Assignment 3: Theory of Naive Bayes Classifier

Shashwat Patel
Metallurgical and Materials Engineering
Indian Institute of Technology,Madras
mm19b053@smail.iitm.ac.in

***Abstract*—Naive Bayes is a term that is collectively used for classification algorithms that are based on Bayes Theorem.It is a probabilistic classifier. It is called "naive" because it makes the assumption that the occurrence of a certain feature is independent of the occurrence of other features. In this paper, we implement naive Bayes technique on a dataset and classify whether a person earns more than $50K a year or not. We study how factors like age, work class, level of education, occupation, gender etc. affect the income of a person.**

***Index Terms*—Naive Bayes, Bayes Theorem, kNN imputation, posterior, prior**

I. Introduction

This income data was extracted from the 1994 Census bureau database by Ronny Kohavi and Barry Becker (Data Mining and Visualization, Silicon Graphics).

Naive Bayes is one of the simplest machine learning algorithms for classification. It is based on Bayes' probability theorem. It is not only known for its simplicity, but also for its effectiveness. It is called "Naive" because the classifier assumes that the input features are independent of each other. Hence, changing one input feature won't affect any of the other feature. It's therefore naive in the sense that this assumption may or may not be true. [1]

This algorithm is quite popular to be used in Natural Language Processing. It is primarily used for text classification. A few examples are spam filtration, sentimental analysis, and classifying news articles. There are different kinds of naive bayes classifier like Gaussian Naive Bayes classifier, Multinomial Naive Bayes classifier and Bernoulli Naive Bayes classifier.

Income is a pretty important factor in deciding a person's life. We have to study what parameters are important in deciding a person's income. Whether a person earns over $50K a year or not is predicted by using the naive bayes algorithm.

This paper majorly deals with the implementation of naive Bayes classifier and the mathematics behind it. We try to understand how parameters like age, level of education, occupation, native country, work class, gender, capital gain/loss, marital status etc. affect the income of a person. Using the adult.csv dataset we train the naive bayes algorithm and predict whether income of a person exceeds $50K per year or not based on census data.

II. Naive Bayes

Naive Bayes is a supervised machine learning algorithm that is primarily used for classification problems. Naive Bayes classifier assumes that the features we use to predict the target are independent and do not affect each other. While in real-life data, many of the features depend on each other in determining the target, but this is ignored by the Naive Bayes classifier. Though the independence assumption is rarely correct in real-world data, but often works well in practice.

It is already learnt that naive bayes is based on Bayes theorem. The Bayes theorem gives us a method to calculate the conditional probability, i.e., the probability of an event based on previous knowledge available on the events. Bayes' Theorem is stated as the following equation:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{1}$$

Where,

- $P(A|B)$ is the probability of occurrence of event A given the event B is true.
- $P(A)\ and\ P(B)$ is the probability of the occurrence of event A and B respectively.
- $P(B|A)$ is the probability of the occurrence of event B given the event A is true

Here, the event A is known as the proposition and event B is known as the evidence. $P(A)$ is called prior probability of proposition and $P(B)$ is known as prior probability of evidence. $P(A|B)$ is called the posterior and $P(B|A)$ is called the likelihood.

The Bayes Rule is a way of going from $P(B|A)$ to finding $P(A|B)$. In simple terms, it provides a way to calculate the probability of a proposition given the evidence.

Bayes's Theorem for Naive Bayes: [2] In classification problems, there are multiple feature and classes $(C_1, C_2, C_3, \ldots, C_n)$. The aim is to calculate the conditional probability of an object with feature vector $(x_1, x_2, x_3, \ldots, x_n)$ belonging to a particular class C_i. Using equation 1,

$$P(C_i|x_1, x_2, \ldots, x_n) = \frac{P(x_1, x_2, \ldots, x_n|C_i)P(C_i)}{P(x_!, x_2, \ldots, x_n)} \tag{2}$$

Here,

$$P(x_1, x_2, \ldots, x_n|C_i)P(C_i) = P(x_1, x_2, \ldots, x_n, C_i) \tag{3}$$

We can write the RHS of equation 3 like,

$$P(x_1, x_2, \ldots, x_n, C_i) = P(x_1|x_2, \ldots, x_n, C_i)P(x_2, \ldots, x_n, C_i) \quad (4)$$

Continuing the equation 4 RHS,

$$P(x_1|x_2, \ldots, x_n, C_i)P(x_2|x_3, \ldots, x_n, C_i)P(x_3, \ldots, x_n, C_i) \quad (5)$$

At end we get,

$$P(x_1|x_2, \ldots, x_n, C_i)P(x_2|x_3, \ldots, x_n, C_i) \ldots P(x_n|C_i).P(C_i) \quad (6)$$

