Descriptive Analytics through Tabulation, Graphs & Statistical Measures Workshop

TBA2102 2020/2021 Semester 2
Tutorial 3

STRUCTURE OF TUTORIALS

Duration:

45 mins

Content:

- Cover previous week’s tutorial assignment
- Descriptive analytics in R

TUTORIAL 2 ASSIGNMENT

QUESTION 1 A-C

What is the output for each of the following sets of codes?

a

```
x <- c(4,2,2,1)
y <- c(2,1,2,1)
z <- x/y
z
```

2 2 1 1

b

i

```
height <- c(110,120,125,100)
order(height, decreasing = TRUE)
```

3 2 1 4

Default argument

ii

```
sort(height,decreasing=FALSE)
```

100 110 120 125

c

```
grade <- c("good","bad","good","bad")
         factor(grade,
         levels=c("good", "bad"),
         ordered =FALSE)
```

good bad good bad

Levels: good bad

Create a vector with all the observations

Create a factor using the created vector

Correctly specify the levels

Tell R whether the factor is ordinal/nominal

What if ordered=TRUE?

QUESTION 1 D-E

What is the output for each of the following sets of codes?

d

i

```
s <- c(11,13,21,15,9,"false")
class(s)
```

character

ii

```
s[c(2,6)]
```

"13" "false"

e

i

```
df <- data.frame(candidate=c("Andy","Bob","Dylan","Elyse","Fay"),
                 score=c(4,8,5,8,7))
class(df$score)
```

numeric

ii

```
df[c(2,4),2]
```

8 8

iii

```
df$candidate <- as.character(df$candidate)
df[4,"candidate"]
```

"Elyse"

Why?

iv

```
subset(df,score>7,select= candidate)
```

	candidate
2	Bob
4	Elyse

QUESTION 2A-B

For each question part below, what is the missing code (“?”) required to return the output?

a

```
x <- c(1,3,10, 8)
y <- c("Mon","Tue","Wed","Thu")
   ? (x) <- y
x
```

Output

```
Mon Tue Wed Thu
  1   3  10   8
```

Missing code

names

b(ii)

```
? >Satisfac[3]
```

Output: TRUE

Missing code: Satisfac[1]>Satisfac[3]

b (i)

```
Satisfaction<-c("good","excellent", "poor","fair")
Satisfac<-factor(Satisfaction,
                 levels=c(          ?          ),
                 ordered=TRUE)
Satisfac
```

Output

```
good      excellent poor      fair
Levels: poor < fair < good < excellent
```

Missing code: "poor","fair","good","excellent"

QUESTION 2C (I)

```
recipe<-list(c("Pancake","Egg","Cereal","Bread"),
             c(          ?          ),
             c(2,3,1))
names(recipe) <- c("Breakfast","Snacks","Qty")
recipe
```

Output

```
$Breakfast
[1] "Pancake" "Egg"     "Cereal"  "Bread"

$Snacks
[1] "Cookie"  "Pretzel"

$Qty
[1] 2 3 1
```

Missing code

"Cookie","Pretzel"

QUESTION 2C

ii

```
recipe$      ?
```

Output: “Egg”

Missing code: Breakfast[2]

iii

```
recipe[[     ?     ]]
```

Output: "Cookie" "Pretzel"

Missing code: "Snacks"

QUESTION 2D

ii

```
petallen<- c(4.5,5.5,2,3,4)
  ?  (petallen)
```

Output: 2.0 3.0 4.0 4.5 5.5

Missing code: Breakfast[2]

ii

```
petal2len<- petallen   ?
petal2len
```

Output: 6.5 7.5 4.0 5.0 6.0

Missing code: +2

QUESTION 2E

i

```
df2<-data.frame(Name=c("Henry", "Mary","James","Pete"),
                Age=c(16, 44, 5, 66),
                Gender=c("M","F","M","M"))
df2[      ?       ]
```

Output: "Henry" "Mary"

Missing code: c(1,2),"Name"

ii

```
subset(df2,Age>40,    ?    )
```

Output:
Name
2 Mary
4 Pete

Missing code: "Name"

iii

```
subset(df2,Name    ?    ,
       select = "Age")
```

Output:
Age
2 44

Missing code: =="Mary"

QUESTION 3

Mary planted 5 seeds. At the end of week 2, she measured the height of each seeding (A, B, C, D, E) and recorded them in the variable ht2 in the respective order (i.e. A, B,..,E).
Height (cm) measurements taken for seedlings A, B… E at the end of week 2 were: 2, 2.5, 4, 3, 3.5

3a What is the code to assign the height measurements to ht2?

```
ht2 <- c(2, 2.5, 4, 3, 3.5)
```

3b What is the code to assign the values "A", "B",… "E" as names for ht2?

```
names(ht2) <- c("A","B","C","D","E")
```

3c What is the code to sort ht2 in decreasing value?

```
sort(ht2,decreasing=TRUE)
```

3d May recorded the height of plant B incorrectly. What code would you write, to change the value to 3?

```
ht2[2]# retrieve current value
ht2[2] <- 3
ht2[2]# retrieve updated value
```

DESCRIPTIVE ANALYTICS

WHAT GRAPH SHOULD YOU CHOOSE?

Variable type

- **Categorical Data**
  - Bar chart
  - Pie chart
- **Continuous Data**
  - Histogram

View relationship between variables

- Scatter plot

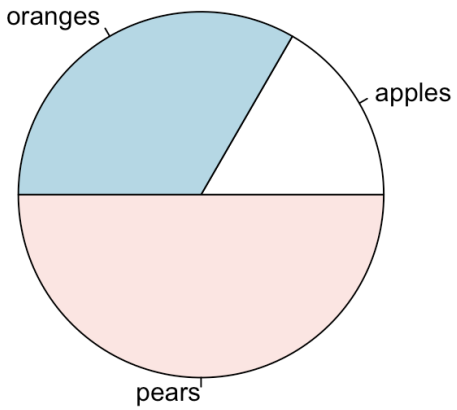
Trends

- Line chart

Remember to always consider your stakeholder's needs!

