Lesson 5 Data Visualization with R

Outline

- Types of Data
- Creating data visualization using base graphics system
  - Visualizing one categorical variable
  - Visualizing one numeric variable
  - Visualizing two categorical variables
  - Visualizing two numeric variables
  - Visualizing both a categorical and a numeric variable
- Creating data visualization using ggplot2

Outline

- Types of Data
- Creating data visualization using base graphics system
  - Visualizing one categorical variable
  - Visualizing one numeric variable
  - Visualizing two categorical variables
  - Visualizing two numeric variables
  - Visualizing both a categorical and a numeric variable
- Creating data visualization using ggplot2

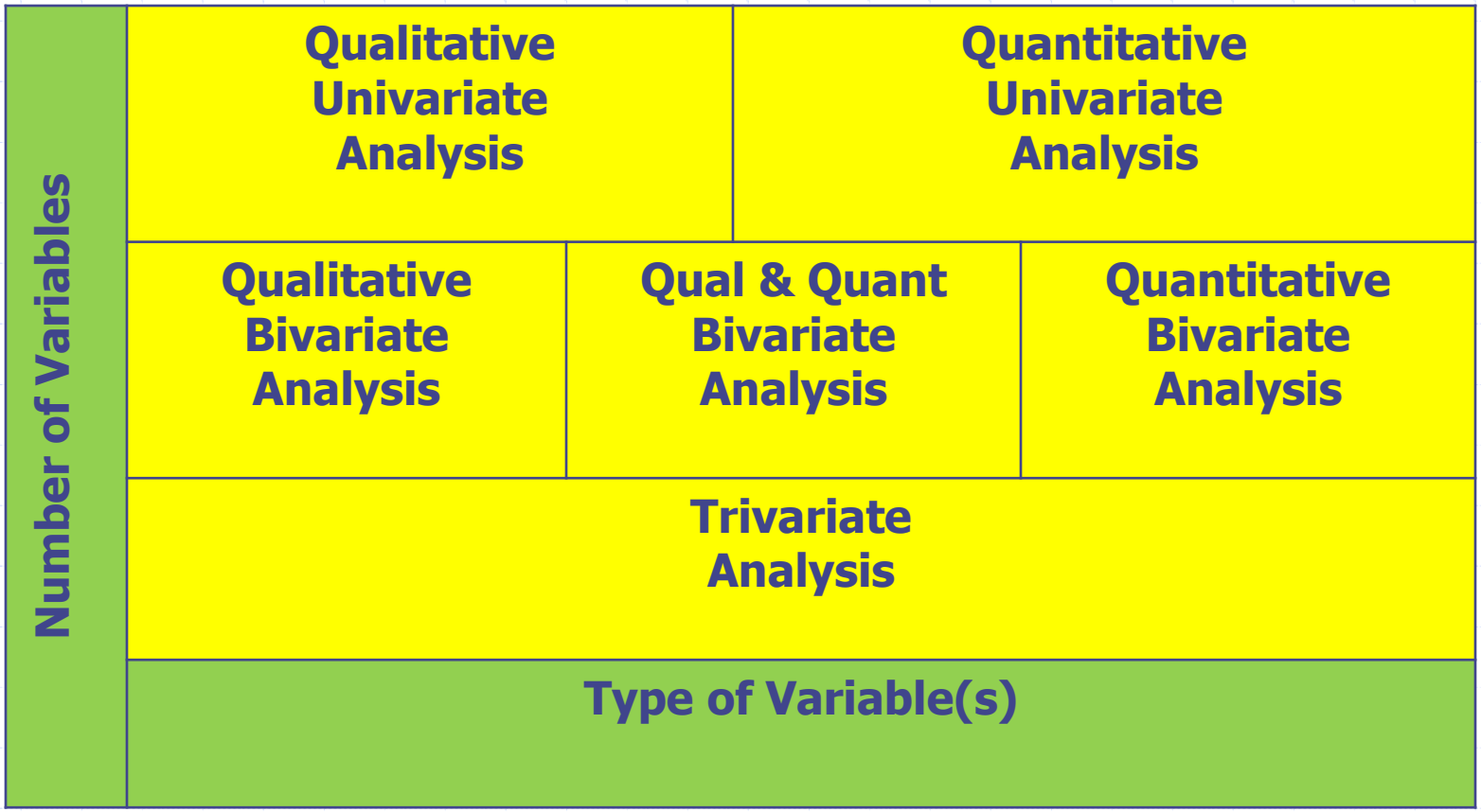
Types of Data

- Data is oftentimes classified into one of two types:
  - Qualitative data (categorical data): associated with a property or a quality
  - Quantitative data (numeric data): associated with a numeric measurement

Exercise

- What type of data is race (i.e. Caucasian, African American, Asian, Hispanic, etc.)?
- What type of data is bmi category (i.e. obese, overweight, normal weight, underweight)?
- What type of data is weight (in lbs)?

