Problem Set 4: Flow Cytometry and DESeq2

Due Friday, November 10th, 2023

For this assignment, you will need to install and load the following packages: dplyr, Rtsne, umap, and DESeq2

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## 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
```

Background: Flow cytometry is a method that biologists use to identify, count, and isolate different cell types in a population. Cells in suspension are pumped through microfluidics systems past detectors that measure many different optical and fluorescent properties. Different cell types in a mixed cell population can then be identified by labeling specific proteins with fluorescent antibodies or other reporters.

The amount of light that cells scatter in the forward direction (called forward-scatter or FSC) as they pass through the detection laser can sometimes be though of as a proxy for cell size. Another property called side-scatter (SSC) refers to the amount of light scattered in the horizontal direction, and can indicate granularity or irregularity in the density of the object passing the detector. Along with the scatter measurements, the experiment we will work with contains fluorescence measurements for three
colors: green, red, and blue. Here, the green channel is measuring the prevalence of the cell surface marker CD24, the red channel measures CD44, and the blue channel measures CD36.

Lung cancer is one of the most common types of cancer in the United States and the one with the highest mortality. There are many distinct cancer subtypes originating in the lung, two of which are **squamous cell carcinoma** and **adenocarcinoma**. For this homework assignment you will analyze data from some patients with lung cancer.

Question 1: visualizing cell populations

1a (7 points)

We have provided data from a flow cytometry experiment (`flow_cytometry_data.txt`) characterizing a mixed population of breast cancer cells. Our goal is to separate cell subpopulations that have high and low cancer stem cell potential based on their cell surface markers, and then to compare gene expression patterns between the two cell populations.

To start, load the flow cytometry data and generate univariate histograms for each of the three measured fluorophores (colors) and two scatter measurements. There should be five different histograms.

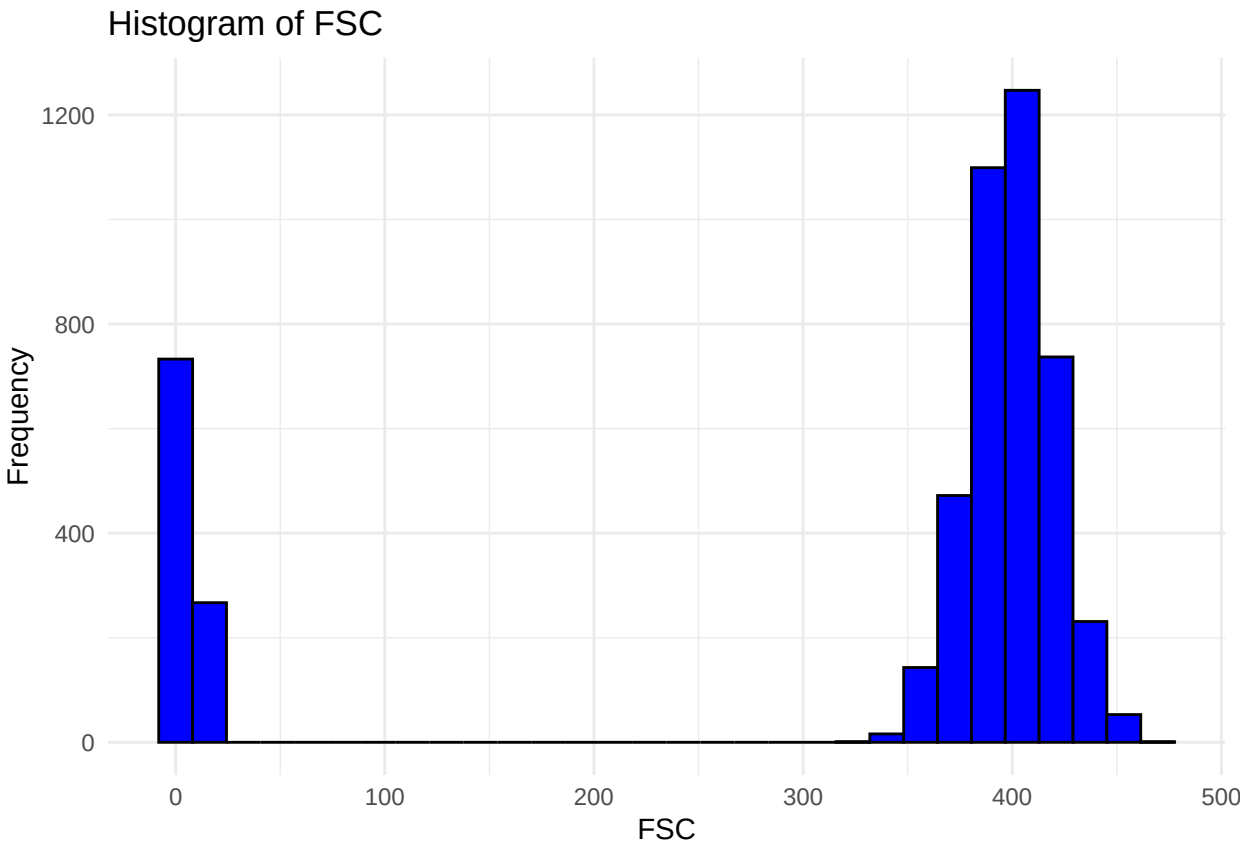
How many cell populations are visible, if the variables are only considered one at a time? That is, for each variable, how many distinct populations do you see in the histogram for that variable?

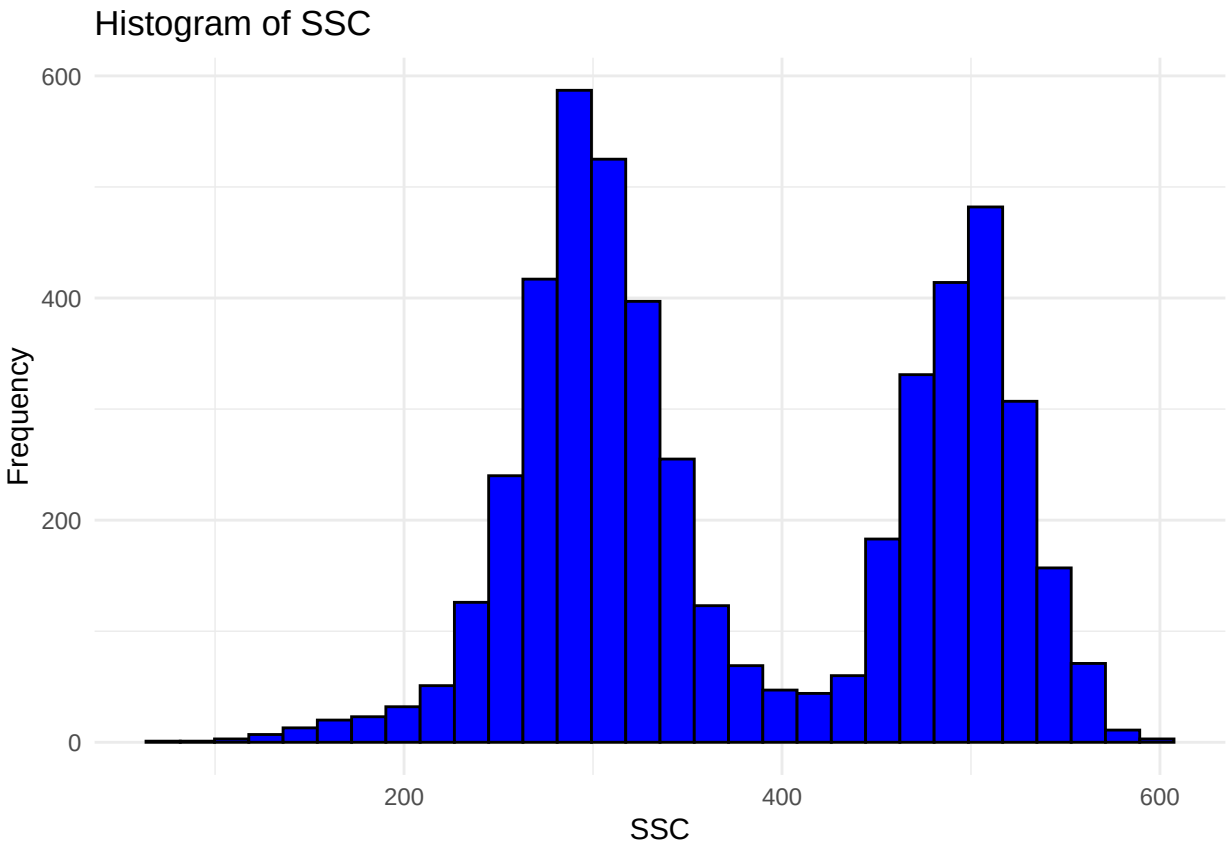
To report your answer, create a data frame, where the first column contains the variable name and the second column contains a numeric value describing the number of populations you expect there are, given the histogram distribution you observed for that variable.

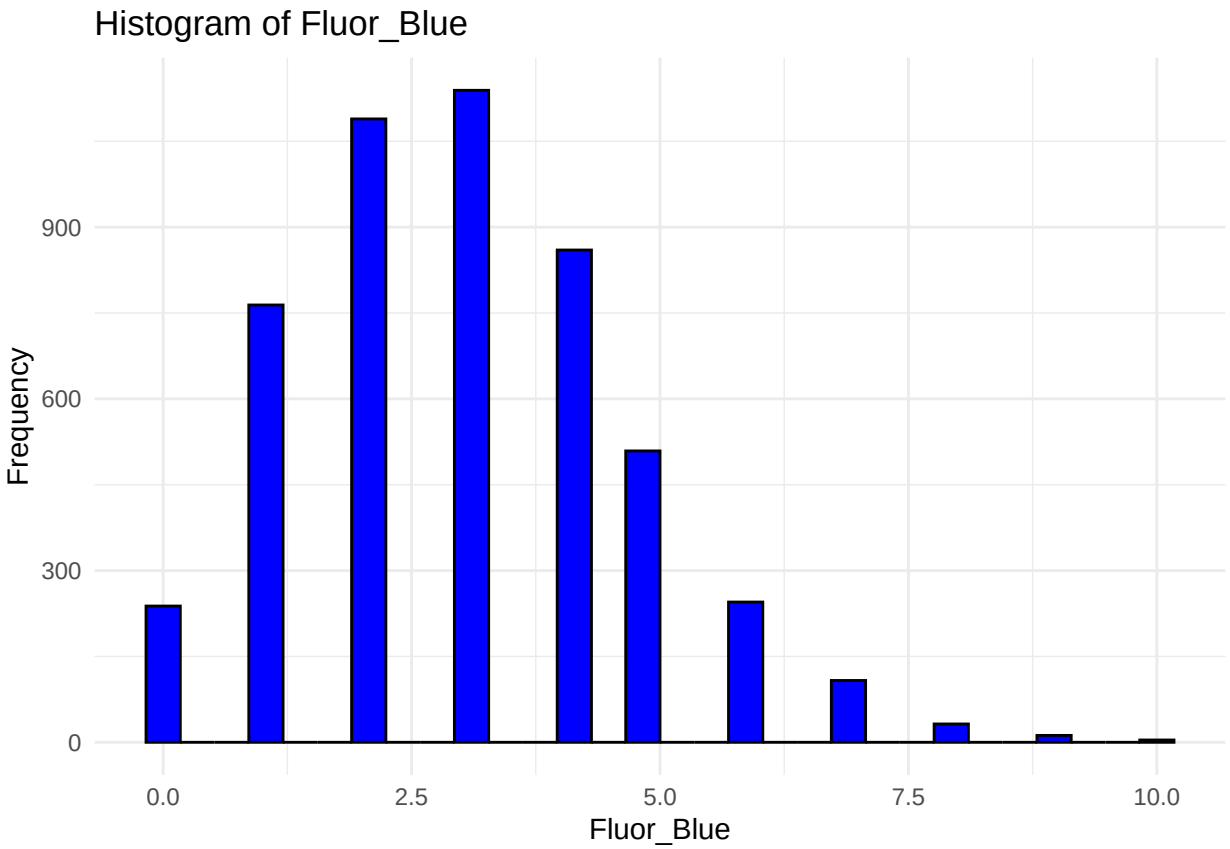
```
library(ggplot2)