Equation 6 reduces to,

$$= P(x_1|C_i)P(x_2|C_i) \ldots P(x_{n-1}|C_i)P(x_n|C_i)P(C_i) \quad (7)$$

The conditional probability term, $(P(x_j|x_{j+1}, \ldots, x_n, C_i))$ becomes $(P(x_j|C_i))$ because of the assumption that features are independent.From the calculation above and the independence assumption, the expression becomes [2]:

$$P(C_i|x_1, x_2, \ldots, x_n) = \left(\prod_{j=1}^{j=n} P(x_j|C_i)\right) \cdot \frac{P(C_i)}{P(x_1, x_2, \ldots, x_n)} \quad (8)$$

$P(x_1, x_2, \ldots, x_n)$ is constant for all classes. It can be said that.

$$P(C_i|x_1, x_2, \ldots, x_n) \propto \left(\prod_{j=1}^{j=n} P(x_j|C_i)\right) .P(C_i) \quad (9)$$

The equation 9 is the naive Bayes probability model. The naive Bayes classifier combines this model with a decision rule.A Bayes classifier, is the function that assigns a class label $\hat{y_k} = C_k$ for some k. [3]

There are different types of naive Bayes classifier [4]:

Multinomial Naive Bayes Classifier-Feature vectors represent the frequencies with which certain events have been generated by a multinomial distribution.

Bernoulli Naive Bayes Classifier-Features are independent binary variables describing inputs.

Gaussian Naive Bayes Classifier-Continuous values associated with each feature are assumed to be distributed according to a Normal distribution.

Since, naive Bayes is an classification algorithm, to assess the model's accuracy confusion matrix is used. The confusion matrix avoids "confusion" by measuring the actual and predicted values in a tabular format. Different metrics can be derived from confusion matrix like accuracy, precision, F-score, sensitivity and specificity.

III. The Problem

The "adult.csv" dataset consists of features like occupation, work class, education, years of education, fnlwgt, age, gender, marital_status, relationship etc. and the task is to predict whether the person earns more than $50K a year or not. This prediction is done using the naive Bayes classifier. The dataset consists of 15 variables and 32561 observations.

Data Cleaning: The data consists of "?" as NAs.Native country had 583 missing observations, Occupation had 1843 observations missing, workclass had 1836 observations missing. In total around 7% of rows had observation missing. kNN imputation [5] was used to replace NAs as it is better for filling up the missing categorical variables. kNN Imputation uses k-Nearest Neighbours approach to impute missing values.kNN imputation does work as follows: For every observation to be imputed, it identifies 'k' closest observations based on the euclidean distance and computes the weighted average of these 'k' observation.

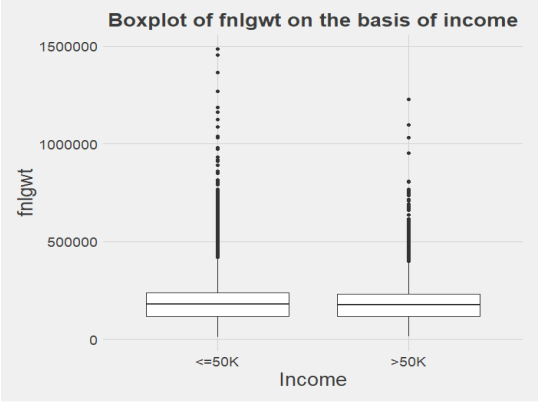

Fig. 1. Box plot of fnlwgt

Figure 1. shows the box plot of fnlwgt on the basis of income. There is not much difference in distribution of data whether person earns more than $50K or not, so the column fnlwgt is not used in the model.

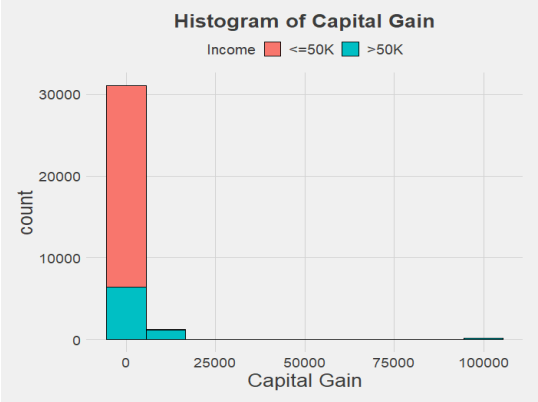

Fig. 2. Capital gain

Figure 2 and Figure 3 are histogram of capital gain and loss respectively. Capital gain and capital loss have very narrow distributions and are highly skewed. More than 90% of data are clustered at zero therefore these both variables are also excluded from the model.

