Assignment\_1

Sanket Praveen Patil

2023-10-01

# Problem 1

# Importing data in R

Data = read.csv("adult.csv", header = T)  
head(Data)

## age workclass fnlwgt education education.num marital.status  
## 1 39 State-gov 77516 Bachelors 13 Never-married  
## 2 50 Self-emp-not-inc 83311 Bachelors 13 Married-civ-spouse  
## 3 38 Private 215646 HS-grad 9 Divorced  
## 4 53 Private 234721 11th 7 Married-civ-spouse  
## 5 28 Private 338409 Bachelors 13 Married-civ-spouse  
## 6 37 Private 284582 Masters 14 Married-civ-spouse  
## occupation relationship race sex capital.gain capital.loss  
## 1 Adm-clerical Not-in-family White Male 2174 0  
## 2 Exec-managerial Husband White Male 0 0  
## 3 Handlers-cleaners Not-in-family White Male 0 0  
## 4 Handlers-cleaners Husband Black Male 0 0  
## 5 Prof-specialty Wife Black Female 0 0  
## 6 Exec-managerial Wife White Female 0 0  
## hours.per.week native.country income.bracket  
## 1 40 United-States <=50K  
## 2 13 United-States <=50K  
## 3 40 United-States <=50K  
## 4 40 United-States <=50K  
## 5 40 Cuba <=50K  
## 6 40 United-States <=50K

# Question a :

summary(Data)

## age workclass fnlwgt education   
## Min. :17.00 Length:32561 Min. : 12285 Length:32561   
## 1st Qu.:28.00 Class :character 1st Qu.: 117827 Class :character   
## Median :37.00 Mode :character Median : 178356 Mode :character   
## Mean :38.58 Mean : 189778   
## 3rd Qu.:48.00 3rd Qu.: 237051   
## Max. :90.00 Max. :1484705   
## education.num marital.status occupation relationship   
## Min. : 1.00 Length:32561 Length:32561 Length:32561   
## 1st Qu.: 9.00 Class :character Class :character Class :character   
## Median :10.00 Mode :character Mode :character Mode :character   
## Mean :10.08   
## 3rd Qu.:12.00   
## Max. :16.00   
## race sex capital.gain capital.loss   
## Length:32561 Length:32561 Min. : 0 Min. : 0.0   
## Class :character Class :character 1st Qu.: 0 1st Qu.: 0.0   
## Mode :character Mode :character Median : 0 Median : 0.0   
## Mean : 1078 Mean : 87.3   
## 3rd Qu.: 0 3rd Qu.: 0.0   
## Max. :99999 Max. :4356.0   
## hours.per.week native.country income.bracket   
## Min. : 1.00 Length:32561 Length:32561   
## 1st Qu.:40.00 Class :character Class :character   
## Median :40.00 Mode :character Mode :character   
## Mean :40.44   
## 3rd Qu.:45.00   
## Max. :99.00

summary(Data$age)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 17.00 28.00 37.00 38.58 48.00 90.00

# For variable “age”, we can see that the 1st Quartile value i.e. at 25th percenile is 28

# Value at 75th percentile is 48 and mean is 38.58. Magnitude of 25th percentile and 75th percentile is around 10 units away from the mean.

# Hence we can say by looking at summary of “age” variable, data is slightly normally distributed.

# Also there are no NA values in the data.

summary(Data$hours.per.week)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.00 40.00 40.00 40.44 45.00 99.00

# For variable “hours.per.week”, 1st Quartile value is 40, 3rd quartile value is 45 and mean is 40.44.

# As value at 75th percentile is more units away from mean than 25th percentile, we can guess that data is left skewed.

# No NA values present in the data.

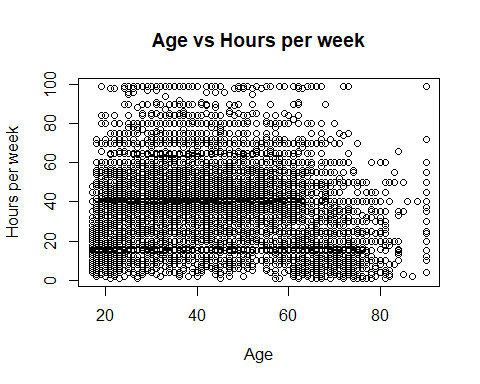
library(psych)  
describe(Data$hours.per.week)