flow_data <- read.table("flow_cytometry_data.txt", header = TRUE)
colnames(flow_data)
```

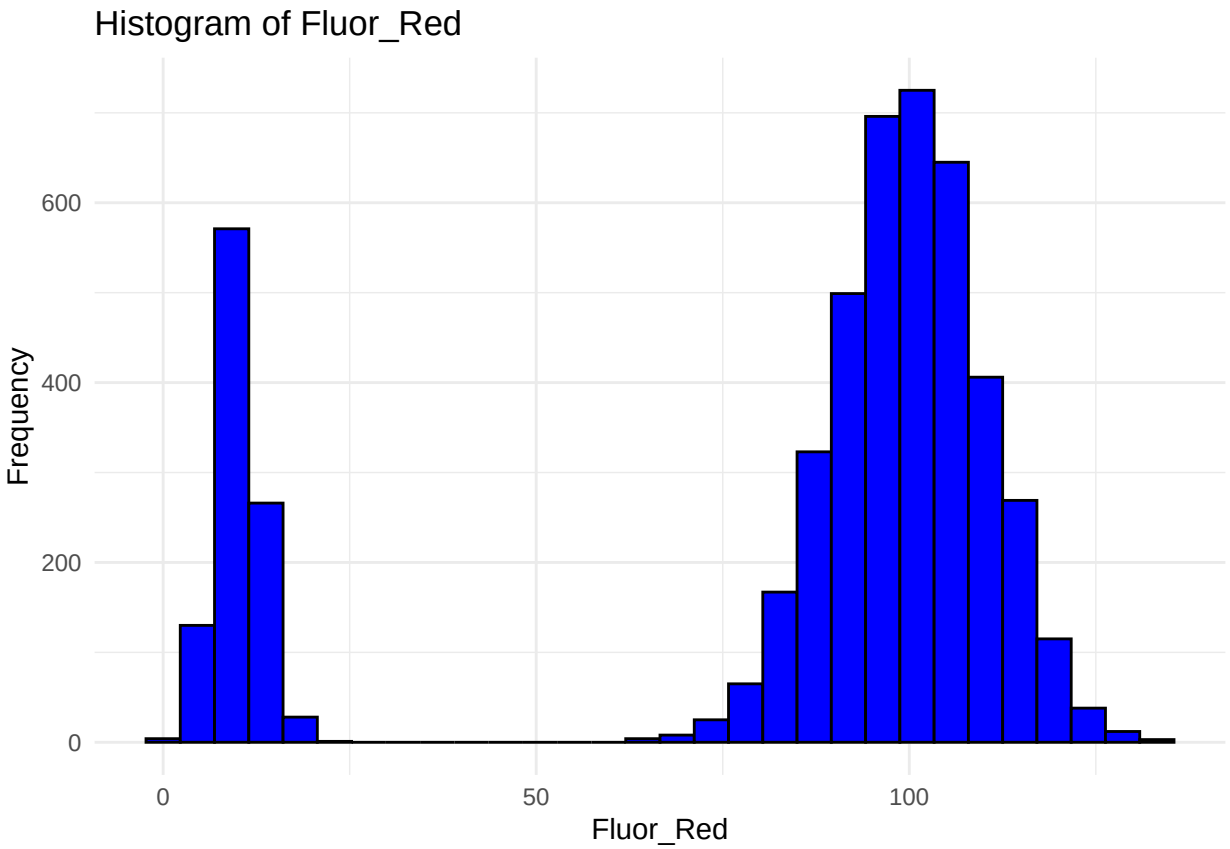
```
## [1] "event_ID"    "FSC"         "SSC"         "Fluor_Blue"  "Fluor_Red"
## [6] "Fluor_Green"
```

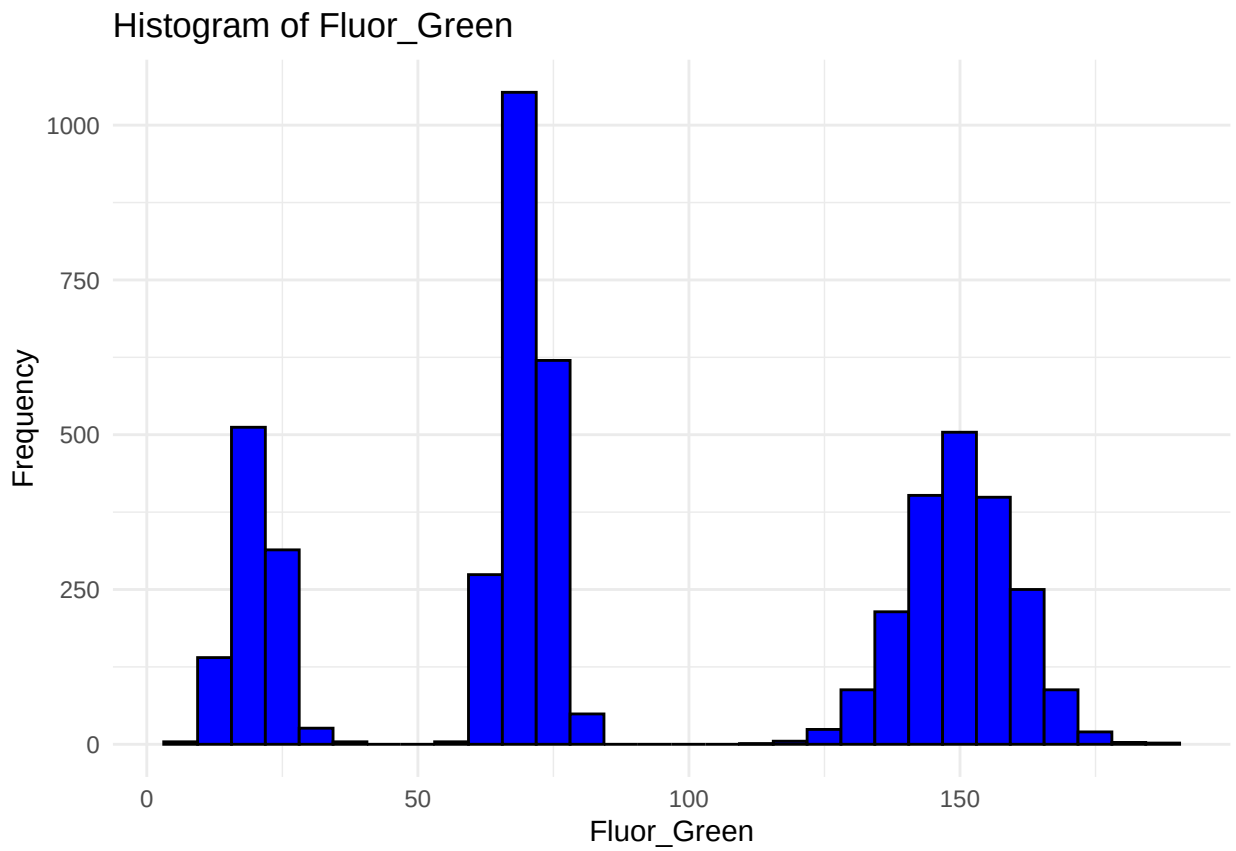
```
measurement_columns <- c("FSC", "SSC", "Fluor_Blue", "Fluor_Red", "Fluor_Green")

count_populations <- function(data, column_name) {
  ggplot(data, aes(x = .data[[column_name]])) +
    geom_histogram(bins = 30, fill = "blue", color = "black") +
    labs(title = paste("Histogram of", column_name), x = column_name, y = "Frequency") +
    theme_minimal()
}

population_counts <- data.frame(
  Variable = character(),
  Populations = numeric(),
  stringsAsFactors = FALSE
)

num_populations <- c(2, 2, 1, 2, 3)

for (i in seq_along(measurement_columns)) {
  print(count_populations(flow_data, measurement_columns[i]))

  population_counts <- rbind(population_counts, data.frame(Variable = measurement_columns[i], Populatio
}
```


Histogram of FSC
Frequency
1200
800
400
0
0
100
200
300
400
500
FSC


Histogram of SSC
Frequency
600
400
200
0
200
400
600
SSC


Histogram of Fluor_Blue
Frequency
900
600
300
0
0.0
2.5
5.0
7.5
10.0
Fluor_Blue


Histogram of Fluor_Red
Frequency
600
400
200
0
0
50
100
Fluor_Red




