ASSIGNMENT

TECHNOLOGY PARK MALAYSIA

CT127-3-2-PFDA

PROGRAMMING FOR DATA ANALYSIS

APU2F2111CS(DA)

HAND OUT DATE: 6 DECEMBER 2021

HAND IN DATE: 31 JANUARY 2021

WEIGHTAGE : 50%

INSTRUCTION TO CANDIDATES:

1. Submit your assignment at the administrative counter.
2. Students are advised to underpin their answers with the use of references(cited using American Psychological Association(APA) Referencing).
3. Late submission will be awarded zero (0) unless Extenuating Circumstances(EC) are upheld.
4. Cases of plagiarism will be penalized.
5. The assignment should be bound in an appropriate style (comb bound or stapled).
6. Where the assignment should be submitted in both hardcopy and softcopy, the softcopy of the written assignment and source code (where appropriate) should be on a CD in an envelope / CD cover and attached to the hardcopy.
7. You must obtain 50% overall to pass this module.

Table of Contents

1.0 Introduction

For this assignment, I am assigned to analyse the identify hidden problem provide meaningful insight for decision making. The dataset provided for this assignment is related to the candidate's personal details. It contains 25 columns and 17007 rows. The dataset includes the personal detail of the candidates, school type, work status, work salary and so on.

The software application used in the project is R studio. The dataset was in .csv format and has been imported for analysis. The data provided has been thoroughly manipulated and analysed. Additionally, graphs are plotted for better visualisation of the data.

2.0 Data Exploration

2.1 Data Import

```
# Data Import
assign_data = read.csv("C:\\Users\\ShiHan\\Desktop\\APU\\Degree\\Level 2 Sem 1\\PFDA\\Assignment\\Placement_Data_Full_Class.csv",header=TRUE)
library(ggplot2)
library(dplyr)
```

Figure 1 Data Import

The code in Figure 1 is used to import csv file into RStudio.

2.2 Summary of Dataset

```
#summary of dataset
summary(assign_data)
```

Figure 2 Summary of Dataset

After importing dataset, the code in Figure 2 is used to summarise all the data in dataset and allow me to study details of dataset.

```
     sl_no          gender               age          address
 Min.   :    1   Length:17007       Min.   :18.00   Length:17007
 1st Qu.: 4252   Class :character   1st Qu.:19.00   Class :character
 Median : 8504   Mode  :character   Median :20.00   Mode  :character
 Mean   : 8504                      Mean   :20.49
 3rd Qu.:12756                      3rd Qu.:22.00
 Max.   :17007                      Max.   :23.00
              Medu                       Fedu           Mjob              Fjob
 None            :   7   None            :   4   Length:17007       Length:17007
 Primary         :4170   Primary         :4313   Class :character   Class :character
 5th to 9th grade:4255   5th to 9th grade:4255   Mode  :character   Mode  :character
 Secondary       :4246   Secondary       :4238
 Higher          :4329   Higher          :4197

    famsup          extra_class         extra_act           internet
 Length:17007       Length:17007       Length:17007       Length:17007
 Class :character   Class :character   Class :character   Class :character
 Mode  :character   Mode  :character   Mode  :character   Mode  :character
```

Figure 3 Output of summarise dataset(1)

```
     ssc_p           ssc_b               hsc_p           hsc_b              hsc_s
 Min.   :40.89   Length:17007       Min.   :37.00   Length:17007       Length:17007
 1st Qu.:61.00   Class :character   1st Qu.:61.00   Class :character   Class :character
 Median :72.00   Mode  :character   Median :72.00   Mode  :character   Mode  :character
 Mean   :72.44                      Mean   :72.45
 3rd Qu.:84.00                      3rd Qu.:84.00
 Max.   :95.00                      Max.   :97.70
    degree_p       degree_t           workex            etest_p      specialisation
 Min.   :50.00   Length:17007       Length:17007       Min.   :50.00   Length:17007
 1st Qu.:61.00   Class :character   Class :character   1st Qu.:61.00   Class :character
 Median :72.00   Mode  :character   Mode  :character   Median :72.00   Mode  :character
 Mean   :72.39                                         Mean   :72.32
 3rd Qu.:84.00                                         3rd Qu.:84.00
 Max.   :95.00                                         Max.   :98.00
     mba_p           status             salary       etest_group mba_group
 Min.   :50.00   Length:17007       Min.   :     0   A:8598      A:8535
 1st Qu.:61.00   Class :character   1st Qu.:     0   B:8409      B:8472
 Median :72.00   Mode  :character   Median :200000
 Mean   :72.54                      Mean   :158593
 3rd Qu.:84.00                      3rd Qu.:300000
 Max.   :95.00                      Max.   :500000
```

Figure 4 Output of summarise dataset(2)

Figure 3 and 4 shows the output of the code that I used to look for the summary of the dataset. It shows the summary of each column in dataset with minimum value, 1st quartile, median, mean, 3rd quartile, and maximum value.

2.3 Column Name in dataset (Attribute names)

```
#column name of dataset
colnames(assign_data)
```

Figure 5 Columns Name in dataset

Figure 5 above shows the code that I have used to check the columns name in the dataset.

```
> colnames(assign_data)
 [1] "sl_no"          "gender"         "age"            "address"        "Medu"
 [6] "Fedu"           "Mjob"           "Fjob"           "famsup"         "paid"
[11] "activities"     "internet"       "ssc_p"          "ssc_b"          "hsc_p"
[16] "hsc_b"          "hsc_s"          "degree_p"       "degree_t"       "workex"
[21] "etest_p"        "specialisation" "mba_p"          "status"         "salary"
```

Figure 6 Output of Columns Name in dataset

Figure 6 above shows the output of columns name after running the code in Figure 5.

3.0 Data Pre-processing

3.1 Change column names

```
# Rename columns
names(assign_data)[names(assign_data)=="paid"] <- "extra_class"
names(assign_data)[names(assign_data)=="activities"] <- "extra_act"
```

Figure 7 Change column names

Figure 7 above shows the code that has been used to change columns name.

```
> colnames(assign_data)
 [1] "sl_no"          "gender"         "age"            "address"        "Medu"
 [6] "Fedu"           "Mjob"           "Fjob"           "famsup"         "extra_class"
[11] "extra_act"      "internet"       "ssc_p"          "ssc_b"          "hsc_p"
[16] "hsc_b"          "hsc_s"          "degree_p"       "degree_t"       "workex"
[21] "etest_p"        "specialisation" "mba_p"          "status"         "salary"
>
```

Figure 8 Output of updated columns name

Figure 8 above shows the output of updated columns name. It can be compared with Figure 6 to see the difference.

3.2 Add new columns

```
# Add new column
assign_data$etest_group <- as.factor(ifelse(assign_data$etest_p <=72,'A','B'))
assign_data$mba_group <- as.factor(ifelse(assign_data$mba_p <=72,'A','B'))
```

Figure 9 Add new columns

Figure 9 above shows the code that has been used to add new columns.

ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t	workex	etest_p	specialisation	mba_p	status	salary	etest_group	mba_group
State	91.00	State	Commerce	58.00	Sci&Tech	No	55.00	Mkt&HR	78	Placed	350000	A	B
State	78.33	Central	Science	77.48	Sci&Tech	Yes	86.50	Mkt&Fin	80	Placed	200000	B	B
Private	68.00	Private	Arts	64.00	Comm&Mgmt	No	75.00	Mkt&Fin	77	Placed	350000	B	B
Central	52.00	State	Science	52.00	Sci&Tech	No	66.00	Mkt&HR	50	Not Placed	0	A	A
Private	73.60	Central	Commerce	73.30	Comm&Mgmt	No	96.80	Mkt&Fin	86	Placed	250000	B	B
Private	49.80	State	Science	67.25	Sci&Tech	Yes	55.00	Mkt&Fin	63	Not Placed	0	A	A
Central	49.20	State	Commerce	79.00	Comm&Mgmt	No	74.28	Mkt&Fin	59	Not Placed	0	B	A
State	64.00	State	Science	66.00	Sci&Tech	Yes	67.00	Mkt&Fin	83	Placed	300000	A	B
State	79.00	Central	Commerce	72.00	Comm&Mgmt	No	91.34	Mkt&Fin	51	Placed	350000	B	A
Private	70.00	State	Commerce	61.00	Comm&Mgmt	No	54.00	Mkt&Fin	67	Not Placed	0	A	A
State	61.00	Central	Commerce	60.00	Comm&Mgmt	Yes	62.00	Mkt&HR	53	Placed	300000	A	A
State	68.40	Central	Commerce	78.30	Comm&Mgmt	Yes	60.00	Mkt&Fin	59	Placed	400000	A	A
Private	55.00	Central	Science	65.00	Comm&Mgmt	No	62.00	Mkt&HR	92	Not Placed	0	A	B
State	87.00	Central	Commerce	59.00	Comm&Mgmt	No	68.00	Mkt&Fin	67	Placed	400000	A	A
State	47.00	Central	Commerce	50.00	Comm&Mgmt	No	76.00	Mkt&HR	52	Not Placed	0	B	A
Private	75.00	Central	Commerce	69.00	Comm&Mgmt	Yes	72.00	Mkt&Fin	88	Placed	250000	A	B
Private	66.20	Central	Commerce	65.60	Comm&Mgmt	Yes	60.00	Mkt&Fin	54	Placed	250000	A	A

Figure 10 Output after adding new columns

Figure 10 above shows the result after the new columns are added. The new columns are etest_group and mba_group.

3.3 Check for null values

```
> apply(assign_data, 2, function(x)
+   any(is.na(x)))
      sl_no      gender         age     address        Medu        Fedu
      FALSE       FALSE       FALSE       FALSE       FALSE       FALSE
       Mjob        Fjob     famsup  extra_class  extra_act    internet
      FALSE       FALSE       FALSE       FALSE       FALSE       FALSE
      ssc_p       ssc_b       hsc_p       hsc_b       hsc_s    degree_p
      FALSE       FALSE       FALSE       FALSE       FALSE       FALSE
   degree_t      workex    etest_p specialisation      mba_p      status
      FALSE       FALSE       FALSE       FALSE       FALSE       FALSE
     salary
       TRUE
>
```

Figure 11 Check for null values

Figure 11 above shows the code that has been used to check for null values (RDocumentation, 2021).

As we can see in figure 9, only 'salary' column shows TRUE while other columns show FALSE which means there are only null values occur in 'salary' column.

```
# Replace missing value to 0
assign_data[is.na(assign_data)] = 0
```

Figure 12 Replace null values

Figure 12 above shows the code that has been used to replace null values.

sl_no	salary
4	NA
6	NA
7	NA
10	NA
13	NA
15	NA
18	NA
19	NA
26	NA
30	NA
32	NA
35	NA
37	NA
42	NA
43	NA
46	NA
47	NA
50	NA

Figure 13 Before replace

sl_no	salary
4	0
6	0
7	0
10	0
13	0
15	0
18	0
19	0
26	0
30	0
32	0
35	0
37	0
42	0
43	0
46	0
47	0
50	0

Figure 14 After replace

Figure 13 and 14 shows data before and after replacing null values in dataset.

