R Notebook

Pradip Basnet

#Categorical

isntalling and loading the necessary packages

```
#install.packages("scales")
library(scales)
```

```
library(mosaicData)
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

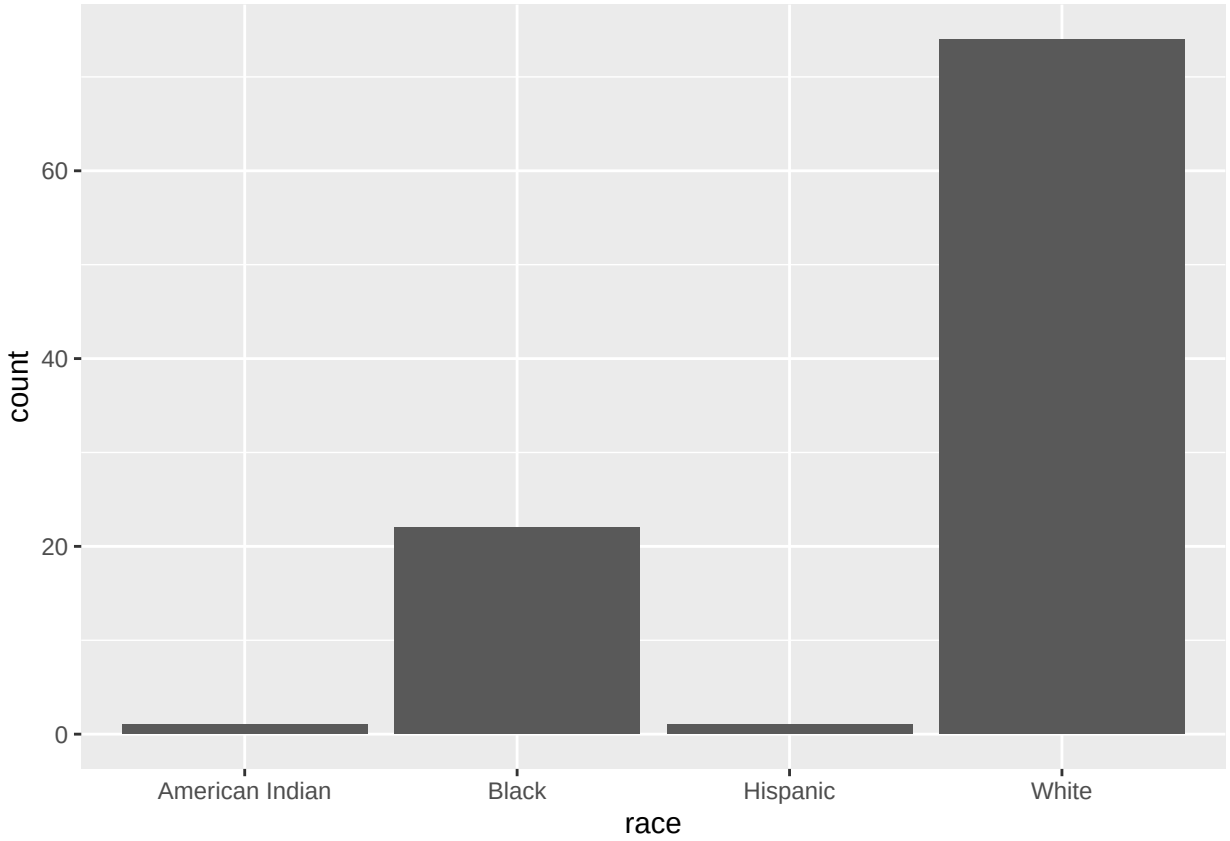
loading the dataset Marriage

```
data(Marriage, package="mosaicData")
head(data)
```

```
##
## 1 function (..., list = character(), package = NULL, lib.loc = NULL,
## 2     verbose = getOption("verbose"), envir = .GlobalEnv, overwrite = TRUE)
## 3 {
## 4     fileExt <- function(x) {
## 5         db <- grepl("\\\\.[^.]+\\\\.(gz|bz2|xz)$", x)
## 6         ans <- sub(".*\\\\.", "", x)
```

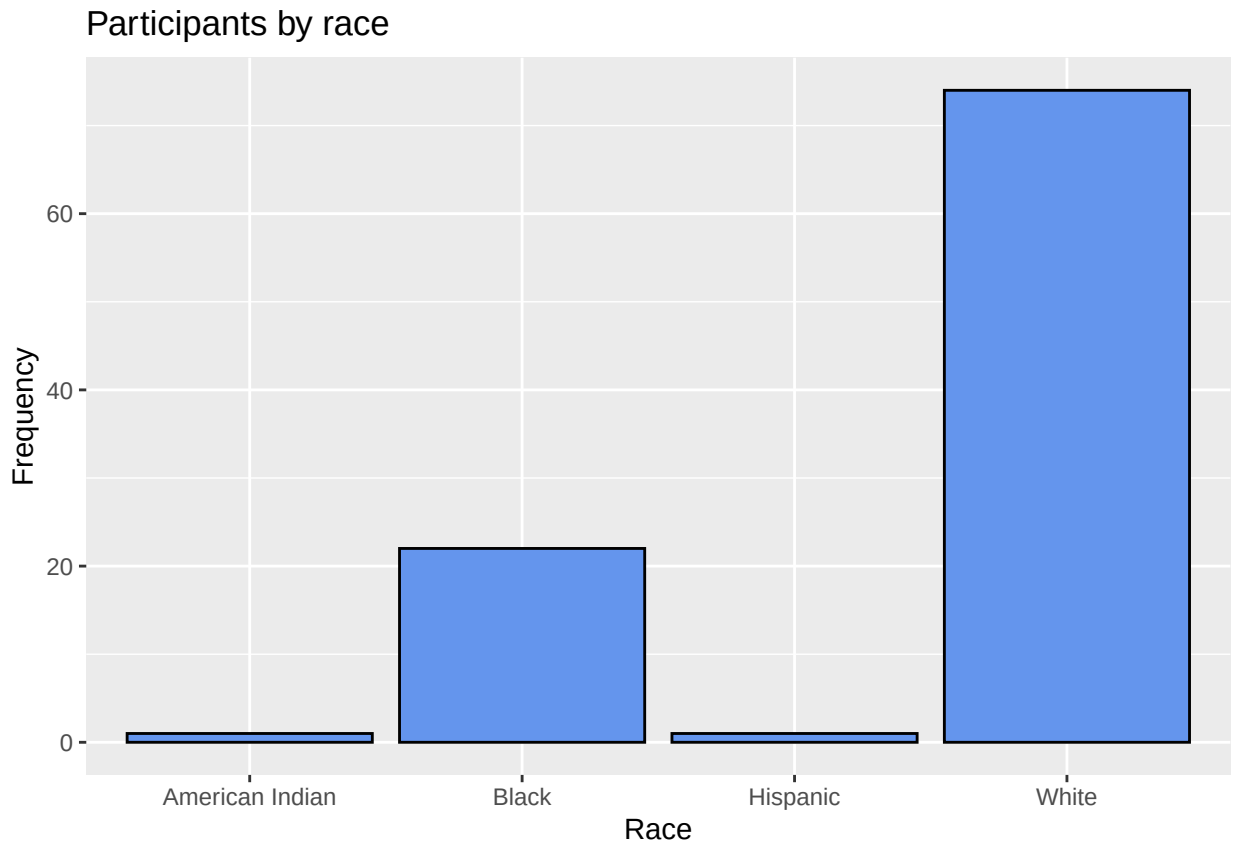
plot the distribution of the race.The bar plot indicates a significant disparity in participant racial distribution, with the White population comprising the majority, followed by Black participants. Representation from American Indian and Hispanic groups is substantially lower compared to the other two groups

```
ggplot(Marriage, aes(x=race)) + geom_bar()
```



plot the distribution of race with modified colors and labels.The bar plot indicates a significant disparity in participant racial distribution, with the White population comprising the majority, followed by Black participants. Representation from American Indian and Hispanic groups is substantially lower compared to the other two groups

```
ggplot(Marriage, aes(x=race))+
  geom_bar(fill="cornflowerblue",color="black")+
  labs(x="Race", y="Frequency",title="Participants by race")
```

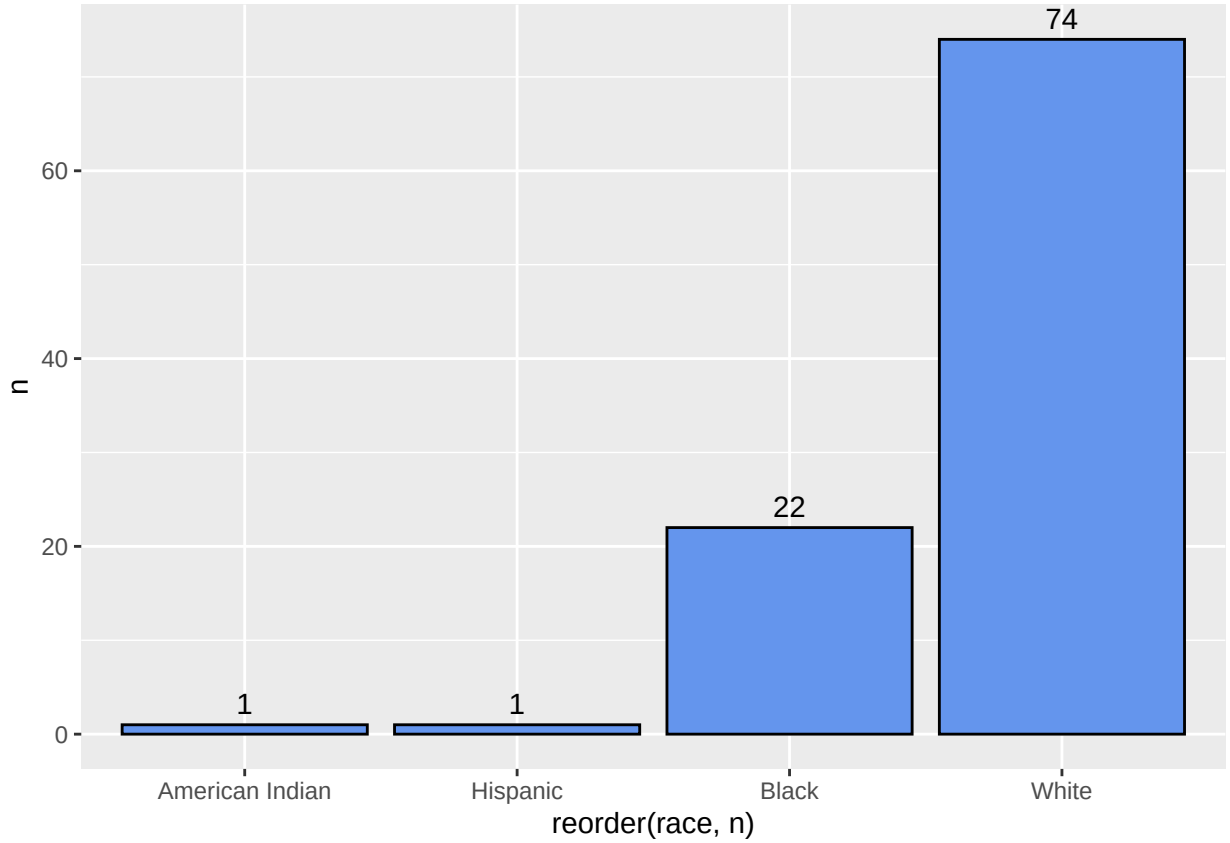


#Sorting categories

```
#calculate number of participants in each race
library(dplyr)
plotdata <- Marriage %>%
  count(race)
```

the code create bar plot, where the x-axis represents different races (reordered by frequency), and the y-axis shows their corresponding frequencies from the plotdata dataframe. The bars are colored "cornflowerblue" with black borders, and the heights of the bars directly reflect the frequency values. Additionally, the plot includes text labels displaying these frequency values above each bar. The x-axis is labeled "Race", the y-axis "Frequency", and the plot is titled "Participants by race The bar plot illustrates a substantial disparity in the racial distribution of participants, with the White population constituting the majority, followed by Black participants. Representation from American Indian and Hispanic groups is markedly lower compared to the other two groups.

```
#now plotting the sorted bars
ggplot(plotdata, aes(x=reorder(race, n), y=n)) +
  geom_bar(stat = "identity", fill="cornflowerblue", color="black") +
  geom_text(aes(label=n), vjust=-0.5)
```



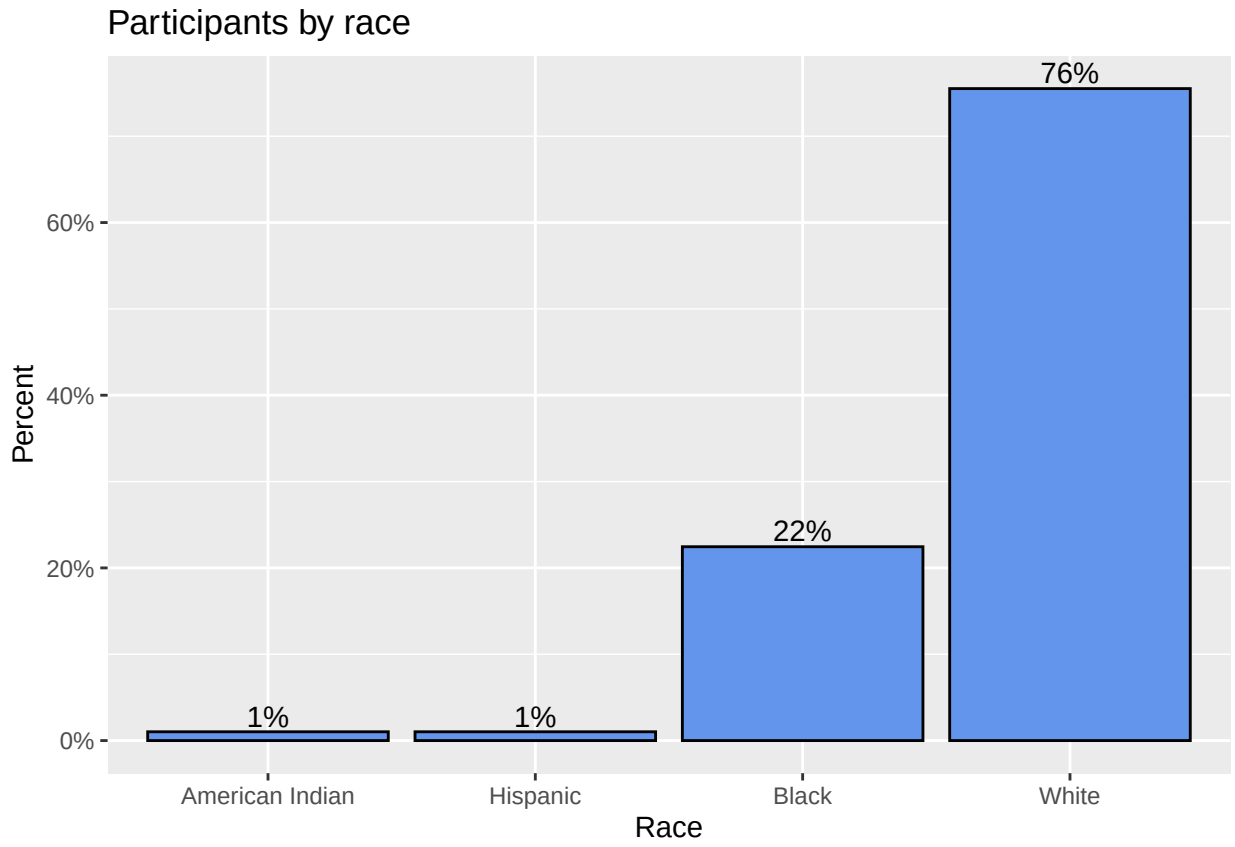

