



## CSE 240 R Data Science with R

### Project Report

Name : Bhuvann V

Unique ID : E0420018

Year : II

Quarter : Q2

Department : B. Tech MED

Faculty Name : Prof. Ramya M

Academic Year : 2020-2021

## CONTENTS

S.NO	TOPICS	PAGE NO.
1.	PROBLEM STATEMENT	3
2.	OBJECTIVE	4
3.	DATA LOADING	5
4.	DATA EXPLORATION	6
5.	DATA CLEANING	9
6.	DATA VISUALIZATION	11
7.	ML ALGORITHM MODEL	13
8.	RESULT	15

# Students Performance in Exam

## **PROBLEM STATEMENT**

This dataset contains the students details and the marks of a particular subject based on their performance. We need to predict the mark of the student based on the marks which was awarded to them based on their skills

## **OBJECTIVE :**

The main objective of this project:

To outline contains the marks of the student after completing an exam and estimated marks based on their skill.

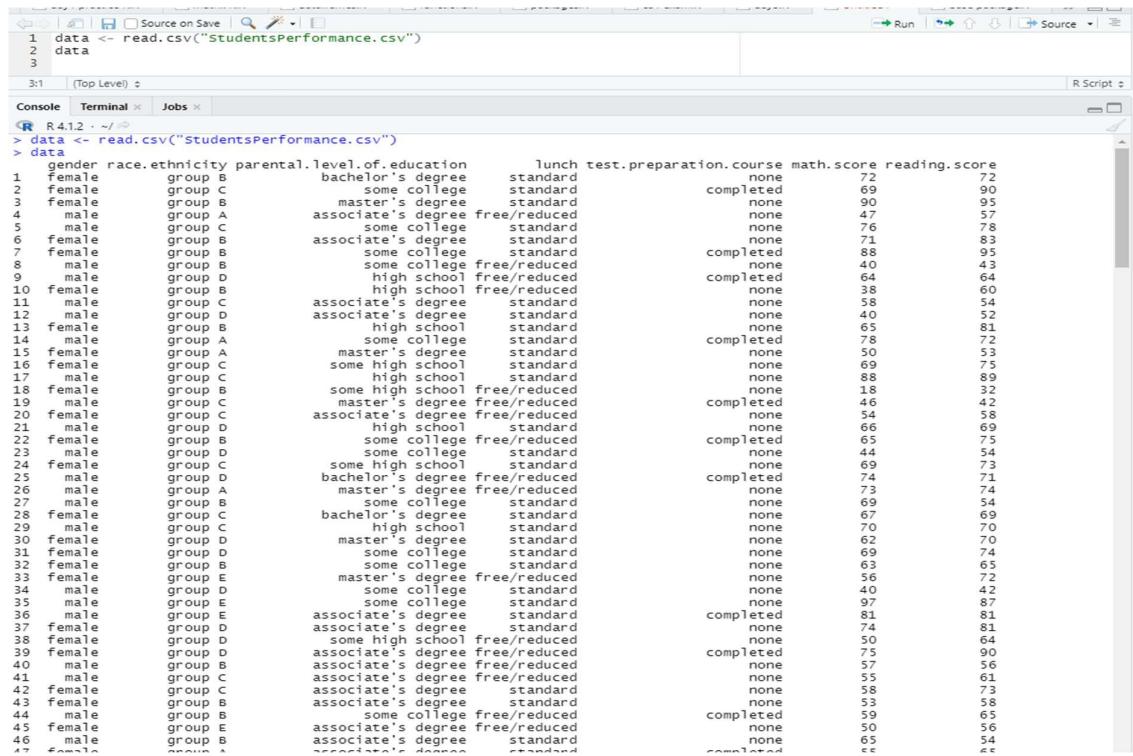
To select appropriate prediction model algorithm that combines

several features to estimate the mark of the student.

To track the mark of the student based on correct model.

