Students Survey

To analyze your survey data in R, I will outline a few steps and provide sample code for each:

1. **Read and Inspect Data**: Import the data into R and take an initial look.
2. **Data Cleaning**: Handle missing values, rename columns, or adjust data types if necessary.
3. **Exploratory Data Analysis (EDA)**: Generate summary statistics and visualizations.
4. **Insights**: Answer specific questions such as GPA trends, work status impact, and satisfaction levels.

Here's the R script to get started:

Step 1: Import Data

```
# Load necessary library
library(readr)
library(ggplot2)
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
```

```
# Replace 'your_file.csv' with the actual CSV file path
survey_data <- read_csv("students_survey_307307_20241_renamed.csv")
```

```
Rows: 28 Columns: 25

── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
chr (14): Start time, Completion time, Email, Gender, year, distance_to_uni,...
dbl (10): Id, Age, high_school_grade, study_hours, sleeping_hours, mid_score...
lgl  (1): Name

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Inspect Data

```
# Inspect the data
View(head(survey_data))
```

Data Cleaning

```
# Convert columns to appropriate data types
survey_data$Gender <- as.factor(survey_data$Gender)
survey_data$year <- as.factor(survey_data$year)
```

Check for missing data

```
# Check for missing values
View(colSums(is.na(survey_data)))
```

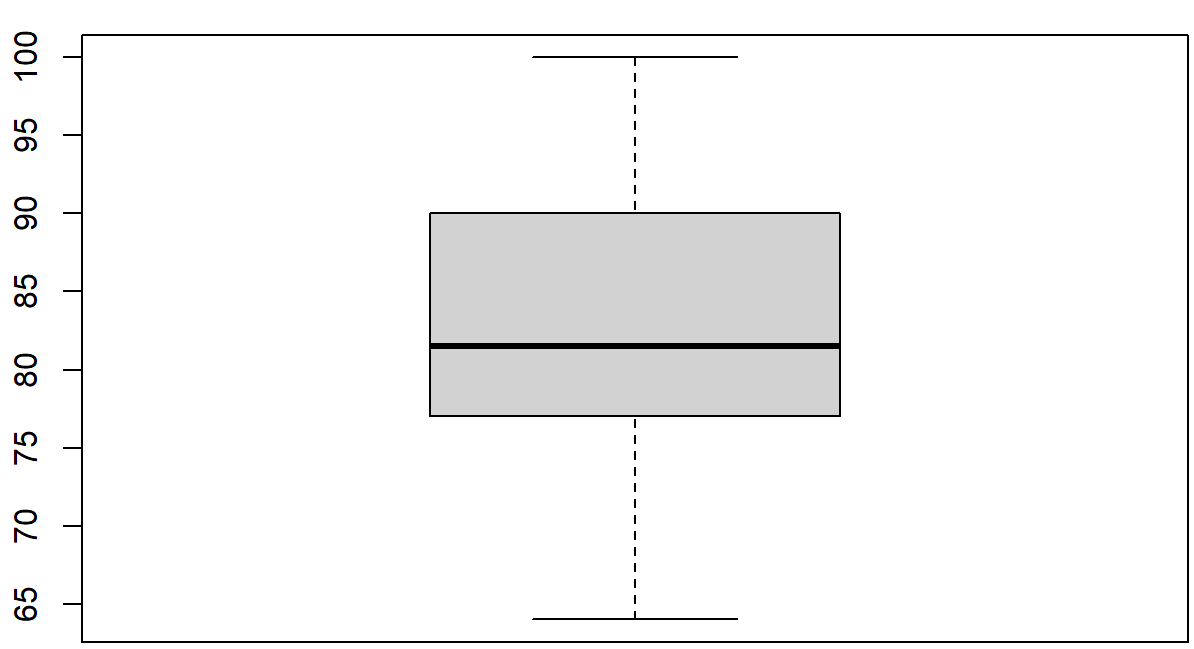
Numerical Variables:

1. Distribution patterns for numerical variables

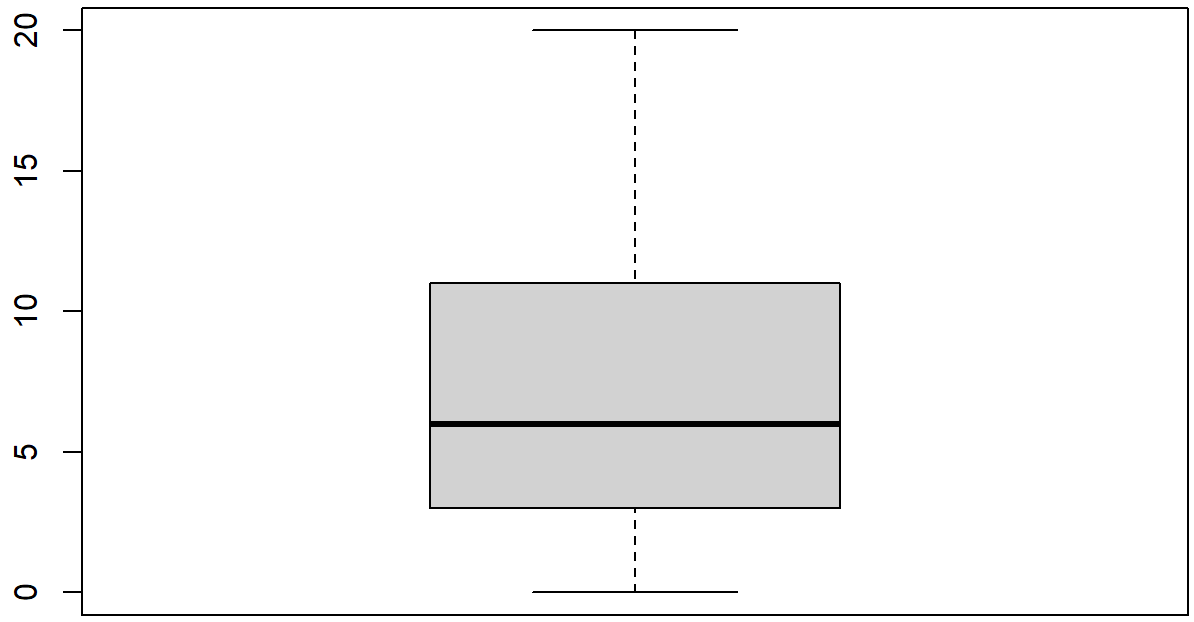
```
numerical_columns <- summary(survey_data[c("Age", "high_school_grade", "study_hours", "sleeping_h

