Adult Census Income

Hany Awadalla

2019 M06 4

Contents

The Adult Cenus Income.

This data was extracted from the 1994 Census bureau database by Ronny Kohavi and Barry Becker (Data Mining and Visualization, Silicon Graphics). A set of reasonably clean records was extracted using the following conditions: ((AAGE>16) && (AGI>100) && (AFNLWGT>1) && (HRSWK>0)).

Our Target

The prediction to determine whether a person makes over $50K a year.

- First we will have and idea about the data.

```
##   age workclass fnlwgt    education education.num marital.status
## 1  90         ?  77053      HS-grad             9        Widowed
## 2  82   Private 132870      HS-grad             9        Widowed
## 3  66         ? 186061 Some-college            10        Widowed
## 4  54   Private 140359      7th-8th             4       Divorced
## 5  41   Private 264663 Some-college            10      Separated
## 6  34   Private 216864      HS-grad             9       Divorced
##          occupation  relationship  race    sex capital.gain capital.loss
## 1                 ? Not-in-family White Female            0         4356
## 2   Exec-managerial Not-in-family White Female            0         4356
## 3                 ?     Unmarried Black Female            0         4356
## 4 Machine-op-inspct     Unmarried White Female            0         3900
## 5    Prof-specialty     Own-child White Female            0         3900
## 6     Other-service     Unmarried White Female            0         3770
##   hours.per.week native.country income
## 1             40  United-States  <=50K
## 2             18  United-States  <=50K
## 3             40  United-States  <=50K
## 4             40  United-States  <=50K
## 5             40  United-States  <=50K
## 6             45  United-States  <=50K
```

The data has total of 32561 rows and 15 columns.

___ For more data details:-

Attributes:

50K, <=50K

age: continuous

workclass: Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked

fnlwgt: continuous

education: Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool

education-num: continuous

marital-status: Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse

occupation: Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces relationship: Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried

race: White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black

sex: Female, Male

capital-gain: continuous

capital-loss: continuous

hours-per-week: continuous

native-country: United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands

– We need to check whether the person earn >50k or not

income	Number
<=50K	24720
>50K	7841

As we can see from the table the number of the people, they earn more than 50K is less than the people they earn less.

The data has some row with "?" or and empty values.

i will convert those values to (NA)s which will be easy to deal with later on .

– Now we have 2399 values to deal with.

They are distributed as shown.

```
## $age
##
## FALSE
## 32561
##
## $workclass
##
## FALSE  TRUE
## 30725  1836
##
## $fnlwgt
##
## FALSE
## 32561
##
## $education
##
## FALSE
## 32561
##
## $education.num
##
```

```
## FALSE
## 32561
##
## $marital.status
##
## FALSE
## 32561
##
## $occupation
##
## FALSE  TRUE
## 30718  1843
##
## $relationship
##
## FALSE
## 32561
##
## $race
##
## FALSE
## 32561
##
## $sex
##
## FALSE
## 32561
##
## $capital.gain
##
## FALSE
## 32561
##
## $capital.loss
##
## FALSE
## 32561
##
## $hours.per.week
##
## FALSE
## 32561
##
## $native.country
##
## FALSE  TRUE
## 31978   583
##
## $income
##
## FALSE
## 32561
```

- Most of the missing data can't be recoverd by using median or mean because they are categorical

variables
- their is shared missing data between the same columns such as *workclass* and *Occupation*.
- 1836 is Number of shared missing data between the two variables which makes it harder to recover some of the missing values.
- No missing data will be allowed in our machine learning algorithm
- the best solution is to omit the missing values and start our machine learning algorithm.

– We need to check all the variale class

- After changing the classes of the colums to apply our machine learning modules.

x	
age	numeric
workclass	factor
fnlwgt	integer
education	factor
education.num	integer
marital.status	factor
occupation	factor
relationship	factor
race	factor
sex	factor
capital.gain	integer
capital.loss	integer
hours.per.week	numeric
native.country	factor
income	factor

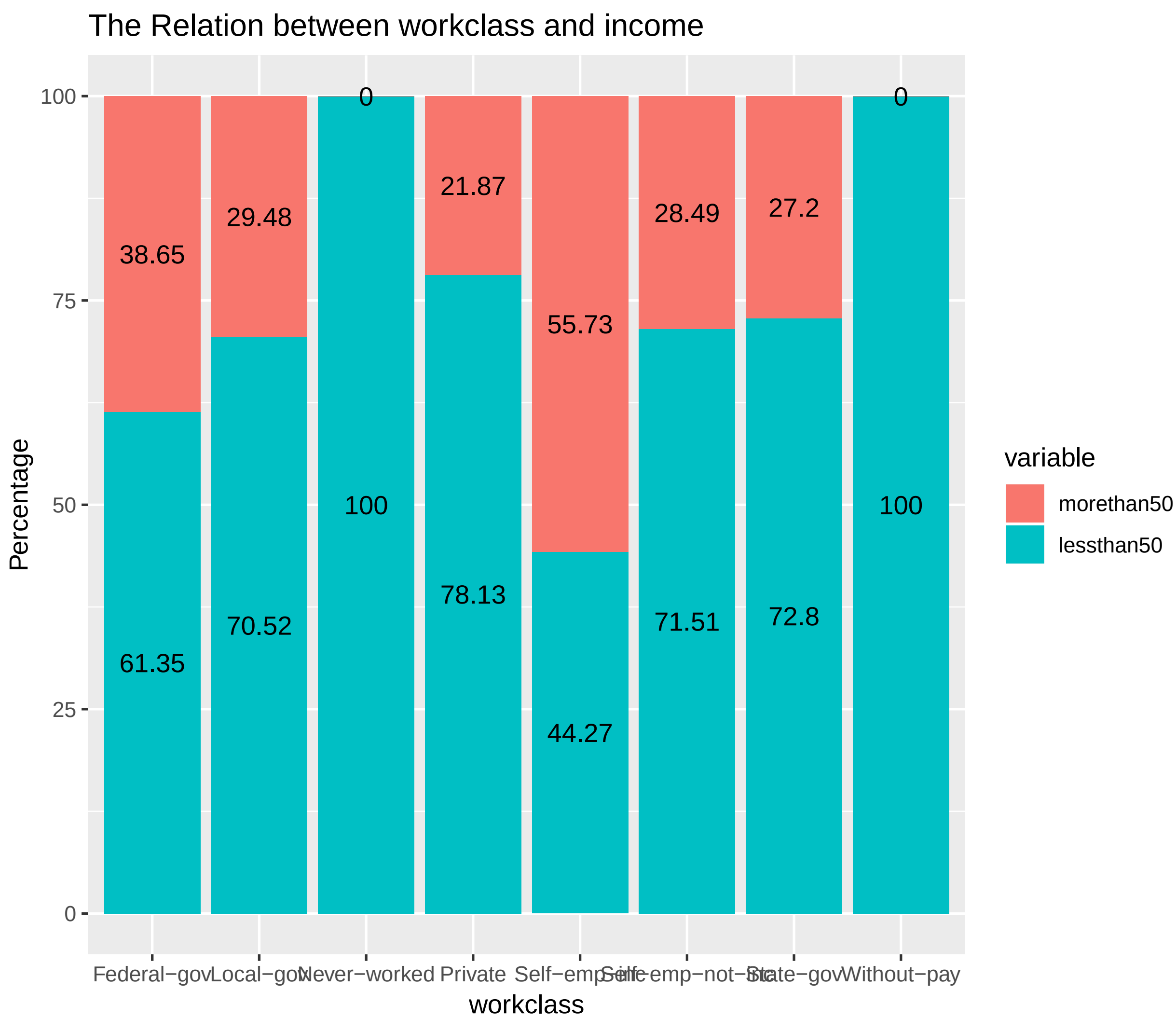
first step is to check how important is the independant variables to our data.

we will start by visualizing each variable and how that could affect the income.

