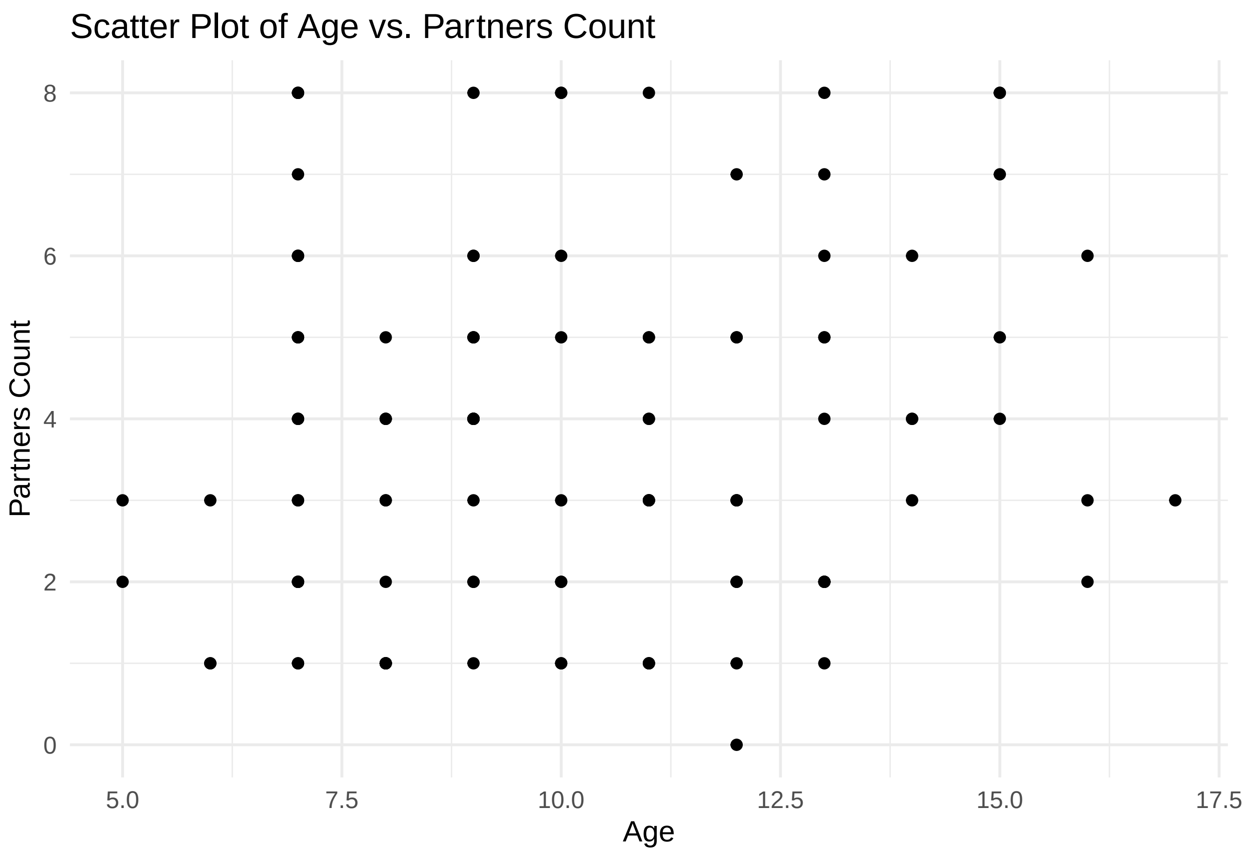
MA675GroupC_Brooke

2024-10-11

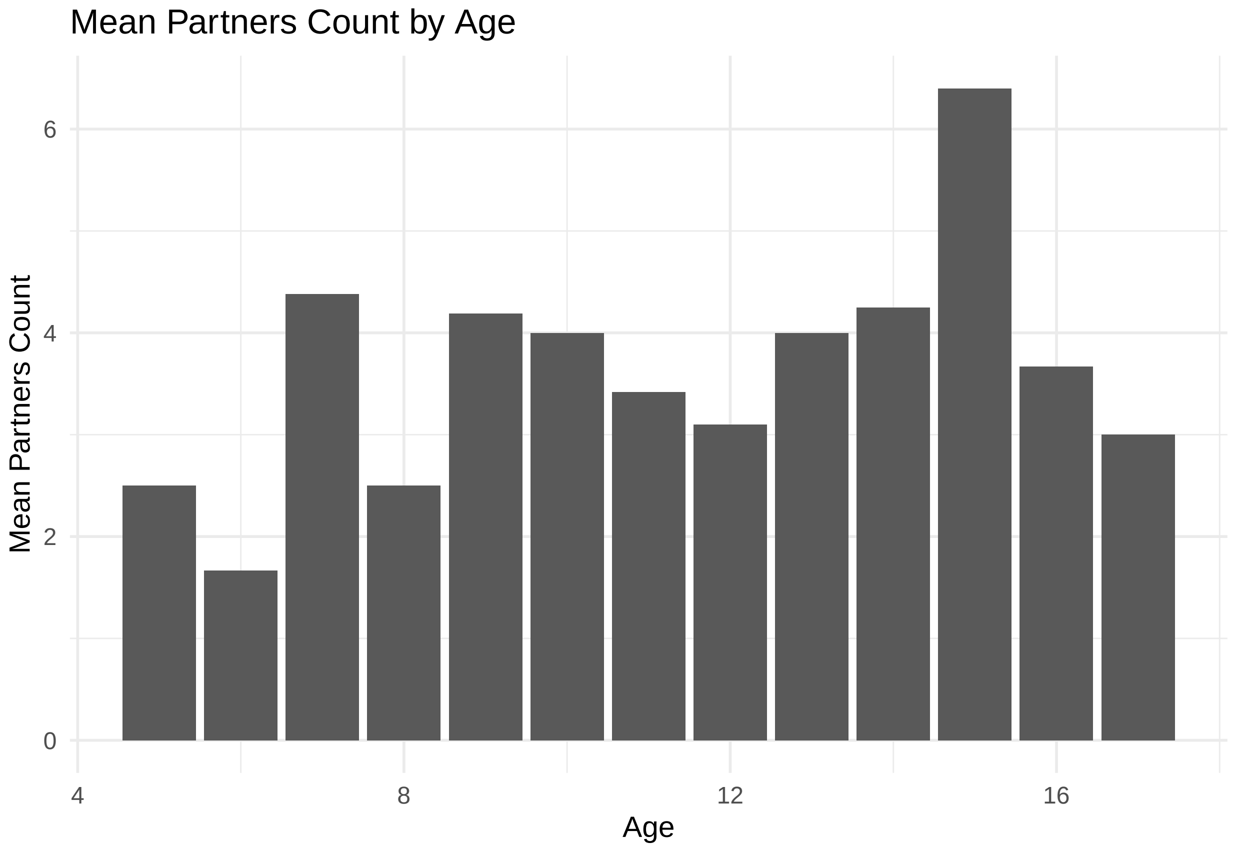
```
# Load necessary libraries
library(tidyverse)
```

```
## -- 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 become errors
```

```
# Load the dataset
data <- read.csv("Individual-level basic variable.csv")

# Ensure Age and Partners_count are numeric
data$Age <- as.numeric(data$Age)
data$Partners_count <- as.numeric(data$Partners_count)


# Create a scatter plot
ggplot(data, aes(x = Age, y = Partners_count)) +
  geom_point() +
  labs(title = "Scatter Plot of Age vs. Partners Count",
       x = "Age",
       y = "Partners Count") +
  theme_minimal()
```