PIE CHARTS

```
count_vector <- c(1,2,3)
labels <- c("apples", "oranges", "pears")

pie(count_vector, labels)
```



When to use:

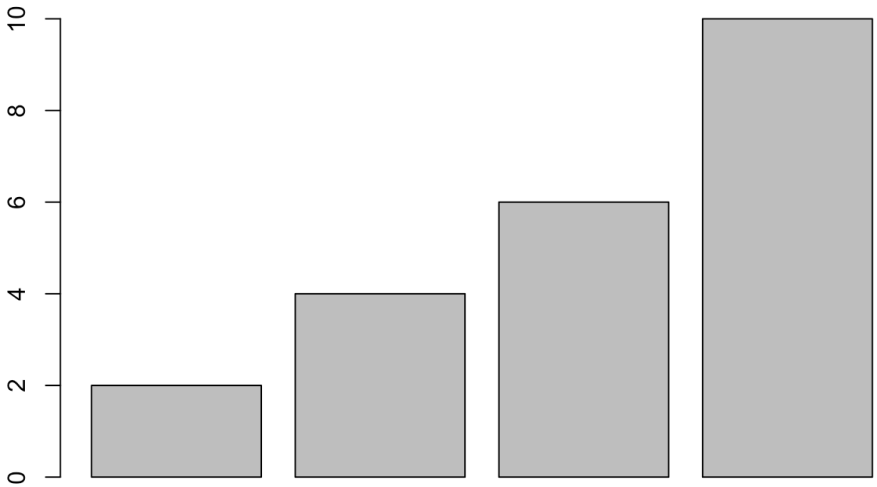
- Comparing categorical data
- Composition of an object, comparing parts to the whole object.
- When you have fewer categories.

HOW DO YOU EXTRACT COUNTS FROM DATAFRAME?

- Dataframe %>% count(category)
- count - count the number of observations in each category

BAR PLOTS

```
height <- c(2,4,6,10)

barplot(height)
```



When to use:

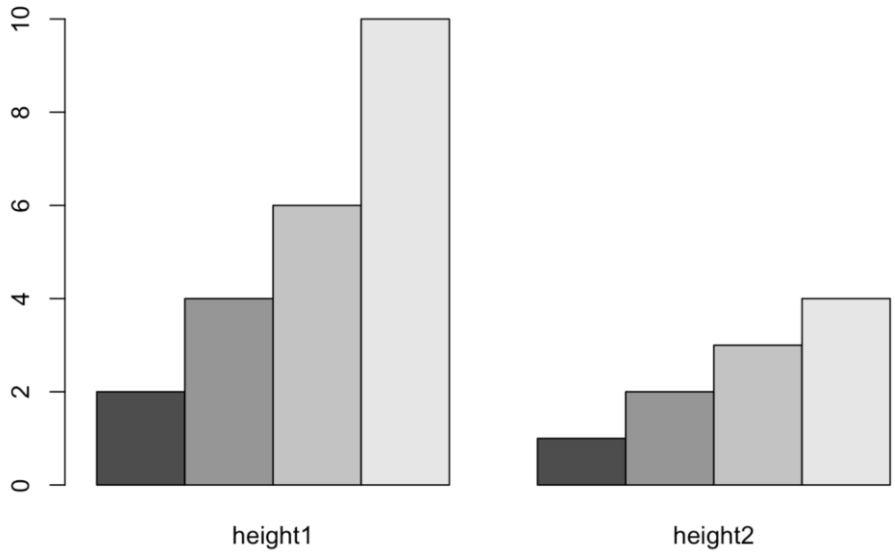
- Comparing categorical data
- Horizontal/ vertical: check the labels

BARPLOTS WITH 2 VARIABLES – NEED MATRIX!

```
height1 <- c(2,4,6,10)
height2 <- c(1,2,3,4)

ht.matrix <- cbind(height1, height2)
ht.matrix
```

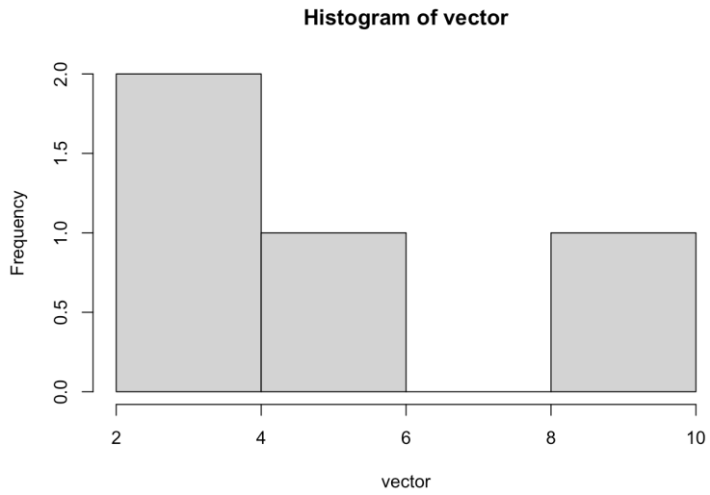
```
##      height1 height2
## [1,]       2       1
## [2,]       4       2
## [3,]       6       3
## [4,]      10       4
```

```
barplot(ht.matrix, beside = T)
```

