NFL_Betting

Kaden Kozak

2023-11-20

Background: This data set is the lines of the NFL spreads and over/unders since 1979.The spread of the game is considered the projected margin of victory by the favorited team and the over/under is the total projected points to be scored in the game by both teams.

Ex: if the Vikings are playing the Packers and the spread is Vikings -3 they are projected to win the game by 3 points and they are favorites and Green Bay is the underdog, so if the Vikings win by 7 the favorite team covers and if they only win by one in this instance the underdog Packers cover .

What I am trying to do with this data is find certain groups or categories that cover the spread more often than the 50/50 likelyhood that the line is set at. I did not look at specific teams because I feel it would not have a good at seeing trends because teams are constantly changing especially over a long period of time.

```
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
```

```
library(forcats)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(readxl)
spreadspoke_scores <- read_excel("spreadspoke_scores.xlsx")
```

```
#making the team names uniform
spreadspoke_scores[spreadspoke_scores$team_home=='Baltimore Colts',"team_home"]<-'Indianapolis Colts'
spreadspoke_scores[spreadspoke_scores$team_away=='Baltimore Colts',"team_away"]<-'Indianapolis Colts'
spreadspoke_scores[spreadspoke_scores$team_home=='Houston Oilers',"team_home"]<-'Tennessee Titans'
spreadspoke_scores[spreadspoke_scores$team_away=='Houston Oilers',"team_away"]<-'Tennessee Titans'
spreadspoke_scores[spreadspoke_scores$team_home=='Tennessee Oilers',"team_home"]<-'Tennessee Titans'
spreadspoke_scores[spreadspoke_scores$team_away=='Tennessee Oilers',"team_away"]<-'Tennessee Titans'
spreadspoke_scores[spreadspoke_scores$team_home=='Los Angeles Raiders',"team_home"]<-'Oakland Raiders'
spreadspoke_scores[spreadspoke_scores$team_away=='Los Angeles Raiders',"team_away"]<-'Oakland Raiders'
spreadspoke_scores[spreadspoke_scores$team_home=='Phoenix Cardinals',"team_home"]<-'Arizona Cardinals'
spreadspoke_scores[spreadspoke_scores$team_away=='Phoenix Cardinals',"team_away"]<-'Arizona Cardinals'
spreadspoke_scores[spreadspoke_scores$team_home=='San Diego Chargers',"team_home"]<-'Los Angeles Charge
spreadspoke_scores[spreadspoke_scores$team_away=='San Diego Chargers',"team_away"]<-'Los Angeles Charge
spreadspoke_scores[spreadspoke_scores$team_home=='St. Louis Cardinals',"team_home"]<-'Arizona Cardinals
spreadspoke_scores[spreadspoke_scores$team_away=='St. Louis Cardinals',"team_away"]<-'Arizona Cardinals
spreadspoke_scores[spreadspoke_scores$team_home=='St. Louis Rams',"team_home"]<-'Los Angeles Rams'
spreadspoke_scores[spreadspoke_scores$team_away=='St. Louis Rams',"team_away"]<-'Los Angeles Rams'
spreadspoke_scores[spreadspoke_scores$team_home=='Washington Commanders',"team_home"]<-'Washington Redsk
spreadspoke_scores[spreadspoke_scores$team_away=='Washington Commanders',"team_away"]<-'Washington Redsk
spreadspoke_scores[spreadspoke_scores$team_home=='Washington Football Team',"team_home"]<-'Washington R
spreadspoke_scores[spreadspoke_scores$team_away=='Washington Football Team',"team_away"]<-'Washington R
spreadspoke_scores[spreadspoke_scores$team_home=='Las Vegas Raiders',"team_home"]<-'Oakland Raiders'
spreadspoke_scores[spreadspoke_scores$team_away=='Las Vegas Raiders',"team_away"]<-'Oakland Raiders'
```

```
spreadspoke_scores[spreadspoke_scores$team_home=='Arizona Cardinals',"team_home"]<-'ARI'
spreadspoke_scores[spreadspoke_scores$team_away=='Arizona Cardinals',"team_away"]<-'ARI'
spreadspoke_scores[spreadspoke_scores$team_home=='Atlanta Falcons',"team_home"]<-'ATL'
spreadspoke_scores[spreadspoke_scores$team_away=='Atlanta Falcons',"team_away"]<-'ATL'
spreadspoke_scores[spreadspoke_scores$team_home=='Baltimore Ravens',"team_home"]<-'BAL'
spreadspoke_scores[spreadspoke_scores$team_away=='Baltimore Ravens',"team_away"]<-'BAL'
spreadspoke_scores[spreadspoke_scores$team_home=='Buffalo Bills',"team_home"]<-'BUF'