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 32561 40.44 12.35 40 40.55 4.45 1 99 98 0.23 2.92 0.07

# Question b :

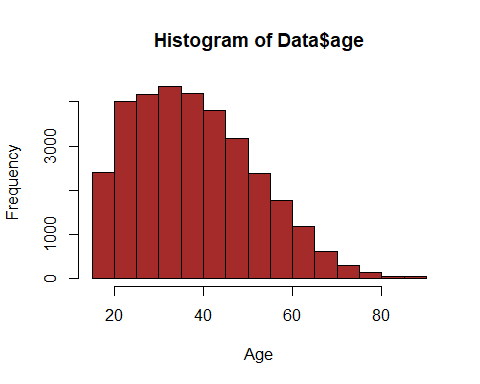
# As both the variables are numeric, we can use scatterplot to compare them.

plot(x = Data$age,y = Data$hours.per.week, xlab = "Age", ylab = "Hours per week", main = "Age vs Hours per week")

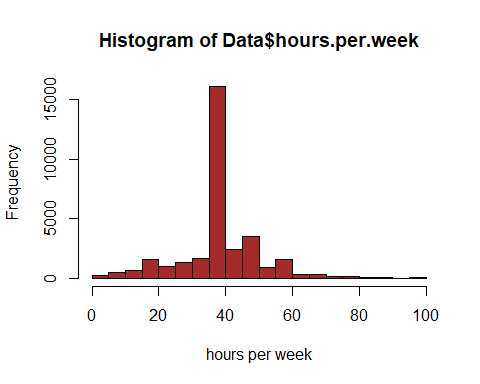


* By looking at the scatter plot, we can conclude that there is no linear relation between the two variables.

hist(Data$age,xlab = "Age",col = "brown",border = "black")



hist(Data$hours.per.week,xlab = "hours per week",col = "brown",border = "black")



* By looking at the histograms, we can say our assumptions based on summary were wrong. Data for variable “age” is right skewed.
* For variable “hours.per.week”, there are more data points near value = 40. Hence we cannot conlude about the normality of the data.

# Question c :

library("dplyr")

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

# Checking which variables are numeric and which variables are categorical  
  
num\_cols <- lapply(Data, is.numeric)  
print(num\_cols)

## $age  
## [1] TRUE  
##   
## $workclass  
## [1] FALSE  
##   
## $fnlwgt  
## [1] TRUE  
##   
## $education  
## [1] FALSE  
##   
## $education.num  
## [1] TRUE  
##   
## $marital.status  
## [1] FALSE  
##   
## $occupation  
## [1] FALSE  
##   
## $relationship  
## [1] FALSE  
##   
## $race  
## [1] FALSE  
##   
## $sex  
## [1] FALSE  
##   
## $capital.gain  
## [1] TRUE  
##   
## $capital.loss  
## [1] TRUE  
##   
## $hours.per.week  
## [1] TRUE  
##   
## $native.country  
## [1] FALSE  
##   
## $income.bracket  
## [1] FALSE

# With the help of lapply function, we found following numeric columns.

age

fnlwgt

education.num

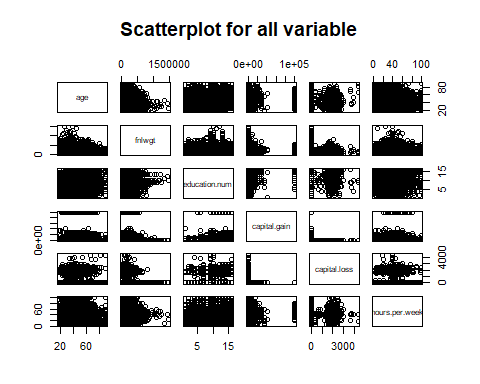
capital.gain

capital.loss

hours.per.week

# Creating scatterplot matrix for all numeric variables.

pairs(~age+fnlwgt+education.num+capital.gain+capital.loss+hours.per.week,data = Data,main = "Scatterplot for all variable")



* It will be more difficult if we create separate scatterplot for two variables.
* But if we create scatterplot matrix, we can compare linear relation between all numeric variables in a single plot.
* By looking at the plot, we can conclude there is no linear relation between any of the variable.