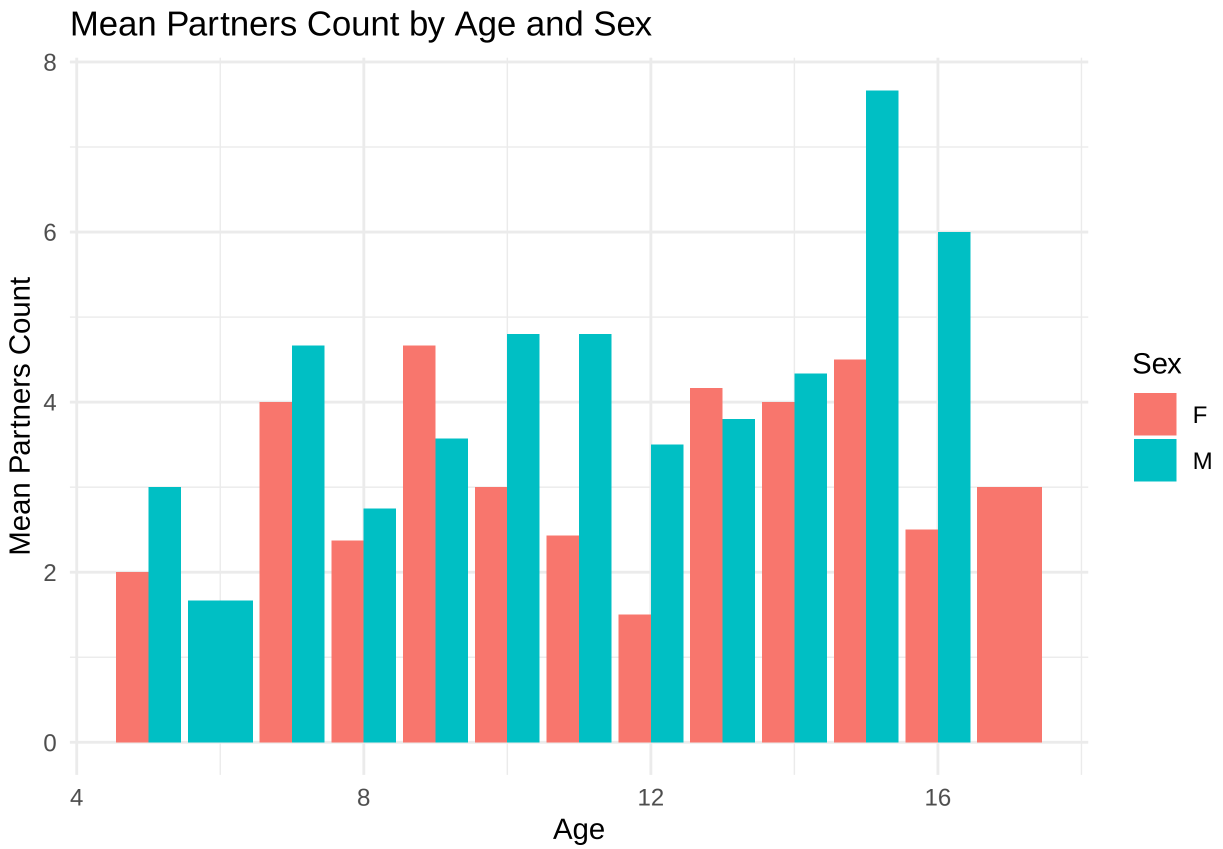
```r
# Load necessary libraries
library(tidyverse)

# Load the dataset
data <- read.csv("Individual-level basic variable.csv")

# Ensure Age and Partners_count are numeric
data$Age <- as.numeric(data$Age)
data$Partners_count <- as.numeric(data$Partners_count)

# Group by Age and calculate the mean of Partners_count
age_partner_mean <- data %>%
  group_by(Age) %>%
  summarise(mean_partners = mean(Partners_count, na.rm = TRUE))

# Create a bar plot
ggplot(age_partner_mean, aes(x = Age, y = mean_partners)) +
  geom_bar(stat = "identity") +
  labs(title = "Mean Partners Count by Age",
       x = "Age",
       y = "Mean Partners Count") +
  theme_minimal()
```



```
# Load necessary libraries
library(tidyverse)

# Load the dataset
data <- read.csv("Individual-level basic variable.csv")

# Ensure Age, Partners_count, and Sex are in the correct format
data$Age <- as.numeric(data$Age)
data$Partners_count <- as.numeric(data$Partners_count)
data$Sex <- as.factor(data$Sex)  # Make sure Sex is a factor (M/F)

# Group by Age and Sex, then calculate the mean of Partners_count
age_sex_partner_mean <- data %>%
  group_by(Age, Sex) %>%
  summarise(mean_partners = mean(Partners_count, na.rm = TRUE))
```

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

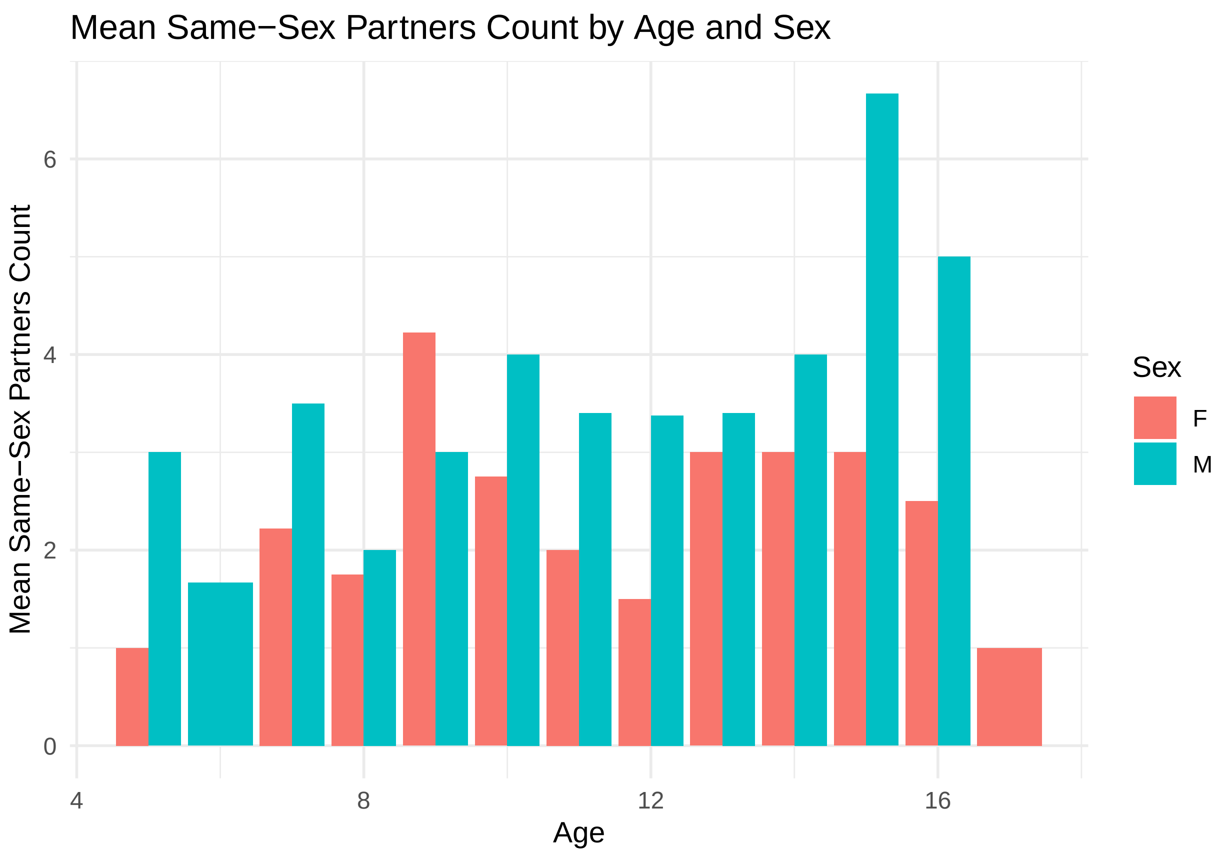
```
# Create a bar plot with divided bars for Males and Females
ggplot(age_sex_partner_mean, aes(x = Age, y = mean_partners, fill = Sex)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Mean Partners Count by Age and Sex",
       x = "Age",
       y = "Mean Partners Count",
       fill = "Sex") +
  theme_minimal()
```



```
# Load necessary libraries
library(tidyverse)

# Load the dataset
data <- read.csv("Individual-level basic variable.csv")

# Ensure Age, Count_same_sex, and Sex are in the correct format
data$Age <- as.numeric(data$Age)
data$Count_same_sex <- as.numeric(data$Count_same_sex)
data$Sex <- as.factor(data$Sex)  # Make sure Sex is a factor (M/F)

# Group by Age and Sex, then calculate the mean of Count_same_sex
age_sex_same_sex_mean <- data %>%
  group_by(Age, Sex) %>%
  summarise(mean_same_sex = mean(Count_same_sex, na.rm = TRUE))
```

