Exploratory Data Analysis on Heathcare Data

Taranpreet Singh

Introduction

Thorough data exploration is one of the most important aspects of the predictive modelling. It helps a lot in building insights for feature engineering and feature selection. In this data notebook, I have covered data visualisation part.

In this classification problem hicov is the target variable.

Missing Values

Variable wkhp has around 50% misssing obseravtions.Other varaibales with missing values are income variables, esr schl and povpip.

Pattern in Missing Values

Missing values are for the rows with variable Agep less than 15. For wkhp missing values other than agep less than 15 are for either unemployed or 'not in labour force' esr levels.

Exploration

Loading libraries and importing data

```
#jsonlite for reading json
library(jsonlite)
#tisyverse to load all wickham family packages
library(tidyverse)
#themes for plots
library(ggthemes)

train <- fromJSON("train.json")
# Base theme for plots
thm=theme_tufte()
```

First look at the data

```
#glimpse(train)
#View(train)



# Converting character vars to factors
fac <- lapply(train, class) == "character"
train[, fac] <- lapply(train[, fac], as.factor)
```

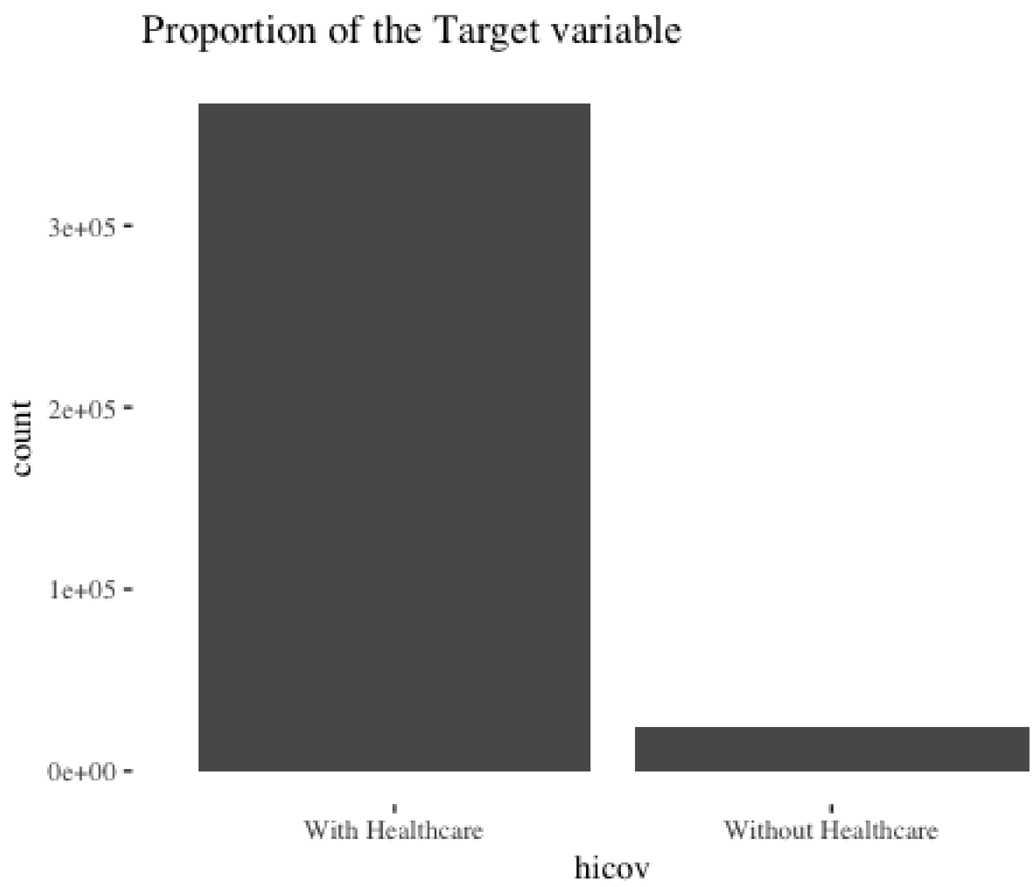
```
#summarising train
#summary(train)

#saving output locally for future record

#capture.output(summary(train), file = "summTrain.txt")
```

Initial obseravations

- 391,282 rows and 20 variables
- id not ordered
- It seems that variables intp, pap, retp have gernerated numbers in them
- Imbalanced binary classifiaction
- Same number of missing rows in income variables

** Target variable**

```
#imbalanced binary classification
ggplot(train,aes(x=hicov))+geom_bar()+thm+ggtitle(" Proportion of the Target
variable")
```




```
round(prop.table(table(train$hicov))*100)
```

```
## 
##    With Healthcare Without Healthcare 
##                  94                 6
```