```
print(population_counts)
```

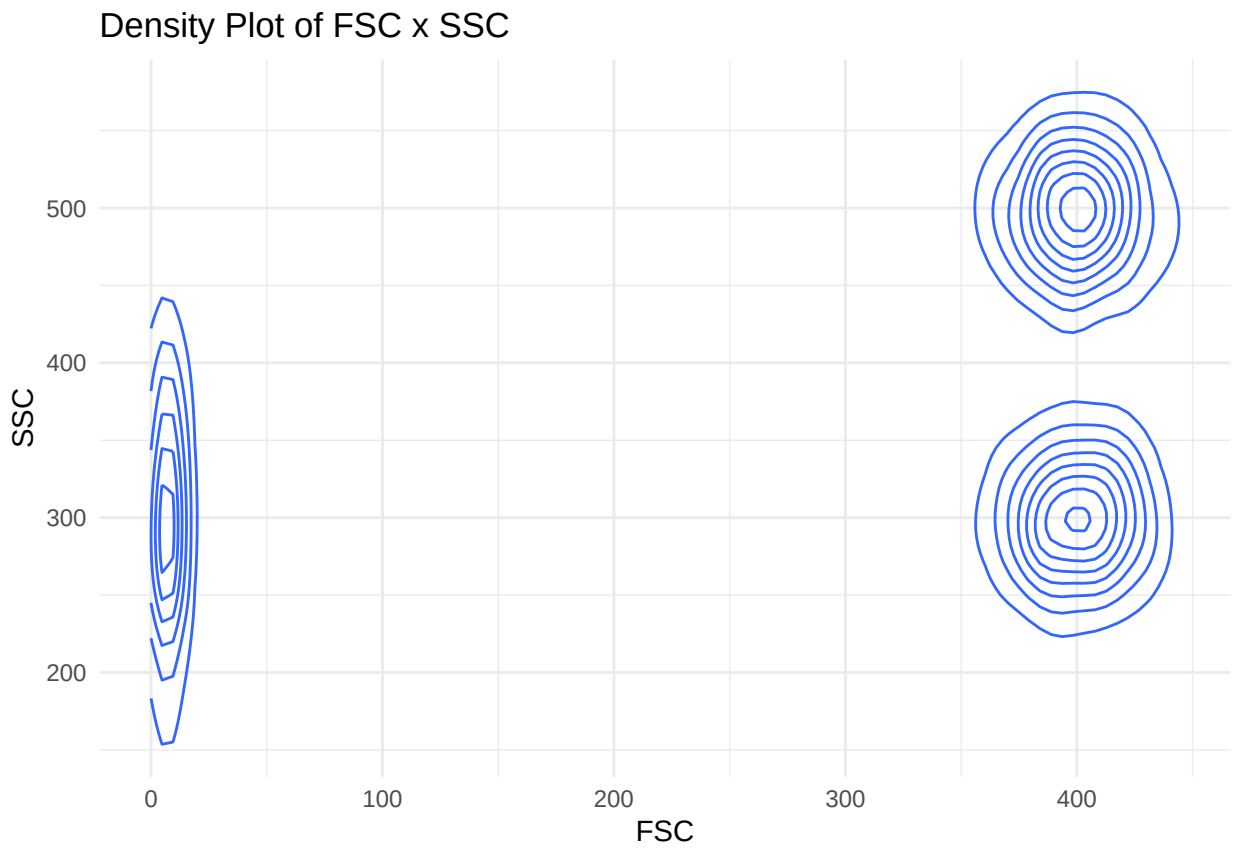
```
##      Variable Populations
## 1         FSC           2
## 2         SSC           2
## 3  Fluor_Blue           1
## 4   Fluor_Red           2
## 5 Fluor_Green           3
```

1b (6 points)

Bivariate scatterplots are frequently used during flow cytometry experiments to identify cell populations by eye. This may enable you to differentiate between cellular debris, which usually have small FSC and moderate to high SSC, and true cells, which tend to have larger FSC values and variable SSC (depending on the structural properties of the cells).

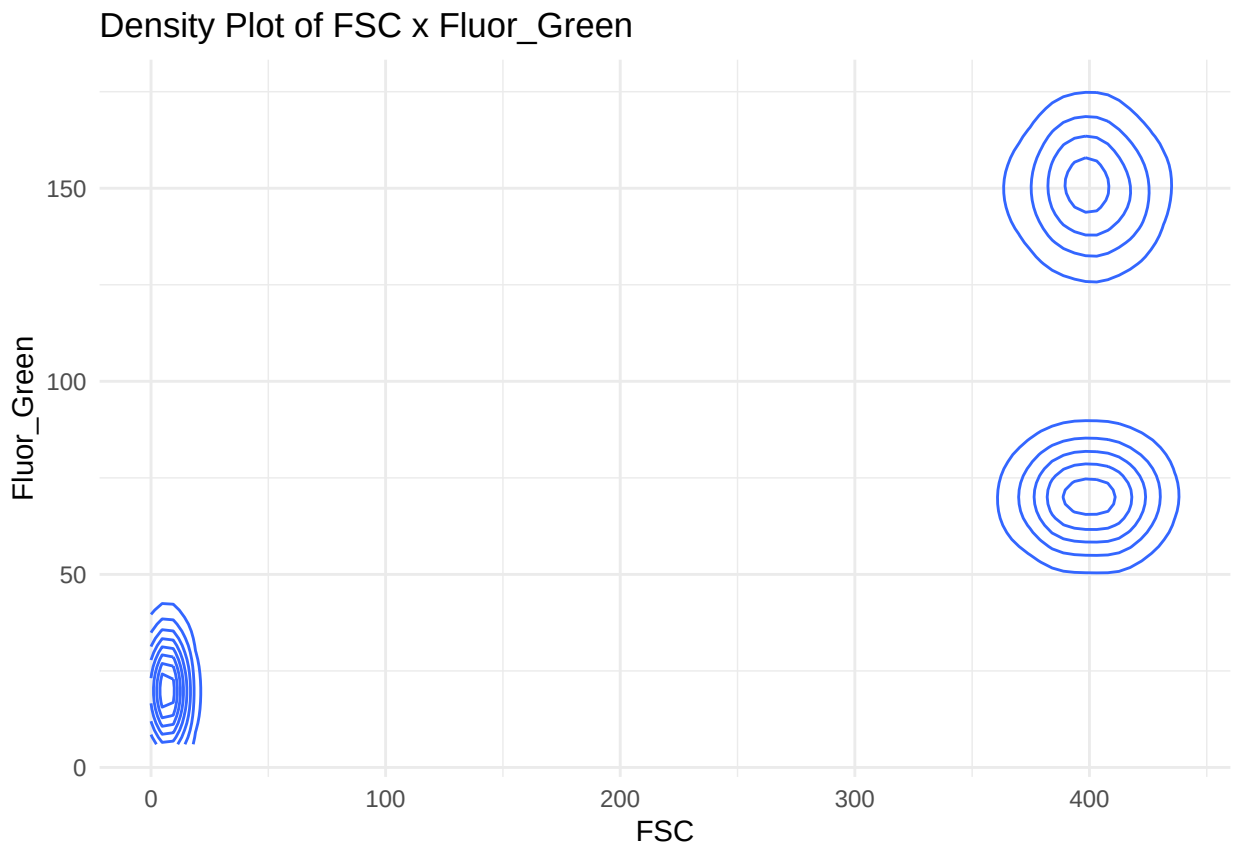
Using `ggplot2` and `geom_density2d()`, generate two two-dimensional density plots: one of FSC x SSC, and one of FSC x Fluor_Green (either orientation is acceptable for both plots).
FSC x SSC

```
ggplot(flow_data, aes(x = FSC, y = SSC)) +
 geom_density2d() +
 labs(title = "Density Plot of FSC x SSC", x = "FSC", y = "SSC") +
 theme_minimal()
```



FSC x Fluor_Green

```
ggplot(flow_data, aes(x = FSC, y = Fluor_Green)) +
  geom_density2d() +
  labs(title = "Density Plot of FSC x Fluor_Green", x = "FSC", y = "Fluor_Green") +
  theme_minimal()
```



1c (4 points)

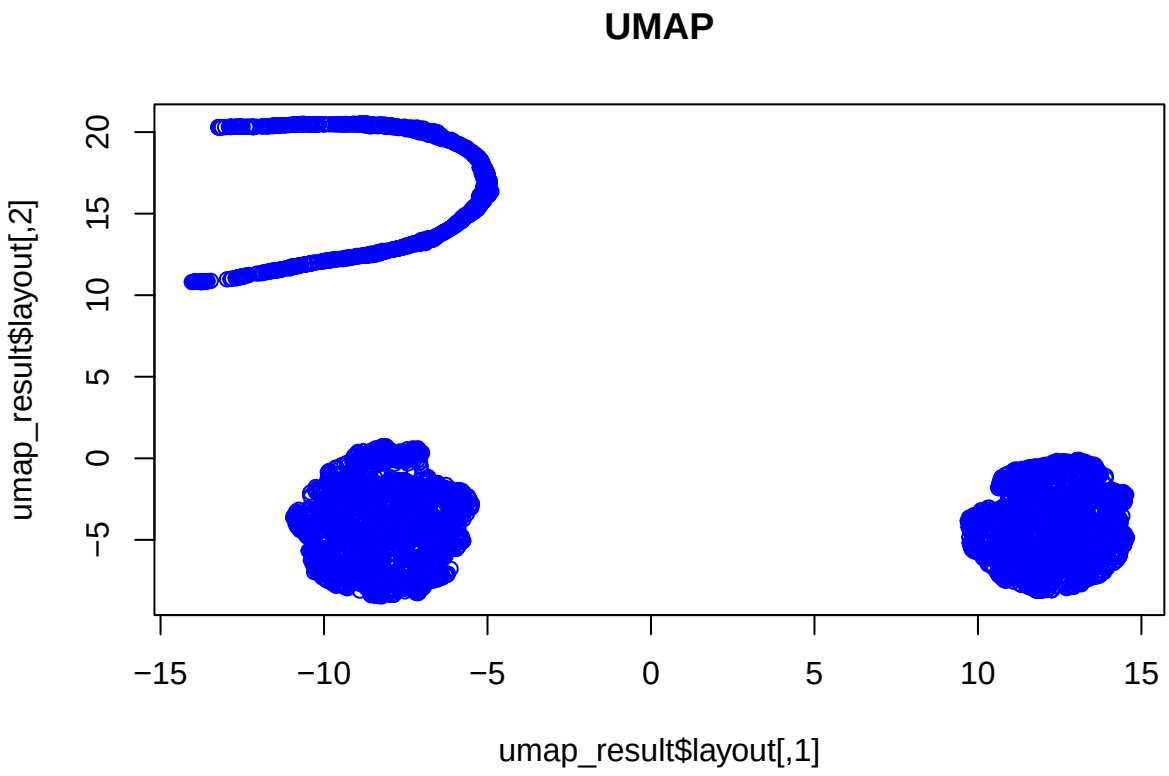
Explain how the density plots you see are consistent with the expected number of populations of cells you found in problem 1a for SSC and Fluor_Green. Given the description of FSC and SSC, above, which clusters are likely to be true clusters of cells and which are likely to be cellular debris?

The density plots, generated for FSC x SSC and FSC x Fluor_Green, align with the expected cell population characteristics outlined in problem 1a. In the FSC x SSC plot, higher density on the right signifies true cell clusters with larger FSC and variable SSC, while lower density on the left indicates regions associated with smaller FSC and moderate to high SSC, indicative of cellular debris. Similarly, in the FSC x Fluor_Green plot, higher-density clusters on the right correspond to cells expressing the Fluor_Green marker, characterized by larger FSC and variable Fluor_Green values, while lower-density clusters on the left represent cells with smaller FSC and potentially lower or no Fluor_Green expression. Overall, these density plots support the distinction between genuine cell populations and cellular debris based on their respective FSC, SSC, and Fluor_Green characteristics.

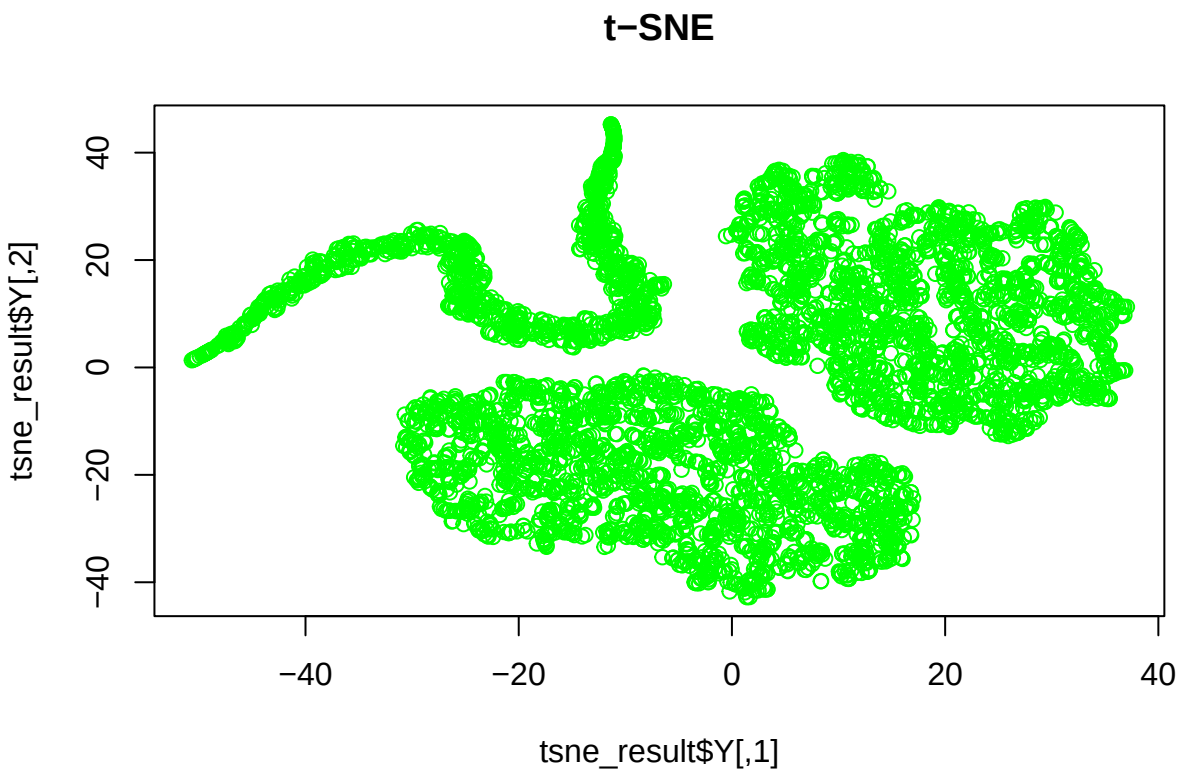
1d (6 points) Further visualize the cell populations in this sample using UMAP, tSNE, and PCA, taking into account all five of the measured variables in each projection (*Hint: refer to the slides from the Dimensionality Reduction lecture if you are unsure how to extract the coordinates*). How many cell populations can you see? Do all three dimensionality reduction methods agree?