```
  labs(x = "Race",
       y = "Frequency",
       title  = "Participants by race")
```

```
## $x
## [1] "Race"
##
## $y
## [1] "Frequency"
##
## $title
## [1] "Participants by race"
##
## attr(,"class")
## [1] "labels"
```

##Percents code calculates the percentage distribution of different races in the Marriage dataset and creates a bar chart to visualize it. The chart displays bars representing each race's percentage of the total, with bars colored in "cornflowerblue" and outlined in black. Percentage labels are shown above each bar. The y-axis is formatted to display percentages, and the x-axis and y-axis are labeled "Race" and "Percent," respectively, with the chart titled "Participants by race

```
plotdata <- Marriage %>%
  count(race) %>%
  mutate(pct = n / sum(n),
```

```
         pctlabel = paste0(round(pct * 100), "%"))

ggplot(plotdata,
       aes(x = reorder(race, pct), y = pct)) +
  geom_bar(stat = "identity", fill = "cornflowerblue", color = "black") +
  geom_text(aes(label = pctlabel), vjust = -0.25) +
  scale_y_continuous(labels = scales::percent_format(accuracy = 1)) +
  labs(x = "Race",
       y = "Percent",
       title = "Participants by race")
```

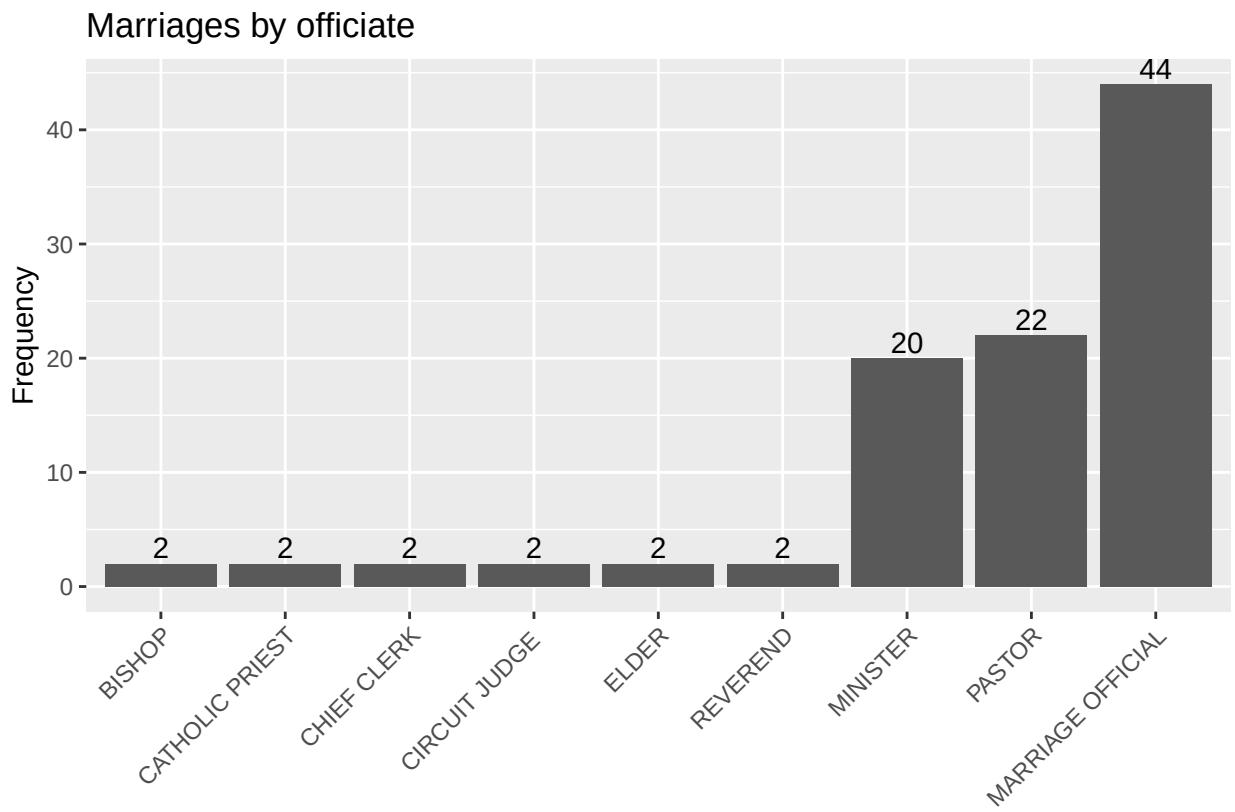


##Overlapping labels

first calculates the frequencies of each officialTitle in the Marriage dataset and stores the results in plotdata. It then creates a bar chart where the x-axis displays these officialTitle values, reordered by frequency, and the y-axis shows the counts. Bars are plotted with heights corresponding to the counts, and text labels displaying these counts are placed above the bars. The chart is titled "Marriages by officiate" with the y-axis labeled "Frequency". The x-axis labels are rotated 45 degrees for better readability.

```
# Calculate frequencies
plotdata <- Marriage %>%
  count(officialTitle)

# Create the bar chart with rotated labels
ggplot(plotdata, aes(x = reorder(officialTitle,n), y = n)) +
  geom_bar(stat = "identity") +
```

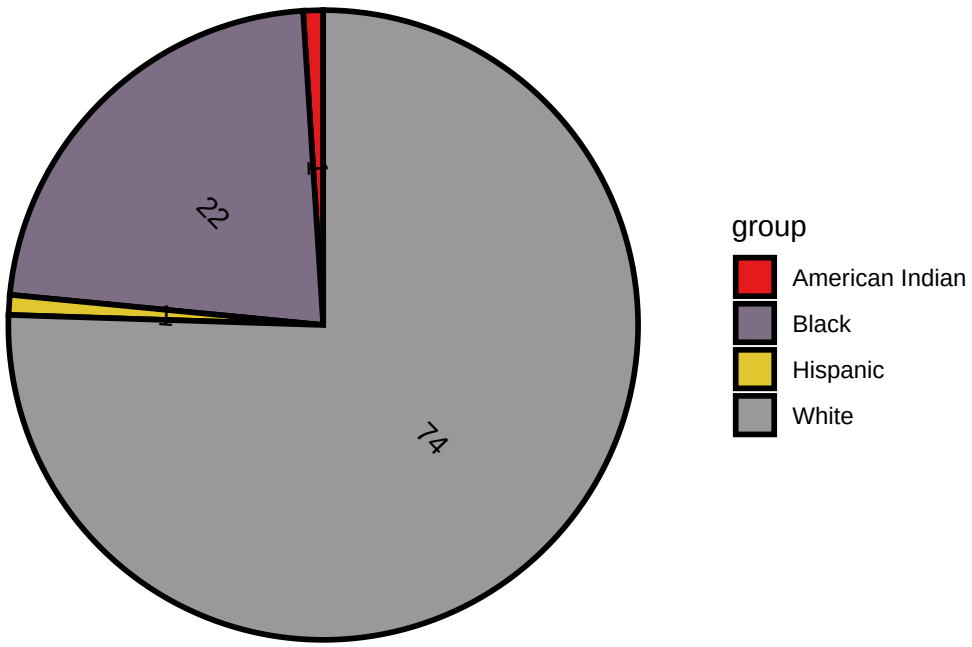
```
  geom_text(aes(label=n), vjust=-0.25)+
  labs(x = "",
       y = "Frequency",
       title = "Marriages by officiate") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



##Pie Chart

The pie chart illustrates a significant disparity in participant racial distribution, with the White population constituting the vast majority (74%), followed by Black participants at 22%. Representation from Hispanic and American Indian groups is minimal, each accounting for a negligible portion of the total.

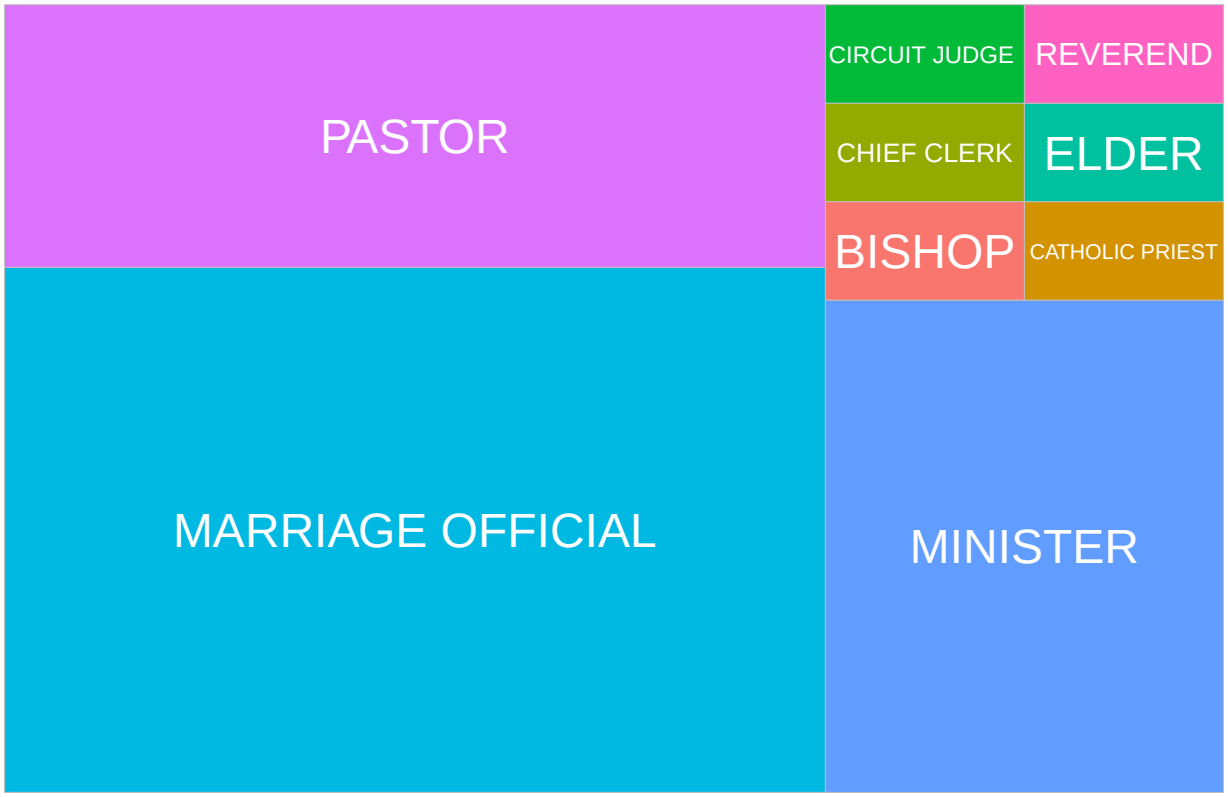
```
#install.packages("ggpie")
library(ggpie)
ggpie(Marriage, group_key = "race", count_type="full", label_info = "count")
```



##Tree Map

The treemap visualizes the distribution of marriage officiants, with 'Marriage Official' and 'Minister' categories dominating the landscape. Within these broad categories, 'Pastor' stands out as the most frequent officiant, followed by 'Circuit Judge' and 'Reverend'. Other officiant types, including 'Chief Clerk', 'Elder', 'Bishop', and 'Catholic Priest', represent smaller proportions of the total marriages.

```
#install.packages("treemapify")
library(treemapify)
ggplot(plotdata,
       aes(fill = officialTitle,
           area=n,
           label=officialTitle))+
  geom_treemap()+
  geom_treemap_text(colour = "white",
                    place = "centre") +
  labs(title = "Marriages by officiate") +
  theme(legend.position = "none")
```

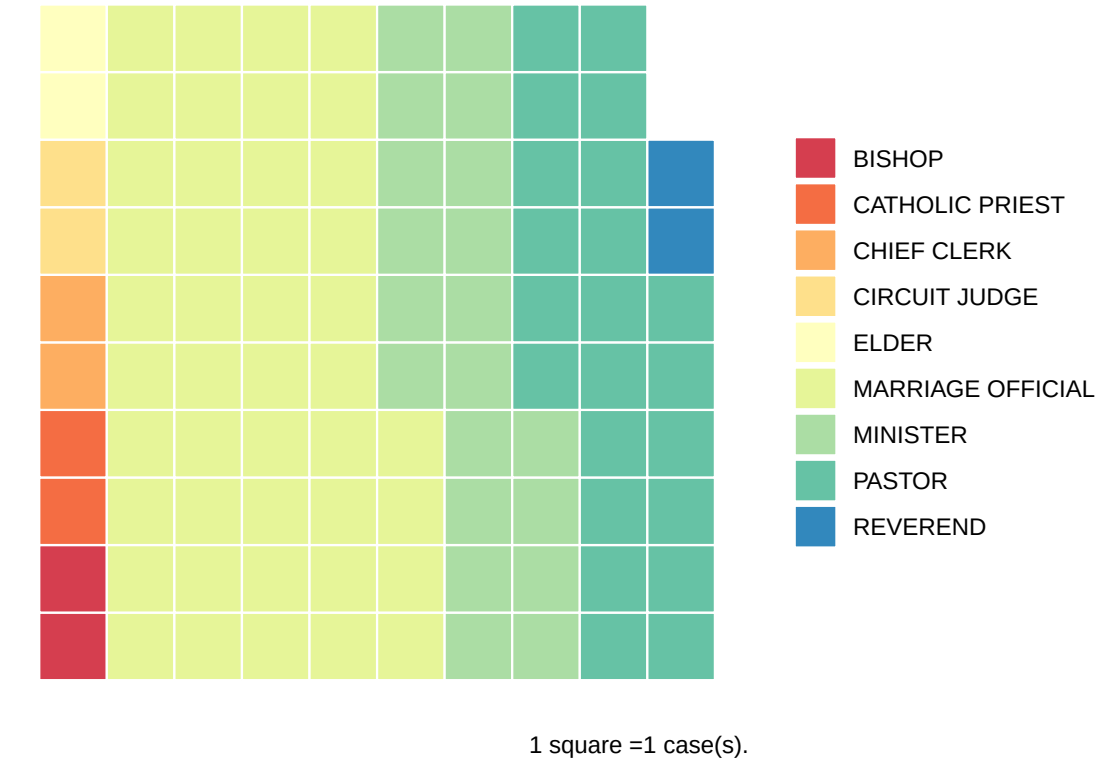


##Waffle Chart

```
#install.packages("waffle")
library(waffle)
```

findings:each square in the chart represents one case, and the color of the square indicates the type of the weeding officials.The chart shows that Minister and Marriage Officials are the most prevalent weeding officials while roles such as Bishop, Catholic Priest, and Reverend are less frequent.

```
cap <- paste0("1 square =", ceiling(sum(plotdata$n) / 100), " case(s).")


ggplot(plotdata, aes(fill=officialTitle, values=n)) +
  geom_waffle(na.rm = TRUE, n_rows = 10, size = 0.4, color = "white") +
  scale_fill_brewer(palette = "Spectral") +
  coord_equal() +
  theme_minimal() +
  theme_enhance_waffle() +
  theme(legend.title = element_blank()) +
  labs(title = "Proportions of wedding officials", caption = cap)
```