Types of Data Analysis



Titanic data

Data Dictionary

Variable	Definition	Key
survival	Survival	0=No, 1=Yes
pclass	Ticket Class	1=1st, 2 = 2nd, 3 = 3rd
sex	Sex	Female, male
age	Age in years	
sibsp	#of siblings/spouses	
parch	# of parents/children	
ticket	Ticket Number	
fare	Passenger fare	
cabin	Cabin number	
…	…	

Summarizing Categorical Data

- To summarize qualitative (categorical) data tables are used. Suppose we want to summarize the survival variable:

```
> table(data$survived)
  0   1
809 500
```

- Here we use the table() command to generate frequencies of qualitative data.

Summarizing Categorical Data

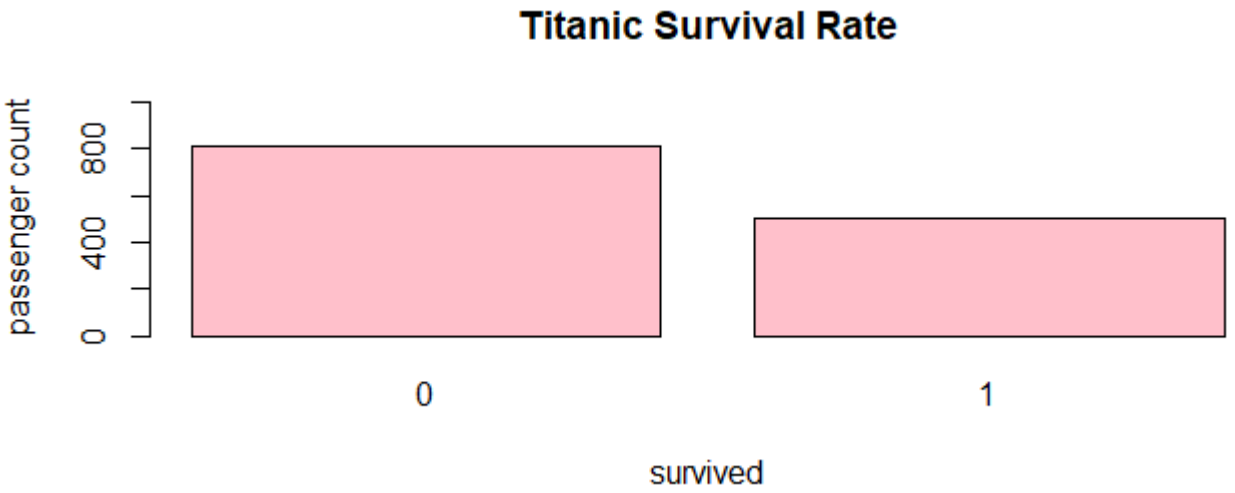
- We can also use the table() command and the prop.table() command to generate proportions for categorical data.

```
> table(data$survived) / length(data$survived)

       0        1
0.618029 0.381971

> prop.table(table(data$survived))

       0        1
0.618029 0.381971
```

Graphical Summary of Categorical Data

- The frequencies of categorical data can be graphically represented in a bar plot (also called a bar chart)

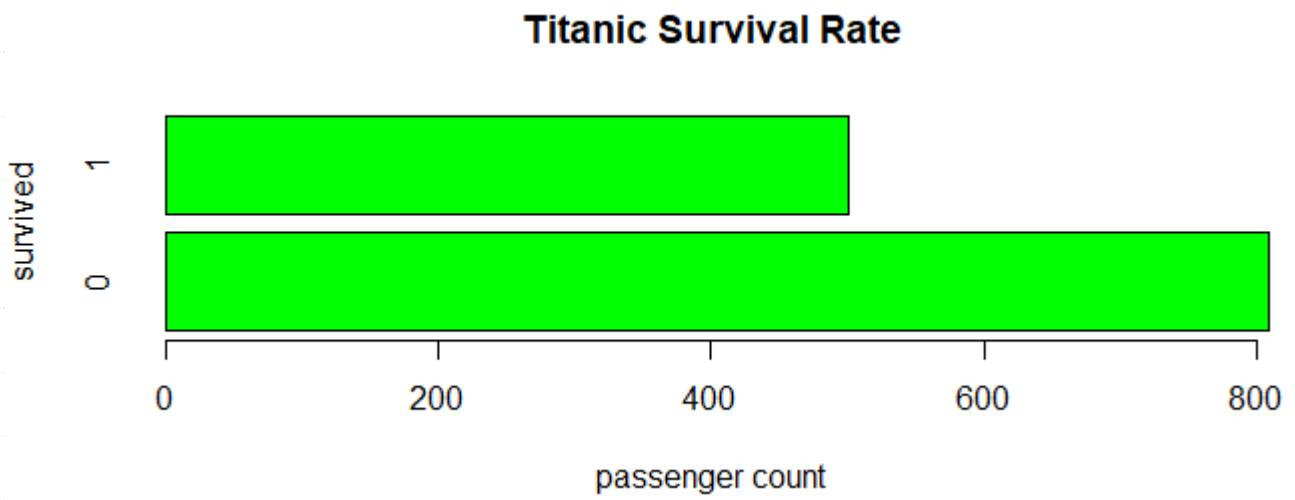
```
barplot(table(data$survived),
        main = "Titanic Survival Rate",
        col = "pink",
        ylim = c(0, 1000),
        xlab = "survived",
        ylab = "passenger count")
```