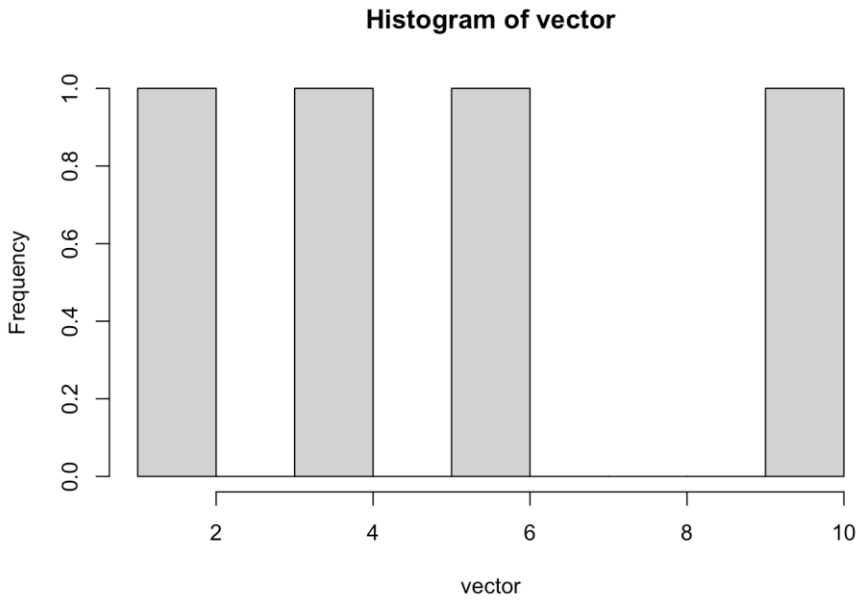


HISTOGRAMS

```
vector <- c(2,4,6,10)

hist(vector)
```




```
hist(vector, breaks = 1:10)
```

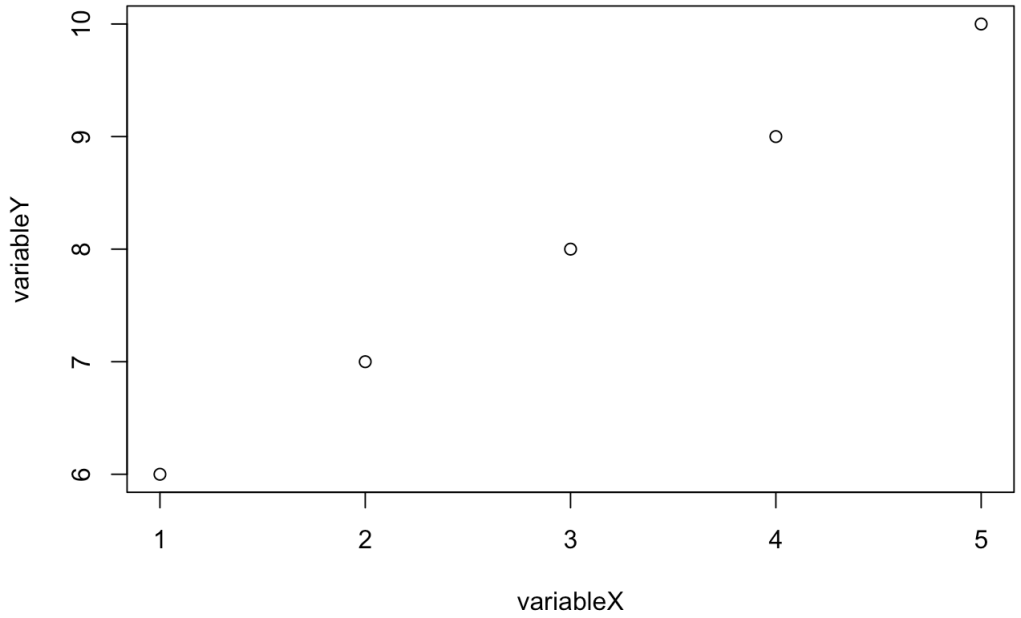


When to use:

- To show distributions of continuous variables.
- Histograms are not bar charts!
- Histograms plot binned quantitative data while bar charts plot categorical data.

SCATTERPLOTS

```
variableX <- 1:5

variableY <- 6:10

plot(variableX, variableY)
```



When to use:

- To show the relationship between two variables.
- It does not matter which variable is on the x-axis or y-axis: association vs. causality!

BANK CREDIT RISK DATA

- `Loan Purpose` : Type of purpose for the loan applied
- `Checking` : Checking account balance
- `Savings` : Savings account balance
- `Months Customer` : Number of months has been a customer of the bank
- `Months Employed` : Number of months in employment
- `Gender` : Gender
- `Marital Status` : Marital status
- `Age` : Age in years
- `Housing` : Housing type
- `Years` : Number of years at current residence
- `Job` : Job type
- `Credit Risk` : Credit-risk classification by the bank

Functions that can help you explore the data

- View(BD)
- str(BD)
- head(BD)
- lapply(BD,class) --- check the data type of all the variables

QUESTION 1A CUSTOMER PROFILE DASHBOARD

1A(i) The credit risk analysts are now interested in the following Customer demographics: Housing, Job, Credit Risk, Months Employed and Total.