4.0 Data Transfromation

```
# Check datatype
str(assign_data)
```

Figure 15 Check datatypes

Figure 15 above shows the code that has been used to check datatype of every attributes.

```
> str(assign_data)
'data.frame':   17007 obs. of  27 variables:
 $ sl_no         : int  1 2 3 4 5 6 7 8 9 10 ...
 $ gender        : chr  "M" "M" "M" "M" ...
 $ age           : int  23 19 19 21 22 19 19 18 19 21 ...
 $ address       : Factor w/ 2 levels "Rural","Urban": 2 2 2 2 2 2 2 2 2 2 ...
 $ Medu          : int  4 1 1 4 3 4 2 4 3 3 ...
 $ Fedu          : int  4 1 1 2 3 3 2 4 2 4 ...
 $ Mjob          : chr  "at_home" "at_home" "at_home" "health" ...
 $ Fjob          : chr  "teacher" "other" "other" "services" ...
 $ famsup        : chr  "no" "yes" "no" "yes" ...
 $ paid          : chr  "no" "no" "yes" "yes" ...
 $ activities    : chr  "no" "no" "no" "yes" ...
 $ internet      : chr  "no" "yes" "yes" "yes" ...
 $ ssc_p         : num  67 79.3 65 56 85.8 ...
 $ ssc_b         : chr  "State" "State" "Private" "Central" ...
 $ hsc_p         : num  91 78.3 68 52 73.6 ...
 $ hsc_b         : chr  "State" "Central" "Private" "State" ...
 $ hsc_s         : chr  "Commerce" "Science" "Arts" "Science" ...
 $ degree_p      : num  58 77.5 64 52 73.3 ...
 $ degree_t      : chr  "Sci&Tech" "Sci&Tech" "Comm&Mgmt" "Sci&Tech" ...
 $ workex        : chr  "No" "Yes" "No" "No" ...
 $ etest_p       : num  55 86.5 75 66 96.8 ...
 $ specialisation: chr  "Mkt&HR" "Mkt&Fin" "Mkt&Fin" "Mkt&HR" ...
 $ mba_p         : int  78 80 77 50 86 63 59 83 51 67 ...
 $ status        : chr  "Placed" "Placed" "Placed" "Not Placed" ...
 $ salary        : num  350000 200000 350000 0 250000 0 0 300000 350000 0 ...
 $ etest_group   : Factor w/ 2 levels "A","B": 1 2 2 1 2 1 2 1 2 1 ...
 $ mba_group     : Factor w/ 2 levels "A","B": 2 2 2 1 2 1 1 2 1 1 ...
```

Figure 16 Result of datatype of every attribute

Figure 16 shows the result of every attributes' datatype.

4.1 Filtering

```
# Filter out data rows which contains salary = 0 for more accurate analysis
acc_salary <- filter(assign_data, salary!=0)
```

Figure 17 Filtering

Figure 17 above shows the code that has been used to do filtering. Salary equals to 0 are filtered for more accurate analysis.

4.2 Change datatype

```
# Change address datatype to factor and rename the levels
assign_data <- assign_data %>%
  mutate(address=as.factor(address))

levels(assign_data$address) <- c("Rural","Urban")
```

Figure 18 Change datatype and rename

Figure 18 above shows the code that has been used to change datatype and do renaming.

```
# Change education level of Medu and Fedu
assign_data = assign_data %>%
  mutate(
    Medu = factor(Medu, labels = c("None", "Primary", "5th to 9th grade", "Secondary", "Higher")),
    Fedu = factor(Fedu, labels = c("None", "Primary", "5th to 9th grade", "Secondary", "Higher"))
  )
```

Figure 19 Change education level of Medu and Fedu

Figure 19 above shows the code that has been used to change datatype of Medu and Fedu and renaming the levels.

Figure 20,21,22 and 23 below shows the result after running the code in Figure 18 and 19.

	address	count
1	R	8213
2	U	8794

Figure 20 Before change address level

	address	count
1	Rural	8213
2	Urban	8794

Figure 21 After change address level

	Medu	Fedu
1	0	1
2	0	2
3	1	0
4	1	1
5	1	2
6	1	3
7	1	4
8	2	1
9	2	2
10	2	3
11	2	4
12	3	1
13	3	2
14	3	3
15	3	4
16	4	0
17	4	1
18	4	2
19	4	3
20	4	4

Figure 23 Before change Medu and Fedu level

	Medu	Fedu
1	None	Primary
2	None	5th to 9th grade
3	Primary	None
4	Primary	Primary
5	Primary	5th to 9th grade
6	Primary	Secondary
7	Primary	Higher
8	5th to 9th grade	Primary
9	5th to 9th grade	5th to 9th grade
10	5th to 9th grade	Secondary
11	5th to 9th grade	Higher
12	Secondary	Primary
13	Secondary	5th to 9th grade
14	Secondary	Secondary
15	Secondary	Higher
16	Higher	None
17	Higher	Primary
18	Higher	5th to 9th grade
19	Higher	Secondary
20	Higher	Higher

Figure 24 After change Medu and Fedu level

5.0 Data Manipulation & Data Visualisation

5.1 Question 1: What factors affect salary?

5.1.1 Analysis 1: Find the relationship between working experience and salary

```
# Analysis (1-1) : Find the relationship between working experience and salary
wx_salary <- acc_salary%>%
  group_by(workex)%>%
  summarise(salary=as.numeric(format(round(mean(salary)),0)))

ggplot(wx_salary,aes(x=workex,y=salary,fill=workex)) +
  geom_bar(stat="identity",width=0.5) +
  scale_fill_manual(values = c("pink","steelblue")) +
  geom_text(aes(label=salary),position=position_dodge(0.9),vjust=-0.25) +
  labs(title = "Analysis between Working Experience and Salary",
       x = "Working Experience",
       y = "Salary",
       fill = "Working Experience") +
  theme_bw()
```

Figure 25 Code snippet Q1 Analysis 1

Figure 25 above shows the code for Question 1 Analysis 1. First of all, a new table wx_salary is created grouped by Working Experience to examine the average salary from acc_salary. The average of salary is obtained and also reformatted to whole number without decimals for better looking analysis and label.

The ggplot() function is used to fill the data for the x-axis and y-axis which is Working Experience(workex) and Salary(salary). Then the geom_bar function is used to plot a bar chart. Next, the geom_text is used to add the label for Salary and adjust the position of label. The labs() function is used to determine the label name of the x-axis, y-axis, legend title and graph title. Lastly, theme is also used to make the background of table. The Data Visualisation result shows at Figure 27 below (tidyverse, ggplot2(geom_bar), 2021).

As mentioned in **4.1 Filtering,** acc_salary is the new dataset that filtered out rows that contained 'salary' == 0 for more accurate analysis. This acc_salary will be used in most of the Question 1 Analysis.

	workex	salary
1	No	308425
2	Yes	308639

Figure 26 Data Exploration for Average Salary grouped by Working Experience

Figure 27 Data Visualisation Q1 Analysis 1

Figure 27 is the analysis between Working Experience and their Average Salary.

We can see that with or without working experience doesn't affect much on how much will the company pay to candidate. There is only slightly different average of salary between who has or has no working experience.

One of the possible reason could be most of the companies nowadays would like to give chance to fresh graduates to prove their abilities and skills learned during college or university. Therefore, they are willing to give the same amount of salary to those who doesn't have working experience

5.1.2 Analysis 2: Find the relationship between gender and salary

```
# Analysis (1-2) : Find the relationship between gender and salary
gen_salary <- acc_salary%>%
  group_by(gender)%>%
  summarise(salary=as.numeric(format(round(mean(salary)),0)))

ggplot(gen_salary,aes(x=gender,y=salary,fill=gender)) +
  geom_bar(stat="identity",width=0.5) +
  ggtitle("Analysis between gender and salary") +
  scale_fill_manual("Legend",values = c("pink","steelblue")) +
  geom_text(aes(label=salary),position=position_dodge(0.9),vjust=-0.25) +
  labs(title = "Analysis between Gender and Salary",
       x = "Gender",
       y = "Salary",
       fill = "Gender") +
  theme_bw()
```

Figure 28 Code snippet for Q1 Analysis 2

Figure 28 above shows the code for Question 1 Analysis 2. First of all, a new table gen_salary is created grouped by Gender to examine the average salary from acc_salary. The average of salary is obtained and also reformatted to whole number without decimals for better looking analysis and label.

The ggplot() function is used to fill the data for the x-axis and y-axis which is Gender(gender) and Salary(salary). Then the geom_bar function is used to plot a bar chart. Next, the geom_text is used to add the label for Salary and adjust the position of label. The labs() function is used to determine the label name of the x-axis, y-axis, legend title and graph title. Lastly, theme is also used to make the background of table. The Data Visualisation result shows at Figure 30 below (tidyverse, ggplot2(geom_bar), 2021).

As mentioned in **4.1 Filtering,** acc_salary is the new dataset that filtered out rows that contained 'salary' == 0 for more accurate analysis. This acc_salary will be used in most of the Question 1 Analysis.

	gender	salary
1	F	309729
2	M	307336

Figure 29 Data Exploration for Average Salary grouped by Gender

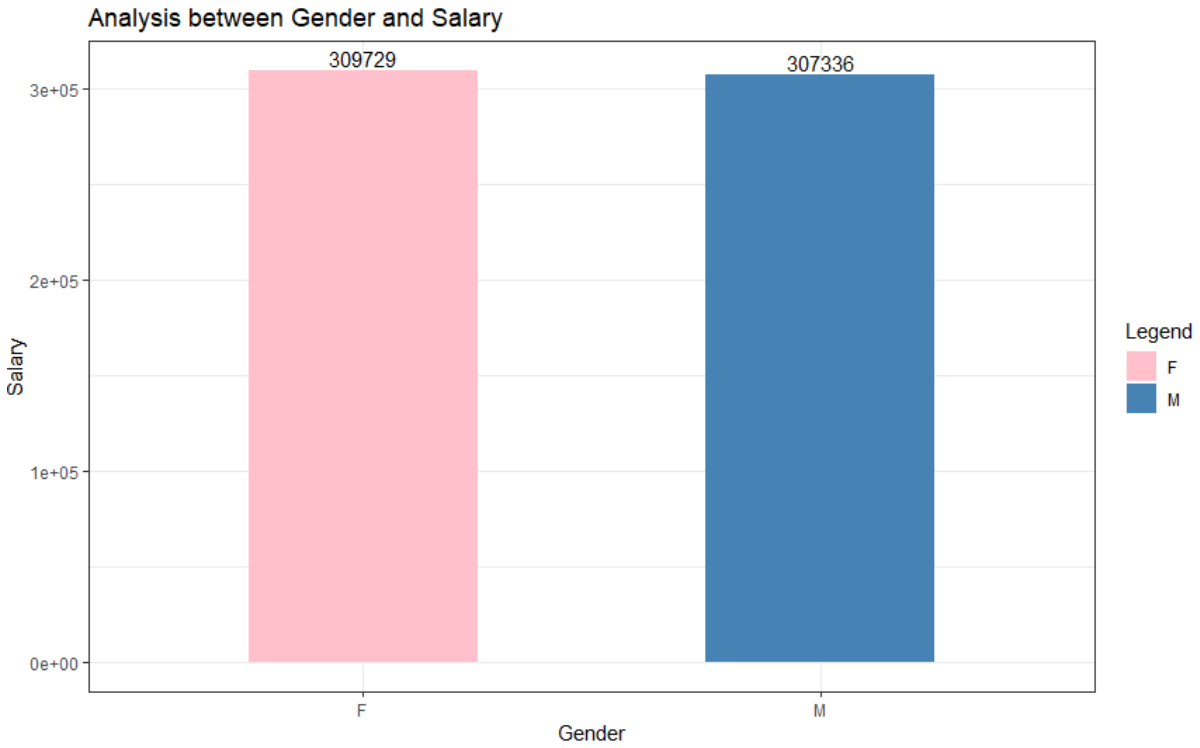


Figure 30 Data visualisation Q1 Analysis 2

Figure 30 is the analysis between Gender and their Average Salary.