```
# Combine all variables into a matrix
data_matrix <- flow_data[, measurement_columns]

# UMAP
umap_result <- umap(data_matrix)

# t-SNE
```

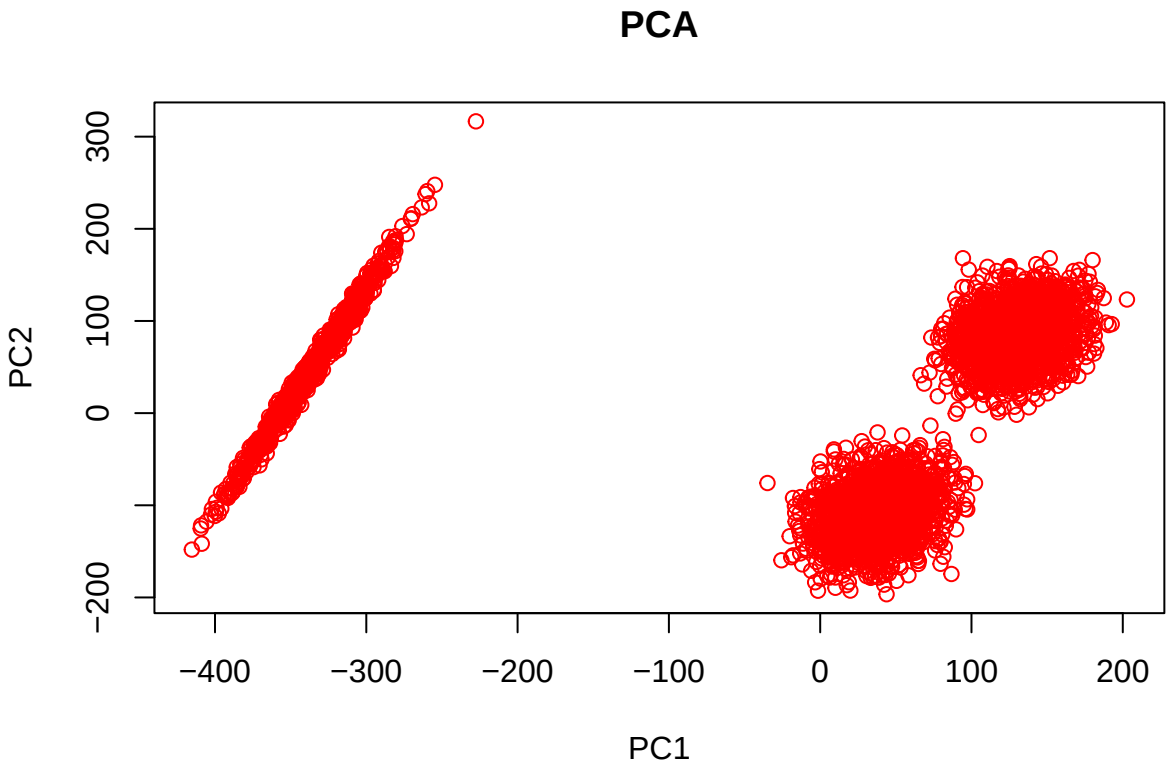
```r
tsne_result <- Rtsne(data_matrix, dims = 2)

# PCA
pca_result <- prcomp(data_matrix)$x


# UMAP plot
plot(umap_result$layout, col = "blue", main = "UMAP")
```




```r
# t-SNE plot
plot(tsne_result$Y, col = "green", main = "t-SNE")
```




```r
# PCA plot
plot(pca_result[, 1:2], col = "red", main = "PCA")
```




```
# Reset graphics parameters
par(mfrow = c(1, 1))
```

1e (4 points) How many cell populations can you see? Do all three dimensionality reduction methods agree?

two clusters (representing cell populations) and a line i can see in all of three dimensionality. All 3 dimensionality reduction methods agree.

Question 2: identifying cancer stem cell potential

2a (4 points) Create a data frame containing the UMAP coordinates from the previous question along with the scatter values and the fluorescence values for each color for the corresponding observation. Add columns to this data frame containing all six possible ratios of the three different fluorescence values (e.g., 'Fluor_Green/Fluor_Red', or 'Fluor_Red/Fluor_Green').

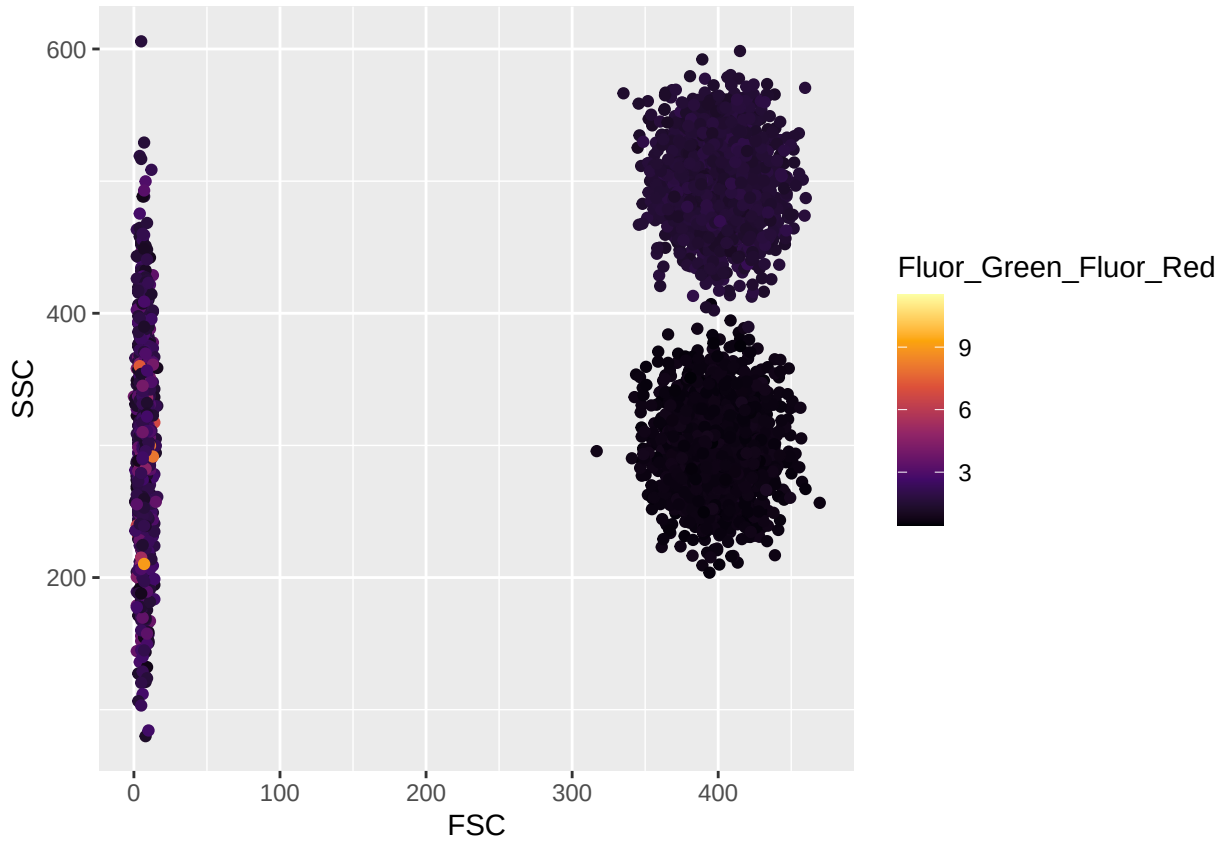
```
fluorescence_data <- flow_data[, c("Fluor_Green", "Fluor_Red", "Fluor_Blue", "FSC", "SSC")]

# Combine UMAP coordinates and fluorescence data
umap_with_fluorescence <- cbind(umap_result$layout, fluorescence_data)

# Calculating six possible ratios
umap_with_ratios <- umap_with_fluorescence
epsilon <- 1e-8
umap_with_ratios$Fluor_Green_Fluor_Red <- umap_with_ratios$Fluor_Green / (umap_with_ratios$Fluor_Red + e
```

```
umap_with_ratios$Fluor_Red_Fluor_Green <- umap_with_ratios$Fluor_Red / (umap_with_ratios$Fluor_Green + e
umap_with_ratios$Fluor_Green_Fluor_Blue <- umap_with_ratios$Fluor_Green / (umap_with_ratios$Fluor_Blue +
umap_with_ratios$Fluor_Blue_Fluor_Green <- umap_with_ratios$Fluor_Blue / (umap_with_ratios$Fluor_Green +
umap_with_ratios$Fluor_Red_Fluor_Blue <- umap_with_ratios$Fluor_Red / (umap_with_ratios$Fluor_Blue + eps
umap_with_ratios$Fluor_Blue_Fluor_Red <- umap_with_ratios$Fluor_Blue / (umap_with_ratios$Fluor_Red + eps

