Data211-Midterm Exam

Peyton Hall

02/29/2024

Required libraries

```
library(ggplot2)
library(readxl)
```

Question 01

```
# Open R Markdown
```

Question 02

```
# Create R data structure:

# a) Create and print a numeric vector
x <- c(1, 3, 5, 12, 20)
x
```

```
## [1]  1  3  5 12 20
```

```
# b) Create and print a character vector
y <- c("red", "yellow", "blue")
y
```

```
## [1] "red"    "yellow" "blue"
```

```
# c) Create and print a list, z.
z <- list(a = x, b = y)
z
```

```
## $a
## [1]  1  3  5 12 20
##
## $b
## [1] "red"    "yellow" "blue"
```

```r
# d) Convert the vector y in b) to a factor
y_factor <- factor(y)
y_factor
```

```
## [1] red    yellow blue
## Levels: blue red yellow
```

Question 03

```r
# Create a function named triangle_identifier with the following requirements:

# a) Have three inputs (arguments): x1, x2, and x3. The x1, x2, and x3 are the
#    lengths of the three sides of a triangle.
triangle_identifier <- function(x1, x2, x3) {
  # stub
}

# b) Include if-else statements in the function to show whether the x1, x2 and
#    x3 can create a triangle.
#    • If x1+x2<x3, then it is not a triangle.
#    • If x1+x3<x2, then it is not a triangle.
#    • If x2+x3<x1, then it is not a triangle.
#    • Otherwise, it is a triangle.
#    Return the result of whether it is a triangle.
triangle_identifier <- function(x1, x2, x3) {
  if (x1 + x2 < x3 || x1 + x3 < x2 || x2 + x3 < x1) {
    result <- "It is not a triangle"
  } else {
    result <- "It is a triangle"
  }
  return(result)
}

# c) Call the function using x1=3, x2=5, x3=6 and print the output; and call the
#    function using x1=2, x2=3, x3=7 and print the output.
output1 <- triangle_identifier(3, 5, 6)
output1
```

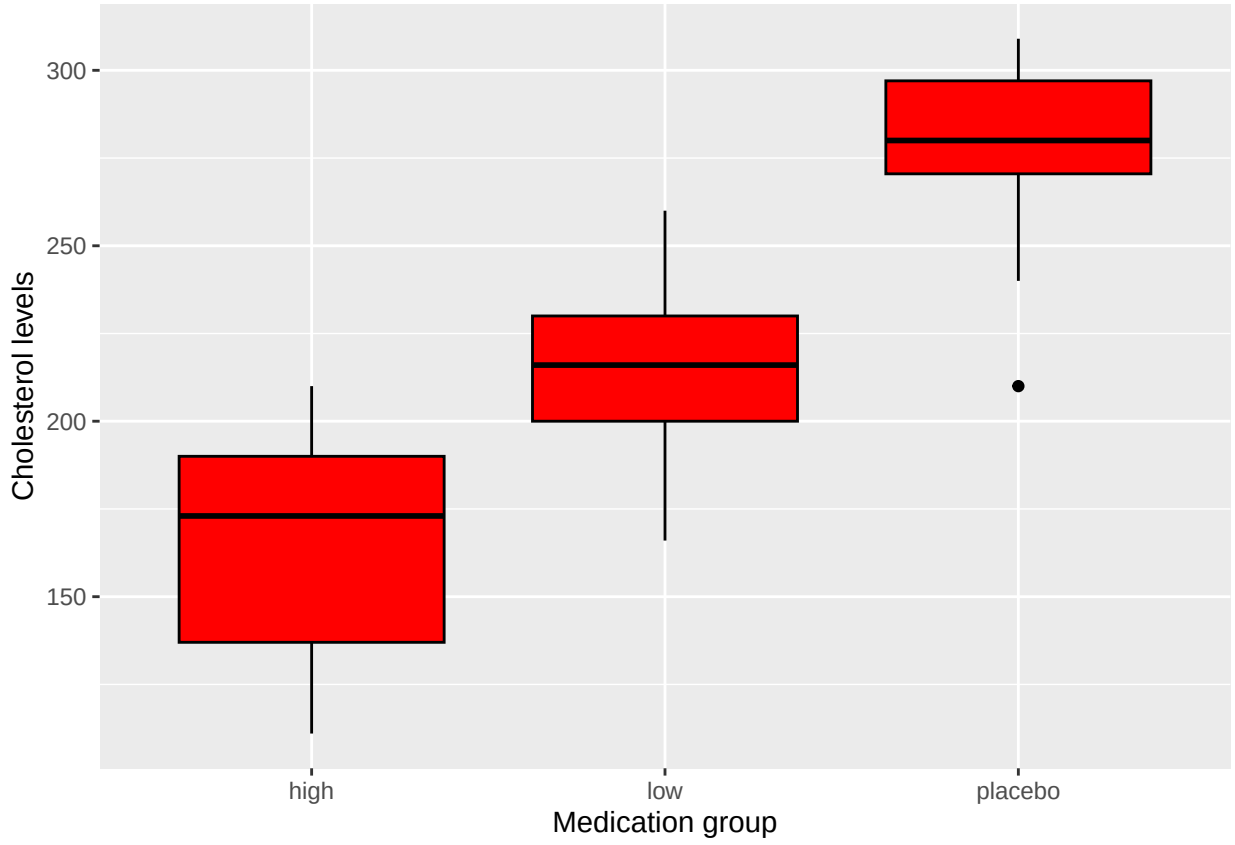
```
## [1] "It is a triangle"
```

```r
output2 <- triangle_identifier(2, 3, 7)
output2
```

```
## [1] "It is not a triangle"
```

