proj data

Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see https://quarto.org.

Running Code

When you click the **Render** button a document will be generated that includes both content and the output of embedded code. You can embed code like this:

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to beco
```

```
library(ggplot2)
```

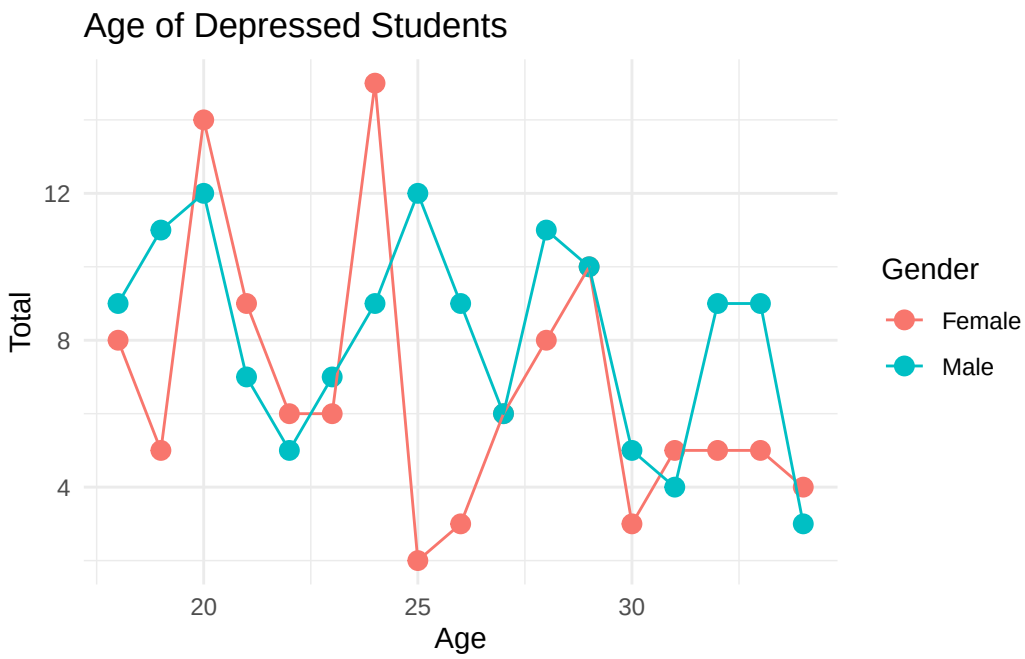
```
depression <- read.csv("Depression Student Dataset.csv")
```

You can add options to executable code like this

```r
depression_age <- depression %>% group_by(Gender, Age, Depression) %>% summarise(count=n())
```

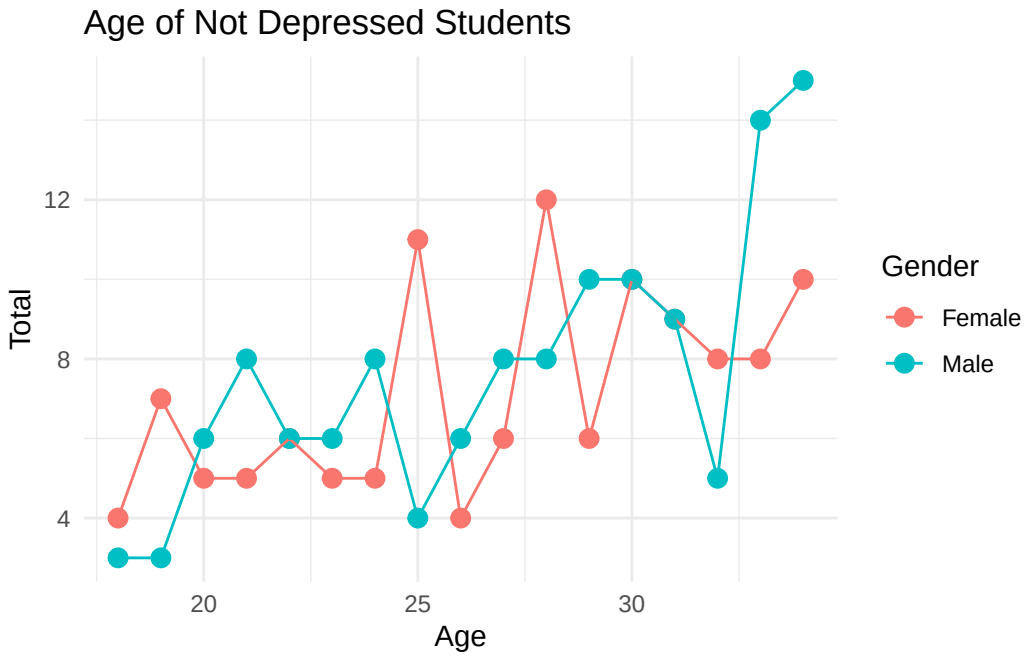
```
`summarise()` has grouped output by 'Gender', 'Age'. You can override using the
`.groups` argument.
```

```r
are_d <- depression_age %>% filter(Depression == "Yes")
are_nd <- depression_age %>% filter(Depression == "No")