# Question d :

# We have already differentiated the categorical and numerical variables in below variable.  
print(num\_cols)

## $age  
## [1] TRUE  
##   
## $workclass  
## [1] FALSE  
##   
## $fnlwgt  
## [1] TRUE  
##   
## $education  
## [1] FALSE  
##   
## $education.num  
## [1] TRUE  
##   
## $marital.status  
## [1] FALSE  
##   
## $occupation  
## [1] FALSE  
##   
## $relationship  
## [1] FALSE  
##   
## $race  
## [1] FALSE  
##   
## $sex  
## [1] FALSE  
##   
## $capital.gain  
## [1] TRUE  
##   
## $capital.loss  
## [1] TRUE  
##   
## $hours.per.week  
## [1] TRUE  
##   
## $native.country  
## [1] FALSE  
##   
## $income.bracket  
## [1] FALSE

workclass

education

marital.status

occupation

relationship

race

sex

native.country

income.bracket

# Use barcharts to visualize the distribution

WorkClass <- table(Data$workclass)  
prop.table(WorkClass)

##   
## ? Federal-gov Local-gov Never-worked   
## 0.0563864746 0.0294831240 0.0642793526 0.0002149811   
## Private Self-emp-inc Self-emp-not-inc State-gov   
## 0.6970301895 0.0342741316 0.0780381438 0.0398636406   
## Without-pay   
## 0.0004299622

# Save the proportion table in an object   
workclass.ptb <- prop.table(WorkClass)  
  
# Convert the proportion table to a data frame  
WorkClass.df <- as.data.frame(workclass.ptb)  
  
# Converting to a data frame lost the names  
names(WorkClass.df) <- c("Class", "Percent")  
  
library(scales)

##   
## Attaching package: 'scales'

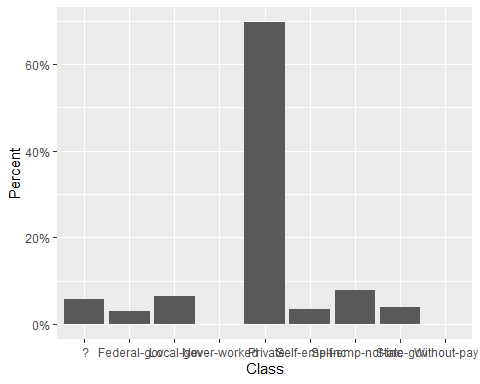
## The following objects are masked from 'package:psych':  
##   
## alpha, rescale

library(ggplot2)

##   
## Attaching package: 'ggplot2'

## The following objects are masked from 'package:psych':  
##   
## %+%, alpha

# Use the proportion table as the data frame in a call to ggplot()  
ggplot(data=WorkClass.df, mapping=aes(x=Class, y=Percent)) + geom\_col() + scale\_y\_continuous(label=percent)



library(tidyverse)

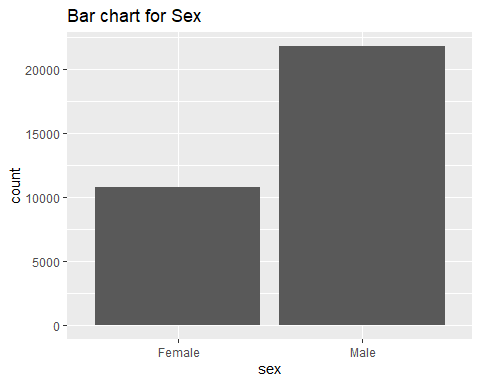
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ lubridate 1.9.3 ✔ tibble 3.2.1  
## ✔ purrr 1.0.2 ✔ tidyr 1.3.0  
## ✔ readr 2.1.4

