Predicting Whether An Average American Makes Over $50,000

By: Manisha Pednekar, Mihir Parikh

Introduction

Studies have been conducted to assess the determinants of individuals' income levels and to basically predict if an individual makes over $50,000 a year based on a census data. The dataset used for the study was extracted from the UCI website *https://archive.ics.uci.edu/ml/datasets/Census+Income* . An example of some of the key variables in the dataset include workclass, education, occupation, hours per week and native country. This dimension reduction exercise focuses on the education aspect of a dataset that contains different categories of 14 variables thought to determine income. There is a solid probability that some of these tests might be repetitive in deciding an individual's income. This undertaking will study if modest number of variables can clarify the greater part of the inconstancy in these factors. The question is inspected with two different approaches – traditional statistical modeling using logistic regression (due to the binary nature of the variable we are predicting) and machine learning using Random Forest Classifiers.

Study Objective

The aim of this project is to come up with principal components that help fit a regression model that can ultimately be used for prediction. The point of this undertaking is to clarify the fluctuation in autonomous factors or variables that determine an individual's Income. In addition, a machine learning approach will also be used to make predictions.

Dataset Description and Cleanup

Variable	Data Type	# Levels
income	Factor {>50K, <=50K}	2
age	Continuous	-
fnlwgt	Continuous	7
education	Categorical Factor {"10th","11th"...}	16
education-num	Continuous	-
marital-status	Categorical Factor {"Divorced","Married-AF-spouse"...}	7
occupation	Categorical Factor {"Adm-clerical"...}	14
relationship	Categorical Factor {"Husband","Not-in-family"...}	6
race	Categorical Factor {"Amer-Indian-Eskimo"...}	5
sex	Categorical Factor {"Female","Male"...}	2
capital-gain	Continuous	-
capital-loss	Continuous	-
hours-per-week	Continuous	-
native-country	Categorical Factor {"Cambodia","Canada"...}	41

Summary of Variables in the dataset

The dataset has 48,842 observations and has 14 variables. The main objective of the data cleanup was to handle NA values, remove unnecessary variables, convert some continuous variables with large ranges into categorical variables, and narrow down levels of categorical factors.

NAs:

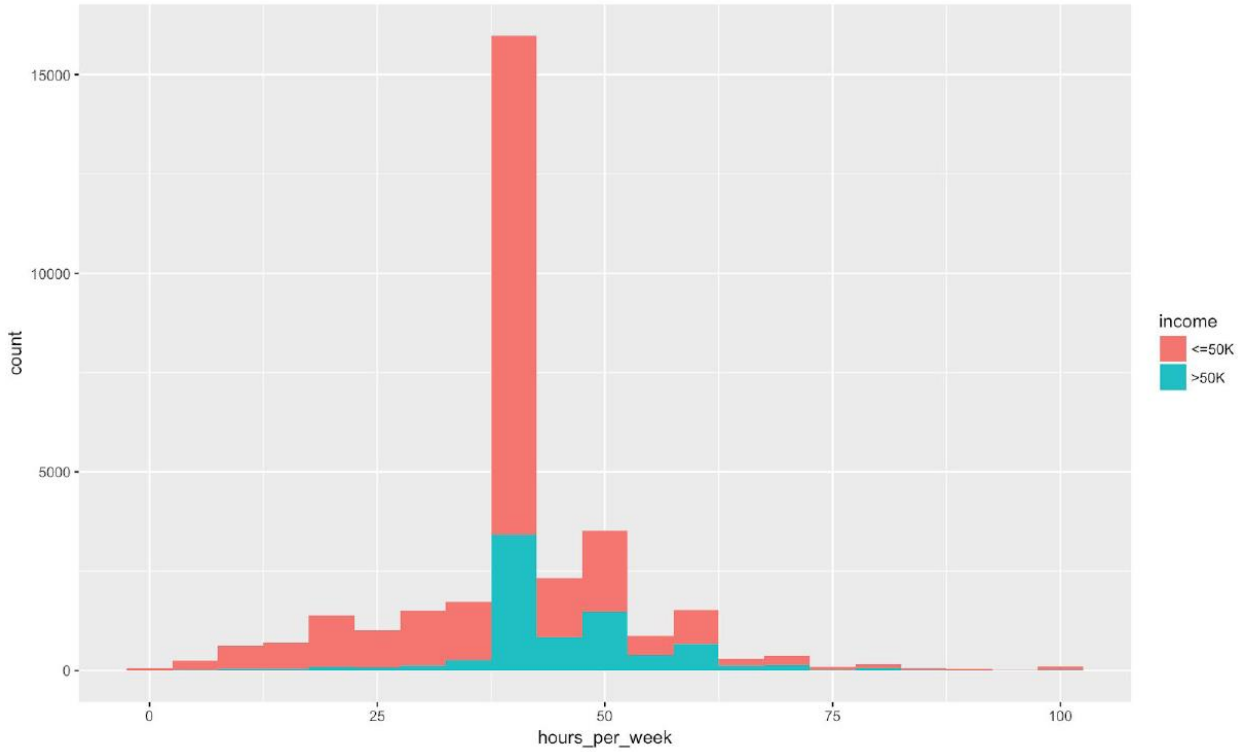
Unnecessary variables were removed from the dataset. Fnlwgt contained just one value which was 1 in all the records. Thus, its presence had no potential impact on the model (and prediction) and was hence removed. Next to get a sense of the number of NAs in the dataset a summary table was utilized to pinpoint counts of NAs for each column. Workclass, occupation, and native country were the three variables that contained NA values with respective counts of 1836, 1843, and 583 (which are relatively low). The percentage of rows with NA values was calculated out to be just 7.36%, thus those rows were removed from the data because they did not represent a significant quantity of data.

Workclass:

Workclass was analyzed for unnecessary levels that could either be dropped or combined. Never-worked was a level that was dropped because it no longer was used in the cleaned dataset. Workclass also had numerous levels that could be combined for ease of modeling. All government related levels were combined into Government and all self-employed levels were combined into Self-employed.

Hours per week:

A histogram was used to understand the distribution of Hours per week relative to income to see if certain hours can be grouped into categories.



By looking at this distribution and the quartile summary the hours were recoded into categorical variables.

Hour lower range	Hour upper range	Hour category	Counts
0	30	part_time	5068
30	37	fringe_fulltime	4276
37	45	regular_fulltime	25605
45	60	overtime	8684
60	-	extreme_overtime	1589

Native country:

Just looking at a summary of Native country it is overwhelming the number of countries that are included. In addition, counts are so imbalanced that it would be difficult to country alone as a predictor. For example, the US has 41292 records while Laos has 21 records. Looking at the world in a more regional manner makes it easier to use this country data for modeling.

Global region	Count
Canada	163
Central Caribbean America	1806
Central Subcontinent Asia	203
Country labeled as South (Not sure what this is?)	101
East Asia	727
Eastern Europe	122
South America	170
United States	41314
Western Europe	616

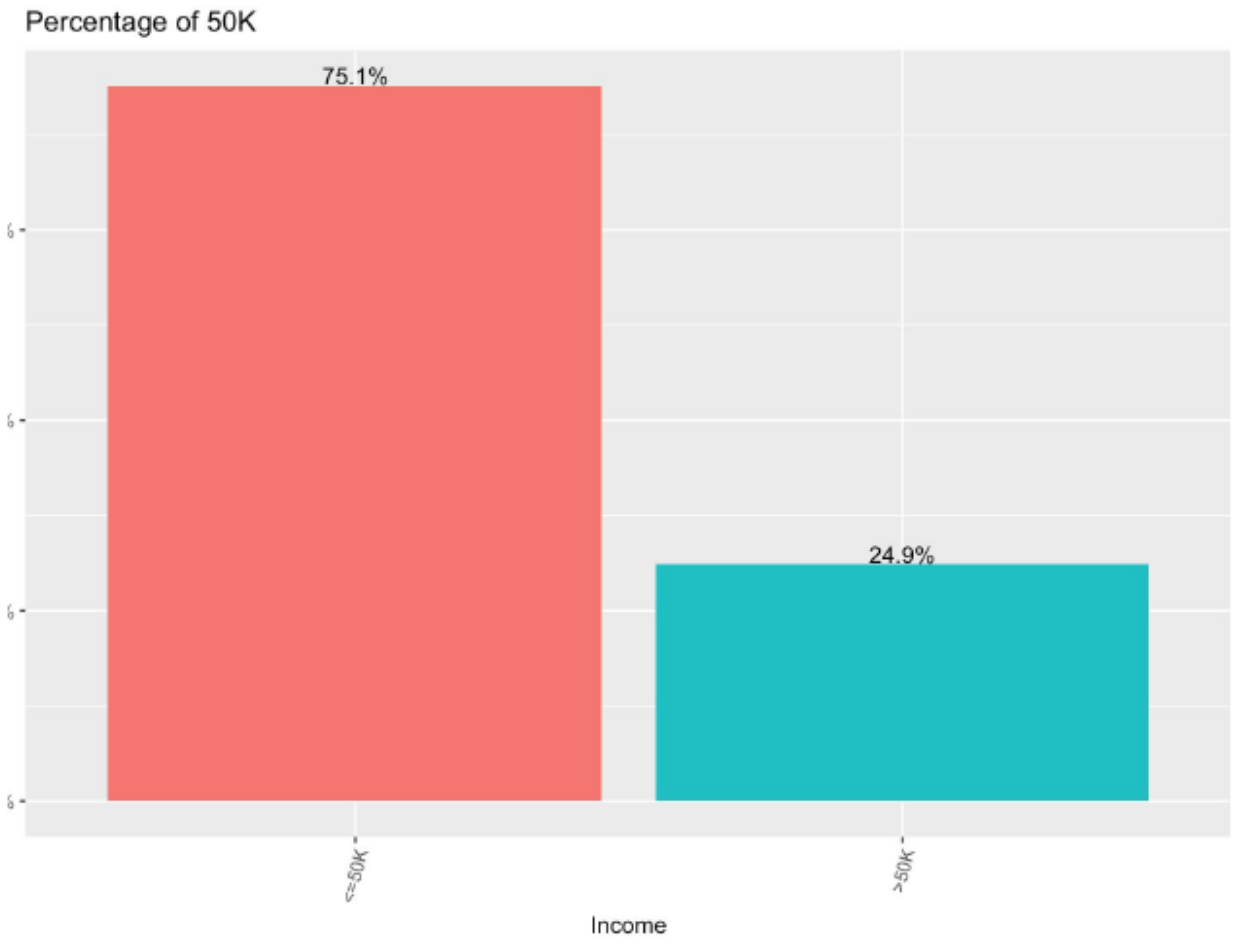
Capital Gains and Losses:

This is a continuous variable that needs to be standardized to gain a better understanding of its implications. The range of non-zero capital gains is between 114 - 99999 dollar with a mean of $12939. These were re-coded by using the first and third quartile as cutoff points between categories. This same tactic was used for Capital losses.

Capital Gain or Loss	Category	Count
Gain	Zero	41432

Gain	Low (0 - 3411)	929
Gain	Medium (3411 - 14084)	1954
Gain	High (> 14084)	907
Loss	Zero	43082
Loss	Low (0 - 1672)	536
Loss	Medium (1672 - 1977)	1118
Loss	High (> 1977)	486

Please note that all the recategorization that was done did not replace the original variables, instead they were appended to the dataset with "_category" suffix. Usage of these variables was left to the discretion of the modeling technique utilized.