ggplot(are_d, aes(x = Age, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Age of Depressed Students",
    x = "Age",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```




```r
ggplot(are_nd, aes(x = Age, y = count, color = Gender)) +
  geom_point(size = 3) +
```

```
  geom_line() +
  labs(
    title = "Age of Not Depressed Students",
    x = "Age",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```

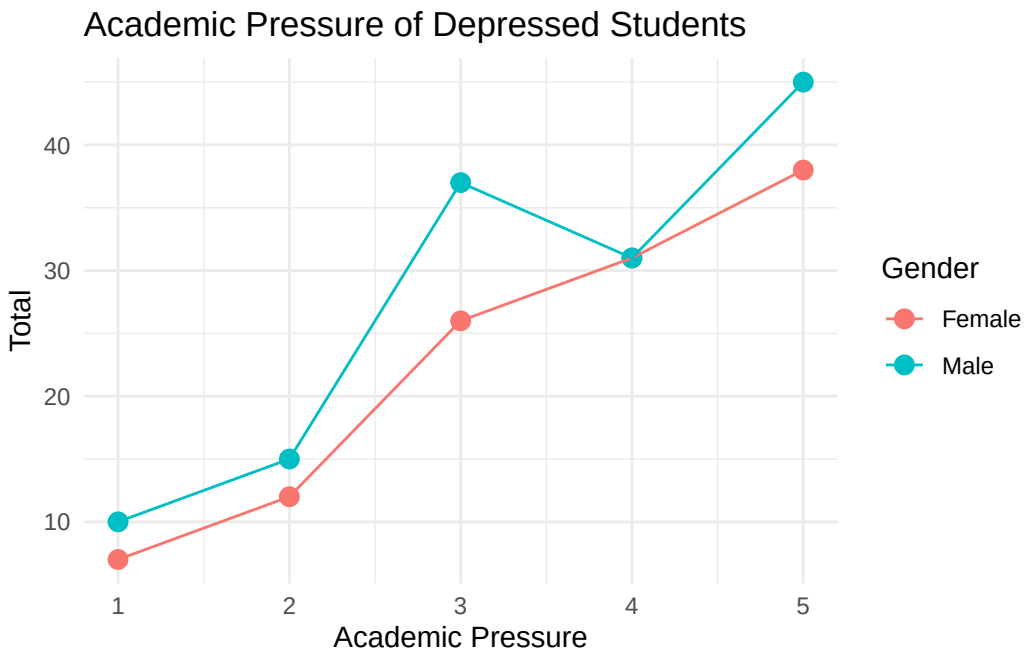



```
depression_press <- depression %>% group_by(Gender, Academic.Pressure, Depression) %>% summa
```

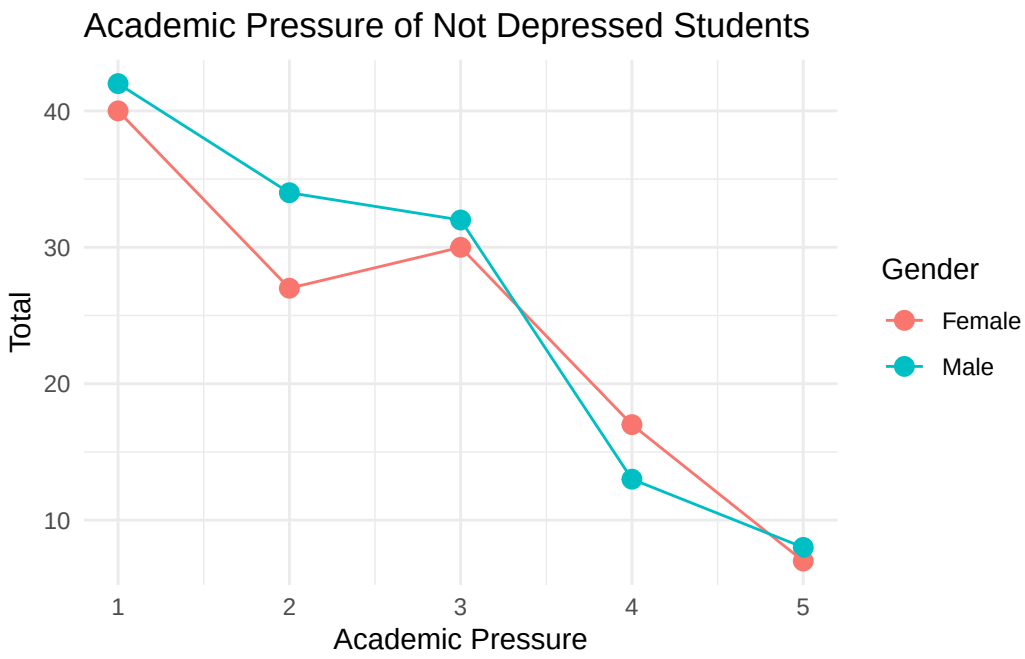
```
`summarise()` has grouped output by 'Gender', 'Academic.Pressure'. You can
override using the `.groups` argument.
```

```
are_d <- depression_press %>% filter(Depression == "Yes")
are_nd <- depression_press %>% filter(Depression == "No")

ggplot(are_d, aes(x = Academic.Pressure, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
```

```
    title = "Academic Pressure of Depressed Students",
    x = "Academic Pressure",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```