Graphical Summary of Categorical Data

- By default, R creates a vertical bar plot, you can add the "horiz=TRUE" option to create a horizontal bar plot.

```
barplot(table(data$survived),
        horiz = TRUE,
        main = "Titanic Survival Rate",
        col = "green",
        ylab = "survived",
        xlab = "passenger count")
```



Graphical Summary of Categorical Data

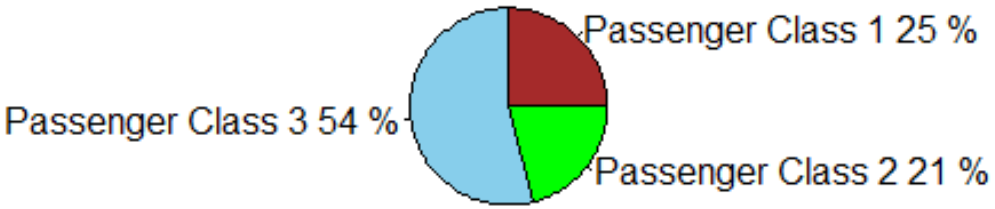
- Bar plots can also be created using plot.

```
plot(x = data$survived,
    main = "Titanic Survival Rate",
    col = "pink",
    xlab = "survived",
    ylab = "passenger count")

plot(x = data$survived,
    horiz = TRUE,
    main = "Titanic Survival Rate",
    col = "pink",
    ylab = "survived",
    xlab = "passenger count")
```

Graphical Summary of Categorical Data

- A pie chart can also be used to display categorical data percentages.

```
lables <- names(table(data$pclass))
percentage <- round(table(data$pclass) / length(data$pclass)*100)
newlables <- paste("Passenger Class", lables, percentage, "%")
pie(table(data$pclass),
    labels = newlables,
    clockwise = TRUE,
    col = c("brown","green", "skyblue"))
```



Summarizing Quantitative Data

- We can summarize quantitative data using the mean, median, standard deviation, quartiles/interquartile range.
- The following commands can be use to generate descriptive statistics for quantitative data:
  - mean()
  - median()
  - min() or max()
  - sd()
  - var()
  - range()
  - summary()

Summarizing Quantitative Data

- When there are missing values for a variable (we have a lot in this dataset), R will by default give the result "NA". To work around this, we need to use the "na.rm=TRUE" option.

```
mean(data$age, na.rm = TRUE)
median(data$age, na.rm = TRUE)
min(data$age, na.rm = TRUE)
sd(data$age, na.rm = TRUE)
var(data$age, na.rm = TRUE)
range(data$age, na.rm = TRUE)
summary(data$age)
```

Summarizing Quantitative Data

- The quantile() function can be used to directly compute percentiles.
- The IQR() function can be used to compute the interquartile range (Q3 – Q1)

```
> quantile(data$age, c(0,0.25,0.5,0.75,1), na.rm = TRUE)
   0%   25%   50%   75%  100%
 0.17 21.00 28.00 39.00 80.00
> IQR(data$age, na.rm = TRUE)
[1] 18
```

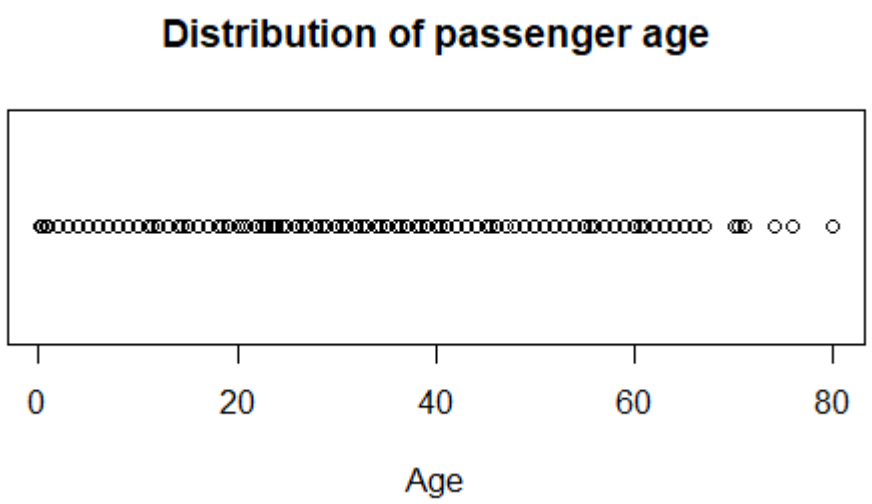
Graphically Displaying Quantitative Data

- Below are ways of graphically displaying quantitative data.
  - Dot plot
  - Jitter plot
  - Box plot (box and whisker plots)
  - Histogram
  - Density plot