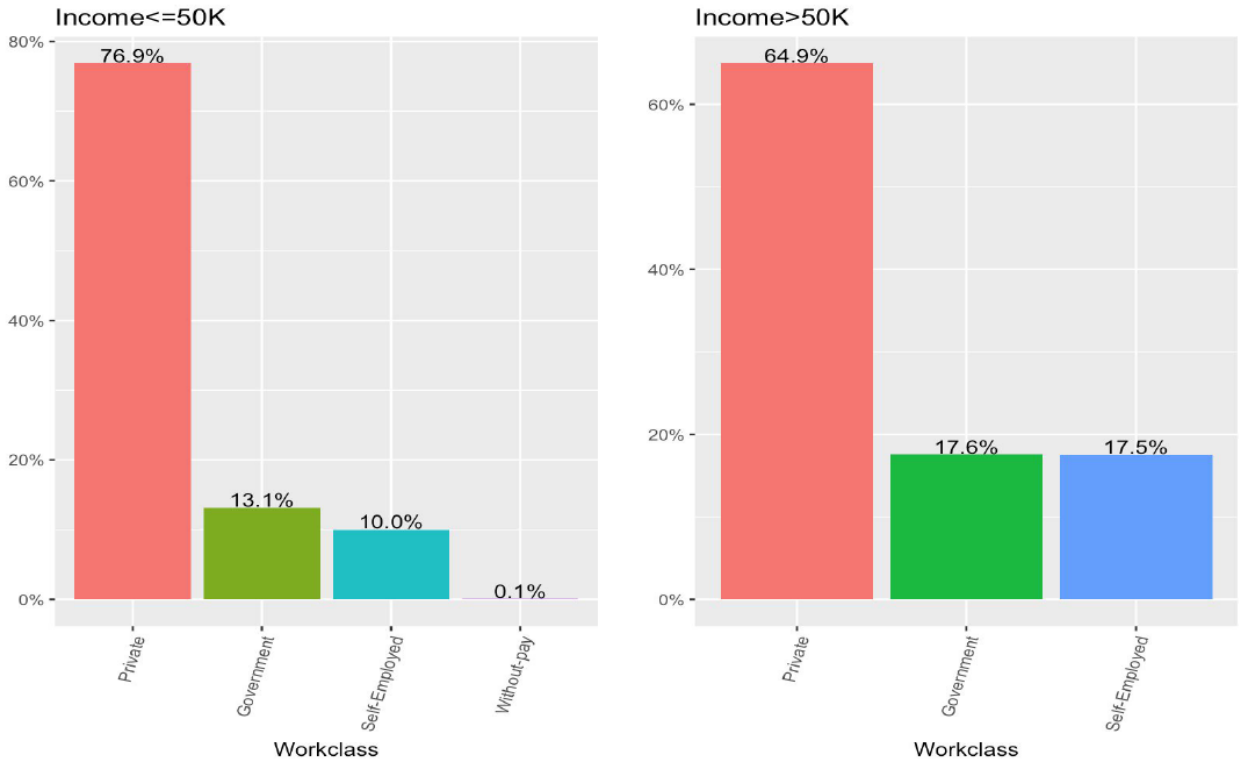
Exploratory Data Analysis

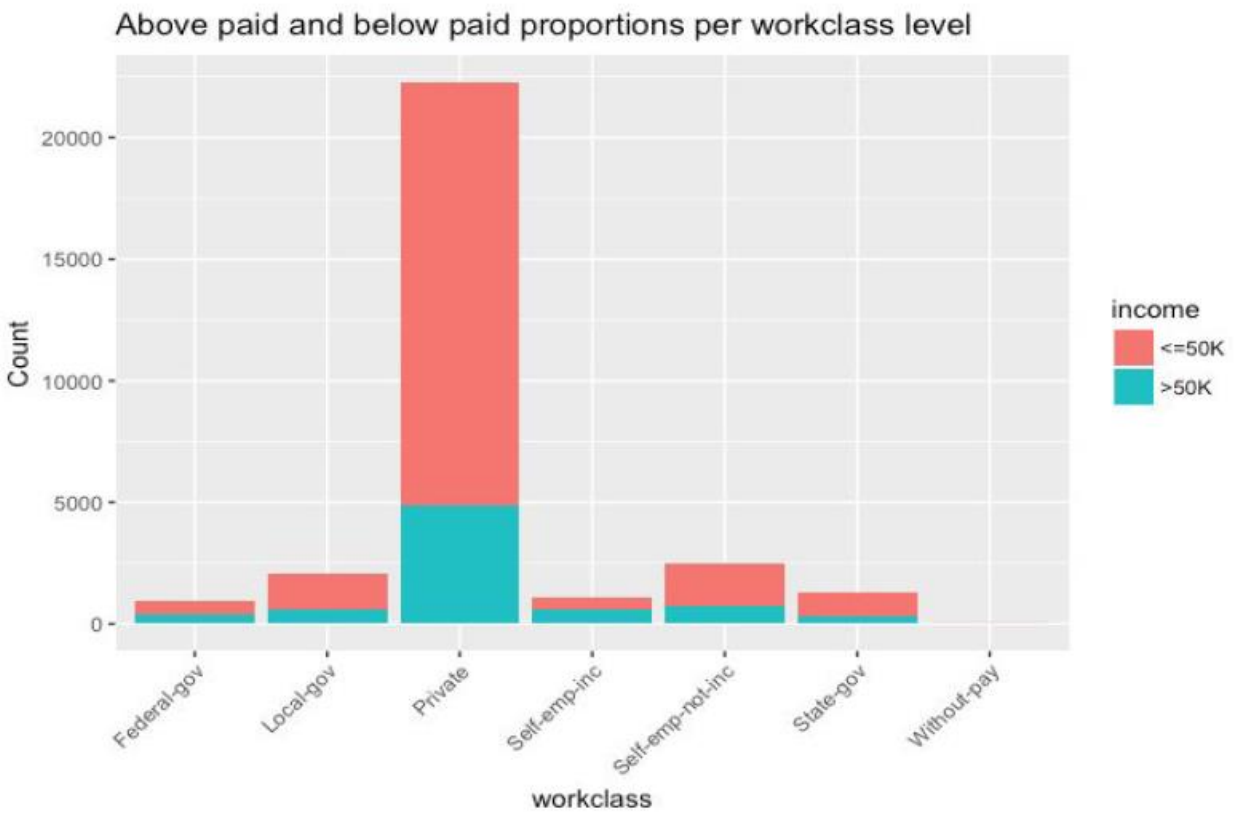
To better understand the factors that were most influential in incomes greater than $50K in 1994, we first explored the data to investigate the distributions of individual variables with respect to income category and in total. We further analyzed the relationship of those individual variables with the income category and the correlation between the variables themselves. The training and test sets are cleaned

up prior to this EDA. Details of the EDA to assess the impact of explanatory variables on the likelihood of the high income are briefly summarized below.

Workclass:

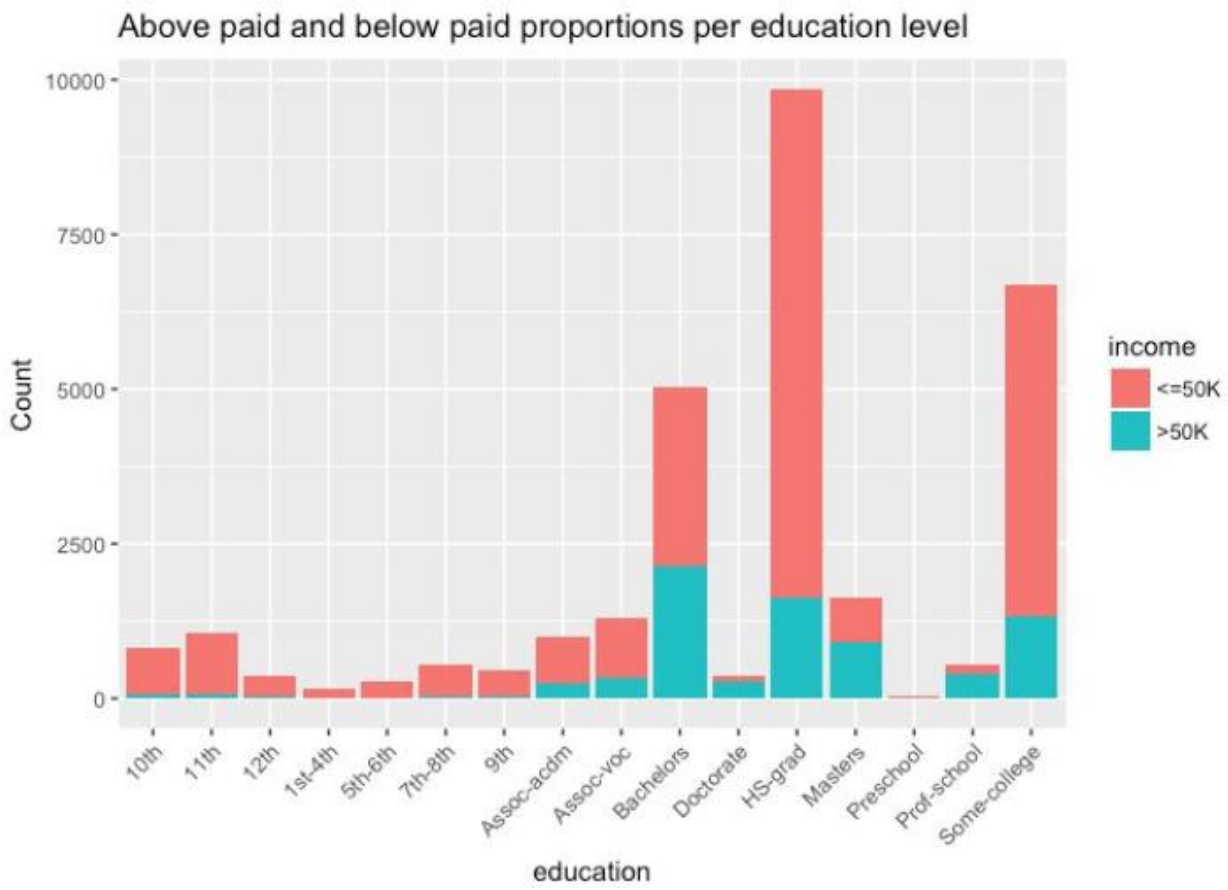
The dataset comprises of 74% of the people who worked in Private sector. Incorporated self-employed workers had highest percentage (more than half, 56%) that earned more than 50K. Column percentages show that incorporated self employed workers earn significantly more than the rest of workclass levels. The dataset has most people work in Private workclass category level which has the highest percentage of people who not only make more than 50K but also the highest percentage of people in both the >50K and <=50K groups. Please look at Appendix II: Workclass for percentage tables.





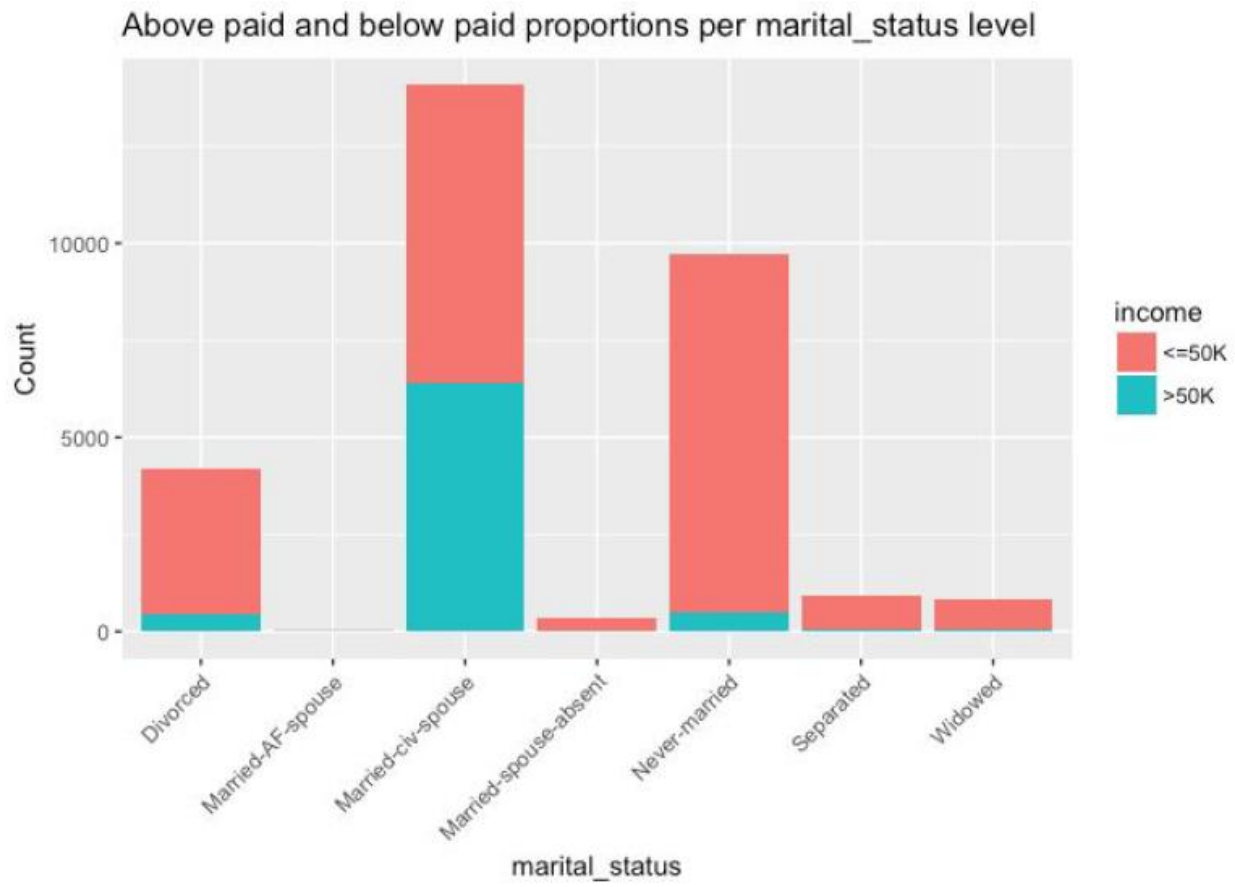
Education:

Data is dominated by HS-grad. As education level increases, percentage of people earning >50K increases. Higher education degree levels like Bachelors to PhD significantly contribute to higher pay grades. Please look at Appendix II: Education for percentage tables.



