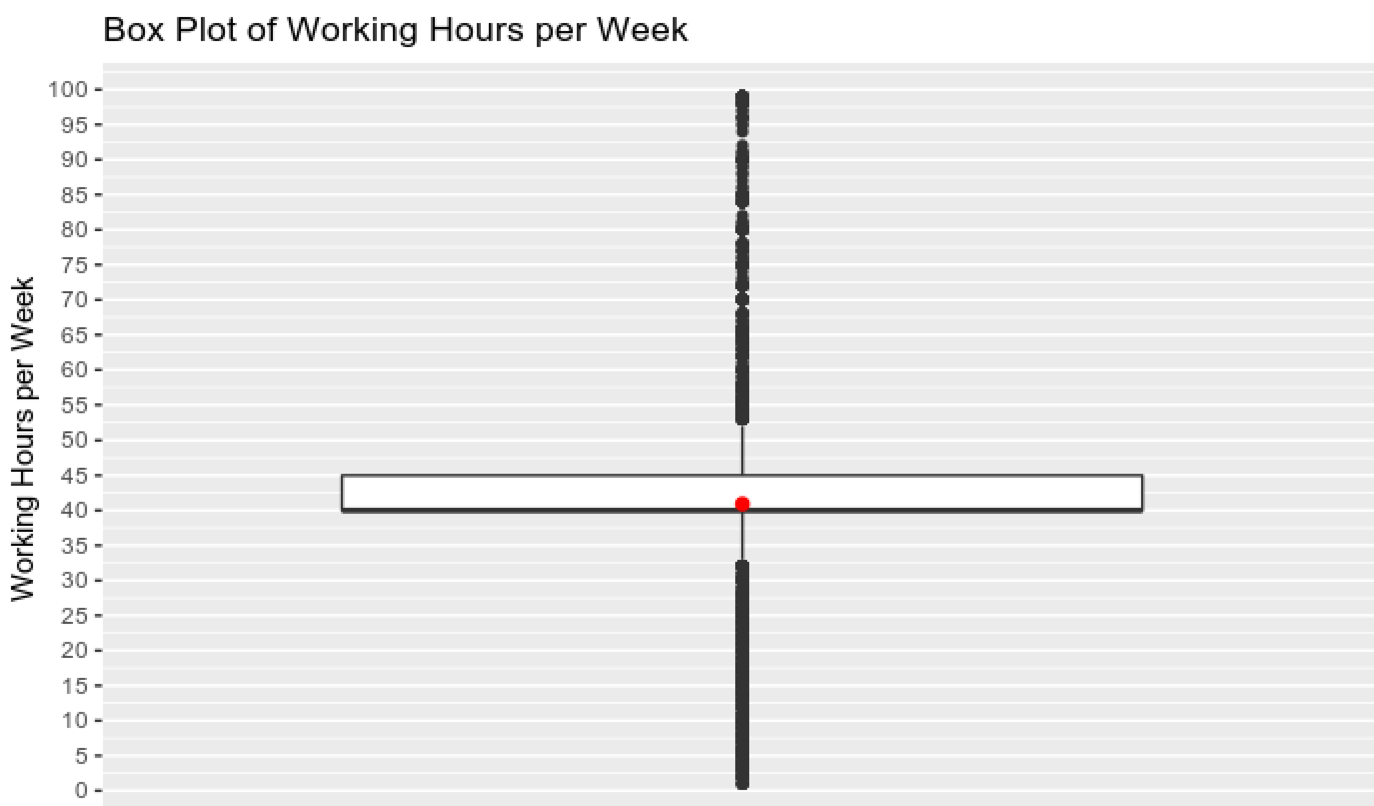
```
summary(adult_train$hours_per_week)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   1.00   40.00   40.00   40.93   45.00   99.00 
```

The boxplot also indicates the many outliers:

Hide

```
ggplot(aes(x = factor(0), y = hours_per_week), data = adult_train) +
  geom_boxplot() +
  stat_summary(fun.y = mean, geom = "point", shape = 19, color = "red", cex = 2) +
  scale_x_discrete(breaks = NULL) +
  scale_y_continuous(breaks = seq(0, 100, 5)) +
  xlab(label = "") +
  ylab(label = "Working Hours per Week") +
  ggtitle("Box Plot of Working Hours per Week")
```



We will group the working hours into 5 categories. We will also create a new factor variable called hours_worked corresponding to these 5 categories.