spreadspoke_scores[spreadspoke_scores$team_away=='Buffalo Bills',"team_away"]<-'BUF'
spreadspoke_scores[spreadspoke_scores$team_home=='Carolina Panthers',"team_home"]<-'CAR'
spreadspoke_scores[spreadspoke_scores$team_away=='Carolina Panthers',"team_away"]<-'CAR'
spreadspoke_scores[spreadspoke_scores$team_home=='Chicago Bears',"team_home"]<-'CHI'
spreadspoke_scores[spreadspoke_scores$team_away=='Chicago Bears',"team_away"]<-'CHI'
spreadspoke_scores[spreadspoke_scores$team_home=='Cincinnati Bengals',"team_home"]<-'CIN'
spreadspoke_scores[spreadspoke_scores$team_away=='Cincinnati Bengals',"team_away"]<-'CIN'
spreadspoke_scores[spreadspoke_scores$team_home=='Cleveland Browns',"team_home"]<-'CLE'
spreadspoke_scores[spreadspoke_scores$team_away=='Cleveland Browns',"team_away"]<-'CLE'
spreadspoke_scores[spreadspoke_scores$team_home=='Dallas Cowboys',"team_home"]<-'DAL'
spreadspoke_scores[spreadspoke_scores$team_away=='Dallas Cowboys',"team_away"]<-'DAL'
spreadspoke_scores[spreadspoke_scores$team_home=='Denver Broncos',"team_home"]<-'DEN'
spreadspoke_scores[spreadspoke_scores$team_away=='Denver Broncos',"team_away"]<-'DEN'
spreadspoke_scores[spreadspoke_scores$team_home=='Detroit Lions',"team_home"]<-'DET'
spreadspoke_scores[spreadspoke_scores$team_away=='Detroit Lions',"team_away"]<-'DET'
spreadspoke_scores[spreadspoke_scores$team_home=='Green Bay Packers',"team_home"]<-'GB'
spreadspoke_scores[spreadspoke_scores$team_away=='Green Bay Packers',"team_away"]<-'GB'
spreadspoke_scores[spreadspoke_scores$team_home=='Houston Texans',"team_home"]<-'HOU'
spreadspoke_scores[spreadspoke_scores$team_away=='Houston Texans',"team_away"]<-'HOU'
spreadspoke_scores[spreadspoke_scores$team_home=='Indianapolis Colts',"team_home"]<-'IND'
spreadspoke_scores[spreadspoke_scores$team_away=='Indianapolis Colts',"team_away"]<-'IND'
spreadspoke_scores[spreadspoke_scores$team_home=='Jacksonville Jaguars',"team_home"]<-'JAX'
```

```
spreadspoke_scores[spreadspoke_scores$team_away=='Jacksonville Jaguars',"team_away"]<-'JAX'
spreadspoke_scores[spreadspoke_scores$team_home=='Kansas City Chiefs',"team_home"]<-'KC'
spreadspoke_scores[spreadspoke_scores$team_away=='Kansas City Chiefs',"team_away"]<-'KC'
spreadspoke_scores[spreadspoke_scores$team_home=='Los Angeles Chargers',"team_home"]<-'LAC'
spreadspoke_scores[spreadspoke_scores$team_away=='Los Angeles Chargers',"team_away"]<-'LAC'
spreadspoke_scores[spreadspoke_scores$team_home=='Los Angeles Rams',"team_home"]<-'LAR'
spreadspoke_scores[spreadspoke_scores$team_away=='Los Angeles Rams',"team_away"]<-'LAR'
spreadspoke_scores[spreadspoke_scores$team_home=='Miami Dolphins',"team_home"]<-'MIA'
spreadspoke_scores[spreadspoke_scores$team_away=='Miami Dolphins',"team_away"]<-'MIA'
spreadspoke_scores[spreadspoke_scores$team_home=='Minnesota Vikings',"team_home"]<-'MIN'
spreadspoke_scores[spreadspoke_scores$team_away=='Minnesota Vikings',"team_away"]<-'MIN'
spreadspoke_scores[spreadspoke_scores$team_home=='New England Patriots',"team_home"]<-'NE'
spreadspoke_scores[spreadspoke_scores$team_away=='New England Patriots',"team_away"]<-'NE'
spreadspoke_scores[spreadspoke_scores$team_home=='New Orleans Saints',"team_home"]<-'NO'
spreadspoke_scores[spreadspoke_scores$team_away=='New Orleans Saints',"team_away"]<-'NO'
spreadspoke_scores[spreadspoke_scores$team_home=='New York Giants',"team_home"]<-'NYG'
spreadspoke_scores[spreadspoke_scores$team_away=='New York Giants',"team_away"]<-'NYG'
spreadspoke_scores[spreadspoke_scores$team_home=='New York Jets',"team_home"]<-'NYJ'
spreadspoke_scores[spreadspoke_scores$team_away=='New York Jets',"team_away"]<-'NYJ'
spreadspoke_scores[spreadspoke_scores$team_home=='Oakland Raiders',"team_home"]<-'LVR'
spreadspoke_scores[spreadspoke_scores$team_away=='Oakland Raiders',"team_away"]<-'LVR'
spreadspoke_scores[spreadspoke_scores$team_home=='Philadelphia Eagles',"team_home"]<-'PHI'
spreadspoke_scores[spreadspoke_scores$team_away=='Philadelphia Eagles',"team_away"]<-'PHI'
spreadspoke_scores[spreadspoke_scores$team_home=='Pittsburgh Steelers',"team_home"]<-'PIT'
spreadspoke_scores[spreadspoke_scores$team_away=='Pittsburgh Steelers',"team_away"]<-'PIT'