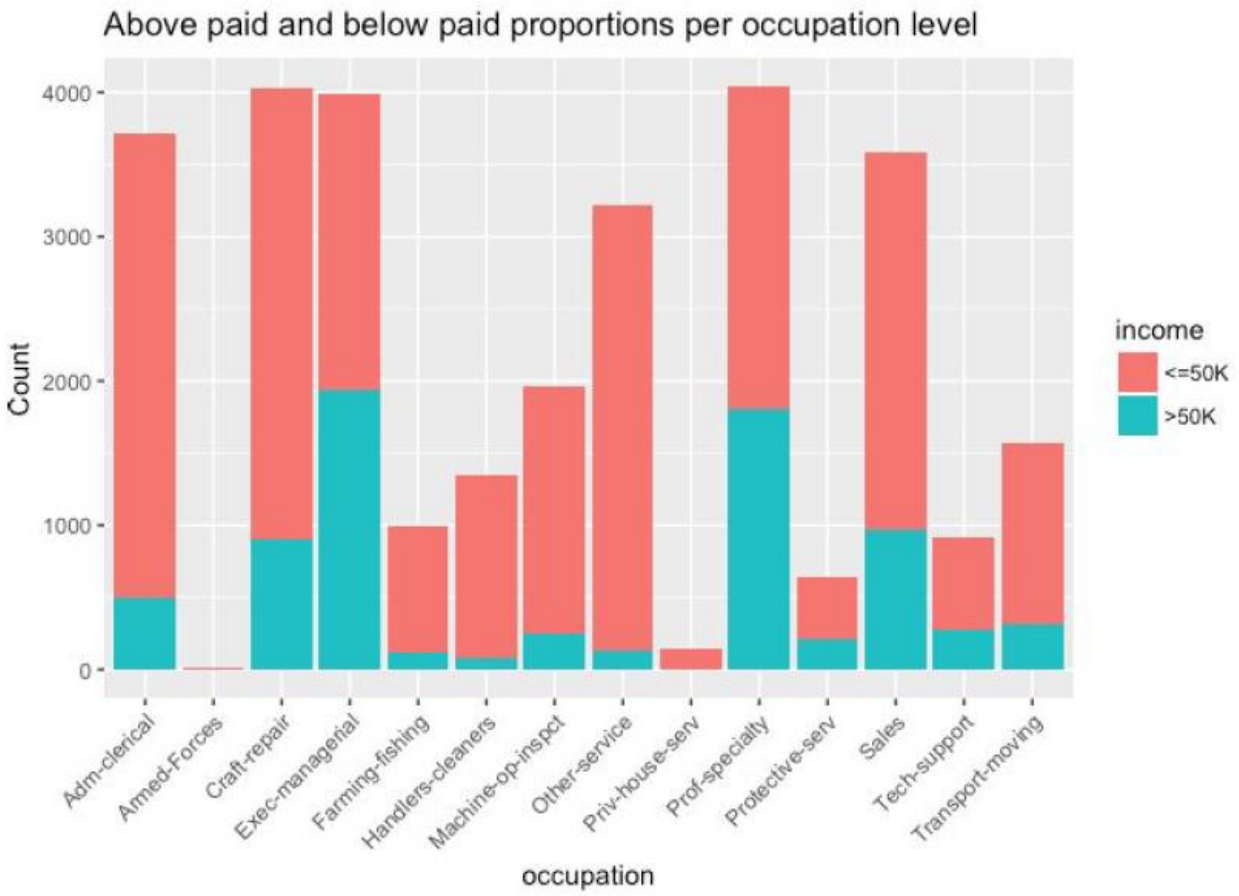
Marital Status:

High income earners tend to be married and living with spouse. Please look at Appendix II: Marital Status for percentage tables.



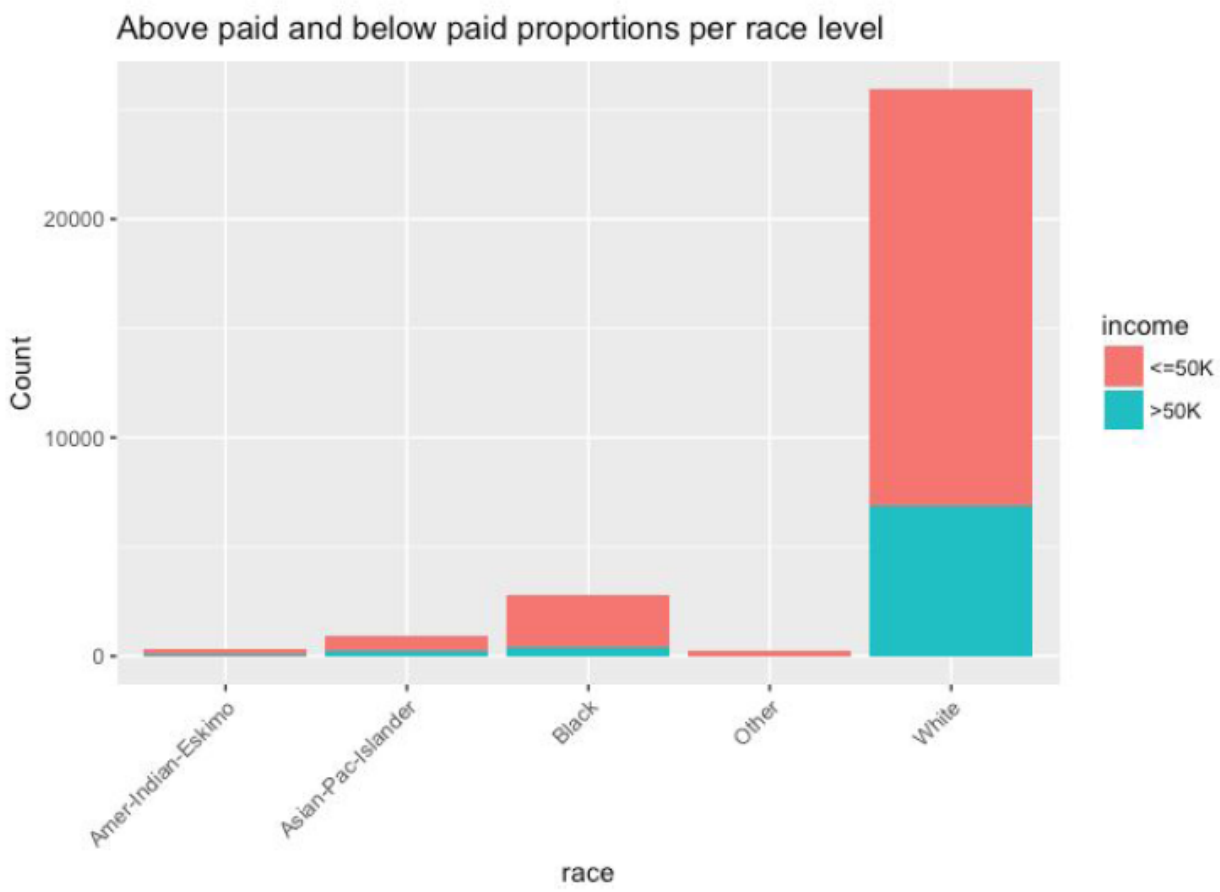
Occupation:

High percentage of >50K earner worked in executive managerial and professional specialty occupations **.** Please look at Appendix II: Occupation for percentage tables.



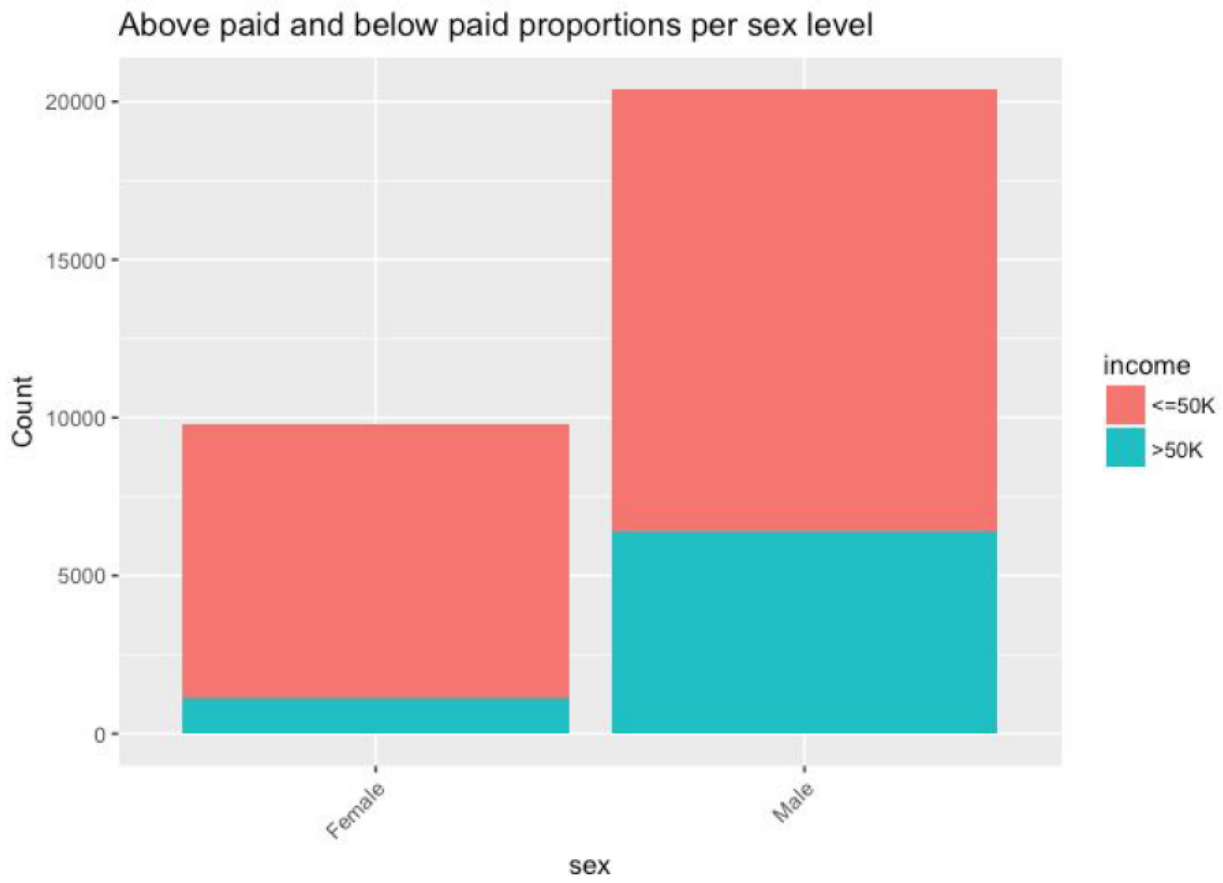
Race:

High earning races are Asian pacific islanders and whites. Please look at Appendix II: Race for percentage tables.