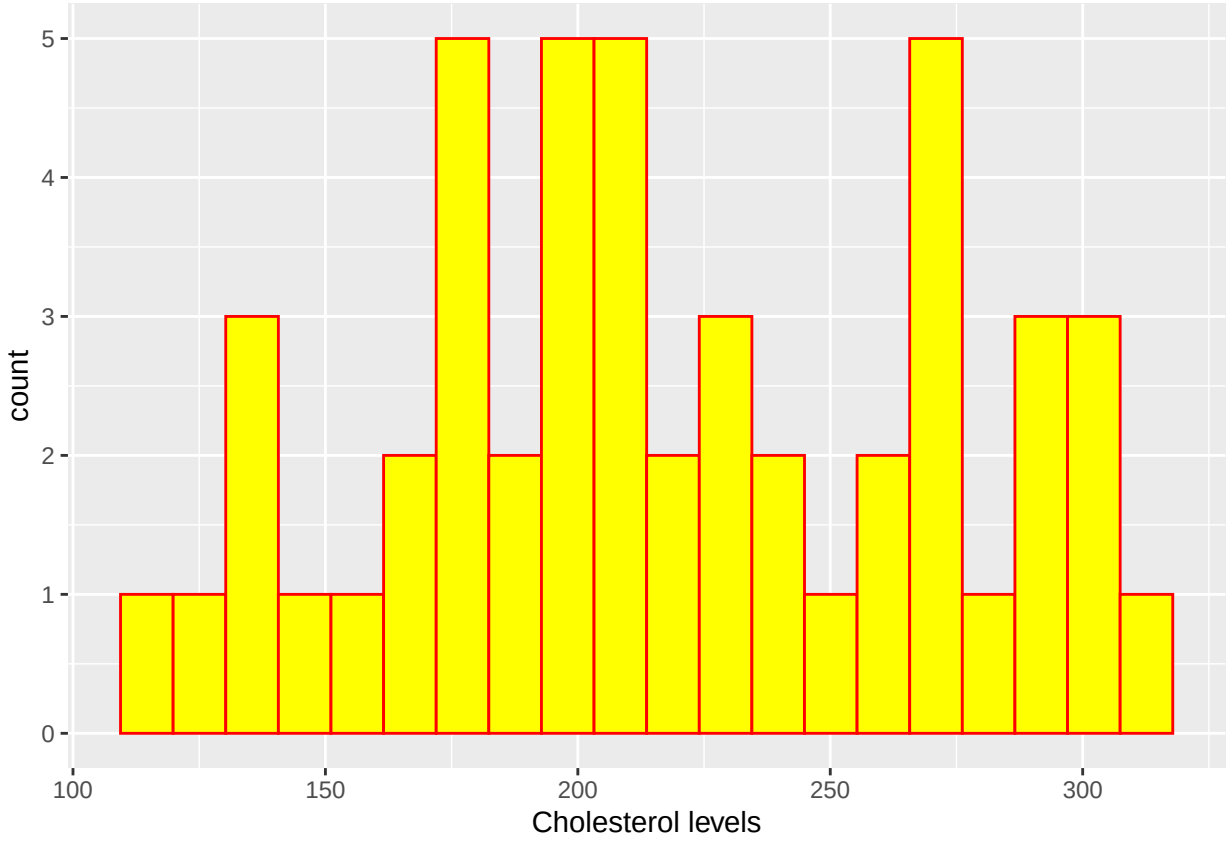
Question 04

```r
# The data Cholesterol on D2L recorded 49 high cholesterol patients' cholesterol
# levels after taking different medications. Among those 49 patients, some of
# them were assigned to take the placebo, some were assigned to take the
# low-dosage medication, and the rest took high dosages.

# press "Import Dataset" -> press "From Excel..." -> press "Browse..." ->
# navigate & select .xlsx file -> copy (command + c) code from "Code Preview:"
# -> press "Import" -> paste (command + v) code to the RStudio Integrated
# Development Environment.
Cholesterol <- read_excel("/Users/peytonhall/Desktop/Data211Midterm/Cholesterol.xlsx")
# View(Cholesterol)

# a) Use ggplot() to generate an appropriate graph to show the five-number
#    summary of cholesterol levels at each medication group. Label the x-axis as
#    "Medication group" and y as "Cholesterol levels."
#    Color the inside red and the borders black.
ggplot(Cholesterol, aes(x = Dosage, y = Chol)) +
  labs(x = "Medication group", y = "Cholesterol levels") + # labels
  geom_boxplot(fill = "red", color = "black") # red insides & black borders
```




```r
# b) Is there any outlier in a)?
# Answer:
#    I see one outlier in a. That is, one of the patients on placebo had a 210
#    cholesterol level. The 210 cholesterol level differs from the average
#    cholesterol level in patients on placebo. This outlier can be observed in
```

```r
#    the boxplot by the dot in the placebo column on the x axis.

# c) Use ggplot() to generate an appropriate graph to show the distribution of
#    all cholesterol levels regardless of medication groups, with 20 bins. Label
#    the x axis as "Cholesterol levels". Color the border with red and the
#    inside of the bars with yellow.
#    create a histogram because bars are required
ggplot(Cholesterol, aes(x = Chol)) +
  labs(x = "Cholesterol levels") + # label the x axis
  geom_histogram(fill = "yellow", color = "red", bins = 20) # red border
```