View(numerical_columns)
```

2. Outliers in numerical variables

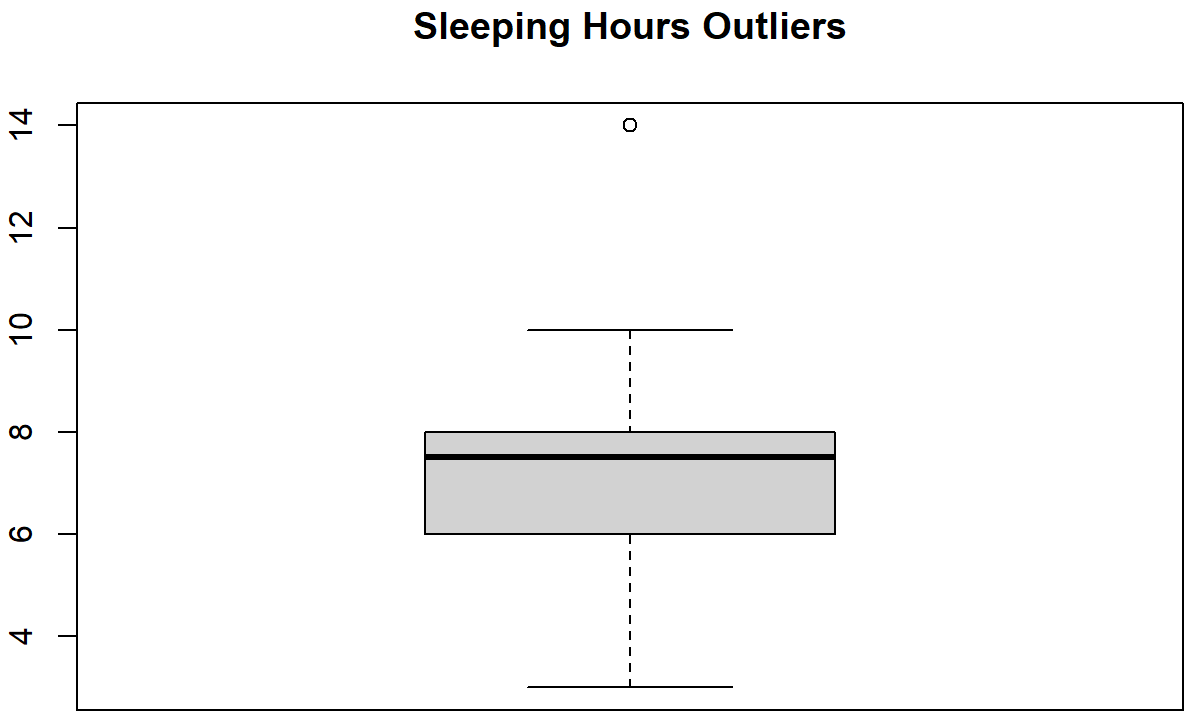
```
boxplot(survey_data$high_school_grade, main="High School Grade Outliers")
```