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ ggplot2::%+%() masks psych::%+%()  
## ✖ ggplot2::alpha() masks scales::alpha(), psych::alpha()  
## ✖ readr::col\_factor() masks scales::col\_factor()  
## ✖ purrr::discard() masks scales::discard()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(ggplot2)  
Data %>% group\_by(sex) %>% count()

## # A tibble: 2 × 2  
## # Groups: sex [2]  
## sex n  
## <chr> <int>  
## 1 " Female" 10771  
## 2 " Male" 21790

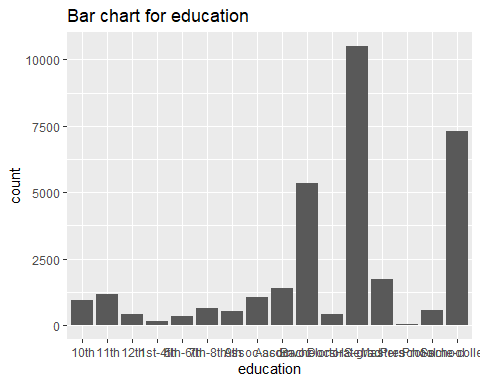
Data %>% ggplot(aes(x = sex)) + geom\_bar() + labs(title = "Bar chart for Sex")



library(ggplot2)  
Data %>% group\_by(education) %>% count()

## # A tibble: 16 × 2  
## # Groups: education [16]  
## education n  
## <chr> <int>  
## 1 " 10th" 933  
## 2 " 11th" 1175  
## 3 " 12th" 433  
## 4 " 1st-4th" 168  
## 5 " 5th-6th" 333  
## 6 " 7th-8th" 646  
## 7 " 9th" 514  
## 8 " Assoc-acdm" 1067  
## 9 " Assoc-voc" 1382  
## 10 " Bachelors" 5355  
## 11 " Doctorate" 413  
## 12 " HS-grad" 10501  
## 13 " Masters" 1723  
## 14 " Preschool" 51  
## 15 " Prof-school" 576  
## 16 " Some-college" 7291

Data %>% ggplot(aes(x = education)) + geom\_bar() + labs(title = "Bar chart for education")

 # Question e :

comparison\_matrix <- table(Data$education,Data$income.bracket)  
comparison\_matrix

##   
## <=50K >50K  
## 10th 871 62  
## 11th 1115 60  
## 12th 400 33  
## 1st-4th 162 6  
## 5th-6th 317 16  
## 7th-8th 606 40  
## 9th 487 27  
## Assoc-acdm 802 265  
## Assoc-voc 1021 361  
## Bachelors 3134 2221  
## Doctorate 107 306  
## HS-grad 8826 1675  
## Masters 764 959  
## Preschool 51 0  
## Prof-school 153 423  
## Some-college 5904 1387

# As per the cross tabulation, we can see that people with Bachelors degree have most number of records which fall in >50K category.

# At education level “Preschool”, there are 0 records who fall under >50K category.

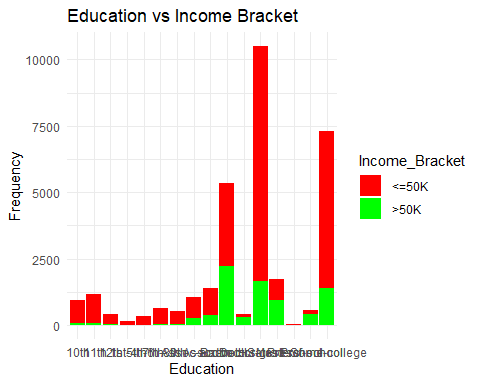
# There are highest number of records with education level “HS-grad”

comparison\_matrix <- as.data.frame(comparison\_matrix)  
print(comparison\_matrix)