Total is the sum of Checking and Savings. Create this variable Total in the dataframe.

```
dim(BD)# check the number of dimensions: 12
BD$Total<-BD$Checking+BD$Savings
dim(BD)# check the number of dimensions: 13
```

You can also use mutate function from the dplyr package to add the new variable:

```
BD2 <- BD %>%
  mutate(Total=Checking+Savings)

View(BD2)
```

With mutate, you can add more then 1 variable at a time. Read more about it here: https://dplyr.tidyverse.org/reference/mutate.html

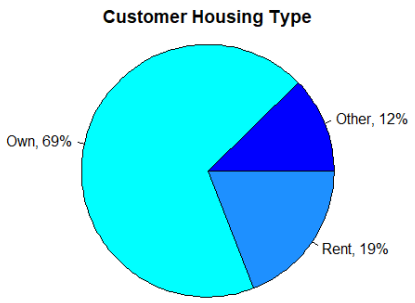
QUESTION 1A CUSTOMER PROFILE DASHBOARD

ii. Generate a chart and table to view the distributions of each of the above customer demographics variables: Housing, Job, Credit Risk, Months Employed and Total.

Housing

- Categorical
- Can use pie chart/barplot

```
HouseFreq<-BD%>%count(Housing) # getting counts
pie(HouseFreq$n, labels = c("Other","Own","Rent")) # bare bones pie chart

kable(HouseFreq, caption = "Frequency of Bank Customers by Housing") # view in table form

slice.house <- HouseFreq$n # get counts in vector form
house.piepercent <- 100*round(HouseFreq$n/sum(HouseFreq$n),2) # compute percentage
label<-HouseFreq$Housing # extract housing labels
label<-paste(label,",",sep="")
label<-paste(label,house.piepercent) #default of sep=" "
label<-paste(label,"%",sep="")
pie(slice.house,
    labels=label,
    col=c("blue","cyan","dodgerblue"),
    radius=1,
    main="Customer Housing Type") # build piechart
```



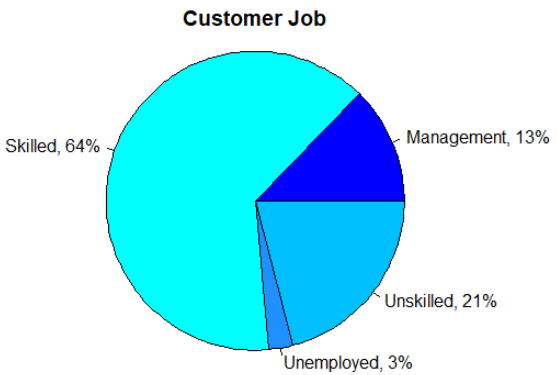
QUESTION 1A CUSTOMER PROFILE DASHBOARD

ii. Generate a chart and table to view the distributions of each of the above customer demographics variables: Housing, Job, Credit Risk, Months Employed and Total.

Job

- **Categorical**
- **Can use pie chart/barplot**

```
JobFreq<-BD%>%count(Job)
kable(JobFreq, caption = "Frequency of Bank Customers by Job")
slice.job <- JobFreq$n
job.piepercent <- 100*round(JobFreq$n/sum(JobFreq$n),2)
label<-JobFreq$Job
label<-paste(label,",",sep="")
label<-paste(label,job.piepercent) #default of sep=" "
label<-paste(label,"%",sep="")
pie(slice.job,
    labels=label,
    col=c("blue","cyan", "dodgerblue", "deepskyblue"),
    radius=1,
    main="Customer Job")
```



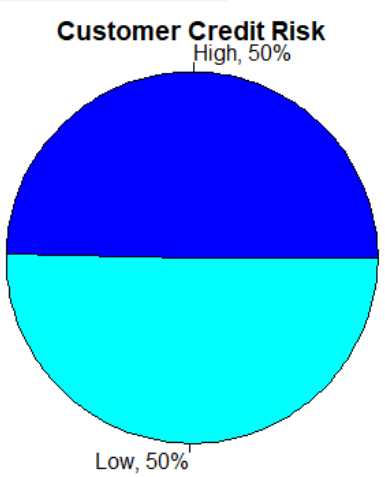
QUESTION 1A CUSTOMER PROFILE DASHBOARD

ii. Generate a chart and table to view the distributions of each of the above customer demographics variables: Housing, Job, Credit Risk, Months Employed and Total.

Credit Risk

- **Categorical**
- **Can use pie chart/barplot**

```
crFreq<-BD%>%count(`Credit Risk`)
kable(crFreq, caption = "Frequency of Bank Customers by Credit Risk")
slice.cr <- crFreq$n
cr.piepercent <- 100*round(crFreq$n/sum(crFreq$n),2)
label<-crFreq$`Credit Risk`
label<-paste(label,",",sep="")
label<-paste(label,cr.piepercent) #default of sep=" "
label<-paste(label,"%",sep="")
pie(slice.cr,
    labels=label,
    col=c("blue","cyan"),
    radius=1,
    main="Customer Credit Risk")
```



ADDITIONAL NOTES ON PIE CHARTS: OPTIONS FOR CREATING THE LABELS

1

```
label<-HouseFreq$Housing # extract housing labels
label<-paste(label,",",sep="")
label<-paste(label,house.piepercent) #default of sep=" "
label<-paste(label,"%",sep="")
```

2

```
label <- HouseFreq$Housing %>%
  paste(",",sep="") %>%
  paste(house.piepercent) %>%
  paste("%",sep="")
label
```

3

```
label <- glue::glue("{HouseFreq$Housing}, {house.piepercent}%")
label
```

Means use the glue function from the glue package

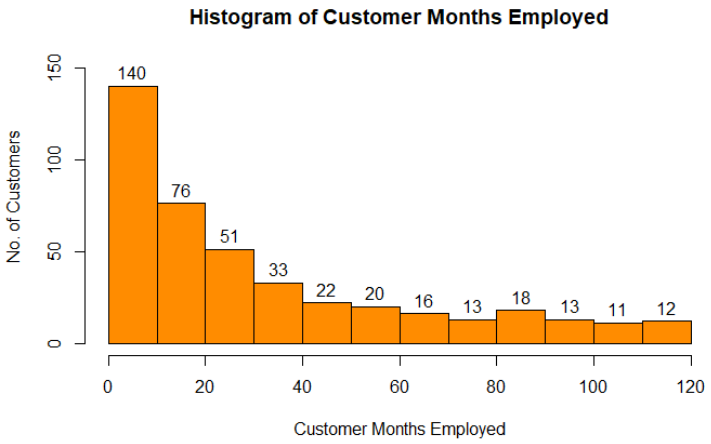
QUESTION 1A CUSTOMER PROFILE DASHBOARD

ii. Generate a chart and table to view the distributions of each of the above customer demographics variables: Housing, Job, Credit Risk, Months Employed and Total.