We can see that gender doesn't affect much on how much will the company pay to candidate. There is approximately $2500 different average of salary between gender.

One of the possible reason could be most of the companies nowadays has abandoned the mindset of Male could do better work than Female. Nowadyas, Female can do the same work as Male while Female can be more careful about the work as they could comeplete the tasks with better details and neatly. Therefore, they are willing to give slightly more amount of salary to female compared to male.

5.1.3 Analysis 3: Find the relationship between age and salary

```
# Analysis (1-3) : Find the relationship between age and salary
age_salary <- acc_salary%>%
  group_by(age)%>%
  summarise(salary=as.numeric(format(round(mean(salary)),0)))

ggplot(age_salary,aes(x=age,y=salary,fill=age)) +
  geom_bar(stat="identity",width=0.5) +
  geom_text(aes(label=salary),position=position_dodge(0.9),vjust=-0.25) +
  labs(title = "Analysis between Age and Salary",
       x = "Age",
       y = "Salary",
       fill = "Age") +
  theme_bw()
```

Figure 31 Code snippet for Q1 Analysis 3

Figure 31 above shows the code for Question 1 Analysis 3. First of all, a new table age_salary is created grouped by Age to examine the average salary from acc_salary. The average of salary is obtained and also reformatted to whole number without decimals for better looking analysis and label.

The ggplot() function is used to fill the data for the x-axis and y-axis which is Age(age) and Salary(salary). Then the geom_bar function is used to plot a bar chart. Next, the geom_text is used to add the label for Salary and adjust the position of label. The labs() function is used to determine the label name of the x-axis, y-axis, legend title and graph title. Lastly, theme is also used to make the background of table. The Data Visualisation result shows at Figure 33 below (tidyverse, ggplot2(geom_bar), 2021).

As mentioned in **4.1 Filtering,** acc_salary is the new dataset that filtered out rows that contained 'salary' == 0 for more accurate analysis. This acc_salary will be used in most of the Question 1 Analysis.

	age	salary
1	18	307896
2	19	310489
3	20	307665
4	21	308953
5	22	307574
6	23	308599

Figure 32 Data Exploration for Average Salary grouped by Age

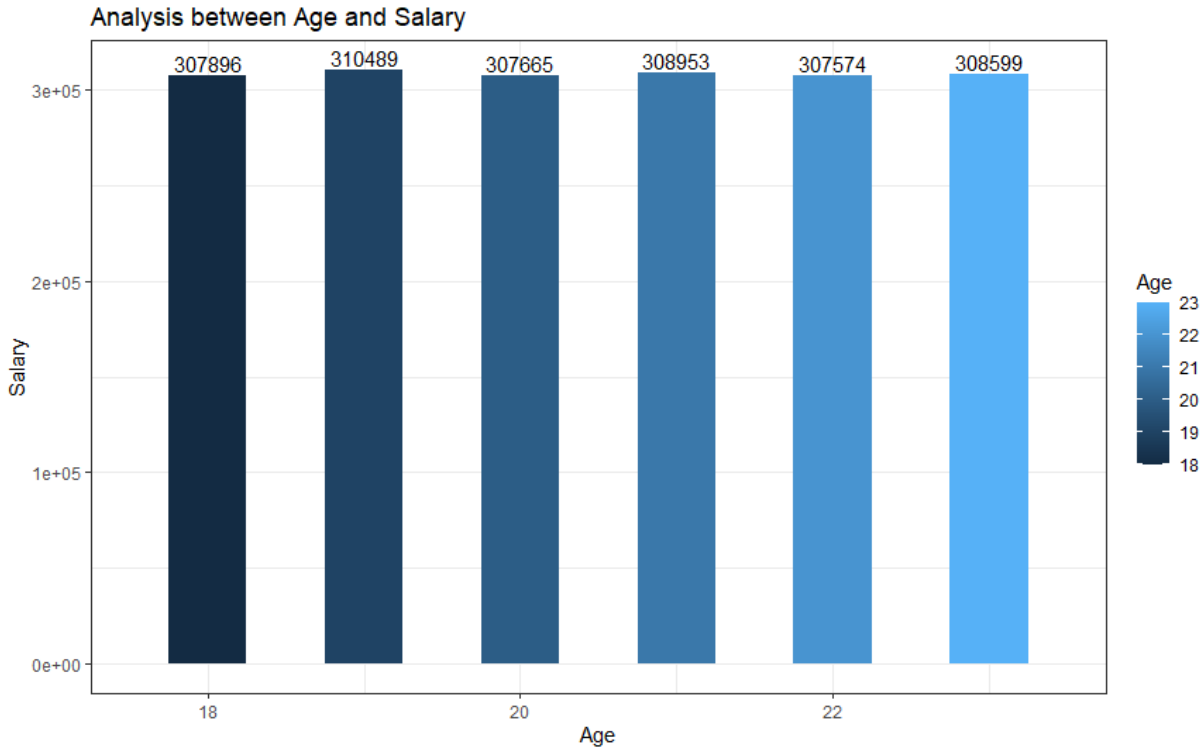


Figure 33 Data visualisation for Q1 Analysis 3

Figure 33 is the analysis between Age and their Average Salary. Age 18-23 candidates are chosen for this dataset and analysis.

We can see that age doesn't affect much on how much will the company pay to candidate. There is only slight different average of salary between different ages.

One of the possible reason could be most of the companies nowadays do not care about whether elder candidates have working experience or not. They probably emphasise on how one could perform better job role for them. Therefore, they are willing to give slightly more amount of salary to one who could do better in the job.

5.1.4 Analysis 4: Find the relationship between employability test and salary

```
assign_data$etest_group <- as.factor(ifelse(assign_data$etest_p <=72,'A','B'))
```

Figure 34 new columns(etest_group)

```
# Analysis (1-4) : Find the relationship between employability test and salary
etest_salary <- acc_salary%>%
  group_by(etest_group)%>%
  summarise(salary=as.numeric(format(round(mean(salary)),0)))

ggplot(etest_salary,aes(x=etest_group,y=salary,fill=etest_group)) +
  geom_bar(stat="identity",width=0.5) +
  geom_text(aes(label=salary),position=position_dodge(0.9),vjust=-0.25) +
  labs(title = "Analysis between Employability test and Salary",
       x = "Employability Test Group",
       y = "Salary",
       fill = "Employability Test Group") +
  theme_bw()
```

Figure 35 Code snippet for Q1 Analysis 4

Figure 34 & 35 above shows the code for Question 1 Analysis 4. First of all, all data in employability test percentage(etest_p) is rearranged and regrouped in "A" and "B". Then, a new table etest_salary is created grouped by Employability Test to examine the average salary from acc_salary. The average of salary is obtained and also reformatted to whole number without decimals for better looking analysis and label.

The ggplot() function is used to fill the data for the x-axis and y-axis which is Employability Test Group(etest_group) and Salary(salary). Then the geom_bar function is used to plot a bar chart. Next, the geom_text is used to add the label for Salary and adjust the position of label. The labs() function is used to determine the label name of the x-axis, y-axis, legend title and graph title. Lastly, theme is also used to make the background of table (tidyverse, ggplot2(geom_bar), 2021). The Data Visualisation result shows at Figure 37 below.

As mentioned in **4.1 Filtering,** acc_salary is the new dataset that filtered out rows that contained 'salary' == 0 for more accurate analysis. This acc_salary will be used in most of the Question 1 Analysis.

	etest_group	salary
1	A	307503
2	B	309587

Figure 36 Data exploration for Average of salary

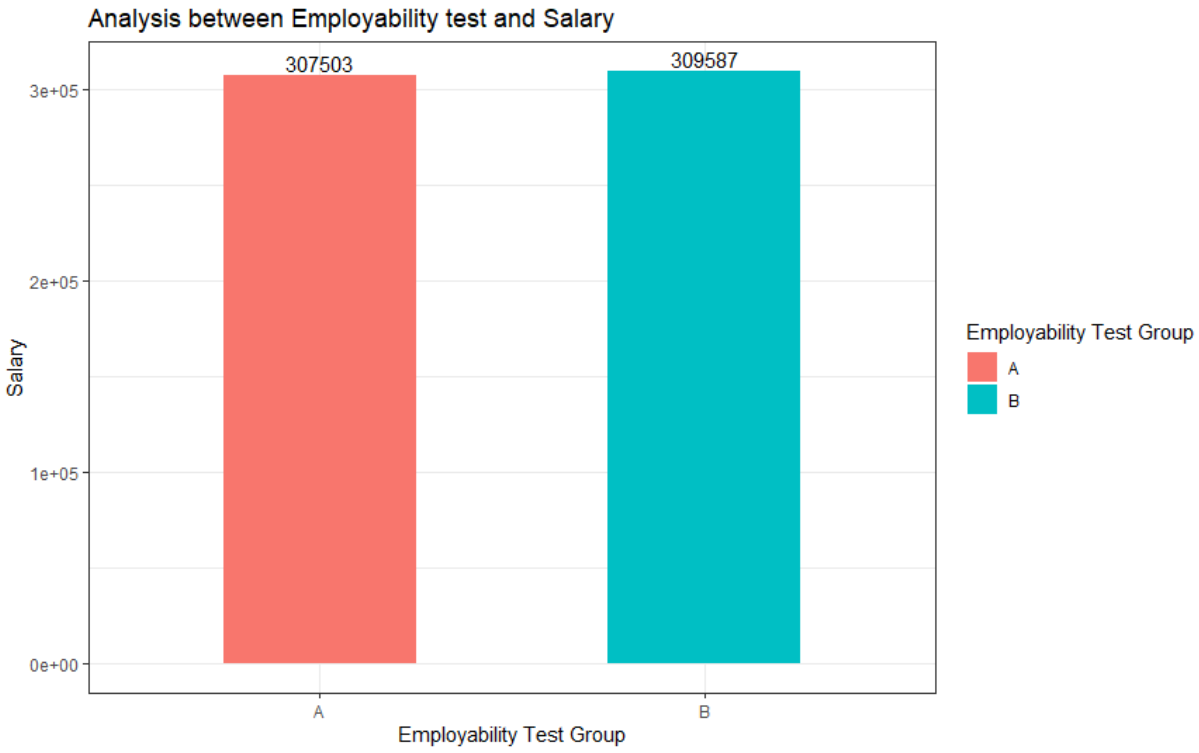


Figure 37 Data visualisation for Q1 Analysis 4

Figure 37 is the analysis between Employability test and their Average Salary. Candidates who have scored more than 72 are assigned to Group B while others are assigned to Group A.