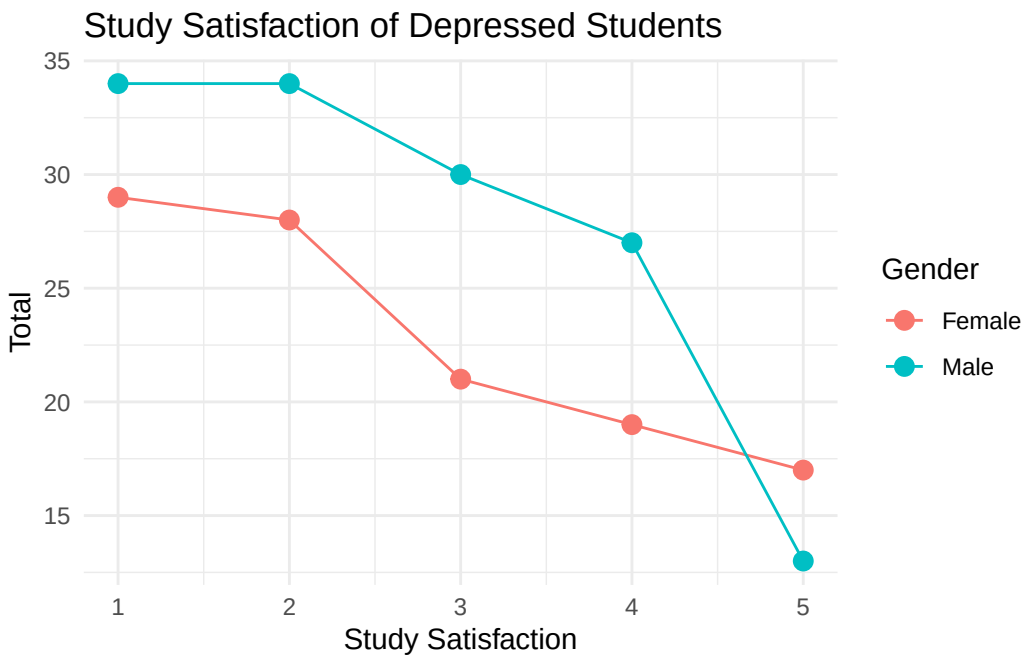

```
ggplot(are_nd, aes(x = Academic.Pressure, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Academic Pressure of Not Depressed Students",
    x = "Academic Pressure",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```



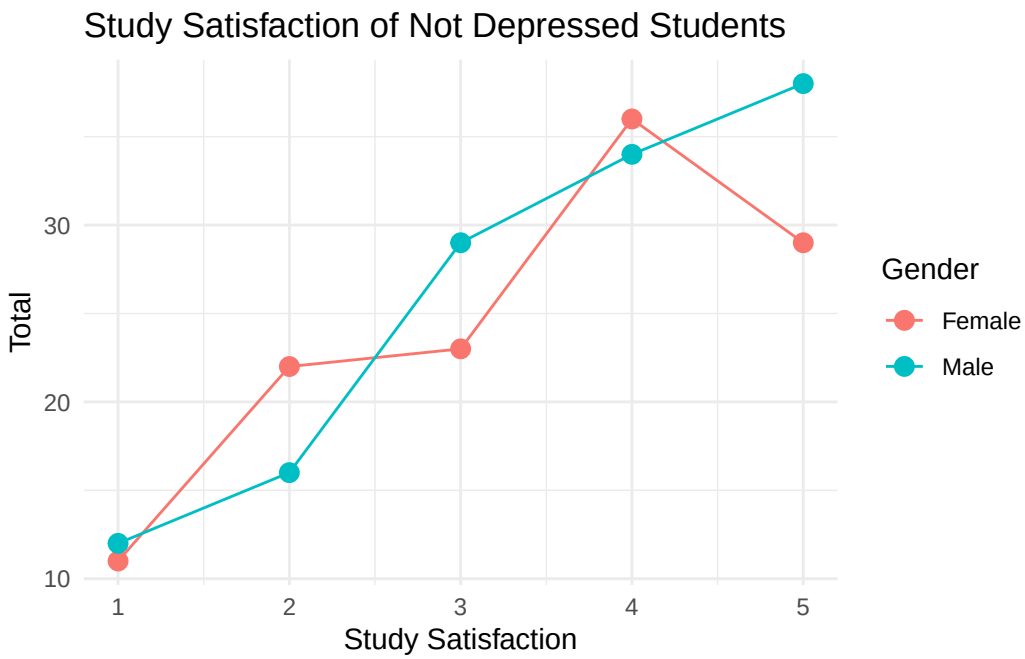

```
depression_study <- depression %>% group_by(Gender, Study.Satisfaction, Depression) %>% summa
```

```
`summarise()` has grouped output by 'Gender', 'Study.Satisfaction'. You can
override using the `.groups` argument.
```

```
are_d <- depression_study %>% filter(Depression == "Yes")
are_nd <- depression_study %>% filter(Depression == "No")

ggplot(are_d, aes(x = Study.Satisfaction, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Study Satisfaction of Depressed Students",
    x = "Study Satisfaction",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```




```
ggplot(are_nd, aes(x = Study.Satisfaction, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Study Satisfaction of Not Depressed Students",
    x = "Study Satisfaction",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```

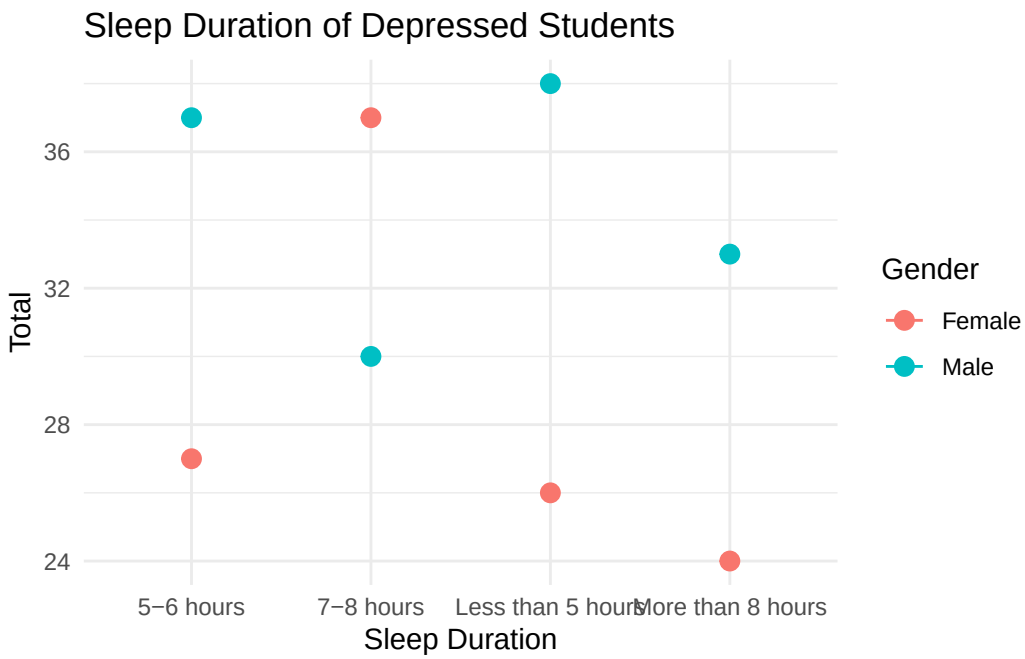



```
depression_sleep <- depression %>% group_by(Gender, Sleep.Duration, Depression) %>% summarise
```