** Proportion of missing values in Columns**

```
round(colSums(is.na(train))/nrow(train)*100,2)
```

```
##     id     st   puma  hicov    vet   deye   dear    sex   race    mar 
##   0.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00 
##    esr    cit   schl   agep  pincp   intp    pap   retp povpip   wkhp 
##  18.36   0.00   3.06   0.00  17.11  17.11  17.11  17.11   3.35  48.83
```

** Exploring relation of factor variables with target**

```
#comparison of relation with target and training variables

#function to calculate proportion of categorical vars wrt Targget i.e hicov
#calculating proportions from table and rounding to 2 digits
catProp <- function(var) {
  round(prop.table(table(train$hicov,var),2)*100,2)
}

# factor vars

# variation of target with fac vars
lapply(train[,names(Filter(is.factor, train))],catProp)
```

```
## $hicov
##                     var
##                      With Healthcare Without Healthcare
##   With Healthcare                100                  0
##   Without Healthcare               0                100
## 
## $vet
##                     var
##                      Not Veteran Veteran
##   With Healthcare          93.42   98.35
##   Without Healthcare        6.58    1.65
## 
## $deye
##                     var
##                         No   Yes
##   With Healthcare    93.69 94.75
##   Without Healthcare  6.31  5.25
## 
## $dear
##                     var
##                         No   Yes
##   With Healthcare    93.57 97.31
##   Without Healthcare  6.43  2.69
```

```
##
## $sex
##                       var
##                        Female  Male
##    With Healthcare      94.74 92.66
##    Without Healthcare    5.26  7.34
##
## $race
##                       var
##                        Alaska Native alone Amer. Indian + Alaska Nat. tribes
##    With Healthcare                   90.27                             89.09
##    Without Healthcare                 9.73                             10.91
##                       var
##                        Amer. Indian alone Asian alone
##    With Healthcare                  86.11       95.58
##    Without Healthcare               13.89        4.42
##                       var
##                        Black or African Amer. alone
##    With Healthcare                            93.24
##    Without Healthcare                          6.76
##                       var
##                        Nat. Hawaiian + Other Pac. Isl. Some other race alone
##    With Healthcare                               91.51                 86.26
##    Without Healthcare                             8.49                 13.74
##                       var
##                        Two or more White alone
##    With Healthcare          94.76       94.58
##    Without Healthcare        5.24        5.42
##
## $mar
##                       var
##                        Divorced Married Never Married Separated Widowed
##    With Healthcare        93.05   94.87         92.53     87.62   97.62
##    Without Healthcare      6.95    5.13          7.47     12.38    2.38
##
## $esr
##                       var
##                        Employed Not in labor force Unemployed
##    With Healthcare        92.86              93.73      84.68
##    Without Healthcare      7.14               6.27      15.32
##
## $cit
##                       var
##                        Citizen Not citizen
##    With Healthcare       95.50       78.16
##    Without Healthcare     4.50       21.84
##
## $schl
##                       var
##                        Grad. Degree HS Degree Less than HS Undergrad. Degree
```

```
##   With Healthcare              98.25     92.76         91.56                96.73
##   Without Healthcare            1.75      7.24          8.44                 3.27
```

```
#saving output
#capture.output(lapply(train[,names(Filter(is.factor, train))],catProp),file=
"catVarsProp.txt")
```