spreadspoke_scores[spreadspoke_scores$team_home=='San Francisco 49ers',"team_home"]<-'SF'
spreadspoke_scores[spreadspoke_scores$team_away=='San Francisco 49ers',"team_away"]<-'SF'
spreadspoke_scores[spreadspoke_scores$team_home=='Seattle Seahawks',"team_home"]<-'SEA'
spreadspoke_scores[spreadspoke_scores$team_away=='Seattle Seahawks',"team_away"]<-'SEA'
spreadspoke_scores[spreadspoke_scores$team_home=='Tampa Bay Buccaneers',"team_home"]<-'TB'
spreadspoke_scores[spreadspoke_scores$team_away=='Tampa Bay Buccaneers',"team_away"]<-'TB'
spreadspoke_scores[spreadspoke_scores$team_home=='Tennessee Titans',"team_home"]<-'TEN'
spreadspoke_scores[spreadspoke_scores$team_away=='Tennessee Titans',"team_away"]<-'TEN'
spreadspoke_scores[spreadspoke_scores$team_home=='Washington Redskins',"team_home"]<-'WAS'
spreadspoke_scores[spreadspoke_scores$team_away=='Washington Redskins',"team_away"]<-'WAS'
```

```
# Create new columns 'home_division' and 'away_division' and initialize them as NA
spreadspoke_scores$home_division <- NA
spreadspoke_scores$away_division <- NA


# Define divisions based on the shortened team names
afc_east <- c('BUF', 'MIA', 'NE', 'NYJ')
afc_north <- c('BAL', 'CIN', 'CLE', 'PIT')
afc_south <- c('HOU', 'IND', 'JAX', 'TEN')
afc_west <- c('DEN', 'KC', 'LAC', 'LVR')
nfc_east <- c('DAL', 'NYG', 'PHI', 'WAS')
nfc_north <- c('CHI', 'DET', 'GB', 'MIN')
nfc_south <- c('ATL', 'CAR', 'NO', 'TB')
nfc_west <- c('ARI', 'LAR', 'SF', 'SEA')

spreadspoke_scores <- spreadspoke_scores %>%
  mutate(
```

```
    home_division = case_when(
      team_home %in% afc_east ~ 'AFC East',
      team_home %in% afc_north ~ 'AFC North',
      team_home %in% afc_south ~ 'AFC South',
      team_home %in% afc_west ~ 'AFC West',
      team_home %in% nfc_east ~ 'NFC East',
      team_home %in% nfc_north ~ 'NFC North',
      team_home %in% nfc_south ~ 'NFC South',
      team_home %in% nfc_west ~ 'NFC West',
      TRUE ~ NA_character_
    ),
    away_division = case_when(
      team_away %in% afc_east ~ 'AFC East',
      team_away %in% afc_north ~ 'AFC North',
      team_away %in% afc_south ~ 'AFC South',
      team_away %in% afc_west ~ 'AFC West',
      team_away %in% nfc_east ~ 'NFC East',
      team_away %in% nfc_north ~ 'NFC North',
      team_away %in% nfc_south ~ 'NFC South',
      team_away %in% nfc_west ~ 'NFC West',
      TRUE ~ NA_character_
    )
  )
```

```
#create column when divisional matchups take place
spreadspoke_scores <- spreadspoke_scores %>%
  mutate(
    matchup_type = case_when(
      home_division == away_division ~ home_division,
      TRUE ~ 'Non-Divisional'
    )
  )
```

```
#spread difference results
spreadspoke_scores <- spreadspoke_scores %>%
  mutate(spread_difference = case_when(
    team_home == team_favorite_id ~ score_away - score_home,
    team_away == team_favorite_id ~ score_home - score_away,
    team_favorite_id == "PICK" ~ score_home - score_away,
    TRUE ~ NA_real_
  ))
```

```
#making column to show the result of spread
spreadspoke_scores <- spreadspoke_scores %>%
  mutate(spread_results = case_when(
    spread_difference < spread_favorite ~ "Favorite",
    spread_difference > spread_favorite ~ "Underdog",
    spread_difference == spread_favorite ~ "Push",
    TRUE ~ NA_character_
  ))
```

```r
#making row for which team is favorited
spreadspoke_scores <- spreadspoke_scores %>%
  mutate(location_favorite = case_when(
    team_favorite_id == "PICK" ~ "No Favorite",
    team_home == team_favorite_id ~ "Home",
    team_away == team_favorite_id ~ "Away",
    TRUE ~ "None"
  ))
```

```r
#results of over under code
spreadspoke_scores$over_under_results <- case_when(
  spreadspoke_scores$over_under_line < spreadspoke_scores$total_points ~ "Over",
  spreadspoke_scores$over_under_line > spreadspoke_scores$total_points ~ "Under",
  TRUE ~ "PUSH"
)
```

```r
#making column to show the result of spread
spreadspoke_scores <- spreadspoke_scores %>%
  mutate(home_spread =
           ifelse(team_favorite_id==team_home,spread_favorite,-1*spread_favorite)
  )
```