Hide

```
adult_train$hours_worked[adult_train$hours_per_week < 40] <- " less_than_40"
adult_train$hours_worked[adult_train$hours_per_week >= 40 & adult_train$hours_per_week <
= 45] <- " between_40_and_45"
adult_train$hours_worked[adult_train$hours_per_week > 45 & adult_train$hours_per_week <=
60] <- " between_45_and_60"
adult_train$hours_worked[adult_train$hours_per_week > 60 & adult_train$hours_per_week <=
80] <- " between_60_and_80"
adult_train$hours_worked[adult_train$hours_per_week > 80] <- " more_than_80"

adult_train$hours_worked <- factor(adult_train$hours_worked, ordered = FALSE, levels = c
(" less_than_40", " between_40_and_45", " between_45_and_60",
                                                                          " between_6
0_and_80", " more_than_80"))
```

We can now see how many people belong to each category of the factor variable hours_worked.

Hide

```
summary(adult_train$hours_worked)
```

```
      less_than_40  between_40_and_45  between_45_and_60  between_60_and_80       more_t
han_80
              6714              16606               5790                857
195
```

It's been already mentioned that the majority of people work between 40 and 45 hours per week, but there's also a considerable amount of people working less than 40 and between 45 and 60 hours per week.

Hide

```
for (i in 1:length(summary(adult_train$hours_worked))){
  print(round(100 * summary(adult_train$hours_worked)[i] / sum(!is.na(adult_train$hours_
worked)), 2))
}
```

```
 less_than_40
        22.26
 between_40_and_45
             55.06
 between_45_and_60
              19.2
 between_60_and_80
              2.84
 more_than_80
         0.65
```

The factor variable native_country has 41 levels. When building a predictive model with native_country as a covariate, it will give 41 degrees of freedom and unnecessarily complicate the analysis. We mus coarsen the data using global regions instead.

Hide

```
levels(adult_train$native_country)
```

```
 [1] " Cambodia"                    " Canada"                    " China"
" Columbia"
 [5] " Cuba"                        " Dominican-Republic"        " Ecuador"
" El-Salvador"
 [9] " England"                     " France"                    " Germany"
" Greece"
[13] " Guatemala"                   " Haiti"                     " Holand-Netherlands"
" Honduras"
[17] " Hong"                        " Hungary"                   " India"
" Iran"
[21] " Ireland"                     " Italy"                     " Jamaica"
" Japan"
[25] " Laos"                        " Mexico"                    " Nicaragua"
" Outlying-US(Guam-USVI-etc)"
[29] " Peru"                        " Philippines"               " Poland"
" Portugal"
[33] " Puerto-Rico"                 " Scotland"                  " South"
" Taiwan"
[37] " Thailand"                    " Trinadad&Tobago"           " United-States"
" Vietnam"
[41] " Yugoslavia"
```

First we'll define the regions:

Hide

```
Asia_East <- c("Cambodia", "China", "Hong", "Laos", "Thailand", "Japan", "Taiwan", "Viet
nam")
Asia_Central <- c("India", "Iran")
Central_America <- c("Cuba", "Guatemala", "Jamaica", "Nicaragua", "Puerto-Rico", "Domini
can-Republic", "El-Salvador", "Haiti",
                     "Honduras", "Mexico", "Trinidad&Tobago")
South_America <- c("Ecuador", "Peru", "Columbia")
Europe_West <- c("England", "Germany", "Holand-Netherlands", "Ireland", "France", "Greec
e", "Italy", "Portugal", "Scotland")
Europe_East <- c("Poland", "Yugoslavia", "Hungary")
```

Then we'll modify the dataframe by adding an additional column named native_region.

Hide

```
adult_train <- mutate(adult_train, native_region = ifelse(native_country %in% Asia_East,
"East-Asia",
                                                        ifelse(native_country %in% Asia_
Central, "Central-Asia",
                                                        ifelse(native_country %in% Centr
al_America, "Central-America",
                                                        ifelse(native_country %in% South
_America, "South-America",
                                                        ifelse(native_country %in% Europ
e_West, "Europe-West",
                                                        ifelse(native_country %in% Europ
e_East, "Europe-East",
                                                        ifelse(native_country == "United
-States", "United-States", "Outlying-US")))))

)))
```

Finally, we'll transform the new variable, native_region, into a factor.

Hide

```
adult_train$native_region <- factor(adult_train$native_region, ordered = FALSE)
```

The summary below tells us that at least 50% of the variables capital_gain and capital_loss are zeros.