1.

–workclass–

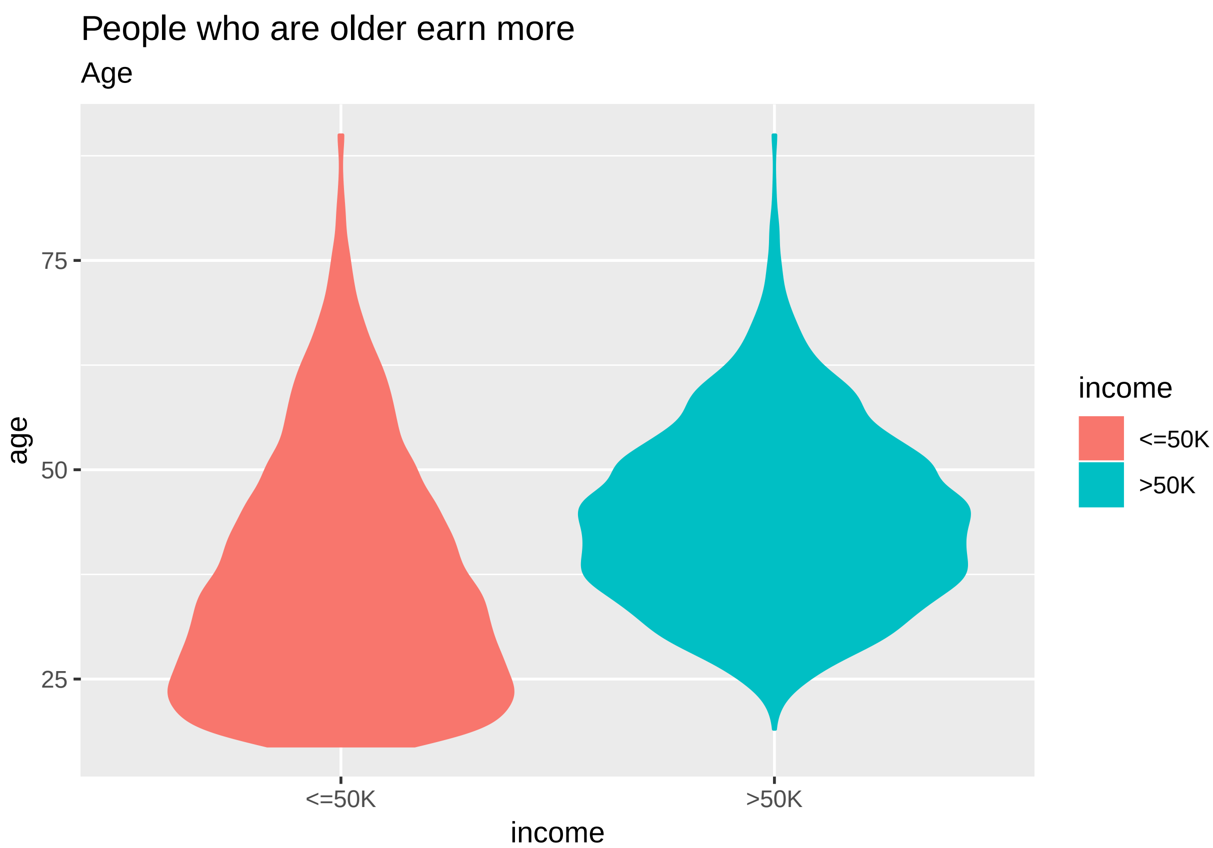
```
workclass_variable <- cenus%>% filter(!is.na(workclass)) %>% group_by(workclass) %>%
  summarise( morethan50 = sum(income == ">50K")/ n()*100, lessthan50 = sum(income == "<=50K")/n()*100) %
workclass_variable <- reshape::melt(workclass_variable, id= "workclass")

ggplot(workclass_variable, aes(x=workclass, y= value, fill = variable, label = round(value,2))) +
  geom_bar(stat="identity") +
  geom_text(position = position_stack(vjust=0.5))+
  labs(x= "workclass", y = "Percentage", title = "The Relation between workclass and income")
```



2.

–age–

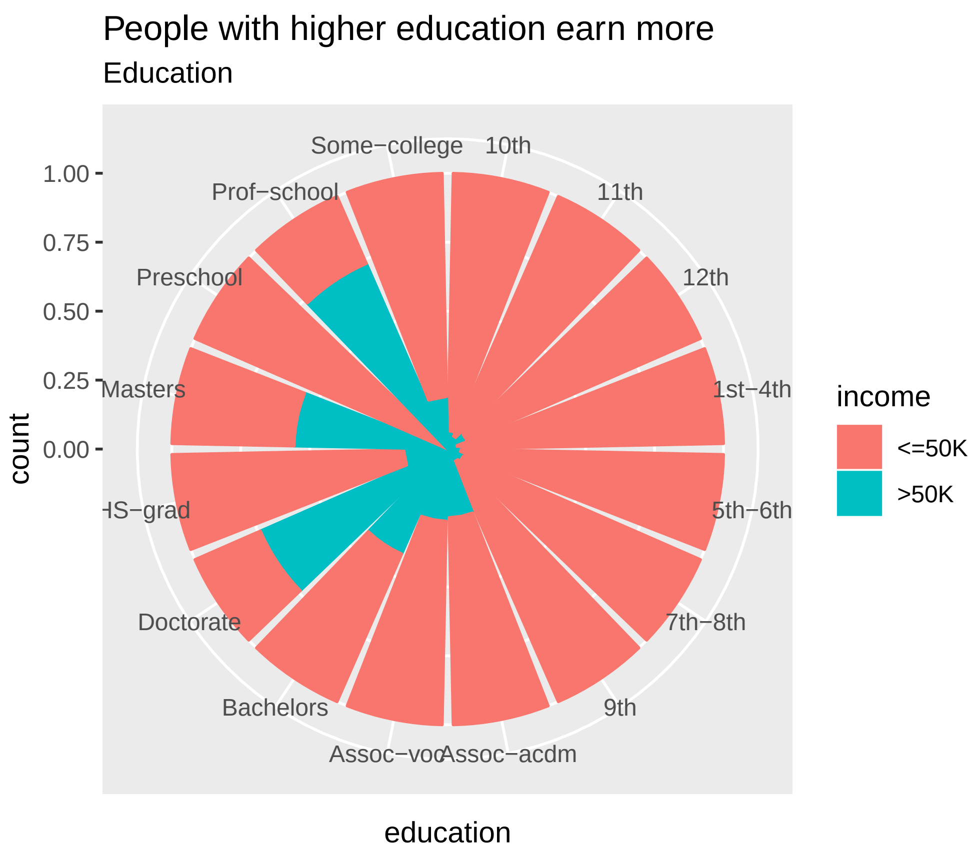
```
ggplot(cenus, aes(x=income, y = age, color = income, fill = income)) +
  geom_violin () +
  labs( title = "People who are older earn more",
        subtitle = "Age")
```



3.