The figure 4 is the histogram of native country, this figure is highly skewed as well. It has a very narrow distribution and

more than 90% of population is from the United State only, therefore this column is excluded from the model as well.

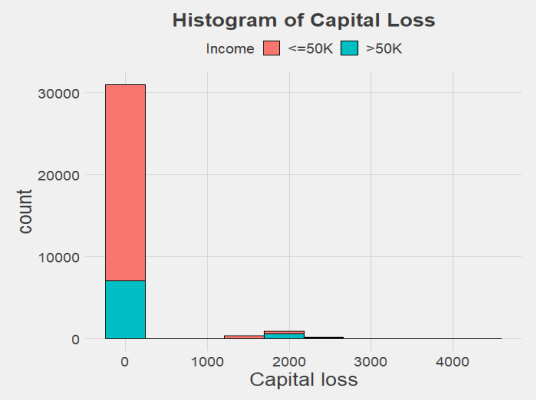

Fig. 3. Capital loss

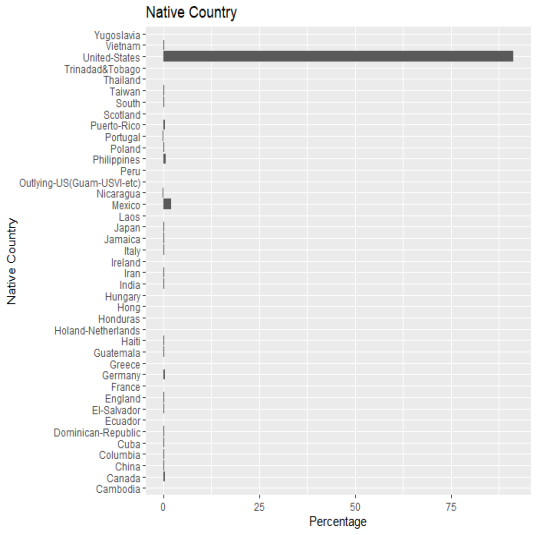

Fig. 4. Native Country

Some new columns like education_processed, occupation_processed and hours_per_week_processed have been created for data visualization purpose. The "number of years of education" column is not used as it very strongly correlated to "education" variable. The "relationship" is not used as well because it can be identified from "marital_status" and "gender" variable. The variables used for model building are age, work class, education, marital_status, Occupation, race, gender and hours_per_week.

*Data Visualization:*certain insights helps in finding important parameters which determine the income of a person. The figure 5 shows the income distribution on the basis of age. It is true that most of the population earns less than $50K a year but the figure 5 helps in finding that people who earn more than $50K a year are mostly in their mid-career i.e in the age group of 30-45.

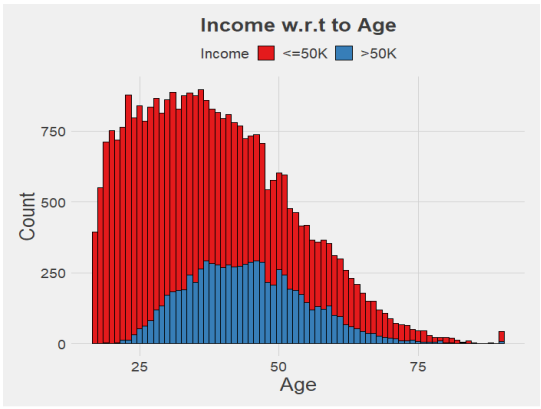

Fig. 5. Age Distribution

Figure 6 the income distribution with respect to gender. In general, according to the data, percentage-wise, the male population has more more number of people earning greater than $50K a year as compared to the female population. Around 30% of the male population and around 11% of female population earns more than $50K a year.

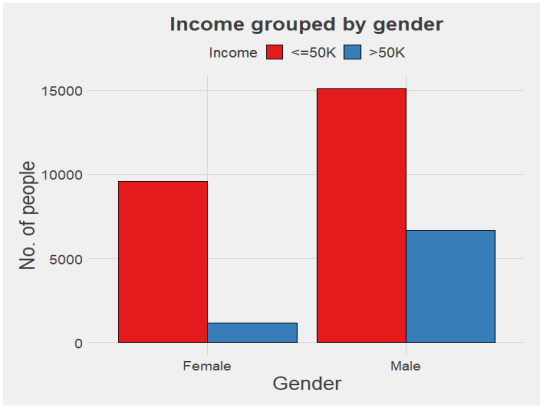

Fig. 6. Incomer w.r.t gender

Figure 7 shows the income distribution with respect to industry.