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

```
# Create a bar plot with divided bars for Males and Females
ggplot(age_sex_same_sex_mean, aes(x = Age, y = mean_same_sex, fill = Sex)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Mean Same-Sex Partners Count by Age and Sex",
       x = "Age",
       y = "Mean Same-Sex Partners Count",
       fill = "Sex") +
  theme_minimal()
```



```
# Group by Age and Sex, then calculate the mean of Count_same_sex
age_sex_same_sex_mean <- data %>%
  group_by(Age, Sex) %>%
  summarise(mean_same_sex = mean(Count_same_sex, na.rm = TRUE))
```

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

```
# Fit linear regression models for each sex
model_male <- lm(mean_same_sex ~ Age, data = age_sex_same_sex_mean %>% filter(Sex == "M"))
model_female <- lm(mean_same_sex ~ Age, data = age_sex_same_sex_mean %>% filter(Sex == "F"))

# Print summaries for both models
summary(model_male)
```

```
##
## Call:
```

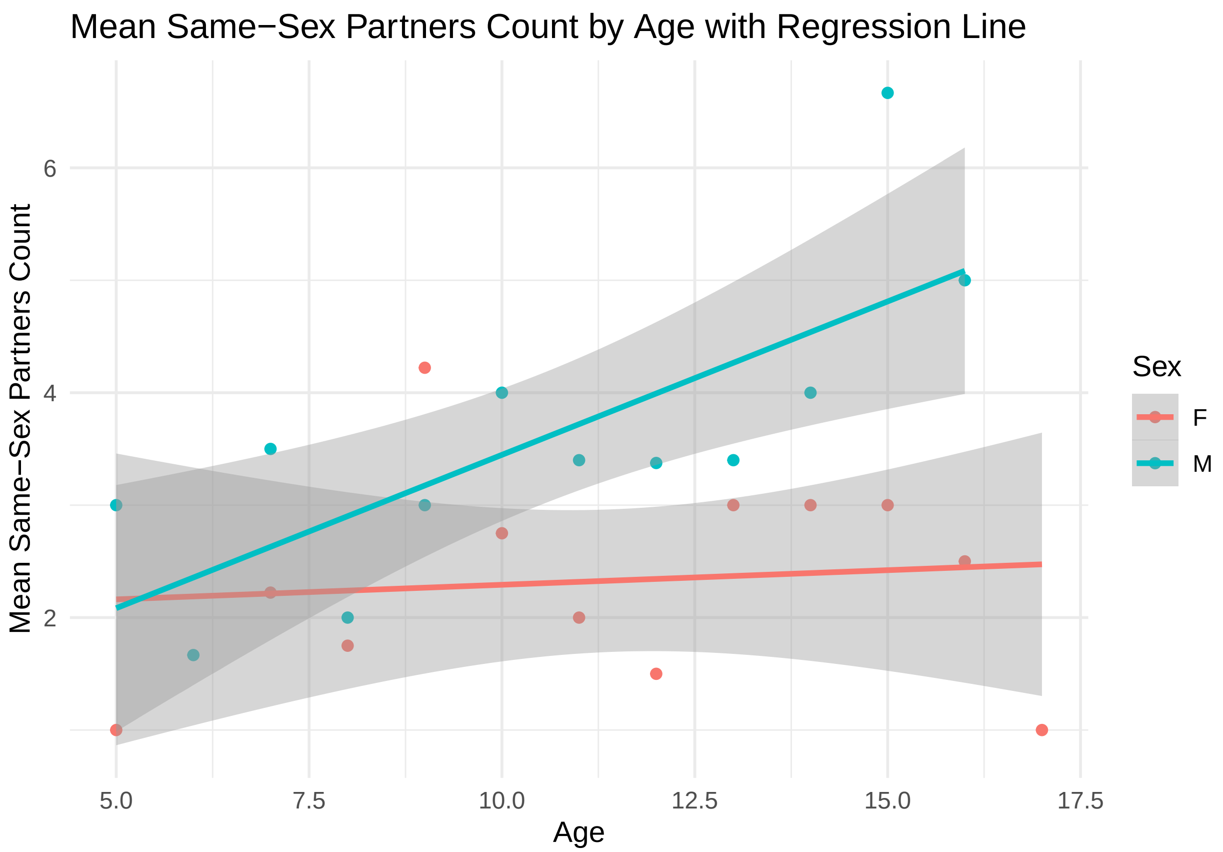
```
## lm(formula = mean_same_sex ~ Age, data = age_sex_same_sex_mean %>%
##     filter(Sex == "M"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.9020 -0.6361 -0.2476  0.6320  1.8550
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.71947    0.83681   0.860  0.41005
## Age          0.27281    0.07571   3.603  0.00482 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.9054 on 10 degrees of freedom
## Multiple R-squared:  0.5649, Adjusted R-squared:  0.5214
## F-statistic: 12.98 on 1 and 10 DF,  p-value: 0.00482
```

```
summary(model_female)
```

```
##
## Call:
## lm(formula = mean_same_sex ~ Age, data = age_sex_same_sex_mean %>%
##     filter(Sex == "F"))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.47370 -0.57844  0.03024  0.58473  1.95628
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.03222    0.94764   2.144   0.0576 .
## Age          0.02597    0.07918   0.328   0.7497
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.9855 on 10 degrees of freedom
## Multiple R-squared:  0.01064,    Adjusted R-squared:  -0.08829
## F-statistic: 0.1076 on 1 and 10 DF,  p-value: 0.7497
```

```
# Scatter plot with regression line for both males and females
ggplot(age_sex_same_sex_mean, aes(x = Age, y = mean_same_sex, color = Sex)) +
  geom_point() +  # Scatter plot
  geom_smooth(method = "lm", se = TRUE) +  # Regression line with confidence interval
  labs(title = "Mean Same-Sex Partners Count by Age with Regression Line",
       x = "Age",
       y = "Mean Same-Sex Partners Count",
       color = "Sex") +
  theme_minimal()
```

```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```

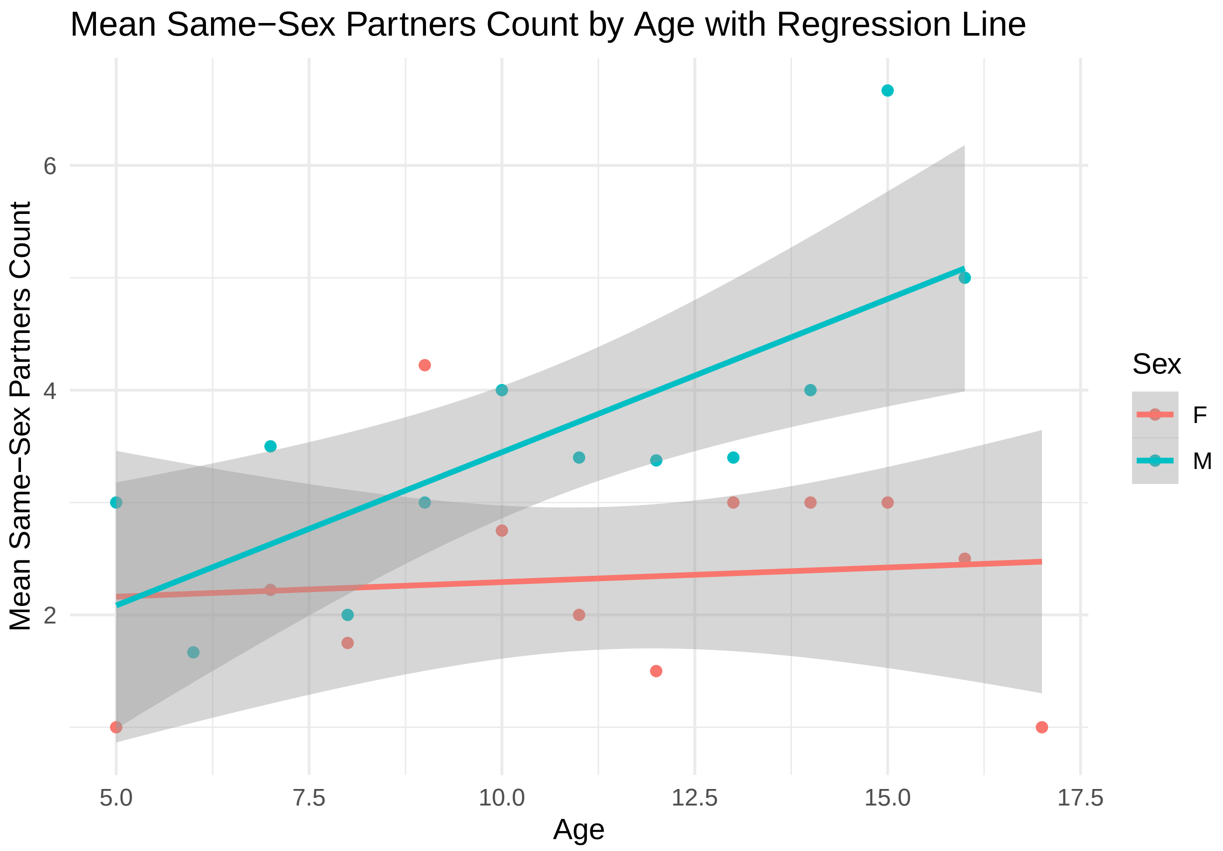