Months

- **Continuous**
- **Can use histogram**

```
h.em<-hist(BD$`Months Employed`,
           main="Histogram of Customer Months Employed",
           xlab="Customer Months Employed",
           ylab="No. of Customers",
           col=c("darkorange"),
           ylim=c(0,160),
           labels=TRUE)#adds numbers on each bin
```



QUESTION 1A CUSTOMER PROFILE DASHBOARD

ii. Generate a chart and table to view the distributions of each of the above customer demographics variables: Housing, Job, Credit Risk, Months Employed and Total.

Frequency distribution by Months Employed

Emp.Group	Freq
(0,10]	115
(10,20]	76
(20,30]	51
(30,40]	33
(40,50]	22
(50,60]	20
(60,70]	16
(70,80]	13
(80,90]	18
(90,100]	13
(100,110]	11
(110,120]	12

Months

- **Continuous**
- **Can use histogram**
- **You can extract frequency table from histogram**

```
Emp.Group<-cut(BD$`Months Employed`,h.em$breaks) # binning
t.emp<-table(Emp.Group)
kable(t.emp, caption = "Frequency distribution by Months Employed")
```

"( )" is not inclusive. For example (0, 2) means all values ranging between 0 and 2 not including 0 and 2.

"[]" is inclusive. For example [0, 2] means all values ranging between 0 and 2 including 0 and 2.

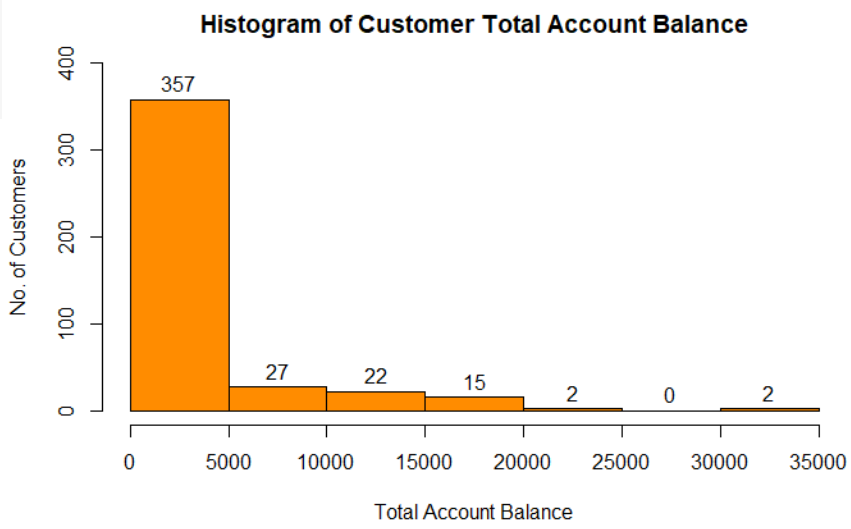
QUESTION 1A CUSTOMER PROFILE DASHBOARD

ii. Generate a chart and table to view the distributions of each of the above customer demographics variables: Housing, Job, Credit Risk, Months Employed and Total.

Total Account Balance

- **Continuous**
- **Can use histogram**

```
h.tot<-hist(BD$Total,
            main="Histogram of Customer Total Account Balance",
            xlab="Total Account Balance",
            ylab="No. of Customers",
            col=c("darkorange"),
            ylim=c(0,400),
            labels=TRUE)
```

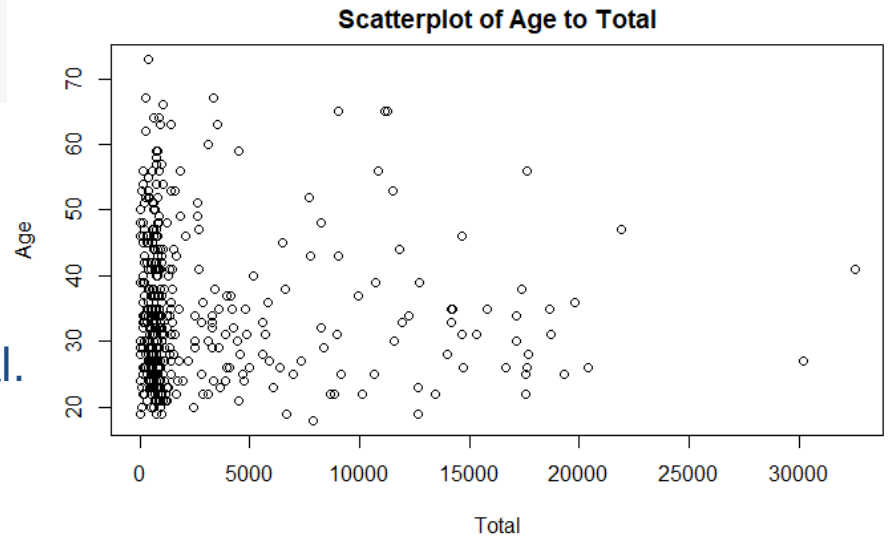


QUESTION 1A CUSTOMER PROFILE DASHBOARD

iii. Generate the appropriate charts to display the relationship between Total and Months Employed as well as Total and Age

```
plot(BD$Total,BD$Age,
     main="Scatterplot of Age to Total",
     ylab="Age",
     xlab="Total")

plot(BD$Total,BD$`Months Employed`,
     main="Scatterplot of Months Employed to Total",
     ylab="Months Employed",
     xlab="Total")
```

- Association and not causality
- Doesn't matter which variable is on either axis.
- Run the code for the relationship between months employed and total. What is the relationship?