–education–

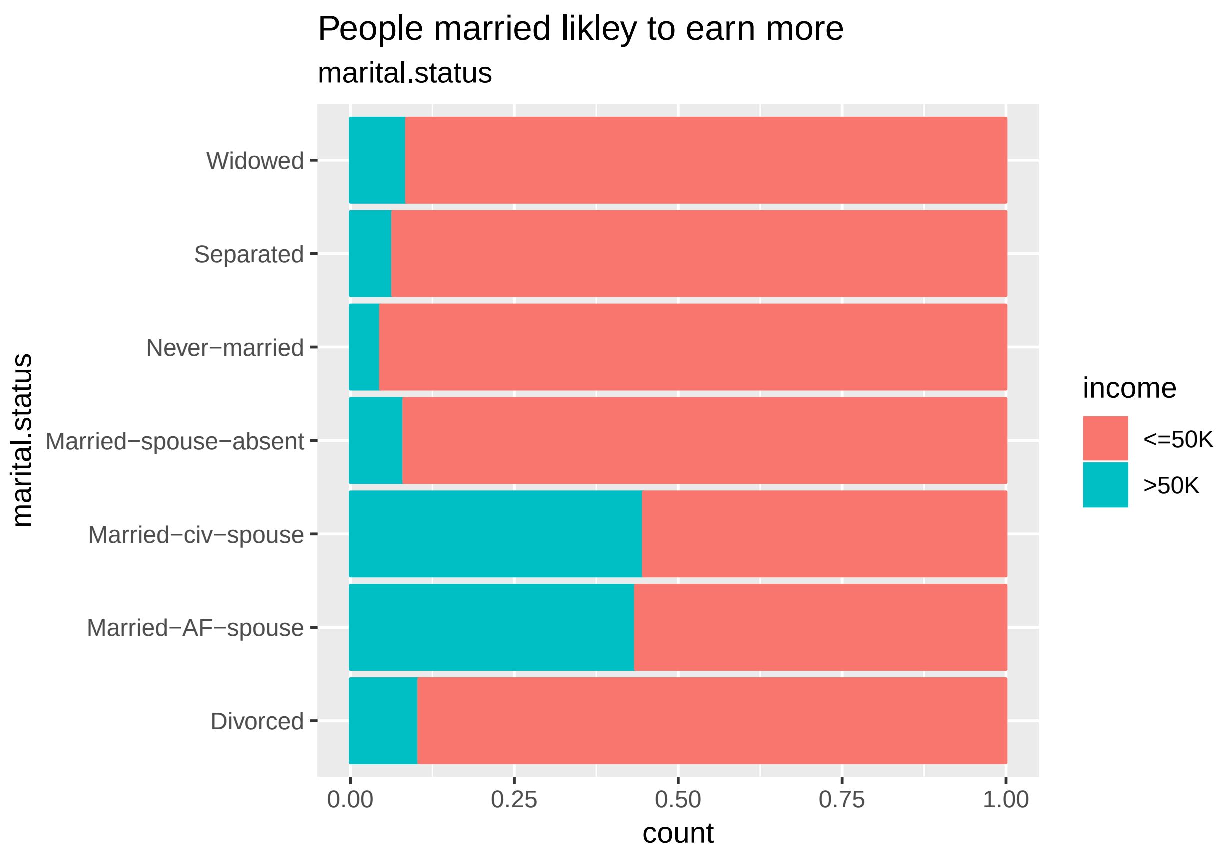
```
ggplot(cenus, aes(x=education, color = income, fill = income)) +
  geom_bar(position = "fill")+
  coord_polar() +
  labs( title = "People with higher education earn more",
        subtitle = "Education")
```



4.

–marital.status–

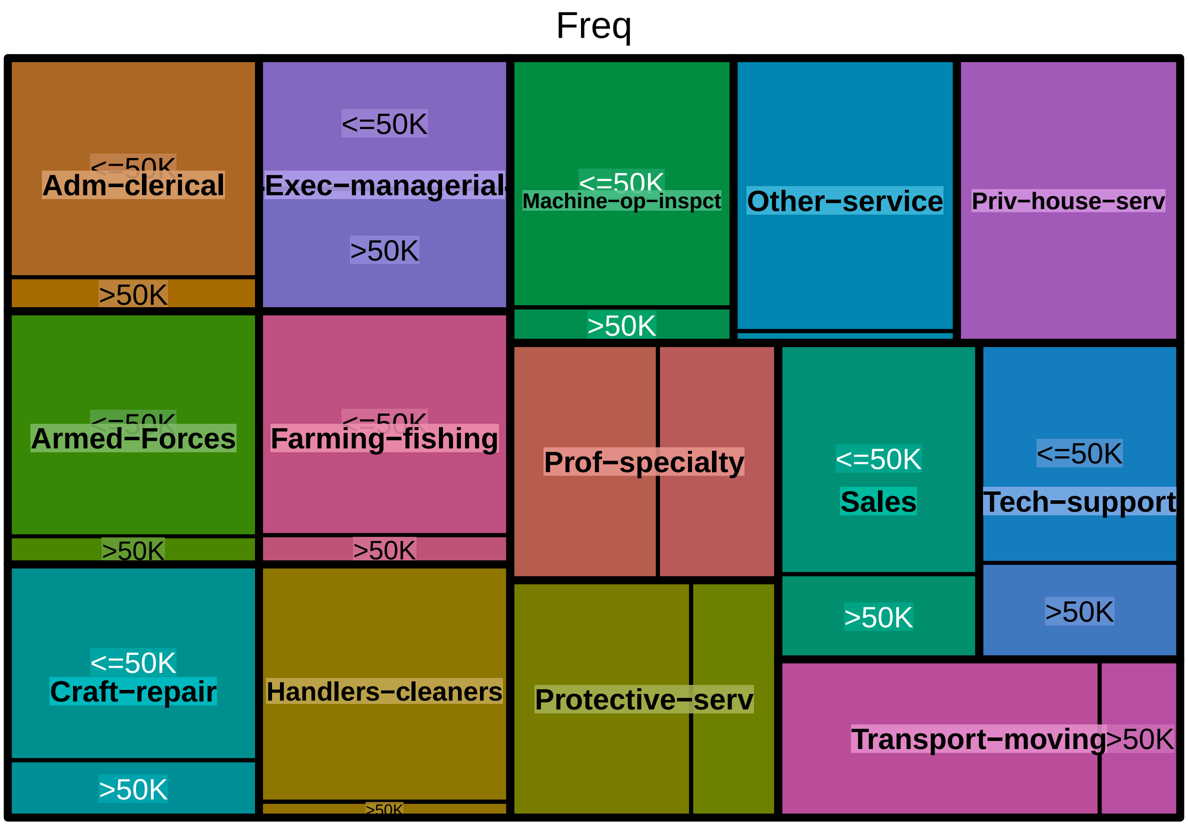
```
ggplot(cenus, aes(x=marital.status, color = income, fill = income)) +
  geom_bar(position = "fill")+
  coord_flip() +
  labs( title = "People married likley to earn more",
        subtitle = "marital.status")
```