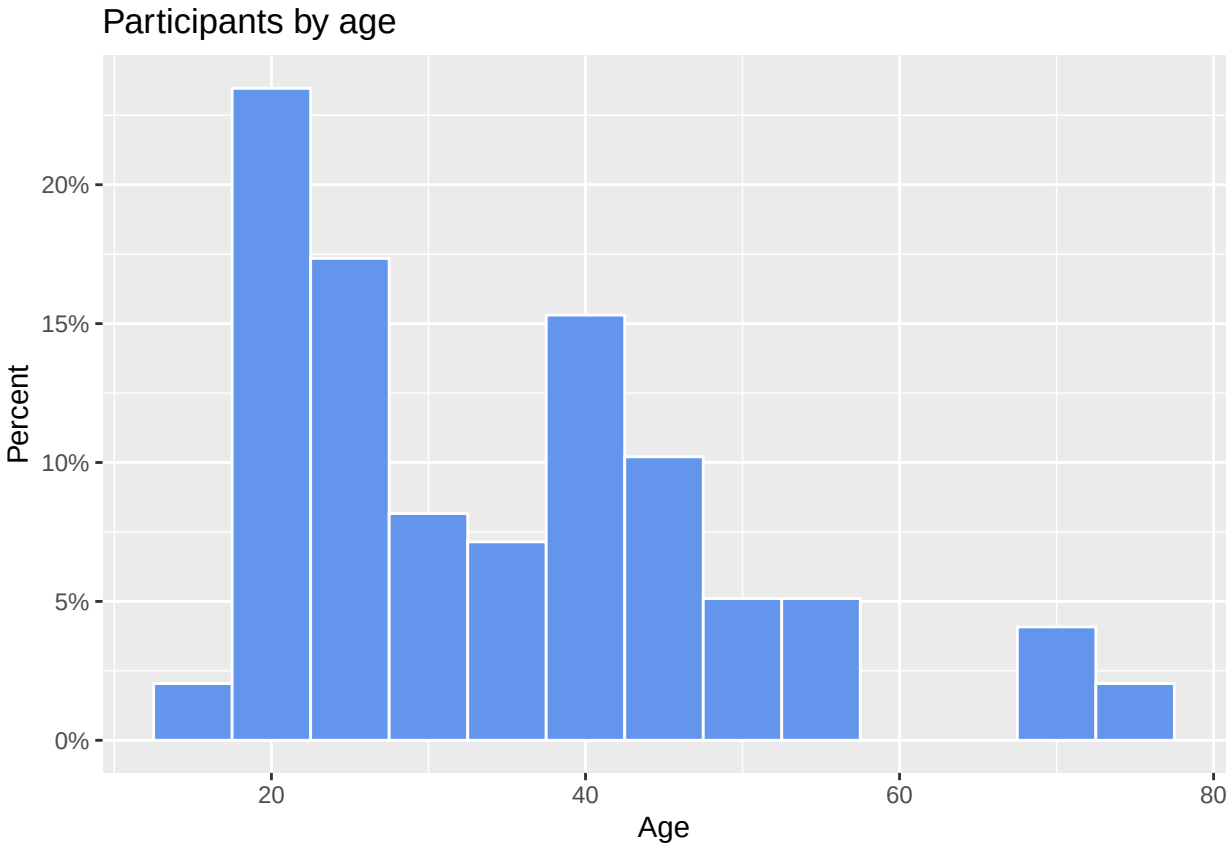


QUANTITATIVE

##HISTOGRAM

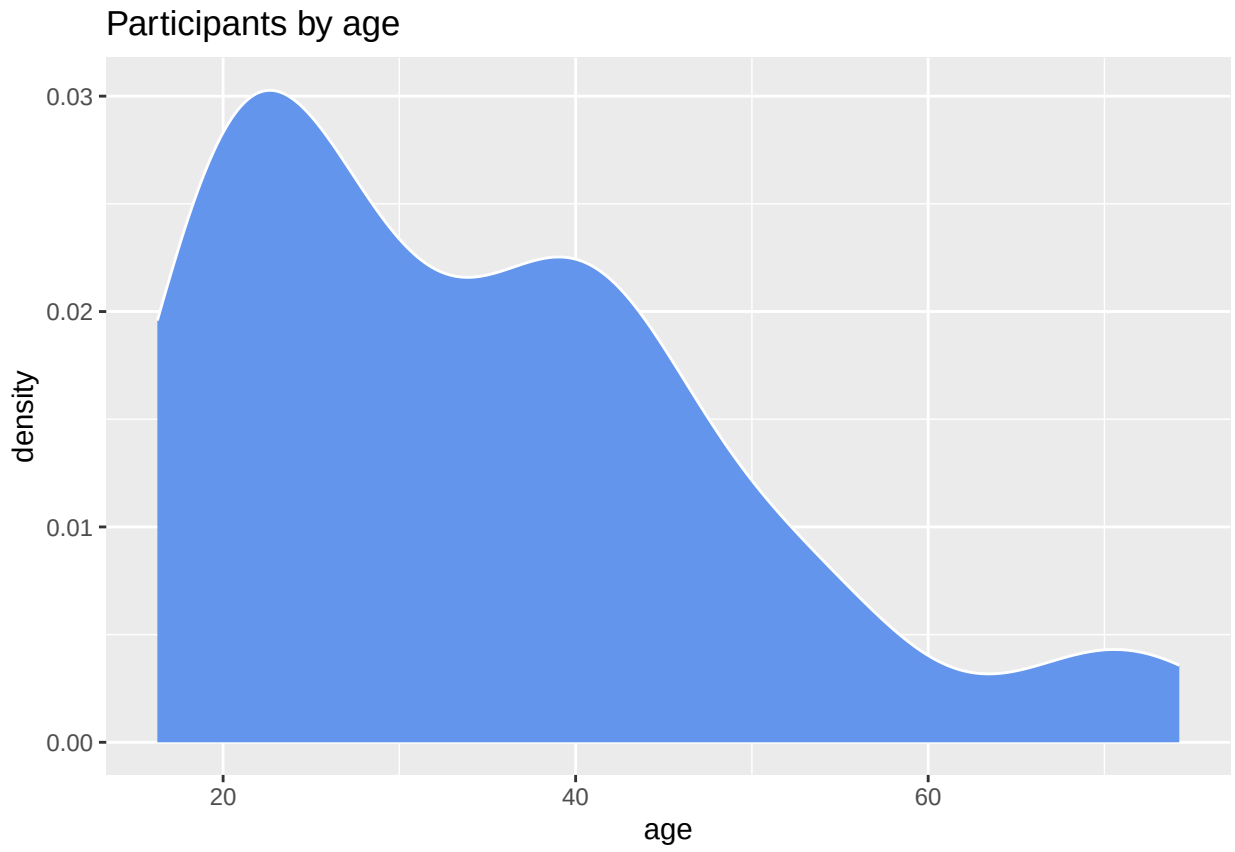
findings: the chart displays the percentage distribution of participants by age, the data suggests that the majority of participants are younger, with significant proportion of middle ages individuals and very few participants. The participation rate is highest among young adults (20-25), indicating a potential focus or interest in this age group. There is another notable group in the middle-aged range (35-45), which could represent a secondary demographic of interest. Participation sharply declines after age 45, with very few participants over age 60.

```
# plot the histogram with percentages on the y-axis
library(scales)
ggplot(Marriage,
       aes(x = age, y= after_stat(count/sum(count)))) +
  geom_histogram(fill = "cornflowerblue",
                 color = "white",
                 binwidth = 5) +
  labs(title="Participants by age",
       y = "Percent",
       x = "Age") +
  scale_y_continuous(labels = percent)
```



##KERNEL DENSITY PLOT findings: the density plot displays the distribution of participants by age, The highest concentration of participants is in the 20-25 age range, similar to the histogram's findings.There is a noticeable secondary concentration of participants around the age of 40-45.The overall trend shows a young participant base, with a steady decrease in density as age increases.

```
ggplot(Marriage,aes(x=age))+
  geom_density(fill="cornflowerblue",color="white")+
  labs(title = "Participants by age")
```



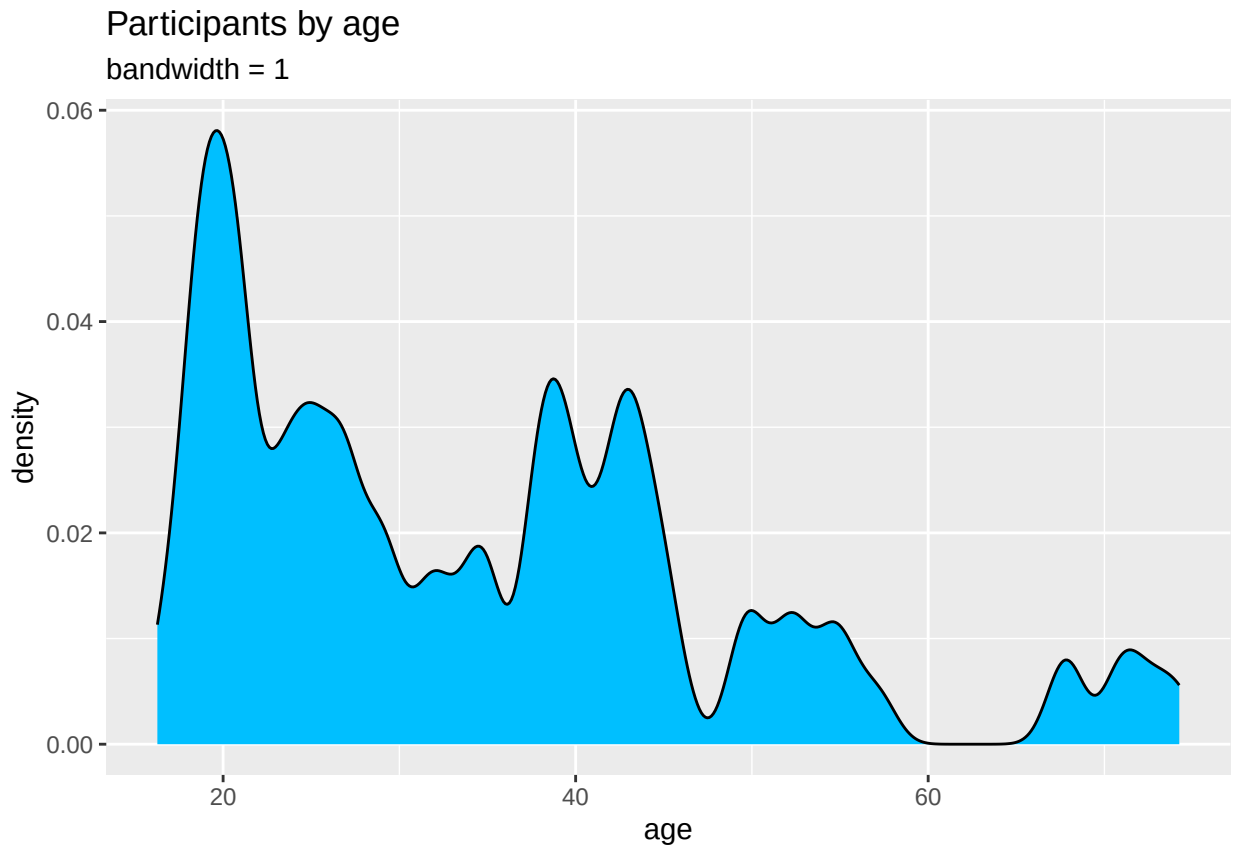
Smoothing parameter

```
# default bandwidth for the age variable
bw.nrd0(Marriage$age)
```

```
## [1] 5.181946
```

here this is the same density graph as above the only difference is that this graph uses smoothing parameter

```
# Create a kernel density plot of age
ggplot(Marriage, aes(x = age)) +
  geom_density(fill = "deepskyblue",
               bw = 1) +
  labs(title = "Participants by age",
       subtitle = "bandwidth = 1")
```

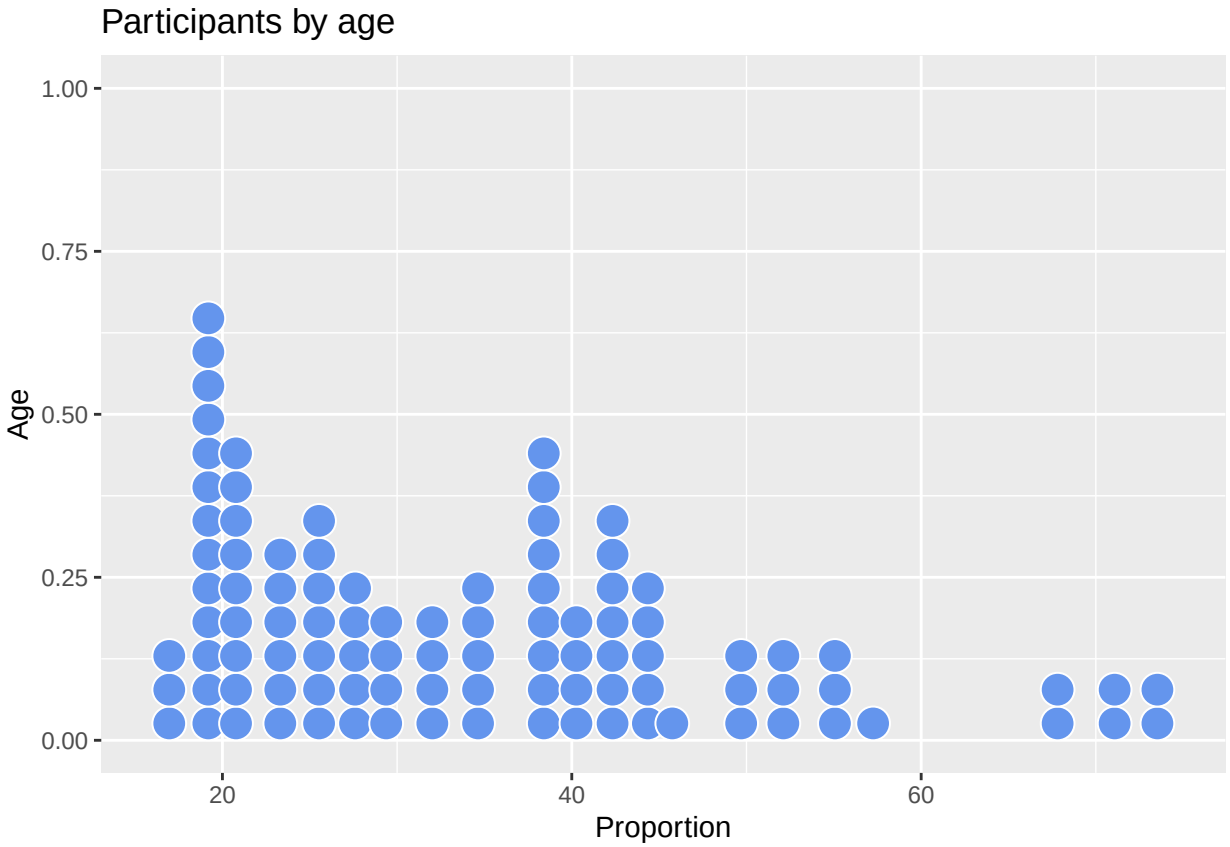


##DOT CHART

findings: The highest concentration of participants is in the 20-25 age range, similar to the histogram's findings.There is a noticeable secondary concentration of participants around the age of 40-45.the plot reveals a decrease in the proportion of participants as age increases

```
ggplot(Marriage, aes(x=age))+
  geom_dotplot(fill="cornflowerblue",color="white")+
  labs(title="Participants by age",
       x="Proportion",
       y="Age")
```

```
## Bin width defaults to 1/30 of the range of the data. Pick better value with
## `binwidth`.
```