Hide

```
summary(adult_train$capital_gain)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
      0       0       0    1092       0   99999 
```

Hide

```
summary(adult_train$capital_loss)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   0.00    0.00    0.00   88.37    0.00 4356.00 
```

The mean values of capital_gain and capital_loss with zero values included are, respectively:

Hide

```
mean_gain <- mean(adult_train$capital_gain)
mean_loss <- mean(adult_train$capital_loss)
kable(data.frame(Mean_Capital_Gain = mean_gain, Mean_Capital_Loss = mean_loss), caption
= "Mean Capital with Zero Values Included")
```

Mean_Capital_Gain	Mean_Capital_Loss
1092.008	88.37249

We also give the mean capital gain and loss in the case where all zero values are removed:

Hide

```
mean_gain <- mean(subset(adult_train$capital_gain, adult_train$capital_gain > 0))
mean_loss <- mean(subset(adult_train$capital_loss, adult_train$capital_loss > 0))
kable(data.frame(Mean_Capital_Gain = mean_gain, Mean_Capital_Loss = mean_loss), caption
 = "Mean Capital Only for Nonzero Values")
```

Mean_Capital_Gain	Mean_Capital_Loss
12977.6	1867.898

Hide

```
NA
```

We display the summary of the nonzero values of capital loss and capital gain as well as their respective interquartile ranges.

Hide

```
iqr_gain <- IQR(subset(adult_train$capital_gain, adult_train$capital_gain > 0))
iqr_loss <- IQR(subset(adult_train$capital_loss, adult_train$capital_loss > 0))
quantile_gain <- quantile(x = subset(adult_train$capital_gain, adult_train$capital_gain
 > 0), probs = seq(0, 1, 0.25))
quantile_loss <- quantile(x = subset(adult_train$capital_loss, adult_train$capital_loss
 > 0), probs = seq(0, 1, 0.25))
kable(x = data.frame(Capital_Gain = quantile_gain, Capital_Loss = quantile_loss), captio
n = "Quantile of the Nonzero Capital")
```

	Capital_Gain	Capital_Loss
0%	114	155
25%	3464	1672
50%	7298	1887
75%	14084	1977
100%	99999	4356

Hide

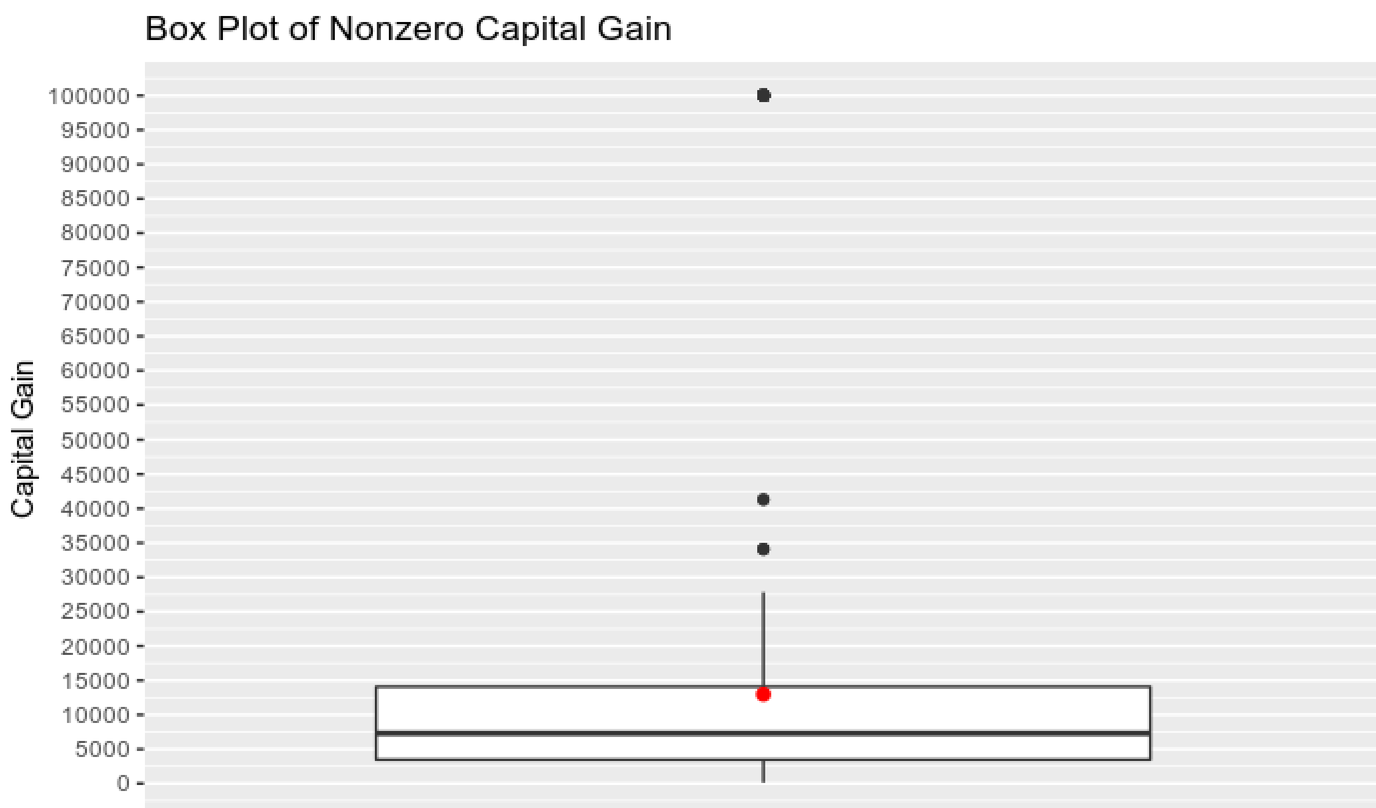
```
kable(x = data.frame(IQR_Capital_Gain = iqr_gain, IQR_Capital_Loss = iqr_loss), caption
= "IQR of the Nonzero Capital")
```