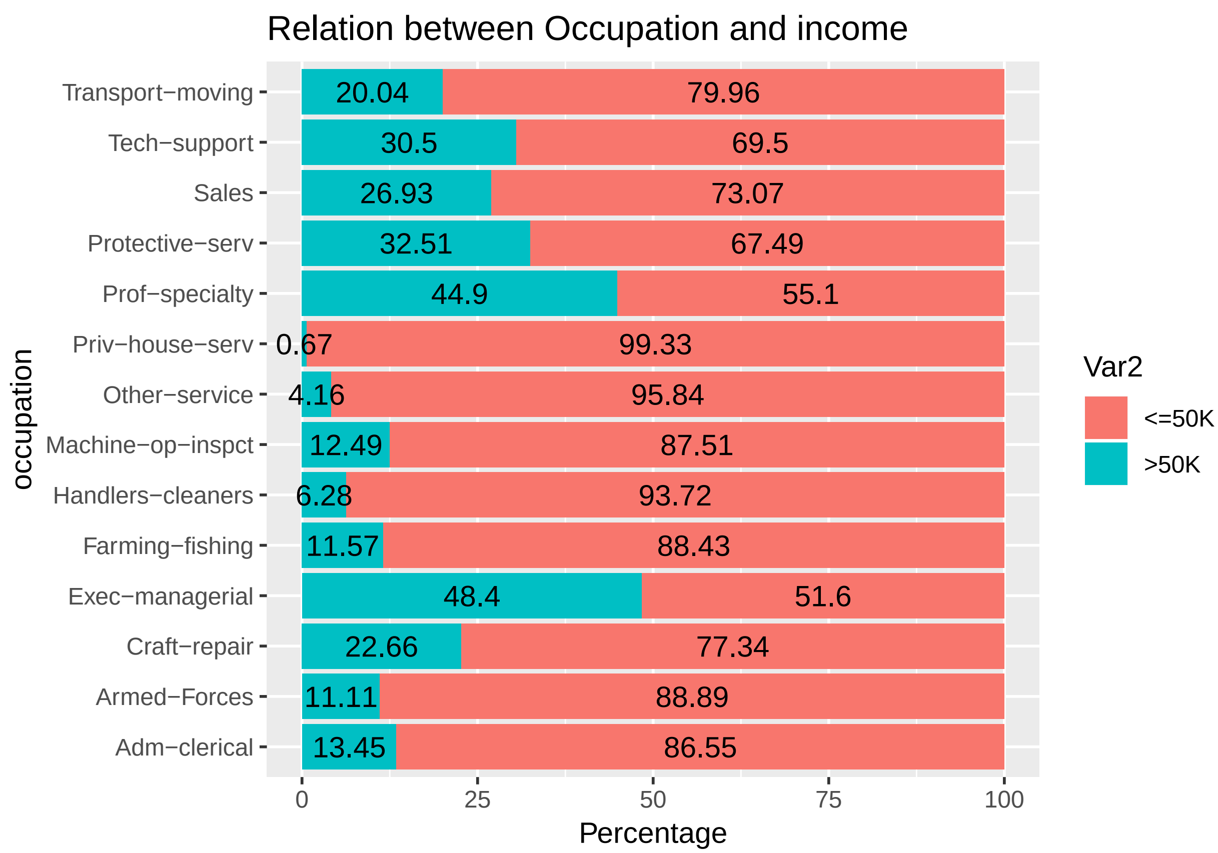
5.

–occupation–

```
occupation_variable <- as.data.frame(prop.table(table(cenus$occupation, cenus$income), 1) * 100) %>% mu
treemap::treemap(occupation_variable, c("Var1", "Var2","Freq"), vSize = "Freq", type="index")
```



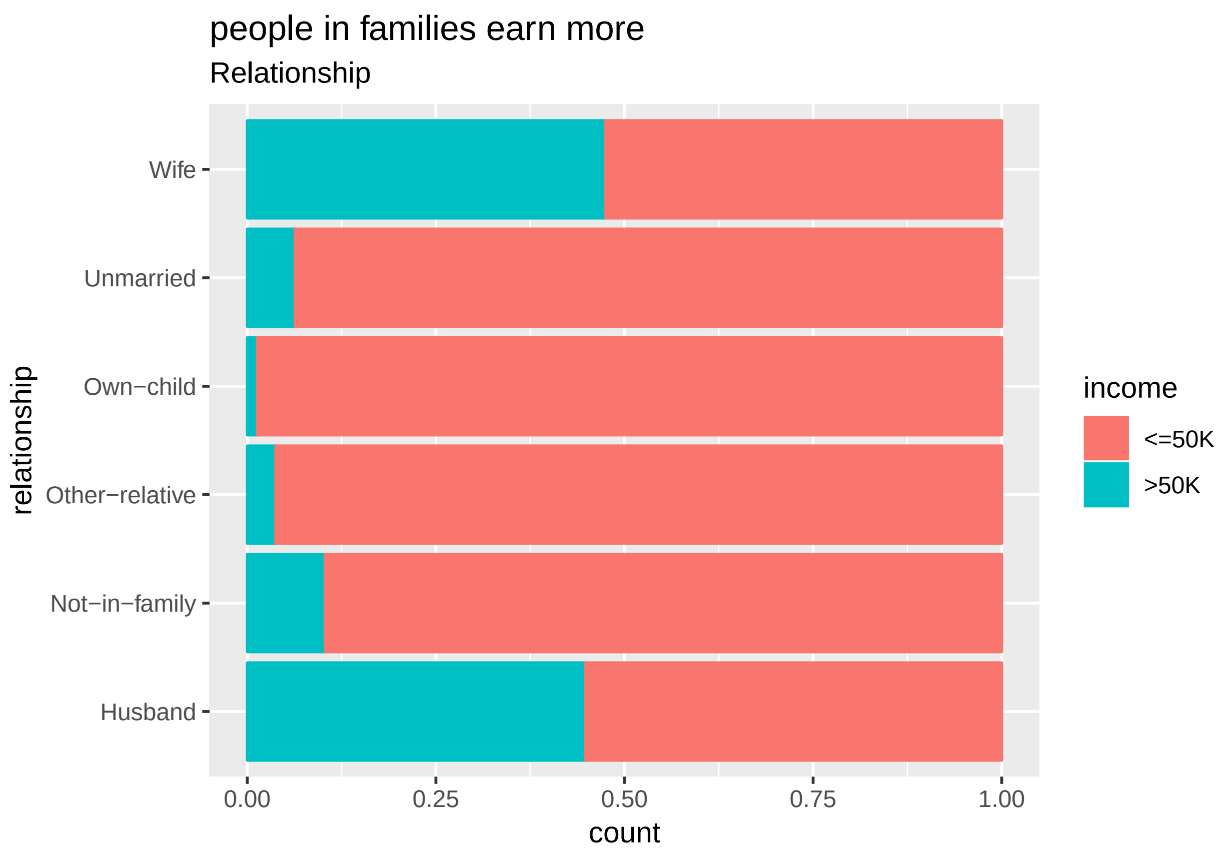

```r
ggplot(occupation_variable, aes(x=Var1, y= Freq, fill = Var2, label = Freq)) +
  geom_bar(stat="identity") +
  coord_flip()+
  geom_text(position = position_stack(vjust=0.5))+
  labs(x= "occupation", y = "Percentage", title = "Relation between Occupation and income")
```



6.