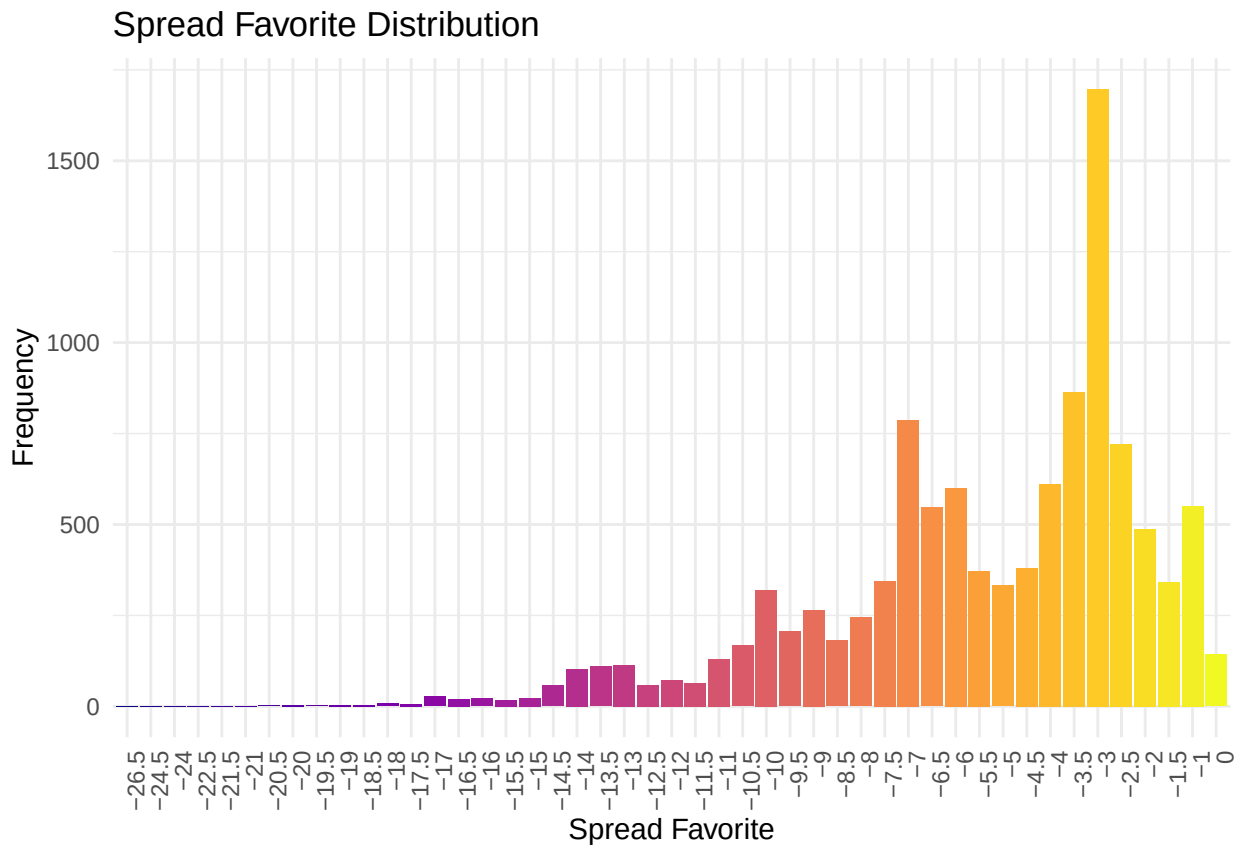
```r
#home team cover column
spreadspoke_scores <- spreadspoke_scores %>%
  mutate(home_cover = case_when(
    score_difference < home_spread  ~ "Covered",
    score_difference > home_spread  ~ "Not Covered",
    spread_results == "Push" ~ "Push",
    TRUE ~ NA_character_
  ))
```

```r
# Create a new variable for spread categories
spreadspoke_scores <- spreadspoke_scores %>%
  mutate(spread_category = case_when(
    spread_favorite <= -7.5 ~ "Large Spread",
    spread_favorite <= -3.5 & spread_favorite >= -7 ~ "Medium Spread",
    spread_favorite <= 0 & spread_favorite >= -3 ~ "Small Spread",
    TRUE ~ "No Spread"  # If none of the conditions match
  ))
```

```r
#Create new variable for over_under_category
spreadspoke_scores <- spreadspoke_scores %>%
  mutate(over_under_category = case_when(
    over_under_line <= 36.5 ~ "Low",
    between(over_under_line, 37, 47) ~ "Average",
    over_under_line >= 47.5 ~ "High",
    TRUE ~ NA_character_
  ))
```