To forecast and visualize the mark obtained based on the dataset

## DATA LOADING



The screenshot shows the RStudio interface with the following code in the console:

```
R 4.1.2 · ~/RStudio  
> data <- read.csv("StudentsPerformance.csv")  
> data
```

The data frame 'data' contains 46 rows and 9 columns. The columns are:

	gender	race.ethnicity	parental.level.of.education	lunch	test.preparation.course	math.score	reading.score
1	female	group B	bachelor's degree	standard	none	72	72
2	female	group C	some college	standard	completed	69	90
3	female	group B	master's degree	standard	none	90	95
4	male	group A	associate's degree	free/reduced	none	47	57
5	male	group C	some college	standard	none	76	78
6	female	group B	associate's degree	standard	none	71	83
7	female	group B	some college	standard	completed	88	95
8	male	group B	some college	free/reduced	none	40	43
9	male	group D	high school	free/reduced	completed	64	64
10	female	group B	high school	free/reduced	none	38	60
11	male	group C	associate's degree	standard	none	58	54
12	male	group D	associate's degree	standard	none	40	52
13	female	group B	high school	standard	none	65	81
14	male	group A	some college	standard	completed	78	72
15	female	group A	master's degree	standard	none	50	53
16	female	group C	some high school	standard	none	69	75
17	male	group C	high school	standard	none	88	89
18	female	group B	some high school	free/reduced	none	18	32
19	male	group C	master's degree	free/reduced	completed	46	42
20	female	group C	associate's degree	free/reduced	none	54	58
21	male	group D	high school	standard	none	66	69
22	female	group B	some college	free/reduced	completed	65	76
23	male	group D	some college	standard	none	44	54
24	female	group C	some high school	standard	none	69	73
25	male	group D	bachelor's degree	free/reduced	completed	74	71
26	male	group A	master's degree	free/reduced	none	73	74
27	male	group B	some college	standard	none	69	54
28	female	group C	bachelor's degree	standard	none	67	69
29	male	group C	high school	standard	none	70	70
30	female	group D	master's degree	standard	none	62	70
31	female	group D	some college	standard	none	69	74
32	female	group B	some college	standard	none	63	65
33	female	group E	master's degree	free/reduced	none	56	72
34	male	group D	some college	standard	none	40	42
35	male	group E	some college	standard	none	97	87
36	male	group E	associate's degree	standard	completed	81	81
37	female	group D	associate's degree	standard	none	74	81
38	female	group D	some high school	free/reduced	none	50	64
39	female	group D	associate's degree	free/reduced	completed	75	90
40	male	group B	associate's degree	free/reduced	none	57	56
41	male	group C	associate's degree	free/reduced	none	55	61
42	female	group C	associate's degree	standard	none	58	73
43	female	group B	associate's degree	standard	none	53	58
44	male	group B	some college	free/reduced	completed	59	65
45	female	group E	associate's degree	free/reduced	none	50	56
46	male	group B	associate's degree	standard	none	65	54
47	female	group X	associate's degree	standard	completed	66	66

```
R 4.1.2 ~/ 
> data <- read.csv("StudentsPerformance.csv")
> head(data)
  gender race.ethnicity parental.level.of.education      lunch test.preparation.course math.score reading.score
1 female    group B        bachelor's degree standard          none       72        72
2 female    group C        some college   standard completed      69        90
3 female    group B        master's degree standard          none       90        95
4 male     group A        associate's degree free/reduced none       47        57
5 male     group C        some college   standard          none       76        78
6 female    group B        associate's degree standard          none       71        83
  writing.score
1             74
2             88
3             93
4             44
5             75
6             78
> |
```

THIS DATASET IS STUDENT PERFORMANCE DATASET, WHICH IS  
SAVED AS A CSV FILE, LOADED AND VIEWED IN THE WORKING  
ENVIRONMENT