## Var1 Var2 Freq  
## 1 10th <=50K 871  
## 2 11th <=50K 1115  
## 3 12th <=50K 400  
## 4 1st-4th <=50K 162  
## 5 5th-6th <=50K 317  
## 6 7th-8th <=50K 606  
## 7 9th <=50K 487  
## 8 Assoc-acdm <=50K 802  
## 9 Assoc-voc <=50K 1021  
## 10 Bachelors <=50K 3134  
## 11 Doctorate <=50K 107  
## 12 HS-grad <=50K 8826  
## 13 Masters <=50K 764  
## 14 Preschool <=50K 51  
## 15 Prof-school <=50K 153  
## 16 Some-college <=50K 5904  
## 17 10th >50K 62  
## 18 11th >50K 60  
## 19 12th >50K 33  
## 20 1st-4th >50K 6  
## 21 5th-6th >50K 16  
## 22 7th-8th >50K 40  
## 23 9th >50K 27  
## 24 Assoc-acdm >50K 265  
## 25 Assoc-voc >50K 361  
## 26 Bachelors >50K 2221  
## 27 Doctorate >50K 306  
## 28 HS-grad >50K 1675  
## 29 Masters >50K 959  
## 30 Preschool >50K 0  
## 31 Prof-school >50K 423  
## 32 Some-college >50K 1387

names(comparison\_matrix) <- c("Education", "Income\_Bracket", "Frequency")

library(ggplot2)  
  
ggplot(comparison\_matrix, aes(x = Education, y = Frequency, fill = Income\_Bracket)) + geom\_bar(stat = "identity") + labs(x = "Education", y = "Frequency", title = "Education vs Income Bracket") + scale\_fill\_manual(values = c(" <=50K" = "red", " >50K" = "green")) + theme\_minimal()



* As per the above plot, we can see that people with Bachelors degree have most number of records which fall in >50K category.
* There are 0 records who fall under >50K category for education = “Preschool”.
* The observations we made based on cross tabulation are matching with the observations based on the bar chart.

# Problem 2

# Question 1 :

**Reading both given datasets**

Even\_Data = read.csv("population\_even.csv", header = T)  
Odd\_Data = read.csv("population\_odd.csv", header = T)

**Merging two datasets into one single dataset.**

df = merge(x = Even\_Data, y = Odd\_Data, by = "STATE")#,ll.x = TRUE)  
head(df)

## STATE NAME.x POPESTIMATE2010 POPESTIMATE2012 POPESTIMATE2014  
## 1 1 Alabama 4785437 4815588 4841799  
## 2 2 Alaska 713910 730443 736283  
## 3 4 Arizona 6407172 6554978 6730413  
## 4 5 Arkansas 2921964 2952164 2967392  
## 5 6 California 37319502 37948800 38596972  
## 6 8 Colorado 5047349 5192647 5350101  
## POPESTIMATE2016 POPESTIMATE2018 NAME.y POPESTIMATE2011 POPESTIMATE2013  
## 1 4863525 4887681 Alabama 4799069 4830081  
## 2 741456 735139 Alaska 722128 737068  
## 3 6941072 7158024 Arizona NA 6632764  
## 4 2989918 3009733 Arkansas 2940667 2959400  
## 5 39167117 39461588 California 37638369 38260787  
## 6 5539215 5691287 Colorado 5121108 5269035  
## POPESTIMATE2015 POPESTIMATE2017 POPESTIMATE2019  
## 1 4852347 4874486 4903185  
## 2 737498 739700 731545  
## 3 6829676 7044008 7278717  
## 4 2978048 3001345 3017804  
## 5 38918045 39358497 39512223  
## 6 5450623 5611885 5758736

# We joined data based on common variable named “STATE”.

# Question 2 :

**a)**

colnames(df)

## [1] "STATE" "NAME.x" "POPESTIMATE2010" "POPESTIMATE2012"  
## [5] "POPESTIMATE2014" "POPESTIMATE2016" "POPESTIMATE2018" "NAME.y"   
## [9] "POPESTIMATE2011" "POPESTIMATE2013" "POPESTIMATE2015" "POPESTIMATE2017"  
## [13] "POPESTIMATE2019"

# There is no duplicate “STATE” column created in the joining process.

# But there are 2 variables with same values which are NAME.x and NAME.y. SO we will remove NAME.y

df <- within(df, rm(NAME.y))  
colnames(df)