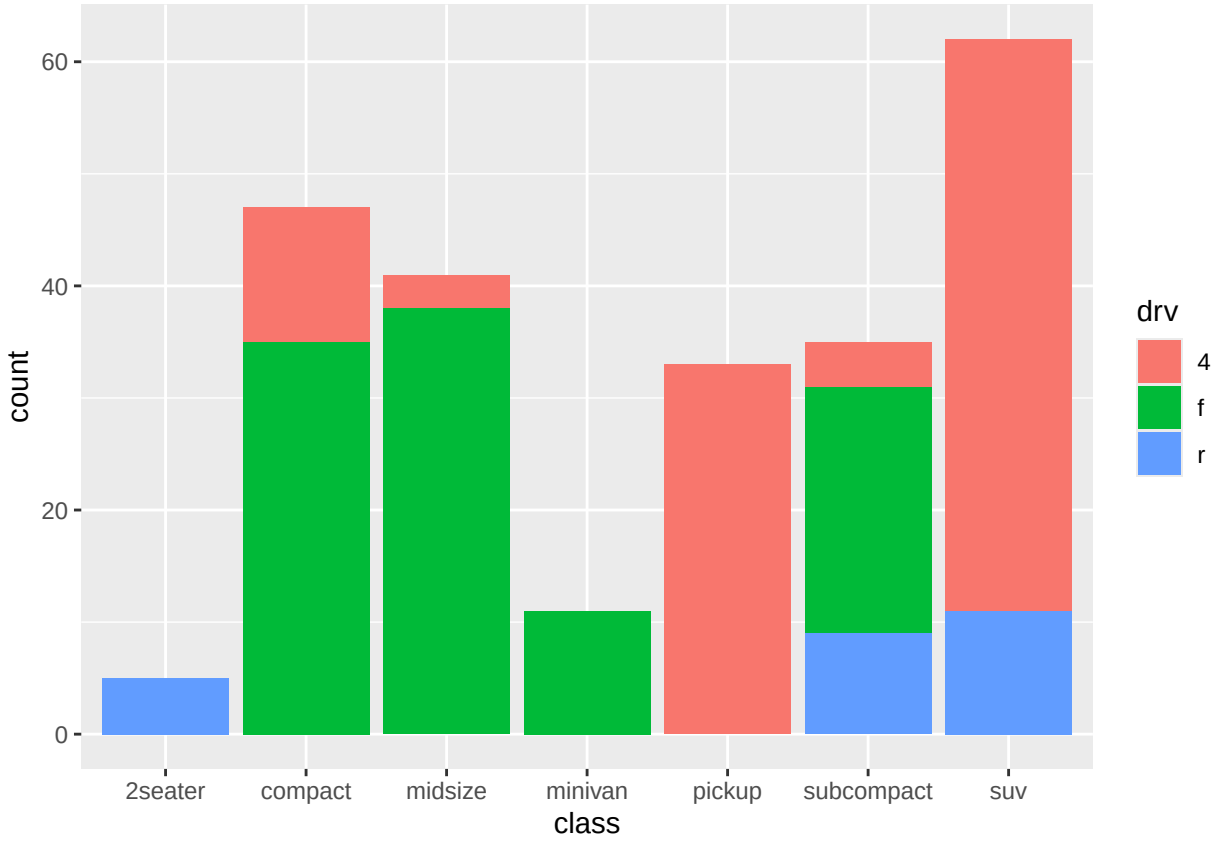


#Bivariate Graphs

##STACKED BAR CHART

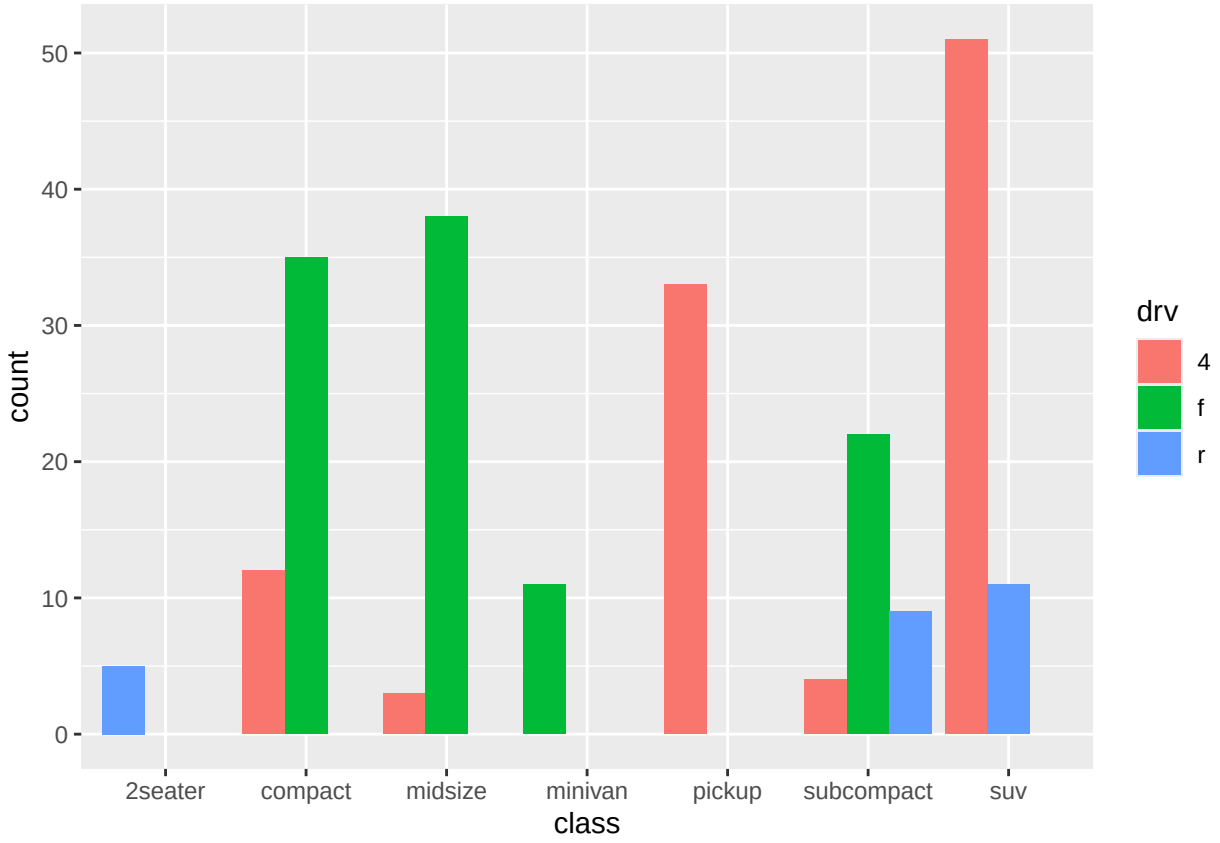
findings: The bar chart illustrates the distribution of different vehicle classes (2seater, compact, midsize, minivan, pickup, subcompact, and SUV) based on their drivetrain configurations ('4', 'f', and 'r'). The compact class has the highest overall count, predominantly in the 'f' category, indicating a preference for front-wheel drive in this segment. In contrast, the pickup class shows a higher count for the '4' category, suggesting a preference for four-wheel drive.

```
ggplot(mpg, aes(x=class,fill=drv))+
 geom_bar(position = "stack")
```



##GROUPED BAR CHART The grouped-bar chart shows the distribution of different vehicle classes (2seater, compact, midsize, minivan, pickup, subcompact, and SUV) based on their drivetrain configurations ('4' for four-wheel drive, 'f' for front-wheel drive, and 'r' for rear-wheel drive). The compact class has the highest overall count, predominantly in the 'f' category, indicating a preference for front-wheel drive in this segment. In contrast, the pickup class shows a higher count for the '4' category, suggesting a preference for four-wheel drive

```
ggplot(mpg, aes(x=class,fill = drv))+
geom_bar(position = position_dodge(preserve = "single"))
```



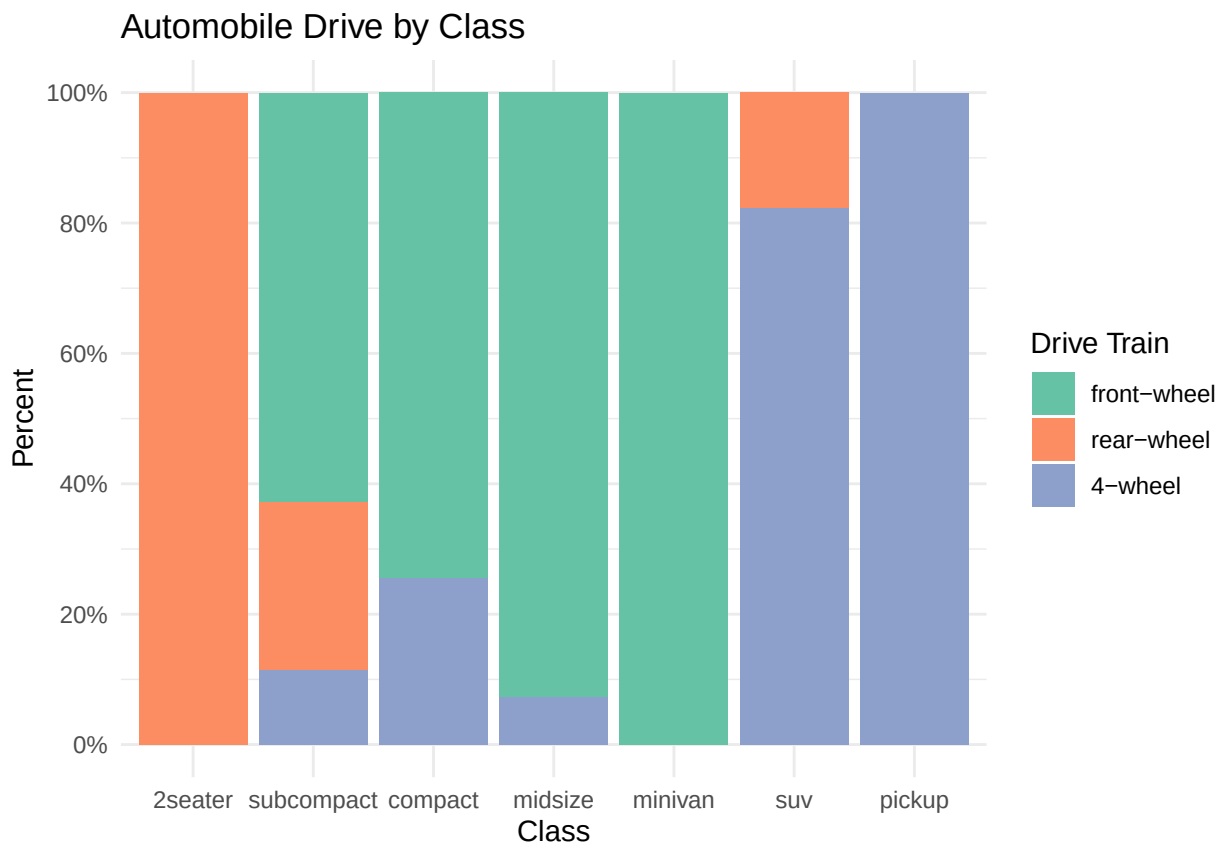
##SEGMENTED BAR CHART

```
#install.packages("scales")
library(scales)
#install.packages("ggplot2")
library(ggplot2)
```

findings: The data reveals that front-wheel drive is predominant in most vehicle classes, especially in compact and subcompact cars, indicating a preference for this drivetrain type in smaller, urban-friendly vehicles. Rear-wheel drive is most common in pickups, likely due to their need for better towing and hauling capabilities. SUVs show a significant mix of all three drivetrain types, reflecting their versatility and varied consumer uses.

```
ggplot(mpg,
       aes(x = factor(class,
                      levels = c("2seater", "subcompact",
                                 "compact", "midsize",
                                 "minivan", "suv", "pickup")),
           fill = factor(drv,
                         levels = c("f", "r", "4"),
                         labels = c("front-wheel",
                                    "rear-wheel",
                                    "4-wheel")))) +
  geom_bar(position = "fill") +
  scale_y_continuous(breaks = seq(0, 1, .2),
                     label = percent) +
```

```
  scale_fill_brewer(palette = "Set2") +
  labs(y = "Percent",
       fill="Drive Train",
       x = "Class",
       title = "Automobile Drive by Class") +
  theme_minimal()
```



add percent labels to each segment. First, we'll create a summary dataset that has the necessary labels.

```
# create a summary dataset
library(dplyr)
data <- mpg %>%
  group_by(class, drv) %>%
  summarize(n = n()) %>%
  mutate(pct = n/sum(n),
         lbl = scales::percent(pct))
```