–relationship–

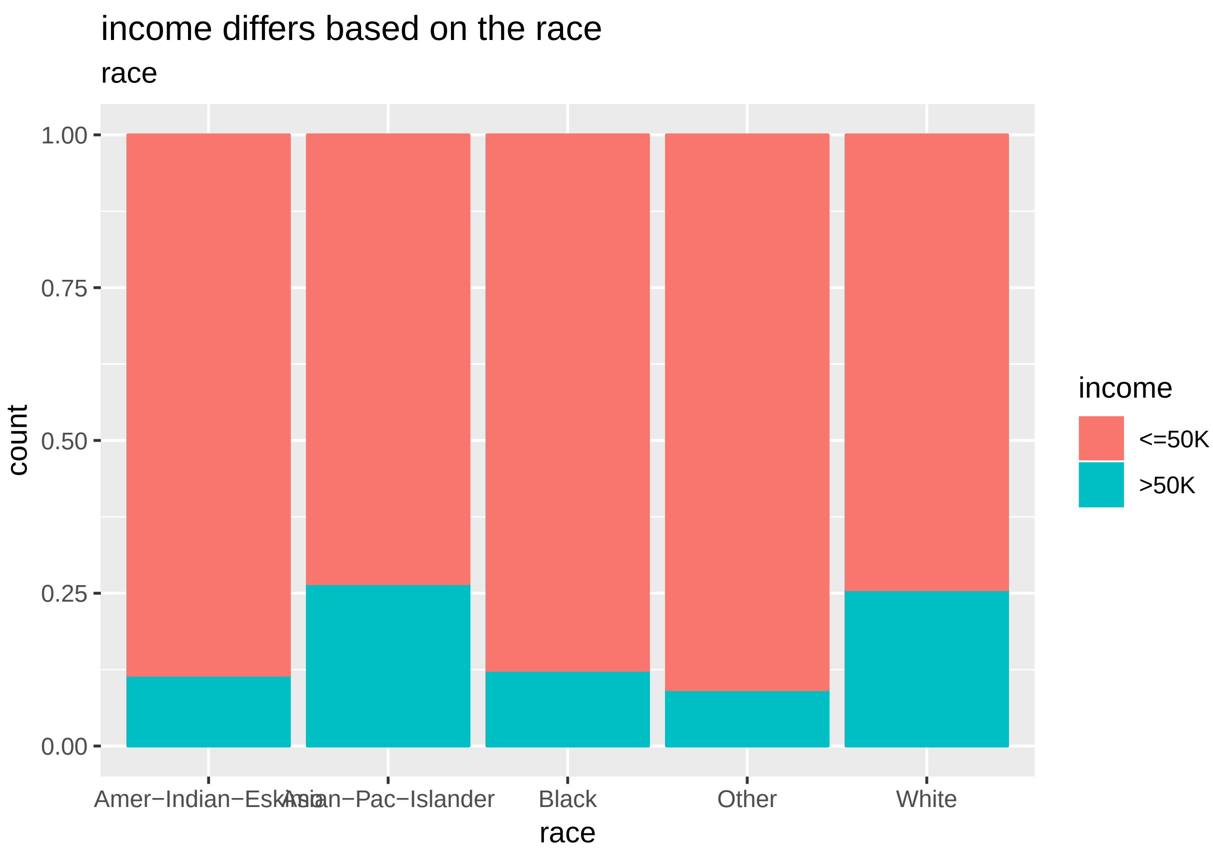
```
ggplot(cenus, aes(x=relationship, color = income, fill = income)) +
  geom_bar(position = "fill")+
  coord_flip() +
  labs( title = "people in families earn more ",
        subtitle = "Relationship")
```



7.

–Race–

```
ggplot(cenus, aes(x=race, color = income, fill = income)) +
  geom_bar(position = "fill")+
  labs( title = "income differs based on the race ",
        subtitle = "race")
```

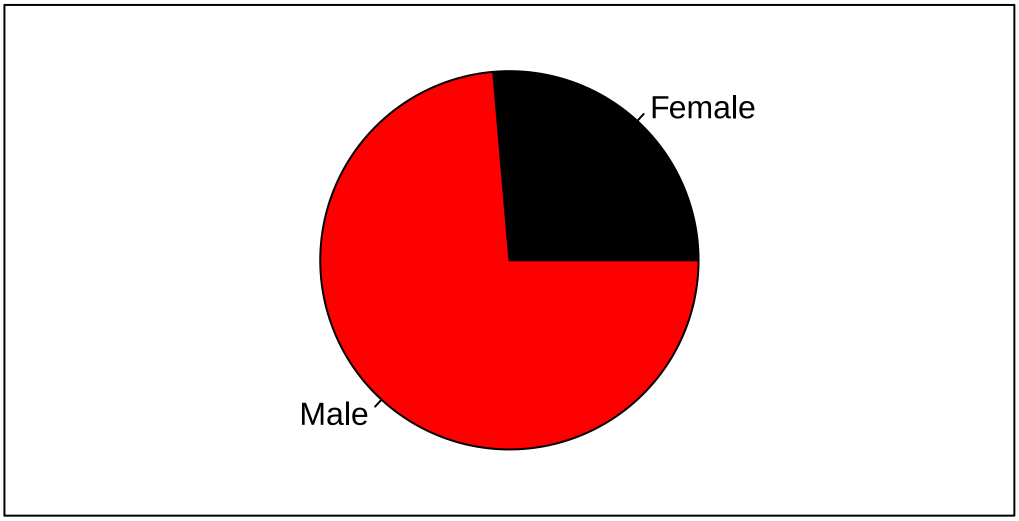


8.

–sex–

```
sex_var <- cenus %>% group_by(sex) %>% summarise(total = sum(income == ">50K")/n()*100)
pie(sex_var$total, labels = sex_var$sex, main = "Male earn >50k more than Female", col = sex_var$sex)
box()
```

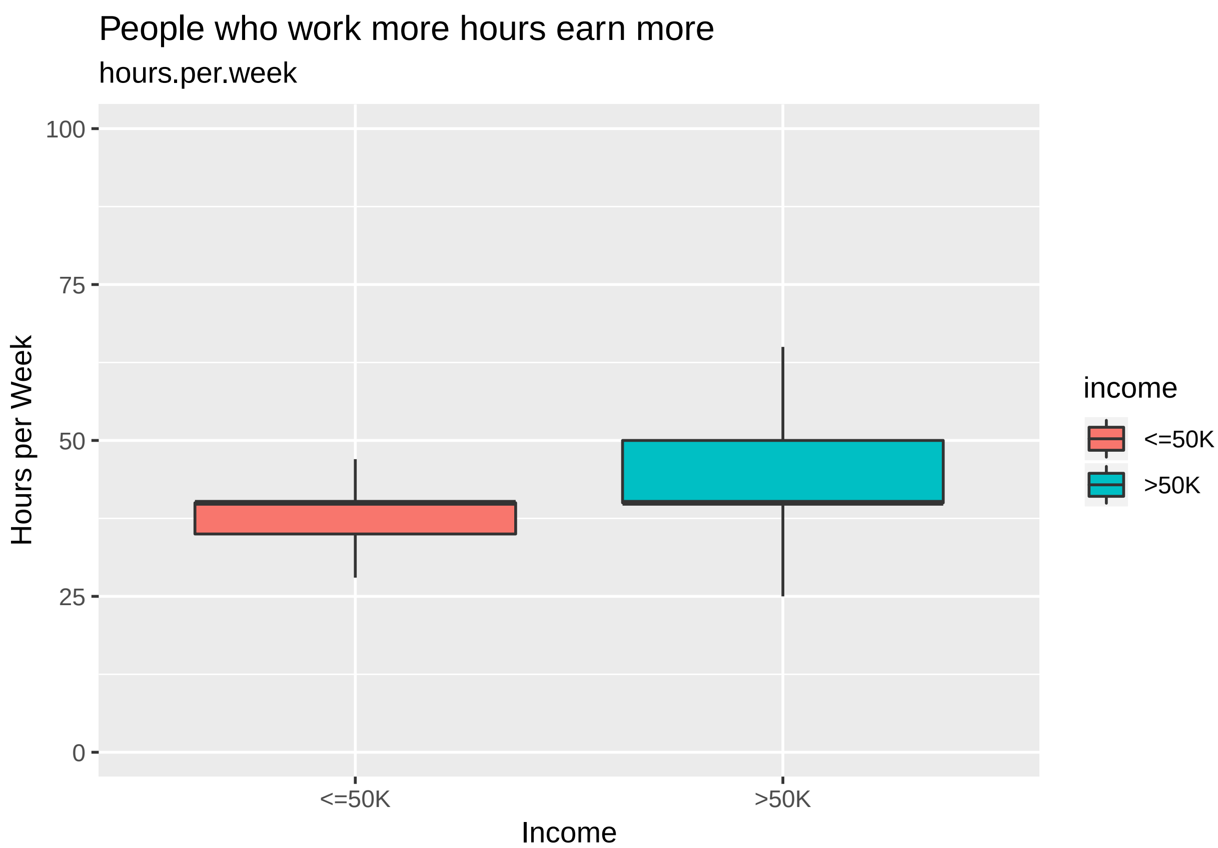
Male earn >50k more than Female



9.