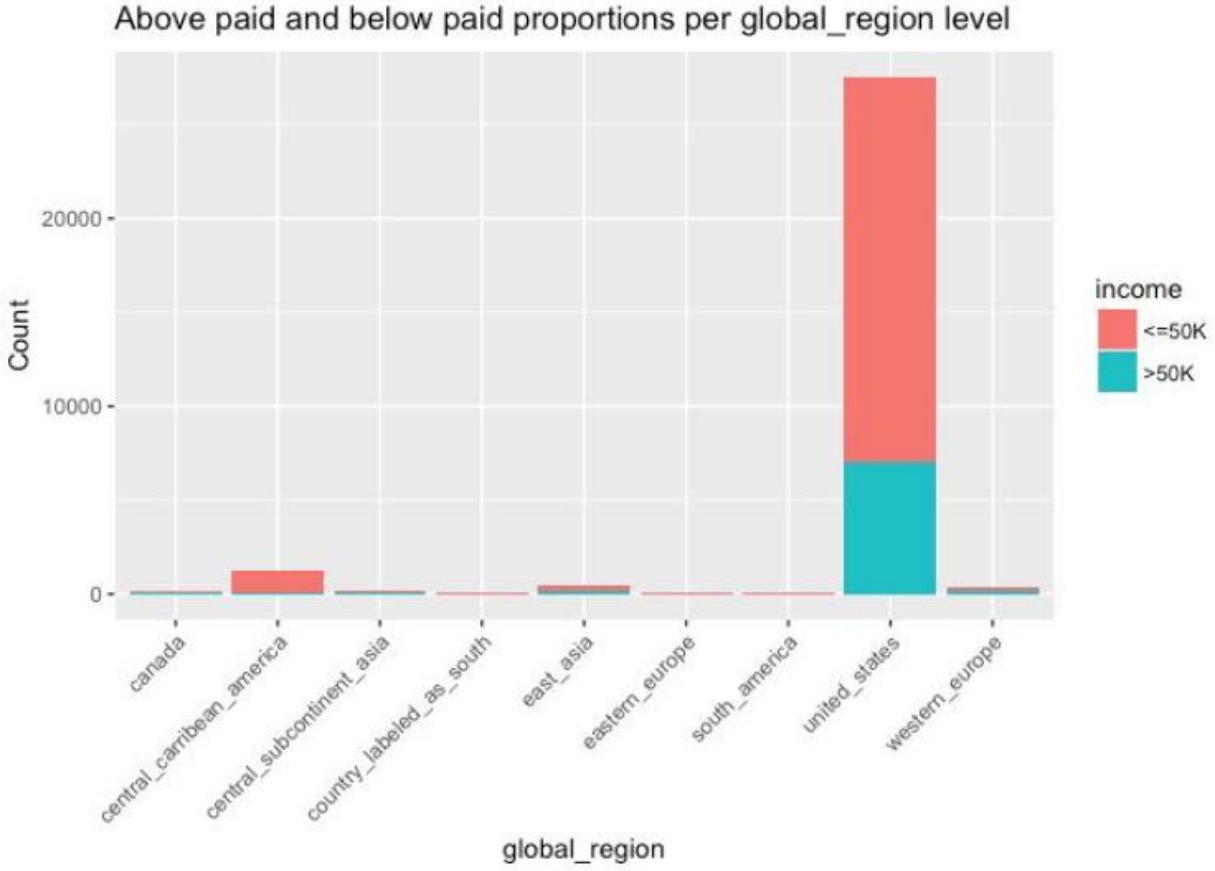
Sex:

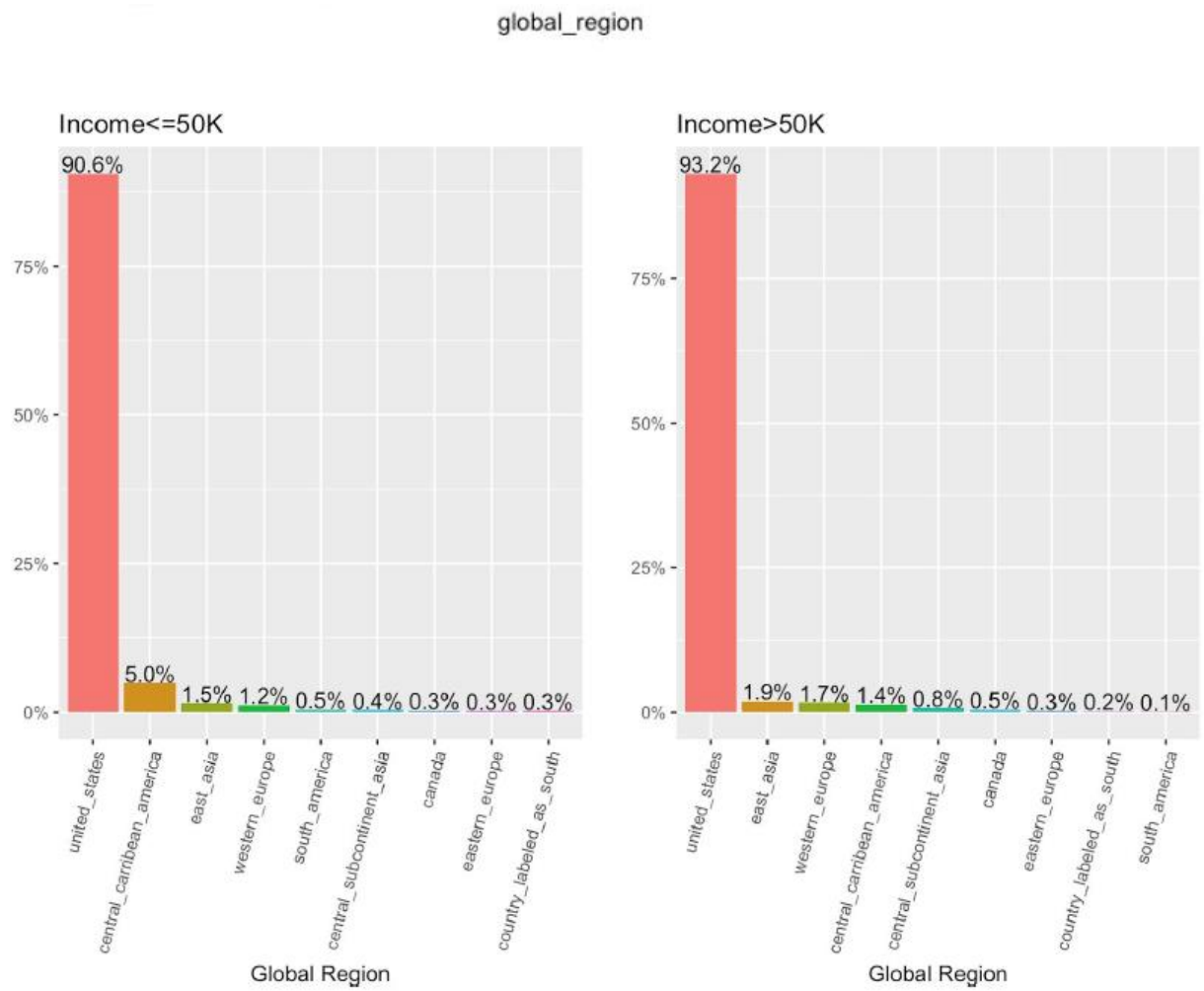
High paid workers are much more likely to be male. Please look at Appendix II: Sex for percentage tables.



Global Region:

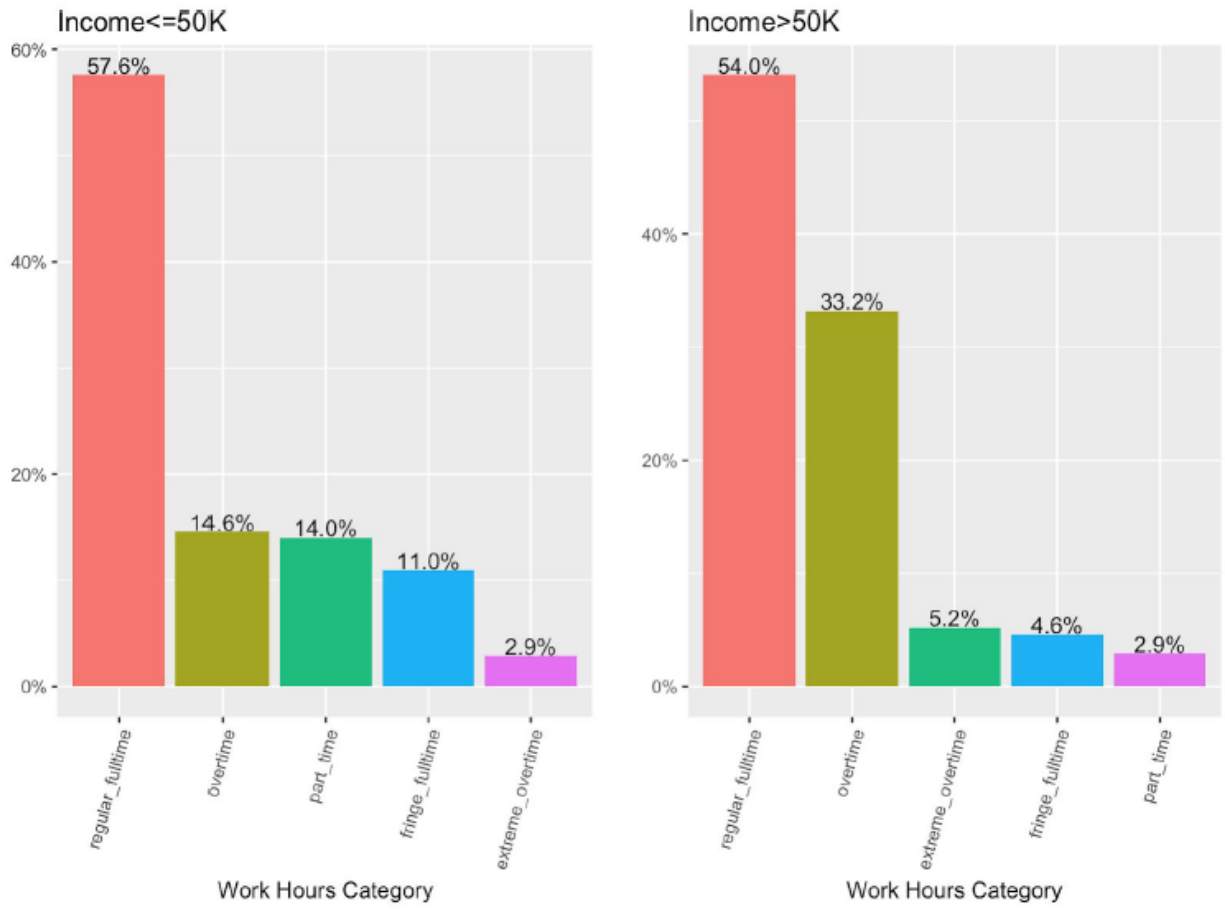
Although there are relatively a high percentage of high earners from the US, the percentage of high earners was higher for people from central_subcontinent_asia region. Please look at Appendix II: Global Region for percentage tables.