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

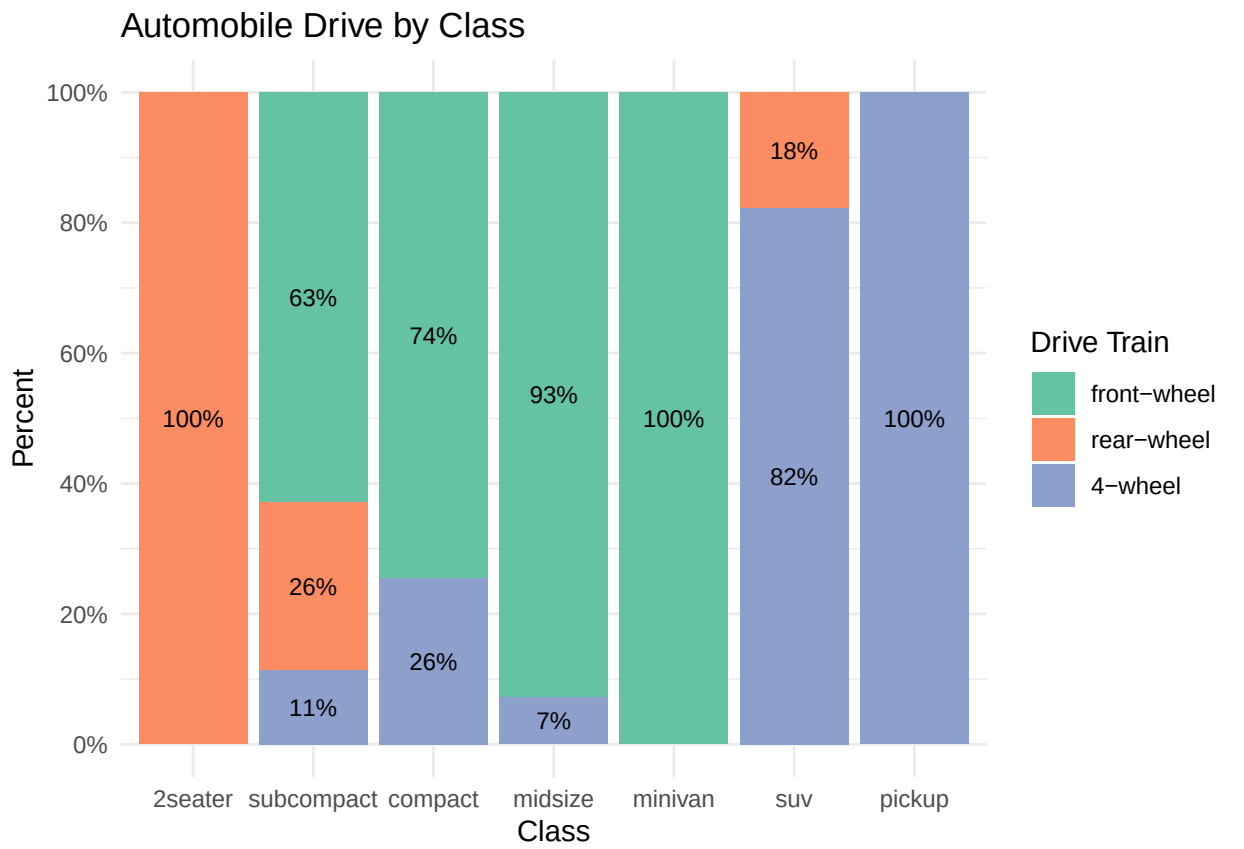
```
data
```

```
## # A tibble: 12 x 5
## # Groups:   class [7]
##    class      drv       n    pct lbl
```

```
##    <chr>      <chr> <int>  <dbl> <chr>
##  1 2seater    r         5 1      100%
##  2 compact    4        12 0.255  26%
##  3 compact    f        35 0.745  74%
##  4 midsize    4         3 0.0732 7%
##  5 midsize    f        38 0.927  93%
##  6 minivan    f        11 1      100%
##  7 pickup     4        33 1      100%
##  8 subcompact 4         4 0.114  11%
##  9 subcompact f        22 0.629  63%
## 10 subcompact r         9 0.257  26%
## 11 suv        4        51 0.823  82%
## 12 suv        r        11 0.177  18%
```

this graph is same as above graph but with added labels

```
# create segmented bar chart
# adding labels to each segment

ggplot(data,
       aes(x = factor(class,
                      levels = c("2seater", "subcompact",
                                 "compact", "midsize",
                                 "minivan", "suv", "pickup")),
           y = pct,
          fill = factor(drv,
                        levels = c("f", "r", "4"),
                        labels = c("front-wheel",
                                   "rear-wheel",
                                   "4-wheel")))) +
  geom_bar(stat = "identity",
           position = "fill") +
  scale_y_continuous(breaks = seq(0, 1, .2),
                     label = percent) +
  geom_text(aes(label = lbl),
            size = 3,
            position = position_stack(vjust = 0.5)) +
  scale_fill_brewer(palette = "Set2") +
  labs(y = "Percent",
       fill="Drive Train",
       x = "Class",
       title = "Automobile Drive by Class") +
  theme_minimal()
```



#QUANTITATIVE VS. QUANTITATIVE

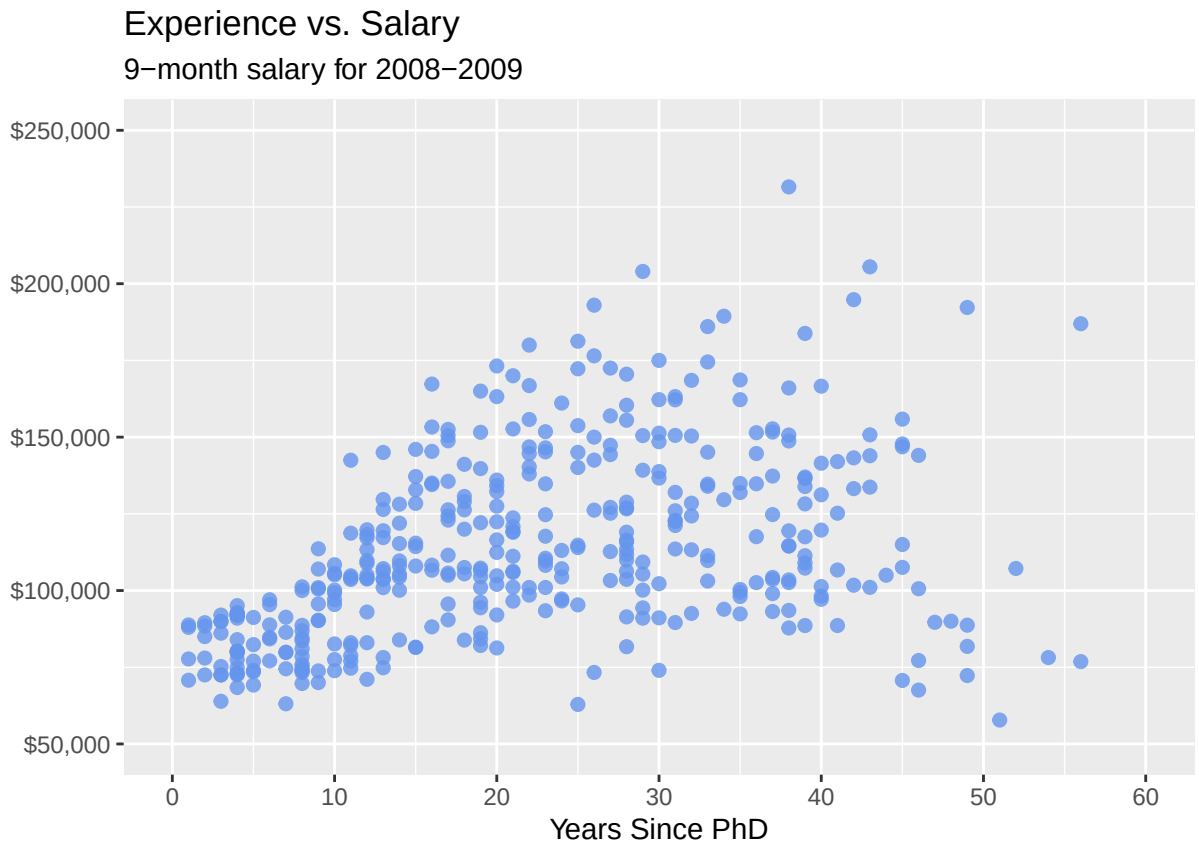
##SCATTERPLOT

```
#install.packages("carData")
library(carData)
```

findings: The scatter plot shows the relationship between the number of years since obtaining a PhD and the 9-month salary for the years 2008-2009. The data points indicate that, generally, salaries tend to increase with more years of experience. However, there is a significant variation in salaries at almost every experience level, suggesting that factors other than just years since obtaining a PhD influence salary. The concentration of data points is denser at the lower end of both axes, indicating that more individuals have fewer years of experience and lower salaries.

```
data(Salaries, package="carData")

ggplot(Salaries,
       aes(x = yrs.since.phd, y = salary)) +
  geom_point(color="cornflowerblue",
             size = 2,
             alpha=.8) +
  scale_y_continuous(label = scales::dollar,
                     limits = c(50000, 250000)) +
  scale_x_continuous(breaks = seq(0, 60, 10),
                     limits=c(0, 60)) +
  labs(x = "Years Since PhD",
```

```
       y = "",
       title = "Experience vs. Salary",
       subtitle = "9-month salary for 2008-2009")
```



##Adding best fit lines

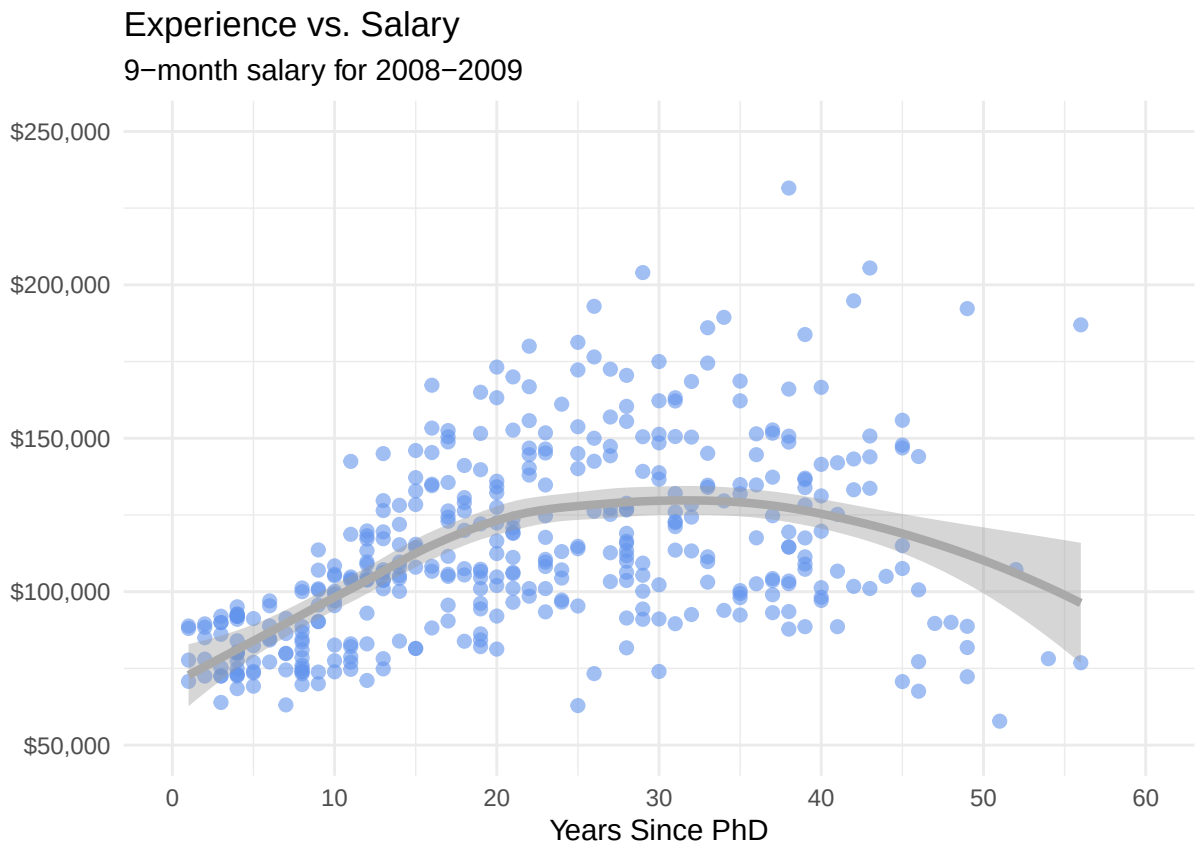
findings: The scatter plot shows a clear trend where salaries increase with years of experience post-PhD, peaking around 25 years. After this peak, salaries tend to decline slightly despite additional years of experience. This suggests that mid-career professionals (20-30 years since PhD) earn the highest salaries, possibly due to accumulated expertise and peak productivity

```
# scatterplot with loess smoothed line
# and better labeling and color
ggplot(Salaries,
       aes(x = yrs.since.phd, y = salary)) +
  geom_point(color="cornflowerblue",
             size = 2,
             alpha=.6) +
  geom_smooth(size = 1.5,
              color = "darkgrey") +
  scale_y_continuous(label = scales::dollar,
                     limits=c(50000, 250000)) +
  scale_x_continuous(breaks = seq(0, 60, 10),
                     limits=c(0, 60)) +
  labs(x = "Years Since PhD",
       y = "",
```

```
       title = "Experience vs. Salary",
       subtitle = "9-month salary for 2008-2009") +
  theme_minimal()
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

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

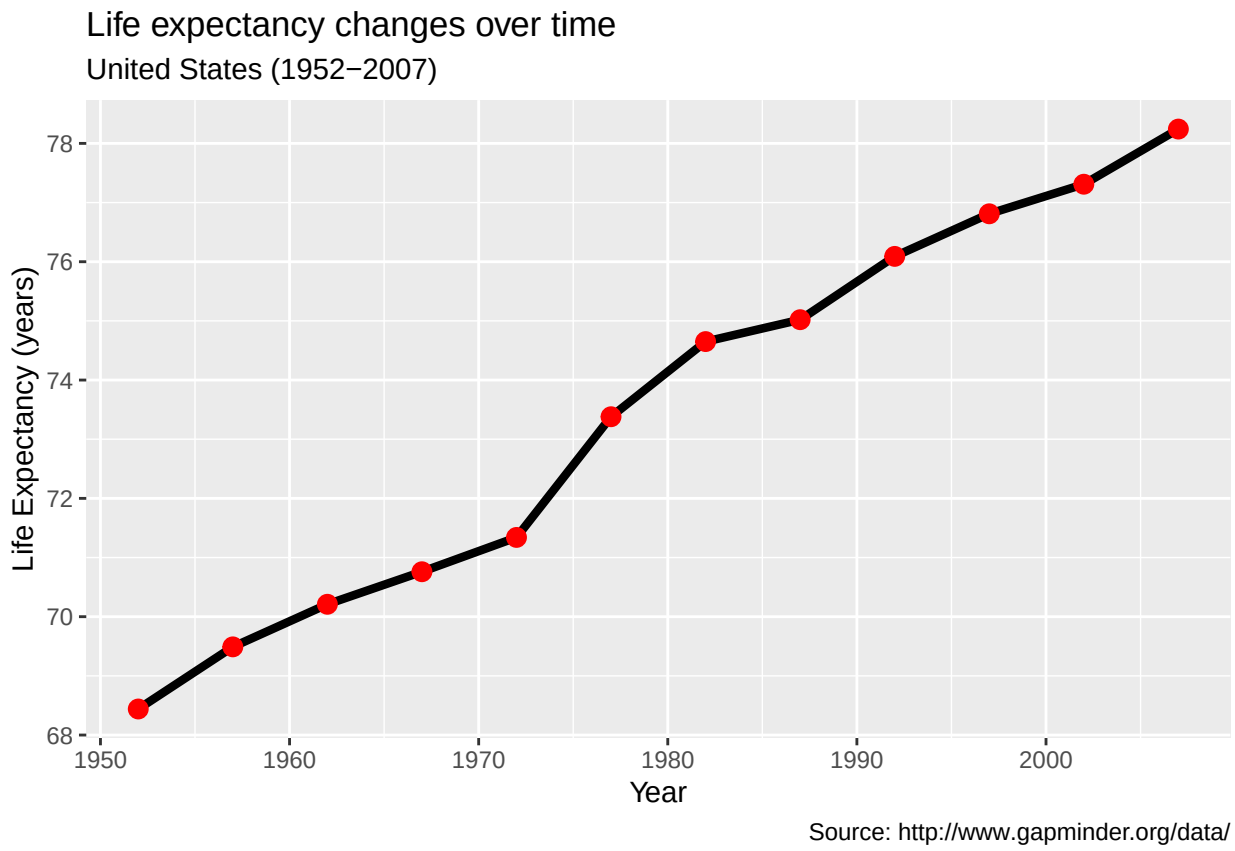


##LINE PLOT

```
#install.packages("gapminder")
library(gapminder)
library(ggplot2)
library(dplyr)
```

findings: the line chart shows a consistent upward trend in life expectancy. Starting just below 70 years in 1952, life expectancy steadily increases, reaching approximately 78 years by 2007. This continuous rise suggests significant improvements in healthcare, nutrition, and living conditions over the decades. The absence of major dips indicates that there were no prolonged crises severely impacting life expectancy during this period. Overall, the chart highlights a positive trend in public health and longevity in the United States over the 55-year span.