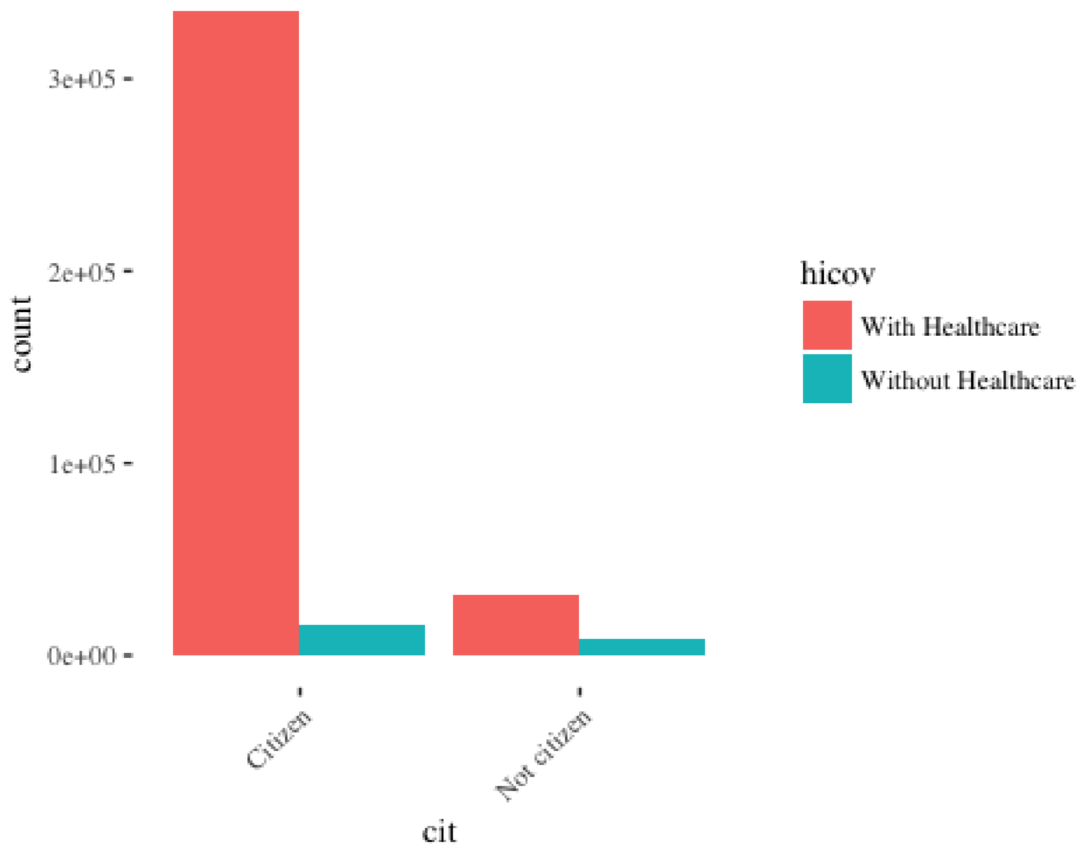
With this table we can clearly see the effect of different classes of the factor variables on the Target variable. For instance, Veterans have very high probability of getting an health coverage

Variables having significant difference with target variable are vet, cit, race.

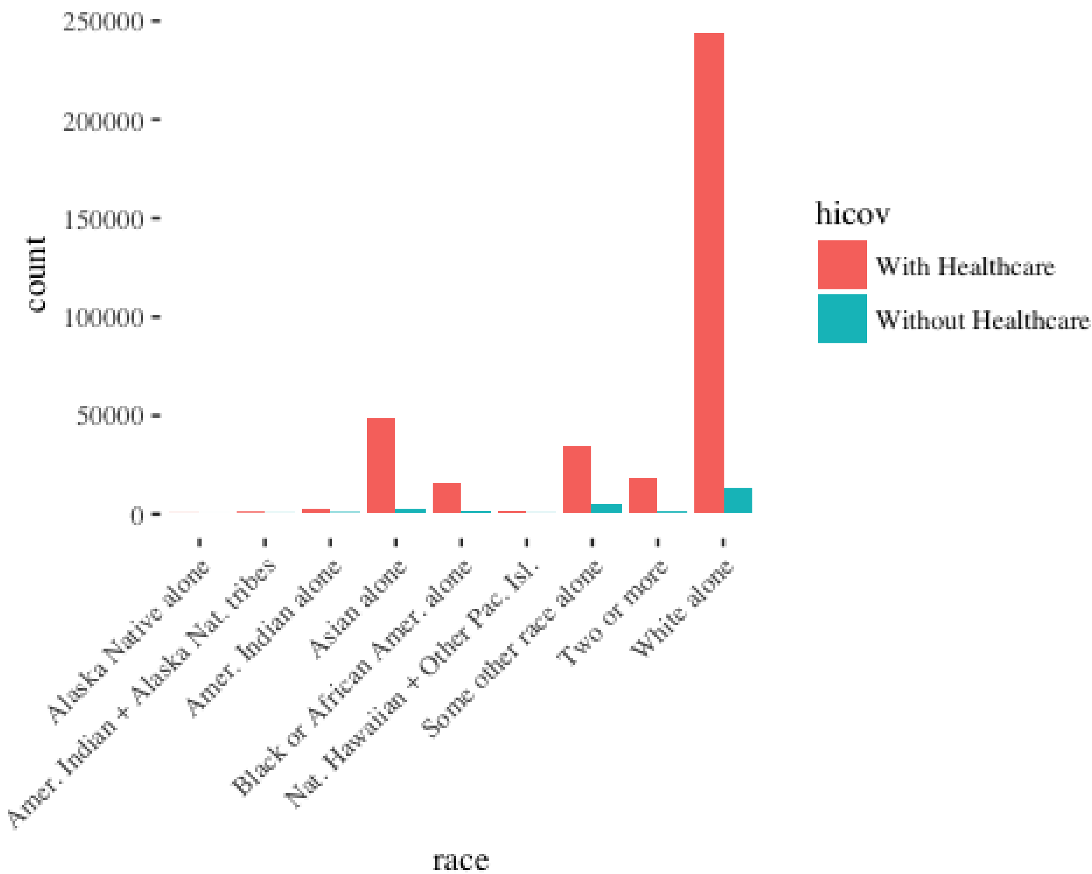
Citizenship has most prominent effect

Plots

```
catPlot <- function(var) {
  ggplot(train,aes_string(x=deparse(substitute(var)),fill="hicov"))+geom_bar(
position="dodge")+thm+
    theme(axis.text.x=element_text(angle=45,hjust=1))
}

# plot of target with factor variables
catPlot(cit)
```