Question 05

```r
# Create the following table as a data frame
# and complete the following questions:

# a) Create the data frame with the name PatientInfo and print it
#    PatientID, Diastolic Blood Pressure (DBP), Systolic Blood Pressure (SBP)
#    A,         90,                             120
#    B,         98,                             135
#    C,         76,                             109
#    D,         112,                            141
PatientInfo <- data.frame(
```

```
  PatientID = c("A", "B", "C", "D"),
  Diastolic_Blood_Pressure_DBP = c(90, 98, 76, 112),
  Systolic_Blood_Pressure_SBP = c(120, 135, 109, 141)
)
PatientInfo
```

```
##   PatientID Diastolic_Blood_Pressure_DBP Systolic_Blood_Pressure_SBP
## 1         A                           90                          120
## 2         B                           98                          135
## 3         C                           76                          109
## 4         D                          112                          141
```

```
# b) Use rbind() to add one row of new individuals with the information:
#    PatientID: E
#    Diastolic Blood Pressure: 118
#    Systolic Blood Pressure: 129.
#    Name the new data frame with the new E as BloodPressure and print it.
new_individual <- data.frame(
  PatientID = "E",
  Diastolic_Blood_Pressure_DBP = 118,
  Systolic_Blood_Pressure_SBP = 129
)
BloodPressure <- rbind(PatientInfo, new_individual) # row bind adds the new row
BloodPressure # print the updated result
```