```
boxplot(survey_data$study_hours, main="Study Hours Outliers")
```




```
boxplot(survey_data$sleeping_hours, main="Sleeping Hours Outliers")
```



3. Correlation between study_hours and mid_score

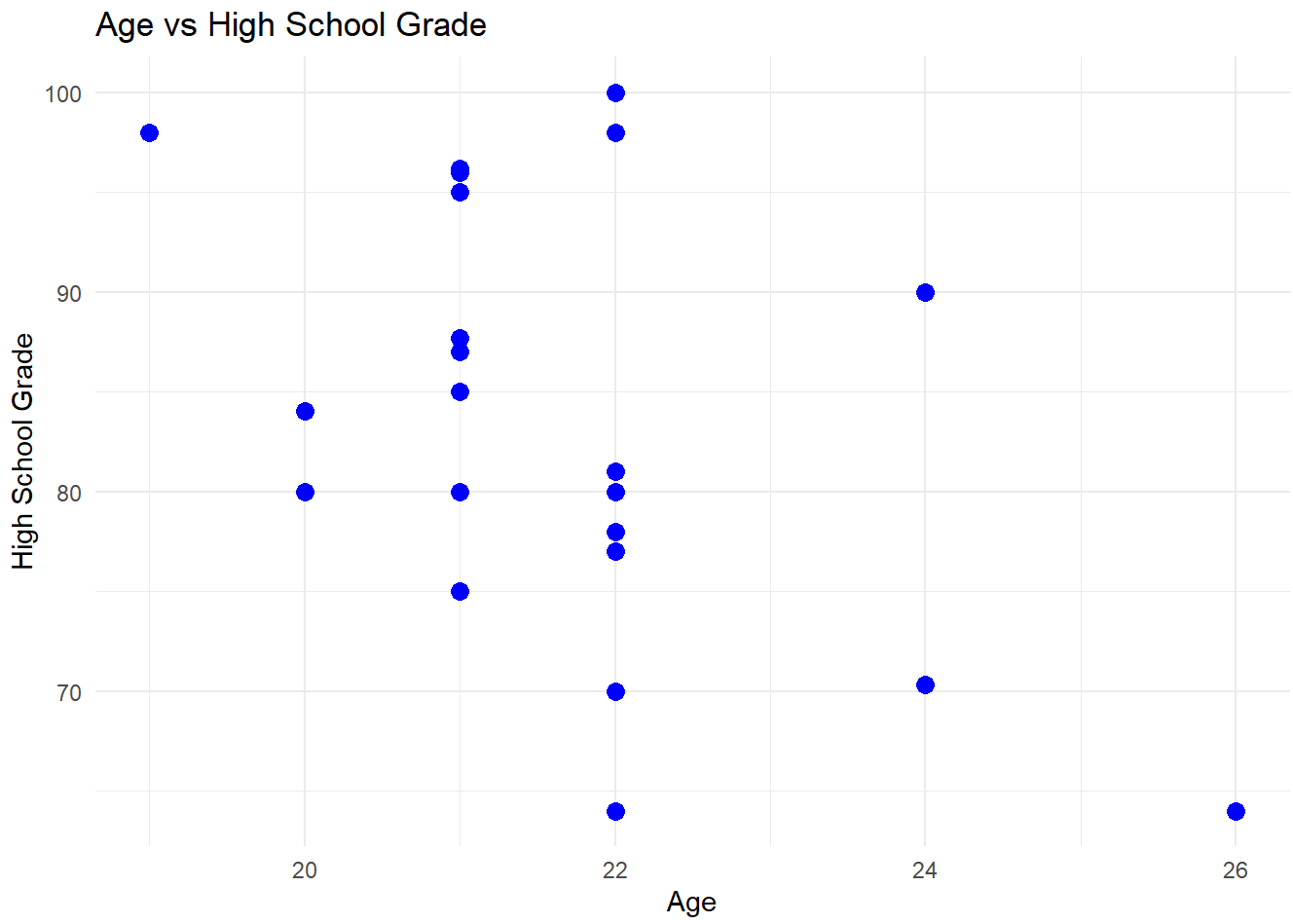
```
cor(survey_data$study_hours, survey_data$mid_score, use="complete.obs")
```