```
`summarise()` has grouped output by 'Gender', 'Sleep.Duration'. You can
override using the `.groups` argument.
```

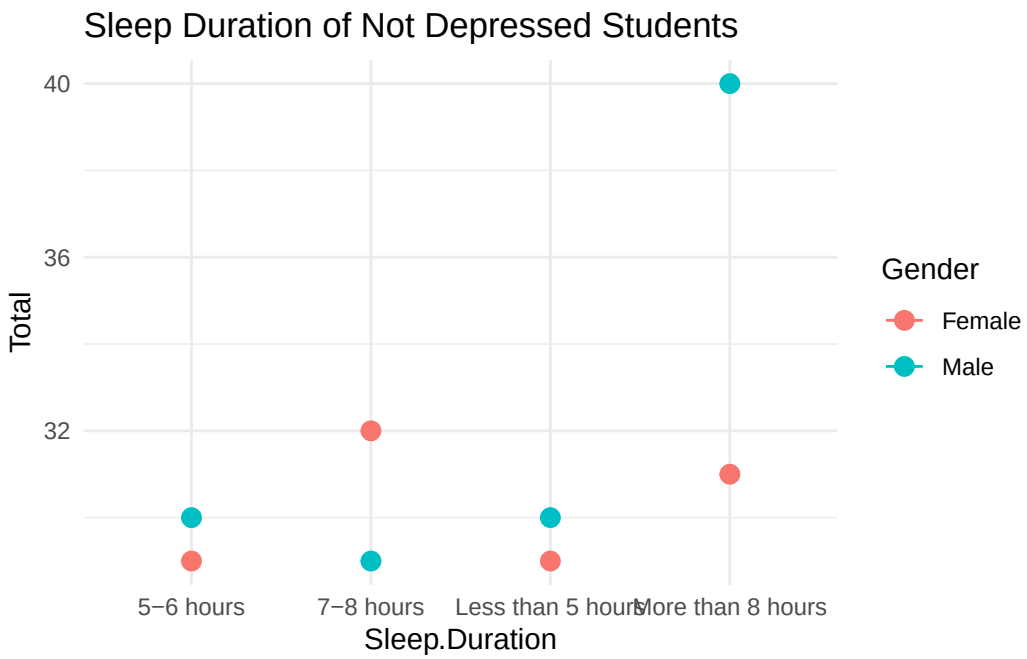
```
are_d <- depression_sleep %>% filter(Depression == "Yes")
are_nd <- depression_sleep %>% filter(Depression == "No")

ggplot(are_d, aes(x = Sleep.Duration, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Sleep Duration of Depressed Students",
    x = "Sleep Duration",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```




```
ggplot(are_nd, aes(x = Sleep.Duration, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Sleep Duration of Not Depressed Students",
    x = "Sleep.Duration",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```

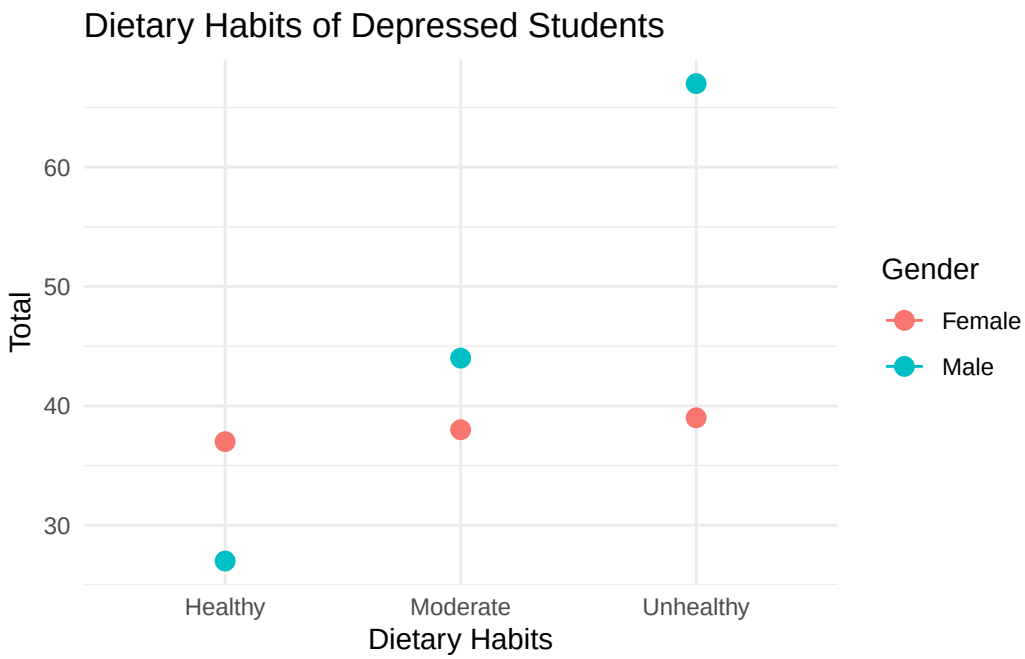



```
depression_diet <- depression %>% group_by(Gender, Dietary.Habits, Depression) %>% summarise
```

```
`summarise()` has grouped output by 'Gender', 'Dietary.Habits'. You can
override using the `.groups` argument.
```

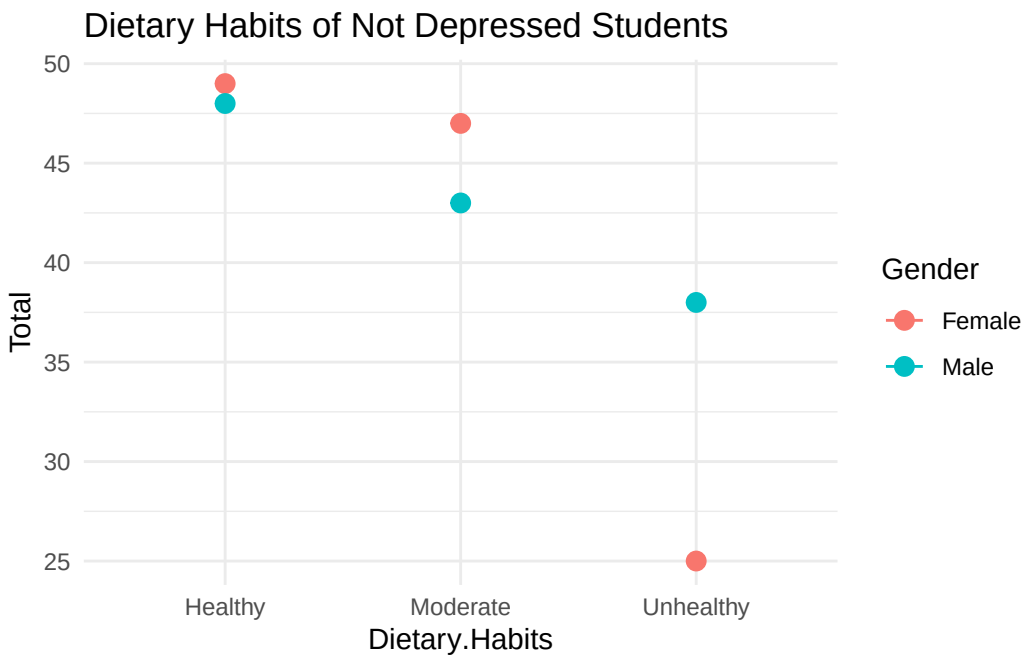
```
are_d <- depression_diet %>% filter(Depression == "Yes")
are_nd <- depression_diet %>% filter(Depression == "No")