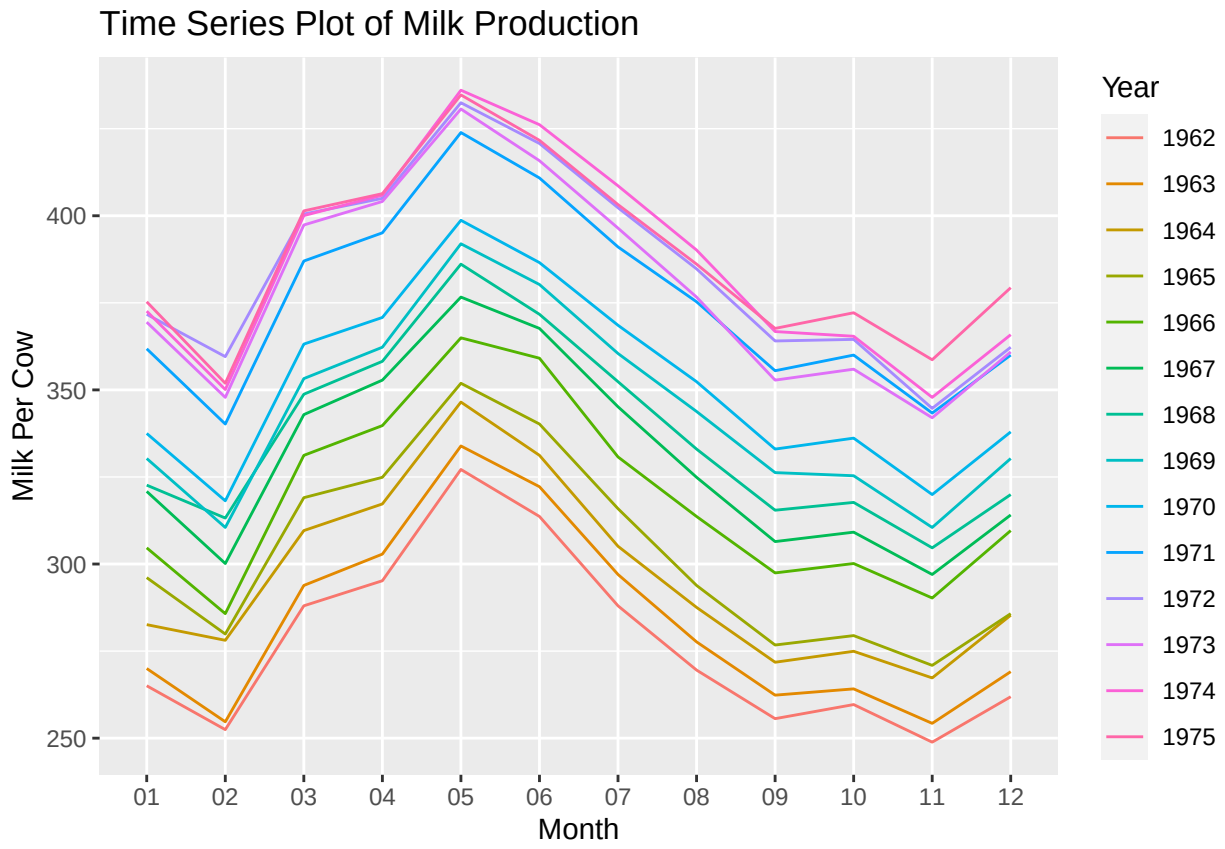
```
##   PatientID Diastolic_Blood_Pressure_DBP Systolic_Blood_Pressure_SBP
## 1         A                           90                          120
## 2         B                           98                          135
## 3         C                           76                          109
## 4         D                          112                          141
## 5         E                          118                          129
```

```
# c) Add a new column named Age to the above data frame (BloodPressure)
#    and name the new data frame as NewPatientInfo.
#    Following are the data in column Age: 28, 35,42, 40, 39.
Age <- c(28, 35, 42, 40, 39)
BloodPressure$Age <- Age # adds a new column
NewPatientInfo <- BloodPressure # assigns new data to a new data frame
NewPatientInfo # print the new data frame
```

```
##   PatientID Diastolic_Blood_Pressure_DBP Systolic_Blood_Pressure_SBP Age
## 1         A                           90                          120  28
## 2         B                           98                          135  35
## 3         C                           76                          109  42
## 4         D                          112                          141  40
## 5         E                          118                          129  39
```