Graphically Displaying Quantitative Data

1. Dot Plot

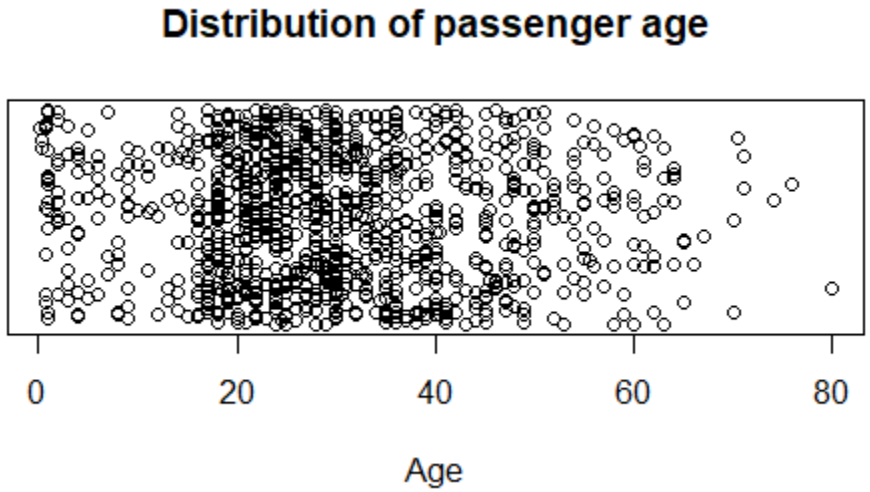
```
plot(x = data$age,
     y = rep(0, nrow(data)), #repeat 0, nrow(data) times
     main = "Distribution of passenger age",
     xlab = "Age",
     ylab = "", #no y label necessary
     yaxt = "n") #to suppress rendering of the y-axis text
```



Graphically Displaying Quantitative Data

2. Jitter Plot

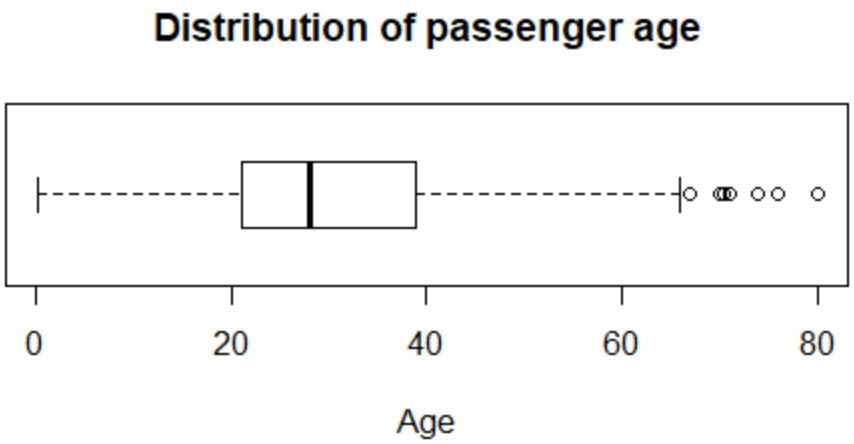
```
plot(x = data$age,
     y = jitter(rep(0, nrow(data))),
     main = "Distribution of passenger age"),
     xlab = "Age",
     ylab = "", #no y label necessary
     yaxt = "n") #to suppress rendering of the y-axis text
```



Graphically Displaying Quantitative Data

3. Box Plot: Boxplot provides the five number summary statistics and outliers.

```
boxplot(x = data$age,
    horizontal = TRUE,
    main = "Distribution of passenger age",
    xlab = "Age")
```

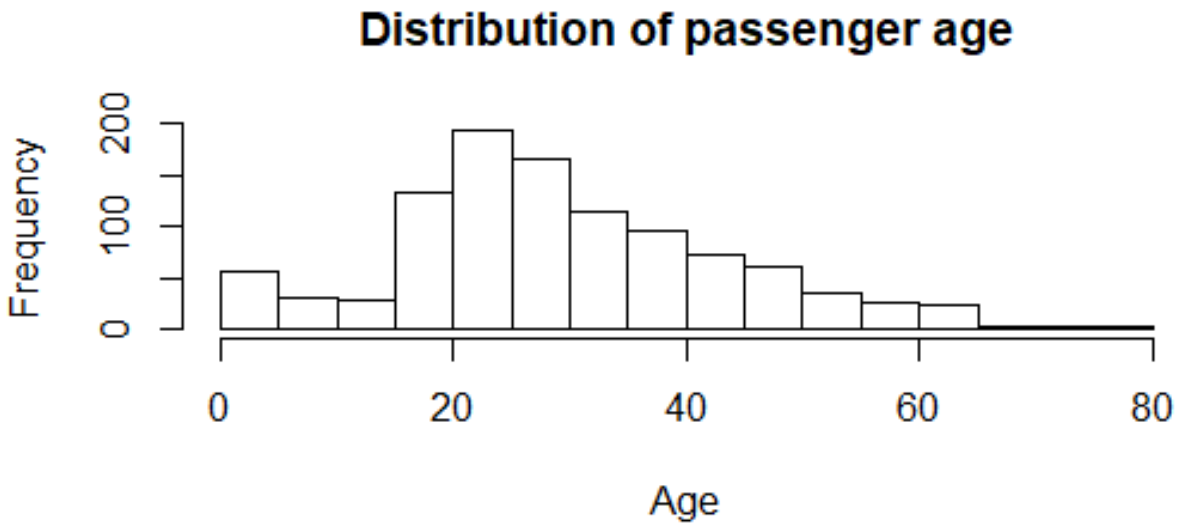


Box Plot shows min, Q1, Q2, Q3, max and outliers. (outliers > Q3+1.5 * IQR or < Q1 – 1.5 * IQR)