head(umap_with_ratios)
```

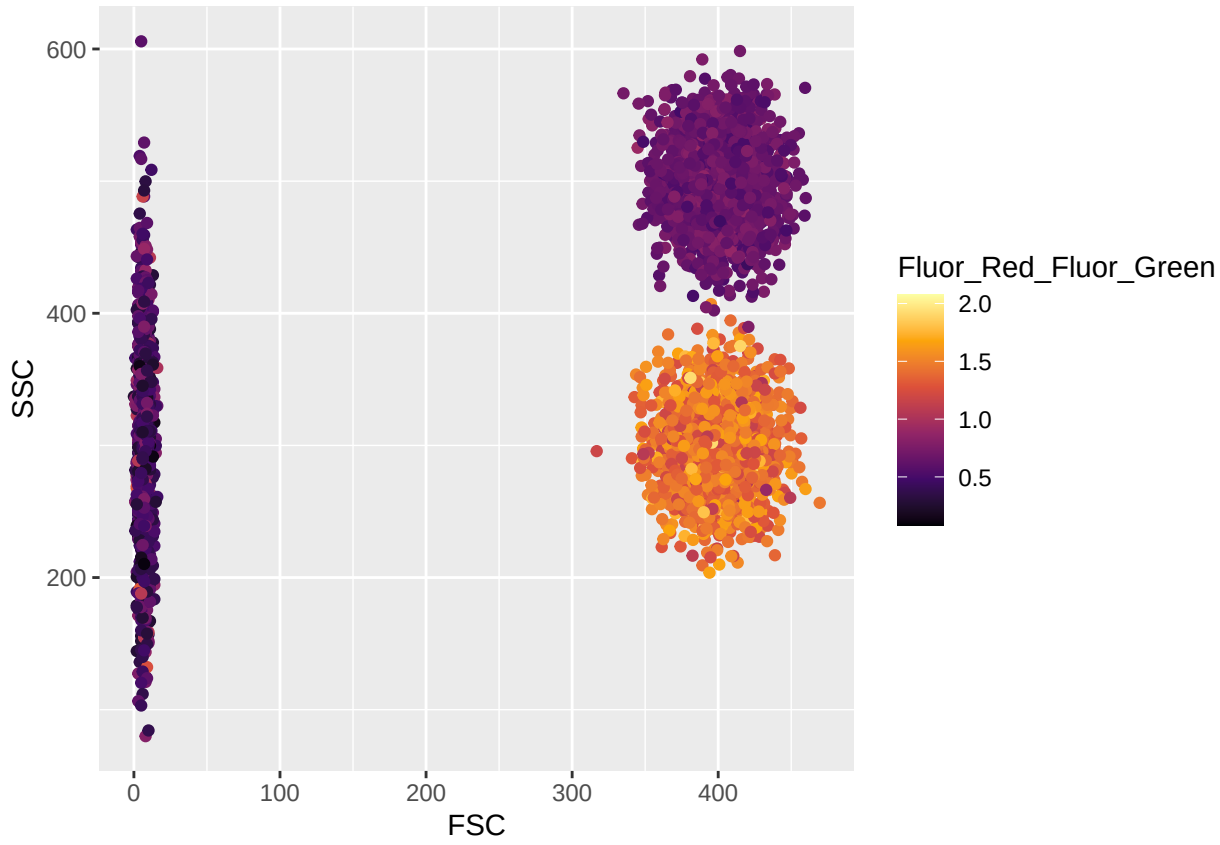
```
##            1        2 Fluor_Green Fluor_Red Fluor_Blue FSC      SSC
## 1  -8.188724 20.38564          21        10          4  13 241.3709
## 2  -9.733579 12.16506          17        11          2   4 375.6844
## 3  -5.473218 15.28829          25        11          4   9 310.1476
## 4 -12.156617 20.30901          26        13          1   5 167.7719
## 5  -5.734536 14.66400          17         9          0   8 316.1580
## 6  -5.475860 18.75609          17        11          5   6 274.6172
##   Fluor_Green_Fluor_Red Fluor_Red_Fluor_Green Fluor_Green_Fluor_Blue
## 1              2.100000             0.4761905               5.25e+00
## 2              1.545455             0.6470588               8.50e+00
## 3              2.272727             0.4400000               6.25e+00
## 4              2.000000             0.5000000               2.60e+01
## 5              1.888889             0.5294118               1.70e+09
## 6              1.545455             0.6470588               3.40e+00
##   Fluor_Blue_Fluor_Green Fluor_Red_Fluor_Blue Fluor_Blue_Fluor_Red
## 1            0.19047619             2.50e+00           0.40000000
## 2            0.11764706             5.50e+00           0.18181818
## 3            0.16000000             2.75e+00           0.36363636
## 4            0.03846154             1.30e+01           0.07692308
## 5            0.00000000             9.00e+08           0.00000000
## 6            0.29411765             2.20e+00           0.45454545
```

2b (6 points) Using `ggplot2` and the new data frame you just made, for each ratio, create a scatter plot of cells with the FSC values on the x axis and SSC values on the y axis, colored by the intensity of the ratio (we suggest adding `+scale_color_viridis_c(option='inferno')` to each ggplot command in order to create color scales that are easily distinguished).

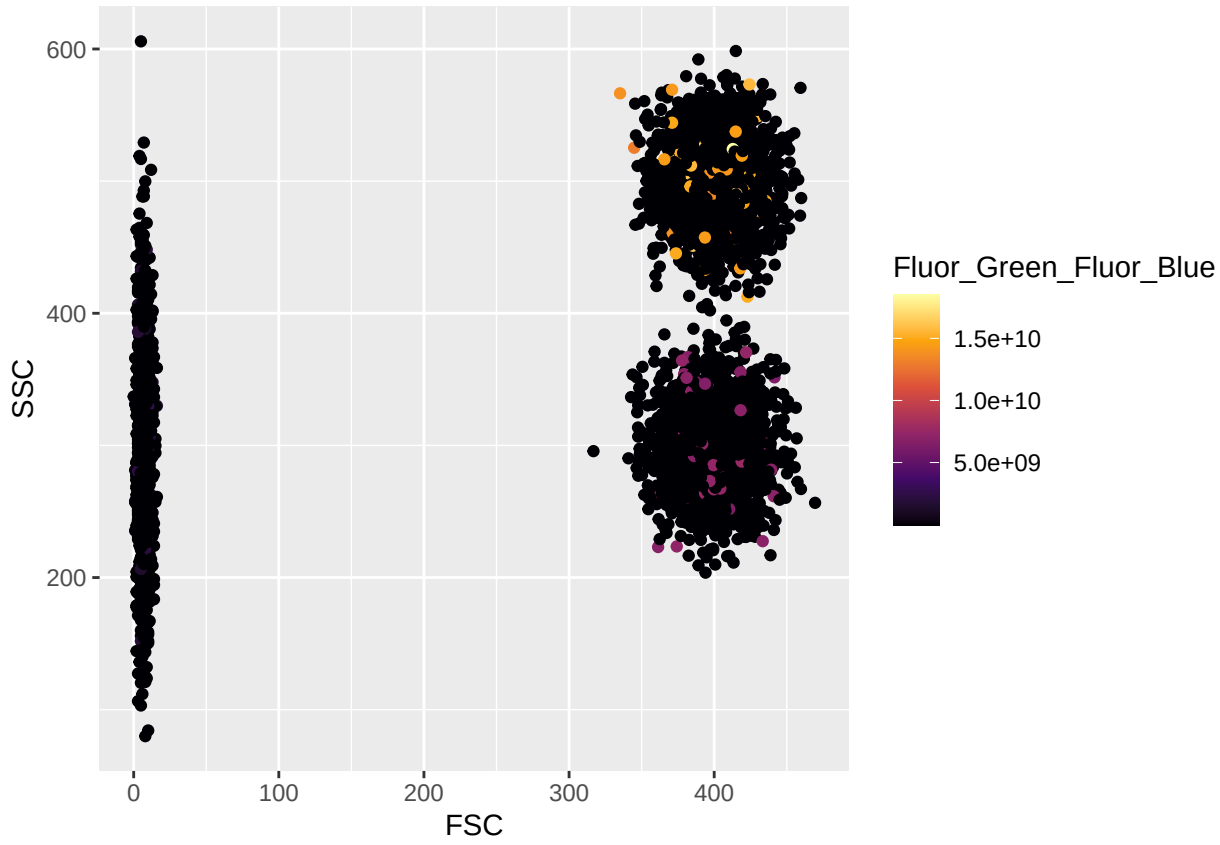
```
library(ggplot2)

# Plot for 'Fluor_Green/Fluor_Red' ratio
ggplot(umap_with_ratios, aes(x = FSC, y = SSC, color = Fluor_Green_Fluor_Red)) +
  geom_point() +
  scale_color_viridis_c(option = "inferno")
```



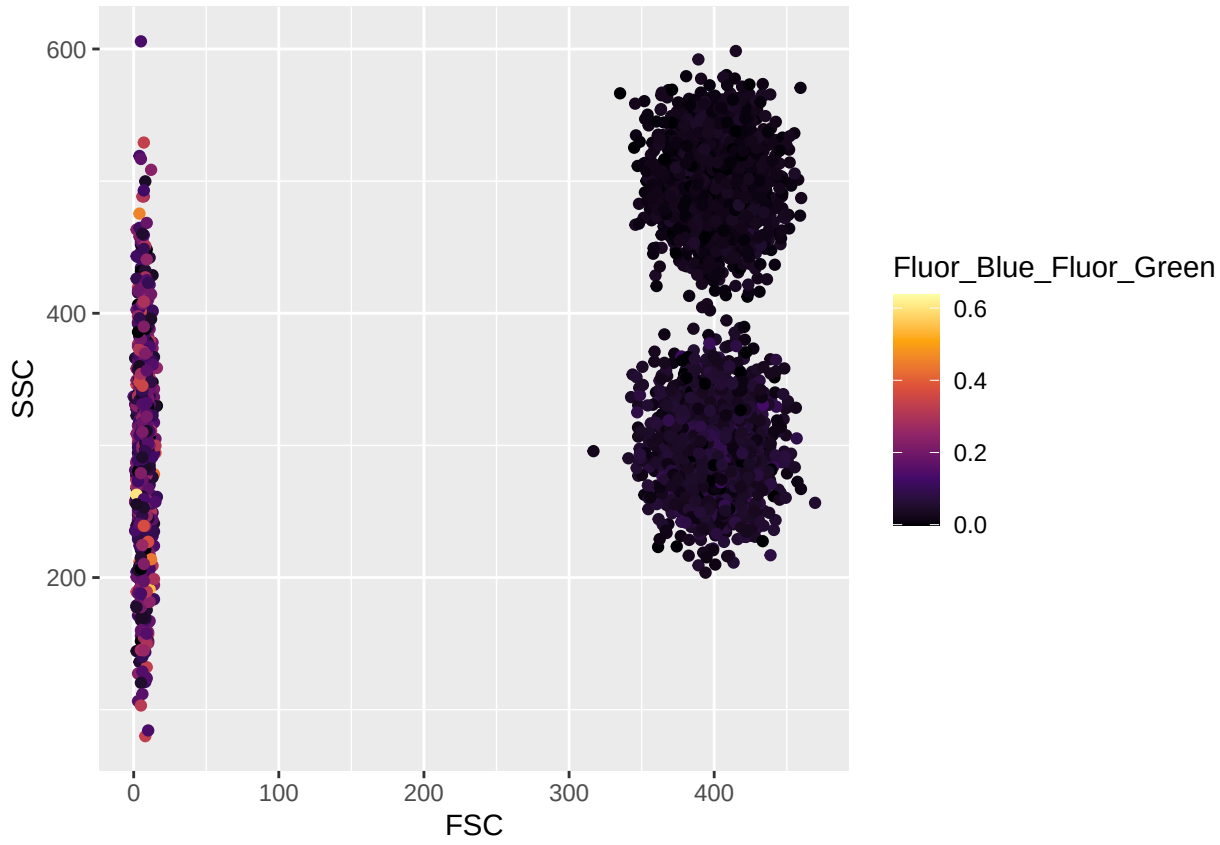