QUESTION 1B: CUSTOMER CREDIT RISK ANALYSES DASHBOARD

i. The credit risk analysts are interested in understanding the demographics of customers with different levels of Credit Risk. They would like to be able to see the appropriate charts and tables to compare Credit Risk with Job as well as Credit Risk with Housing. They think a stacked barplot might provide a good visualization. Could you develop this dashboard for them?

QUESTION 1B: CUSTOMER CREDIT RISK ANALYSES DASHBOARD

i. CREDIT RISK AND JOB (step 1)

```
BDb1 <- BD %>%
  group_by(`Credit Risk`, Job) %>%
  tally()
```

tally can be replaced with count

A tibble: 8 x 3 | Groups: Credit Risk [2]

Credit Risk <chr>	Job <chr>	n <int>
High	Management	28
High	Skilled	135
High	Unemployed	5
High	Unskilled	43
Low	Management	26
Low	Skilled	136
Low	Unemployed	6
Low	Unskilled	46

8 rows

This is a long data format. We will use the spread () function from the tidry package to convert it to a wide data format

see next slide for step 2

QUESTION 1B: CUSTOMER CREDIT RISK ANALYSES DASHBOARD

i. CREDIT RISK AND JOB (step 2)

```
# change from long to wide form
BDb1.spread<- BDb1 %>%
  spread(key=`Job`,value=n)
```

spread is from tidyr package

```
# A tibble: 8 x 3
# Groups:   Credit Risk [2]
  `Credit Risk` Job             n
  <chr>         <chr>       <int>
1 High          Management     28
2 High          Skilled       135
3 High          Unemployed      5
4 High          Unskilled      43
5 Low           Management     26
6 Low           Skilled       136
7 Low           Unemployed      6
8 Low           Unskilled      46
```

```
## # A tibble: 2 x 5
## # Groups:   Credit Risk [2]
##   `Credit Risk` Management Skilled Unemployed Unskilled
##   <chr>              <int>   <int>      <int>      <int>
## 1 High                  28     135          5         43
## 2 Low                   26     136          6         46
```

QUESTION 1B: CUSTOMER CREDIT RISK ANALYSES DASHBOARD

i. CREDIT RISK AND JOB (step 3)

```
kable(BDb1.spread, caption = "Contingency table for Credit Risk and Job")
```

Contingency table for Credit Risk and Job

Credit Risk	Management	Skilled	Unemployed	Unskilled
High	28	135	5	43
Low	26	136	6	46

There are 5 columns in total: credit risk. Management, skills, unemployed & unskilled

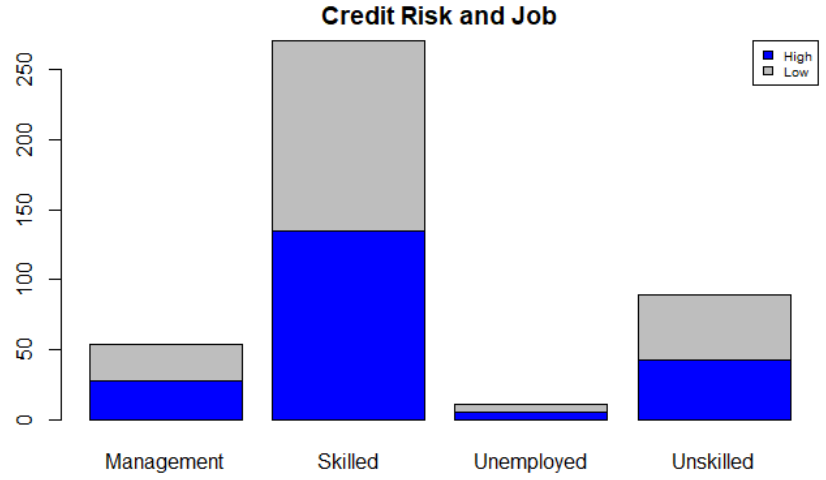
- We need columns 2-5 (the job types)

```
#extract and convert the 2nd to 5th columns into a matrix
#Plot the grouped stack barplot
barmatrix.BDb1<-as.matrix(BDb1.spread[,c(2:5)])
bar_Col1<-c("blue","gray")
barplot(barmatrix.BDb1,
        col=bar_Col1,
        main="Credit Risk and Job")
legend("topright",
       cex=0.6,
       fill=bar_Col1,
       BDb1.spread$`Credit Risk`)
```