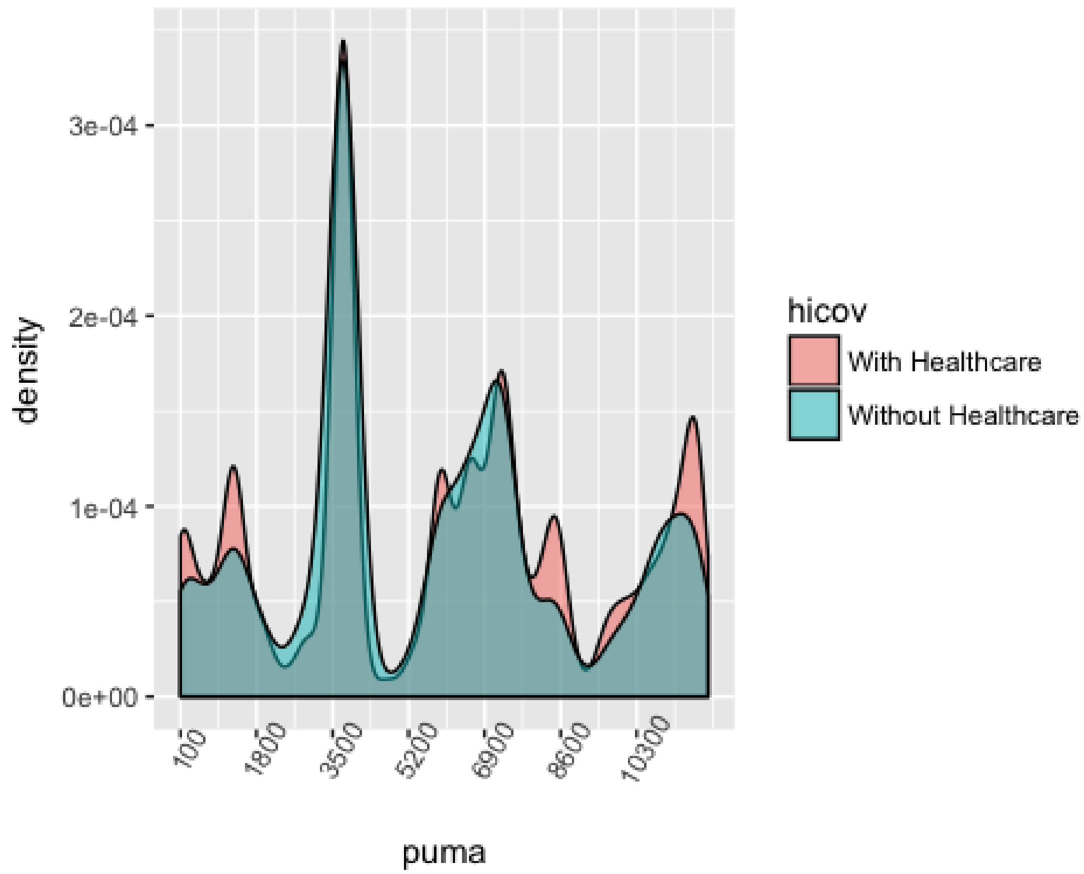

```
catPlot(race)
```



Puma and state should be factor variables, state has only 3 levels but puma has more than 300 levels. For modelling Puma can be encoded as factor but that will increase number of variables a lot. Another approach can be to cluster it or create buckets. I have trained model with puma classes as independent variables and then selected important variables from that.

Distribution of Puma with target variable

```
ggplot(train,aes(x=puma,fill=hicov))+geom_density(alpha=0.5)+
  theme(axis.text.x=element_text(angle=60))+
  scale_x_continuous(breaks=round(seq(min(train$puma), max(train$puma), by =
1700),1))
```