```r
data(gapminder, package="gapminder")

plott <- filter(gapminder, country == "United States")

ggplot(plott, aes(x = year, y = lifeExp)) +
  geom_line(size = 1.5,
            color = "black") +
  geom_point(size = 3,
             color = "red") +
  labs(y = "Life Expectancy (years)",
       x = "Year",
       title = "Life expectancy changes over time",
       subtitle = "United States (1952-2007)",
       caption = "Source: http://www.gapminder.org/data/")
```

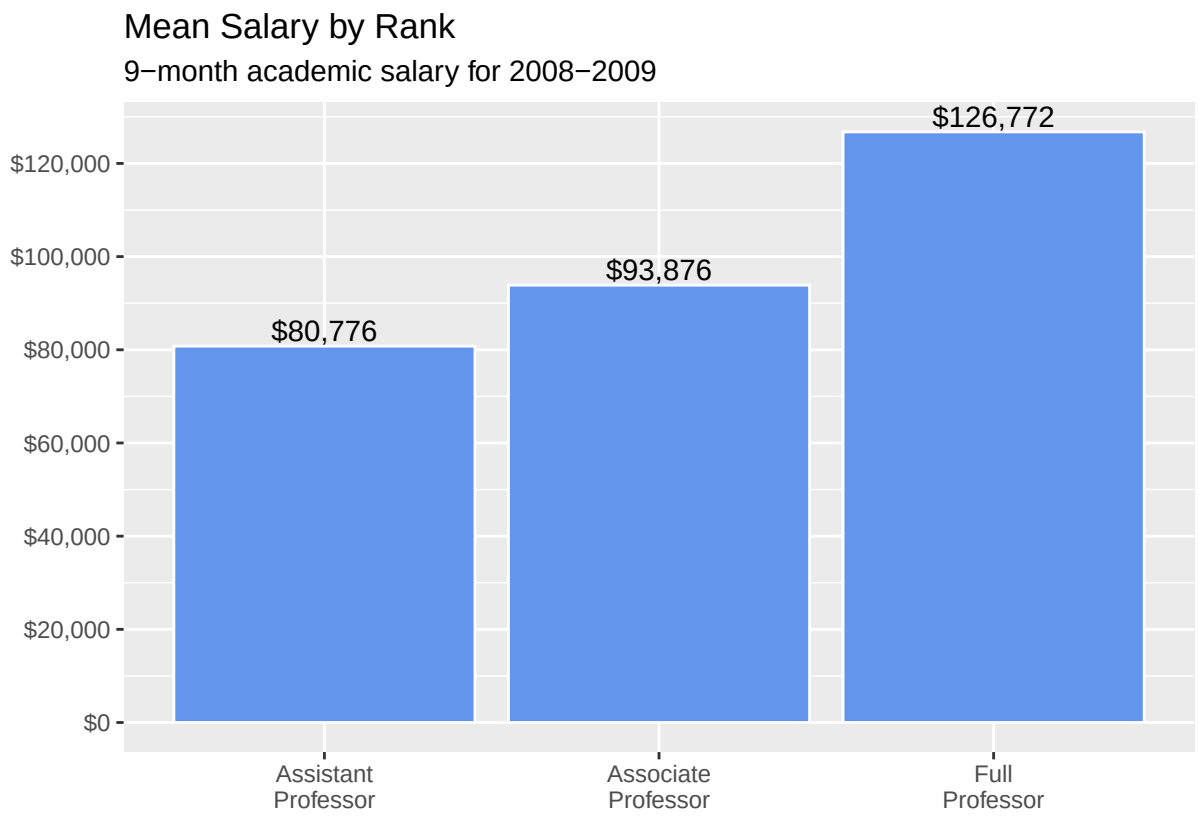


#CATEGORICAL VS. QUANTITATIVE

BAR CHART (ON SUMMARY STATISTICS)

findings: the barchart displays the average salaries of the professors,There is a clear upward trend in mean salary as academic rank increases. The most substantial increase in salary is observed between Associate Professors and Full Professors. The data suggests that advancing in academic rank is associated with substantial increases in salary

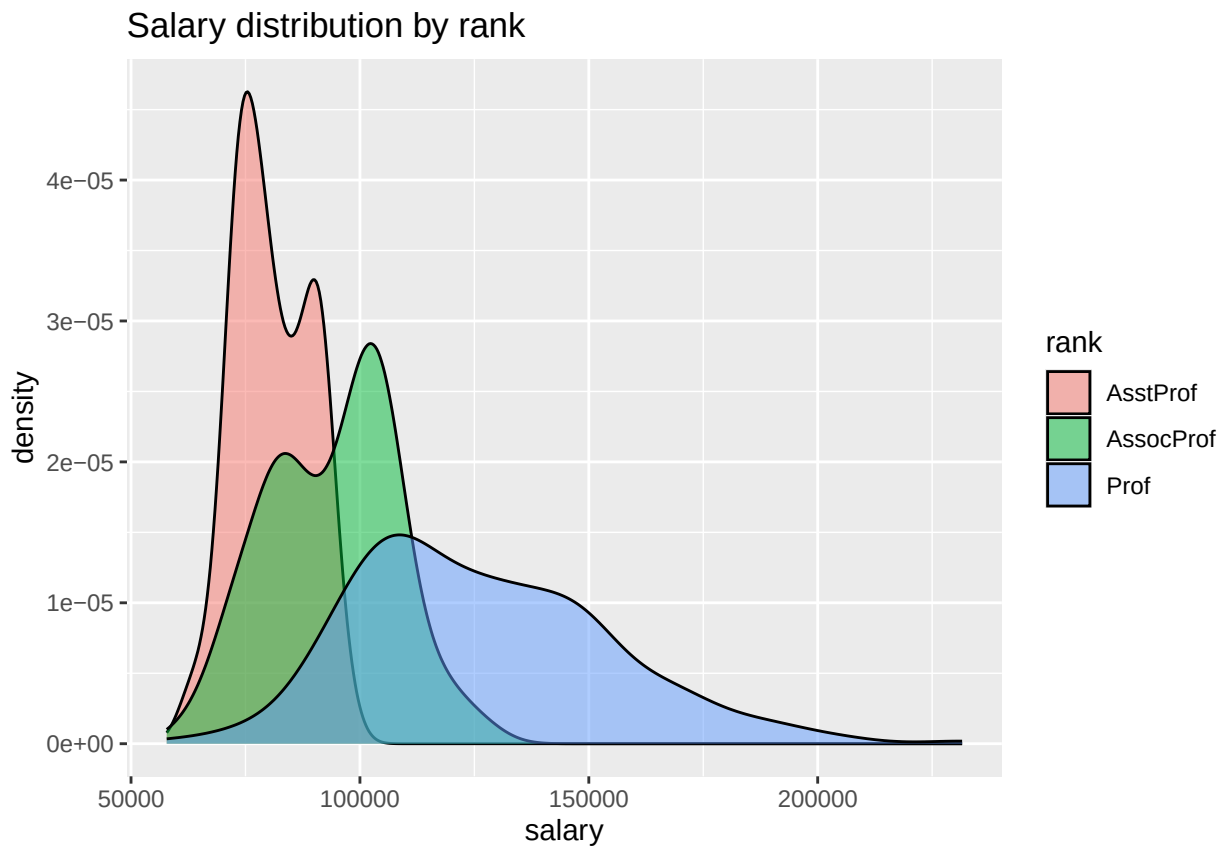
```r
data(Salaries, package="carData")

# calculate mean salary for each rank
plotdata <- Salaries %>%
  group_by(rank) %>%
  summarize(mean_salary = mean(salary))

# plot mean salaries
ggplot(plotdata,
       aes(x = factor(rank,
                      labels = c("Assistant\nProfessor",
                                 "Associate\nProfessor",
                                 "Full\nProfessor")),
                      y = mean_salary)) +
  geom_bar(stat = "identity",
           fill = "cornflowerblue",color="white") +
  geom_text(aes(label = dollar(mean_salary)),
            vjust = -0.25) +
  scale_y_continuous(breaks = seq(0, 130000, 20000),
                     label = dollar) +
  labs(title = "Mean Salary by Rank",
       subtitle = "9-month academic salary for 2008-2009",
       x = "",
       y = "")
```



##GROUPED KERNEL DENSITY PLOTS

findings: the density plot shows the distribution of salaries for three academic ranks: Assistant Professor, Associate Professor, and Professor.This graph visually represents how academic salaries vary by rank and suggests that advancing in rank can lead to higher and more varied salaries

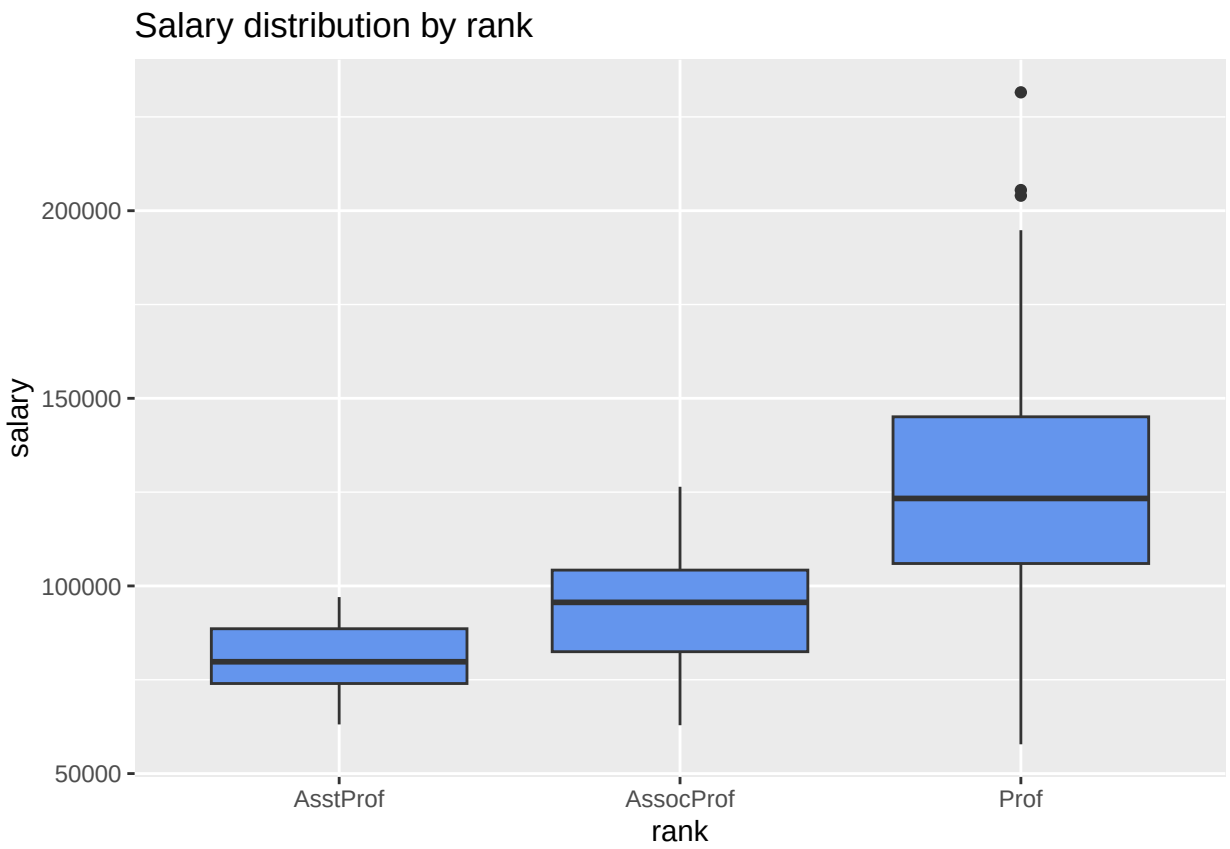
```
ggplot(Salaries,aes(x=salary,fill = rank))+
  geom_density(alpha = 0.5)+
  labs(title = "Salary distribution by rank")
```



##BOX PLOTS

findings: The box plot shows that salaries increase with academic rank. Assistant Professors have the lowest median salary with a narrow range, indicating less variation. Associate Professors earn more with a slightly wider range. Professors have the highest median salary and the widest range, suggesting significant variability. Outliers in the Professor rank indicate some earn much more than their peers. This graph highlights how salaries rise with rank and the variability within each rank's salary distribution.

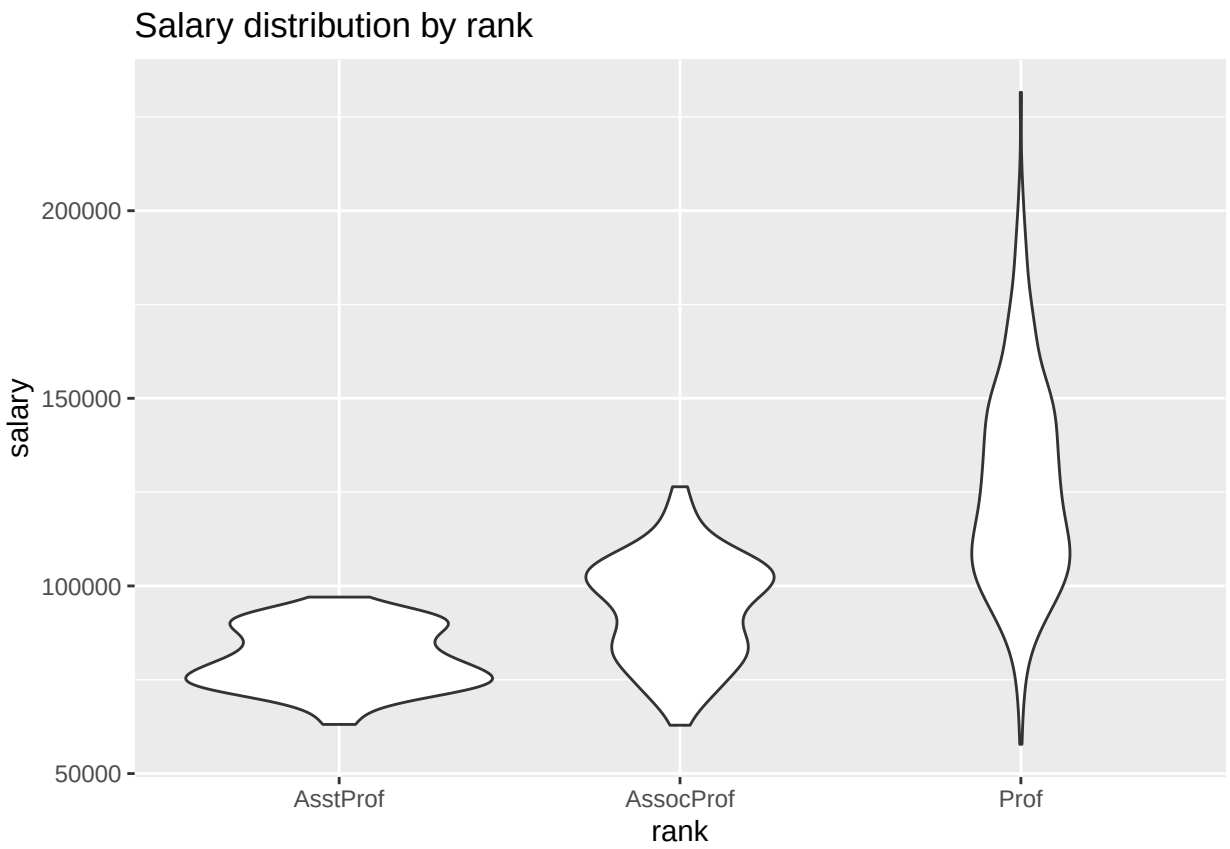
```
ggplot(Salaries,aes(x=rank,y=salary))+
  geom_boxplot(fill="cornflowerblue")+
  labs(title = "Salary distribution by rank")
```



##VIOLIN PLOTS

findings: The violin plot illustrates the salary distributions for Assistant Professors, Associate Professors, and Professors. Assistant Professors have a wide distribution at lower salary levels, indicating variability and a higher concentration of individuals in this range. Associate Professors show a more uniform distribution with a slight bulge in the mid-salary range. Professors have a narrow distribution at lower salaries, which dramatically widens towards higher salaries, indicating significant variability and some extreme high salaries. This plot effectively highlights how salaries increase with rank and the spread of salaries within each rank, providing a comprehensive view of salary structures in academia.

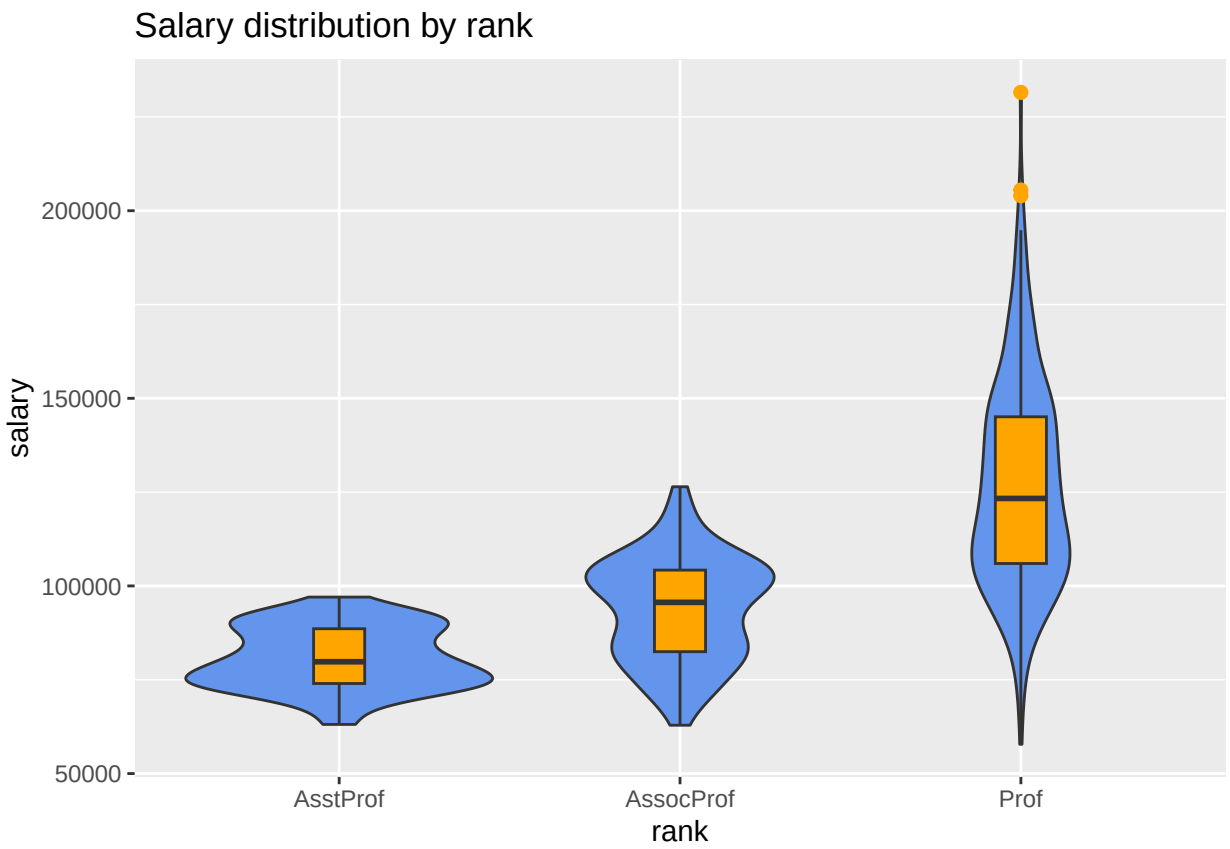
```
ggplot(Salaries,aes(x=rank,y=salary))+
  geom_violin()+
  labs(title = "Salary distribution by rank")
```



##Voilin and Boxplot

findings: The violin plot shows the salary distributions for Assistant Professors, Associate Professors, and Professors. Assistant Professors have a compact distribution with salaries mostly clustered around the median, indicating less variation. Associate Professors show a wider range and distribution, but most salaries are still concentrated near the median. Professors have the widest distribution and highest salary values, including an outlier at the top end, suggesting that Professors have a higher potential for earning and greater variability in their salaries. This plot effectively illustrates how salaries increase with rank and highlights the disparities within each rank's salary distribution.