We can see that Employability test does affect on how much will the company pay to candidate. There is only slight more salary are given to candidates in Group B.

One of the possible reason could be this employability test is reliable and very accurate for most of the companies. They could refer to the test to see one's personalities in workplace and believe they could perform better job role for them. Therefore, they are willing to give slightly more amount of salary to candidates who have high score in Employability Test.

5.1.5 Analysis 5: Find the relationship between MBA Percentage and salary

```
assign_data$mba_group <- as.factor(ifelse(assign_data$mba_p <=72,'A','B'))
```

Figure 38 Add new column(mba_group)

```
# Analysis (1-5) : Find the relationship between MBA Percentage and salary
mba_salary <- acc_salary%>%
  group_by(mba_group)%>%
  summarise(salary=as.numeric(format(round(mean(salary)),0)))

ggplot(mba_salary,aes(x=mba_group,y=salary,fill=mba_group)) +
  geom_bar(stat="identity",width=0.5) +
  geom_text(aes(label=salary),position=position_dodge(0.9),vjust=-0.25) +
  labs(title = "Analysis between MBA Percentage and Salary",
       x = "MBA Group",
       y = "Salary",
       fill = "MBA Group") +
  theme_bw()
```

Figure 39 Code snippet for Q1 Analysis 5

Figure 38 & 39 above shows the code for Question 1 Analysis 5. First of all, all data in mba percentage(mba_p) is rearranged and regrouped in “A” and “B”. Then, a new table mba_salary is created grouped by MBA Group to examine the average salary from acc_salary. The average of salary is obtained and also reformatted to whole number without decimals for better looking analysis and label.

The ggplot() function is used to fill the data for the x-axis and y-axis which is MBA Group(mba_group) and Salary(salary). Then the geom_bar function is used to plot a bar chart. Next, the geom_text is used to add the label for Salary and adjust the position of label. The labs() function is used to determine the label name of the x-axis, y-axis, legend title and graph title. Lastly, theme is also used to make the background of table. The Data Visualisation result shows at Figure 41 below (tidyverse, ggplot2(geom_bar), 2021).

As mentioned in **4.1 Filtering,** acc_salary is the new dataset that filtered out rows that contained ‘salary’ == 0 for more accurate analysis. This acc_salary will be used in most of the Question 1 Analysis.

	mba_group	salary
1	A	308791
2	B	308267

Figure 40 Data exploration for Average of salary grouped by MBA Group

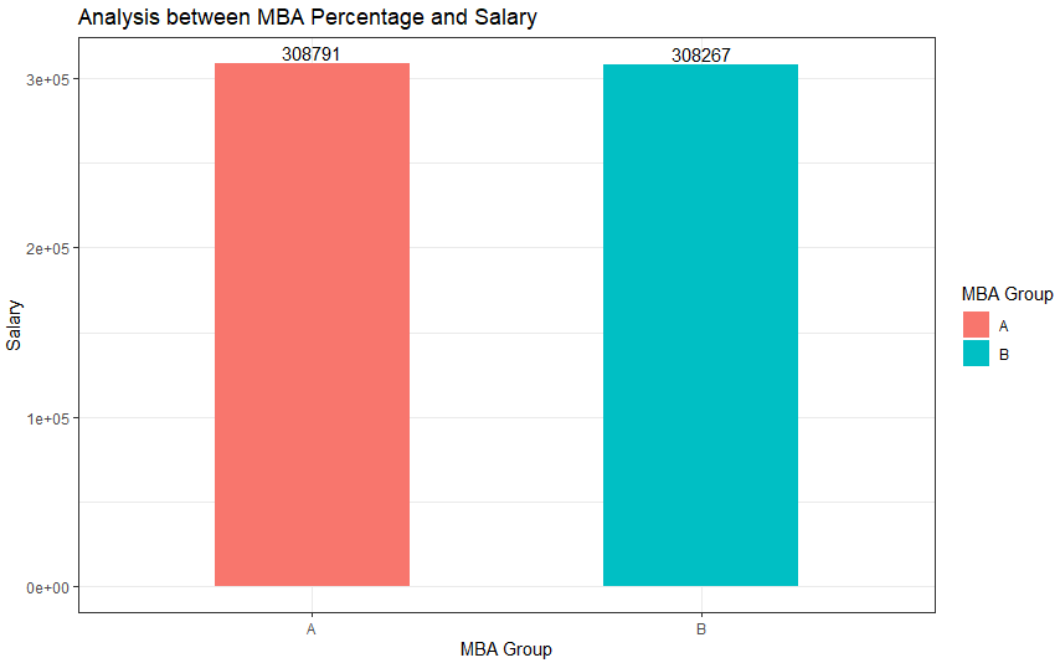


Figure 41 Data visualisation for Q1 Analysis 5

Figure 41 is the analysis between MBA percentage and their Average Salary. Candidates who have scored more than 72 are assigned to Group B while others are assigned to Group A.

We can see that MBA percentage doesn't affect on how much will the company pay to candidate. There is only slight more salary are given to candidates in Group A which is an unexpected outcome.

One of the possible reason could be this employability test is reliable and very accurate for most of the companies. They could refer to the test to see one's personalities in workplace and believe they could perform better job role for them. Therefore, they are willing to give slightly more amount of salary to candidates who have high score in Employability Test.

5.2 Question 2: How does living place affect education level and degree percentage

5.2.1 Analysis 1: Analysis on candidate's living place

```
# Analysis (2-1) : Analysis of student staying in Urban or Rural

add <- assign_data%>%
  group_by(address)%>%
  summarise(count=n())

ggplot(add, aes(x="", y=count, fill=address)) +
  geom_bar(stat="identity", width=1, color="white") +
  coord_polar("y", start=0) +
  labs(title = "Analysis on Candidate's living place") +
  theme_void()
```

Figure 42 Code snippet for Q2 Analysis 1

Figure 42 above shows the code for Question 2 Analysis 1. First of all, a new table add is created grouped by address to examine the count of student in different living place.

The ggplot() function is used to fill the data for the y-axis and fill which is Count(count) and address. Then the geom_bar function is used to plot a bar chart. Next, the coord_polar() is used to convert bar chart to pie chart. The labs() function is used to determine the graph title. Lastly, theme is also used to make the background of table (tidyverse, ggplot2(geom_bar), 2021). The Data Visualisation result shows at Figure 44 below.

	address	count
1	Rural	8213
2	Urban	8794

Figure 43 Data exploration for count of student grouped by address

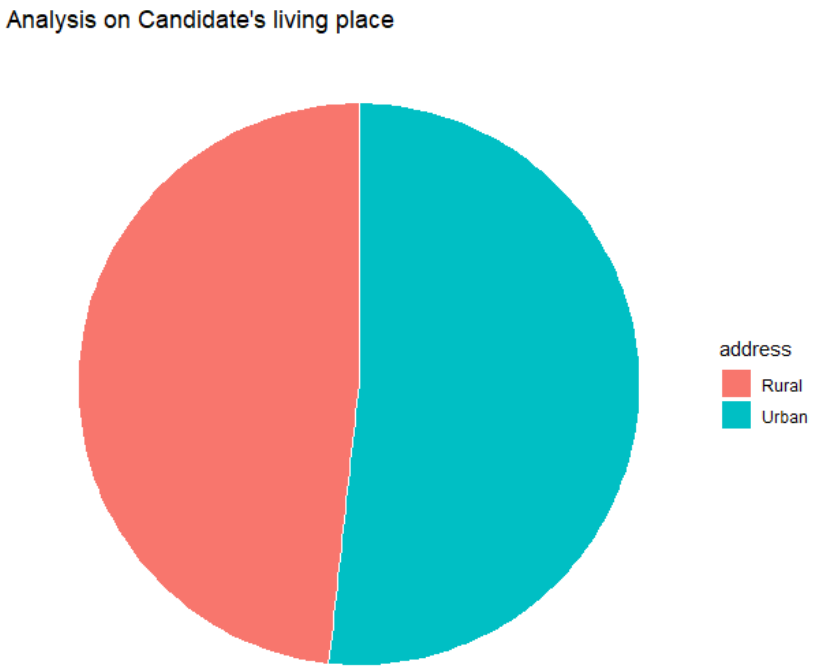


Figure 44 Data visualisation for Q2 Analysis 1

From Figure 43 & 44, we can see that approximately 52% of candidates live in urban while other 48% of candidates live in rural. With this analysis, we may proceed to next analysis to see how living place of candidates will affect their education level.

5.2.2 Analysis 2: Analysis of student type of school

```
# Analysis (2-2) : Analysis of candidate's type of school
school_type <- mutate(school_type%>%
                        group_by(ssc_b,hsc_b)%>%
                        summarise(count=n()))

ggplot(school_type,aes(x=ssc_b,y=count,fill=hsc_b)) +
  geom_bar(stat="identity",position="dodge") +
  geom_text(aes(label=count),position=position_dodge(0.9),vjust=-0.25) +
  labs(title = "Where students study for secondary education and higher education?",
       x = "Secondary School",
       y = "Salary",
       fill = "Higher Secondary School") +
  theme_bw()
```

Figure 45 Code snippet for Q2 Analysis 2

Figure 45 above shows the code for Question 2 Analysis 2. First of all, a new table school_type is created grouped by secondary school and higher secondary school type to examine the count of student in different school.