```
# Fit a combined model with interaction between Age and Sex
combined_model <- lm(mean_same_sex ~ Age * Sex, data = age_sex_same_sex_mean)

# Print the summary of the combined model
summary(combined_model)
```

```
##
## Call:
## lm(formula = mean_same_sex ~ Age * Sex, data = age_sex_same_sex_mean)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4737 -0.6361 -0.1297  0.5847  1.9563
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.03222    0.90993   2.233   0.0371 *
## Age          0.02597    0.07603   0.342   0.7362
## SexM        -1.31274    1.26212  -1.040   0.3107
## Age:SexM     0.24684    0.10973   2.249   0.0359 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9463 on 20 degrees of freedom
## Multiple R-squared:  0.5301, Adjusted R-squared:  0.4596
## F-statistic: 7.521 on 3 and 20 DF,  p-value: 0.001469
```

```r
# Scatter plot with regression line for both males and females
ggplot(age_sex_same_sex_mean, aes(x = Age, y = mean_same_sex, color = Sex)) +
  geom_point() +  # Scatter plot
  geom_smooth(method = "lm", se = TRUE) +  # Regression line with confidence interval
  labs(title = "Mean Same-Sex Partners Count by Age with Regression Line",
       x = "Age",
       y = "Mean Same-Sex Partners Count",
       color = "Sex") +
  theme_minimal()
```

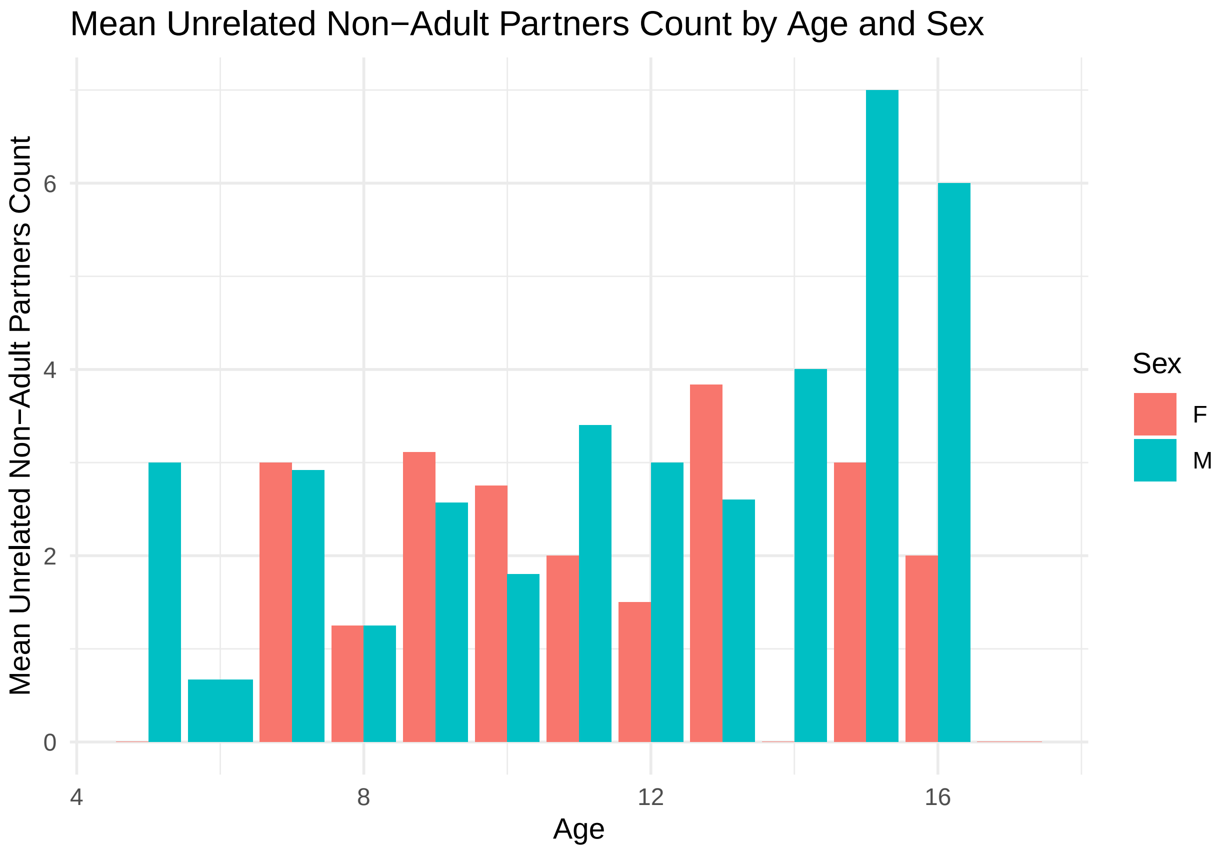
```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```



```r
# Load necessary libraries
library(tidyverse)

# Load the dataset
data <- read.csv("Individual-level basic variable.csv")

# Ensure Age, Count_unrelated_noadult, and Sex are in the correct format
data$Age <- as.numeric(data$Age)
data$Count_unrelated_noadult <- as.numeric(data$Count_unrelated_noadult)  # Assuming this variable repr
data$Sex <- as.factor(data$Sex)  # Ensure Sex is a factor (M/F)

# Group by Age and Sex, then calculate the mean of Count_unrelated_noadult
age_sex_unrelated_noadult_mean <- data %>%
  group_by(Age, Sex) %>%
  summarise(mean_unrelated_noadult = mean(Count_unrelated_noadult, na.rm = TRUE))
```

```
## ‘summarise()‘ has grouped output by ’Age’. You can override using the ‘.groups‘
## argument.
```

```
# Create a bar plot with divided bars for Males and Females
ggplot(age_sex_unrelated_noadult_mean, aes(x = Age, y = mean_unrelated_noadult, fill = Sex)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Mean Unrelated Non-Adult Partners Count by Age and Sex",
       x = "Age",
       y = "Mean Unrelated Non-Adult Partners Count",
       fill = "Sex") +
  theme_minimal()
```



```
# Group by Age and Sex, then calculate the mean of Count_unrelated_noadult
age_sex_unrelated_noadult_mean <- data %>%
  group_by(Age, Sex) %>%
  summarise(mean_unrelated_noadult = mean(Count_unrelated_noadult, na.rm = TRUE))
```

```
## ‘summarise()‘ has grouped output by ’Age’. You can override using the ‘.groups‘
## argument.
```

```r
# Fit separate linear regression models for Males and Females
model_male <- lm(mean_unrelated_noadult ~ Age, data = age_sex_unrelated_noadult_mean %>% filter(Sex ==
model_female <- lm(mean_unrelated_noadult ~ Age, data = age_sex_unrelated_noadult_mean %>% filter(Sex ==

# Print the summary of the models for Males and Females
cat("Summary for Male Model:\n")
```

```
## Summary for Male Model:
```

```r
summary(model_male)
```

```
##
## Call:
## lm(formula = mean_unrelated_noadult ~ Age, data = age_sex_unrelated_noadult_mean %>%
##     filter(Sex == "M"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.5210 -0.8717 -0.2729  0.8270  2.1293
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.7526     1.1675  -0.645  0.53366
## Age           0.3749     0.1056   3.549  0.00528 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 1.263 on 10 degrees of freedom
## Multiple R-squared:  0.5575, Adjusted R-squared:  0.5132
## F-statistic:  12.6 on 1 and 10 DF,  p-value: 0.005275
```

```r
cat("\nSummary for Female Model:\n")
```

```
##
## Summary for Female Model:
```