IQR_Capital_Gain	IQR_Capital_Loss
10620	305

We notice that the IQR of the nonzero capital gain is much larger than that of the capital loss. We display a boxplot of the nonzero capital gain.

Hide

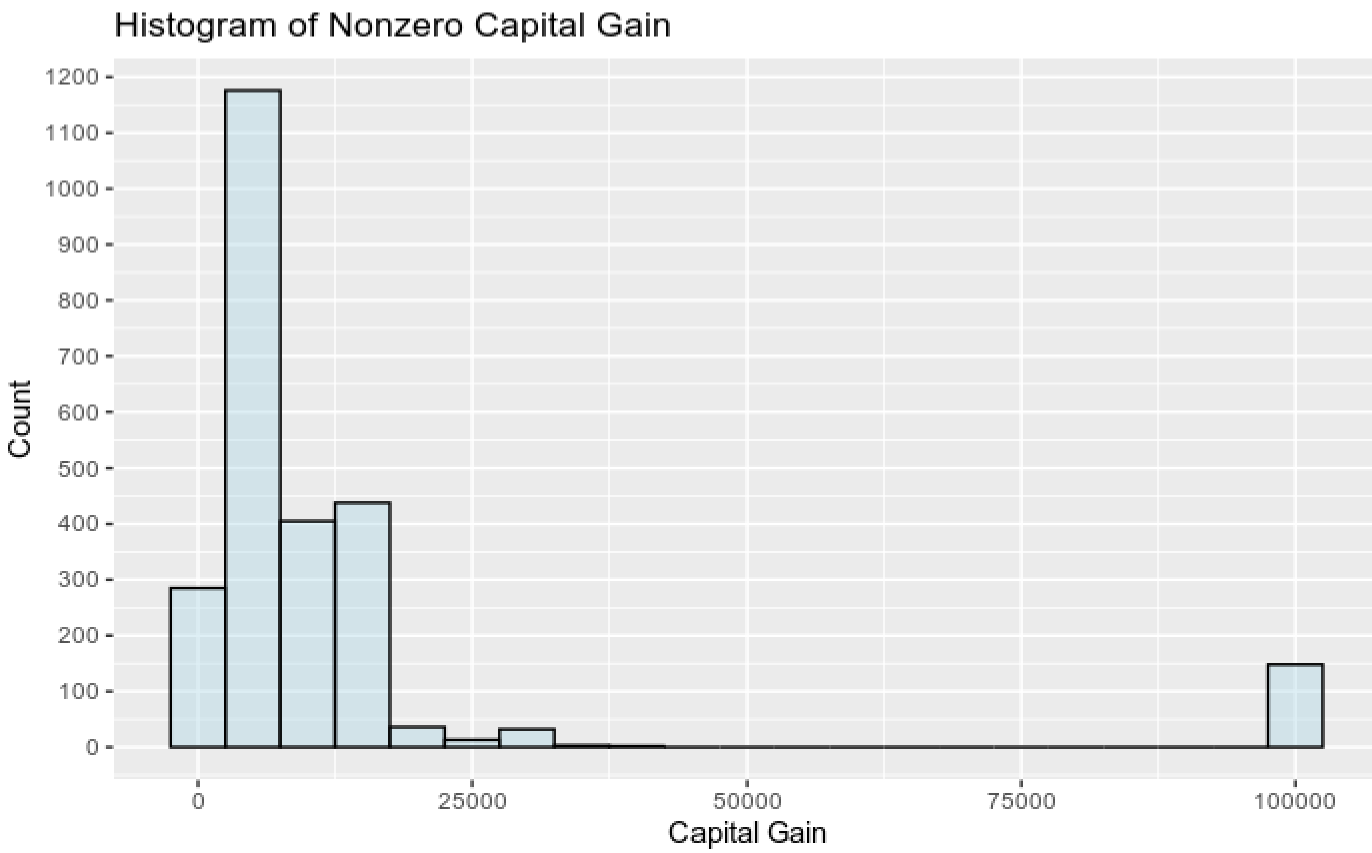
```
ggplot(aes(x = factor(0), y = capital_gain),
       data = subset(adult_train, adult_train$capital_gain > 0)) +
  geom_boxplot() +
  stat_summary(fun.y = mean,
               geom = "point",
               shape = 19,
               color = "red",
               cex = 2) +
  scale_x_discrete(breaks = NULL) +
  scale_y_continuous(breaks = seq(0, 100000, 5000)) +
  ylab("Capital Gain") +
  xlab("") +
  ggtitle("Box Plot of Nonzero Capital Gain")
```