## DATA EXPLORATION

### 1) Str(data)

```
Console Terminal Jobs 
R 4.1.2 ~/ 
> str(data)
'data.frame': 1000 obs. of 8 variables:
 $ gender           : chr "female" "female" "female" "male" ...
 $ race.ethnicity   : chr "group B" "group C" "group B" "group A" ...
 $ parental.level.of.education: chr "bachelor's degree" "some college" "master's degree" "associate's degree" ...
 $ lunch            : chr "standard" "standard" "standard" "free/reduced" ...
 $ test.preparation.course: chr "none" "completed" "none" "none" ...
 $ math.score        : int 72 69 90 47 76 71 88 40 64 38 ...
 $ reading.score    : int 72 90 95 57 78 83 95 43 64 60 ...
 $ writing.score    : int 74 88 93 44 75 78 92 39 67 50 ...
> |
```

## 2) summary(data)

```
Console Terminal Jobs 
R 4.1.2 : ~/ 
> summary(data)
  gender      race.ethnicity    parental.level.of.education   lunch      test.preparation.course
  Length:1000  Length:1000      Length:1000          Length:1000  Length:1000
  Class :character  Class :character  Class :character  Class :character  Class :character
  Mode  :character  Mode  :character  Mode  :character  Mode  :character  Mode  :character

  math.score    reading.score    writing.score
  Min.   : 0.00  Min.   :17.00  Min.   :10.00
  1st Qu.: 57.00 1st Qu.: 59.00 1st Qu.: 57.75
  Median : 66.00 Median : 70.00 Median : 69.00
  Mean   : 66.09 Mean   : 69.17 Mean   : 68.05
  3rd Qu.: 77.00 3rd Qu.: 79.00 3rd Qu.: 79.00
  Max.   :100.00 Max.   :100.00 Max.   :100.00
> |
```

## 3) dim(data)

```
Console Terminal Jobs 
R 4.1.2 : ~/ 
> dim(data)
[1] 1000   8
> |
```

DIMENTION FUNCTION RETURNING THE COUNT OF  
TOTAL NUMBER OF ROWS AND COLUMNS IN THE DATA  
SET AND HEAD FUNCTION DISPLAYING THE FIRST FIVE  
ROWS

#### 4)head(data)

```
Console | terminal | jobs |
R 4.1.2 - ~/ |
> head(data)
   gender race.ethnicity parental.level.of.education      lunch test.preparation.course math.score reading.score
1 female    group B           bachelor's degree standard       none        72        72
2 female    group C           some college     standard completed      69        90
3 female    group B           master's degree  standard       none        90        95
4 male      group A           associate's degree free/reduced      none        47        57
5 male      group C           some college     standard       none        76        78
6 female    group B           associate's degree standard      none        71        83
writing.score
1          74
2          88
3          93
4          44
5          75
6          78
> |
```

#### 5) tail(data)

```
Console | terminal | jobs |
R 4.1.2 - ~/ |
> tail(data)
   gender race.ethnicity parental.level.of.education      lunch test.preparation.course math.score reading.score
995 male    group A           high school    standard       none        63        63
996 female  group E           master's degree standard completed      88        99
997 male    group C           high school   free/reduced      none        62        55
998 female  group C           high school   free/reduced completed      59        71
999 female  group D           some college  standard completed      68        78
1000 female group D           some college  free/reduced      none        77        86
writing.score
995      62
996      95
997      55
998      65
999      77
1000     86
```

```
Console Terminal Jobs
R 4.1.2 · ~ · ⌂
> colnames(data) <- c("gender", "race_ethnicity", "parental_level_of_education", "lunch", "test_preparation_course", "math_score", "reading_score", "writing score")
> head(data)
   gender race_ethnicity parental_level_of_education      lunch test_preparation_course math_score reading_score
1 female    group B           bachelor's degree standard        none       72          72
2 female    group C            some college standard completed     69          90
3 female    group B           master's degree standard        none       90          95
4 male      group A associate's degree free/reduced      none       47          57
5 male      group C            some college standard        none       76          78
6 female    group B           associate's degree standard        none       71          83
   writing score
1             74
2             88
3             93
4             44
5             75
6             78
> |
```