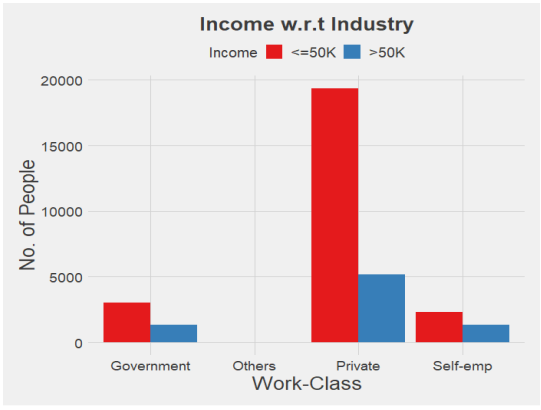

Fig. 7. Industry-Income distribution

Here, ”other”(Fig 7) consists of people have never-worked or are working without pay. There are no people who earns more than $50K a year in ”others” category. Most of the people who earn more than $50K a year work in private sector. Around 30% of people who work in government sector earn more than $50K a year. 21% of people working in private sector and 37% of people who are self-employed earn more than $50K a year.

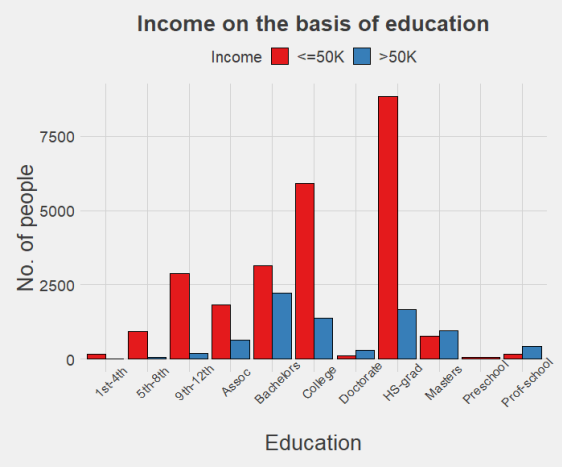

Fig. 8. Income-education distribution

Figure 8, shows the income distribution on the basis of education. Very few people who have completed high-school education earn more then $50K a year.Very few people have completed their doctorate or prof-school but a high percentage of persons who have completed it, tend to earn $50K a year. Completing their bachelors, masters, doctor or prof-school is very fruitful for people as those people tend to earn more. A higher level of education makes people get higher income.

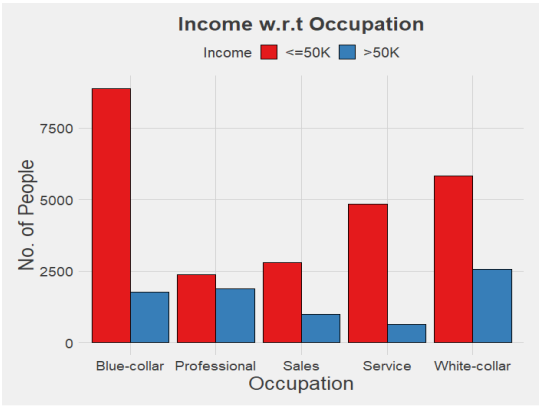

Fig. 9. Income-occupation distribution

Figure 9 shows the income distribution with respect to occupation. Here, ”blue-collar” refers to people who do manual labor or work in a division of manufacturing. ”White-collar” refers to people who work behind the desk in service industry. Generally, ”white-collar” earn more than ”blue-collar” [6]. From figure 9 it is seen that high population of our data works as blue-collar and few of them earn more than $50K a year. The professionals earn more in general and it is seen that a good amount of professionals earn more than $50K a year.

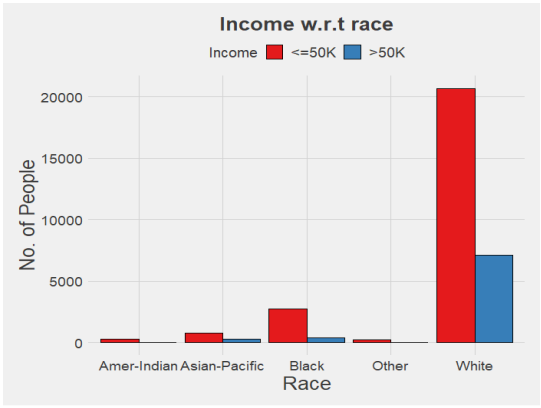

Fig. 10. Income w.r.t race

Figure 10, shows the income distribution on the basis of race. Majority of the data consists of whites, so the race data is also skewed as well. It is not necessary to use the race data for model building as well.