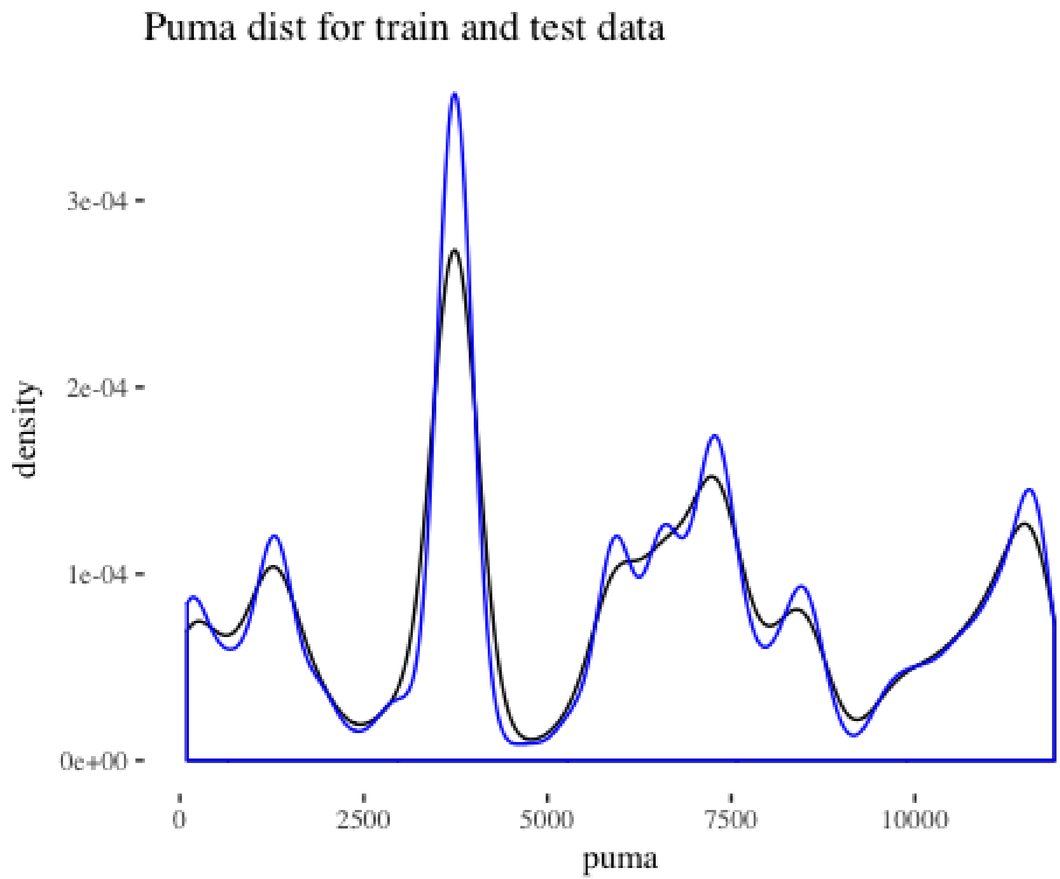
My approach for Handling variables such as age and puma

By zooming on x axis we can find the values to create buckets, I use this approach extensively in my analysis. This helps in creating buckets for variables such as age.

Since we have similar distribution of variables for train and test set. For this challenge this approach might help.

Loading test set

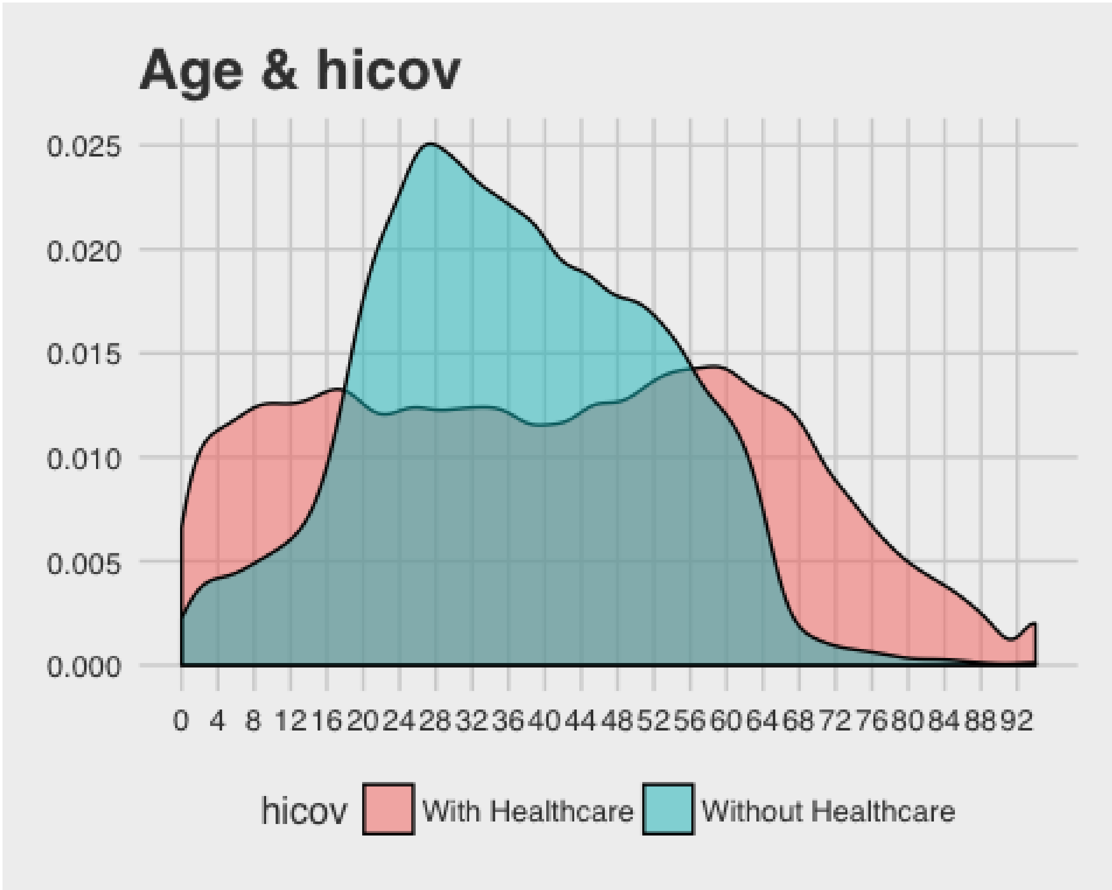
```
test <- fromJSON("test.json")
# 97405 obs

#glimpse(test)

#converting char to factors

test[, fac] <- lapply(test[, fac], as.factor)
#round(colSums(is.na(test))/nrow(test)*100,2)

ggplot()+geom_density(data=test,aes(x=puma))+
  geom_density(data=train,aes(x=puma),color='blue')+
  thm+ggtitle("Puma dist for train and test data")
```