```
# Plot for 'Fluor_Red/Fluor_Green' ratio
ggplot(umap_with_ratios, aes(x = FSC, y = SSC, color = Fluor_Red_Fluor_Green)) +
  geom_point() +
  scale_color_viridis_c(option = "inferno")
```



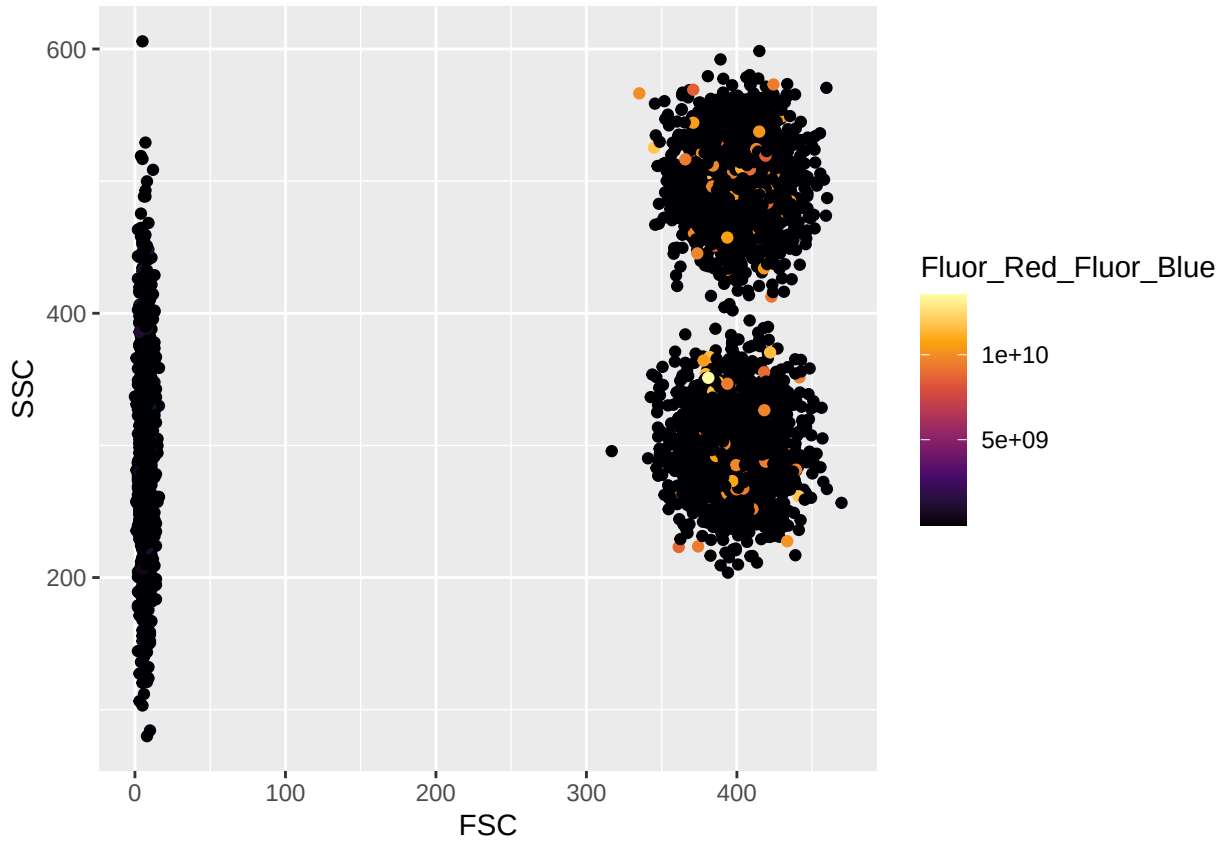

```
# Plot for 'Fluor_Green/Fluor_Blue' ratio
ggplot(umap_with_ratios, aes(x = FSC, y = SSC, color = Fluor_Green_Fluor_Blue)) +
  geom_point() +
  scale_color_viridis_c(option = "inferno")
```



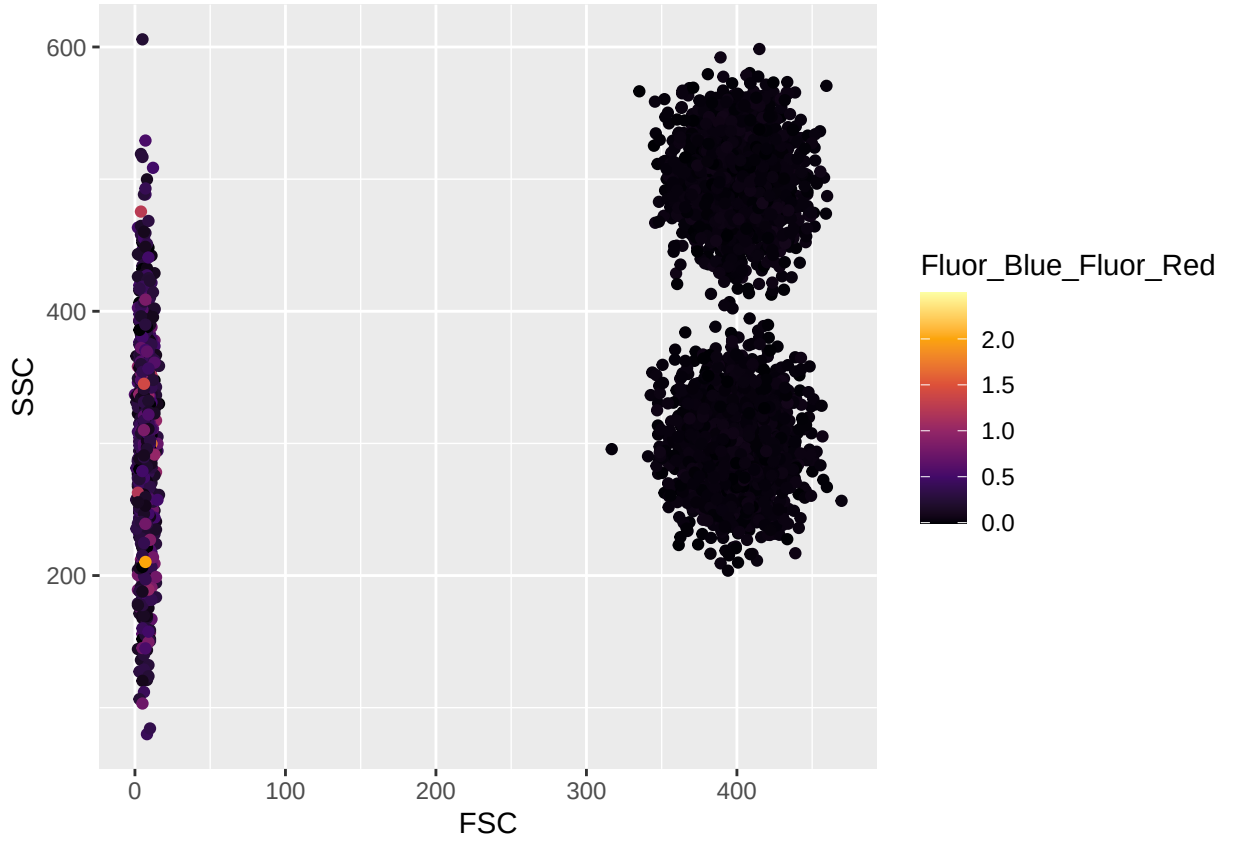

```
# Plot for 'Fluor_Blue/Fluor_Green' ratio
ggplot(umap_with_ratios, aes(x = FSC, y = SSC, color = Fluor_Blue_Fluor_Green)) +
  geom_point() +
  scale_color_viridis_c(option = "inferno")
```




```
# Plot for 'Fluor_Red/Fluor_Blue' ratio
ggplot(umap_with_ratios, aes(x = FSC, y = SSC, color = Fluor_Red_Fluor_Blue)) +
  geom_point() +
  scale_color_viridis_c(option = "inferno")
```




```
# Plot for 'Fluor_Blue/Fluor_Red' ratio
ggplot(umap_with_ratios, aes(x = FSC, y = SSC, color = Fluor_Blue_Fluor_Red)) +
  geom_point() +
  scale_color_viridis_c(option = "inferno")
```



2c (6 points) Looking at these six plots, can you see one that best separates the true cell clusters from each other? To what ratio of cell surface markers does it correspond?

'Fluor_Red/Fluor_Green' ratio plot might exhibit better separation of cell clusters within this dataset.For the 'Fluor_Red/Fluor_Green' ratio plot, I can notice areas with higher color intensity, indicating data points with higher 'Fluor_Red/Fluor_Green' ratios. Cells within these areas might express surface markers different from those in other regions, possibly representing specific subgroups of cells.The figure shows that there are subpopulations of samples with significant expression of both markers.

The ratio by the graph could be 1 (or 1.25).

3. Hypothesis testing.

3a (3 points)

Download from the private course page, and load, the data set `clindata.rds`, which contains data from 50 lung cancer patients. The first 25 patients have LUng ADenocarcinoma (as indicated in sample names containing the string 'LUAD'); the next 25 have LUng Squamous Cell carconoma (their sample names contain 'LUSC'). The file includes patient ages, platelet levels, and real-valued normalized expression data for eight genes.

```
data <- readRDS("clindata.rds")

str(data)
```

```
## 'data.frame':    50 obs. of  11 variables:
##  $ samples  : chr  "TCGA.LUAD.070" "TCGA.LUAD.23d" "TCGA.LUAD.278" "TCGA.LUAD.482" ...
```

```
##  $ ages     : num  57.3 54.4 55.1 63.1 77.6 ...
##  $ platelets: num  288 402 103 200 426 223 300 289 225 142 ...
##  $ IL4      : num  42.4 47.1 69.2 32.5 56.5 ...
##  $ IL10     : num  10.69 8.97 8.28 6.54 10.62 ...
##  $ CRISP3   : num  173 229 222 245 176 ...
##  $ MMP8     : num  11.8 23 20.2 24.1 23.3 ...
##  $ PDL5     : num  103.3 93.3 106.2 91.6 82 ...
##  $ SLC1A3   : num  102.3 81.2 65.2 115.6 96.9 ...
##  $ TNFA     : num  4.64 3.14 21.11 13.53 8.57 ...
##  $ IFNG     : num  25.4 32.7 32.8 18.4 28 ...
```

```
head(data)
```

```
##         samples  ages platelets   IL4  IL10 CRISP3  MMP8   PDL5 SLC1A3  TNFA
## 1 TCGA.LUAD.070 57.34       288 42.38 10.69 172.60 11.83 103.35 102.35  4.64
## 2 TCGA.LUAD.23d 54.40       402 47.09  8.97 228.64 23.03  93.26  81.18  3.14
## 3 TCGA.LUAD.278 55.14       103 69.15  8.28 222.12 20.23 106.23  65.25 21.11
## 4 TCGA.LUAD.482 63.12       200 32.50  6.54 244.98 24.08  91.61 115.63 13.53
## 5 TCGA.LUAD.5eb 77.59       426 56.48 10.62 175.64 23.29  81.99  96.89  8.57
## 6 TCGA.LUAD.712 61.02       223 59.99  9.99 241.36 24.56  99.71  65.58  6.46
##    IFNG
## 1 25.41
## 2 32.68
## 3 32.77
## 4 18.44
## 5 28.02
## 6 31.72
```