–hours.per.week–

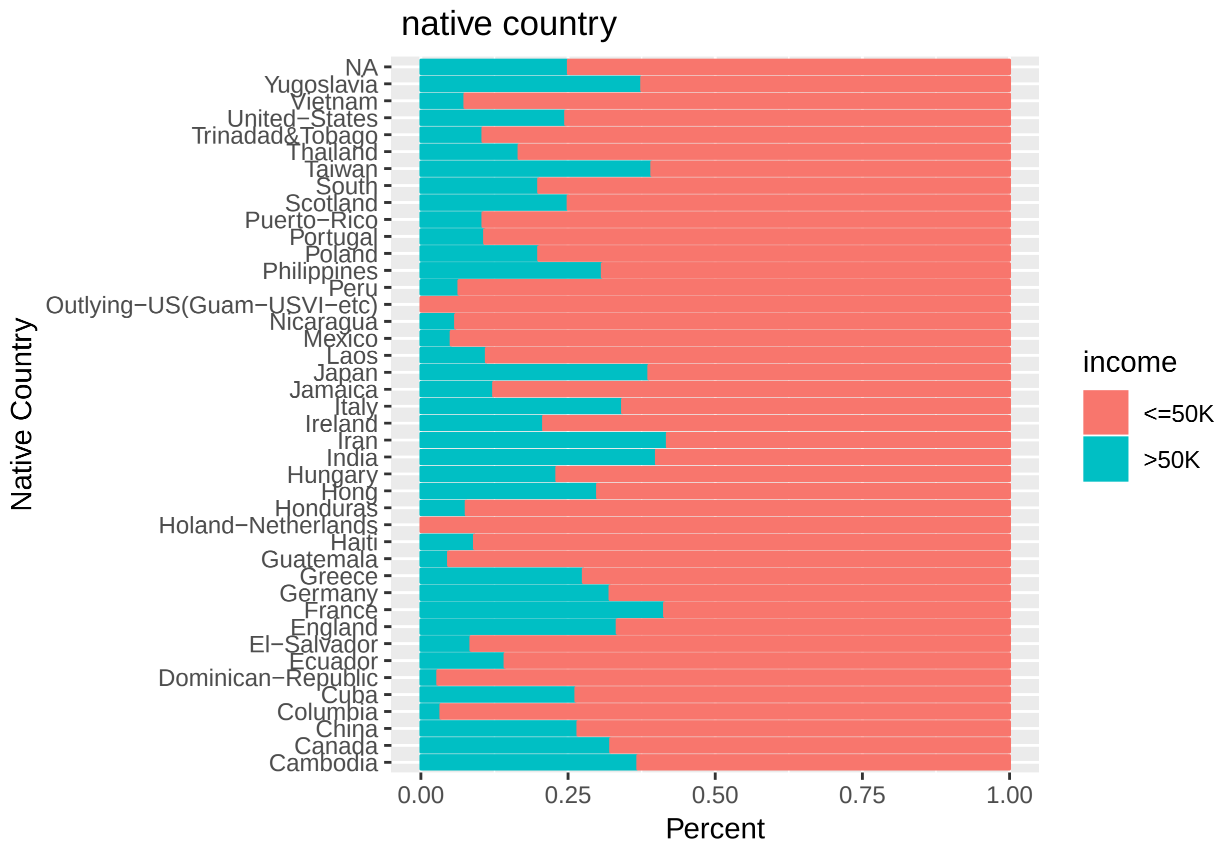
```
ggplot(cenus, aes(x = income, y = hours.per.week, fill = income)) +
  geom_boxplot(outlier.shape = NA ) +
  labs(x = "Income", y = "Hours per Week", title = "People who work more hours earn more",
       subtitle = "hours.per.week")
```



10.

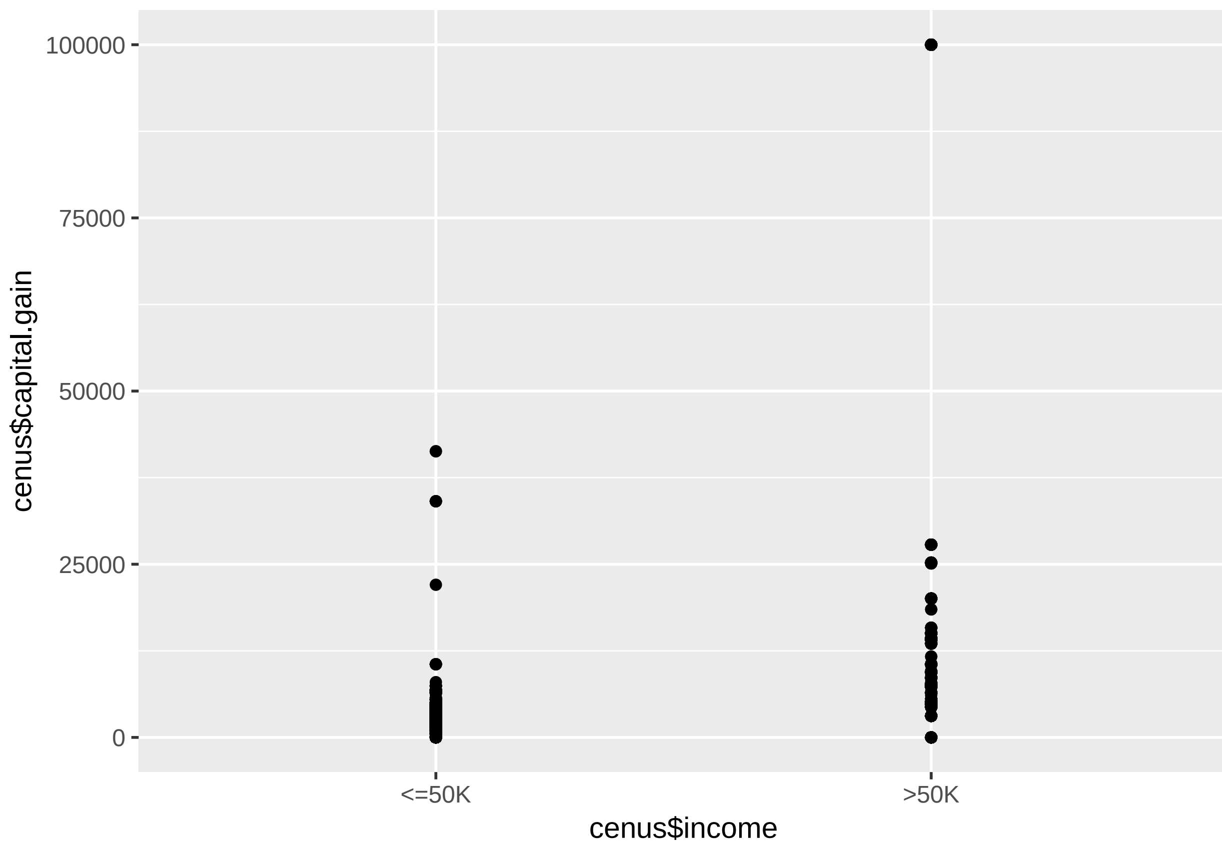
–native.country–

```r
ggplot(cenus, aes(x = native.country, fill = income, color = income)) +
  geom_bar( width = 0.8, position = "fill") +
  coord_flip() +
  labs(x = "Native Country", y = "Percent", title = " native country")
```