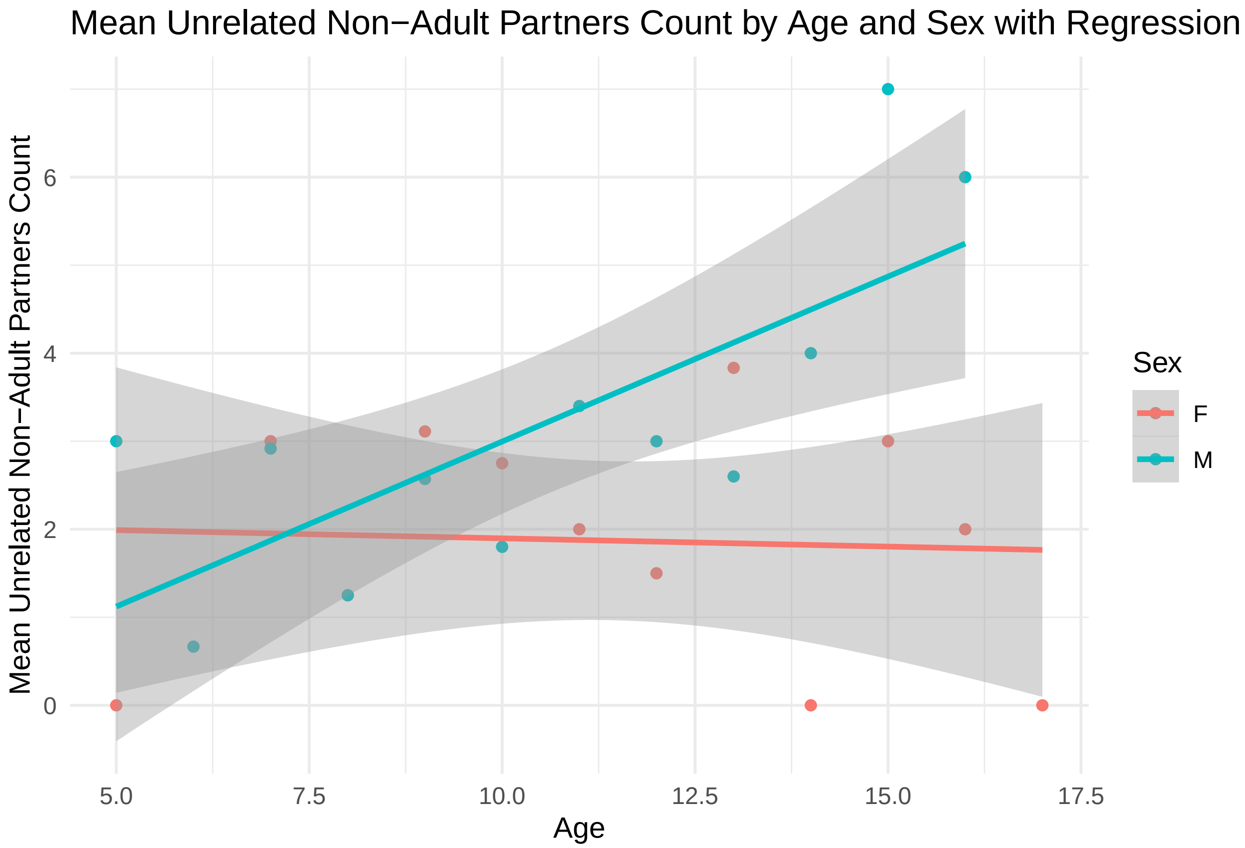
```r
summary(model_female)
```

```
##
## Call:
## lm(formula = mean_unrelated_noadult ~ Age, data = age_sex_unrelated_noadult_mean %>%
##     filter(Sex == "F"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9908 -0.9548  0.1687  1.0839  1.9927
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.08463    1.35018   1.544    0.154
## Age         -0.01877    0.11281  -0.166    0.871
```

```
##
## Residual standard error: 1.404 on 10 degrees of freedom
## Multiple R-squared:  0.00276,    Adjusted R-squared:  -0.09696
## F-statistic: 0.02768 on 1 and 10 DF,  p-value: 0.8712
```

```
# Scatter plot with regression line for both males and females
ggplot(age_sex_unrelated_noadult_mean, aes(x = Age, y = mean_unrelated_noadult, color = Sex)) +
  geom_point() +  # Scatter plot
  geom_smooth(method = "lm", se = TRUE) +  # Regression line with confidence interval
  labs(title = "Mean Unrelated Non-Adult Partners Count by Age and Sex with Regression",
       x = "Age",
       y = "Mean Unrelated Non-Adult Partners Count",
       color = "Sex") +
  theme_minimal()
```