From the boxplot, we see that the bulk of the data is between 3,000 and 15,000 dollars with a few outliers. Next, we'll show a histogram of the nonzero capital gain:

Hide

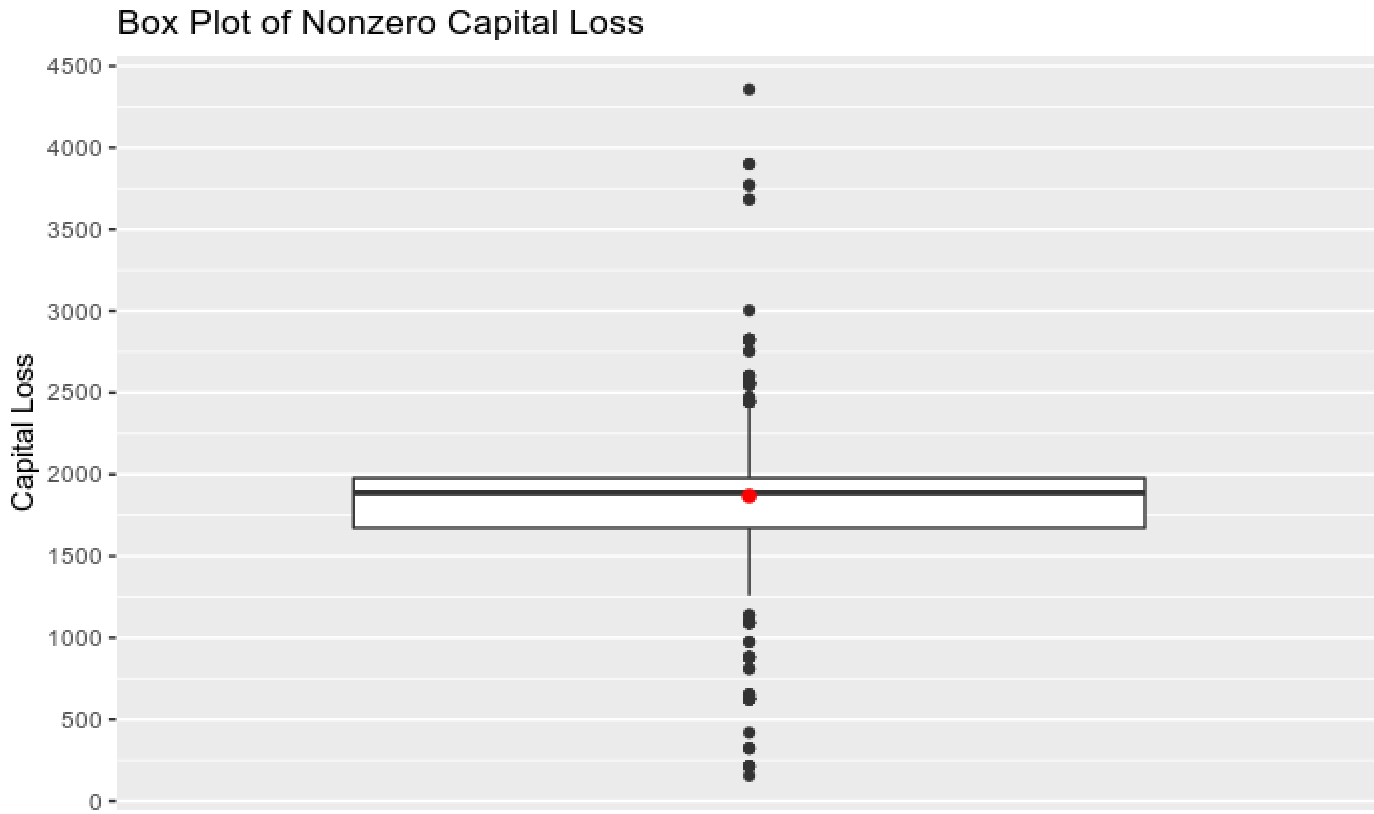
```
df <- adult_train[adult_train$capital_gain > 0,]
ggplot(data = df, aes(x = df$capital_gain)) +
  geom_histogram(binwidth = 5000,
                 color = "black",
                 fill = "lightblue",
                 alpha = 0.4) +
  scale_y_continuous(breaks = seq(0, 4000, 100)) +
  labs(x = "Capital Gain", y = "Count") +
  ggtitle("Histogram of Nonzero Capital Gain")
```