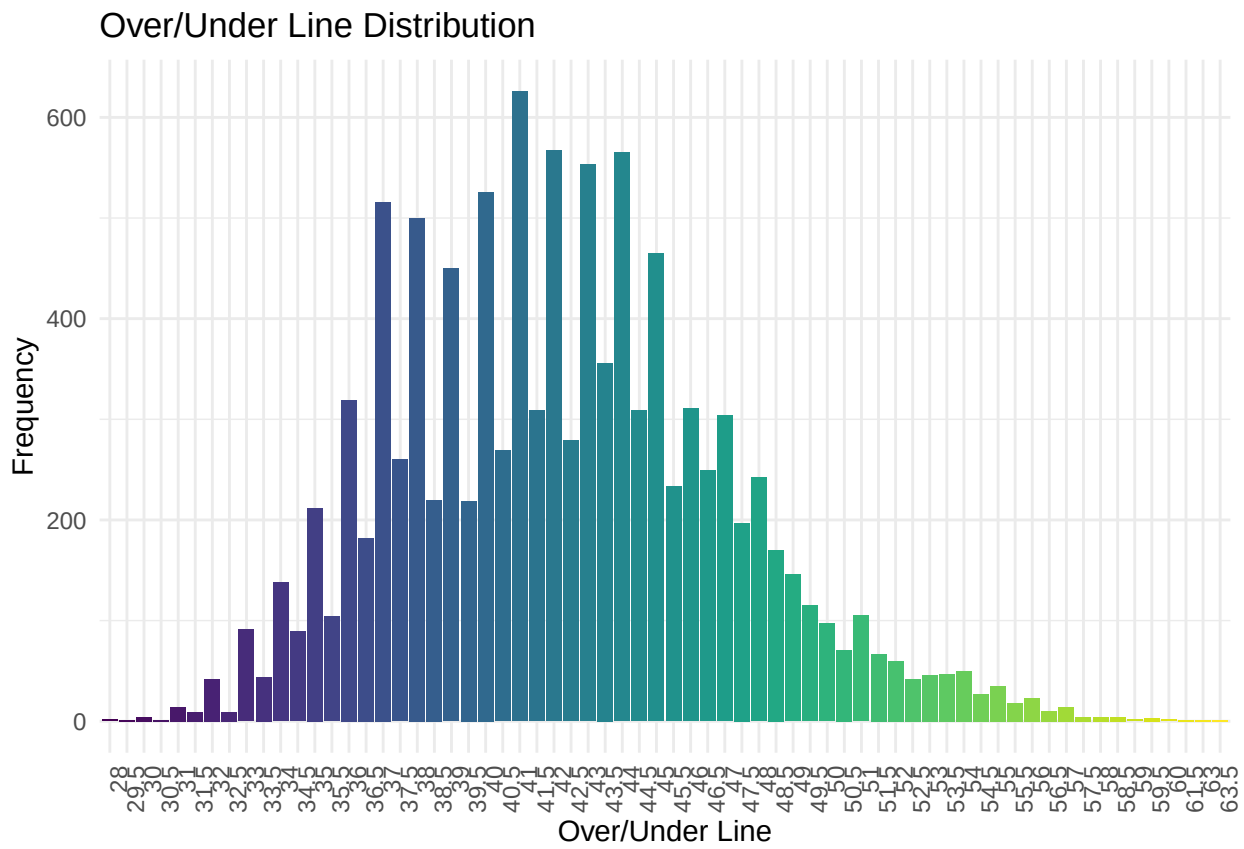
```r
ggplot(spreadspoke_scores, aes(x = as.factor(spread_favorite), fill = as.factor(spread_favorite))) +
  geom_bar() +
  labs(title = "Spread Favorite Distribution",
```

```
      x = "Spread Favorite",
      y = "Frequency") +
  coord_cartesian(ylim = c(0, max(table(spreadspoke_scores$spread_favorite)))) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1),
        legend.position = "none") +  # Remove legend since color represents spread_favorite
  scale_fill_viridis_d(option="plasma")  # Use the "viridis" color palette
```