ggplot(are_d, aes(x = Dietary.Habits, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Dietary Habits of Depressed Students",
    x = "Dietary Habits",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```




```
ggplot(are_nd, aes(x = Dietary.Habits, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Dietary Habits of Not Depressed Students",
    x = "Dietary.Habits",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```

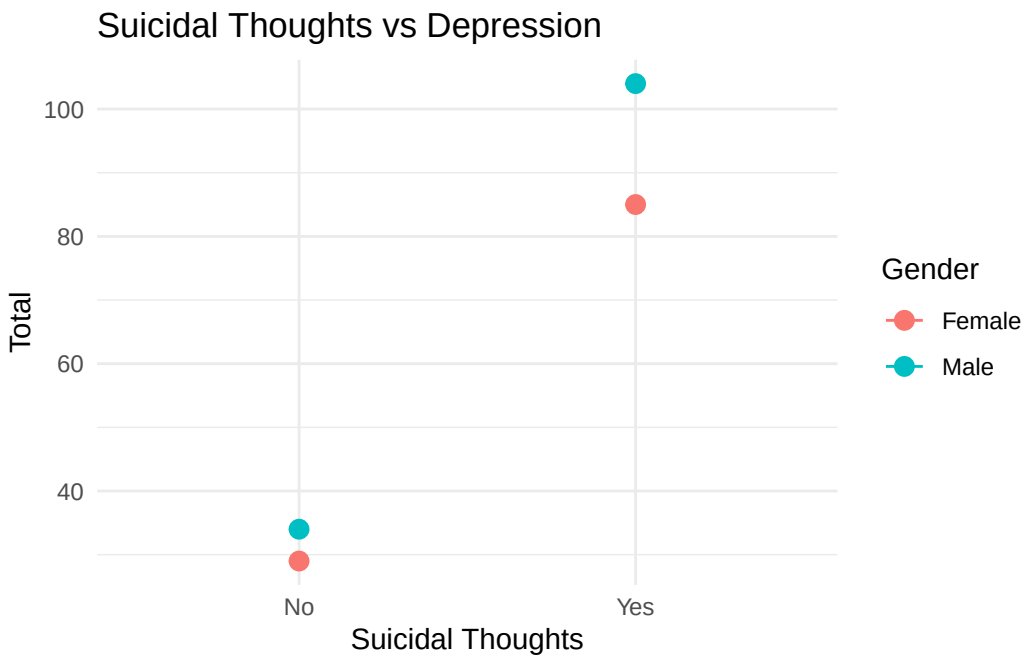



```
depression_suicidal <- depression %>% group_by(Gender, Have.you.ever.had.suicidal.thoughts..
```

```
`summarise()` has grouped output by 'Gender',
'Have.you.ever.had.suicidal.thoughts..'. You can override using the `.groups`
argument.
```

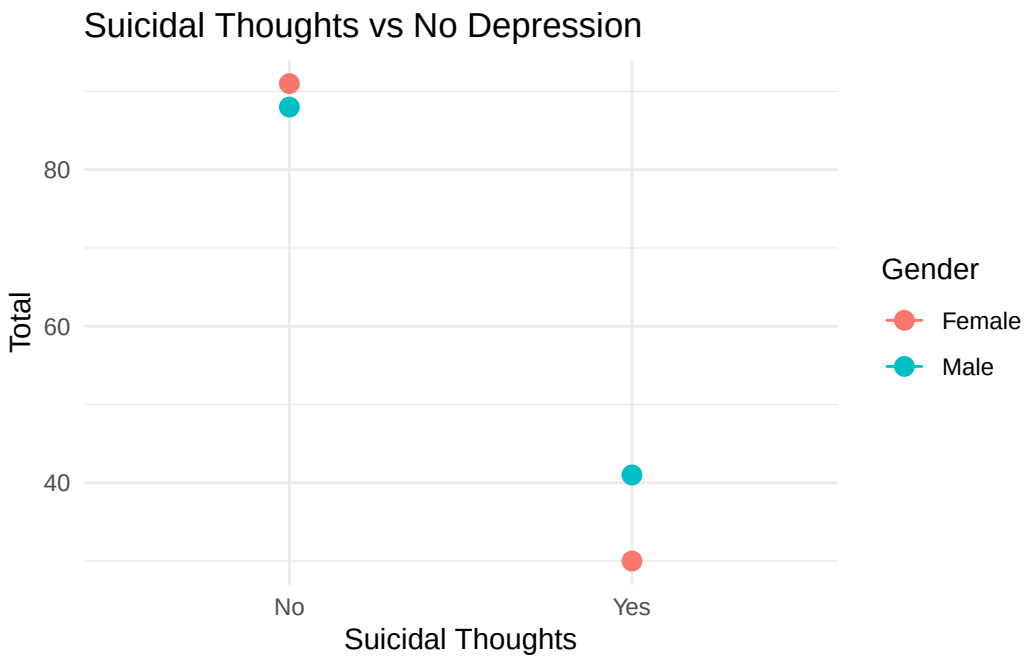
```
are_d <- depression_suicidal %>% filter(Depression == "Yes")
are_nd <- depression_suicidal %>% filter(Depression == "No")