## Data Cleaning:

The screenshot shows the RStudio interface with two main panes. The left pane, titled 'R Script', contains R code for data cleaning and visualization. The right pane, titled 'Data', lists the objects available in the global environment, including datasets like 'anestest', 'authors', 'books', 'data', 'datas', 'df', 'emails', 'house', 'iris', 'knf', 'mat', and 'mat2'. The 'Data' pane also includes a search bar and a sidebar with project-related links.

```
install.packages("tidyverse")
library(tidyverse)

# reading in the data
df = read.csv("studentsPerformance.csv")
# taking a quick look
glimpse(df)
# is.na(df$math_score)
# is.na(df$reading_score)
# is.na(df$writing_score)
# is.na(df$dtest.preparation.course)
# is.na(df$math)
# df = df %>
#   mutate(test.preparation.course = replace(test.preparation.course, test.preparation.course == "none", NA)) %>%
#   mutate(test.preparation.course = replace(test.preparation.course, test.preparation.course == "N/A", NA))
# is.na(dtest.preparation.course)
# df = df %>
#   mutate(replace(~., with = NA))
#   mutate(test.preparation.course = replace(test.preparation.course, is.na(test.preparation.course), "unavailable"))
# dtest.preparation.course
# data visual
# scatter.smooth(~df$math_score,y=df$reading_score,xlab="math score",
# ylab="reading score",
# color="red",pch=19)
```

# Code:

The screenshot shows the RStudio interface with the following details:

- Header:** RStudio, File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Toolbar:** Go to file/function, Addins.
- Script Editor:** The main workspace contains an R script with the following code:

```
1 install.packages("tidyverse")
2 library(tidyverse)
3
4
5 # reading in the data
6 df <- read.csv("studentsPerformance.csv")
7 # taking a quick look
8 glimpse(df)
9
10 is.na(df$math.score)
11 is.na(df$reading.score)
12 is.na(df$writing.score)
13 is.na(df$test.preparation.course)
14 #non standard na
15 df <- df %>%
16   mutate(test.preparation.course = replace(test.preparation.course, test.preparation.course == "none", NA)) %>%
17   mutate(test.preparation.course = replace(test.preparation.course, test.preparation.course == "N/A", NA))
18
19 is.na(df$test.preparation.course)
20
21 # replacing "--" with NA
22 df <- df %>%
23   mutate(test.preparation.course= replace(test.preparation.course, is.na(test.preparation.course), "unavailable"))
24 df$test.preparation.course
25
26 #data visual
27 scatter.smooth(x=df$math.score,y=df$reading.score,xlab="maths score",
28                  ylab="reading score",
29                  col="red",pch=19)
30
32 scatter.smooth(x=df$math.score,y=df$writing.score,xlab="maths score",
33                  ylab="writing score",
34                  col="blue",pch=19)
35
36 #box
37
38 install.packages("vioplot")
39 library("vioplot")
40
41 vioplot(df$math.score-df$gender ,col=2:length(levels(df$gender )))
42 #bar
43 couts <- table(df$gender , df$race.ethnicity )
44 barplot(couts,main = "simple com",xlab = "gears", legend=rownames(couts),col = c("red","blue"))
45 #hist
46 getwd()
47 x = read.csv("studentsPerformance.csv")
48 x
49 hist(x$math.score,
50       main = "Marks in maths",
51       xlab = "Students in nos.",
52       ylab = "Marks",
53       las = 1,
54       col = c("skyblue", "chocolate2"))
55 )
```

The code is for data cleaning and visualization. It starts by installing and loading the tidyverse package. It then reads a CSV file named "studentsPerformance.csv". The script handles missing values (NA) and replaces them with appropriate values or "unavailable". It performs two scatter plots comparing math and reading scores, and writing and reading scores. Finally, it creates a box plot for gender differences in math scores and a histogram for marks in maths.

# Data Visualization: Scatter Plot:

The screenshot shows the RStudio interface with two panes. The left pane displays an R script with code for data manipulation and plotting. The right pane shows the R Environment view with a scatter plot of 'writing score' versus 'math score'.