## [1] "STATE" "NAME.x" "POPESTIMATE2010" "POPESTIMATE2012"  
## [5] "POPESTIMATE2014" "POPESTIMATE2016" "POPESTIMATE2018" "POPESTIMATE2011"  
## [9] "POPESTIMATE2013" "POPESTIMATE2015" "POPESTIMATE2017" "POPESTIMATE2019"

# Removing POPESTIMATE from all variables and keeping only year values such as 2010, 2011 etc.

df <- df %>% rename("2010" = "POPESTIMATE2010",  
 "2011" = "POPESTIMATE2011",  
 "2012" = "POPESTIMATE2012",  
 "2013" = "POPESTIMATE2013",  
 "2014" = "POPESTIMATE2014",  
 "2015" = "POPESTIMATE2015",  
 "2016" = "POPESTIMATE2016",  
 "2017" = "POPESTIMATE2017",  
 "2018" = "POPESTIMATE2018",  
 "2019" = "POPESTIMATE2019"  
)  
colnames(df)

## [1] "STATE" "NAME.x" "2010" "2012" "2014" "2016" "2018" "2011"   
## [9] "2013" "2015" "2017" "2019"

# Reordering data by Year

df<-df %>% select(order(colnames(df)))  
head(df)

## 2010 2011 2012 2013 2014 2015 2016 2017  
## 1 4785437 4799069 4815588 4830081 4841799 4852347 4863525 4874486  
## 2 713910 722128 730443 737068 736283 737498 741456 739700  
## 3 6407172 NA 6554978 6632764 6730413 6829676 6941072 7044008  
## 4 2921964 2940667 2952164 2959400 2967392 2978048 2989918 3001345  
## 5 37319502 37638369 37948800 38260787 38596972 38918045 39167117 39358497  
## 6 5047349 5121108 5192647 5269035 5350101 5450623 5539215 5611885  
## 2018 2019 NAME.x STATE  
## 1 4887681 4903185 Alabama 1  
## 2 735139 731545 Alaska 2  
## 3 7158024 7278717 Arizona 4  
## 4 3009733 3017804 Arkansas 5  
## 5 39461588 39512223 California 6  
## 6 5691287 5758736 Colorado 8

# Question 3

# Checking if dataframe has NA values or not.

sum(is.na(df))

## [1] 5

which(is.na(df))

## [1] 55 192 273 391 518

# Finding which column has the missing values.

sapply(df, function(x) sum(is.na(x)))

## 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 NAME.x   
## 0 1 0 1 0 1 0 1 0 1 0   
## STATE   
## 0

# Finding which state has NA value in a respective column.

df$NAME.x[is.na(df$'2011')]

## [1] "Arizona"

# Replacing NA values for Column “POPESTIMATE2011” :

df$'2011'[is.na(df$'2011')]<-mean(c(df$'2010'[df$NAME.x==df$NAME.x[is.na(df$'2011')]],df$'2012'[df$NAME.x==df$NAME.x[is.na(df$'2011')]]))

# Replacing NA values for Column “2013” :

df$'2013'[is.na(df$'2013')]<-mean(c(df$'2012'[df$NAME.x==df$NAME.x[is.na(df$'2013')]],df$'2014'[df$NAME.x==df$NAME.x[is.na(df$'2013')]]))

# Replacing NA values for Column “2015” :

df$'2015'[is.na(df$'2015')]<-mean(c(df$'2014'[df$NAME.x==df$NAME.x[is.na(df$'2015')]],df$'2016'[df$NAME.x==df$NAME.x[is.na(df$'2015')]]))

# Replacing NA values for Column “2017” :

df$'2017'[is.na(df$'2017')]<-mean(c(df$'2016'[df$NAME.x==df$NAME.x[is.na(df$'2017')]],df$'2018'[df$NAME.x==df$NAME.x[is.na(df$'2017')]]))