ggplot(are_d, aes(x = Have.you.ever.had.suicidal.thoughts.., y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Suicidal Thoughts vs Depression",
    x = "Suicidal Thoughts",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```




```
ggplot(are_nd, aes(x = Have.you.ever.had.suicidal.thoughts.., y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Suicidal Thoughts vs No Depression",
    x = "Suicidal Thoughts",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```

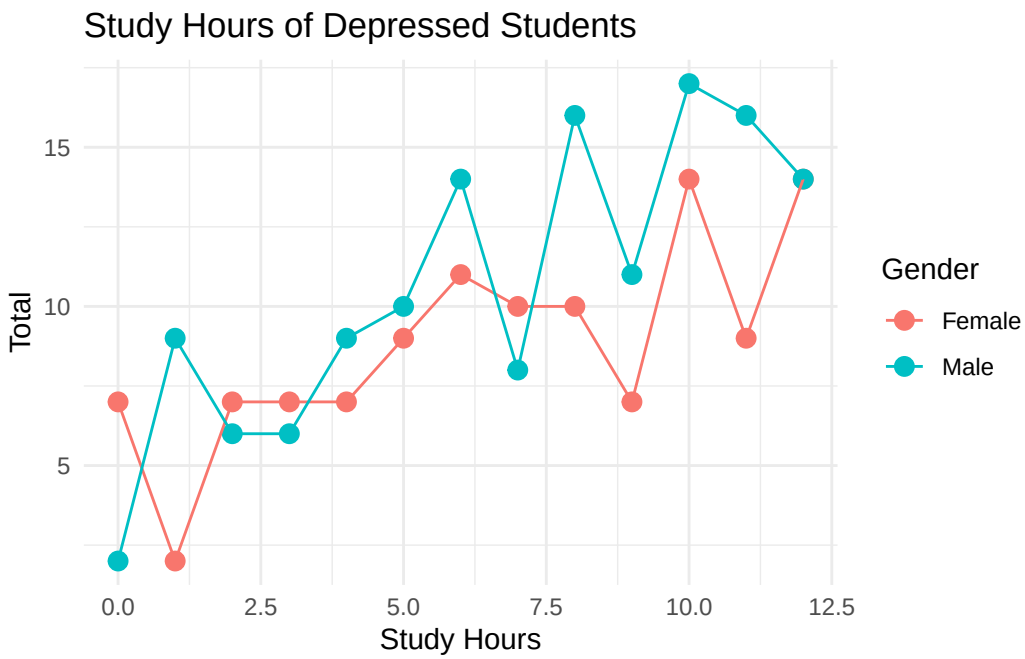
```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```



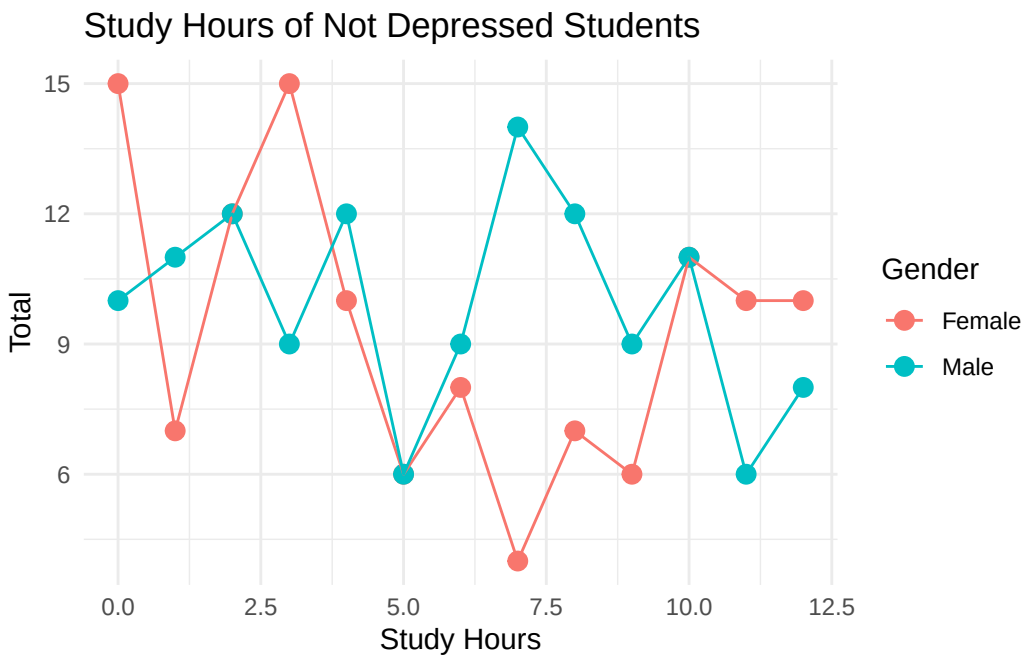

```
depression_study_h <- depression %>% group_by(Gender, Study.Hours, Depression) %>% summarise
```

```
`summarise()` has grouped output by 'Gender', 'Study.Hours'. You can override
using the `.groups` argument.
```

```
are_d <- depression_study_h %>% filter(Depression == "Yes")
are_nd <- depression_study_h %>% filter(Depression == "No")

ggplot(are_d, aes(x = Study.Hours, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Study Hours of Depressed Students",
    x = "Study Hours",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```



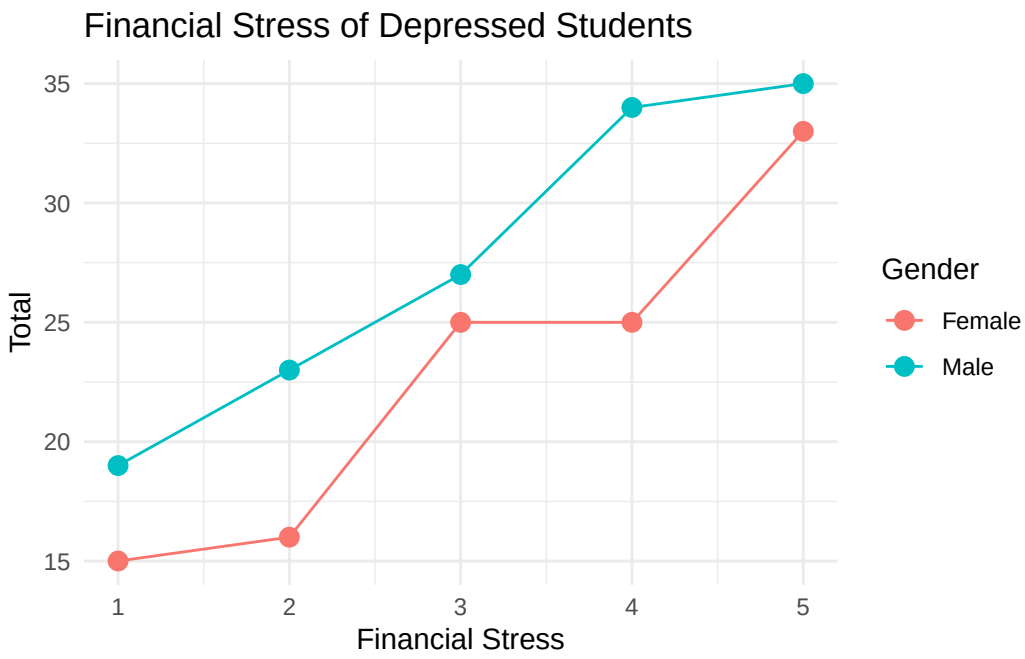