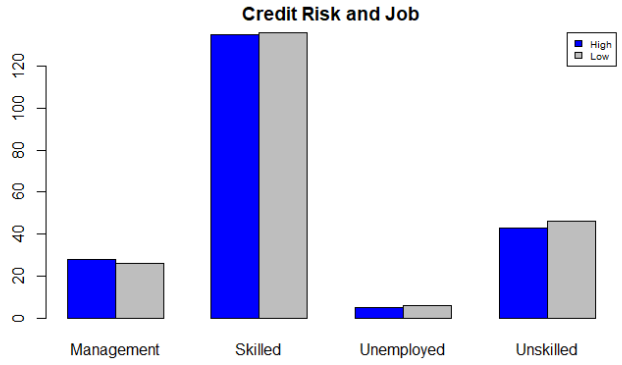
QUESTION 1B: CUSTOMER CREDIT RISK ANALYSES DASHBOARD

i. CREDIT RISK AND JOB (final output)

```
barmatrix.BDb1<-as.matrix(BDb1.spread[,c(2:5)])
bar_Col1<-c("blue","gray")
barplot(barmatrix.BDb1,
        col=bar_Col1,
        main="Credit Risk and Job")
legend("topright",
       cex=0.6,
       fill=bar_Col1,
       BDb1.spread$`Credit Risk`)
```




```
barplot(barmatrix.BDb1,
        col=bar_Col1,
        main="Credit Risk and Job",
        beside=TRUE)
legend("topright",
       cex=0.6,
       fill=bar_Col1,
       BDb1.spread$`Credit Risk`)
```



QUESTION 1B: CUSTOMER CREDIT RISK ANALYSES DASHBOARD

i. CREDIT RISK AND HOUSING (step 1)

```
BDb2 <- BD %>%
  group_by(`Credit Risk`, Housing) %>%
  count()
```

count can be replaced with tally

A tibble: 6 x 3 | Groups: Credit Risk, Housing [6]

Credit Risk <chr>	Housing <chr>	n <int>
High	Other	31
High	Own	131
High	Rent	49
Low	Other	21
Low	Own	161
Low	Rent	32

6 rows

This is a long data format. We will use the spread () function from the tidry package to convert it to a wide data format

see next slide for step 2

QUESTION 1B: CUSTOMER CREDIT RISK ANALYSES DASHBOARD

i. CREDIT RISK AND HOUSING (step 2)

```
BDb2.spread<- BDb2 %>%
  spread(key=`Housing`,value=n)
```

spread is from tidyr package

A tibble: 6 x 3 Groups: Credit Risk, Housing [6]

Credit Risk <chr>	Housing <chr>	n <int>
High	Other	31
High	Own	131
High	Rent	49
Low	Other	21
Low	Own	161
Low	Rent	32

6 rows

A tibble: 2 x 4 Groups: Credit Risk [2]

Credit Risk <chr>	Other <int>	Own <int>	Rent <int>
High	31	131	49
Low	21	161	32

2 rows

QUESTION 1B: CUSTOMER CREDIT RISK ANALYSES DASHBOARD

i. CREDIT RISK AND HOUSING (step 3)

```
kable(BDb2.spread, caption = "Contingency table for Credit Risk and Housing")
```

Contingency table for Credit Risk and Housing

Credit Risk	Other	Own	Rent
High	31	131	49
Low	21	161	32

There are 4 columns in total: credit risk, Other, Own, Rent

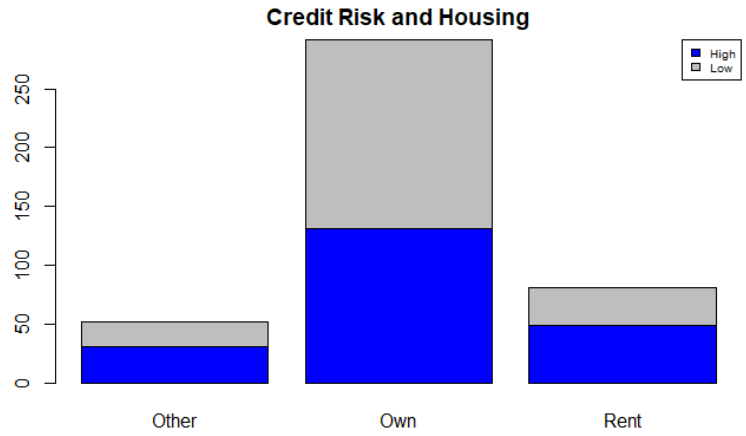
- We need columns 2-4 (the housing types)

```
#plot the grouped stack barplot
#extract and convert the 2nd to 4th columns into a matrix
barmatrix.BDb2<-as.matrix(BDb2.spread[,c(2:4)])
barplot(barmatrix.BDb2,
        col=bar_Col1,
        main="Credit Risk and Housing")
legend("topright",
       cex=0.6,
       fill=bar_Col1,
       BDb2.spread$`Credit Risk`)
```

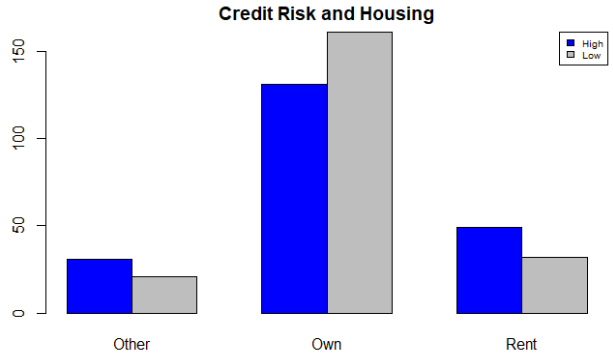
QUESTION 1B: CUSTOMER CREDIT RISK ANALYSES DASHBOARD

i. CREDIT RISK AND HOUSING (final output)

```
barmatrix.BDb2<-as.matrix(BDb2.spread[,c(2:4)])
barplot(barmatrix.BDb2,
        col=bar_Col1,
        main="Credit Risk and Housing")
legend("topright",
       cex=0.6,
       fill=bar_Col1,
       BDb2.spread$`Credit Risk`)
```




```
barplot(barmatrix.BDb2,
        col=bar_Col1,
        main="Credit Risk and Housing",
        beside=TRUE)
legend("topright",
       cex=0.6,
       fill=bar_Col1,
       BDb2.spread$`Credit Risk`)
```



Always interpret the charts

The differences for frequency of Jobs type between High and Low Credit Risk is very minimal. It is hard to visualize this using stacked barplot.