# Replacing NA values for Column “2019” :

df$'2019'[is.na(df$'2019')]<-mean(c(df$'2017'[df$NAME.x==df$NAME.x[is.na(df$'2019')]],df$'2018'[df$NAME.x==df$NAME.x[is.na(df$'2019')]]))

# Checking if all missing values replaced or not.

sapply(df, function(x) sum(is.na(x)))

## 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 NAME.x   
## 0 0 0 0 0 0 0 0 0 0 0   
## STATE   
## 0

# All missing values replaced as given.

# Question 4 :

# Get the maximum population for every state

max\_population\_year <- df %>% rowwise() %>% mutate(Max\_Population = max(c\_across(starts\_with("20"))))  
head(max\_population\_year)

## # A tibble: 6 × 13  
## # Rowwise:   
## `2010` `2011` `2012` `2013` `2014` `2015` `2016` `2017` `2018` `2019` NAME.x  
## <int> <dbl> <int> <dbl> <int> <dbl> <int> <dbl> <int> <dbl> <chr>   
## 1 4785437 4.80e6 4.82e6 4.83e6 4.84e6 4.85e6 4.86e6 4.87e6 4.89e6 4.90e6 Alaba…  
## 2 713910 7.22e5 7.30e5 7.37e5 7.36e5 7.37e5 7.41e5 7.40e5 7.35e5 7.32e5 Alaska  
## 3 6407172 6.48e6 6.55e6 6.63e6 6.73e6 6.83e6 6.94e6 7.04e6 7.16e6 7.28e6 Arizo…  
## 4 2921964 2.94e6 2.95e6 2.96e6 2.97e6 2.98e6 2.99e6 3.00e6 3.01e6 3.02e6 Arkan…  
## 5 37319502 3.76e7 3.79e7 3.83e7 3.86e7 3.89e7 3.92e7 3.94e7 3.95e7 3.95e7 Calif…  
## 6 5047349 5.12e6 5.19e6 5.27e6 5.35e6 5.45e6 5.54e6 5.61e6 5.69e6 5.76e6 Color…  
## # ℹ 2 more variables: STATE <int>, Max\_Population <dbl>

# To get the sum of population statewise

total\_population\_year <- df %>% rowwise() %>% mutate(total\_population = sum(c\_across(starts\_with("20"))))  
head(total\_population\_year)

## # A tibble: 6 × 13  
## # Rowwise:   
## `2010` `2011` `2012` `2013` `2014` `2015` `2016` `2017` `2018` `2019` NAME.x  
## <int> <dbl> <int> <dbl> <int> <dbl> <int> <dbl> <int> <dbl> <chr>   
## 1 4785437 4.80e6 4.82e6 4.83e6 4.84e6 4.85e6 4.86e6 4.87e6 4.89e6 4.90e6 Alaba…  
## 2 713910 7.22e5 7.30e5 7.37e5 7.36e5 7.37e5 7.41e5 7.40e5 7.35e5 7.32e5 Alaska  
## 3 6407172 6.48e6 6.55e6 6.63e6 6.73e6 6.83e6 6.94e6 7.04e6 7.16e6 7.28e6 Arizo…  
## 4 2921964 2.94e6 2.95e6 2.96e6 2.97e6 2.98e6 2.99e6 3.00e6 3.01e6 3.02e6 Arkan…  
## 5 37319502 3.76e7 3.79e7 3.83e7 3.86e7 3.89e7 3.92e7 3.94e7 3.95e7 3.95e7 Calif…  
## 6 5047349 5.12e6 5.19e6 5.27e6 5.35e6 5.45e6 5.54e6 5.61e6 5.69e6 5.76e6 Color…  
## # ℹ 2 more variables: STATE <int>, total\_population <dbl>

# Refer last column to see max population and total population state wise. We just replaced max function by sum function to get sum of the population.