The ggplot() function is used to fill the data for the x-axis, y-axis and fill which is Count(count) and school type(ssc_b,hsc_b). Then the geom_bar function is used to plot a bar chart. Next, the geom_text is used to add the label for countand adjust the position of label. The labs() function is used to determine the label of x-axis, y-axis legend title, graph title. Lastly, theme is also used to make the background of table. The Data Visualisation result shows at Figure 46 below (tidyverse, ggplot2(geom_bar), 2021).

	ssc_b	hsc_b	count
1	Central	Central	1
2	Central	Private	1
3	Central	State	1
4	Private	Central	1
5	Private	Private	1
6	Private	State	1
7	State	Central	1
8	State	Private	1
9	State	State	1

Figure 46 Data exploration for count of candidates grouped by secondary school and higher secondary school

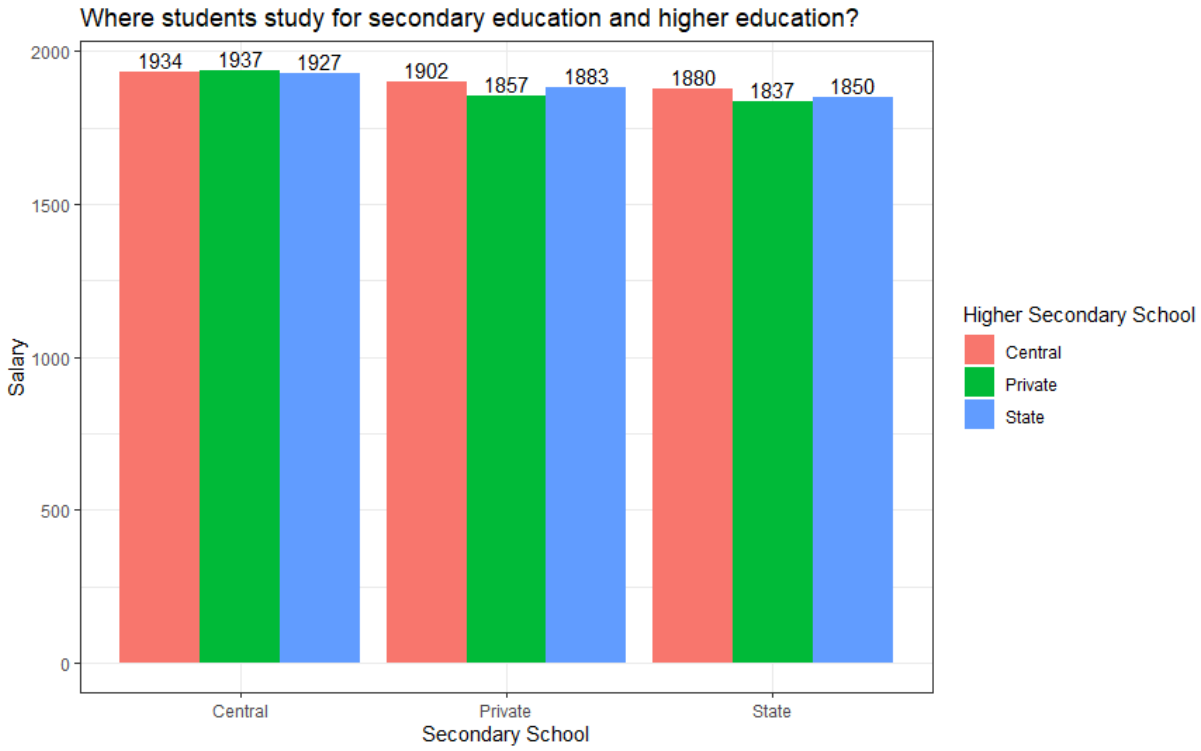


Figure 47 Data visualisation for Q2 Analysis 2

Figure 41 is the analysis between Secondary School Type, Higher Secondary School type and their count of candidates.

We can see only there are more candidates change their school type when they went to higher secondary school from secondary school.

One of the possible reason could be most of the candidates find that they don't like the environment of their secondary school, therefore they changed to another school type .

5.2.3 Analysis 3: Find the relationship between school type and degree percentage

```
# Analysis 2-3: Find the relationship between school type and degree percentage
school_type = select(assign_data, ssc_b, hsc_b, degree_p)
school_type <- mutate(school_type%>%
                      group_by(ssc_b,hsc_b)%>%
                      summarise(degree_p))

ggplot(school_type,aes(x=ssc_b,y=degree_p,fill=hsc_b)) +
  geom_violin(trim=FALSE) +
  geom_boxplot(width=0.1,position=position_dodge(0.9)) +
  labs(title = "Where students study affects their degree course mark",
       x = "Secondary School Type",
       y = "Degree Course Mark",
       fill = "Higher Secondary School Type")+
  theme_bw()
```

Figure 48 Code snippet for Q2 Analysis 3

Figure 48 above shows the code for Question 2 Analysis 2. First of all, a new table school_type is created grouped by secondary school and higher secondary school type to examine the degree percentage.

The ggplot() function is used to fill the data for thex-axis, y-axis and fill which is Count(count) and Secondary school(ssc_b) and Higher secondary school(hsc_b). Geom_violin() is used to plot violin plot to support boxplot. Then the geom_boxplot function is used to plot a boxplot. Next, the geom_text is used to add the label for countand adjust the position of label. The labs() function is used to determine the label of x-axis, y-axis legend title, graph title. Lastly, theme is also used to make the background of table (tidyverse, ggplot2(geom_boxplot), 2021). The Data Visualisation result shows at Figure 50 below.

	ssc_b	hsc_b	degree_p
1	Central	Central	64.00
2	Central	Central	70.00
3	Central	Central	72.23
4	Central	Central	67.50
5	Central	Central	81.00
6	Central	Central	56.20
7	Central	Central	64.00
8	Central	Central	66.00
9	Central	Central	69.00
10	Central	Central	50.80

Figure 49 Data exploration for degree percentage grouped by secondary school type and higher secondary school type

There is more rows for this data exploration but first 10 rows are only shown for example purpose.

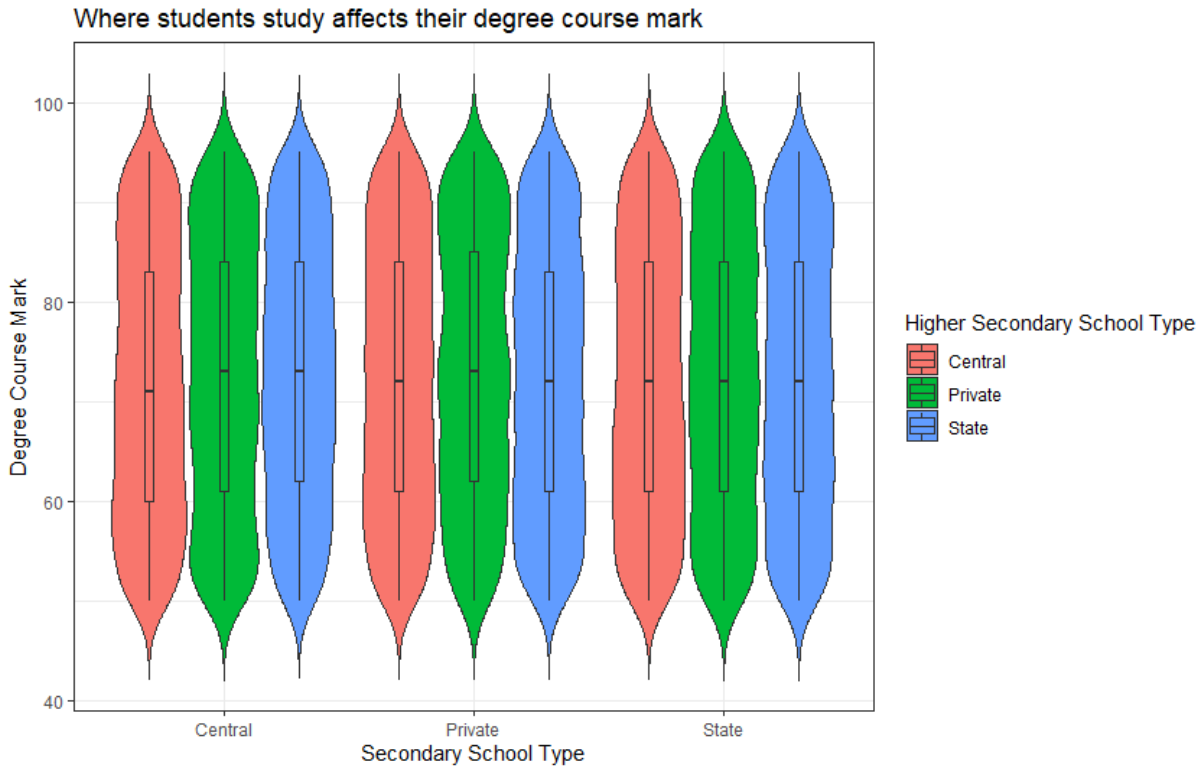


Figure 50 Data visualisation for Q2 Analysis 3

Figure 50 is the analysis between Secondary school type, higher secondary school type and their degree percentage distribution.

We can see that where candidates study doesn't affect on how much will they score for their degree course mark. Distribution of their degree course are actually quite average grouped by whey they study.

One of the possible reason could be knowledge and skills are provided evenly and properly in different type of school. Most elder will find that central school might provide the best but from this analysis we can see that this is completely wrong for now. Even though different type of school has prepared their students well for their degree university life.

5.2.4 Analysis 4: Find the relationship between living place and degree percentage

```
# Analysis (2-4) : Find the relationship between living place and degree percentage
add_degree <- mutate(assign_data%>%
                       group_by(address)%>%
                       summarise(degree_p))

ggplot(add_degree, aes(x=address , y=degree_p , fill=address)) +
  geom_violin(trim=FALSE) +
  geom_boxplot(width=0.1,fill=c("pink","steelblue")) +
  labs(title = "Degree Course Mark grouped by Living Place",
       x = "Living Place",
       y = "Degree Course Mark",
       fill = "Living Place") +
  theme_bw()
```

Figure 51 Code snippet for Q2 Analysis 4

Figure 48 above shows the code for Question 2 Analysis 4. First of all, a new table add_degree is created grouped by address and degree_p to examine the degree percentage.