```
#Voilin and Boxplot..
ggplot(Salaries, aes(x = rank, y = salary)) +
  geom_violin(fill = "cornflowerblue") +
  geom_boxplot(width = .15,
               fill = "orange",
               outlier.color = "orange",
               outlier.size = 2) +
  labs(title = "Salary distribution by rank")
```



##RIDGELINE PLOTS

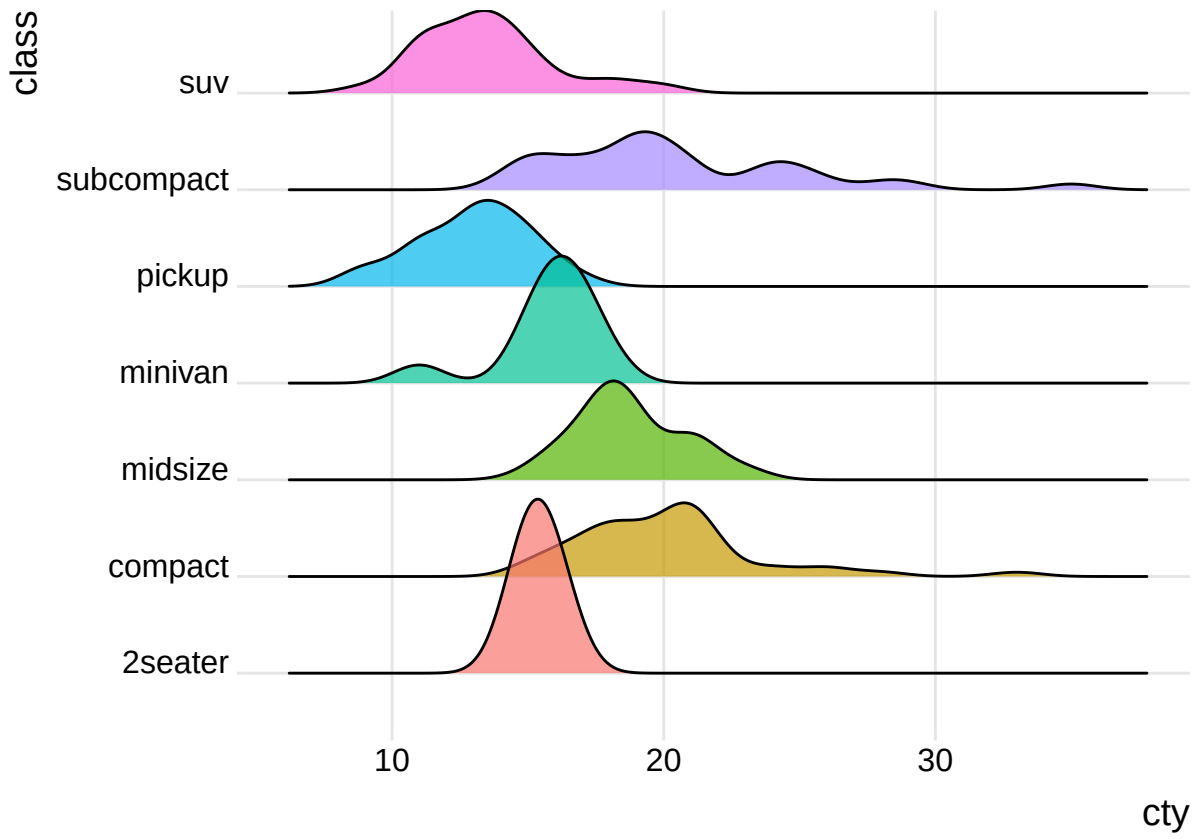
installiing and loading the necessary package

```
#install.packages("ggridges")
library(ggridges)
```

findings: The ridge plot shows the distribution of city values for different vehicle classes, including 'SUV', 'subcompact', 'pickup', 'minivan', 'midsize', 'compact', and '2seater'. The x-axis represents city values ranging from approximately 5 to 35, which likely indicates fuel efficiency in an urban setting. Each vehicle class has a distinct density curve, revealing how frequently certain city values occur within that class. For instance, 'subcompact' and 'compact' cars tend to have higher city values, suggesting better fuel efficiency, while 'pickup' and 'SUV' classes show lower city values, indicating lower fuel efficiency. This plot allows for a clear comparison of fuel efficiency across different vehicle types, highlighting the variations in urban fuel consumption.

```
ggplot(mpg,aes(x=cty,y=class,fill = class))+
  geom_density_ridges(alpha=0.7)+
  theme_ridges()+
  labs("Highway mileage by auto class")+
  theme(legend.position = "none")
```

```
## Picking joint bandwidth of 0.929
```

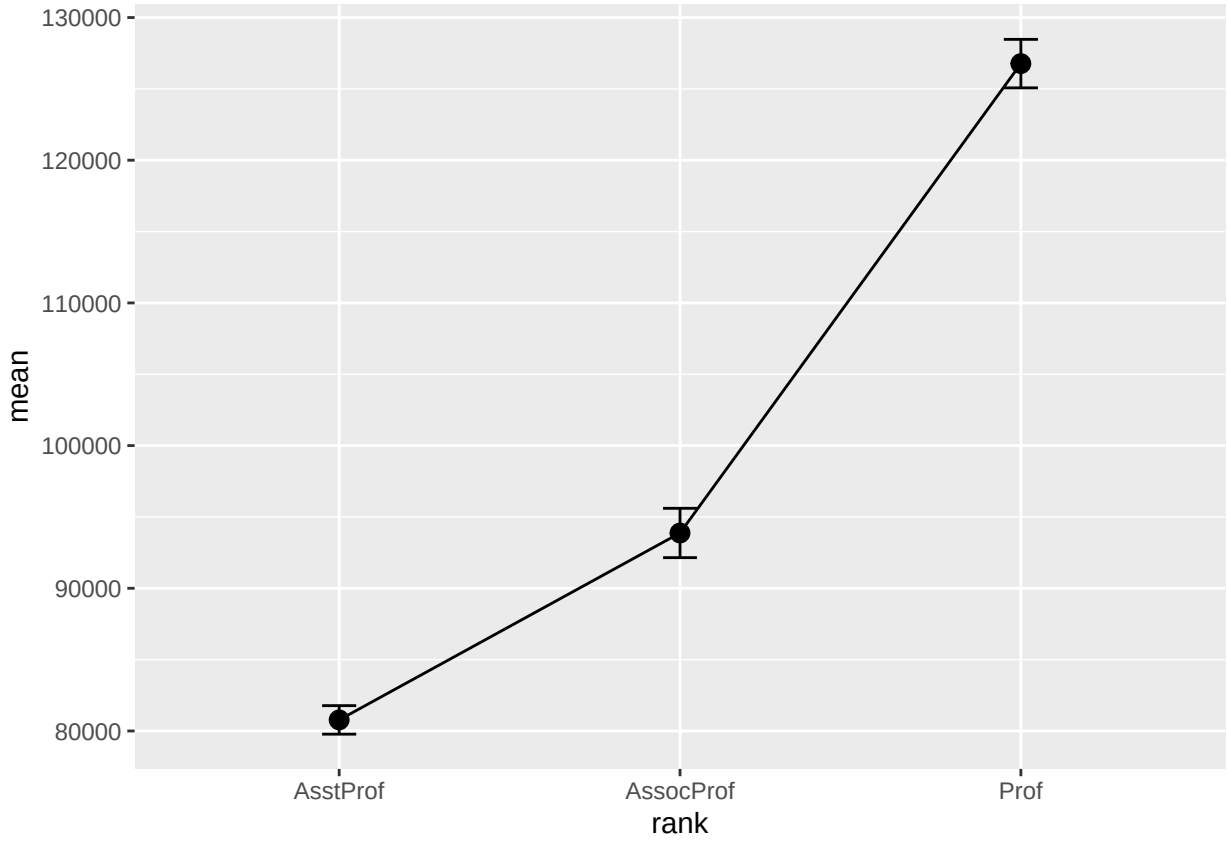


##MEAN/SEM PLOTS

```
df <- Salaries %>%
  group_by(rank) %>%
  summarise(n=n(),
mean=mean(salary),
sd=sd(salary),
se=sd/sqrt(n),
ci=qt(0.975, df=n-1) * sd/sqrt(n))
```

findings: The line plot with error bars illustrates the mean values for three professional ranks: Assistant Professor (AsstProf), Associate Professor (AssocProf), and Professor (Prof).The error bars indicate variability within each rank, suggesting some salary differences among individuals in the same rank.