# Question 5:

# Get the total US population for one single year

total\_us\_population\_2010 <- sum(df$'2010')  
total\_us\_population\_2010

## [1] 313043191

# 

# Problem 3

# Reshape the data to have year and population columns

library(tidyverse)  
df\_new <- df %>%  
 pivot\_longer(cols = starts\_with("20"), names\_to = "year", values\_to = "population") %>%  
 mutate(year = as.integer(str\_extract(year, "\\d+")))   
print(df\_new)

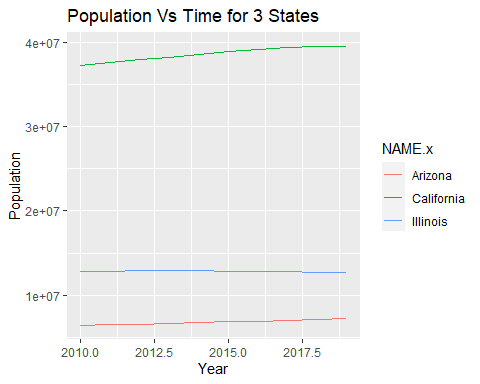
## # A tibble: 520 × 4  
## NAME.x STATE year population  
## <chr> <int> <int> <dbl>  
## 1 Alabama 1 2010 4785437  
## 2 Alabama 1 2011 4799069  
## 3 Alabama 1 2012 4815588  
## 4 Alabama 1 2013 4830081  
## 5 Alabama 1 2014 4841799  
## 6 Alabama 1 2015 4852347  
## 7 Alabama 1 2016 4863525  
## 8 Alabama 1 2017 4874486  
## 9 Alabama 1 2018 4887681  
## 10 Alabama 1 2019 4903185  
## # ℹ 510 more rows

# Choosing 3 states for aalysis

selected\_states <- c("Arizona", "California", "Illinois")

# Filtering the data

df\_states <- df\_new %>% filter(NAME.x %in% selected\_states)  
  
library(ggplot2)  
  
ggplot(df\_states, aes(x = year, y = population, color = NAME.x)) + geom\_line() + labs(title = "Population Vs Time for 3 States", x = "Year", y = "Population")



* As per above line graph, we can conclude that Population for Illinois state is not varied since 2010 whereas population for California increased gradually.
* Population of Arizona also increased slightly.

# Prolem 4

**Question a :**

1. Missing values

Issue - The accuracy of the analysis can be significantly affected by missing values since they can result in incorrect conclusions about the data.

Solution - We can use imputation methods such as mean imputation, KNN Imputation, Forward fill etc. Also if the number of rows impacted by NA are less than 2% of overall rows, we can simply remove those rows from data which are having NA values.

1. Outliers

Issue - The outcomes of statistical modelling and data analysis might be affected by outliers. The mean and the standard deviation may be significantly impacted. Normality may be reduced if the outliers aren’t distributed randomly.

Solution - First detect outliers by following methods.

1. IQR method
2. Standard deviation method

There are several approaches to handle outliers, such as capping, removing them or by treating them as missing values.

# Question b :

1. We can use clustering algorithms to group customers who buy similar things. We can use K-Means clustering, KNN etc.
2. Binary outcomes, such whether a consumer would purchase milk, can be predicted using classification algorithms like decision trees, random forest or logistic regression.
3. We can use Association Rule Mining technique here to find which products are purchased more. We can discover interesting relationships or patterns in large datasets.

# Question c :

1. Organizing the customers of a company according to education level.

-> Yes this is a data mining task as we are categorizing customers based on some of the attributes such as education level. We can pull out outcomes from it such as which education level has more number of customers etc.

1. Computing the total sales of a company.

-> No, this is not a data mining task as it doesn’t involve discovering patterns or relationships in the data.

1. Sorting a student database according to identification numbers.

-> We cannot say this is a data mining task as its just ordering our data based on some of the attributes. We are not discovering any patterns here.

1. Predicting the outcomes of tossing a (fair) pair of dice.

-> It is a process of predicting something but we cannot say this is a data mining process as we are not discovering any relationships or patterns here. We are just predicting some of the outcomes.

1. Predicting the future stock price of a company using historical records.

-> Yes, this is a data mining process as we need to analyse the historical data first, we need a lot of preprocessing of the data if required. Also we need to find patterns and relations between variables to decide if we can keep those variables or not. And then we can predict the future stock price.