```
ggplot(are_nd, aes(x = Study.Hours, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Study Hours of Not Depressed Students",
    x = "Study Hours",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```



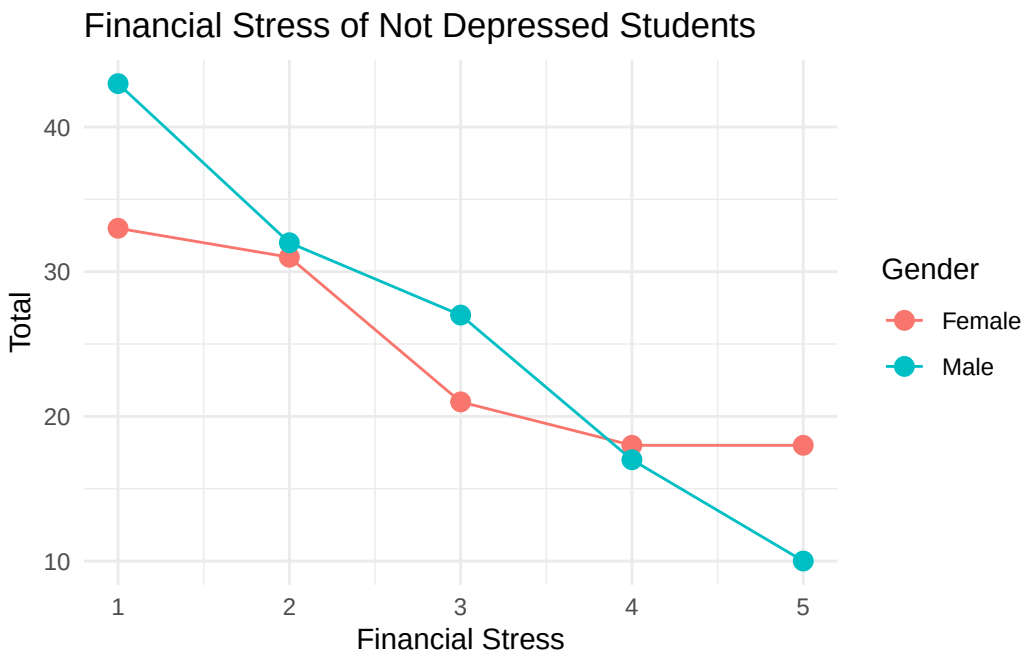

```
depression_finance <- depression %>% group_by(Gender, Financial.Stress, Depression) %>% summa
```

```
`summarise()` has grouped output by 'Gender', 'Financial.Stress'. You can
override using the `.groups` argument.
```

```
are_d <- depression_finance %>% filter(Depression == "Yes")
are_nd <- depression_finance %>% filter(Depression == "No")

ggplot(are_d, aes(x = Financial.Stress, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Financial Stress of Depressed Students",
    x = "Financial Stress",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```




```
ggplot(are_nd, aes(x = Financial.Stress, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Financial Stress of Not Depressed Students",
    x = "Financial Stress",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```

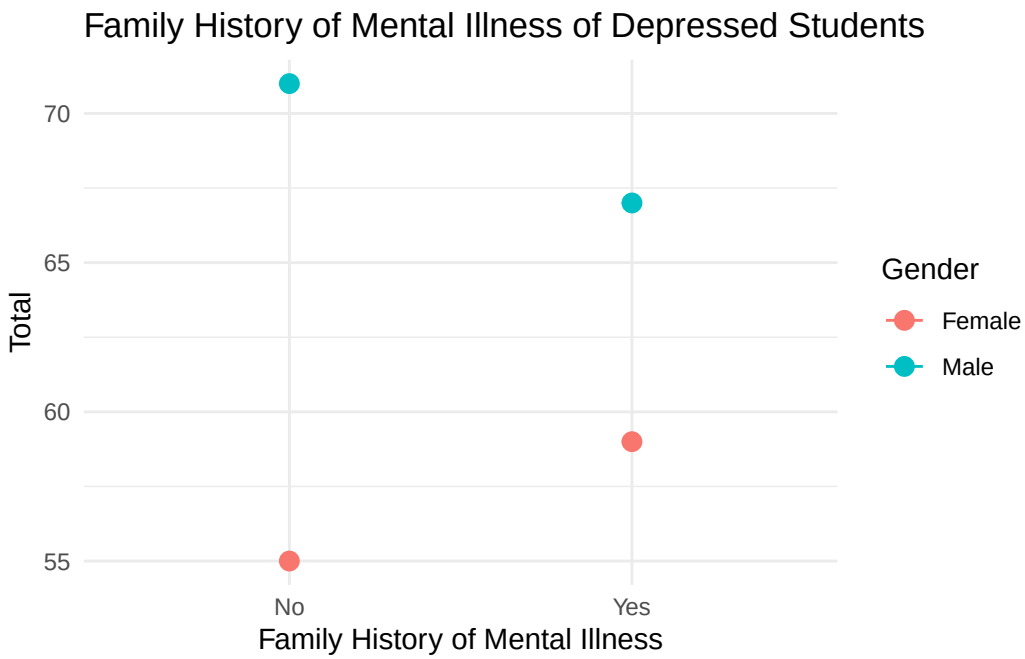



```
depression_fam <- depression %>% group_by(Gender, Family.History.of.Mental.Illness, Depressio
```

```
`summarise()` has grouped output by 'Gender',
'Family.History.of.Mental.Illness'. You can override using the `.groups`
argument.
```