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



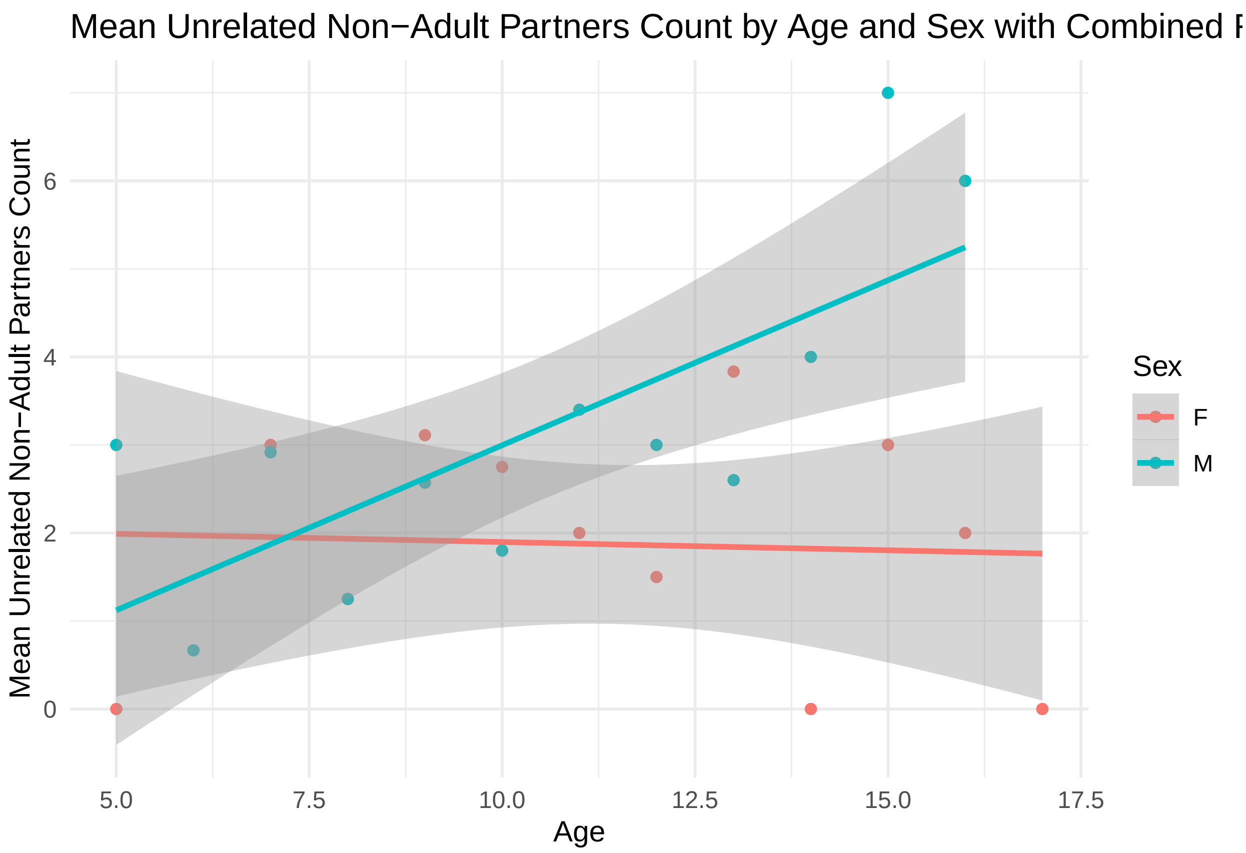
```
# Fit a combined model with interaction between Age and Sex
combined_model <- lm(mean_unrelated_noadult ~ Age * Sex, data = age_sex_unrelated_noadult_mean)

# Print the summary of the combined model
summary(combined_model)
```

```
##
```

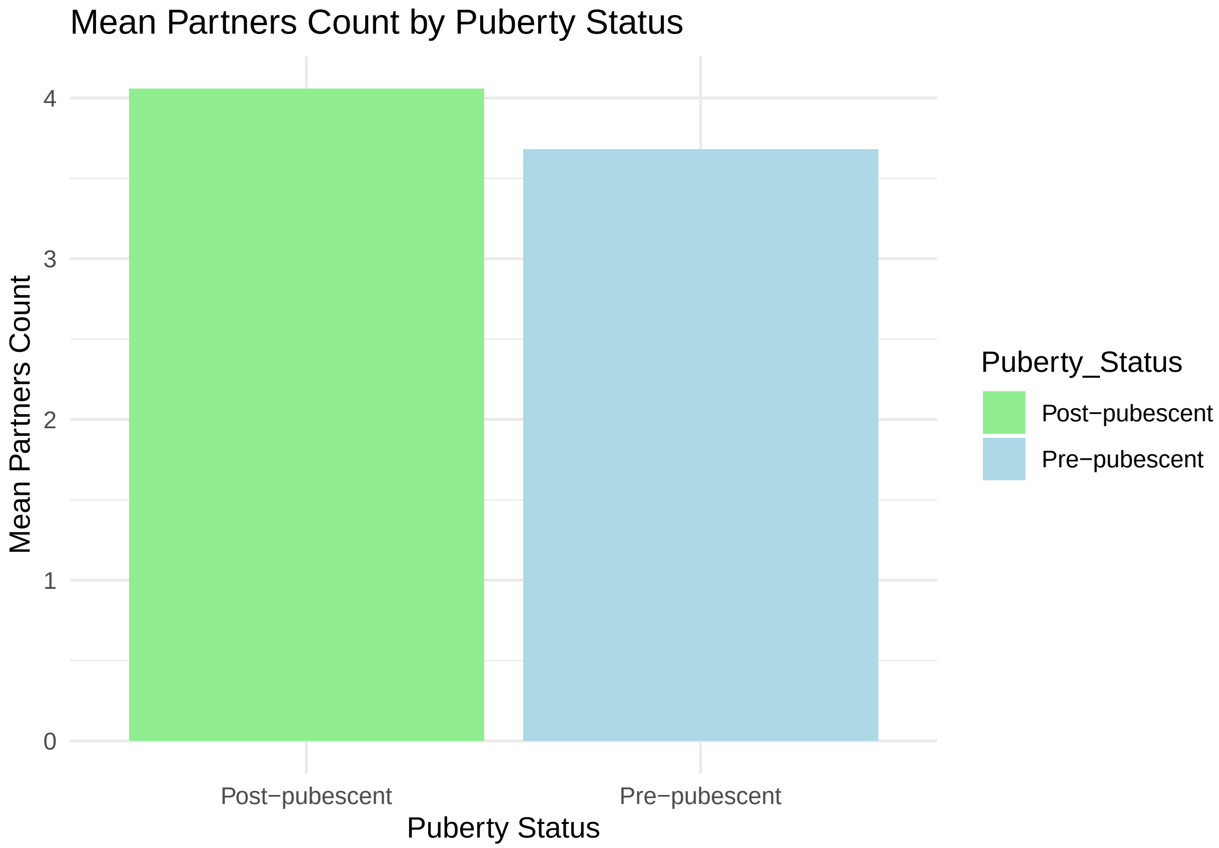
```
## Call:
## lm(formula = mean_unrelated_noadult ~ Age * Sex, data = age_sex_unrelated_noadult_mean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.99080 -0.87166 -0.01057  1.04548  2.12926
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.08463    1.28419   1.623   0.1202
## Age         -0.01877    0.10730  -0.175   0.8629
## SexM        -2.83727    1.78123  -1.593   0.1269
## Age:SexM     0.39366    0.15487   2.542   0.0194 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.335 on 20 degrees of freedom
## Multiple R-squared:  0.461,	Adjusted R-squared:  0.3801
## F-statistic: 5.701 on 3 and 20 DF,  p-value: 0.005468
```

```r
# Scatter plot with regression line for both males and females
ggplot(age_sex_unrelated_noadult_mean, aes(x = Age, y = mean_unrelated_noadult, color = Sex)) +
  geom_point() +  # Scatter plot
  geom_smooth(method = "lm", se = TRUE) +  # Regression line with confidence interval
  labs(title = "Mean Unrelated Non-Adult Partners Count by Age and Sex with Combined Regression",
       x = "Age",
       y = "Mean Unrelated Non-Adult Partners Count",
       color = "Sex") +
  theme_minimal()
```