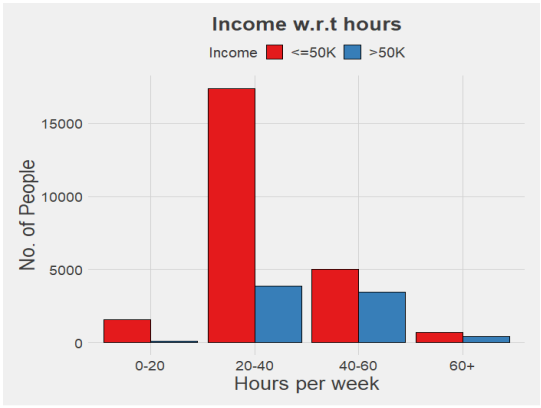

Fig. 11. Income w.r.t hours per week

Figure 11, shows the income distribution with respect to hours per week put by a person in his job. People who work very less hours in a week, generally these people consists of students who are working part-time and they tend to earn less. Most of the population works 20-40 hours per week and some them earn more than $50K but a high amount of people who earn more than $50K work for around 40-60 hours per week. Very few people in our data work more than 60 hours per week, so not much can be inferred form that but in general, people who work more hours per week tends to earn more.

Naive Bayes Model: A single naive Bayes model is built which uses these following features: age, gender, industry, education, marital status, occupation and hours per work. (To

see the classification model and the conditional probabilities check code documentation)

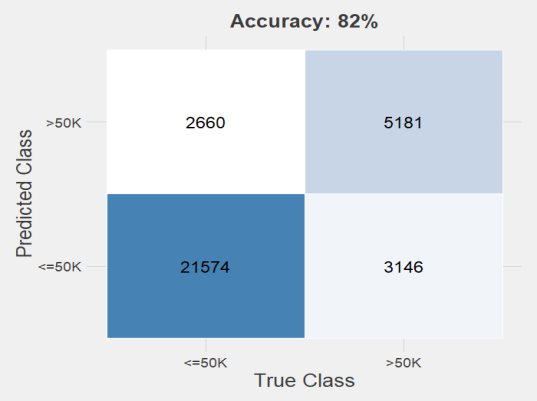

Fig. 12. Confusion Matrix

The accuracy of our naive Bayes model is 82%. 2660 persons who actually earn less than $50K were predicted to earn more than $50K and 3146 people who earn more than $50K were predicted to earn less than $50K. The model correctly predicted 21574 people who earn less than $50K and 5181 people who earn more than $50K.

Age, hour per week, occupation and education are pretty important variables affecting the income of a person. Marital status doesn't affect the income very much.

IV. Conclusion

From the model and the visualizations it is learnt that people whoa are in their mid career tend to earn more. Male population tends to earn more and higher level of education is very fruitful for earning high income as it was seen that bachelors, masters and doctorate tend to earn more. Regarding occupation, professionals earn more compared to others. We also saw that people who put more hours in work tend to earn more as compared to others. Our naive Bayes models had an accuracy of 82%.

Further improvements in the model are possible. The outliers were not accounted for while developing the model, outliers can be accounted when 2nd model is built. "fnlwgt, capital gain, capital loss, native country" column can be used for further insights. Instead of using only naive Bayes other models like random forest, support vector machine and further sophisticated models like neural networks can also be used.

References

[1] https://blog.paperspace.com/introduction-to-naive-bayes/
[2] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. An Introduction to Statistical Learning : with Applications in R. New York :Springer, 2013.
[3] https://en.wikipedia.org/wiki/Naive_Bayes_classifier
[4] https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html
[5] https://cran.r-project.org/web/packages/VIM/VIM.pdf
[6] https://www.investopedia.com/articles/wealth-management/120215/blue-collar-vs-white-collar-different-social-classes.asp

Code Documentation

Importing required libraries

```
library(tidyverse)
library(VIM)
library(ggthemes)
library(scales)
library(e1071)
library(caret)
```

Reading and giving column names to dataset

- In the dataset "?" are NAs.

```
data<-read_csv("adult.csv",col_names =FALSE,na="?")

colnames(data)<-c("age","workclass","fnlwgt","education","education_num",
                  "marital_status","Occupation","relationship","race","sex",
                  "capital_gain","capital_loss","hours_per_week","native_country","Income")
```

Function for checking number of NAs

```
na_check<-function(dataset){
  sapply(dataset,function(x) sum(is.na(x)))
}

na_check(data)
```

Basic summary of data

```
str(data)
summary(data)
```

Character columns converted to factors

```
data<-data%>% mutate_if(is.character,as.factor)
```