**R Script:**

```
13 is.na(df$test.preparation.course)
14 df$test.preparation.course[is.na(df$test.preparation.course)] <- "none"
15 df <- df %>
16   mutate(test.preparation.course = replace(test.preparation.course, test.preparation.course == "none", NA)) %>%
17   mutate(test.preparation.course = replace(test.preparation.course, test.preparation.course == "/n'", NA))
18 is.na(df$test.preparation.course)
19 df %>
20   # replacing "-" with NA
21   df %>%
22     mutate(test.preparation.course= replace(test.preparation.course, is.na(test.preparation.course), "unavailable"))
23 df$test.preparation.course
24
25 #data visual
26 scatter.smooth(x=df$math.score,y=df$reading.score,xlab="maths score",
27                 ylab="reading score",
28                 col="red",pch=19)
29
30 scatter.smooth(x=df$math.score,y=df$writing.score,
31                 ylab="writing score",
32                 col="blue",pch=19)
33
34
35 install.packages("vtiplot")
36 library("vtiplot")
37
38 vtiplot(df$math.score-df$gender ,col=2:length(levels(df$gender )))
39
40 vtiplot(df$math.score-df$race.ethnicity )
41
42 couts <- table(df$gender ,df$race.ethnicity )
43
44 couts
```

**R Environment:**

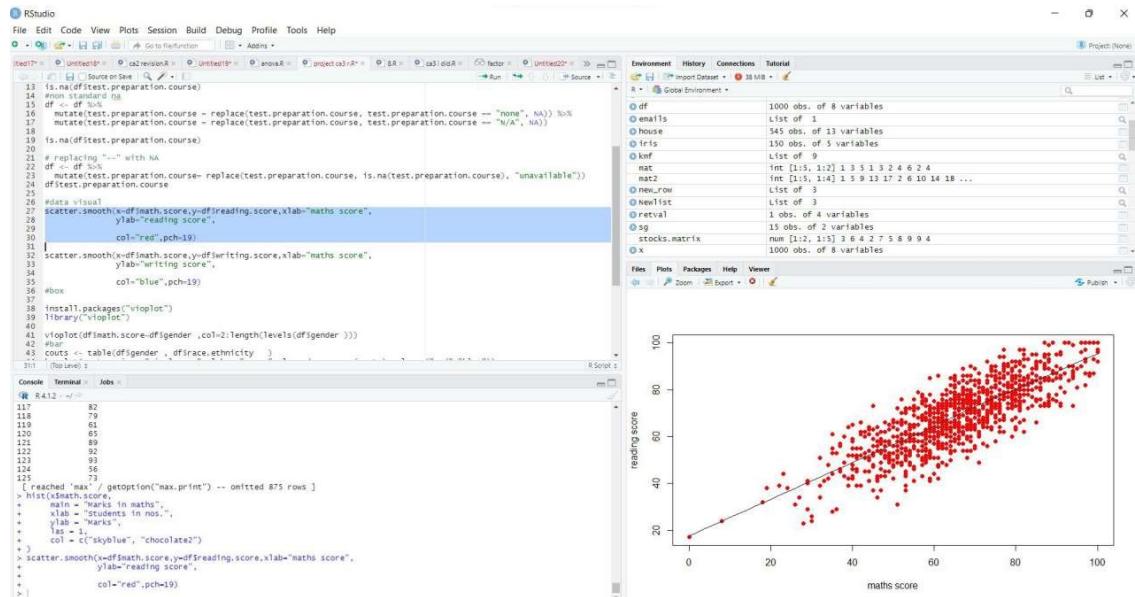
df  
emails  
house  
irls  
kmf  
mat  
mt  
new\_col  
newlist  
retval  
sg  
stocks.matrix  
x

1000 obs. of 8 variables  
List of 1  
545 obs. of 13 variables  
150 obs. of 5 variables  
List of 9  
int [1:15, 1:2] 3 5 1 3 2 4 2 4  
int [1:15, 1:4] 1 5 9 13 2 7 6 10 14 18 ...  
List of 3  
List of 3  
1 obs. of 4 variables  
15 obs. of 2 variables  
num [1:12, 1:5] 3 6 4 2 7 5 8 9 9 4  
1000 obs. of 8 variables