The ggplot() function is used to fill the data for thex-axis, y-axis and fill which is Living place(address) and Degree Course Mark(degree_p). Geom_violin() is used to plot violin plot to support boxplot. Then the geom_boxplot function is used to plot a boxplot. Next, the geom_text is used to add the label for countand adjust the position of label. The labs() function is used to determine the label of x-axis, y-axis legend title, graph title. Lastly, theme is also used to make the background of table (tidyverse, ggplot2(geom_boxplot), 2021). The Data Visualisation result shows at Figure 52 below.

	address	degree_p
1	Rural	78.86
2	Rural	66.40
3	Rural	65.60
4	Rural	66.00
5	Rural	64.00
6	Rural	72.00
7	Rural	72.70
8	Rural	66.00
9	Rural	78.00
10	Rural	65.00

Figure 52 Data exploration for degree percentage grouped by address

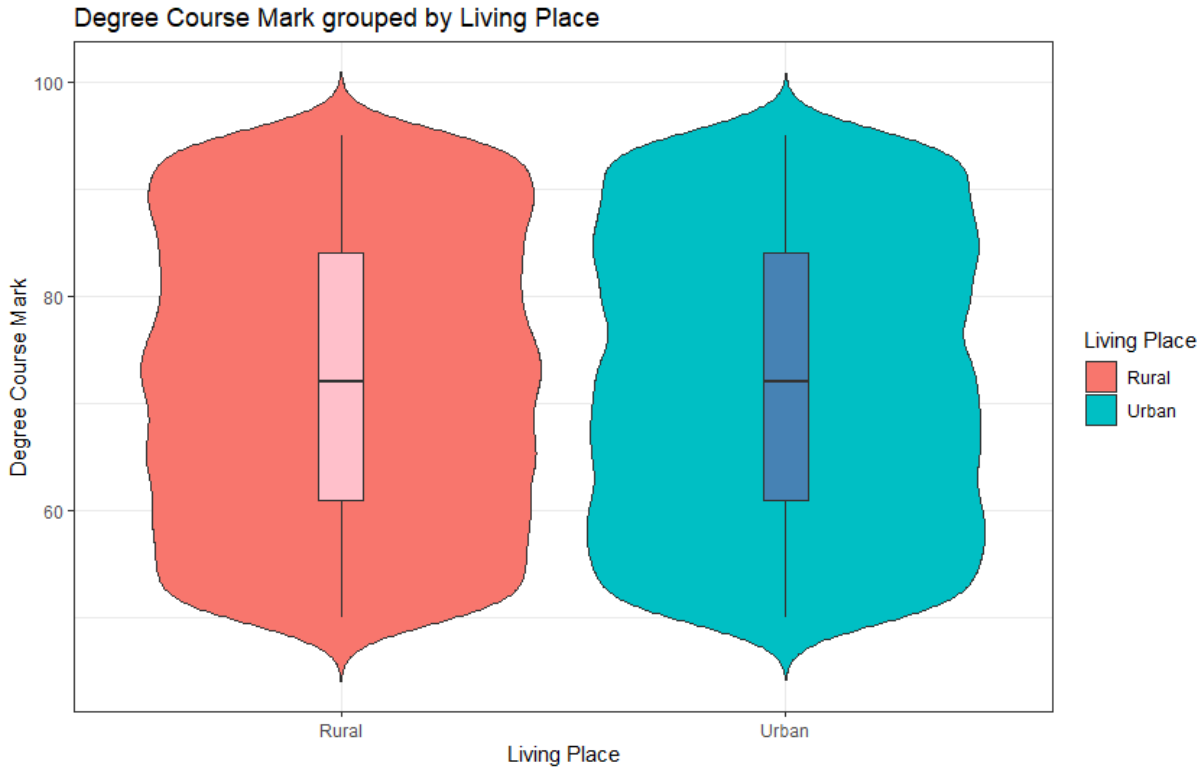


Figure 52 Data visualisation for Q2 Analysis 4

Figure 52 is the analysis between living place and their degree percentage distribution.

We can see that where candidates live doesn't affect on how much will they score for their degree course mark. The boxplot is showing almost same level. But the distribution of their degree course is slight different at course mark 80-90. It is obvious that there are more candidates who lived in urban has scored 80-90 compared to candidates who lived in rural.

One of the possible reason could be candidates who live in Urban has more chance to gain more knowledge by doing outdoor activities and so on.

5.2.5 Analysis 5: Find the relationship between living place and employability test

```
# Analysis 2-5 : Find the relastionship between living place and employability test
add_etest <- assign_data%>%
  group_by(address)%>%
  summarise(etest_p)

ggplot(add_etest,aes(x=address,y=etest_p,fill=address)) +
  geom_violin(trim=FALSE) +
  geom_boxplot(width=0.1,fill=c("pink","steelblue")) +
  labs(title = "Employability Test Percentage grouped by Living Place",
       x = "Living Place",
       y = "Employability Test Percentage",
       fill = "Living Place") +
  theme_bw()
```

Figure 53 Code snippet for Q2 Analysis 5

Figure 48 above shows the code for Question 2 Analysis 5. First of all, a new table add_etest is created grouped by address and etest_p to examine the employability test percentage.

The ggplot() function is used to fill the data for thex-axis, y-axis and fill which is Living place(address) and Employability test percentage(etest_p). Geom_violin() is used to plot violin plot to support boxplot. Then the geom_boxplot function is used to plot a boxplot. Next, the geom_text is used to add the label for countand adjust the position of label. The labs() function is used to determine the label of x-axis, y-axis legend title, graph title. Lastly, theme is also used to make the background of table (tidyverse, ggplot2(geom_boxplot), 2021). The Data Visualisation result shows at Figure 55 below.

	address	etest_p
1	Rural	97.40
2	Rural	50.89
3	Rural	58.00
4	Rural	53.70
5	Rural	93.00
6	Rural	60.00
7	Rural	79.00
8	Rural	70.00
9	Rural	95.50
10	Rural	95.46

Figure 54 Data exploration for employability test grouped by address

Figure 55 Data visualisation for Q2 Analysis 5

Figure 55 above is the analysis between living place and their degree percentage distribution.

We can see that where candidates live doesn't affect on how much will they score for their employability test percentage. The boxplot and the distribution of their employability test percentage is showing almost same level.

5.3 Question 3: What factors affect degree percentage

5.3.1 Analysis 1: Find the relationship between degree type and degree percentage

```
# Analysis (3-1) : Find the relationship between degree type and degree percentage
t_degree <- assign_data%>%
  group_by(degree_t)%>%
  summarise(degree_p)

ggplot(t_degree,aes(x=degree_t,y=degree_p,fill=degree_t)) +
  geom_violin(trim=FALSE) +
  geom_boxplot(width = 0.1) +
  labs(title = "Degree Course Mark grouped by Degree Type",
       x = "Degree Type",
       y = "Degree Course Mark",
       fill = "Degree Type") +
  theme_bw()
```

Figure 56 Code snippet for Q3 Analysis 1

Figure 56 above shows the code for Question 3 Analysis 1. First of all, a new table t_degree is created grouped by degree_t and degree_p to examine the degree percentage.

The ggplot() function is used to fill the data for the x-axis, y-axis and fill which is Degree Type(degree_t) and Degree Percentage(degree_p). Geom_violin () is used to plot violin plot to support boxplot. Then the geom_boxplot() function is used to plot a boxplot. Next, the geom_text is used to add the label for count and adjust the position of label. The labs() function is used to determine the label of x-axis, y-axis legend title, graph title. Lastly, theme is also used to make the background of table. The Data Visualisation result shows at Figure 57 below (tidyverse, ggplot2(geom_boxplot), 2021).

	degree_t	degree_p
1	Comm&Mgmt	64.00
2	Comm&Mgmt	73.30
3	Comm&Mgmt	79.00
4	Comm&Mgmt	72.00
5	Comm&Mgmt	61.00
6	Comm&Mgmt	60.00
7	Comm&Mgmt	78.30
8	Comm&Mgmt	65.00
9	Comm&Mgmt	59.00
10	Comm&Mgmt	50.00

Figure 57 Data exploration for Degree Percentage grouped by Degree type

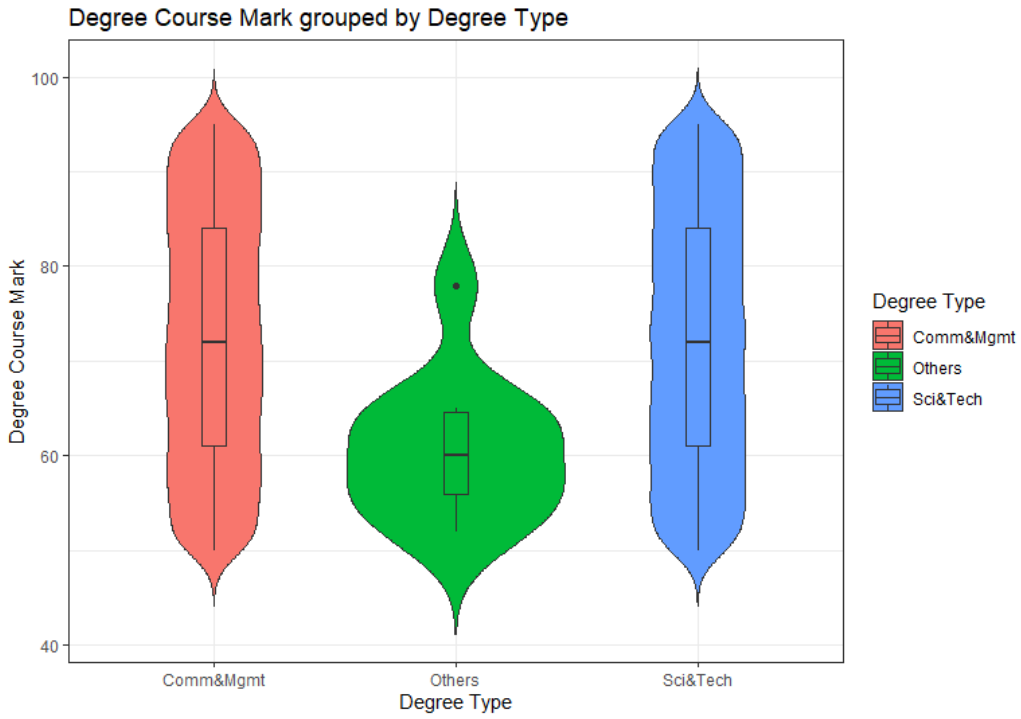


Figure 57 Data visualisation for Q3 Analysis 1

Figure 57 above is the analysis between Degree Type and their degree percentage distribution.

We can see that there is almost no difference if we only compared Comm&Mgmt and Sci&Tech, which means if the candidates are interested in one of the category, they are able to score higher degree course mark. But at the other side, category "Others" has a very high distribution above and below 60 marks, which means the "Others" type might be difficult for them to score higher marks compared to Comm&Mgmt and Sci&Tech.

5.3.2 Analysis 2: Find the relationship between extra activities and degree percentage

```
# Analysis (3-2) : Find the relationship between extra activities and degree percentage
act_degree <- assign_data%>%
  group_by(extra_act)%>%
  summarise(degree_p)

ggplot(act_degree,aes(x=extra_act,y=degree_p,fill=extra_act)) +
  geom_violin(trim = FALSE) +
  geom_boxplot(width=0.1, fill=c("pink","steelblue")) +
  labs(title = "Student Degree Course Mark grouped by Extra Activities",
       x = "Extra Activities",
       y = "Degree Course Mark",
       fill = "Extra Activities") +
  theme_bw()
```

Figure 58 Code snippet for Q3 Analysis 2

Figure 58 above shows the code for Question 3 Analysis 2. First of all, a new table act_degree is created grouped by extra_act and degree_p to examine the degree percentage.

The ggplot() function is used to fill the data for the x-axis, y-axis and fill which is Extra Activities(extra_act) and Degree Percentage(degree_p). Geom_violin () is used to plot violin plot to support boxplot. Then the geom_boxplot() function is used to plot a boxplot. Next, the geom_text is used to add the label for count and adjust the position of label. The labs() function is used to determine the label of x-axis, y-axis legend title, graph title. Lastly, theme is also used to make the background of table. The Data Visualisation result shows at Figure 60 below (tidyverse, ggplot2(geom_boxplot), 2021).

	extra_act	degree_p
1	no	58.00
2	no	77.48
3	no	64.00
4	no	73.30
5	no	79.00
6	no	66.00
7	no	72.00
8	no	60.00
9	no	59.00
10	no	50.00

Figure 59 Data exploration for Degree Percentage grouped by Extra Activities

Figure 60 Data visualisation for Q3 Analysis 2

Figure 60 above is the analysis between Extra Activities and their degree percentage distribution.