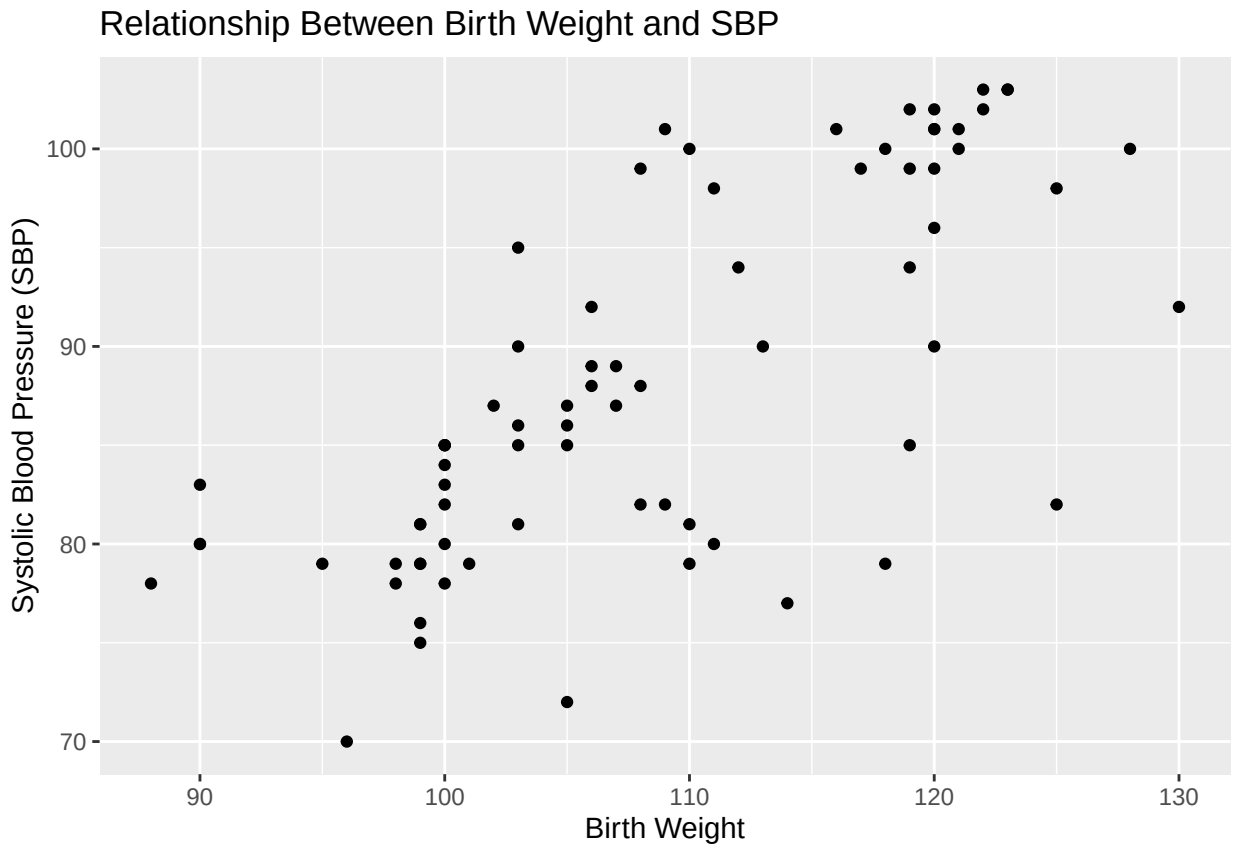
Question 06

```r
# Use the milk data to generate a time series plot.

# press "Import Dataset" -> press "From Excel..." -> press "Browse..." ->
# navigate & select .xlsx file -> copy (command + c) code from "Code Preview:"
# -> press "Import" -> paste (command + v) code to the RStudio Integrated
# Development Environment.
milk <- read_excel("/Users/peytonhall/Desktop/Data211Midterm/milk.xlsx")
# View(milk)

# a) Use the following to extract the month information and year information:
milk$year<-format( milk$timep , format="%Y")
milk$month<-format( milk$timep , format="%m")

# b) Generate a time series plot with the axis being a month and the
#    axis being milk per cow and use color to show different years.
ggplot(milk, aes(x = month, y = milk_per_cow_kg)) +
  geom_line(aes(group = year, color = year)) + # aes function to represent the grouping and the columns
  labs(x = "Month", y = "Milk Per Cow", color = "Year") +
  ggtitle("Time Series Plot of Milk Production")
```