```
[1] 0.2201416
```

4. Age variation with high_school_grade or GPA

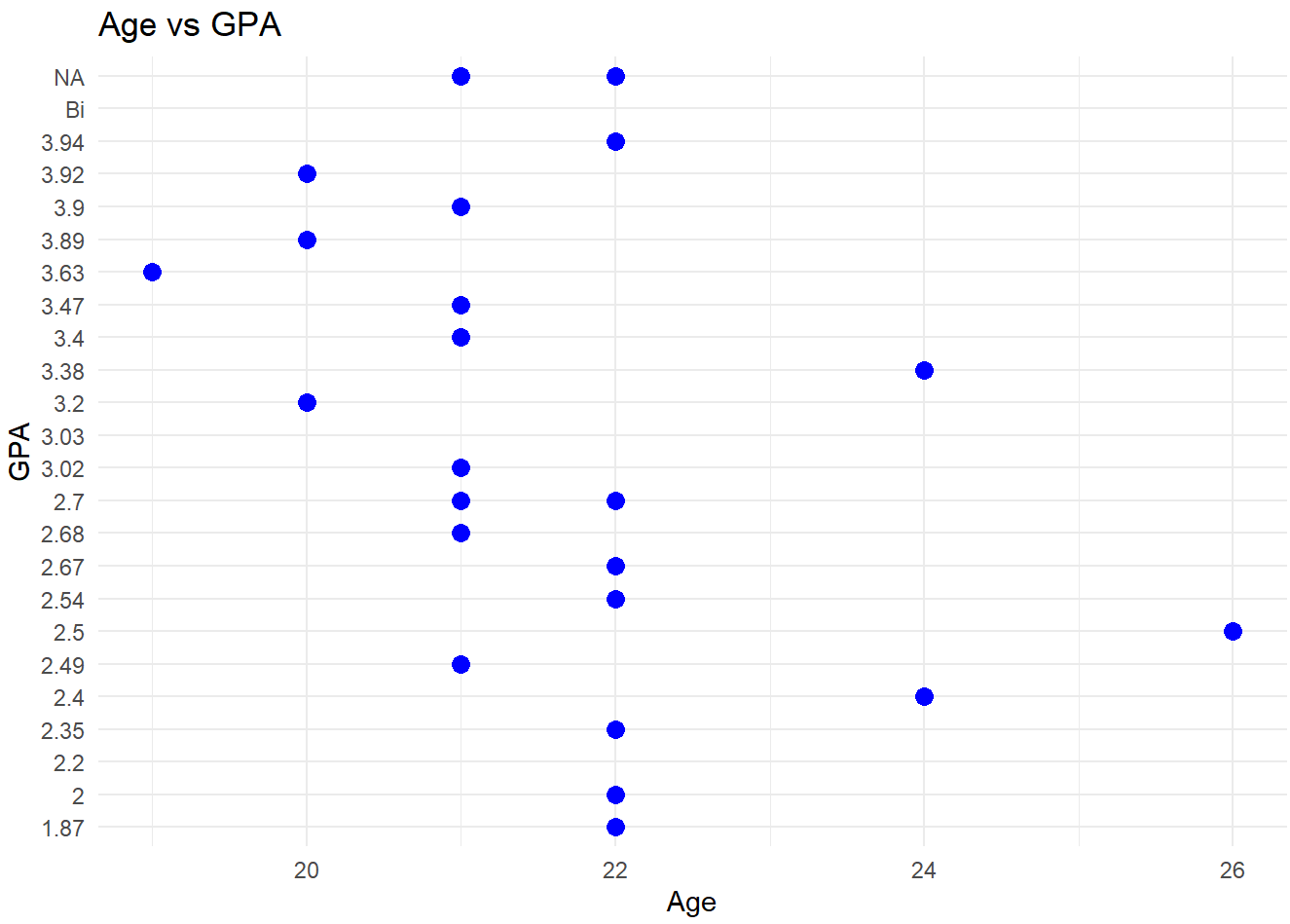
```
# Assuming your data frame is called `survey_data`
ggplot(survey_data, aes(x = Age, y = high_school_grade)) +
  geom_point(color = "blue", size = 3) +  # Adds scatterplot points
  labs(title = "Age vs High School Grade", x = "Age", y = "High School Grade") +
  theme_minimal()  # Applies a clean minimal theme
```

```
Warning: Removed 4 rows containing missing values or values outside the scale range
(`geom_point()`).
```