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

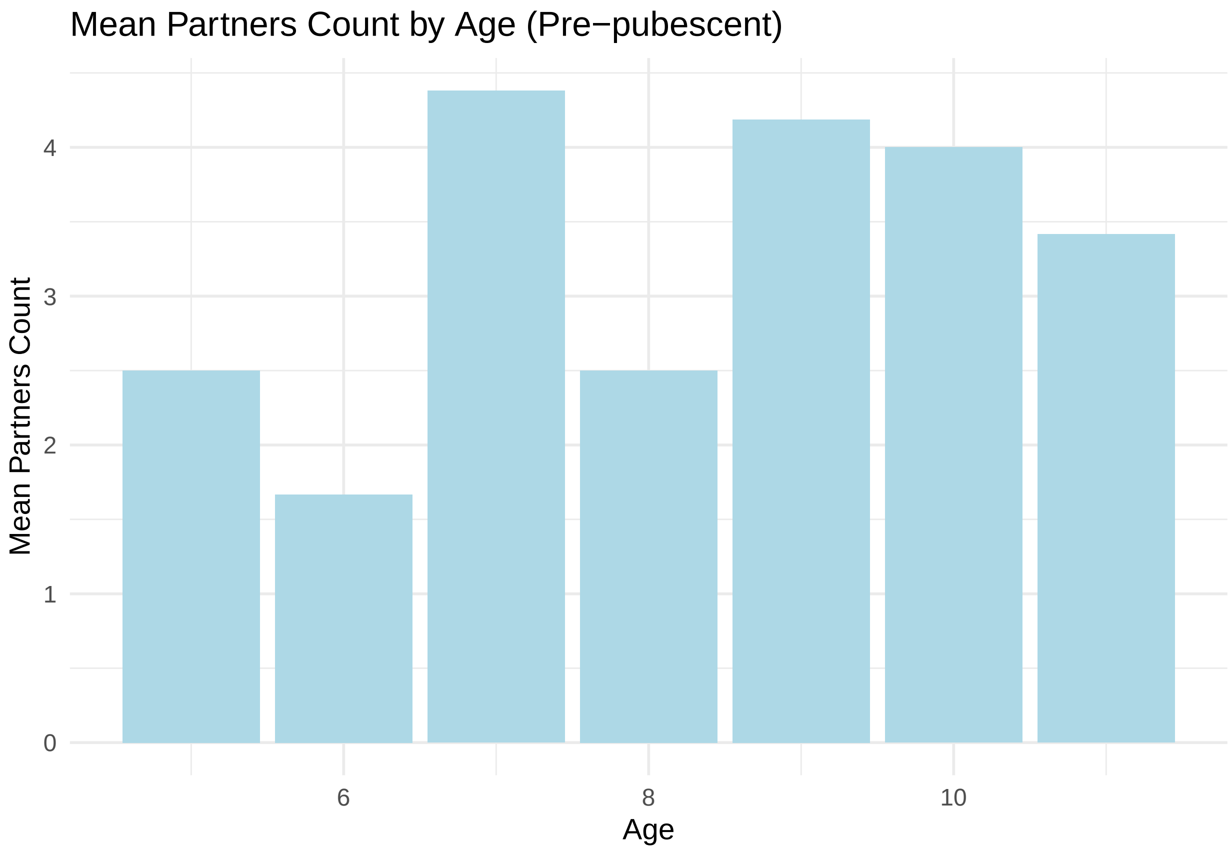


```r
# Load necessary libraries
library(tidyverse)

# Load the dataset
data <- read.csv("Individual-level basic variable.csv")

# Ensure Age and Partners_count are numeric
data$Age <- as.numeric(data$Age)
data$Partners_count <- as.numeric(data$Partners_count)

# Create a new variable for pre-pubescent and post-pubescent
data <- data %>%
  mutate(Puberty_Status = ifelse(Age < 12, "Pre-pubescent", "Post-pubescent"))

# Group by Puberty_Status and calculate the mean of Partners_count
puberty_partner_mean <- data %>%
  group_by(Puberty_Status) %>%
  summarise(mean_partners = mean(Partners_count, na.rm = TRUE))

# Create a bar plot
ggplot(puberty_partner_mean, aes(x = Puberty_Status, y = mean_partners, fill = Puberty_Status)) +
  geom_bar(stat = "identity") +
  labs(title = "Mean Partners Count by Puberty Status",
       x = "Puberty Status",
       y = "Mean Partners Count") +
  theme_minimal() +
  scale_fill_manual(values = c("Pre-pubescent" = "lightblue", "Post-pubescent" = "lightgreen"))
```