Question 07

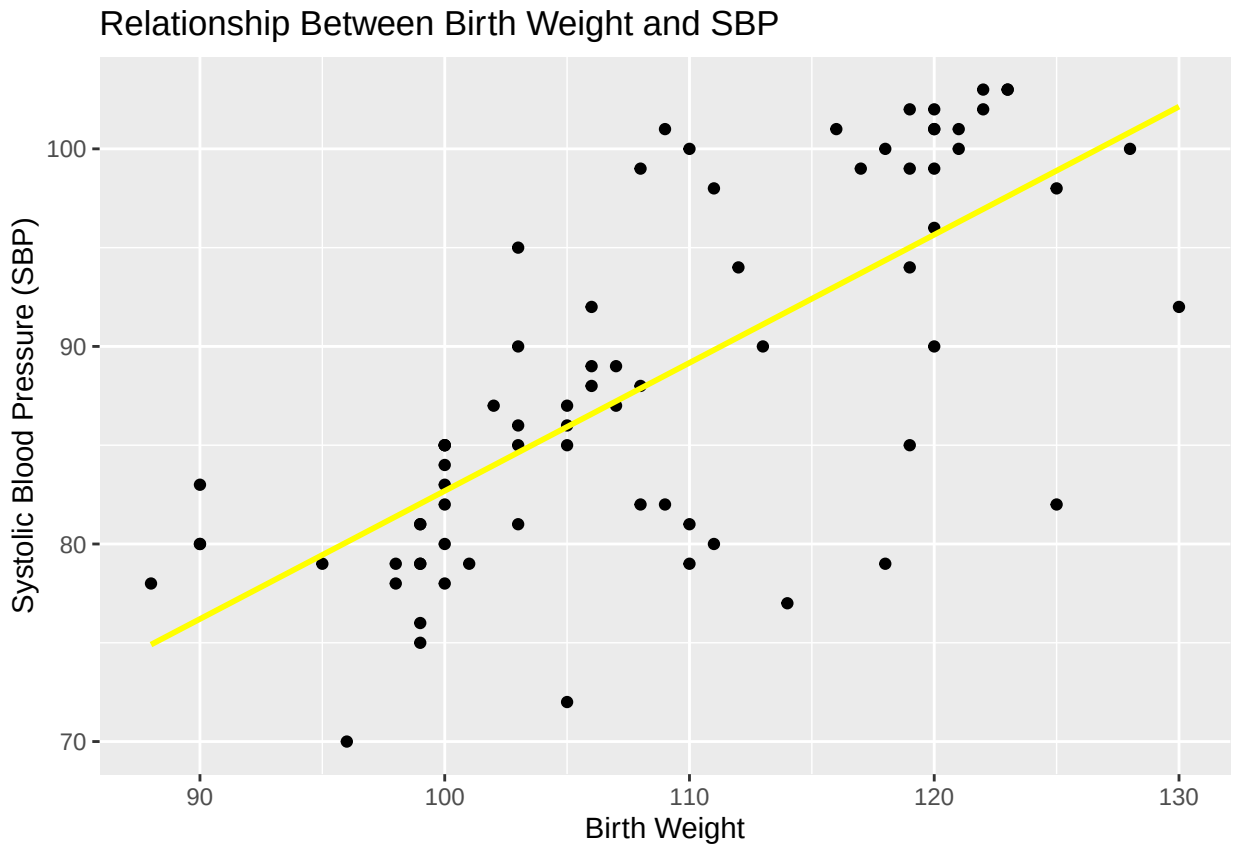
```r
# The data birthweightSBP1 on D2L recorded the systolic blood pressure of 78
# children, together with their birth weight and age. Use the data
# birthweightSBP on D2L to do the following:

# press "Import Dataset" -> press "From Excel..." -> press "Browse..." ->
# navigate & select .xlsx file -> copy (command + c) code from "Code Preview:"
# -> press "Import" -> paste (command + v) code to the RStudio Integrated
# Development Environment.
birthweightSBP1 <- read_excel("/Users/peytonhall/Desktop/Data211Midterm/birthweightSBP1.xlsx")
# View(birthweightSBP1)

# a) Use ggplot() to generate an appropriate graph to show the relationship
#    between birth weight(x) and SBP (y).
ggplot(birthweightSBP1, aes(x = Birthweight, y = SBP)) +
  geom_point() +
  labs(x = "Birth Weight", y = "Systolic Blood Pressure (SBP)") +
  ggtitle("Relationship Between Birth Weight and SBP")
```




```r
# b) Use appropriate functions in the ggplot2 to generate a regression line to
#    the graph. Color the line yellow without
#    showing the confidence interval band.
ggplot(birthweightSBP1, aes(x = Birthweight, y = SBP)) +
  geom_point() +
  geom_smooth(method = "lm", color = "yellow", se = FALSE) +  # regression line
```

```
  labs(x = "Birth Weight", y = "Systolic Blood Pressure (SBP)") +
  ggtitle("Relationship Between Birth Weight and SBP")
```