We can see that there is almost no difference for candidate who takes or doesn't take extra activities when they are in degree. Therefore, there is no excuse for student who doesn't take extra activites to score low mark in degree.

5.3.3 Analysis 3: Find the relationship between extra classes and degree percentage

```
# Analysis (3-3) : Find the relationship between extra classes and degree percentage
class_degree <- assign_data%>%
  group_by(extra_class)%>%
  summarise(degree_p)

ggplot(class_degree,aes(x=extra_class,y=degree_p,fill=extra_class)) +
  geom_violin(trim=FALSE)+
  geom_boxplot(width=0.1,fill=c("pink","steelblue")) +
  labs(title = "Degree Course Mark grouped by Extra Classes",
       x = "Extra Classes",
       y = "Degree Course Mark",
       fill = "Extra Classes") +
  theme_bw()
```

Figure 61 Code snippet for Q3 Analysis 3

Figure 61 above shows the code for Question 3 Analysis 3. First of all, a new table class_degree is created grouped by extra_class and degree_p to examine the degree percentage.

The ggplot() function is used to fill the data for the x-axis, y-axis and fill which is Extra Classes(extra_class) and Degree Percentage(degree_p). Geom_violin () is used to plot violin plot to support boxplot. Then the geom_boxplot() function is used to plot a boxplot. Next, the geom_text is used to add the label for count and adjust the position of label. The labs() function is used to determine the label of x-axis, y-axis legend title, graph title. Lastly, theme is also used to make the background of table. The Data Visualisation result shows at Figure 63 below (tidyverse, ggplot2(geom_boxplot), 2021).

	extra_class	degree_p
1	no	58.00
2	no	77.48
3	no	79.00
4	no	66.00
5	no	78.30
6	no	50.00
7	no	69.00
8	no	64.00
9	no	64.00
10	no	66.00

Figure 62 Data Exploration for Degree Percentage grouped by extra Class

Figure 63 Data visualisation for Q3 Analysis 3

Figure 63 above is the analysis between Extra Classes and their degree percentage distribution.

We can see that there is almost no difference for candidate who takes or doesn't take extra classes when they are in degree. Therefore, it might not be a factor to affect Degree Course Mark but it is still better to take extra class if it is affordable since it could help to strengthen basic knowledges.

5.3.4 Analysis 4: Find the relationship between family support and degree percentage

```
# Analysis (3-4) : Find the relationship between family education support and degree percentage

famsup_d <- assign_data%>%
  group_by(famsup)%>%
  summarise(degree_p)

ggplot(famsup_d, aes(x = famsup, y = degree_p, fill = famsup)) +
  geom_violin(trim=FALSE) +
  geom_boxplot(width = 0.1, fill=c("pink","steelblue")) +
  labs(title = "Degree Course Mark grouped by family education support",
       x = "Family Support",
       y = "Degree Course Mark",
       fill = "Family Support") +
    theme_bw()
```

Figure 64 Code snippet for Q3 Analysis 4

Figure 64 above shows the code for Question 3 Analysis 4. First of all, a new table famsup_d is created grouped by famsup and degree_p to examine the degree percentage.

The ggplot() function is used to fill the data for the x-axis, y-axis and fill which is Family Educational Support (famsup) and Degree Percentage(degree_p). Geom_violin () is used to plot violin plot to support boxplot. Then the geom_boxplot() function is used to plot a boxplot. Next, the geom_text is used to add the label for count and adjust the position of label. The labs() function is used to determine the label of x-axis, y-axis legend title, graph title. Lastly, theme is also used to make the background of table. The Data Visualisation result shows at Figure 66 below (tidyverse, ggplot2(geom_boxplot), 2021).

	famsup	degree_p
1	no	58.00
2	no	64.00
3	no	79.00
4	no	70.00
5	no	66.00
6	no	72.23
7	no	66.00
8	no	81.00
9	no	81.00
10	no	57.00

Figure 65 Data exploration for Degree Percentage grouped by family educational support

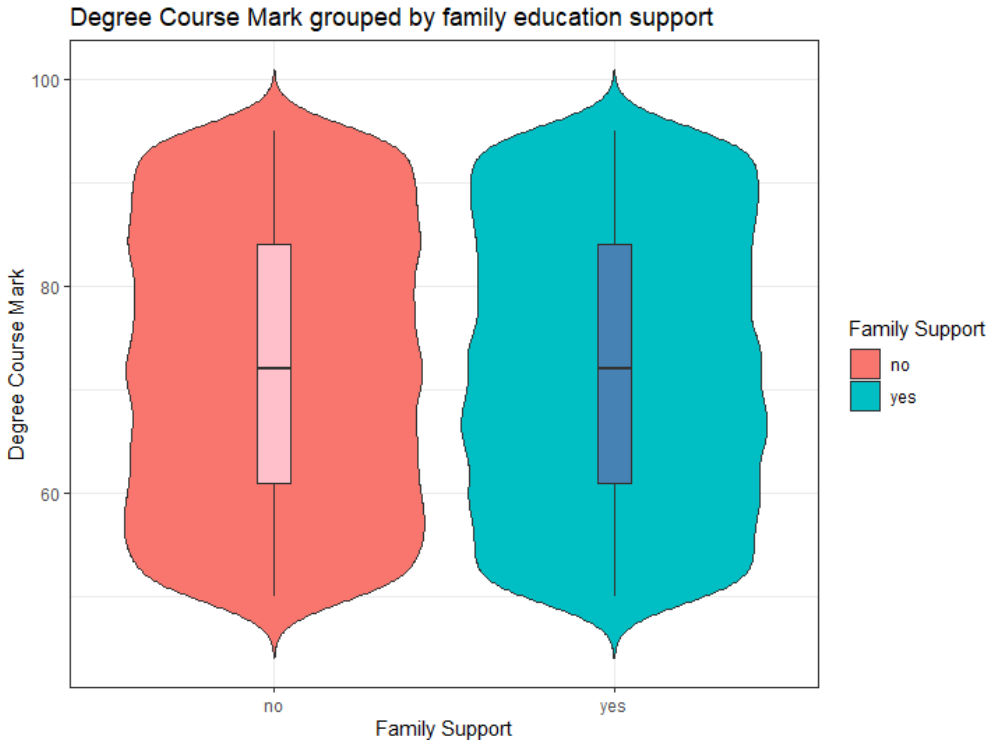


Figure 66 Data visualisation for Q3 Analysis 4

Figure 66 above is the analysis between Extra Classes and their degree percentage distribution.

We can see that there is almost no difference for candidate who has or hasn't educational support from family. Therefore, it is not be a factor to affect Degree Course Mark but it is still better to have educational support from family since there is more candidates score around below 60% for candidates who doesn't has educational support compared to who has educational support.

5.3.5 Analysis 5: Find the relationship between family education level and degree percentage

```
# Analysis (3-5) : Find the relationship between Parent Education Level and degree percentage
pedu_degree <- assign_data%>%
  group_by(Medu, Fedu)%>%
  summarise(degree_p)

ggplot(pedu_degree, aes(x = Medu, y = degree_p, fill = Fedu)) +
  geom_boxplot(width=0.5,position = position_dodge(0.9)) +
  labs(title = "Degree course mark of students grouped by mother's education level",
       x = "Mother's Education Level",
       y = "Average Grade",
       fill = "Father's Education Level") +
  theme_bw()
```

Figure 67 Code snippet for Q3 Analysis 5

Figure 67 above shows the code for Question 3 Analysis 5. First of all, a new table pedu_degree is created grouped by Medu, Fedu and degree_p to examine the degree percentage.

The ggplot() function is used to fill the data for the x-axis, y-axis and fill which is Mother Education Level(Medu), Father educational value(Fedu) and Degree Percentage(degree_p). Geom_violin () is used to plot violin plot to support boxplot. Then the geom_boxplot() function is used to plot a boxplot. Next, the geom_text is used to add the label for count and adjust the position of label. The labs() function is used to determine the label of x-axis, y-axis legend title, graph title. Lastly, theme is also used to make the background of table. The Data Visualisation result shows at Figure 69 below (tidyverse, ggplot2(geom_boxplot), 2021).

	Medu	Fedu	degree_p
1	None	Primary	69.00
2	None	Primary	72.00
3	None	5th to 9th grade	77.00
4	None	5th to 9th grade	83.00
5	None	5th to 9th grade	87.00
6	None	5th to 9th grade	80.00
7	None	5th to 9th grade	95.00
8	Primary	None	72.00
9	Primary	None	69.00
10	Primary	Primary	77.48

	Medu	Fedu	count
1	None	Primary	2
2	None	5th to 9th grade	5
3	Primary	None	2
4	Primary	Primary	1205
5	Primary	5th to 9th grade	996
6	Primary	Secondary	1004
7	Primary	Higher	963
8	5th to 9th grade	Primary	1040
9	5th to 9th grade	5th to 9th grade	1092
10	5th to 9th grade	Secondary	1086
11	5th to 9th grade	Higher	1037
12	Secondary	Primary	1073
13	Secondary	5th to 9th grade	1088
14	Secondary	Secondary	1072
15	Secondary	Higher	1013
16	Higher	None	2
17	Higher	Primary	993
18	Higher	5th to 9th grade	1074
19	Higher	Secondary	1076
20	Higher	Higher	1184

Figure 68 Data exploration for Degree percentage grouped by Mother Education Level and Father Education Level

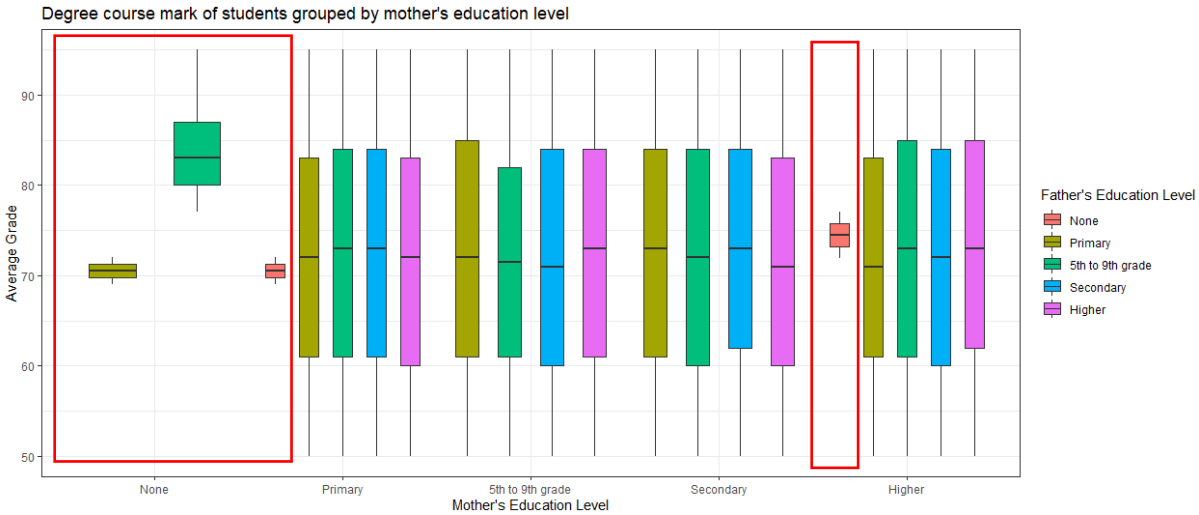


Figure 69 Data visualisation for Q3 Analysis 5

Figure 69 above is the analysis between Parent Education Level and degree percentage distribution.