```
ggplot(survey_data, aes(x = Age, y = gpa)) +
  geom_point(color = "blue", size = 3) +  # Adds scatterplot points
  labs(title = "Age vs GPA", x = "Age", y = "GPA") +
  theme_minimal()  # Applies a clean minimal theme
```

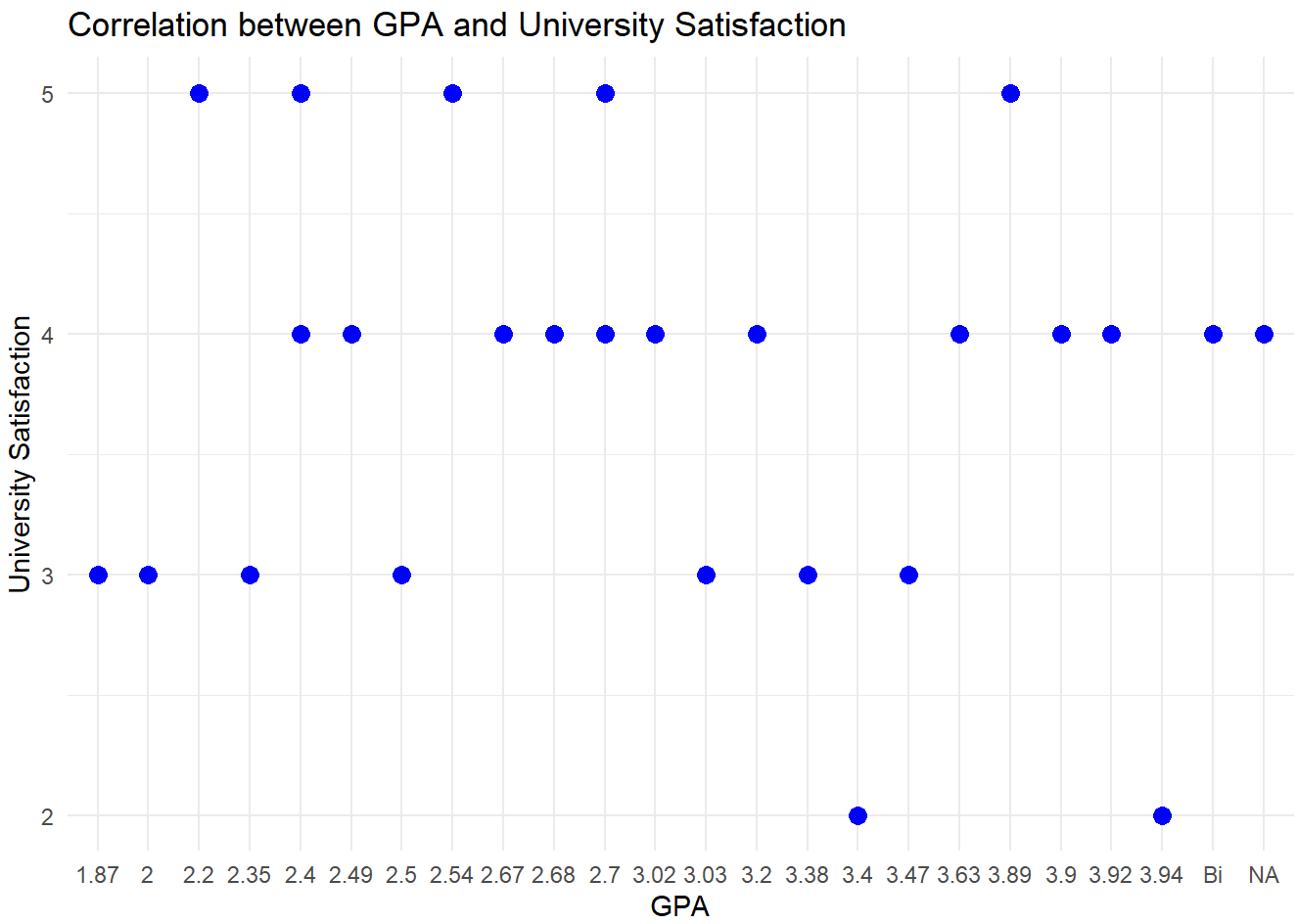
```
Warning: Removed 4 rows containing missing values or values outside the scale range
(`geom_point()`).
```



5. University_satisfaction vs GPA, Is there a relation?

```
ggplot(survey_data, aes(x = gpa, y = university_satisfaction)) +
  geom_point(color = "blue", size = 3) +  # Scatterplot
  geom_smooth(method = "lm", se = FALSE, color = "red") +  # Add regression line
  labs(title = "Correlation between GPA and University Satisfaction",
       x = "GPA", y = "University Satisfaction") +
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
```




```
# Ensure GPA and university_satisfaction are numeric
survey_data$gpa <- as.numeric(as.character(survey_data$gpa))
```

```
Warning: NAs introduced by coercion
```

```
survey_data$university_satisfaction <- as.numeric(survey_data$university_satisfaction)

# Calculate correlation
correlation <- cor(survey_data$gpa, survey_data$university_satisfaction, use = "complete.obs")
print(paste("Correlation between GPA and University Satisfaction: ", round(correlation, 2)))
```