```
summary(data)
```

```
##    samples               ages         platelets          IL4
##  Length:50          Min.   :49.60   Min.   : 27.0   Min.   : 21.74
##  Class :character   1st Qu.:59.97   1st Qu.:100.8   1st Qu.: 59.36
##  Mode  :character   Median :65.69   Median :142.5   Median : 71.81
##                     Mean   :65.20   Mean   :181.0   Mean   : 72.38
##                     3rd Qu.:69.94   3rd Qu.:245.0   3rd Qu.: 87.90
##                     Max.   :80.46   Max.   :429.0   Max.   :105.77
##       IL10            CRISP3            MMP8            PDL5
##  Min.   : 0.19   Min.   : 85.02   Min.   : 1.62   Min.   : 79.57
##  1st Qu.: 1.71   1st Qu.:127.73   1st Qu.: 9.62   1st Qu.: 96.93
##  Median : 6.65   Median :160.13   Median :13.97   Median :111.30
##  Mean   : 6.08   Mean   :162.34   Mean   :14.17   Mean   :111.92
##  3rd Qu.: 9.75   3rd Qu.:196.99   3rd Qu.:20.07   3rd Qu.:125.23
##  Max.   :14.58   Max.   :244.98   Max.   :24.56   Max.   :152.12
##      SLC1A3            TNFA             IFNG
##  Min.   : 21.74   Min.   : 0.420   Min.   :18.44
##  1st Qu.: 49.87   1st Qu.: 7.058   1st Qu.:31.24
##  Median : 74.83   Median :16.985   Median :43.52
##  Mean   : 74.52   Mean   :17.171   Mean   :40.79
##  3rd Qu.: 99.25   3rd Qu.:27.538   3rd Qu.:50.80
##  Max.   :126.84   Max.   :39.730   Max.   :59.08
```

```r
LUAD_data <- data[1:25, ]

LUSC_data <- data[26:50, ]
```

3b (6 points)

Look at the data set. Is this data frame in long or wide format? Is it tidy data? Explain your answers.

Write answer here. the data is in a wide format. Each row represents a different sample, and the columns hold distinct types of measurements. This arrangement adheres to the characteristics of tidy data, as each variable is in its own separate column, and each observation is in its respective row.

3c (6 points) Suppose you are interested in whether there are differences in the ages of the patients with adenocarcinoma and those with squamous cell carcinoma.

To compare these, you should first ensure that the ages are sufficiently close to normally distributed that you can use a t-test to compare their means.

Read about the `shapiro.test()` function (see notes from 11/1/23). Use `shapiro.test()` to assess the normality of all of the age data. Do it again to assess normality of just the adenocarcinoma ages, and for just the squamous cell carcinoma ages.

Shapiro-Wilk test for normality of all age data

```r
shapiro_all <- shapiro.test(data$ages)
cat("Shapiro-Wilk test for all ages data:\n")
```

```
## Shapiro-Wilk test for all ages data:
```

```r
print(shapiro_all)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data$ages
## W = 0.98015, p-value = 0.5582
```

Shapiro-Wilk test for LUAD (LUng ADenocarcinoma) ages

```r
shapiro_luad <- shapiro.test(LUAD_data$ages)
cat("\nShapiro-Wilk test for LUAD ages:\n")
```

```
##
## Shapiro-Wilk test for LUAD ages:
```

```r
print(shapiro_luad)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  LUAD_data$ages
## W = 0.95081, p-value = 0.2614
```

Shapiro-Wilk test for LUSC (LUng Squamous Cell Carcinoma) ages

```
shapiro_lusc <- shapiro.test(LUSC_data$ages)
cat("\nShapiro-Wilk test for LUSC ages:\n")
```

```
##
## Shapiro-Wilk test for LUSC ages:
```

```
print(shapiro_lusc)
```

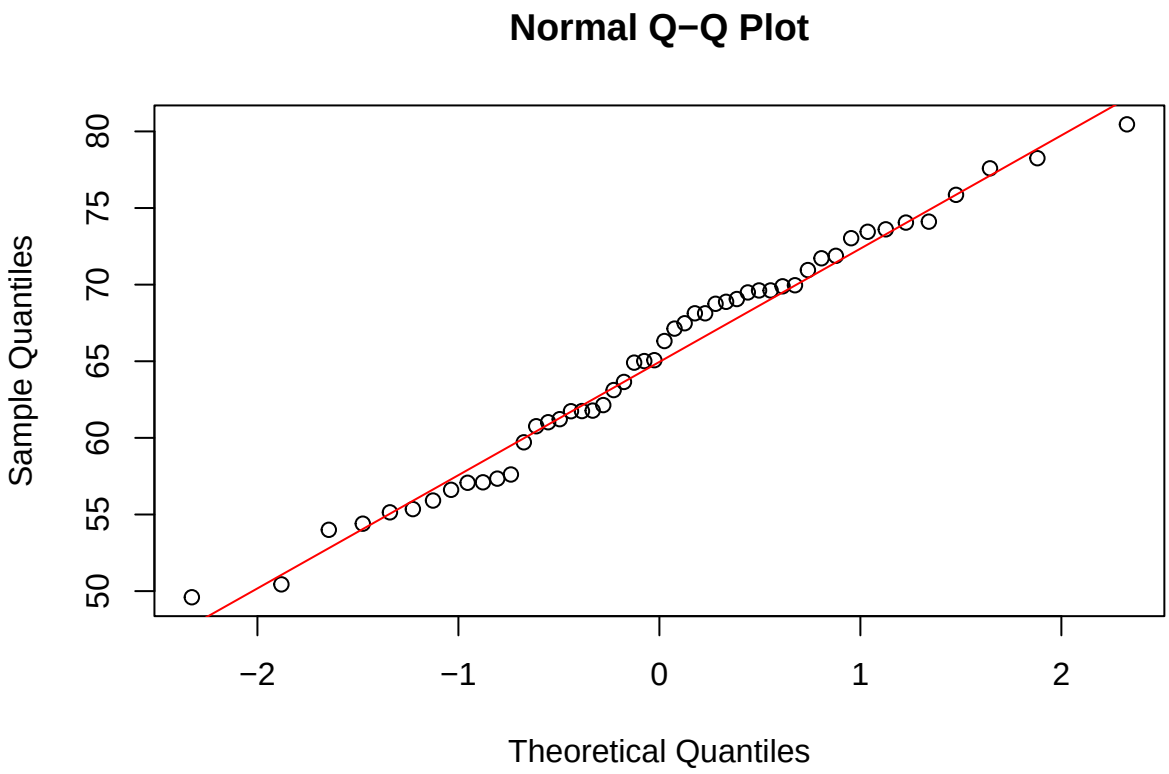
```
##
##  Shapiro-Wilk normality test
##
## data:  LUSC_data$ages
## W = 0.97962, p-value = 0.8775
```

3d (6 points)

Next, use `qqnorm()` plus `qqline()` in base R graphics to visualize the normality of the age data. Do this for the entire set of age data, and then just for the ages of adenocarcinoma patients, and just for the ages of squamous cell carcinoma patients.

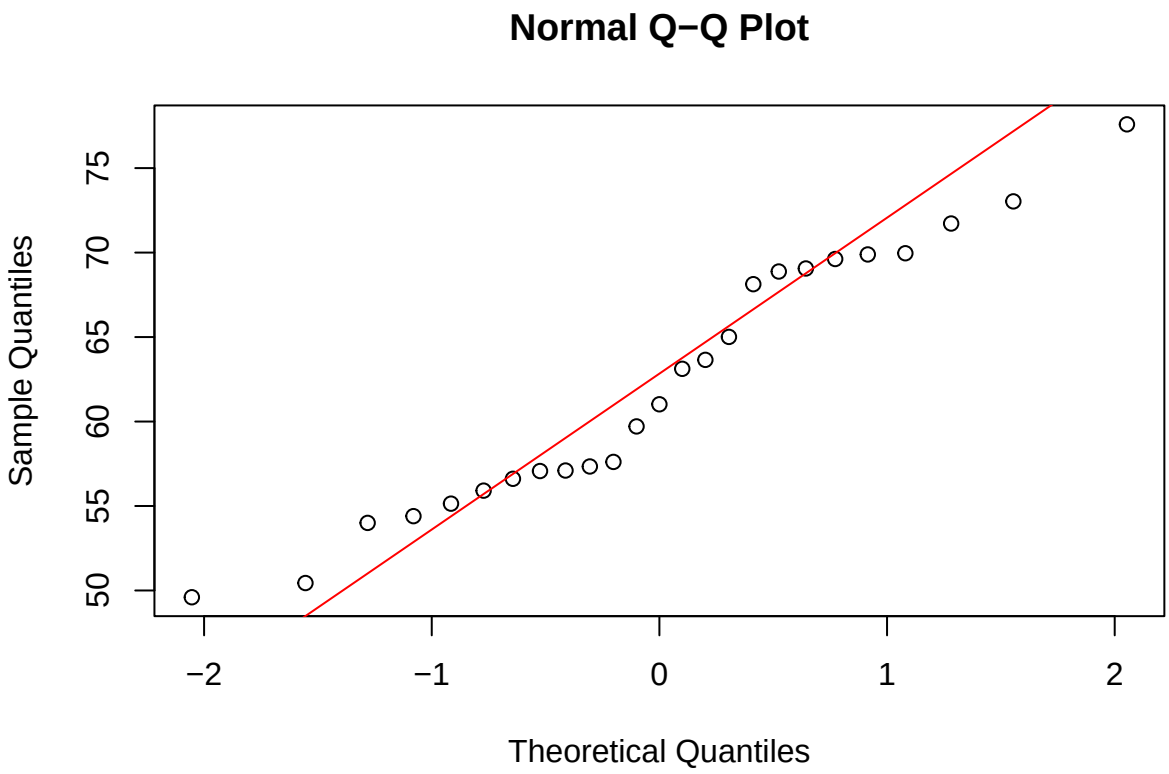
Q-Q plot for all age data

```
qqnorm(data$ages)
qqline(data$ages, col = "red")
```



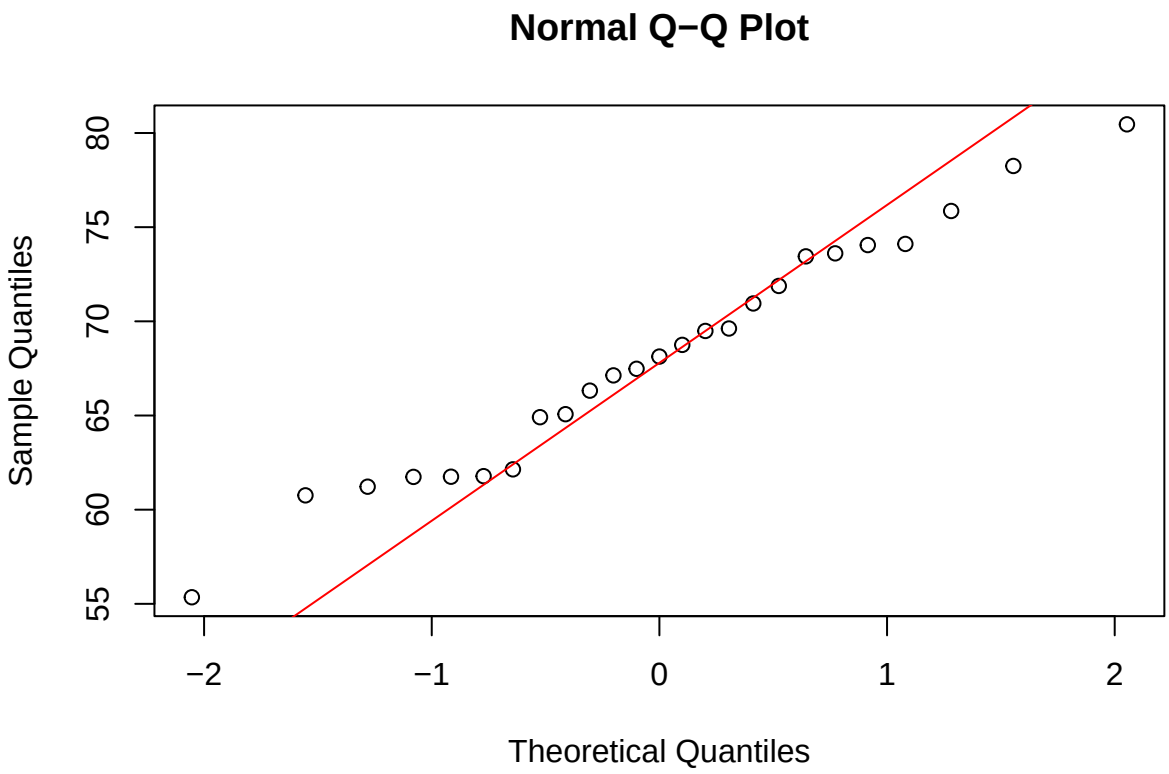
Q-Q plot for LUAD (LUng ADenocarcinoma) ages