The histogram confirms what we've observed. The majority of people with positive capital gain have a capital gain between 0 and 25,000 dollars. The largest number of people with positive capital gain are those with about 5,000 dollars. Below, we display a box plot of the nonzero capital loss values.

Hide

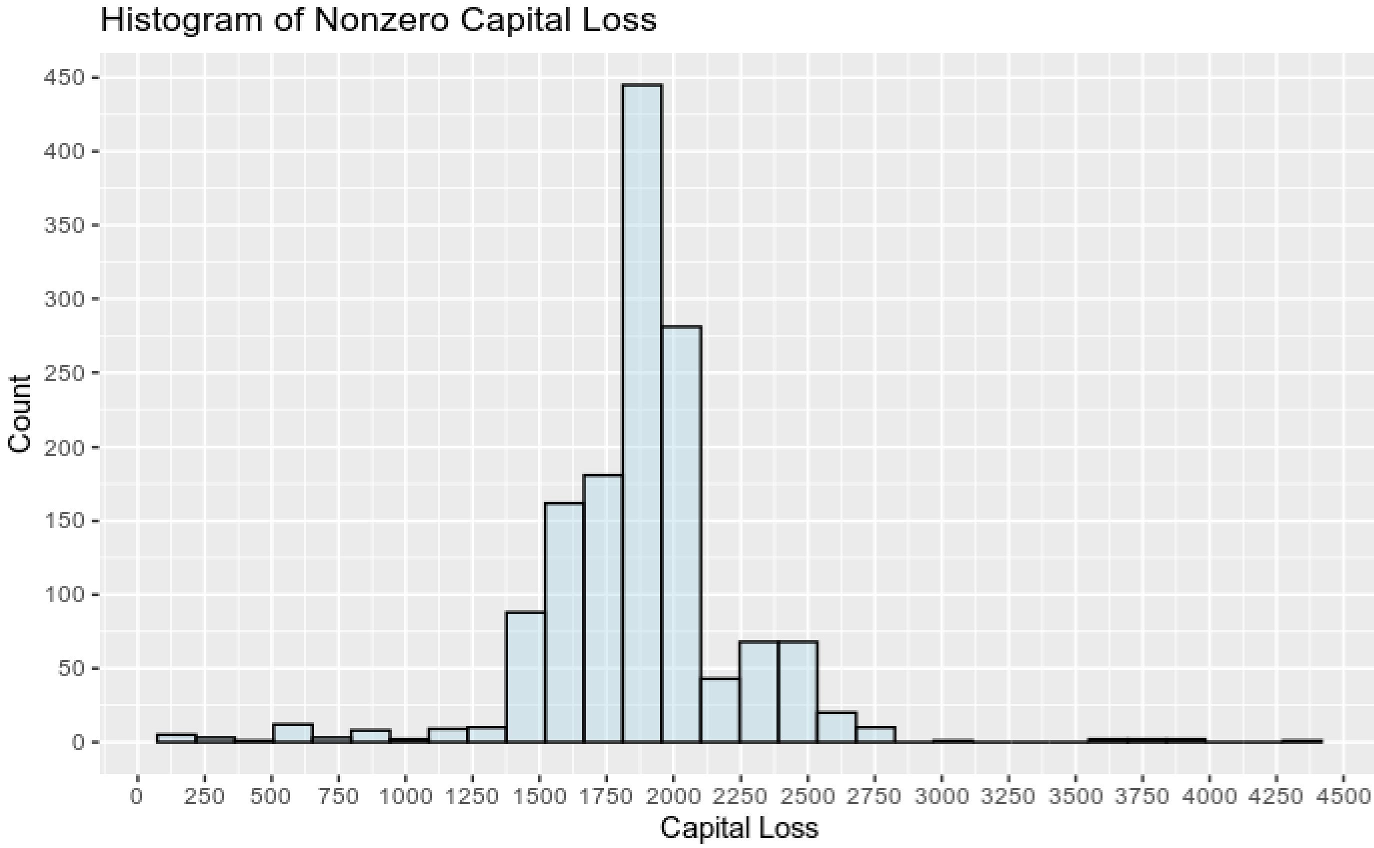
```
ggplot(aes(x = factor(0), y = capital_loss), data = subset(adult_train, adult_train$capi
tal_loss > 0)) +
  geom_boxplot() +
  stat_summary(fun.y = mean, geom = "point", shape = 19, color = "red", cex = 2) +
  scale_x_discrete(breaks = NULL) +
  scale_y_continuous(breaks = seq(0, 5000, 500)) +
  ylab("Capital Loss") +
  xlab("") +
  ggtitle("Box Plot of Nonzero Capital Loss")
```