We can see that parent educational level is not really a factor to affect the degree course mark if we exclude the data in the red box. All the boxplot (exclude boxplot in the red box), each boxplot has the same maximum and mininum and the distribution of their degree course mark is almost the same. Therefore, if one has the passionate to study well, parent's education level is not really affect them to score a good mark.

5.3.6 Analysis 6: Find the relationship between internet access and degree percentage

```
# Analysis (3-6) : Find the relationship between internet access and degree percentage
internet_n <- assign_data%>%
  group_by(internet)%>%
  summarise(count=n())

internet_degree <- assign_data%>%
  group_by(internet)%>%
  summarise(degree_p)

ggplot(internet_degree,aes(x=internet,y=degree_p,fill=internet)) +
  geom_violin(trim=FALSE) +
  geom_boxplot(width=0.1, fill = c("pink","steelblue")) +
  labs(title = "Degree Course Mark grouped by Internet Access and Living Place",
       x = "Internet Access",
       y = "Degree Course Mark",
       fill = "Internet Access") +
  theme_bw()
```

Figure 70 Code Snippet for Q3 Analysis 6

Figure 70 above shows the code for Question 3 Analysis 6. First of all, a new table internet_degree is created grouped by internet and degree_p to examine the degree percentage.

The ggplot() function is used to fill the data for the x-axis, y-axis and fill which is Internet Access(internet) and Degree Percentage(degree_p). Geom_violin () is used to plot violin plot to support boxplot. Then the geom_boxplot() function is used to plot a boxplot. Next, the geom_text is used to add the label for count and adjust the position of label. The labs() function is used to determine the label of x-axis, y-axis legend title, graph title. Lastly, theme is also used to make the background of table. The Data Visualisation result shows at Figure 72 below (tidyverse, ggplot2(geom_boxplot), 2021).

	internet	degree_p
1	no	58.00
2	no	73.30
3	no	66.00
4	no	64.00
5	no	72.00
6	no	64.00
7	no	68.00
8	no	53.00
9	no	74.00
10	no	72.00

	internet	count
1	no	8290
2	yes	8717

Figure 71 Data Exploration for Degree Percentage grouped by Internet

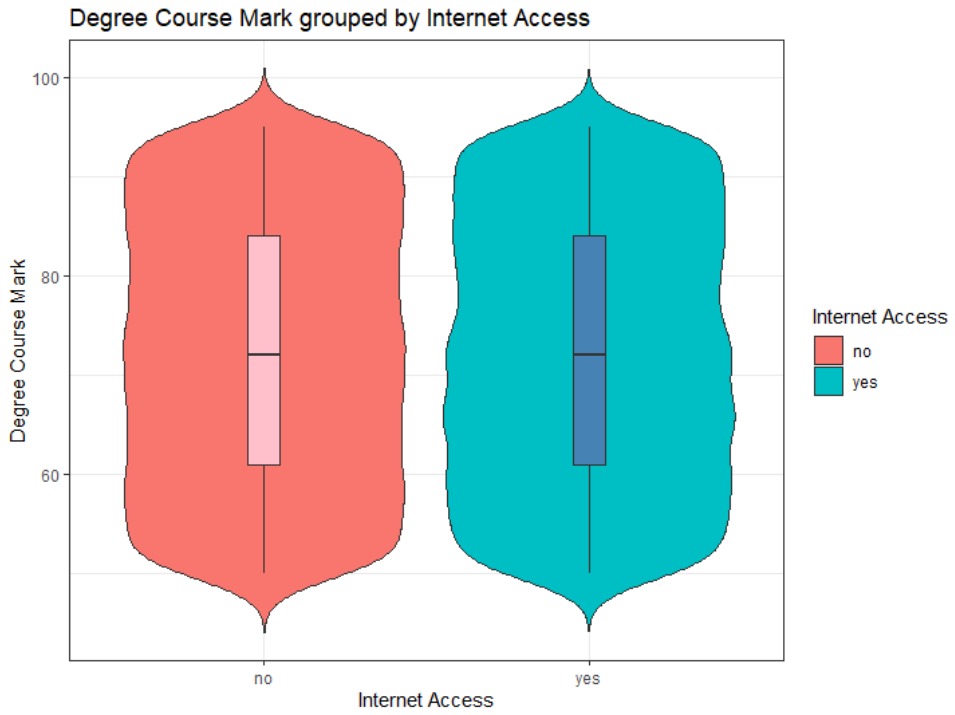


Figure 72 Data visualisation for Q3 Analysis 6

Figure 72 above is the analysis between Internet Access and degree percentage distribution.

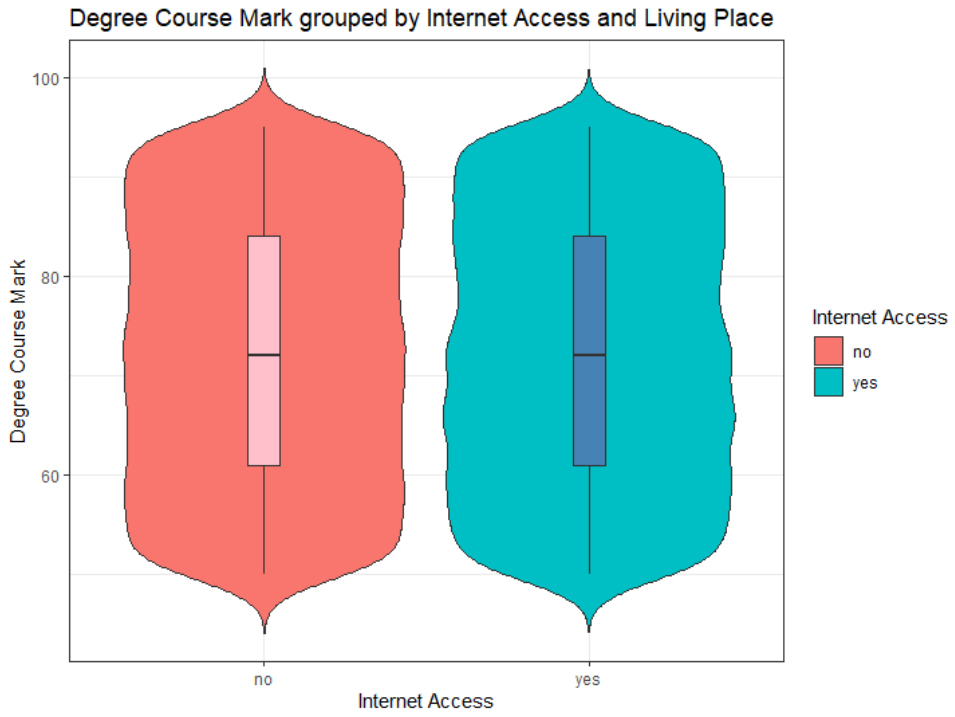
We can see that there is almost no difference for candidate who has or hasn't internet access at home. Therefore, it is not be a factor to affect Degree Course Mark but it is still better to have internet access since they can explore more on internet to learn about new knowledges and also take some online courses to learn more for their courses.

6.0 Extra Features

6.1 Theme_bw()

```
ggplot(internet_degree,aes(x=internet,y=degree_p,fill=internet)) +
  geom_violin(trim=FALSE) +
  geom_boxplot(width=0.1, fill = c("pink","steelblue")) +
  labs(title = "Degree Course Mark grouped by Internet Access and Living Place",
       x = "Internet Access",
       y = "Degree Course Mark",
       fill = "Internet Access") +
  theme_bw()
```

Figure above shows the code for using the theme bw() function, which is the second new feature used in this data analysis project. The theme bw() function is used to create a custom theme with a white backdrop and black grid lines (tidyverse, ggplot2, 2021). The result show below:



This theme bw() function is used in the majority of the graphs for this assignment. The reason for using this function is because of a white background against a graph makes it stand out more clearly for study and visualisation purpose. A graph with a white backdrop and black gridlines provides more contrast between the variables' results.

6.2 Geom_text()

```
ggplot(wx_salary,aes(x=workex,y=salary,fill=workex)) +
  geom_bar(stat="identity",width=0.5) +
  scale_fill_manual(values = c("pink","steelblue")) +
  geom_text(aes(label=salary),position=position_dodge(0.9),vjust=-0.25) +
  labs(title = "Analysis between Working Experience and Salary",
       x = "Working Experience",
       y = "Salary",
       fill = "Working Experience") +
  theme_bw()
```

Figure above shows that geom_text() has been used. This method is used to create the label for the bar chart. The parameter "vjust=-0.25" is used to ensure that the label is placed above the bar (tidyverse, ggplot2(geom_text), 2021). The result is as the following figure:

7.0 Conclusion

In conclusion, I hope that every analysis I have done could help candidates to see which factors could have affect them in their life. Every necessary steps before starting analysis have been implemented.

Special thanks to Miss Minnu Helen Joseph who have taught me on R Language and how to use RStudio. This is a good assignment for me to practice more about data analysis skill to prepare me well in my future career in data analytics field.

References

N.A. (2021). *Themes ggplot*. Retrieved from Applied R Code: http://applied-r.com/themes-ggplot/

RDocumentation. (30 1, 2021). *RDocumentation*. Retrieved from RDocumentation: https://www.rdocumentation.org/packages/base/versions/3.6.2/topics/apply

tidyverse. (30 1, 2021). *ggplot2*. Retrieved from ggplot2: https://ggplot2.tidyverse.org/reference/ggtheme.html

tidyverse. (30 1, 2021). *ggplot2(geom_bar)*. Retrieved from https://ggplot2.tidyverse.org/reference/geom_bar.html

tidyverse. (30 1, 2021). *ggplot2(geom_boxplot)*. Retrieved from https://ggplot2.tidyverse.org/reference/geom_boxplot.html

tidyverse. (30 1, 2021). *ggplot2(geom_text)*. Retrieved from ggplot2(geom_text): https://ggplot2.tidyverse.org/reference/geom_text.html

tidyverse. (6 May, 2021). *Text*. Retrieved from ggplot2: https://ggplot2.tidyverse.org/reference/geom_text.html