Graphically Displaying Quantitative Data

4. Histogram

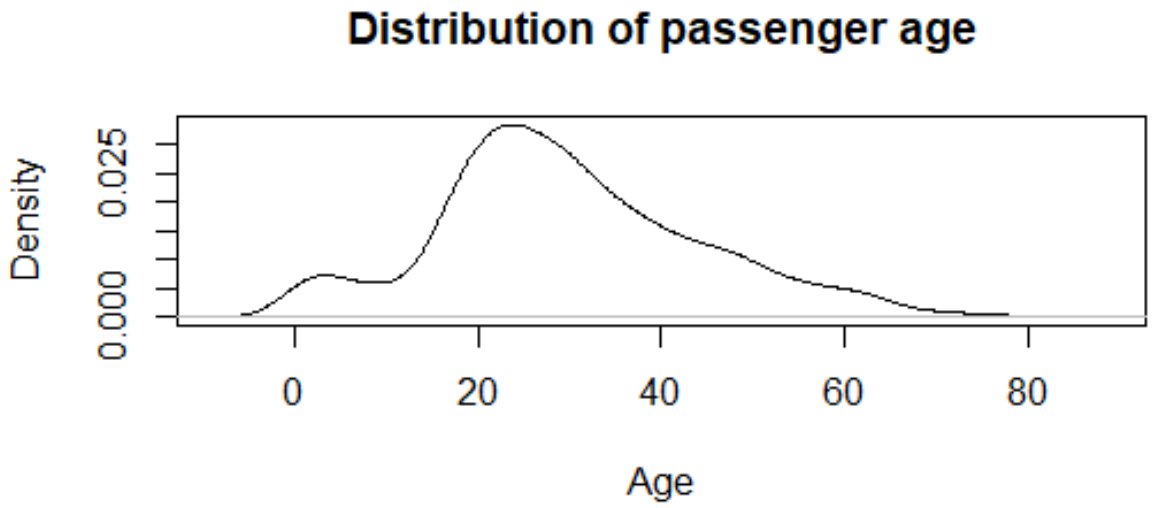
```
hist(x = data$age,
    breaks = 15, #of bins, default 20
    main = "Distribution of passenger age",
    xlab = "Age")
```



Graphically Displaying Quantitative Data

5. Density Plot

```
plot(x = density(na.omit(data$age)), #omit missing age values
     main = "Distribution of passenger age",
     xlab = "Age")
```



Bivariate Data - two categorical variables

- Bivariate data analysis examines the relationship between two variables.
- Suppose we want to know the survival rate by gender for the titanic data.
- To do this, we would want to create a contingency table of the data.

Bivariate Data - two categorical variables

- To create a contingency table we can use the table() command. Note that the first variable specifies the rows, the second variable specifies the columns.

```
> table(data$sex, data$survived)

            0   1
female    127 339
male      682 161
```

Bivariate Data - two categorical variables

- Row names and column names can be added to a contingency table to make interpretation easier.

```
> x <- table(data$sex, data$survived)
> colnames(x) <- c("perished", "survived")
> x

        perished survived
female       127      339
male         682      161
```

- Question: What is the probability of the females who survived among all the passengers?

Bivariate Data - two categorical variables

- In R, the prop.table() command can be used to compute proportions.

```
> prop.table(x)

          not survived   survived
female     0.09702063 0.25897632
male       0.52100840 0.12299465
```

- Example interpretation: "Among all the passengers, 52% are male and did not survive. "

Bivariate Data - two categorical variables

- To get the proportions by row or by column we can use the prop.table() function again.

```
> prop.table(x, 1)

        not survived  survived
female     0.2725322 0.7274678
male       0.8090154 0.1909846
> prop.table(x, 2)

        not survived  survived
female     0.1569839 0.6780000
male       0.8430161 0.3220000
```

proportions by row

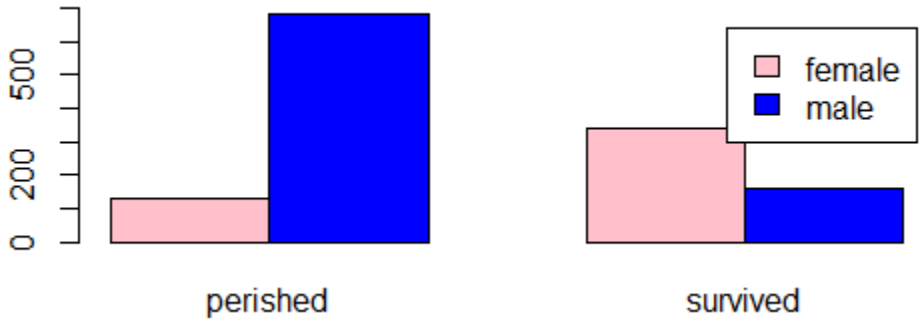
proportions by column

Graphically Displaying Bivariate Qualitative Data

- Below are ways of graphically displaying bivariate qualitative data.
  - grouped frequency bar chart
  - stacked frequency bar chart
  - spine plot
  - mosaic plot