```
[1] "Correlation between GPA and University Satisfaction:  -0.09"
```

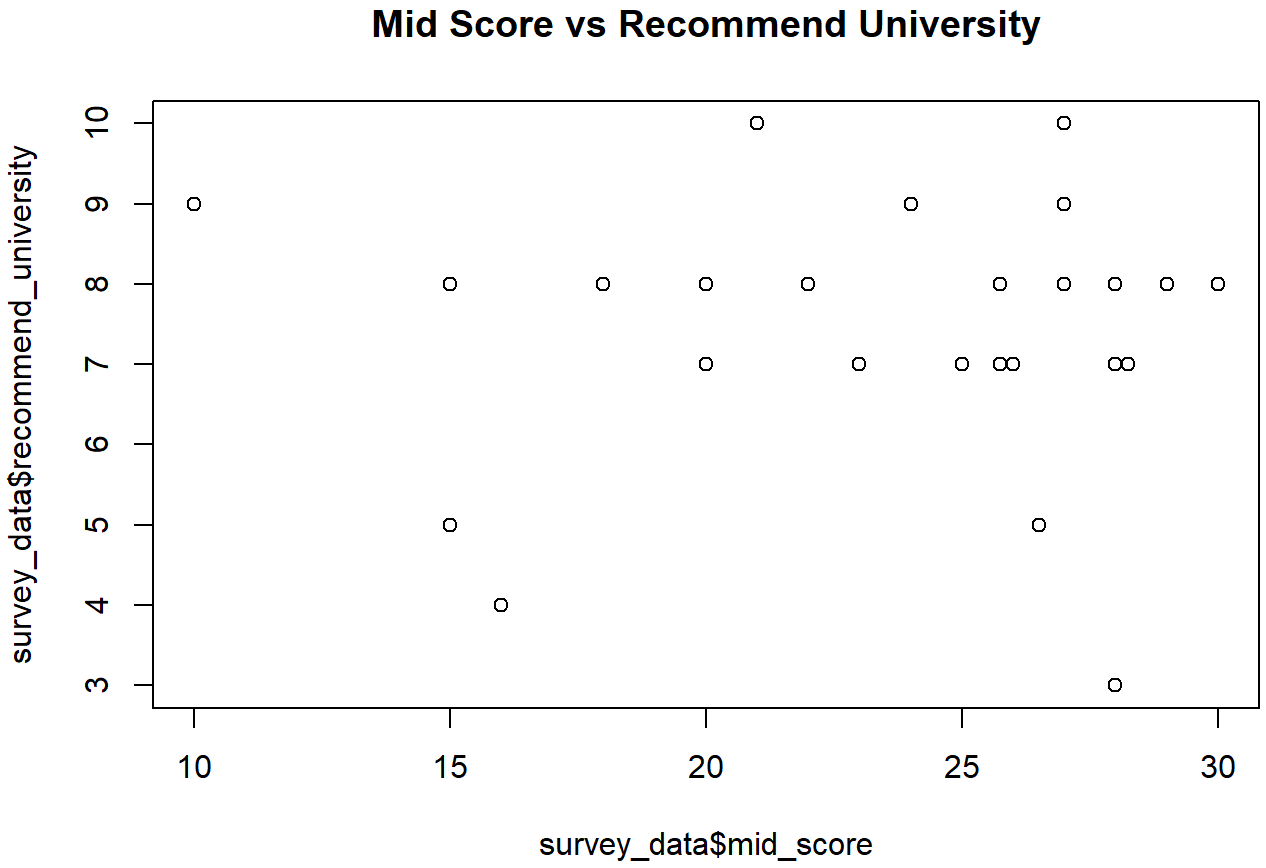
6. Is there a correlation between sleeping_hours vs university_satisfaction?

```
cor(survey_data$sleeping_hours, survey_data$university_satisfaction, use="complete.obs")
```

```
[1] -0.01644064
```

7. Trends between mid_score and recommend_university

```
plot(survey_data$mid_score, survey_data$recommend_university, main="Mid Score vs Recommend Univers
```



Categorical Variables:

1. Distribution of categorical variables

Gender Distribution

```
table(survey_data$Gender)
```

```
Female   Male
    16     12
```

Year Distribution

```
table(survey_data$year)
```

```
 Fifth Fourth Second  Third
```

```
      2      18       1       7
```

Work Distribution

```
table(survey_data$work)
```

```
No Yes
17  11
```

2. University satisfaction across high_school_category

```
# Calculate average university_satisfaction by high_school_category
survey_data %>%
  group_by(high_school_category) %>%
  summarise(avg_university_satisfaction = mean(university_satisfaction, na.rm = TRUE)) %>%
  arrange(desc(avg_university_satisfaction))  # Optional: Sort by descending satisfaction
```

```
# A tibble: 2 × 2
  high_school_category avg_university_satisfaction
  <chr>                                      <dbl>
1 National                                    3.83
2 International                               3.25
```

The same result using aggregate method

```
aggregate(university_satisfaction ~ high_school_category, data = survey_data, mean)
```