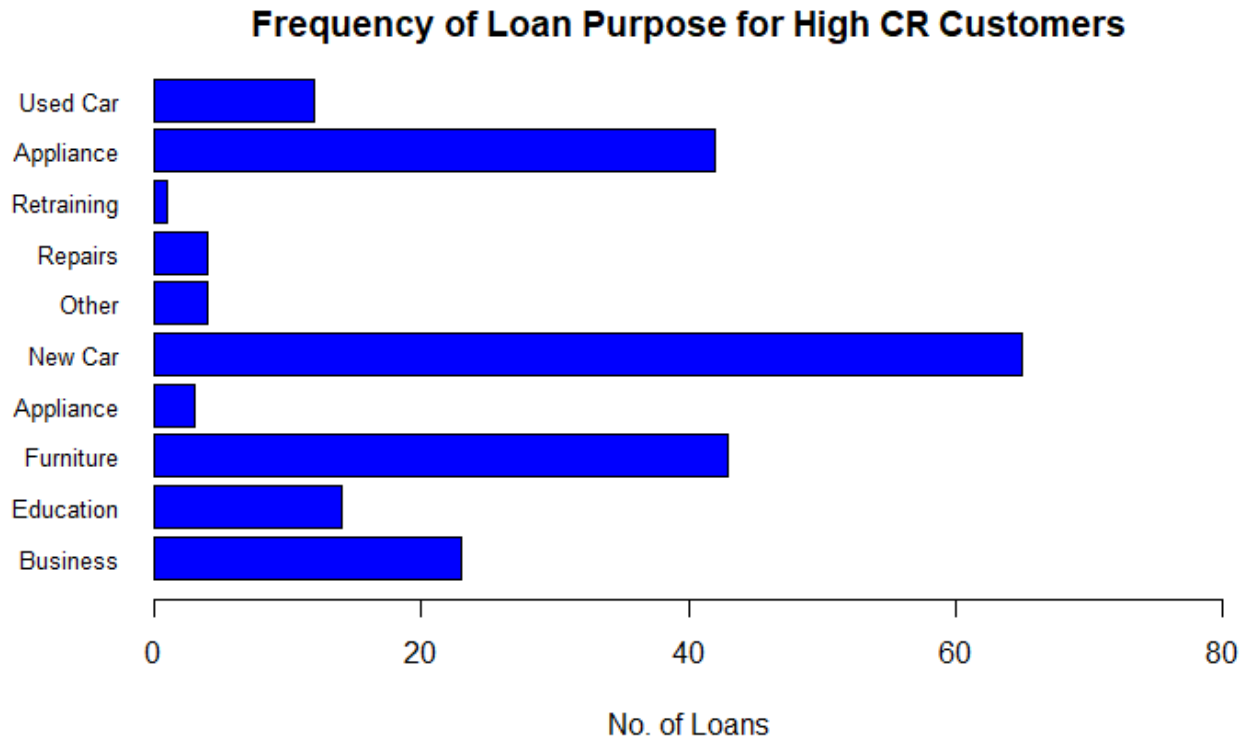
QUESTION 1C: CUSTOMER LOAN ANALYSES DASHBOARD

i. The credit risk analysts are interested in understanding the Loan Purpose of customers with "High" levels of Credit Risk. Could you generate the table and chart for them to visualize the distribution of Loan Purpose for "High" Credit Risk customers?

```
#extract records for High Credit Risk
LoanHRFreq<-BD%>%
  filter (`Credit Risk` == "High") %>%
  count(`Loan Purpose`)

kable(LoanHRFreq, caption = "Frequency Distribution for Loan Purpose for High CR
Customers")

LoanHRbar <- LoanHRFreq$n

# Horizontal
barplot(LoanHRbar,
        names.arg=LoanHRFreq$`Loan Purpose`,
        col="blue",
        beside = TRUE,
        main="Frequency of Loan Purpose for High CR Customers",
        cex.names = 0.8,
        xlim=c(0,80),
        xlab="No. of Loans",
        horiz=TRUE, las=1)
```

QUESTION 1C: CUSTOMER LOAN ANALYSES DASHBOARD

i. The credit risk analysts are interested in understanding the Loan Purpose of customers with "High" levels of Credit Risk. Could you generate the table and chart for them to visualize the distribution of Loan Purpose for "High" Credit Risk customers?



Code on previous slide

Always interpret the charts

Most common loan is New Car. Least common is Retraining.

QUESTION 1D: CUSTOMER ACCOUNT BALANCE PARETO ANALYSES

i. The credit risk analyses would like to conduct pareto analyses on `Total` to understand if there is a small proportion of customers that contribute to significant amount of total account balances with the bank. Could you help to generate the analyses?

- Sort from Richest to Poorest (remember last tutorial!)
- Compute Cumulative Percentage (cumsum)
- Find out how much savings the richest 20% have!
- Let's Try it in R now!

QUESTION 1D: CUSTOMER ACCOUNT BALANCE PARETO ANALYSES

```
#extract only the Total column and sort in descending order
BD.tot<-BD %>% select (Total)%>% arrange(desc(Total))

#compute the percentage of savings over total savings
BD.tot$Percentage<-BD.tot$Total/sum(BD.tot$Total)

#compute cumulative percentage for Total
BD.tot$Cumulative<-cumsum(BD.tot$Percentage)

#compute cumulative percentage of customers from top most savings
BD.tot$Cumulative.cust<-as.numeric(rownames(BD))/nrow(BD)

# compute percentage ofcustomers with top 80% savings
101/nrow(BD)
```

THANK YOU. SEE YOU NEXT WEEK.