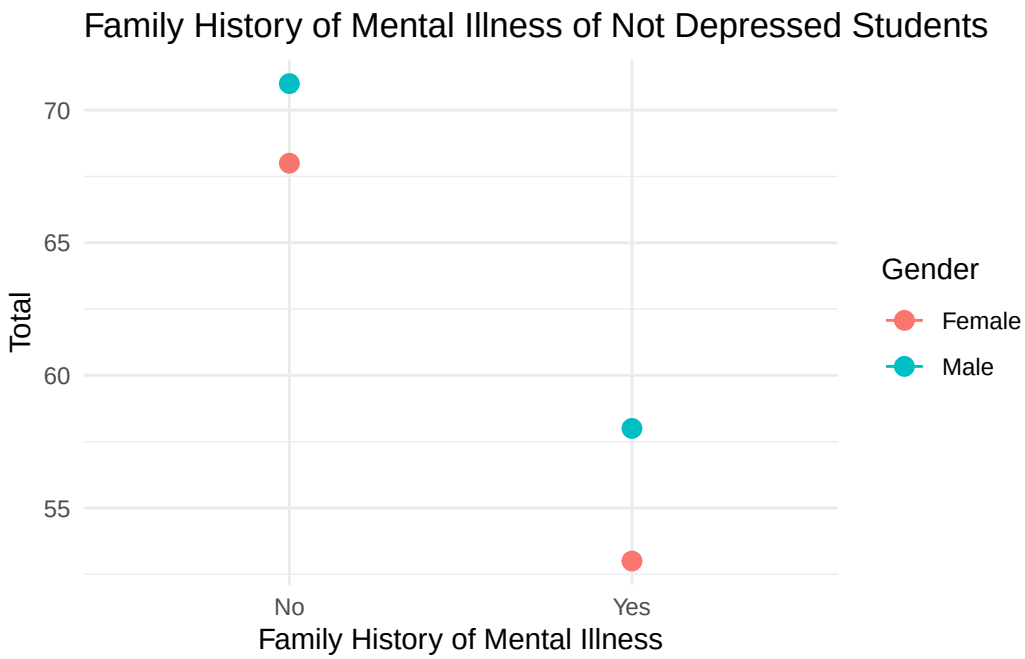
```
are_d <- depression_fam %>% filter(Depression == "Yes")
are_nd <- depression_fam %>% filter(Depression == "No")

ggplot(are_d, aes(x = Family.History.of.Mental.Illness, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Family History of Mental Illness of Depressed Students",
    x = "Family History of Mental Illness",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```




```
ggplot(are_nd, aes(x = Family.History.of.Mental.Illness, y = count, color = Gender)) +
  geom_point(size = 3) +
  geom_line() +
  labs(
    title = "Family History of Mental Illness of Not Depressed Students",
    x = "Family History of Mental Illness",
    y = "Total",
    color = "Gender"
  ) +
  theme_minimal()
```

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```



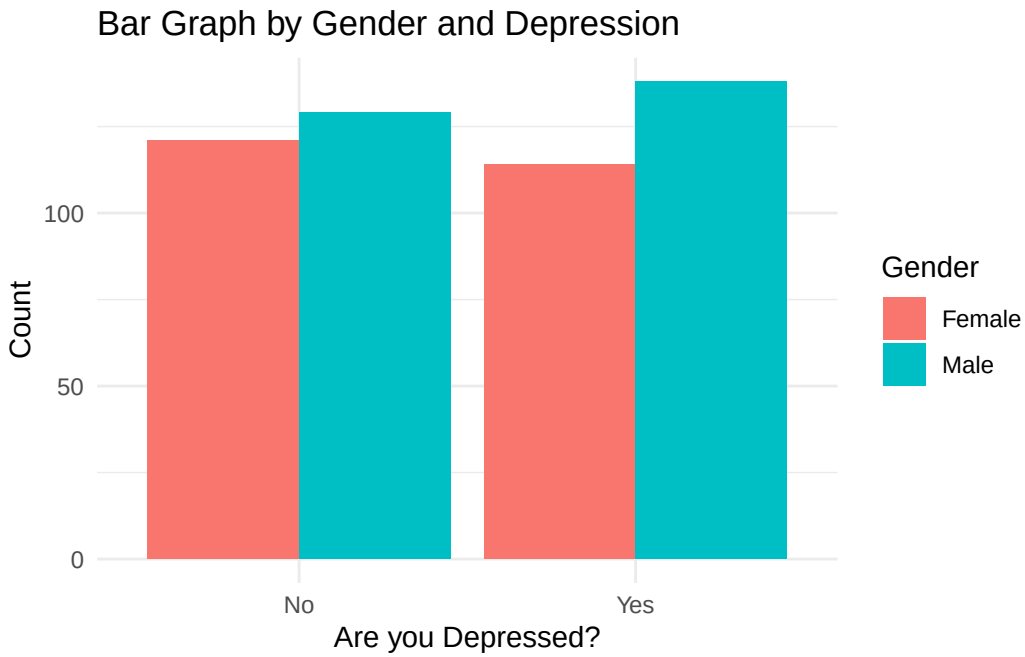

```
depression_is_depressed <- depression %>% group_by(Gender, Depression) %>% summarise(count =
```

```
`summarise()` has grouped output by 'Gender'. You can override using the
`.groups` argument.
```

```
depression_is_depressed
```

```
# A tibble: 4 x 3
# Groups:   Gender [2]
  Gender Depression count
  <chr>  <chr>      <int>
1 Female No           121
2 Female Yes          114
3 Male   No           129
4 Male   Yes          138
```

```
ggplot(depression_is_depressed, aes(x = Depression, y = count, fill = Gender)) +
  geom_bar(stat = "Identity", position = "dodge") + labs(title = "Bar Graph by Gender and Dep
       x = "Are you Depressed?",
       y = "Count") +
  theme_minimal()
```



The `echo: false` option disables the printing of code (only output is displayed).