Above paid and below paid proportions per global_region level
Count
20000
10000
0
canada
central_carribean_america
central_subcontinent_asia
country_labeled_as_south
east_asia
eastern_europe
south_america
united_states
western_europe
global_region
income
<=50K
>50K



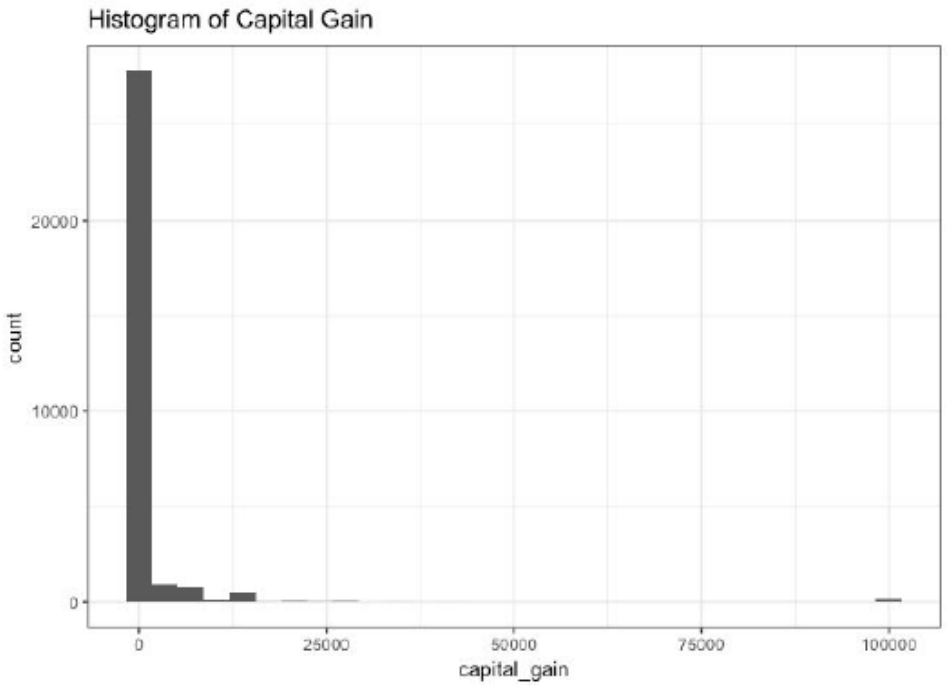
Work Hours Category:

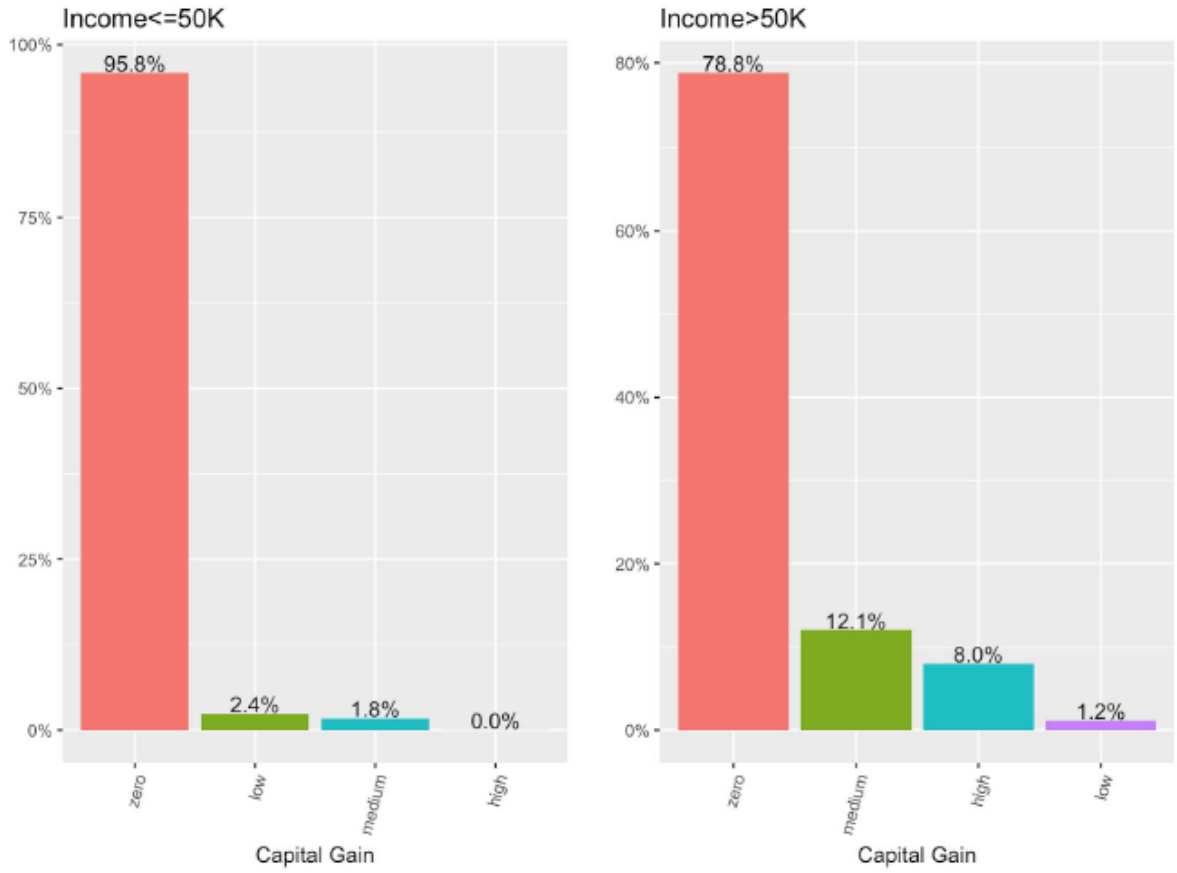
Noticeable difference is that >50K earners tend to work more overtime.



Capital Gain:

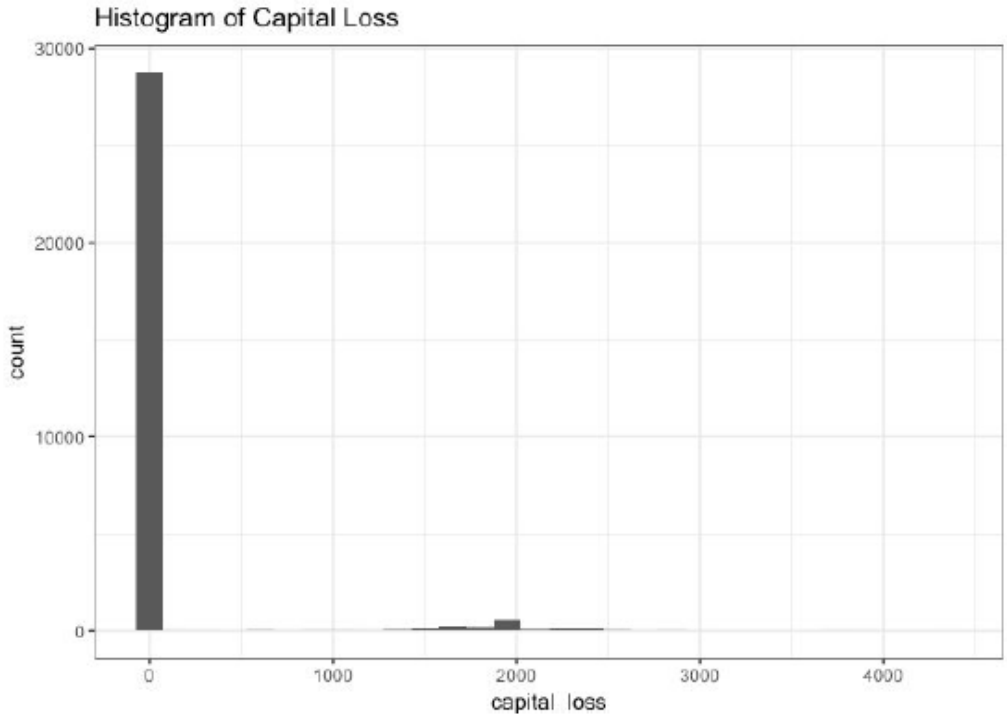
Statement about capital gain here. >50K earners have higher capital gains relative to <= 50K earners. Please look at Appendix II: Capital Gain for percentage tables.

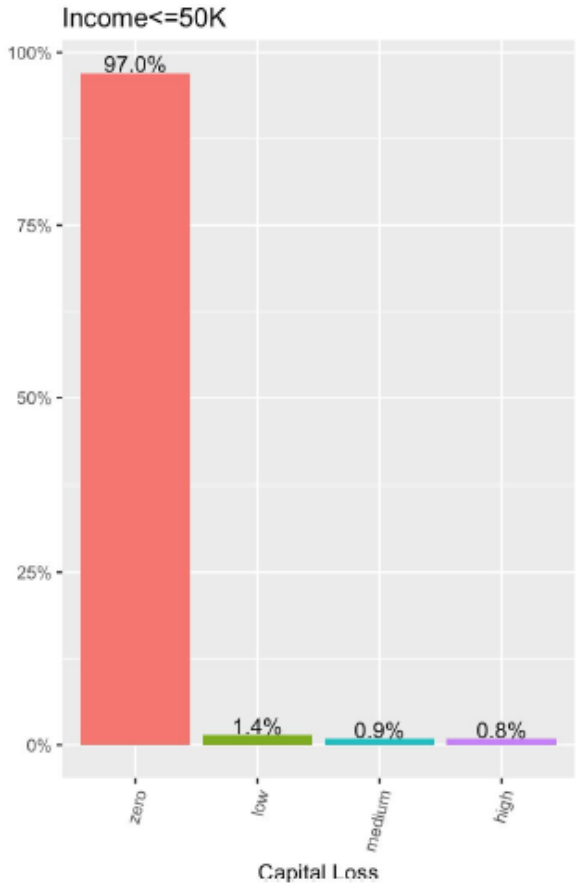


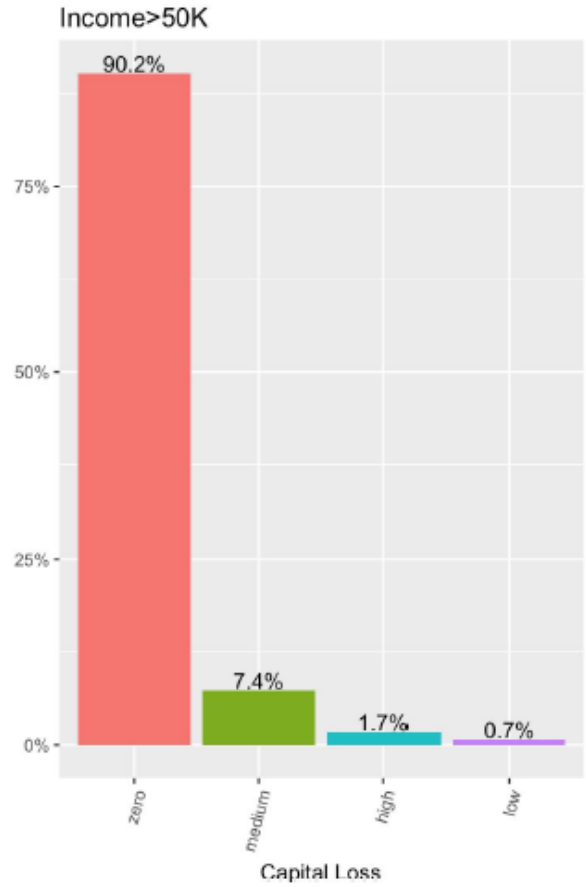


Capital Loss:

>50K earners tend to have a greater percentage of people with capital loss. Please look at Appendix II: Capital Loss for percentage tables.

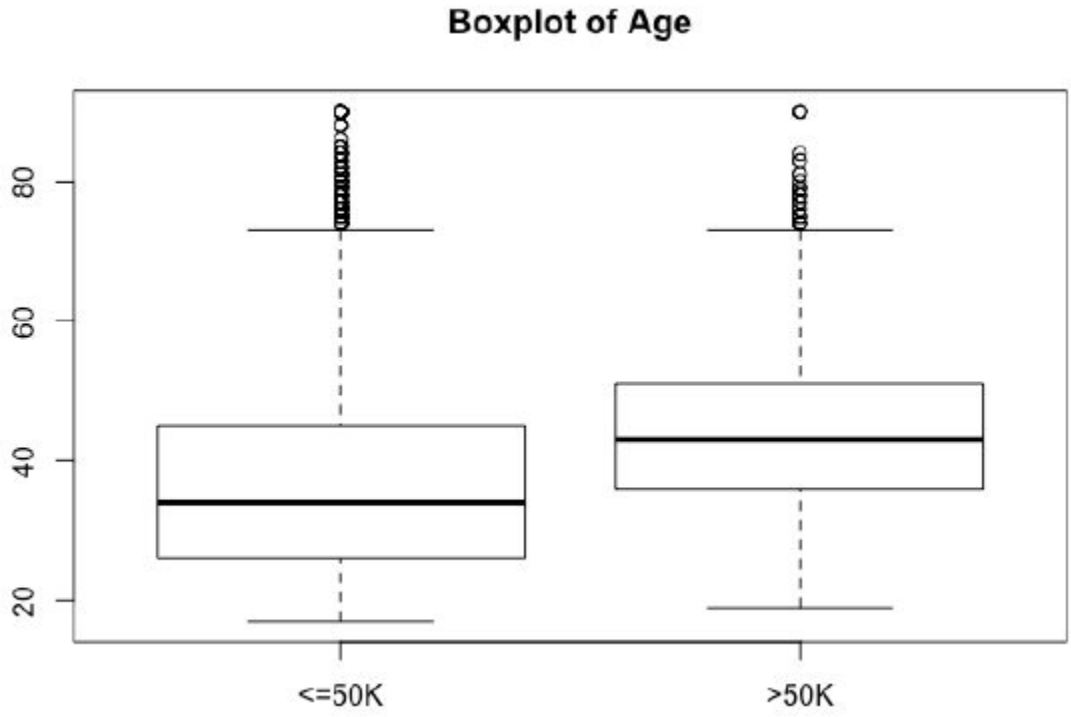

Histogram of Capital Loss
30000
20000
10000
0
count
0
1000
2000
3000
4000
capital_loss


Income<=50K
100%
75%
50%
25%
0%
97.0%
1.4%
0.9%
0.8%
zero
low
medium
high
Capital Loss


Income>50K
75%
50%
25%
0%
90.2%
7.4%
1.7%
0.7%
zero
medium
high
low
Capital Loss

Age:

Boxplot of Age shows people who earn >50k are between age of 36 to 51 with 43 as the median. Please look at Appendix II: Age for percentage tables.