A histogram of the nonzero capital loss:

Hide

```
df <- adult_train[adult_train$capital_loss > 0,]
ggplot(data = df, aes(x = df$capital_loss)) +
  geom_histogram(color = "black", fill = "lightblue", alpha = 0.4) +
  scale_x_continuous(breaks = seq(0, 5000, 250)) +
  scale_y_continuous(breaks = seq(0, 450, 50)) +
  labs(x = "Capital Loss", y = "Count") +
  ggtitle("Histogram of Nonzero Capital Loss")
```



The box plot tells us that most values are between 1,700 and 2,000 dollars and there are many outliers. The largest number of people have a capital loss of about 1,875 dollars.

Based on these results, we will group the values of the variables capital_loss, and capital_gain into categories and we will create two new factor variables called cap_gain and cap_loss.

We will mark all values of capital_gain which are less than the first quartile of the nonzero capital gain as “Low”, all values that are between the first and third quartile as “Medium”, and all values greater than or equal to the third quartile are marked “High”.

We mark alll values of capital_loss which are less than the first quartile of the nonzero capital gain as “Low”, all values that are between the first and third quartile as “Medium”, and all values greater than or equal to the third quartile are marked “High”.

Hide

```
adult_train <- mutate(adult_train, cap_gain = ifelse(adult_train$capital_gain < 3464, "L
ow",
                                           ifelse(adult_train$capital_gain >= 3464 & ad
ult_train$capital_gain <= 14080, "Medium", "High")))
adult_train$cap_gain <- factor(adult_train$cap_gain, ordered = TRUE, levels = c("Low",
"Medium", "High"))

adult_train <- mutate(adult_train, cap_loss = ifelse(adult_train$capital_loss< 1672, "Lo
w",
                                           ifelse(adult_train$capital_loss >= 1672 & ad
ult_train$capital_loss <= 1977, "Medium","High")))
adult_train$cap_loss <- factor(adult_train$cap_loss, ordered = TRUE, levels = c("Low",
"Medium", "High"))
```

We notice that there is one unused factor level in the variable workclass, the level “Never-worked”.

Hide

```
summary(adult_train$workclass)
```

```
      Federal-gov         Local-gov      Never-worked           Private      Self-emp-in
c  Self-emp-not-inc         State-gov
              943              2067                 0             22286              107
4              2499              1279
      Without-pay
               14
```

We will remove the unused factor level Never-worked from the categorical variable workclass.

Hide

```
adult_train$workclass <- droplevels(adult_train$workclass)
levels(adult_train$workclass)
```

```
[1] " Federal-gov"      " Local-gov"        " Private"          " Self-emp-inc"     " Se
lf-emp-not-inc" " State-gov"
[7] " Without-pay"
```

The census data comes with a separate test data set, which we use to test out-of-sample accuracy of the constructed predictive models. We repeat the same steps as in the transformation of the training dataframe adult_train.

Hide

```
adult_test <- read.table("https://archive.ics.uci.edu/ml/machine-learning-databases/adul
t/adult.test", sep = ",", header = FALSE, skip = 1, na.strings = " ?")
colnames(adult_test) <- c("age", "workclass", "fnlwgt", "education", "education_num", "m
arital_status", "occupation", "relationship",
                          "race", "sex", "capital_gain", "capital_loss", "hours_per_wee
k", "native_country", "income")
```

Cleaning missing values from the test data.