Age Variable

Using similar method we can find buckets for income and age. Sometimes models work better with the bucketed data than continous. Moreover, for very large data set creating sparse dataframe saves memory.

```
ggplot(train,aes(x=agep,fill=hicov))+geom_density(alpha=0.5)+ggtitle("Age & h
icov")+
  scale_x_continuous(breaks=round(seq(min(train$agep), max(train$agep), by =
4),1))+
  theme_fivethirtyeight()
```

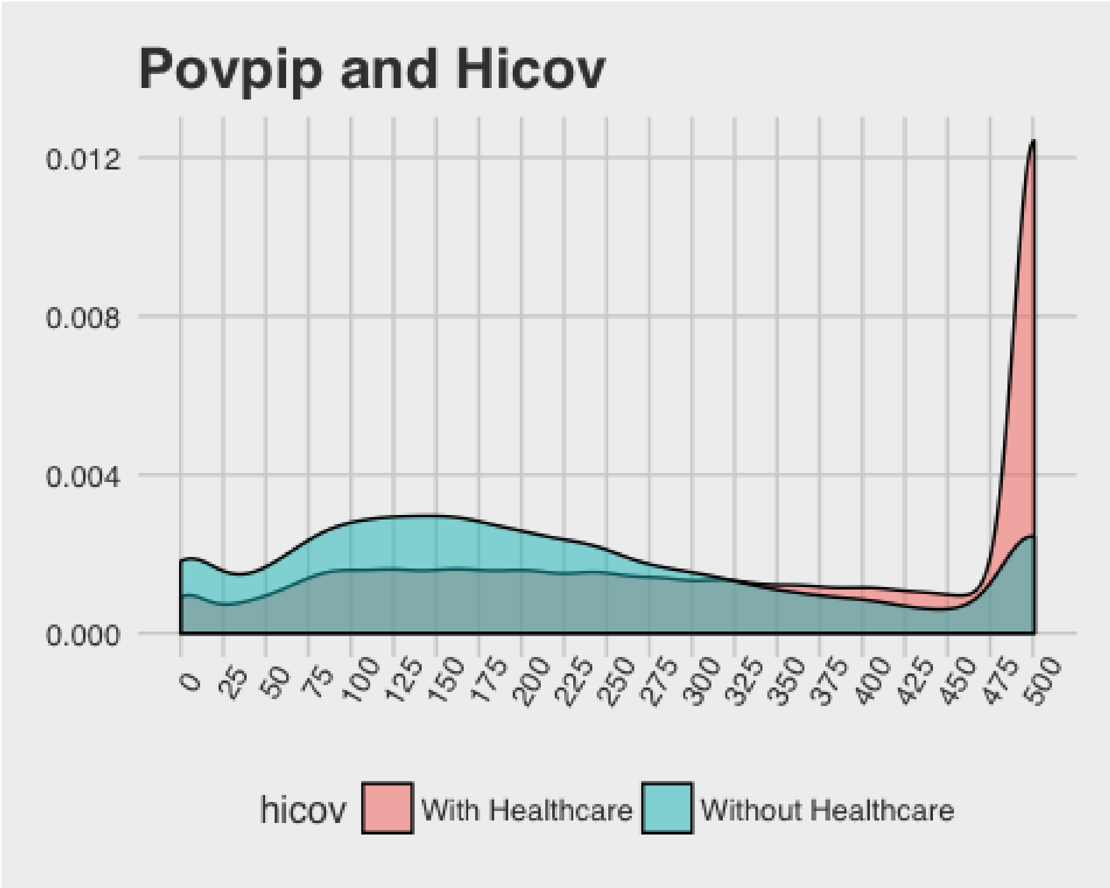



```
#ggplot()+geom_density(data=test,aes(x=agep))+
 # geom_density(data=train,aes(x=agep),color='blue')+
  #thm+ggtitle("Agep dist for train and test data")
```