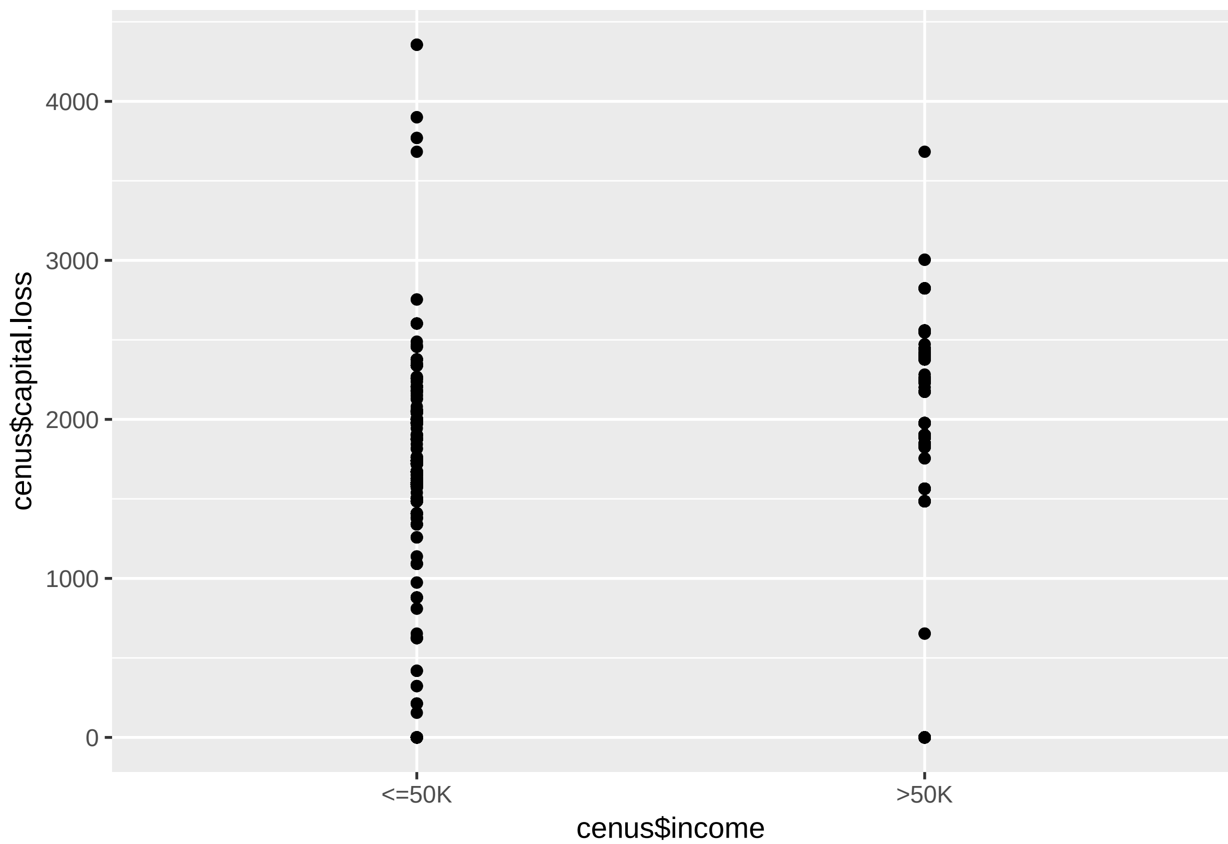


11.12. #–capital.gain & capital.loss–##

```
qplot(cenus$income,cenus$capital.gain)
```



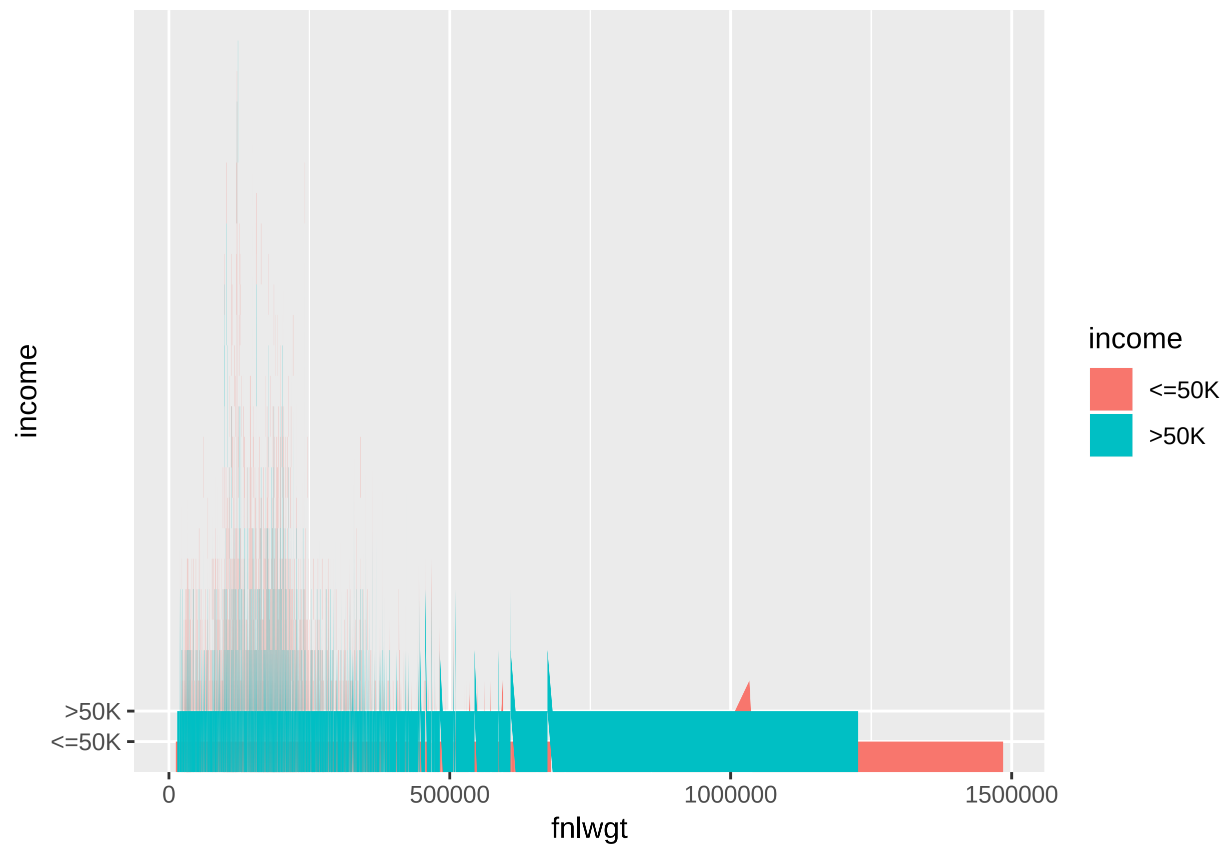

```
qplot(cenus$income,cenus$capital.loss)
```