```
ggplot(df,
       aes(x = rank,
           y = mean,
           group = 1)) +
  geom_point(size = 3) +
  geom_line() +
  geom_errorbar(aes(ymin = mean - se,
                    ymax = mean + se),
                width = .1)
```



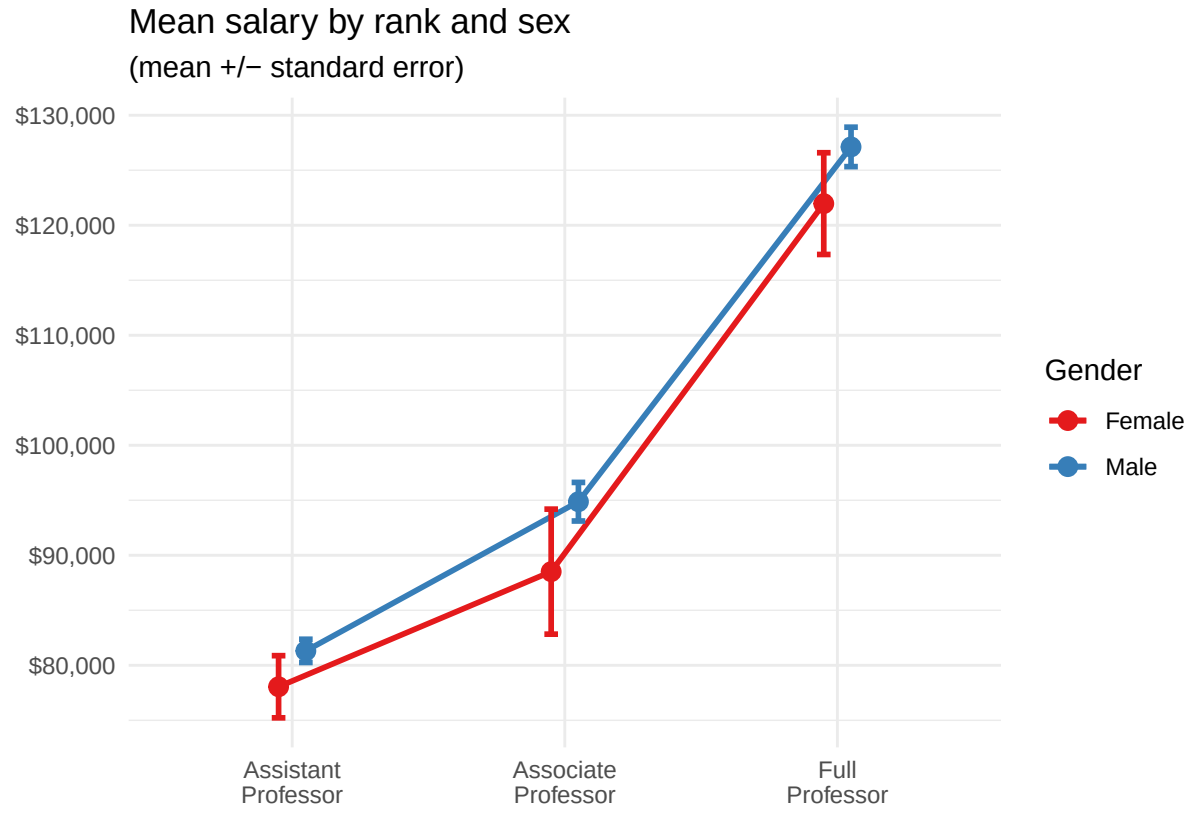
findings: The plot compares the mean salaries of males and females across three academic ranks: Assistant Professor, Associate Professor, and Professor.The error bars indicate variability within each rank, suggesting some differences in salaries among individuals in the same rank. This plot highlights gender disparities in academic salaries and shows that these disparities persist across different ranks

```
# calculate means and standard errors by rank and sex
plotdata <- Salaries %>%
  group_by(rank, sex) %>%
  summarize(n = n(),
            mean = mean(salary),
            sd = sd(salary),
            se = sd/sqrt(n))
```

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

```
# improved means/standard error plot
pd <- position_dodge(0.2)
ggplot(plotdata,
       aes(x = factor(rank,
                      labels = c("Assistant\nProfessor",
                                 "Associate\nProfessor",
                                 "Full\nProfessor")),
                      y = mean, group=sex, color=sex)) +
  geom_point(position=pd,
             size=3) +
```

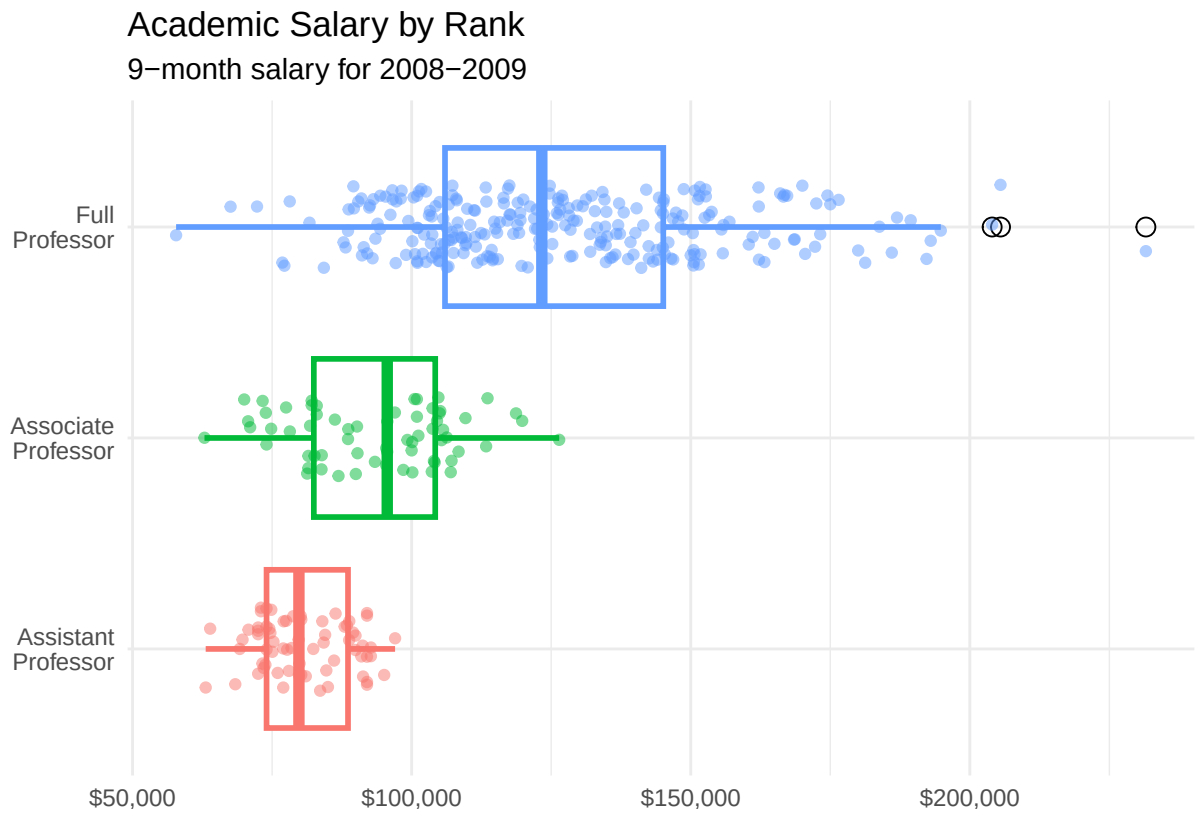
```
  geom_line(position=pd,
            size = 1) +
  geom_errorbar(aes(ymin = mean - se,
                    ymax = mean + se),
                width = .1,
                position=pd,
                size=1) +
  scale_y_continuous(label = scales::dollar) +
  scale_color_brewer(palette="Set1") +
  theme_minimal() +
  labs(title = "Mean salary by rank and sex",
       subtitle = "(mean +/- standard error)",
       x = "",
       y = "",
       color = "Gender")
```



##STRIP PLOTS

```
# plot the distribution of salaries by rank using jittering
ggplot(Salaries, aes(x = factor(rank,
                                labels = c("Assistant\nProfessor",
                                           "Associate\nProfessor",
                                           "Full\nProfessor")),
                     y = salary,
                     color = rank)) +
  geom_boxplot(size = 1,
```

```
              outlier.shape = 1,
              outlier.color = "black",
              outlier.size  = 3) +
  geom_jitter(alpha = 0.5,
              width = 0.2) +
  scale_y_continuous(labels = dollar) +
  labs(title = "Academic Salary by Rank",
       subtitle = "9-month salary for 2008-2009",
       x = "",
       y = "") +
  theme_minimal() +
  theme(legend.position = "none") +
  coord_flip()
```



installing the necessary package

```
#install.packages("ggpol")
library(ggpol)
```

This code creates a combined box plot and jitter plot of academic salaries for three different ranks (Assistant Professor, Associate Professor, and Full Professor). The plot uses a minimal theme, removes the x and y-axis labels, formats the y-axis as currency, and includes a title and subtitle. The boxes are outlined in black, the jittered points are colored dark grey, and error bars are included. The legend is omitted to keep the plot clean

The plot displays the distribution of academic salaries for three ranks (Assistant Professor, Associate Professor, and Full Professor) during the 9-month period of 2008-2009, using a combination of box plots and jitter

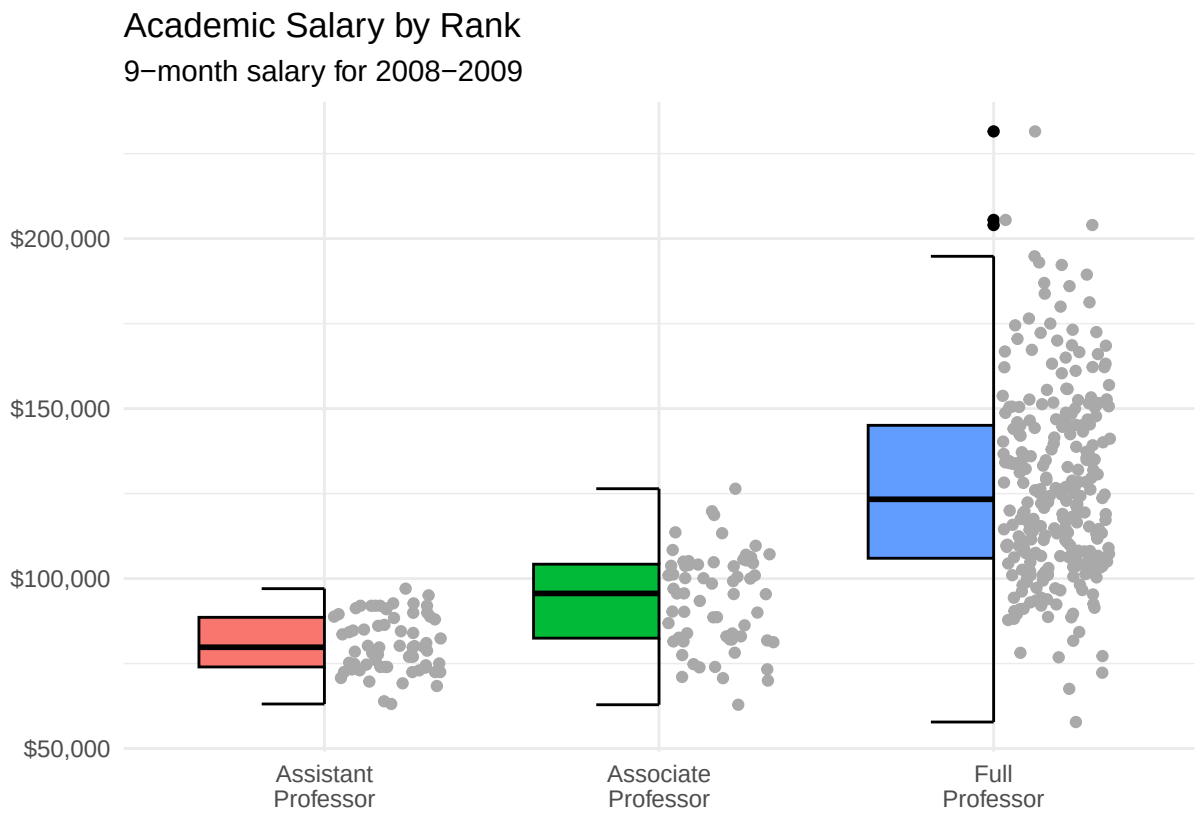
plots. It reveals that Full Professors have the highest median salary at around $150,000, with a wide range of salaries extending from $100,000 to over $200,000, and notable variability indicated by scattered points. Associate Professors have a median salary of approximately $100,000, with salaries mostly ranging from $80,000 to $120,000 and a tighter interquartile range. Assistant Professors have the lowest median salary, around $90,000, with salaries primarily between $70,000 and $100,000. The plot highlights the increasing salary and variability as the academic rank increases

```
ggplot(Salaries,
       aes(x = factor(rank,
                      labels = c("Assistant\nProfessor",
                                 "Associate\nProfessor",
                                 "Full\nProfessor")),
           y = salary,
           fill=rank)) +
  geom_boxjitter(color="black",
                 jitter.color = "darkgrey",
                 errorbar.draw = TRUE) +
  scale_y_continuous(label = dollar) +
  labs(title = "Academic Salary by Rank",
       subtitle = "9-month salary for 2008-2009",
       x = "",
       y = "") +
  theme_minimal() +
  theme(legend.position = "none")
```

```
## Warning: Using the ‘size‘ aesthetic in this geom was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ in the ‘default_aes‘ field and elsewhere instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```
## Warning: Using the ‘size‘ aesthetic with geom_segment was deprecated in ggplot2 3.4.0.
## i Please use the ‘linewidth‘ aesthetic instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```
## Warning: Using the ‘size‘ aesthetic with geom_crossbar was deprecated in ggplot2 3.4.0.
## i Please use the ‘linewidth‘ aesthetic instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```



##CLEVELAND DOT CHARTS

This code creates a horizontal dot plot showing the life expectancy of various Asian countries in 2007. Each country is represented by a blue dot, and light grey segments extend from a baseline of 40 years to the respective life expectancy value. The countries are ordered by life expectancy on the y-axis. The plot is customized with a minimal theme, and grid lines are removed for a cleaner appearance.

The plot showcases the life expectancy of various Asian countries in 2007, revealing significant disparities. Japan, Hong Kong, China, and Israel have the highest life expectancies, exceeding 80 years, while Afghanistan has the lowest at around 40 years. Countries like Singapore, South Korea, and Taiwan fall in the mid-70s to low-80s range. The data indicates that more developed or wealthier nations tend to have higher life expectancies compared to less developed or conflict-affected countries, highlighting the broad range of life expectancies within Asia.

```
data(gapminder, package="gapminder")

# subset Asian countries in 2007
library(dplyr)
plotdata <- gapminder %>%
  filter(continent == "Asia" &
           year == 2007)


ggplot(plotdata, aes(x=lifeExp,
                     y=reorder(country, lifeExp))) +
  geom_point(color="blue", size = 2) +
  geom_segment(aes(x = 40,
```

```
              xend = lifeExp,
              y = reorder(country, lifeExp),
              yend = reorder(country, lifeExp)),
              color = "lightgrey") +
labs (x = "Life Expectancy (years)",
      y = "",
      title = "Life Expectancy by Country",
      subtitle = "GapMinder data for Asia - 2007") +
theme_minimal() +
theme(panel.grid.major = element_blank(),
      panel.grid.minor = element_blank())
```