Hide

```
adult_test <- na.omit(adult_test)
row.names(adult_test) <- 1:nrow(adult_test)
```

Let’s take a look at what we’re working with.

Hide

```
head(adult_test)
```

...	workclass	fnlwgt	education	education_num	marital_status	occupation
<int>	<fctr>	<int>	<fctr>	<int>	<fctr>	<fctr>

…	workclass	fnlwgt	education	education_num	marital_status	occupation
<int>	<fctr>	<int>	<fctr>	<int>	<fctr>	<fctr>
1 25	Private	226802	11th	7	Never-married	Machine-op
2 38	Private	89814	HS-grad	9	Married-civ-spouse	Farming-fish
3 28	Local-gov	336951	Assoc-acdm	12	Married-civ-spouse	Protective-s
4 44	Private	160323	Some-college	10	Married-civ-spouse	Machine-op
5 34	Private	198693	10th	6	Never-married	Other-servic
6 63	Self-emp-not-inc	104626	Prof-school	15	Married-civ-spouse	Prof-special

6 rows | 1-8 of 15 columns

From the display of the first 5 observations of the data, we notice that the names of the levels of the factor variable income differ from the respective names in the training data adult_train by the symbol ".". We remove the "." from the names of the factor levels of "income" in the test data.

Hide

```
levels(adult_test$income)[1] <- "<=50K"
levels(adult_test$income)[2] <- ">50K"
levels(adult_test$income)
```

```
[1] "<=50K" ">50K"
```

Just like the training data we create a new variable called hours_worked.

Hide

```
adult_test$hours_worked[adult_test$hours_per_week < 40] <- "less_than_40"
adult_test$hours_worked[adult_test$hours_per_week >= 40 & adult_test$hours_per_week <= 4
5] <- "between_40_and_45"
adult_test$hours_worked[adult_test$hours_per_week > 45 & adult_test$hours_per_week <= 60
] <- "between_45_and_60"
adult_test$hours_worked[adult_test$hours_per_week > 60 & adult_test$hours_per_week <= 80
] <- "between_60_and_80"
adult_test$hours_worked[adult_test$hours_per_week > 80] <- "more_than_80"

adult_test$hour_w <- factor(adult_test$hours_worked, ordered = FALSE,
                            levels = c("less_than_40", "between_40_and_45", "between_45_
and_60", "between_60_and_80", "more_than_80"))
```

We also have to create the variable native_region.

Hide

```
adult_test <- mutate(adult_test, native_region = ifelse(native_country %in% Asia_East, 
"East-Asia",
                                                  ifelse(native_country %in% Asia_Centra
l, "Central-Asia",
                                                  ifelse(native_country %in% Central_Amer
ica, "Central-America",
                                                  ifelse(native_country %in% South_Americ
a, "South-America",
                                                  ifelse(native_country %in% Europe_West,
"Europe-West",
                                                  ifelse(native_country %in% Europe_East,
"Europe-East",
                                                  ifelse(native_country == "United-State
s", "United-States", "Outlying-US"))))))))
adult_test$native_region <- factor(adult_test$native_region, ordered = FALSE)
```

Create the variables cap_gain and cap_loss.

Hide

```
adult_test <- mutate(adult_test, cap_gain = ifelse(adult_test$capital_gain < 3464, "Low"
,
                                             ifelse(adult_test$capital_gain >= 3464 & adu
lt_test$capital_gain <= 14080, "Medium", "High")))
adult_test$cap_gain <- factor(adult_test$cap_gain, ordered = FALSE, levels = c("Low", "M
edium", "High"))

adult_test <- mutate(adult_test, cap_loss = ifelse(adult_test$capital_loss < 1672, "Low"
,
                                             ifelse(adult_test$capital_loss >= 1672 & adu
lt_test$capital_loss <= 1977, "Medium", "High")))
adult_test$cap_loss <- factor(adult_test$cap_loss, ordered = FALSE, levels = c("Low", "M
edium", "High"))
```

We drop the unused level Never-worked from the factor variable workclass.

Hide

```
adult_test$workclass <- droplevels(adult_test$workclass)
```

Research Questions and Other Implications

How would a company that makes high-end products identify customers to target for their products, or potentially customer rich markets? Is using census data an effective way to accomplish these goals? This is a sort of market research question that could be answered in a project like this. Perhaps I may discover other interesting relationships that may be ascertained from the data.