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



Question 08

```
# Use the USMerchants data on D2L. The dataset recorded the amount of US
# merchants of Corn, Rice, and Soybeans from 1970 to 1994. However, the data
# contains some mistakes. Use the for-loop and if-else conditional statement to
# do the following fix-up:

# press "Import Dataset" -> press "From Excel..." -> press "Browse..." ->
# navigate & select .xlsx file -> copy (command + c) code from "Code Preview:"
# -> press "Import" -> paste (command + v) code to the RStudio Integrated
# Development Environment.
USMerchants <- read_excel("/Users/peytonhall/Desktop/Data211Midterm/USMerchants.XLSX")
# View(USMerchants)

# a) For the years from 1970 to 1979, the corn prices
#    are actually the original amount +10
for (i in 1:nrow(USMerchants)) { # loop through the rows
```

```
  # if the year is between 1970 and 1979
  if (USMerchants$Year[i] >= 1970 && USMerchants$Year[i] <= 1979) {
    # Add 10 to the corn prices for those years
    USMerchants$Corn[i] <- USMerchants$Corn[i] + 10
  } # end if
} # end for
USMerchants
```

```
## # A tibble: 300 x 4
##     Year  Corn  Rice Soybeans
##    <dbl> <dbl> <dbl>    <dbl>
##  1  1970  137.  91.7     70.4
##  2  1970  137.  91.9     70.6
##  3  1970  137.  92.1     70.7
##  4  1970  137.  92.2     71.2
##  5  1970  138.  92.0     71.4
##  6  1970  139.  91.9     71.5
##  7  1970  139.  92.0     71.6
##  8  1970  139.  93.3     71.7
##  9  1970  138.  93.9     71.8
## 10  1970  138.  94.1     71.9
## # i 290 more rows
```

```
# b) For the years from 1980 to 1989, the rice prices
#    are actually the original amount*1.05
for (i in 1:nrow(USMerchants)) { # loop through the rows
  # if the year is between 1980 and 1989
  if (USMerchants$Year[i] >= 1980 && USMerchants$Year[i] <= 1989) {
    # Multiply the rice prices by 1.05 for those years
    USMerchants$Rice[i] <- USMerchants$Rice[i] * 1.05
  } # end if
} # end for
USMerchants
```

```
## # A tibble: 300 x 4
##     Year  Corn  Rice Soybeans
##    <dbl> <dbl> <dbl>    <dbl>
##  1  1970  137.  91.7     70.4
##  2  1970  137.  91.9     70.6
##  3  1970  137.  92.1     70.7
##  4  1970  137.  92.2     71.2
##  5  1970  138.  92.0     71.4
##  6  1970  139.  91.9     71.5
##  7  1970  139.  92.0     71.6
##  8  1970  139.  93.3     71.7
##  9  1970  138.  93.9     71.8
## 10  1970  138.  94.1     71.9
## # i 290 more rows
```

```
# c) For the years from 1990 to the end, the soybean prices
#    are the original amount-15
for (i in 1:nrow(USMerchants)) { # loop through the rows
```

```
  # if the year is between 1990 and 1994
  if (USMerchants$Year[i] >= 1990 && USMerchants$Year[i] <= 1994) {
    # Subtract 15 from the soybean prices for those years
    USMerchants$Soybeans[i] <- USMerchants$Soybeans[i] - 15
  } # end if
} # end for
USMerchants
```

```
## # A tibble: 300 x 4
##     Year  Corn  Rice Soybeans
##    <dbl> <dbl> <dbl>    <dbl>
##  1  1970  137.  91.7     70.4
##  2  1970  137.  91.9     70.6
##  3  1970  137.  92.1     70.7
##  4  1970  137.  92.2     71.2
##  5  1970  138.  92.0     71.4
##  6  1970  139.  91.9     71.5
##  7  1970  139.  92.0     71.6
##  8  1970  139.  93.3     71.7
##  9  1970  138.  93.9     71.8
## 10  1970  138.  94.1     71.9
## # i 290 more rows
```