```
  high_school_category university_satisfaction
1        International                 3.250000
2             National                 3.833333
```

3. Distance_to_uni and recommend_university scores

```
aggregate(recommend_university ~ distance_to_uni, data = survey_data, mean)
```

```
  distance_to_uni recommend_university
1         > 10 KM              7.750000
2          0-5 KM              7.250000
3         5-10 KM              7.285714
```

4. Business owners vs employees

```
table(survey_data$business_owner)
```

```
Business Owner       Employee
             2              9
```

```
table(survey_data$type_of_work)
```

```
Full-Time Part-Time
        4         7
```

5. Gender and work type on bida_satisfaction

```
aggregate(bida_satisfaction ~ Gender + work, data = survey_data, mean)
```

```
  Gender work bida_satisfaction
1 Female   No          4.071429
2   Male   No          4.333333
3 Female  Yes          3.500000
4   Male  Yes          4.666667
```

6. top_concerns

```
library(stringr)
concerns <- unlist(strsplit(as.character(survey_data$top_concerns), ";"))

View(table(concerns))
```

7. Recommend_bida by high_school_category

```
  # Calculate average recommend_bida by high_school_category
survey_data %>%
  group_by(high_school_category) %>%
  summarise(avg_recommend_bida = mean(recommend_bida, na.rm = TRUE)) %>%
  arrange(desc(avg_recommend_bida))  # Optional: Sort by descending average
```

```
# A tibble: 2 × 2
  high_school_category avg_recommend_bida
  <chr>                             <dbl>
1 National                           7.88
2 International                      6.75
```

```
aggregate(recommend_bida ~ high_school_category, data = survey_data, mean)
```

```
  high_school_category recommend_bida
1        International          6.750
2             National          7.875
```

Mixed Analysis:

1. Mid_score and GPA variation by year

```
aggregate(mid_score ~ year, data = survey_data, mean)
```

```
    year mid_score
1  Fifth  25.00000
2 Fourth  22.61111
3 Second  28.00000
4  Third  23.46429
```

```
aggregate(as.numeric(as.character(gpa)) ~ year, data = survey_data, mean)
```

```
    year as.numeric(as.character(gpa))
1  Fifth                       2.585000
2 Fourth                       2.705625
3 Second                       2.700000
4  Third                       3.520000
```

2. Mid_score by work status

```
aggregate(mid_score ~ work, data = survey_data, mean)
```

```
  work mid_score
1   No  23.85294
2  Yes  22.15909
```

3. Distance_to_uni and study_hours/sleeping_hours

```
aggregate(study_hours ~ distance_to_uni, data = survey_data, mean)
```

```
  distance_to_uni study_hours
1         > 10 KM    7.666667
2          0-5 KM    7.666667
3         5-10 KM    5.428571
```

```
aggregate(sleeping_hours ~ distance_to_uni, data = survey_data, mean)
```

```
  distance_to_uni sleeping_hours
1         > 10 KM       8.333333
2          0-5 KM       7.111111
3         5-10 KM       5.857143
```

4. Gender and satisfaction levels

```
aggregate(university_satisfaction ~ Gender, data = survey_data, mean)
```

```
  Gender university_satisfaction
1 Female                 3.5625
2   Male                 4.0000
```

```
aggregate(bida_satisfaction ~ Gender, data = survey_data, mean)
```

```
  Gender bida_satisfaction
1 Female          4.000000
2   Male          4.583333
```

5. High_school_category and GPA/mid_score

```
aggregate(mid_score ~ high_school_category, data = survey_data, mean)
```

```
  high_school_category mid_score
1        International  23.31250
2             National  23.16667
```

```
aggregate(as.numeric(as.character(gpa)) ~ high_school_category, data = survey_data, mean)
```