```r
qqnorm(LUAD_data$ages)
qqline(LUAD_data$ages, col = "red")
```



Q-Q plot for LUSC (LUng Squamous Cell Carcinoma) ages

```
qqnorm(LUSC_data$ages)
qqline(LUSC_data$ages, col = "red")
```



3e (8 points)

Answer the following questions:

What is the null hypothesis for the shapiro test? Can you reject it for the age distribution of the 50 patients? How about for the ages within each of the tumor types (i.e., just in adenocarcinoma, or just in squamous cell carcinoma)?

For the entire age data (p-value = 0.5582), with a significance level of 0.05, since the p-value is greater than 0.05, there is insufficient evidence to reject the null hypothesis. The data for all 50 patients' ages does not significantly deviate from a normal distribution.

For LUAD ages (p-value = 0.2614), again, with a significance level of 0.05, the p-value is greater than 0.05, indicating insufficient evidence to reject the null hypothesis. The data for LUAD patients' ages does not significantly deviate from a normal distribution.

For LUSC ages (p-value = 0.8775), the p-value is much greater than 0.05. Therefore, there is no significant evidence to reject the null hypothesis. The data for LUSC patients' ages does not significantly deviate from a normal distribution.

Based on the p-values, there is no strong evidence to suggest that the age distributions of either the entire set of patients, LUAD patients, or LUSC patients significantly deviate from a normal distribution.

Write answers here. Given both the shapiro test and qqplot results, what do you conclude about the suitability of a t-test for comparing patient ages across the two tumor types?

In assessing patient age comparison within adenocarcinoma and squamous cell carcinoma, both Shapiro-Wilk tests and QQ plots suggest a reasonable approximation to normality for age data within each tumor type. This supports the use of a t-test for comparing ages between the two tumor types. However, caution is warranted due to the relatively small dataset (50 observations), which might impact the reliability of statistical analyses and result in uncertain findings.

3f (6 points) Regardless of the outcome above, use a t-test to compare the ages of patients with adenocarcinoma to those with squamous cell carcinoma.

```
t_test_result <- t.test(LUAD_data$ages, LUSC_data$ages)

print(t_test_result)
```

```
##
##  Welch Two Sample t-test
##
## data:  LUAD_data$ages and LUSC_data$ages
## t = -3.0208, df = 45.959, p-value = 0.004108
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -9.908133 -1.983867
## sample estimates:
## mean of x mean of y
##   62.2244   68.1704
```

Then, answer these questions:

What is the null hypothesis for this test? Can you reject it? What do you conclude?

The results of the Welch's two-sample t-test indicate a comparison of ages between patients with adenocarcinoma (LUAD) and squamous cell carcinoma (LUSC). According to the test results, the null hypothesis, suggesting no difference in the average ages between the two groups, was rejected. The p-value of 0.004108, being less than the typical significance level of 0.05, indicates a significant difference in the mean ages of patients with adenocarcinoma and squamous cell carcinoma. Specifically, the mean age of patients with adenocarcinoma is 62.22 years, significantly lower than the mean age of patients with squamous cell carcinoma at 68.17 years. Therefore, based on these statistical findings, it can be concluded that there is a significant difference in ages between LUAD and LUSC patients.

3g (6 points) Reformulate the above t-test as a linear model (as in notes from 10/23/23). Show the linear model here:

```
lm_age <- lm(ages ~ group, data = rbind(data.frame(ages = LUAD_data$ages, group = "LUAD"),
                                        data.frame(ages = LUSC_data$ages, group = "LUSC")))
 summary(lm_age)
```

```
##
## Call:
## lm(formula = ages ~ group, data = rbind(data.frame(ages = LUAD_data$ages,
##     group = "LUAD"), data.frame(ages = LUSC_data$ages, group = "LUSC")))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.8204  -5.9264  -0.3654   5.8991  15.3656
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   62.224      1.392  44.707  < 2e-16 ***
## groupLUSC      5.946      1.968   3.021  0.00403 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 6.959 on 48 degrees of freedom
## Multiple R-squared:  0.1597, Adjusted R-squared:  0.1422
## F-statistic: 9.125 on 1 and 48 DF,  p-value: 0.004033
```

What are the identified group means, and how do you compute them from the slope and intercept of the model?

The estimated mean age for LUAD is 62.224. Calculated by adding the 'groupLUSC' coefficient to the intercept, the estimated mean age for LUSC is 68.170. Mean age of LUSC = Intercept (LUAD) + Coefficient (groupLUSC) = 62.224 + 5.946 = 68.170 These identified group means signify that the average age of patients with LUSC is approximately 5.946 years higher than those with LUAD, according to the linear model analysis.

Question 4: Low platelet counts, regression modeling

"Thrombocytopenia" refers to having a low platelet count. Such patients are, in severe cases, at risk of life-threatening bleeding events. Platelet counts normally range from 150,000 to 450,000 (although numbers reported here and elsewhere are typically divided by 1000 and rounded, so the normal range would be 150-450). Patients with counts between 100 and 150 are said to have mild thrombocytopenia; those between 50-100 are said to have moderate cases; and those below 50 are called severe.

Measuring platelet counts in lung cancer patients is important: the cancer itself can cause thrombocytopenia, but so can some treatments. Oddly, in some cases, severity of thrombocytopenic reactions to treatment correlates with overall survival, although pre-treatment severity tends to be a poor prognostic factor.

4a (6 points) We will use the data in clindata.rds in linear regression models to try to predict the platelet count.

One might ask why we can use linear regression to predict "count" data, such as platelet counts, when technically this is discrete rather than continuous data. However, when count data can take on many possible values, it can typically be co nsidered as essentially continuous.

First, build a regression model called `pllma` that predicts platelet levels using just ages as an independent variable. Then, build a linear regression model, called `pllm1`, that predicts platelets using only the expression data from the gene IL4 as an independent variable.

Run `summary( )` on each model to see data evaluating the model fit.

```
data <- readRDS("clindata.rds")
pllma <- lm(platelets ~ ages, data = data)
summary(pllma)
```

```
##
## Call:
## lm(formula = platelets ~ ages, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -150.45  -81.96  -23.57   59.16  267.89
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  301.572    138.114   2.183   0.0339 *
## ages          -1.849      2.105  -0.879   0.3840
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 110.7 on 48 degrees of freedom
## Multiple R-squared:  0.01582,    Adjusted R-squared:  -0.004679
## F-statistic: 0.7718 on 1 and 48 DF,  p-value: 0.384
```

```
pllm1 <- lm(platelets ~ IL4, data = data)
summary(pllm1)
```

```
##
## Call:
## lm(formula = platelets ~ IL4, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -172.260  -78.532   -4.478   40.079  257.083
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 394.7956    54.3331   7.266 2.88e-09 ***
## IL4          -2.9537     0.7268  -4.064 0.000178 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 96.25 on 48 degrees of freedom
## Multiple R-squared:  0.256,  Adjusted R-squared:  0.2405
## F-statistic: 16.52 on 1 and 48 DF,  p-value: 0.0001779
```

4b (4 points)

The R-squared value measures the proportion of the variation in the dependent variable that can be explained by the model. This value ranges between 0 and 1.

To be able to compare across models with different numbers of variables, one should use the *adjusted R-squared* value, which accounts for the number of degrees of freedom.

Which variable appears to capture more of the variation in platelet count, ages or IL4?

Does either model appear to do a good job of predicting patients' platelet counts, in the sense of explaining most of the variability in platelet counts by the model?

The model with 'ages' as the independent variable resulted in a negative adjusted R-squared, which implies that adding 'ages' as a predictor in the model did not improve the fit and the model might be worse than just using the mean to predict platelet counts. However, the model with 'IL4' as the independent variable has a positive adjusted R-squared value of 0.2405, indicating that it explains about 24.05% of the variability in platelet counts. Therefore, the 'IL4' variable appears to capture more of the variation in platelet count compared to 'ages'. However, neither model seems to do a good job of predicting patients' platelet counts exceptionally well, as the adjusted R-squared values are relatively low. The model with 'IL4' as the independent variable explains more variability, but there is still a substantial portion of platelet count variation unexplained by this model.

4c (6 points) Next, build a multiple linear regression model called `pllm8` that predicts platelets using all 8 genes' expression values as the independent variables.

Recall that to build a model that uses all variables in a data frame `foo.df` *except* the dependent variable `dep`, you can use the shorthand `lm(dep ~ ., data=foo.df)` to avoid typing all of the variable names. To

do this, however, you'd need to create a data frame or tibble that includes only the variables you want to have in the model.

By looking at the t-scores and significance levels associated with the model, which genes' expression values (if any) appear to be significantly associated with platelet count?

```
gene_data <- data[, -which(names(data) %in% c("ages", "samples"))]
pllm8 <- lm(platelets ~ ., data = gene_data)
summary(pllm8)
```

```
##
## Call:
## lm(formula = platelets ~ ., data = gene_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -150.541  -38.013   -2.903   43.065  168.854
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 486.2559   197.8583   2.458   0.0183 *
## IL4          -0.4256     0.8169  -0.521   0.6052
## IL10          7.7668     5.4446   1.427   0.1613
## CRISP3       -0.1642     0.3929  -0.418   0.6782
## MMP8          3.0782     3.2098   0.959   0.3432
## PDL5         -1.7704     1.0923  -1.621   0.1127
## SLC1A3       -1.2349     0.6837  -1.806   0.0782 .
## TNFA         -4.7004     1.8421  -2.552   0.0145 *
## IFNG          0.7909     2.2835   0.346   0.7308
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 74.54 on 41 degrees of freedom
## Multiple R-squared:  0.6189, Adjusted R-squared:  0.5445
## F-statistic: 8.322 on 8 and 41 DF,  p-value: 1.265e-06
```

Based on the t-scores and significance levels presented in the model summary, the genes' expression values significantly associated with platelet count are TNFA. Genes with p-values less than 0.05 are usually considered statistically significant in their association with the platelet count. In this case, TNFA has a p-value of 0.0145, suggesting a significant association with platelet count.