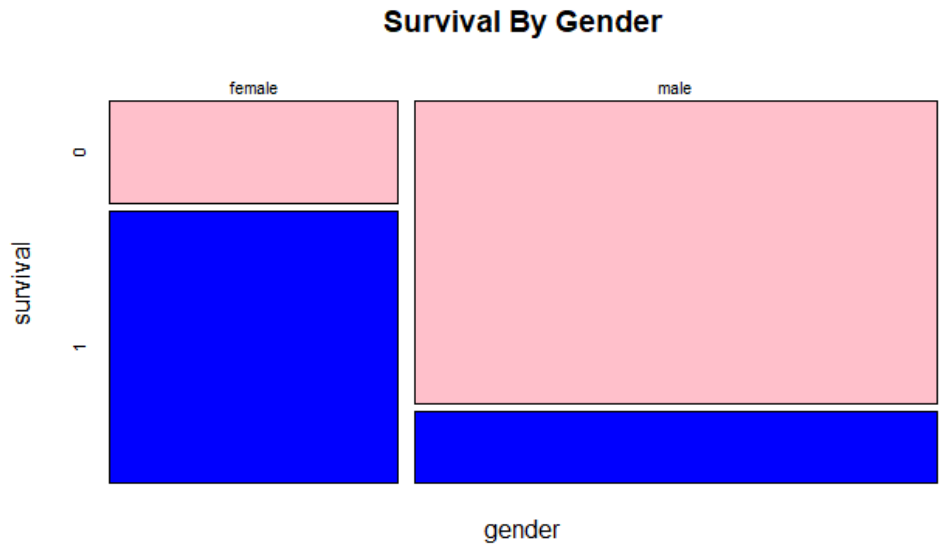
An Example: Grouped Frequency Bar Chart

```
barplot(table(data$sex, data$survived),
        ylim = c(0, 700),
        col=c("pink", "blue"),
      beside=TRUE, #the bars will be side by side
      legend.text=TRUE) #add the legend
```



An Example: Mosaic Plot

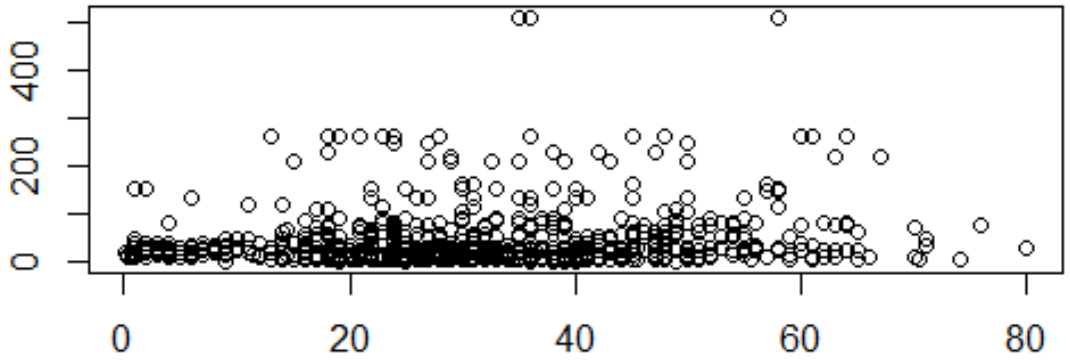
```
mosaicplot(table(data$sex, data$survived),
           main = "Survival By Gender",
           xlab ="gender",
           ylab="survival",
           color = c("pink", "blue"))
```



Graphically Displaying Bivariate Quantitative Data

- Thus far we have been examining bivariate relationships with categorical data. Now we will explore bivariate relationships with numeric data.
- A scatterplot is typically used to examine the relationship between two numeric variables. These can be generated with plot() command.

```
plot(data$age, data$fare)
```