We can see that the range for people with Health cover is less and more skewed. Using this grid we can find the boundaries for the age variable buckets. The distribution for test and train is also same.

Povpip variable

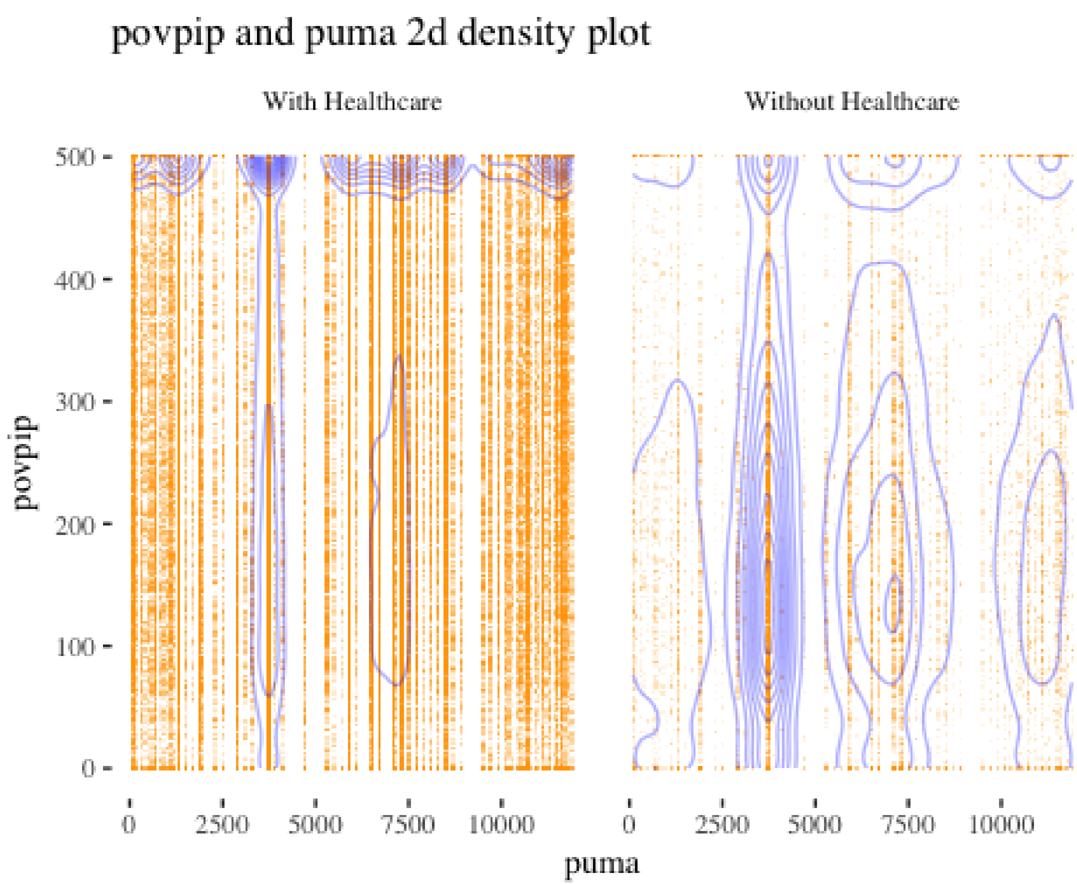
```
ggplot(train,aes(x=povpip,fill=hicov))+geom_density(alpha=0.5)+
  scale_x_continuous(breaks=round(seq(0, 500, by = 25),1))+
  theme_fivethirtyeight()+theme(axis.text.x=element_text(angle=60))+
  ggtitle("Povpip and Hicov")
```



As this plot shows high values of Pivpop have very high rate of health coverage. Since this variable has missing values, so this variable needs careful imputation. Mode for this variable is 501. To impute it we can use value of 330 as at that value both the target classes have same distribution.