Correlation Scatterplot Matrix:

The scatterplots and correlation matrix shows no significant correlation between any of the continuous variables. Please look at Appendix II: Correlation Scatterplot Matrix for scatterplot matrix image.

	age	education_num	hours_per_week	capital_gain
age	1.00000000	0.04352609	0.10159876	0.08015423
education_num	0.04352609	1.00000000	0.15252207	0.12441600
hours_per_week	0.10159876	0.15252207	1.00000000	0.08043180
capital_gain	0.08015423	0.12441600	0.08043180	1.00000000
capital_loss	0.06016548	0.07964641	0.05241705	-0.03222933

	capital_loss
age	0.06016548
education_num	0.07964641
hours_per_week	0.05241705
capital_gain	-0.03222933
capital_loss	1.00000000

Statistical Modeling

Predictive Model:

This dataset consists of majority of qualitative variables. Some of the qualitative variables are highly skewed. e.g. 86% of the data coming from “White” in “race” variable and 74 % of the data from “Private” sector “work class” variable. We have different scales for different variables. To avoid few variables overweighting others in Principal Component Analysis (PCA) simply because of absolute scale we normalized all the variables. One of the limitations of PCA in this dataset is that the dependent variable levels are heavily skewed e.g. 75.1% observations from low income level and rest 24.9% from high income levels. So, we chose to use logistic regression to create a predictive model.

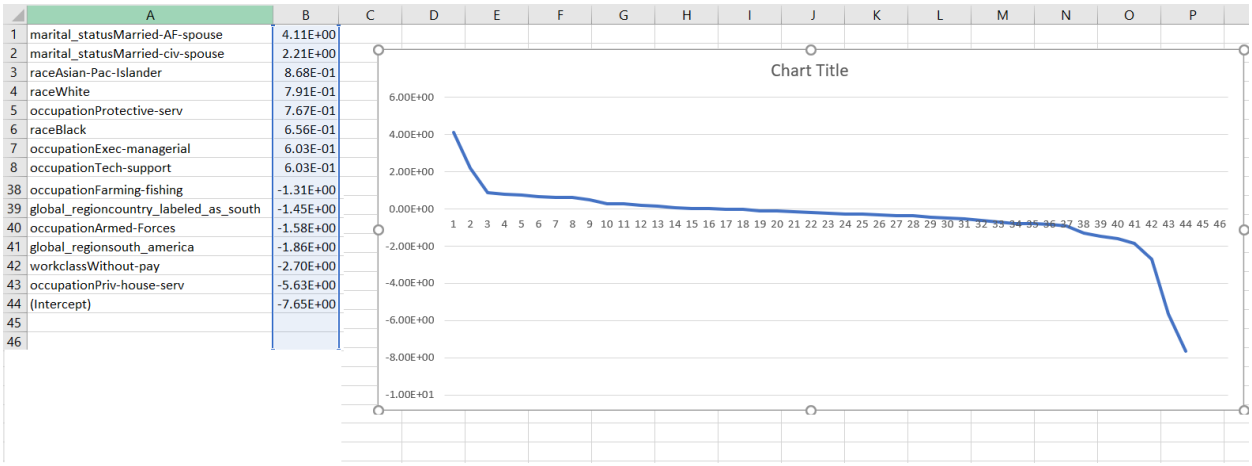
Logistic Regression:

Since the response variable is binary, we are going to use logistic regression to predict who can earn >50k on test data set provided with this Adult census data. Based on EDA, we chose age, workclass, education_num, marital_status, occupation, race, sex, capital_gain, capital_loss, hours_per_week, global_region as predictive explanatory variables in our logistic regression model. All the variables used in the model are statistically significant. Briefly writing the logistic regression equation below:

*Ln(p/1-p) = y = -7.648 + (0.0341*age) + (0.283*education_num) + … .*

Both age (indirectly work experience) and education level increase the log odds of earning higher income in 1994.

The following image shows the coefficient values of the predictor variables and their plot (Copy pasted from R output to Excel sheet to order them and plot for the ease of identification and to get better idea of the factors significance using data visualization) in significance order which are most important predictors of earning higher income.



	A	B
1	marital_statusMarried-AF-spouse	4.11E+00
2	marital_statusMarried-civ-spouse	2.21E+00
3	raceAsian-Pac-Islander	8.68E-01
4	raceWhite	7.91E-01
5	occupationProtective-serv	7.67E-01
6	raceBlack	6.56E-01
7	occupationExec-managerial	6.03E-01
8	occupationTech-support	6.03E-01
38	occupationFarming-fishing	-1.31E+00
39	global_regioncountry_labeled_as_south	-1.45E+00
40	occupationArmed-Forces	-1.58E+00
41	global_regionsouth_america	-1.86E+00
42	workclassWithout-pay	-2.70E+00
43	occupationPriv-house-serv	-5.63E+00
44	(Intercept)	-7.65E+00
45		
46		

Goodness of Fit:

This logistic regression model is statistically significant. We also verified whether the first six predictions by our model matches with actual data.

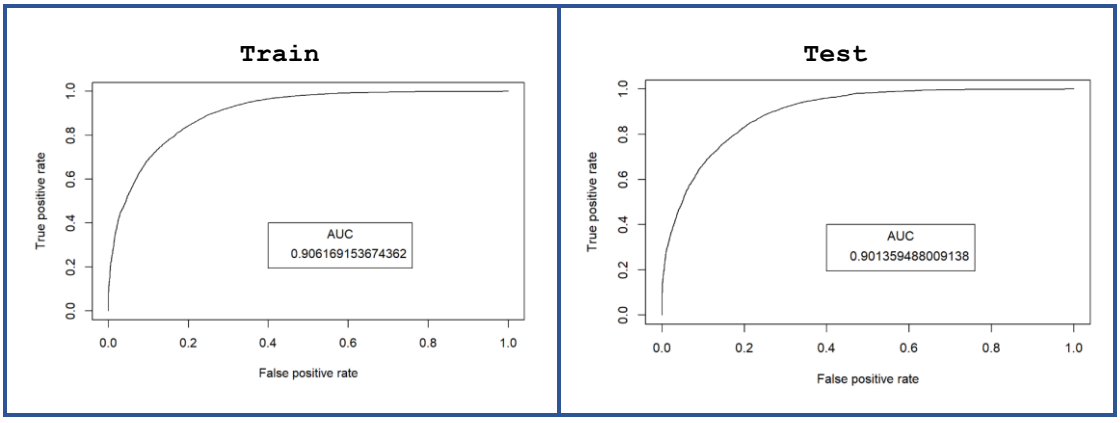
Measure	Train	Test

Confusion Matrix	<table><tr><td></td><td>0</td><td>1</td></tr><tr><td>0</td><td>6042</td><td>1205</td></tr><tr><td>1</td><td>1485</td><td>6305</td></tr></table>	<table><tr><td></td><td>0</td><td>1</td></tr><tr><td>0</td><td>9015</td><td>584</td></tr><tr><td>1</td><td>2345</td><td>3116</td></tr></table>
Misclassification rate	0.179	0.19
Accuracy	82.26%	84.54%

Synopsis:

I n summary, higher income community comprised of people who are married civilians and living with spouse with good marital relationship ("marital_status" category), people of "race" Asian-Pac-Islander or White and people who are working ("occupation" category) in Protective services or Exec-managerial or Tech-support. Lower income community comprised of people who are without pay ("workclass"), people who are in the "occupation" of Priv-house-serv, people from south_america "region".

ROC curve and AUC



Appendix:

I. Data Cleanup

```
library("ggplot2")
library("dplyr")
#read file, NA values in file are denoted by question marks
train <- read.table("adult.data", sep = ",", header = FALSE, na.strings = "?", strip.white=TRUE)
test <- read.table("adult.test", sep = ",", header = FALSE, na.strings = "?", skip = 1, strip.white=TRUE)
#set column names
colnames(train) <- c("age", "workclass", "fnlwgt", "education", "education_num", "marital_status", "occupation",
"relationship", "race", "sex", "capital_gain", "capital_loss", "hours_per_week", "native_country", "income")
colnames(test) <- c("age", "workclass", "fnlwgt", "education", "education_num", "marital_status", "occupation", "relationship",
"race", "sex", "capital_gain", "capital_loss", "hours_per_week", "native_country", "income")
str(train)
#set test column
```

```
train$test <- FALSE
test$test <- TRUE
# update test income
levels(test$income)[1] <- "<=50K"
levels(test$income)[2] <- ">50K"
#merge test and train
merged <- rbind(train, test)
#remove unnecessary columns, there is only one value thus is unncessary
merged$fnlwgt <- NULL
#Remove NAs
#based off visual inspection of the data, there are a number of NA values. Lets first find the number of rows with atleast one NA value
subsetOfNAData <- train[!complete.cases(train),]
summary(subsetOfNAData)
numberNARows <- nrow(train[!complete.cases(train),])
percentNARows <- (numberNARows / nrow(train)) * 100
percentNARows
# Based off the small percentage of NA data and the overall number of rows in the dataset (32561), I am going to make the decisison that removing the rows with NA values will be more beneficial then making assumptions on filling those values.
#remove NA rows
merged <- na.omit(merged)
#re-number row names of dataframe
row.names(merged) <- 1:nrow(merged)
# for each categorical variable we want to analyze if categories can be combined or if levels need to be recoded or dropped:
# I. for workclass:
summary(merged$workclass)
merged$workclass <- droplevels(merged$workclass)
levels(merged$workclass)
merged$workclass_category <- merged$workclass
# combine into Government job
merged$workclass_category <- gsub('Federal-gov', 'Government', merged$workclass_category)
merged$workclass_category <- gsub('Local-gov', 'Government', merged$workclass_category)
merged$workclass_category <- gsub('State-gov', 'Government', merged$workclass_category)
# combine into Self-Employed job
merged$workclass_category <- gsub('Self-emp-inc', 'Self-Employed', merged$workclass_category)
merged$workclass_category <- gsub('Self-emp-not-inc', 'Self-Employed', merged$workclass_category)
merged$workclass_category <- as.factor(merged$workclass_category)
# II. for hours per week
summary(train$hours_per_week)
ggplot(train) + aes(x=hours_per_week, group=income, fill=income) + geom_histogram(binwidth = 5)
merged$hours_per_week_category[merged$hours_per_week < 30] <- "part_time"
merged$hours_per_week_category[merged$hours_per_week >= 30 & merged$hours_per_week <= 37] <- "fringe_fulltime"
merged$hours_per_week_category[merged$hours_per_week > 37 & merged$hours_per_week <= 45 ] <- "regular_fulltime"
merged$hours_per_week_category[merged$hours_per_week > 45 & merged$hours_per_week <= 60 ] <- "overtime"
merged$hours_per_week_category[merged$hours_per_week > 60] <- "extreme_overtime"
merged$hours_per_week_category <- as.factor(merged$hours_per_week_category)
# III. separate native region into column by global regions
east_asia <- c("Cambodia", "China", "Hong", "Laos", "Thailand", "Japan", "Taiwan", "Vietnam", "Philippines")
central_subcontinent_asia <- c("India", "Iran")
central_carribean_america <- c("Cuba", "Guatemala", "Jamaica", "Nicaragua", "Puerto-Rico", "Dominican-Republic",
"El-Salvador", "Haiti", "Honduras", "Mexico", "Trinadad&Tobago")
south_america <- c("Ecuador", "Peru", "Columbia")
western_europe <- c("England", "Germany", "Holand-Netherlands", "Ireland", "France", "Greece", "Italy", "Portugal",
"Scotland")
eastern_europe <- c("Poland", "Yugoslavia", "Hungary")
united_states <- c("United-States", "Outlying-US(Guam-USVI-etc)", "Outlying-US")
merged$global_region[merged$native_country %in% east_asia] <- "east_asia"
merged$global_region[merged$native_country %in% central_subcontinent_asia] <- "central_subcontinent_asia"
```

```
merged$global_region[merged$native_country %in% central_carribean_america] <- "central_carribean_america"
merged$global_region[merged$native_country %in% south_america] <- "south_america"
merged$global_region[merged$native_country %in% western_europe] <- "western_europe"
merged$global_region[merged$native_country %in% eastern_europe] <- "eastern_europe"
merged$global_region[merged$native_country %in% united_states] <- "united_states"
merged$global_region[merged$native_country == "Canada"] <- "canada"
merged$global_region[merged$native_country == "South"] <- "country_labeled_as_south"
merged$global_region <- as.factor(merged$global_region)
# IV. look at capital gains and loses
non_zero_capital_gains_subset = subset(train, train$capital_gain != 0)
non_zero_capital_loss_subset = subset(train, train$capital_loss != 0)
summary_gain <- summary(non_zero_capital_gains_subset$capital_gain)
gain_low_cutoff <- as.integer(summary_gain[[2]])
gain_high_cutoff <- as.integer(summary_gain[[5]])
merged$capital_gain_category[merged$capital_gain == 0] <- "zero"
merged$capital_gain_category[merged$capital_gain > 0 & merged$capital_gain <= gain_low_cutoff] <- "low"
merged$capital_gain_category[merged$capital_gain > gain_low_cutoff & merged$capital_gain <= gain_high_cutoff] <-
"medium"
merged$capital_gain_category[merged$capital_gain > gain_high_cutoff] <- "high"
merged$capital_gain_category <- factor(merged$capital_gain_category, ordered=TRUE, levels=c("zero", "low", "medium",
"high"))
summary_loss <- summary(non_zero_capital_loss_subset$capital_loss)
loss_low_cutoff <- as.integer(summary_loss[[2]])
loss_high_cutoff <- as.integer(summary_loss[[5]])
merged$capital_loss_category[merged$capital_loss == 0] <- "zero"
merged$capital_loss_category[merged$capital_loss > 0 & merged$capital_loss <= loss_low_cutoff] <- "low"
merged$capital_loss_category[merged$capital_loss > loss_low_cutoff & merged$capital_loss <= loss_high_cutoff] <- "medium"
merged$capital_loss_category[merged$capital_loss > loss_high_cutoff] <- "high"
merged$capital_loss_category <- factor(merged$capital_loss_category, ordered=TRUE, levels=c("zero", "low", "medium",
"high"))
# separate train and test sets
train <- subset(merged, merged$test == FALSE)
train$test <- NULL
test <- subset(merged, merged$test == TRUE)
test$test <- NULL
row.names(test) <- 1:nrow(test)
# write to csv
write.csv(train, "train.csv", row.names = FALSE)
write.csv(test, "test.csv", row.names = FALSE)
```

Appendix II: EDA

I. Workclass

Column Percentages:

Income	Federal - gov	Local - gov	Private	Self-emp -inc	Self - not -inc	state-gov	Without - pay
<= 50K	61.29	70.54	78.12	44.13	71.43	73.1	100
> 50K	38.71	29.46	21.88	55.87	28.57	26.9	0
Total	100.0	100.0	100.0	100.0	100.0	100.0	100.0

Row Percentages:

Income	Federal - gov	Local - gov	Private	Self-emp -inc	Self - not -inc	state-gov	Without - pay

Total
<= 50K 2.55 6.44 76.85 2.09 7.88 4.13 .06 100.0
> 50K 4.86 8.11 64.94 7.99 9.51 4.58 0.0 100.0
II. Education
III. Marital Status
IV. Occupation
V. Race
VI. Sex
VII. Global Region
VIII. Work Hours Category: No contents
IX. Capital Gain
X. Capital Loss
XI. Age: No contents
XII. Correlation Scatterplot Matrix

III. EDA Code

#Import data
```{r echo = TRUE}
library(tigerstats)
library(sqldf)
library(ggplot2)
trainDf <- read.csv("train.csv", header = TRUE)
testDf <- read.csv("test.csv", header = TRUE)
summary(trainDf)
```
EDA of Variables of Interest
For building model, We decided not to use "education", "relationship" variables from the summary of the data.Instead of "native_country", we are using "global_region".
Percentages of above and below paid Per level of categorical variables and plot for Catergorical variables.
Column Percentages and row percentages of above and below paid Per workclass category.
```{r echo = TRUE}
colPerc(xtabs(~income+workclass, trainDf))
rowPerc(xtabs(~income+workclass, trainDf))
workclass_inc <-sqldf("select workclass, income, count(workclass) as Count from trainDf group by workclass, income")
# Plot of above paid and below paid counts per workclass level
ggplot(workclass_inc, aes(x=workclass, y=Count, fill=income)) +
geom_bar(stat="identity") +
theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
ggtitle('Above paid and below paid proportions per workclass level')
```
* Column percentages shows that incorporated self employed workers earn significanly more than the rest of the catergories.
* The dataset has most people work in Private workclass category level, thus most people earning >50K.
Percentages of above and below paid Per education category
```{r echo = TRUE}
colPerc(xtabs(~income+education, trainDf))#Variable of Interest
education_inc <-sqldf("select education, income, count(education) as Count from trainDf group by education, income")
### Plot of above paid and below paid counts per education level
ggplot(education_inc, aes(x=education, y=Count, fill=income)) +
geom_bar(stat="identity") +
theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
ggtitle('Above paid and below paid proportions per education level')
```
* As education level increases, the probability of of getting paid >50K increases. Higher education levels like Bachlors to PhD significanly contribute to higher pay grades.
###Marital_status: High income earners are married and living with spouse, good marital relationship.
```{r echo = TRUE}
colPerc(xtabs(~income+marital_status, trainDf))#Variable of Interest
maritalStatus_inc <-sqldf("select marital_status, income, count(marital_status) as Count from trainDf group by marital_status,
```

```
income")
# Plot of above paid and below paid counts per marital_status level
ggplot(maritalStatus_inc, aes(x=marital_status, y=Count, fill=income)) +
geom_bar(stat="identity") +
theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
ggtitle('Above paid and below paid proportions per marital_status level')
```
Occupation: High earner are likely be working in executive managerial and professional specialty occupataions.
```{r echo = TRUE}
colPerc(xtabs(~income+occupation, trainDf))
rowPerc(xtabs(~income+occupation, trainDf))
occupation_inc <-sqldf("select occupation, income, count(occupation) as Count from trainDf group by occupation, income")
# Plot of above paid and below paid counts per education level
ggplot(occupation_inc, aes(x=occupation, y=Count, fill=income)) +
geom_bar(stat="identity") +
theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
ggtitle('Above paid and below paid proportions per occupation level')
```
Race: High earner are more likely to be Asian pacific islanders and whites.
```{r echo = TRUE}
colPerc(xtabs(~income+race, trainDf))#Variable of Interest
rowPerc(xtabs(~income+race, trainDf))
race_inc <-sqldf("select race, income, count(race) as Count from trainDf group by race, income")
# Plot of above paid and below paid counts per race level
ggplot(race_inc, aes(x=race, y=Count, fill=income)) +
geom_bar(stat="identity") +
theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
ggtitle('Above paid and below paid proportions per race level')
```
####Sex: High paid workers are much more Likely to be Male, men tend to earn more moeny than women.
```{r echo = TRUE}
colPerc(xtabs(~income+sex, trainDf))
rowPerc(xtabs(~income+sex, trainDf))
sex_inc <-sqldf("select sex, income, count(sex) as Count from trainDf group by sex, income")
# Plot of above paid and below paid counts per sex level
ggplot(sex_inc, aes(x=sex, y=Count, fill=income)) +
geom_bar(stat="identity") +
theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
ggtitle('Above paid and below paid proportions per sex level')
```
Global_region: people from central_subcontinent_asia region are high earners.
```{r echo = TRUE}
colPerc(xtabs(~income+ global_region, trainDf))#Variable of Interest
rowPerc(xtabs(~income+ global_region, trainDf))
region_inc <-sqldf("select global_region, income, count(global_region) as Count from trainDf group by global_region, income")
# Plot of above paid and below paid counts per race level
ggplot(region_inc, aes(x=global_region, y=Count, fill=income)) +
geom_bar(stat="identity") +
theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
ggtitle('Above paid and below paid proportions per global_region level')
```
Summary stats by above and below paid groups for Continuous variables
```{r echo = TRUE}
aggregate(trainDf$capital_gain~trainDf$income,data=trainDf,summary)
qplot(capital_gain, data=trainDf, geom="histogram")+theme_bw()+ggtitle('Histogram of Capital Gain')
aggregate(trainDf$capital_loss~trainDf$income,data=trainDf,summary)
qplot(capital_loss, data=trainDf, geom="histogram")+theme_bw()+ggtitle('Histogram of Capital Loss')
```
```
```