```
  high_school_category as.numeric(as.character(gpa))
1        International                       3.533333
2             National                       2.803636
```

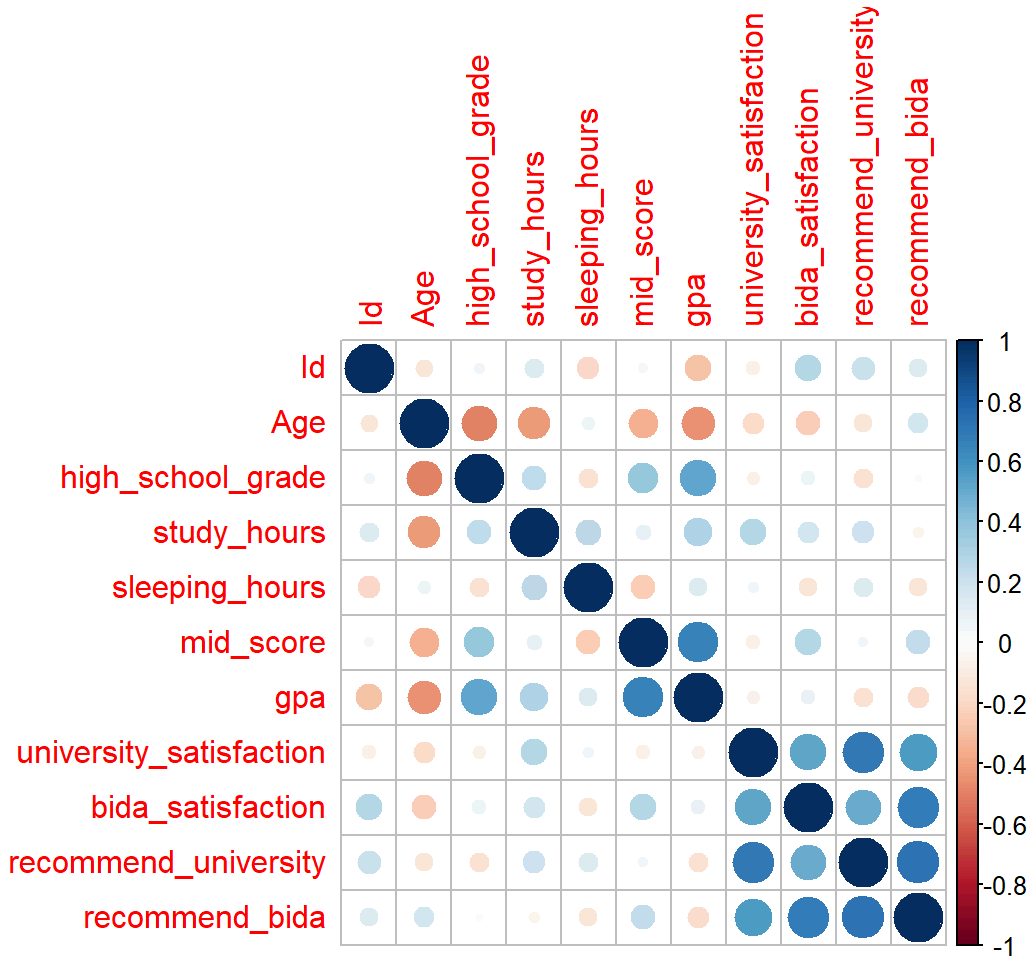
Practical Business Insights:

1. Factors most associated with university satisfaction

```
library(corrplot)
```

```
corrplot 0.95 loaded
```

```
numeric_cols <- survey_data[sapply(survey_data, is.numeric)]
corrplot(cor(numeric_cols, use="complete.obs"), method="circle")
```



2. Feedback from recommend_university and recommend_bida

```
summary(survey_data$recommend_university)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
  3.000   7.000   8.000   7.481   8.000  10.000       1 
```

```
summary(survey_data$recommend_bida)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  1.000   6.750   8.000   7.714  10.000  10.000 
```

3. Working students vs non-working students satisfaction

```
# Calculate average university_satisfaction by work status
survey_data %>%
  group_by(work) %>%
  summarise(avg_university_satisfaction = mean(university_satisfaction, na.rm = TRUE)) %>%
  arrange(desc(avg_university_satisfaction))  # Optional: Sort by descending satisfaction
```

```
# A tibble: 2 × 2
  work  avg_university_satisfaction
  <chr>                       <dbl>
1 Yes                          3.91
2 No                           3.65
```

```
aggregate(university_satisfaction ~ work, data = survey_data, mean)
```

```
  work university_satisfaction
1   No                3.647059
2  Yes                3.909091
```

4. Groups needing support based on GPA and satisfaction

```
# Ensure GPA is numeric
survey_data$gpa <- as.numeric(as.character(survey_data$gpa))

# Calculate mean GPA and university satisfaction by year
survey_data %>%
  group_by(year) %>%
  summarise(
    avg_gpa = mean(gpa, na.rm = TRUE),
    avg_university_satisfaction = mean(university_satisfaction, na.rm = TRUE)
  ) %>%
  arrange(desc(avg_gpa))  # Optional: Sort by GPA or satisfaction
```

```
# A tibble: 4 × 3
  year   avg_gpa avg_university_satisfaction
  <fct>    <dbl>                       <dbl>
1 Third     3.52                        4
2 Fourth    2.71                        3.67
3 Second    2.7                         4
4 Fifth     2.58                        3.5
```

```
aggregate(cbind(as.numeric(as.character(gpa)), university_satisfaction) ~ year, data = survey_data
```

```
    year       V1 university_satisfaction
1  Fifth 2.585000                   3.500
2 Fourth 2.705625                   3.625
3 Second 2.700000                   4.000
4  Third 3.520000                   4.000
```