For imputation of povpip, I have also tried to find the relation of povpip and puma with the assumption that povpip should be similar for puma values but I could not find any strong relation.

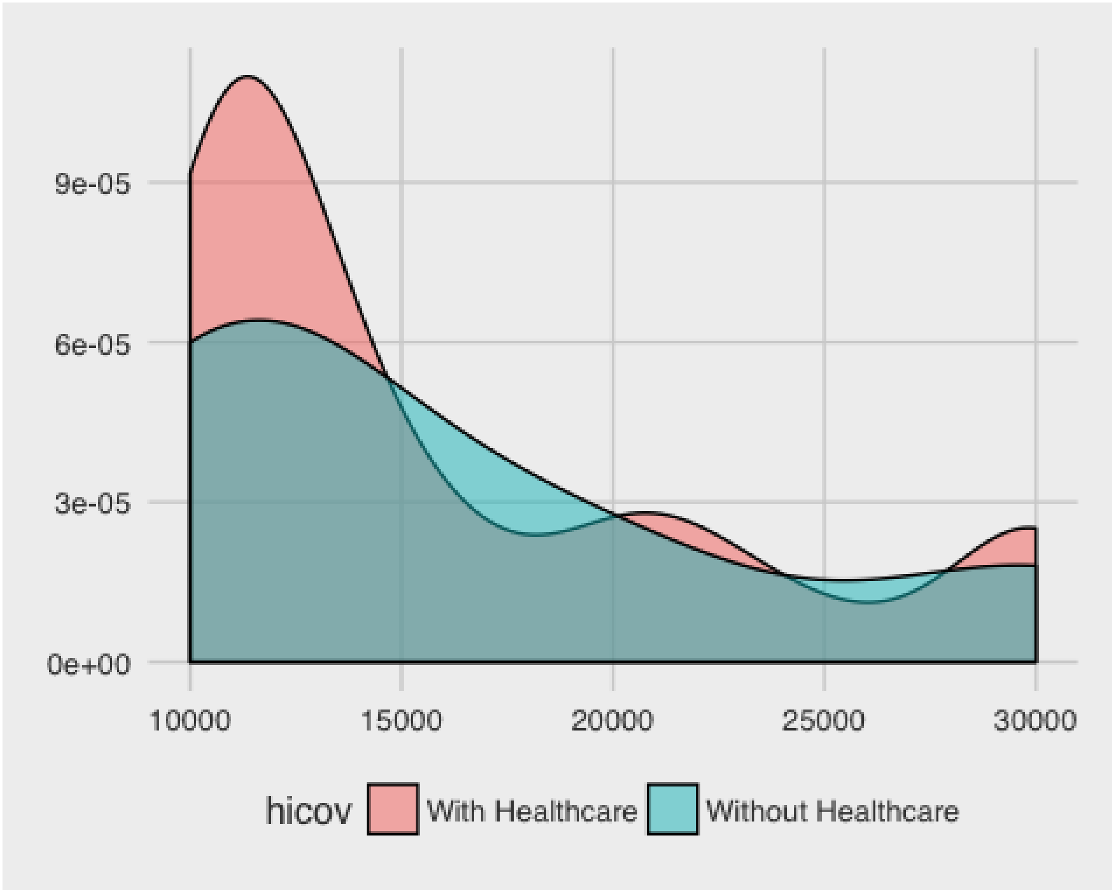
```
ggplot(train,aes(x=puma,y=povpip))+ geom_point(alpha=0.1,shape='.',color='ora
nge')+
  geom_density2d(color='blue',alpha=0.3)+thm+facet_grid(~hicov)+
  ggtitle("povpip and puma 2d density plot")
```



Exploring Income variables

I have created extensive graphs to see the relationship and found that these variables are important for the target variable. Though the actual usefulness can be found after modelling. We can also see the difference in the mean for the target variable.

```
train %>% select(intp,pap,retp,pincp,hicov,wkhp) %>%
  filter(!is.na(intp)) %>%
  group_by(hicov) %>%
  summarise(mean(intp),mean(pap),mean(retp),mean(pincp))

## # A tibble: 2 x 5
##                hicov `mean(intp)` `mean(pap)` `mean(retp)` `mean(pincp)`
##               <fctr>        <dbl>       <dbl>        <dbl>         <dbl>
## 1    With Healthcare    3044.1290    63.95369    2928.1522      44540.46
## 2 Without Healthcare     498.9271    50.27221     237.7802      19195.83

train %>% filter(pap>=10000 ) %>% ggplot(aes(x=pap,fill=hicov))+geom_density(
alpha=0.5)+
  #scale_x_continuous(breaks=round(seq(0,100, by =5),1))+
  theme_fivethirtyeight()
```



- The detailed analysis can be found in the EDA file attached alongwith.