Graphically Displaying Bivariate Quantitative Data

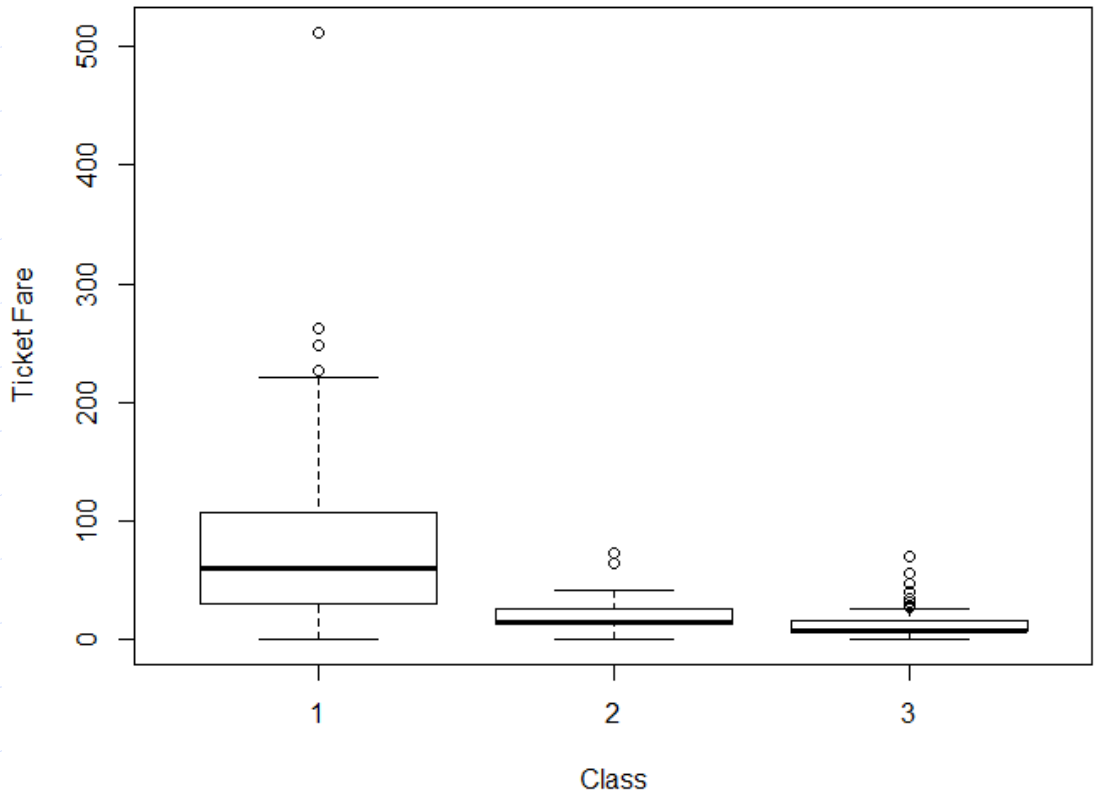
- Other plots for displaying bivariate quantitative data are
  - binned frequency Heatmap
  - contour plot
  - level plot
  - mesh plot
  - surface plot
  - step chart
  - line chart
  - area chart

Graphically Displaying Bivariate Categorical and Numeric data

- Plots for displaying bivariate categorical and numeric data are
  - bivariate bar chart
  - bivariate box plot
  - bivariate violin plot

An Example: bivariate box plot

```
boxplot(data$fare ~ data$pclass,
        xlab = "Class",
        ylab = "Ticket
```



Outline

- Types of Data
- Creating data visualization using base graphics system
  - Visualizing one categorical variable
  - Visualizing one numeric variable
  - Visualizing two categorical variables
  - Visualizing two numeric variables
  - Visualizing both a categorical and a numeric variable
- Creating data visualization using ggplot2

Introduction to ggplot2

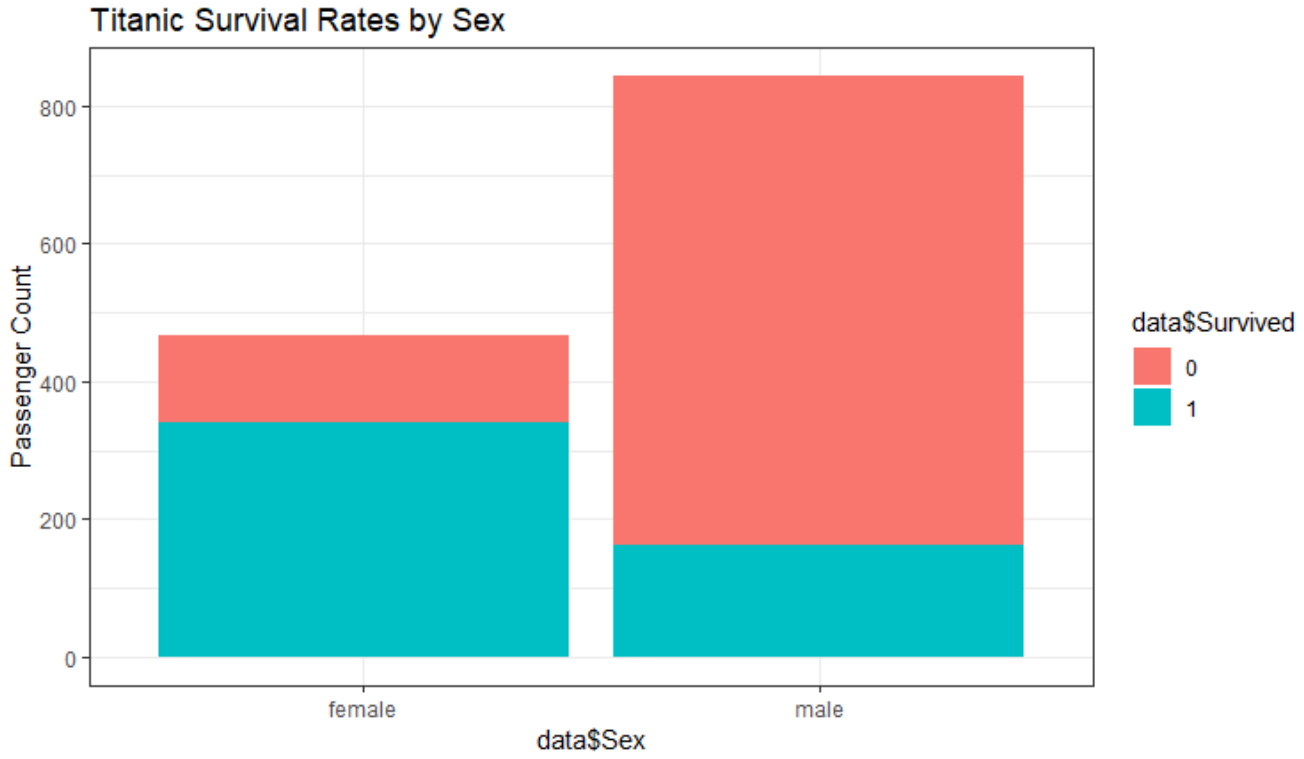
- Standard visualization package in R
  - install.packages("ggplot2")
  - library(ggplot2)
- Elegant data visualization using the grammar of graphics.
- Designed for print-quality graphics in seconds.