The highest frequencies are at 1, 3, 3.5, and 7.Most of the spreads are by 3 and 7 becasue those are the increments teams score at most often.

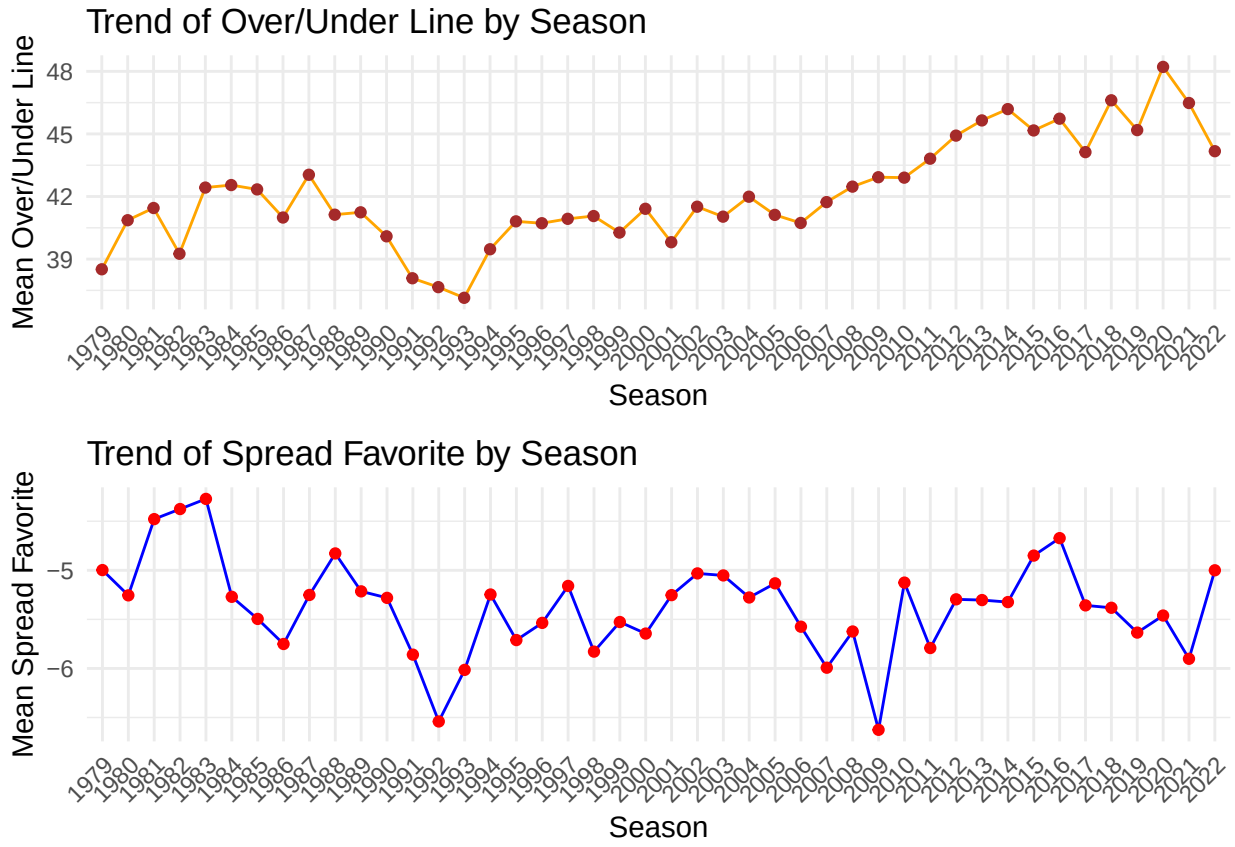
```
# Filter the dataframe to remove NA values and the value 36.6
filtered_over_under_line <- spreadspoke_scores[!is.na(spreadspoke_scores$over_under_line) & spreadspoke_

# Create a ggplot bar plot with color and improved aesthetics using the "viridis" palette
ggplot(filtered_over_under_line, aes(x = as.factor(over_under_line), fill = as.factor(over_under_line)))
  geom_bar(width = 0.9) +
  labs(title = "Over/Under Line Distribution",
       x = "Over/Under Line",
       y = "Frequency") +
  coord_cartesian(ylim = c(0, max(table(filtered_over_under_line$over_under_line)))) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 1),  # Adjust angle and justification
        legend.position = "none") +
  scale_fill_viridis_d()
```



Over under lines have much a pretty normal shaped distrabution, but the whole number spreads are much more popular and it has a longer right tale.

```
# Your existing code for the second two plots
season_summary_over_under <- spreadspoke_scores %>%
  group_by(schedule_season) %>%
  summarize(mean_over_under_line = mean(over_under_line, na.rm = TRUE))

trend_over_under_plot <- ggplot(season_summary_over_under, aes(x = as.factor(schedule_season), y = mean_
  geom_line(color = "orange") +
  geom_point(color = "brown") +
  labs(title = "Trend of Over/Under Line by Season",
       x = "Season",
       y = "Mean Over/Under Line") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

season_summary_spread_favorite <- spreadspoke_scores %>%
  group_by(schedule_season) %>%
  summarize(mean_spread_favorite = mean(spread_favorite, na.rm = TRUE))

trend_spread_favorite_plot <- ggplot(season_summary_spread_favorite, aes(x = as.factor(schedule_season)
  geom_line(color = "blue") +
  geom_point(color = "red") +
  labs(title = "Trend of Spread Favorite by Season",
       x = "Season",
```

```
       y = "Mean Spread Favorite") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

# Arrange the new plots using gridExtra in a 2x1 grid
grid.arrange(
  trend_over_under_plot,
  trend_spread_favorite_plot,
  ncol = 1
)
```