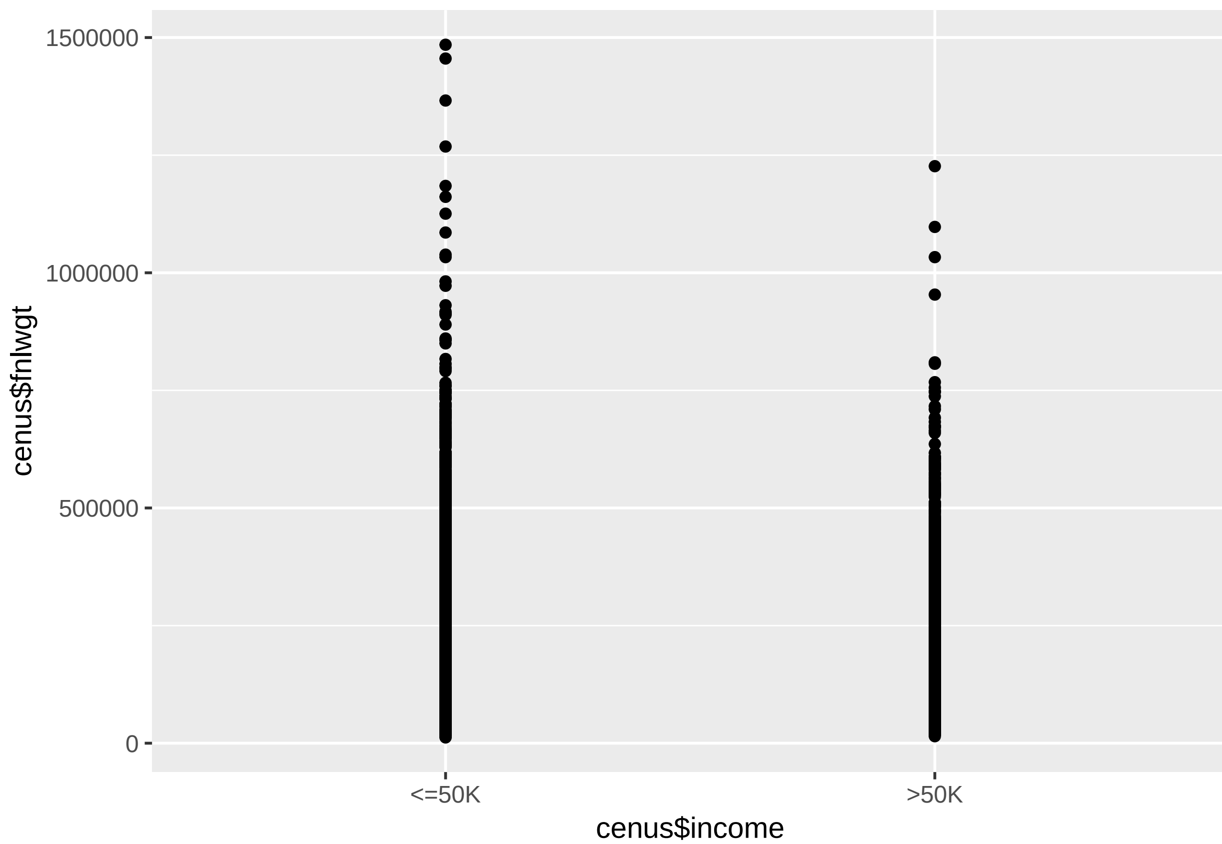
13.

–fnlwgt–

```
ggplot(cenus, aes(x=fnlwgt, y= income, fill=income )) +geom_area()
```




```
qplot(cenus$income,cenus$fnlwgt )
```



###Description of fnlwgt (final weight)

The weights on the Current Population Survey (CPS) files are controlled to independent estimates of the civilian noninstitutional population of the US. These are prepared monthly for us by Population Division here at the Census Bureau. We use 3 sets of controls. These are:

```
A single cell estimate of the population 16+ for each state.

Controls for Hispanic Origin by age and sex.

Controls by Race, age and sex.
```

We use all three sets of controls in our weighting program and “rake” through them 6 times so that by the end we come back to all the controls we used. The term estimate refers to population totals derived from CPS by creating “weighted tallies” of any specified socio-economic characteristics of the population. People with similar demographic characteristics should have similar weights. There is one important caveat to remember about this statement. That is that since the CPS sample is actually a collection of 51 state samples, each with its own probability of selection, the statement only applies within state.

fnlwgt gives us a noisy result and it seems like it will not be usful for our modules

Remove unusfull variable

```
dat_train<- select(cenus,-fnlwgt)
```

first divide the data to training and test sets.

```
library(caret)
index <- createDataPartition(dat_train$income, times = 1, p= 0.70, list = FALSE)
train_set <- dat_train %>% slice(index)
test_set <- dat_train %>% slice(-index)
```

Number of training set and test set rows and colums .

fit a decession tree for categorical outcome ,since we have more than two variables

– decession tree–

```
tree.model <- train(income ~ . ,
                    data = train_set,method = "rpart")
plot(tree.model)

**Accuracy**
**0.8268126**

y_hat_tree <- predict(tree.model,test_set)
confusionMatrix(y_hat_tree,reference = test_set$income)$overall["Accuracy"]

tree.model_2 <- train(income ~ . ,
                      data = train_set,method = "rpart",
                      tuneGrid=data.frame(cp=seq(0,0.05,len=25)))
`plot(tree.model_2)`
y_hat_tree_2 <- predict(tree.model_2,test_set)
confusionMatrix(y_hat_tree_2,reference = test_set$income)$overall["Accuracy"]
**Accuracy**
**0.8456012**
```

–KNN module–

```
control <- trainControl(method = "cv",number= 10, p=0.9)
knn_model_fit <- train(income~. , data= train_set, method= "knn", tuneGrid = data.frame(k=seq(5,25,2)),
plot(knn_model_fit)

model_fitbest <- knn3(income ~ . , data = train_set , k= 17)
y_hat_knn <- predict(model_fitbest,test_set,type  ="class")
confusionMatrix(y_hat_knn,test_set$income)$overall["Accuracy"]

**Accuracy**
**0.8440539**
```

–Random forest method–

```
library(randomForest)
control <- trainControl( method = "cv", number = 5, p= .8)
```

```
grid <- expand.grid(minNode=c(1,2,3,4,5),predFixed=c(10,15,25,35,50))
ff <- train(income ~., method = "Rborist",data=train_set,nTree=50,trControl= control, tuneGrid=grid, nSa
plot(ff)
ff$bestTune
**predFixed minNode**
**1        10       1**
ranfor.model <- randomForest::randomForest(income ~ .
                                          , data = train_set, trees=1000, minNode=1, predFixed=10  )

y_hat_forest<- predict(ranfor.model,test_set)
confusionMatrix(y_hat_forest, test_set$income)$overall["Accuracy"]
**Accuracy**
**0.8607427**
imp <- importance(ranfor.model)
```

adjust the module by removing the least effective variable

```
adjus_ranfor.model <- randomForest::randomForest(income ~ .-native.country
                                          , data = train_set, trees=1000, minNode=1, predFixed=10  )
y_hat_forestad<- predict(adjus_ranfor.model,test_set)
confusionMatrix(y_hat_forestad, test_set$income)$overall["Accuracy"]
** Accuracy**
**0.8627321**
  importance of variables
imp <- importance(adjus_ranfor.model)
imp %>% kable()
```

Adult Census Income Summary

Var	MeanDecreaseGini
age	813.37117
workclass	264.10423
education	478.70119
education.num	472.50479
marital.status	721.47571
occupation	621.42287
relationship	868.80386
race	112.39663
sex	83.86122
capital.gain	817.78815
capital.loss	244.20488
hours.per.week	482.95512

Algorithm	Accuracy
decession tree	0.845
KNN	0.844
Random forest	0.862

From the table above we can concolde the importance of each variable in our data set and how this could help us to predict the income of people.

managed to built suitable machine learning to predict the income with highest accuracy of 0.86.

for more details about the dataset DATA ON KAGGLE