Age: Boxplot of Age shows people who earn >50k are between age of 36 to 51 with 43 as the median
```{r echo = TRUE}
boxplot(age~income,trainDf,main="Boxplot of Age")
aggregate(trainDf$age~trainDf$income,data=trainDf,summary)
```
###Examine the correlation between the continous predictors
#####The scatterplots and correlation matrix shows no significant correlation between any of the continuous variables
####pairs(trainDf[,c("age","education_num","hours_per_week", "capital_gain", "capital_loss")])
```{r echo = TRUE}
my_cor<-cor(trainDf[,c("age","education_num","hours_per_week", "capital_gain", "capital_loss")])
my_cor
```
when I ran logistic regression code on a modified train dataset made up of equal obs for high income and low income(as per Dr. Turner's suggestion to get a good prediction model using this type of stratified sampling method), the sample workclass and occupation variables had atleast one level with Zero observations. Thus I am sampling equla proportions of oservations for high and low income
```{r echo = TRUE}
trainTest <- rbind(trainDf, testDf)
ftable(addmargins(table(trainTest$income,trainTest$occupation)))
ftable(addmargins(table(trainTest$income,trainTest$workclass)))
noPayobs<-trainTest[which(trainTest$workclass=="Without-pay"),]
```
#####I decided to not include "relationship" factor variable since it seems redundant and for the sake of simplicity of the model.From the summary of this dataset, it seems that marital_status and sex variables are capturing the information provided by "relationship" variable. To build the predictive model to predict who earn more than 50 K per annum, we decided to include age, workclass, education_num, marital_status, occupation, race, sex, capital_gain, capital_loss, hours_per_week, global_region after reviewing plots and the summary stats for all the variables.
```{r echo = TRUE}
trainDf2 <-subset(trainDf, select=c("age", "workclass", "education_num", "marital_status", "occupation", "race", "sex", "capital_gain", "capital_loss", "hours_per_week", "global_region", "income"))
testDf2 <-subset(testDf, select=c("age", "workclass", "education_num", "marital_status", "occupation", "race", "sex", "capital_gain", "capital_loss", "hours_per_week", "global_region", "income"))
noPayobs <-subset(noPayobs, select=c("age", "workclass", "education_num", "marital_status", "occupation", "race", "sex", "capital_gain", "capital_loss", "hours_per_week", "global_region", "income"))
#summary(trainDf2)
above50k <- trainDf2[trainDf2$income==">50K",]
below50k <- trainDf2[trainDf2$income=="<=50K",]
set.seed(111)
sampleBelow50 <- below50k[sample(nrow(below50k), size = 7508), ]
trainDf3 <-rbind(sampleBelow50, above50k, noPayobs)
```
####making sure that each category level of each categorical variable is represented in the sampled train data
```{r echo = TRUE}
xtabs(~income+workclass, data = trainDf3)
xtabs(~income+marital_status, data = trainDf3)
xtabs(~income+occupation, data = trainDf3)
xtabs(~income+race, data = trainDf3)
xtabs(~income+sex, data = trainDf3)
xtabs(~income+global_region, data = trainDf3)
```
# Additional plots with percentages
library(ggplot2)
library(plyr)
library(gridExtra)
library(gmodels)
library(grid)
library(vcd)
library(scales)

```
library(ggthemes)
train <- read.csv("train.csv", header = TRUE)
test <- read.csv("test.csv", header = TRUE)
# percent 50K
ggplot(data = train, mapping = aes(x = train$income, fill = train$income)) +
geom_bar(mapping = aes(y = (..count..)/sum(..count..))) +
geom_text(mapping = aes(label = scales::percent((..count..)/sum(..count..)),
y = (..count..)/sum(..count..) ), stat = "count", vjust = -.1) +
theme(legend.position = 'none', axis.text.x=element_text(angle=75,hjust=1)) +
labs(title="Percentage of 50K", x = "Income", y = "", fill = "Income") + scale_y_continuous(labels = percent)
# capital_gain_category
plot_gain <- lapply(X = levels(train$income), FUN = function(v){
df <- subset(train, train$income == v)
df <- within(df, capital_gain_category <- factor(capital_gain_category,
levels = names(sort(table(capital_gain_category), decreasing = TRUE))))
ggplot(data = df, aes(x = capital_gain_category, fill = capital_gain_category)) +
geom_bar(aes(y = (..count..)/sum(..count..))) +
geom_text(aes(label = scales::percent((..count..)/sum(..count..)), y = (..count..)/sum(..count..) ),
stat = "count", vjust = -.1) + labs(x = "Capital Gain", y = "", fill = "Capital Gain") +
theme(legend.position = 'none', axis.text.x=element_text(angle=75,hjust=1)) +
ggtitle(paste("Income", v, sep = "")) +
scale_y_continuous(labels = percent)
})
grid.arrange(grobs = plot_gain, ncol = 2)
# capital_loss_category
plot_loss <- lapply(X = levels(train$income), FUN = function(v){
df <- subset(train, train$income == v)
df <- within(df, capital_loss_category <- factor(capital_loss_category,
levels = names(sort(table(capital_loss_category), decreasing = TRUE))))
ggplot(data = df, aes(x = capital_loss_category, fill = capital_loss_category)) +
geom_bar(aes(y = (..count..)/sum(..count..))) +
geom_text(aes(label = scales::percent((..count..)/sum(..count..)), y = (..count..)/sum(..count..) ),
stat = "count", vjust = -.1) + labs(x = "Capital Loss", y = "", fill = "Capital Loss") +
theme(legend.position = 'none', axis.text.x=element_text(angle=75,hjust=1)) +
ggtitle(paste("Income", v, sep = "")) +
scale_y_continuous(labels = percent)
})
grid.arrange(grobs = plot_loss, ncol = 2)
# workclass_category
plot_workclass_category <- lapply(X = levels(train$income), FUN = function(v){
df <- subset(train, train$income == v)
df <- within(df, workclass_category <- factor(workclass_category,
levels = names(sort(table(workclass_category), decreasing = TRUE))))
ggplot(data = df, aes(x = workclass_category, fill = workclass_category)) +
geom_bar(aes(y = (..count..)/sum(..count..))) +
geom_text(aes(label = scales::percent((..count..)/sum(..count..)), y = (..count..)/sum(..count..) ),
stat = "count", vjust = -.1) + labs(x = "Workclass", y = "", fill = "Workclass") +
theme(legend.position = 'none', axis.text.x=element_text(angle=75,hjust=1)) +
ggtitle(paste("Income", v, sep = "")) +
scale_y_continuous(labels = percent)
})
grid.arrange(grobs = plot_workclass_category, ncol = 2)
# hours_per_week_category
plot_hours_per_week_category <- lapply(X = levels(train$income), FUN = function(v){
df <- subset(train, train$income == v)
df <- within(df, hours_per_week_category <- factor(hours_per_week_category,
levels = names(sort(table(hours_per_week_category), decreasing = TRUE))))
ggplot(data = df, aes(x = hours_per_week_category, fill = hours_per_week_category)) +
```

```
geom_bar(aes(y = (..count..)/sum(..count..))) +
geom_text(aes(label = scales::percent((..count..)/sum(..count..)), y = (..count..)/sum(..count..) ),
stat = "count", vjust = -.1) + labs(x = "Work Hours Category", y = "", fill = "Work Hours Category") +
theme(legend.position = 'none', axis.text.x=element_text(angle=75,hjust=1)) +
ggtitle(paste("Income", v, sep = "")) +
scale_y_continuous(labels = percent)
})
grid.arrange(grobs = plot_hours_per_week_category, ncol = 2)
# global_region
plot_global_region <- lapply(X = levels(train$income), FUN = function(v){
df <- subset(train, train$income == v)
df <- within(df, global_region <- factor(global_region,
levels = names(sort(table(global_region), decreasing = TRUE))))
ggplot(data = df, aes(x = global_region, fill = global_region)) +
geom_bar(aes(y = (..count..)/sum(..count..))) +
geom_text(aes(label = scales::percent((..count..)/sum(..count..)), y = (..count..)/sum(..count..) ),
stat = "count", vjust = -.1) + labs(x = "Global Region", y = "", fill = "Global Region") +
theme(legend.position = 'none', axis.text.x=element_text(angle=75,hjust=1)) +
ggtitle(paste("Income", v, sep = "")) +
scale_y_continuous(labels = percent)
})
grid.arrange(grobs = plot_global_region, ncol = 2)
```

IV. Logistic Regression Code

```
grid.arrange(grobs = plot_global_region, ncol = 2)
###Logistic regression
```{r echo = TRUE}
#install.packages("nnet")
library(nnet)
#install.packages("ROCR")
library(ROCR)
trainDf3$income<-ifelse(trainDf3$income=='>50K',1,0)
testDf2$income<-ifelse(testDf2$income=='>50K',1,0)
mymodel<- multinom(income~., data = trainDf3)
```
Goodness of fit test: This model is statistically significant
```{r echo = TRUE}
#To get ready-made z-values and p-values for the coeffients and to get Goodness-of-fit test ran the same model using glm
funtion.
#Residual Deviance: 11579.03 and AIC: 11667.03 are exactly same for both models using differnt R packages
model2<-glm(income~., data = trainDf3, family = "binomial")
summary(model2)
#Goodness of fit test: This model is statistically significant
with(model2, pchisq(null.deviance-deviance, df.null-df.residual, lower.tail = F ))
```
See whether predProb(prediction Probalbilties) matches with actual data and it does for atleast first 6 obs
```{r echo = TRUE}
predProb <- predict(mymodel, trainDf3, type = "prob")
#hist(predProb)
head(predProb)
head(trainDf3)
###### Confusion Matrix and Misclassification Rate on Train
confMatrix <- predict(mymodel, trainDf3)
tab <- table(confMatrix, trainDf3$income)
tab ### this print confsion matrix
#### Misclassification Rate
1-sum(diag(tab))/sum(tab)
predProb <- prediction(predProb, trainDf3$income)
perfEval <- performance(predProb, "acc")
```
```

```
plot(perfEval)
maxYval <- which.max(slot(perfEval, "y.values")[[1]])
maxYval
acc <- slot(perfEval, "y.values")[[1]][maxYval]
acc
```
Thus, the accuracy of the logistic regression on Train data is 82.26%
ROC on Train data
```{r echo = TRUE}
roc <- performance(predProb, "tpr", "fpr")
plot(roc)
#abline(a= 0, b=1)
#### AUC(Area Under Curve) on Train data
auc <- performance(predProb, "auc")
auc<- unlist(slot(auc, "y.values"))
auc # area under the curve for train is .9062
legend(.4,.4, auc, title= "AUC")
```
ROC on Test data
```{r echo = TRUE}
myTstmodel<- multinom(income~., data = trainDf3)
predProbTst <- predict(myTstmodel, testDf2, type = "prob")
#hist(predProbTst)
#see whether predProb(prediction Probalbilties) matches with actual data and it does for atleast first 6 obs
head(predProbTst)
head(testDf2)
#Confusion Matrix and Misclassification Rate of test
confMatrixTst <- predict(myTstmodel, testDf2)
tabTst <- table(confMatrixTst, testDf2$income)
tabTst ### this prints confsion matrix
#### Misclassification Rate
1-sum(diag(tabTst))/sum(tabTst)
predProbTst <- prediction(predProbTst, testDf2$income)
perfEvalTst <- performance(predProbTst, "acc")
plot(perfEvalTst)
maxYvalTst <- which.max(slot(perfEvalTst, "y.values")[[1]])
maxYvalTst
accTst <- slot(perfEvalTst, "y.values")[[1]][maxYvalTst]
accTst ## thus the accuracy of the logistic regression on test data is 84.54%
```
Thus, the accuracy of the logistic regression on test data is 84.54%.
```{r echo = TRUE}
# ROC test
rocTst <- performance(predProbTst, "tpr", "fpr")
plot(rocTst)
#abline(a= 0, b=1)
##### AUC(Area Under Curve) on Test data
aucTst <- performance(predProbTst, "auc")
aucTst<- unlist(slot(aucTst, "y.values"))
aucTst # area under the curve for train is 0.90135
legend(.4,.4, aucTst, title= "AUC")
```
V. Random Forest Code
```
library(ggplot2)
library(scales)
library(plyr)
library(caret)
library(randomForest)
```

```
library(e1071)
library(nnet)
train <- read.csv("train.csv", header = TRUE)
test <- read.csv("test.csv", header = TRUE)
#set test column
train$test <- FALSE
test$test <- TRUE
#merge to remove some variables
merged <- rbind(train, test)
merged <- merged[ , -which(names(merged) %in% c("workclass","hours_per_week", "native_country", "capital_gain",
"capital_loss"))]
# separate train and test sets
train <- subset(merged, merged$test == FALSE)
train$test <- NULL
test <- subset(merged, merged$test == TRUE)
test$test <- NULL
row.names(test) <- 1:nrow(test)
# oversampling dataset
oversampleTrain <- subset(train, train$income == ">50K")
rownames(oversampleTrain) <- 1:nrow(oversampleTrain)
partition <- createDataPartition(y = oversampleTrain$income, times = 1, p = 0.8, list = FALSE)
train.oversampled <- rbind(train, oversampleTrain[partition, ])
rownames(train.oversampled) <- 1:nrow(train.oversampled)
rm(oversampleTrain)
#######
# Normal train set
#######
# running random forest with normal train set
set.seed(1234)
rfmodel <- randomForest(formula = income ~ age + education + education_num + marital_status + occupation +
relationship + race + sex + workclass_category + hours_per_week_category +
global_region + capital_gain_category + capital_loss_category,
data = train)
# training accuracy
mean(predict(rfmodel, newdata = train) == train$income)
#test accuracy
mean(predict(rfmodel, newdata = test) == test$income)
#plot
plot(rfmodel)
#confusion matrix
confusionMatrix(data = predict(rfmodel, newdata = train), reference = train$income, positive = levels(test$income)[2])
#######
# Oversampled
#######
# running random forest with oversampled train set
set.seed(1234)
rfmodeloversample <- randomForest(formula = income ~ age + education + education_num + marital_status + occupation +
relationship + race + sex + workclass_category + hours_per_week_category +
global_region + capital_gain_category + capital_loss_category,
data = train.oversampled)
# training accuracy
mean(predict(rfmodeloversample, newdata = train.oversampled) == train.oversampled$income)
#test accuracy
mean(predict(rfmodeloversample, newdata = test) == test$income)
#plot
plot(rfmodeloversample)
#confusion matrix
confusionMatrix(data = predict(rfmodeloversample, newdata = train.oversampled), reference = train.oversampled$income,
```

```
positive = levels(test$income)[2])
```