lower over/under= better defenses

closer spread= more competativeness

With the over/under there was a dip down in the early 90's but was on a steady uptrend until the last few years it has gone down. There is not much to see at with the spread besides the 1992 and 2009 having large mean spreads throughout the year.

```
# Assuming the two ggplot objects are stored in plot1 and plot2
spread <- ggplot(spreadspoke_scores, aes(x = "", fill = spread_results)) +
  geom_bar(width = 1) +
  coord_polar(theta = "y") +
  stat_count(aes(label = sprintf("%d\n%.1f%%", ..count.., 100 * ..count../sum(..count..))),
             position = position_stack(vjust = 0.5)) +
  geom_text(stat = "count", aes(label = paste0(sprintf("%.2f%%", prop.table(..count..) * 100))),
            position = position_stack(vjust = .5), color = "black", size = 3) +
```

```
  labs(title = "Spread Results Distribution",
       x = NULL,
       y = NULL) +
  theme_minimal() +
  scale_fill_manual(values = c("Favorite" = "green", "Underdog" = "red", "Push" = "blue"))
```

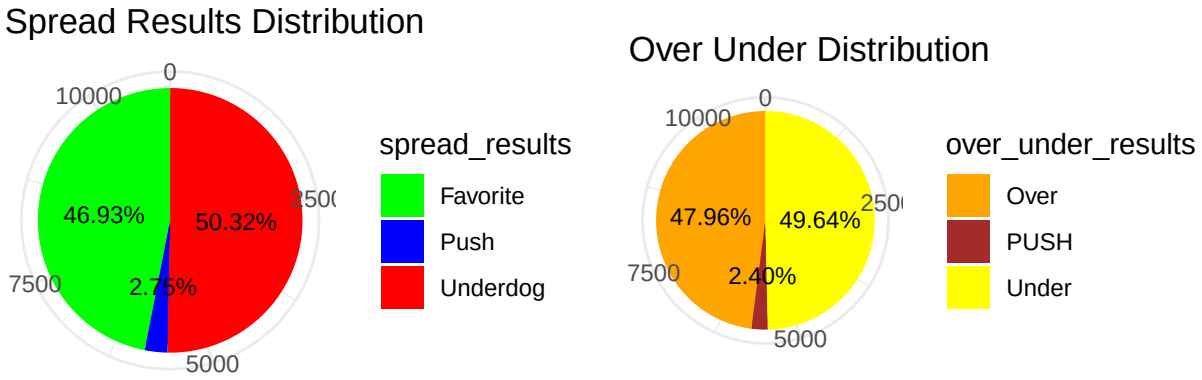
```
## Warning in stat_count(aes(label = sprintf("%d\n%.1f%%", ..count.., 100 * :
## Ignoring unknown aesthetics: label
```

```
over_under <- ggplot(spreadspoke_scores, aes(x = "", fill = over_under_results)) +
  geom_bar(width = 1) +
  coord_polar(theta = "y") +
  stat_count(aes(label = sprintf("%d\n%.1f%%", ..count.., 100 * ..count../sum(..count..))),
             position = position_stack(vjust = 0.5)) +
  geom_text(stat = "count", aes(label = paste0(sprintf("%.2f%%", prop.table(..count..) * 100))),
            position = position_stack(vjust = .5), color = "black", size = 3) +
  labs(title = "Over Under Distribution",
       x = NULL,
       y = NULL) +
  theme_minimal() +
  scale_fill_manual(values = c("Over" = "orange", "Under" = "yellow", "PUSH" = "brown"))
```

```
## Warning in stat_count(aes(label = sprintf("%d\n%.1f%%", ..count.., 100 * :
## Ignoring unknown aesthetics: label
```

```
# Arrange plots side by side using grid.arrange
grid.arrange(spread, over_under, ncol = 2)
```

```
## Warning: The dot-dot notation (`..count..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(count)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



Both the spread and the over/under are very close to 50/50. With the under and underdog having slight edges.

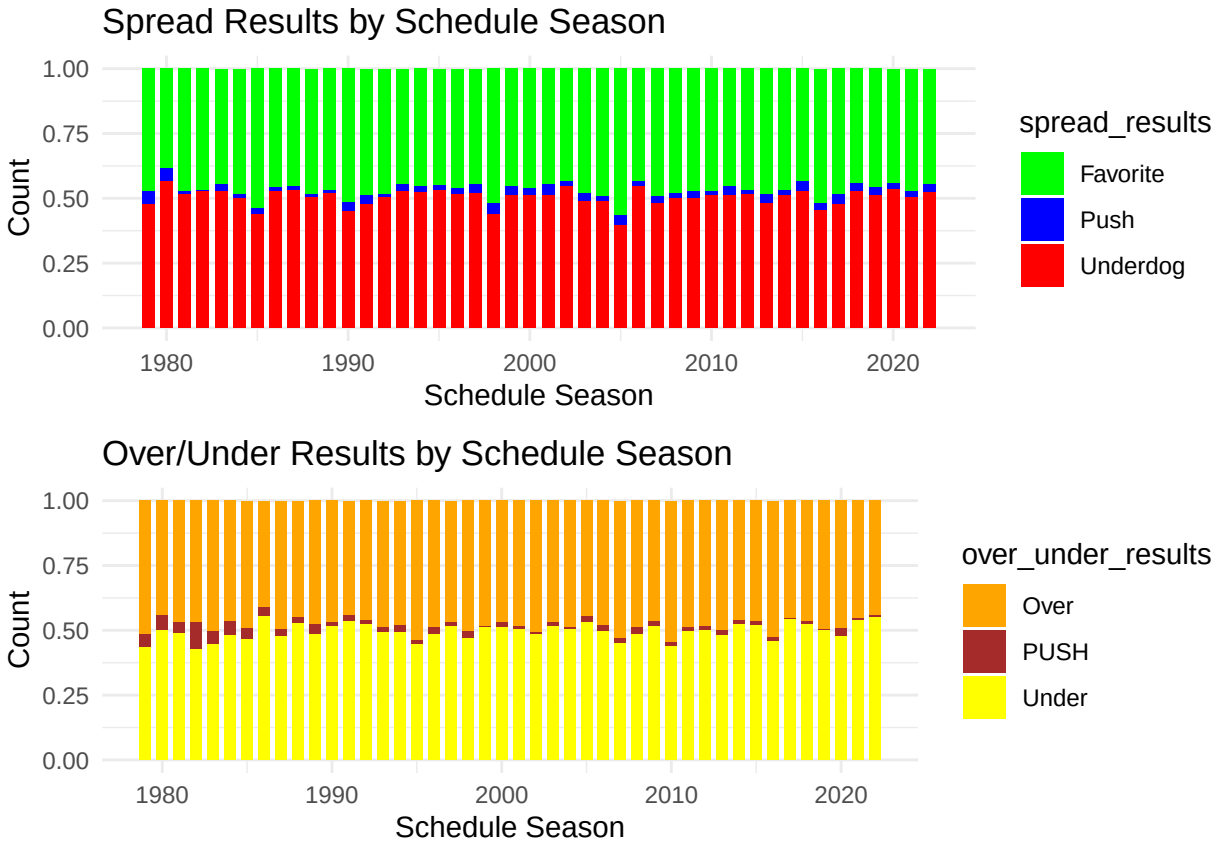
```
# Exclude "Push" category and focus on "Over" and "Under"
filtered_data <- spreadspoke_scores[spreadspoke_scores$over_under_results %in% c("Over", "Under"), ]