Introduction to ggplot2

- Each ggplot2 visualization has three required components
  - Data: the raw material of your visualization. (type of data frame)
  - Aesthetics: the mapping of your data to the visualization. For example, mapping the value of Titanic passenger ages to the y-axis of a graph.
  - Layers: what you see on the plots (e.g. points, lines, etc.) These layers typically take the form of a ggplot2 geom function.
- Example:
  - ggplot(data, aes(x = data$Survived)) + geom_bar()

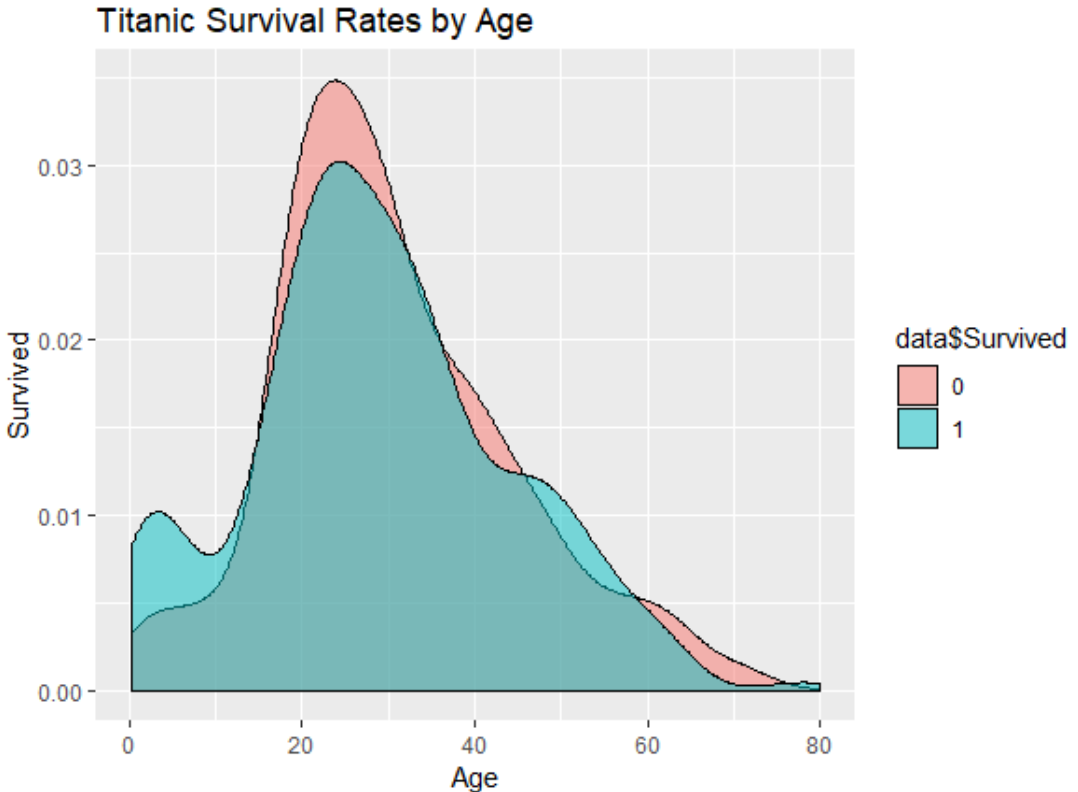
An Example

```
ggplot(data, aes(x = data$Sex, fill = data$Survived)) +
  theme_bw() +
  geom_bar() +
  labs(y = "Passenger Count", title = "Titanic Survival Rates by Sex")
```



Another Example

```
ggplot(data, aes(x=data$age, fill=data$Survived)) +
  geom_density(alpha=0.5) +
  labs(x = "Age",  y = "Survived",
       title = "Titanic Survival Rates by Age")
```



Interesting Example of Data Visualizations

- Hans Rosling: Master of Data Visualizations - ED talk on public health and longevity:
  https://www.youtube.com/watch?v=hVimVzgtD6w