Removing education_num, relationship column

```
data<-data%>%select(-c(education_num,relationship))
```

Imputation

- kNN imputation
- some extra columns get created due to kNN imputation, these are removed as well.

```
data_tidy<-kNN(data,variable = c("workclass","Occupation","native_country"),k=sqrt(nrow(data)))
data_tidy<-data_tidy[,1:13]
```

Visualizations

Theme set

```
my_theme<-theme_fivethirtyeight()+theme(plot.title = element_text(hjust = 0.5,size=20),
                                        axis.title = element_text(size=20),
                                        axis.text = element_text(size=14),
                                        plot.subtitle = element_text(hjust=0.5),
                                        legend.position = "top",legend.title = element_text(size=15),
                                        legend.text = element_text(size=15))
```

Boxplot of fnlwgt(Fig 1)

```
ggplot(data_tidy)+geom_boxplot(aes(x=Income,y=fnlwgt))+
  labs(title="Boxplot of fnlwgt on the basis of income")+
  my_theme
```

Fig 2

```
ggplot(data_tidy,aes(x=capital_gain, group=Income, fill=Income)) +
  geom_histogram(bins=10, color='black') +
  ggtitle('Histogram of Capital Gain')+
  labs(x="Capital Gain")+
  my_theme
```

Fig 3

```
ggplot(data_tidy,aes(x=capital_loss, group=Income, fill=Income)) +
  geom_histogram(bins=10, color='black') +
  ggtitle('Histogram of Capital Loss')+
  labs(x="Capital loss")+
  my_theme
```

Fig 4

```
ggplot(data_tidy, aes(x=native_country)) +
  ggtitle("Native Country") + xlab("Native Country") +
  geom_bar(aes(y = 100*(..count..)/sum(..count..))) +
  ylab("Percentage") + coord_flip()
```

Fig 5

```
ggplot(data_tidy)+geom_histogram(aes(age,fill=Income),binwidth=1,col="black")+
  labs(x="Age",y="Count",title = "Income w.r.t to Age")+
  my_theme+
  scale_fill_brewer(palette = "Set1")
```

Fig 6

```
ggplot(data_tidy)+geom_bar(aes(sex,fill=Income),col="black",position = "dodge")+
  labs(x="Gender",y="No. of people",title = "Income grouped by gender")+
  my_theme+
  scale_fill_brewer(palette = "Set1")
```

- To find percentage of male/female population earning greater than $50K

```
data_tidy%>%group_by(Income)%>%summarise(sum(sex=="Male"))
```

```
## # A tibble: 2 x 2
##   Income `sum(sex == "Male")`
##   <fct>                <int>
## 1 <=50K                15128
## 2 >50K                  6662
```

```
6662/sum(data_tidy$sex=="Male")
```

```
## [1] 0.3057366
```

```
data_tidy%>%group_by(Income)%>%summarise(sum(sex=="Female"))
```

```
## # A tibble: 2 x 2
##   Income `sum(sex == "Female")`
##   <fct>                   <int>
## 1 <=50K                    9592
## 2 >50K                     1179
```

```
1179/sum(data_tidy$sex=="Female")
```

```
## [1] 0.1094606
```

Fig 7

- New column "workclass_processed" created
- Percentage of people earning more than $50K on the basis of workclass

```
data_tidy<-data_tidy%>%mutate(workclass_processed=case_when(
                                          workclass=="State-gov" ~ "Government",
                                          workclass=="Self-emp-not-inc" ~ "Self-emp",
                                          workclass=="Private" ~ "Private",
                                          workclass=="Federal-gov" ~ "Government",
                                          workclass=="Local-gov" ~ "Government",
                                          workclass=="Self-emp-inc" ~ "Self-emp",
                                          workclass=="Without-pay" ~ "Others",
                                          workclass=="Never-worked"~"Others"
))

data_tidy%>%group_by(workclass_processed)%>%summarise(n())

data_tidy%>%group_by(workclass_processed)%>%filter(Income==">50K")%>%summarise(n())
```

```
data_tidy%>%ggplot()+geom_bar(aes(workclass_processed,fill=Income),position = "dodge")+
  my_theme+
  labs(x="Work-Class",y="No. of People",title="Income w.r.t Industry")+
  scale_fill_brewer(palette = "Set1")
```