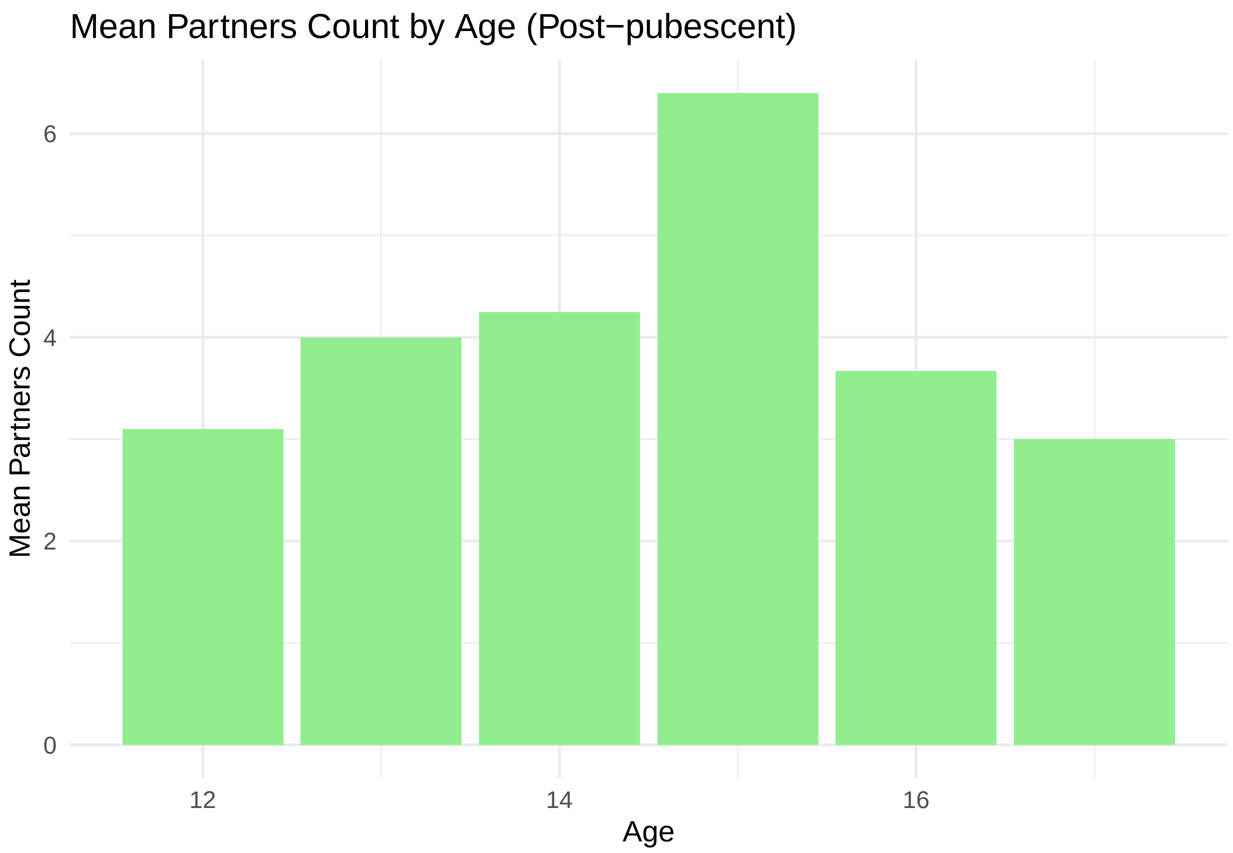


```
pre_pubescent_data <- data %>% filter(Puberty_Status == "Pre-pubescent")
post_pubescent_data <- data %>% filter(Puberty_Status == "Post-pubescent")

# Group by Age and calculate the mean of Partners_count for both groups
pre_pubescent_mean <- pre_pubescent_data %>%
  group_by(Age) %>%
  summarise(mean_partners = mean(Partners_count, na.rm = TRUE))

post_pubescent_mean <- post_pubescent_data %>%
  group_by(Age) %>%
  summarise(mean_partners = mean(Partners_count, na.rm = TRUE))

# Create the bar plots for both groups

# Pre-pubescent plot
pre_plot <- ggplot(pre_pubescent_mean, aes(x = Age, y = mean_partners)) +
  geom_bar(stat = "identity", fill = "lightblue") +
  labs(title = "Mean Partners Count by Age (Pre-pubescent)",
       x = "Age",
       y = "Mean Partners Count") +
  theme_minimal()

# Post-pubescent plot
```

```r
post_plot <- ggplot(post_pubescent_mean, aes(x = Age, y = mean_partners)) +
  geom_bar(stat = "identity", fill = "lightgreen") +
  labs(title = "Mean Partners Count by Age (Post-pubescent)",
       x = "Age",
       y = "Mean Partners Count") +
  theme_minimal()

# Print the plots
print(pre_plot)
```



```r
print(post_plot)
```

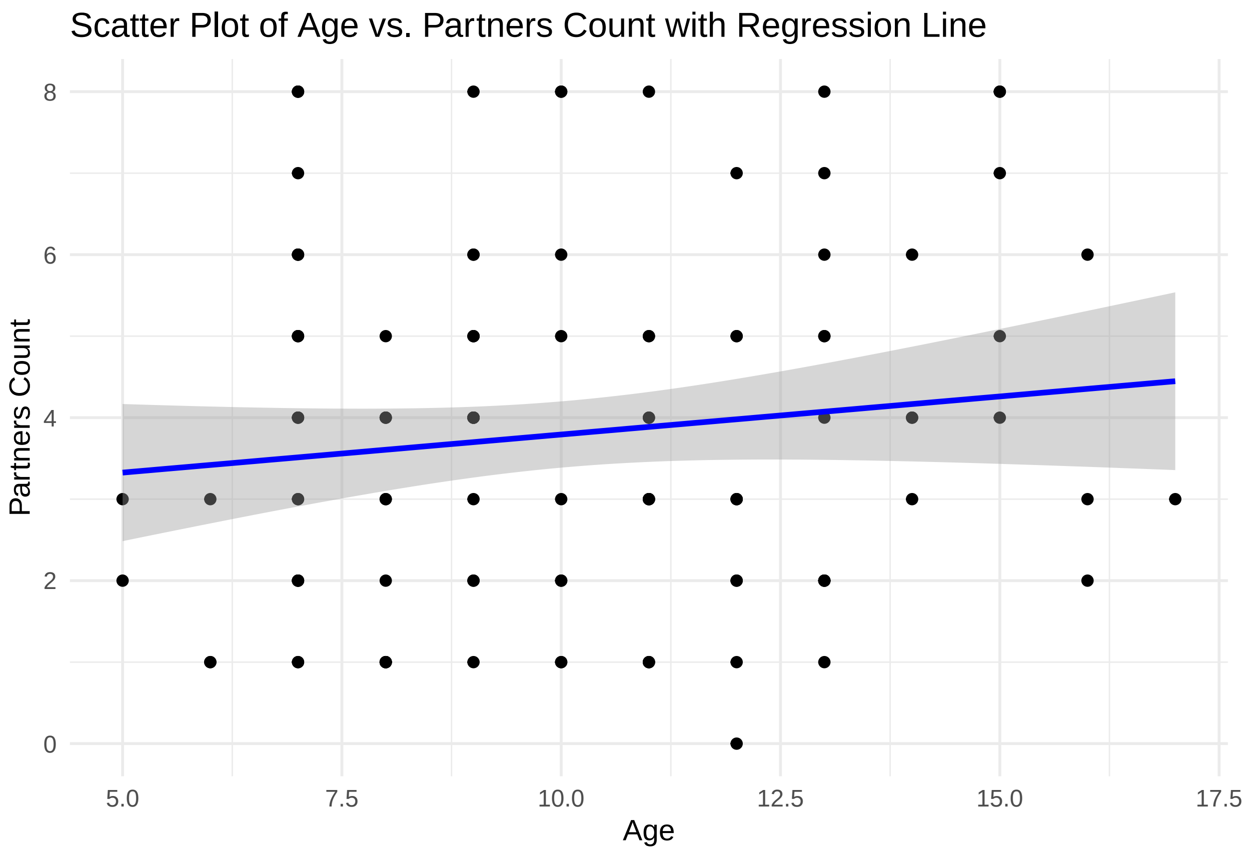


```
model <- lm(Partners_count ~ Age, data = data)

# Print the summary of the regression model
summary(model)
```

```
##
## Call:
## lm(formula = Partners_count ~ Age, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9799 -1.6996 -0.1668  1.3004  4.4873
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.85857    0.76708   3.727 0.000312 ***
## Age          0.09345    0.07351   1.271 0.206432
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.14 on 107 degrees of freedom
## Multiple R-squared:  0.01488,    Adjusted R-squared:  0.00567
## F-statistic: 1.616 on 1 and 107 DF,  p-value: 0.2064
```

```
# Create a scatter plot with regression line
ggplot(data, aes(x = Age, y = Partners_count)) +
  geom_point() +  # Scatter plot
  geom_smooth(method = "lm", se = TRUE, color = "blue") +  # Regression line with confidence interval
  labs(title = "Scatter Plot of Age vs. Partners Count with Regression Line",
       x = "Age",
       y = "Partners Count") +
  theme_minimal()
```

```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```


Scatter Plot of Age vs. Partners Count with Regression Line
Partners Count
Age