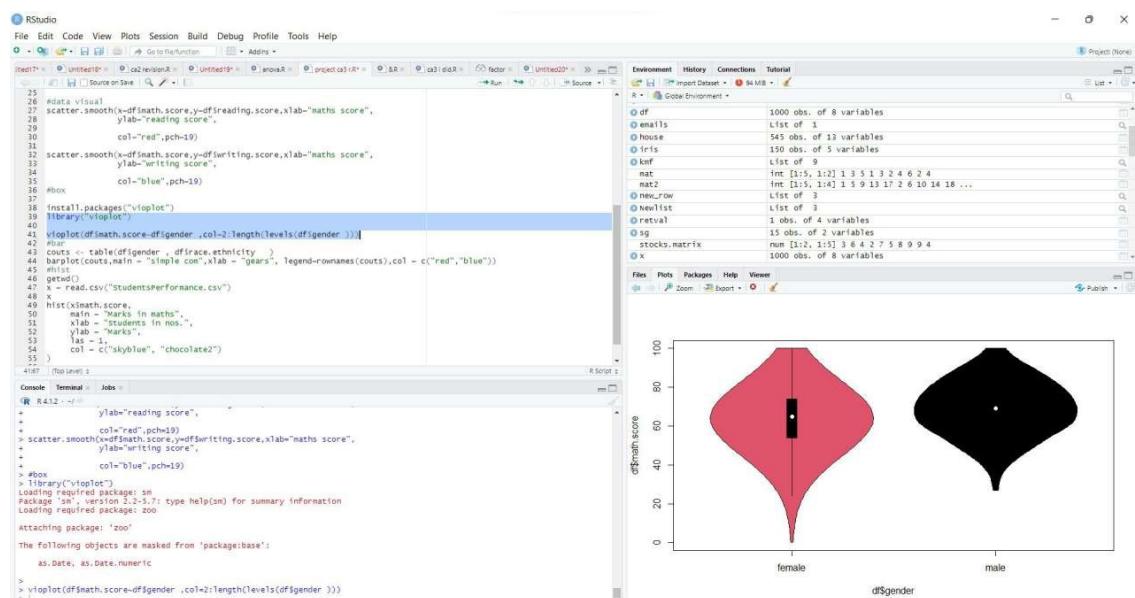
Files Plots Packages Help Viewer

**Plots:**

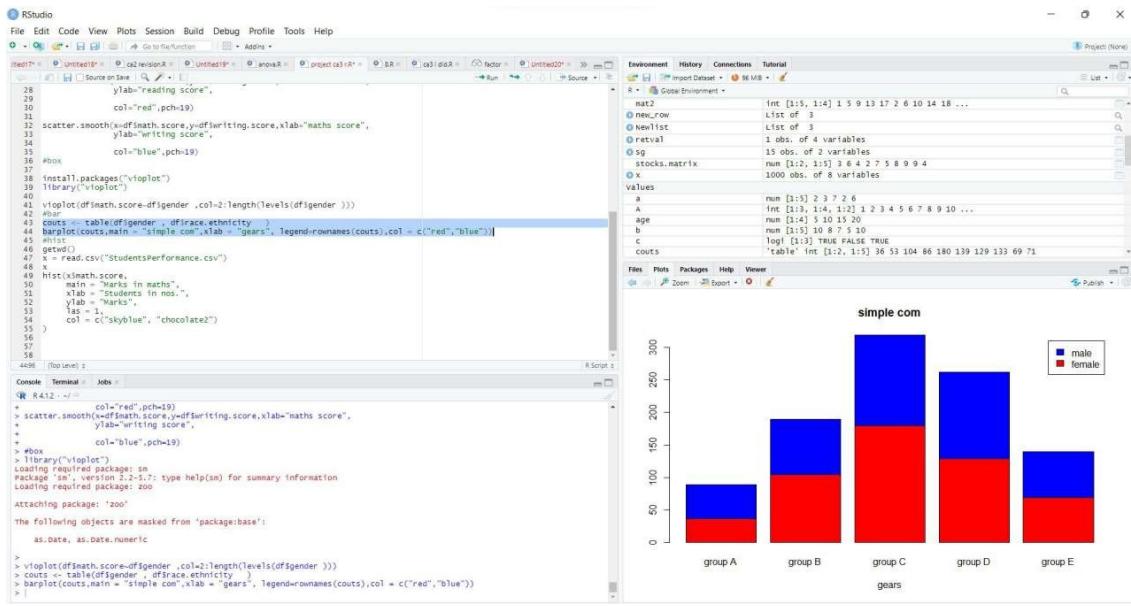
A scatter plot titled 'writing score' vs 'math score'. The x-axis ranges from 0 to 100, and the y-axis ranges from 0 to 100. A positive linear regression line is drawn through the data points, which are colored red (math score) and blue (writing score).