Fig 8

- "Education processed" column created

```
data_tidy<-data_tidy%>%mutate(education_processed=case_when(
                                            education=="Bachelors" ~ "Bachelors",
                                            education=="HS-grad" ~ "HS-grad",
                                            education=="11th" ~ "9th-12th",
                                            education=="Masters" ~ "Masters",
                                            education=="9th" ~ "9th-12th",
                                            education=="Some-college" ~ "College",
```

```
                                                education=="Assoc-acdm" ~ "Assoc",
                                                education=="Assoc-voc" ~ "Assoc",
                                                education=="7th-8th" ~ "5th-8th",
                                                education=="Doctorate" ~ "Doctorate",
                                                education=="Prof-school" ~ "Prof-school",
                                                education=="5th-6th" ~ "5th-8th",
                                                education=="10th" ~ "9th-12th",
                                                education=="1st-4th" ~ "1st-4th",
                                                education=="Preschool" ~ "Preschool",
                                                education=="12th" ~ "9th-12th"

))
```

```
ggplot(data_tidy)+geom_bar(aes(education_processed,fill=Income),col="black",position = "dodge")+
  theme_fivethirtyeight()+theme(plot.title = element_text(hjust = 0.5,size=20),
                                axis.title = element_text(size=20),
                                axis.text = element_text(size=14),
                                plot.subtitle = element_text(hjust=0.5),
                                legend.position = "top",legend.title = element_text(size=15),
                                legend.text = element_text(size=15),
                                axis.text.x = element_text(size=12,angle=45,hjust=0.5))+
  labs(x="Education",y="No. of people",title = "Income on the basis of education")+
  scale_fill_brewer(palette = "Set1")
```

Fig 9

- "occupation processed" columns created.

```
data_tidy<-data_tidy%>%mutate(occupation_processed=case_when(
                                                    Occupation=="Adm-clerical" ~ "White-collar",
                                                    Occupation=="Exec-managerial" ~ "White-collar",
                                                    Occupation=="Handlers-cleaners" ~ "Blue-collar",
                                                    Occupation=="Prof-specialty" ~ "Professional",
                                                    Occupation=="Other-service" ~ "Service",
                                                    Occupation=="Sales" ~ "Sales",
                                                    Occupation=="Craft-repair" ~ "Blue-collar",
                                                    Occupation=="Transport-moving" ~ "Blue-collar",
                                                    Occupation=="Farming-fishing" ~ "Blue-collar",
                                                    Occupation=="Machine-op-inspct" ~ "Blue-collar",
                                                    Occupation=="Tech-support" ~ "Service",
                                                    Occupation=="Protective-serv" ~ "Service",
                                                    Occupation=="Armed-Forces" ~ "Service",
                                                    Occupation=="Priv-house-serv" ~ "Service"


))
```

```
ggplot(data_tidy)+geom_bar(aes(occupation_processed,fill=Income),col="black", position = "dodge")+
  labs(x="Occupation",y="No. of People",title = "Income w.r.t Occupation")+
  my_theme+
  scale_fill_brewer(palette = "Set1")
```

Fig 10

```
data_tidy<-data_tidy%>%mutate(race=case_when(race=="Asian-Pac-Islander" ~ "Asian-Pacific",
                                             race=="Amer-Indian-Eskimo" ~ "Amer-Indian",
                                             race=="White" ~ "White",
                                             race=="Black" ~ "Black",
                                             race=="Other" ~ "Other"))

ggplot(data_tidy)+geom_bar(aes(race,fill=Income),col="black", position = "dodge")+
  labs(x="Race",y="No. of People",title = "Income w.r.t race")+
  my_theme+
  scale_fill_brewer(palette = "Set1")
```

Fig 11

- hours_per_week_processed column created

```
data_tidy<-data_tidy%>% mutate(hours_per_week_processed=case_when(
                                        hours_per_week<20~"0-20",
                                        hours_per_week>=20 & hours_per_week<=40 ~ "20-40",
                                        hours_per_week>40 & hours_per_week<=60 ~ "40-60",
                                        hours_per_week>60 ~ "60+"))


ggplot(data_tidy)+geom_bar(aes(hours_per_week_processed,fill=Income),col="black",position = "dodge")+
  labs(x="Hours per week",y="No. of People",title = "Income w.r.t hours")+
  my_theme+
  scale_fill_brewer(palette = "Set1")
```

Model

```
data_model_1<-data_tidy[,c(1:2,4,5,6,8,11,13)]

set.seed(1)

classifier<-naiveBayes(Income~.,data=data_model_1)

predicted_data<-predict(classifier,newdata=data_model_1)

classifier
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
```

```
## Y
##     <=50K      >50K
## 0.7591904 0.2408096
##
## Conditional probabilities:
##        age
## Y           [,1]     [,2]
##   <=50K 36.78374 14.02009
##   >50K  44.24984 10.51903
##
##        workclass
## Y        Federal-gov    Local-gov Never-worked      Private Self-emp-inc
##   <=50K 0.0238268608 0.0597087379 0.0002831715 0.7838592233 0.0199838188
##   >50K  0.0473153934 0.0786889427 0.0000000000 0.6573141181 0.0793266165
##        workclass
## Y       Self-emp-not-inc    State-gov  Without-pay
##   <=50K     0.0735436893 0.0382281553 0.0005663430
##   >50K      0.0923351613 0.0450197679 0.0000000000
##
##        education
## Y               10th         11th         12th      1st-4th      5th-6th
##   <=50K 0.0352346278 0.0451051780 0.0161812298 0.0065533981 0.0128236246
##   >50K  0.0079071547 0.0076520852 0.0042086469 0.0007652085 0.0020405561
##        education
## Y            7th-8th          9th   Assoc-acdm    Assoc-voc    Bachelors
##   <=50K 0.0245145631 0.0197006472 0.0324433657 0.0413025890 0.1267799353
##   >50K  0.0051013901 0.0034434383 0.0337967096 0.0460400459 0.2832546869
##        education
## Y          Doctorate      HS-grad      Masters    Preschool  Prof-school
##   <=50K 0.0043284790 0.3570388350 0.0309061489 0.0020631068 0.0061893204
##   >50K  0.0390256345 0.2136207116 0.1223058283 0.0000000000 0.0539472006
##        education
## Y       Some-college
##   <=50K 0.2388349515
##   >50K  0.1768907027
##
##        marital_status
## Y          Divorced Married-AF-spouse Married-civ-spouse Married-spouse-absent
##   <=50K 0.161003236       0.000525890        0.335113269           0.015533981
##   >50K  0.059048591       0.001275348        0.853462569           0.004336182
##        marital_status
## Y       Never-married   Separated     Widowed
##   <=50K   0.412297735 0.038794498 0.036731392
##   >50K    0.062619564 0.008417294 0.010840454
##
##        Occupation
## Y       Adm-clerical Armed-Forces Craft-repair Exec-managerial Farming-fishing
##   <=50K 0.1503640777 0.0003236246 0.1464805825    0.0861245955    0.0355987055
##   >50K  0.0681035582 0.0001275348 0.1268970795    0.2594056880    0.0146664966
##        Occupation
## Y       Handlers-cleaners Machine-op-inspct Other-service Priv-house-serv
##   <=50K      0.0539239482      0.0720064725  0.1448220065    0.0059870550
##   >50K       0.0109679888      0.0318836883  0.0174722612    0.0001275348
##        Occupation
```

```
## Y       Prof-specialty Protective-serv        Sales Tech-support
##   <=50K   0.0956715210    0.0177184466 0.1132281553 0.0260922330
##   >50K    0.2409131488    0.0269098329 0.1256217319 0.0360923352
##        Occupation
## Y       Transport-moving
##   <=50K     0.0516585761
##   >50K      0.0408111210
##
##        sex
## Y          Female      Male
##   <=50K 0.3880259 0.6119741
##   >50K  0.1503635 0.8496365
##
##        hours_per_week
## Y           [,1]     [,2]
##   <=50K 38.84021 12.31899
##   >50K  45.47303 11.01297
```

Confusion Matrix

```
conf_mat<-confusionMatrix(data_model_1$Income, predicted_data)

conf_mat
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction <=50K  >50K
##      <=50K 21676  3044
##      >50K   2680  5161
##
##                Accuracy : 0.8242
##                  95% CI : (0.82, 0.8283)
##     No Information Rate : 0.748
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5267
##
##  Mcnemar's Test P-Value : 1.603e-06
##
##             Sensitivity : 0.8900
##             Specificity : 0.6290
##          Pos Pred Value : 0.8769
##          Neg Pred Value : 0.6582
##              Prevalence : 0.7480
##          Detection Rate : 0.6657
##    Detection Prevalence : 0.7592
##       Balanced Accuracy : 0.7595
##
##        'Positive' Class : <=50K
##
```

Confusion Matrix Plot

```
cm_dataframe<-as.data.frame(conf_mat$table)


ggplot(data =cm_dataframe ,
       aes(x = Reference, y = Prediction)) +
  geom_tile(aes(fill = log(Freq)), colour = "white") +
  scale_fill_gradient(low = "white", high = "steelblue") +
  geom_text(aes(x = Reference, y = Prediction, label = Freq),size=6) +
  labs(x="True Class",y="Predicted Class")+
  ggtitle(paste("Accuracy:",percent_format()(conf_mat$overall[1])))+
  theme_fivethirtyeight()+
  theme(legend.position = "none",
        axis.title = element_text(size=20),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust=0.5,size=20))
```