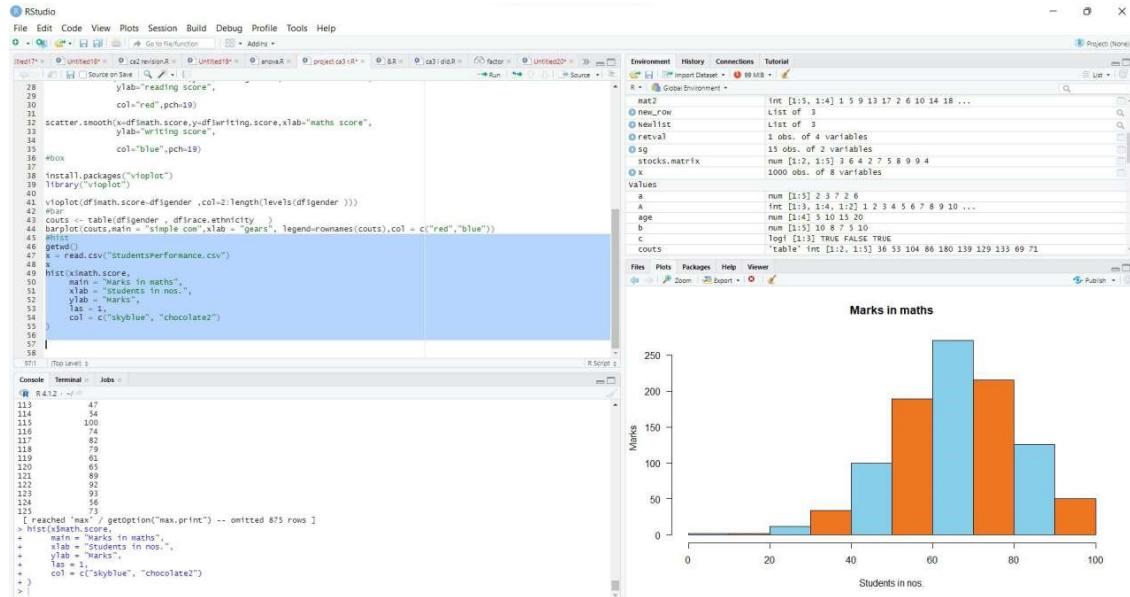


## Violin Plot:



# Bar Chart:





# ML Algorithm model:

## SINGLE LINEAR REGRESSION

## #BASIC LINEAR REGRESSION

## #PREDICTING MATH SCORE W/

```

1 data <- read.csv("studentperformance.csv")
2 dput(data)
3 str(data)
4 summary(data)
5 dim(data)
6 head(data)
7 tail(data)
8 ncol(data)
9 colnames(data) <- c("gender", "race_ethnicity", "parental_level_of_education", "lunch", "test_preparation_course", "math_score", "reading_score")
10 head(data)
11 tail(data)
12 model <- lm(math_score ~ reading_score, data = data)
13 int(model)
14

```

Call:  
`lm(formula = math_score ~ reading_score, data = data)`

Coefficients:

(Intercept)	reading_score
7.3576	0.8491

#THE MODEL PREDICTS THE FORMULA BELOW:

$$\text{#MATH\_SCORE} = 7.3576 + 0.8491 * \text{READING\_SCORE}$$

```

14 summary(model)
15

```

Call:  
`summary(model)`

Residuals:

Min	1Q	Median	3Q	Max
-24.3419	-6.3419	-0.0221	6.2713	24.6581

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	7.35759	1.33818	5.498	4.87e-08 ***
reading_score	0.84910	0.01893	44.855	< 2e-16 ***

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

residual standard error: 8.736 on 998 degrees of freedom  
Multiple R-squared: 0.6684, Adjusted R-squared: 0.6681  
F-statistic: 2012 on 1 and 998 DF, p-value: < 2.2e-16

```
15 a <- data.frame(reading_score=90)
16 predict(model,a)
17
```

17: [Top level] +

Console Terminal Jobs R Script

R R412 : ~/r
> a <- data.frame(reading\_score=90)
> predict(model,a)
1
83.77661
> |

# Accuracy

```
sqrt(mean((data$reading_score-predicted_value)^2))
```

20.6478715883146

```
model2 <- lm(math_score ~ writing_score, data=data)
print(model2)
```

Call:

```
lm(formula = math_score ~ writing_score, data = data)
```

Coefficients:

(Intercept)	writing_score
11.5831	0.8009

```
: summary(model2)
```

```
Call:  
lm(formula = math_score ~ writing_score, data = data)  
  
Residuals:  
    Min      1Q  Median      3Q     Max  
-24.8467 -6.4600  0.1464  6.4356 25.5515  
  
Coefficients:  
            Estimate Std. Error t value Pr(>|t|)  
(Intercept) 11.58310   1.31369   8.817 <2e-16 ***  
writing_score 0.80092   0.01884  42.511 <2e-16 ***  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 9.049 on 998 degrees of freedom  
Multiple R-squared:  0.6442,    Adjusted R-squared:  0.6439  
F-statistic: 1807 on 1 and 998 DF,  p-value: < 2.2e-16
```

```
b<-data.frame(writing_score=80)  
predicted_value_two=predict(model2,b)  
predicted_value_two
```

```
1: 75.6568060570767
```

```
# Accuracy  
sqrt(mean((data$writing_score-predicted_value_two)^2))
```

```
16.9846914585318
```

# MULTIPLE LINEAR REGRESSION

```
##MULTIPLE LINEAR REGRESSION

#Predicting math scores for reading and writing score
model3 = lm(math.score~reading.score + writing.score, data =df)
print(model3)

Call:
lm(formula = math.score ~ reading.score + writing.score, data = df)

Coefficients:
```

(Intercept)	reading.score	writing.score
7.5241	0.6013	0.2494

```
#Review the results
summary(model3)
```

```
Call:
lm(formula = math.score ~ reading.score + writing.score, data = df)
```

Residuals:

Min	1Q	Median	3Q	Max
-23.8779	-6.1750	0.2693	6.0184	24.8727

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	7.52409	1.32823	5.665	1.93e-08 ***
reading.score	0.60129	0.06304	9.538	< 2e-16 ***
writing.score	0.24942	0.06057	4.118	4.14e-05 ***

---

Signif. codes: 0 '\*\*\*\*' 0.001 '\*\*\*' 0.01 '\*\*' 0.05 '\*' 0.1 ' ' 1

Residual standard error: 8.667 on 997 degrees of freedom  
Multiple R-squared: 0.674, Adjusted R-squared: 0.6733  
F-statistic: 1031 on 2 and 997 DF, p-value: < 2.2e-16

```
b <-data.frame(reading.score=90,writing.score=88)
predicted_value <-predict(model3,b)
```

```
predicted_value
```

```
1: 83.5894767553443
```

```
#Redidual Standard Error
RSE=sigma(model3)/mean(df$math.score)
RSE
```

```
0.131133840006564
```

```
#Accuracy
sqrt(mean((predicted_value)^2))
```

```
83.5894767553443
```

## Conclusion:

- From the given dataset we can able to predict the mark obtained by each student using the linear regression model.
- Here we use both Single and Multiple Linear Regression, and we use Multiple Linear Regression since the given dataset contain more than 2 columns and it has more accuracy compared to Single Linear Regression.
- So we are able to predict the mark of the student using the Multiple Linear regression model.