# Perform a proportion test
result <- prop.test(table(filtered_data$over_under_results))

# Display the test result
print(result)
```

```
##
##  1-sample proportions test with continuity correction
##
## data:  table(filtered_data$over_under_results), null probability 0.5
## X-squared = 3.1491, df = 1, p-value = 0.07597
## alternative hypothesis: true p is not equal to 0.5
## 95 percent confidence interval:
##  0.4819045 0.5008940
## sample estimates:
##         p
## 0.4913961
```

The p-value (0.07597) is greater than the typical significance level of 0.05. Therefore, there isn't enough evidence to reject the null hypothesis. The observed proportion (p-hat)is 0.4913961, which is close to 0.5.

The confidence interval also includes 0.5, supporting the idea that the observed proportion is not significantly different from 0.5.

```
# Create the first plot
spread_plot <- ggplot(spreadspoke_scores, aes(x = schedule_season, fill = spread_results)) +
  geom_bar(position = "fill", width = 0.7) +
  labs(title = "Spread Results by Schedule Season",
       x = "Schedule Season",
       y = "Count") +
  theme_minimal() +
  scale_fill_manual(values = c("Favorite" = "green", "Underdog" = "red", "Push" = "blue"))

# Create the second plot
over_under_plot <- ggplot(spreadspoke_scores, aes(x = schedule_season, fill = over_under_results)) +
  geom_bar(position = "fill", width = 0.7) +
  labs(title = "Over/Under Results by Schedule Season",
       x = "Schedule Season",
       y = "Count") +
  theme_minimal() +
  scale_fill_manual(values = c("Over" = "orange", "Under" = "yellow", "PUSH" = "brown"))

# Arrange the plots using gridExtra
grid.arrange(spread_plot, over_under_plot, ncol = 1)
```



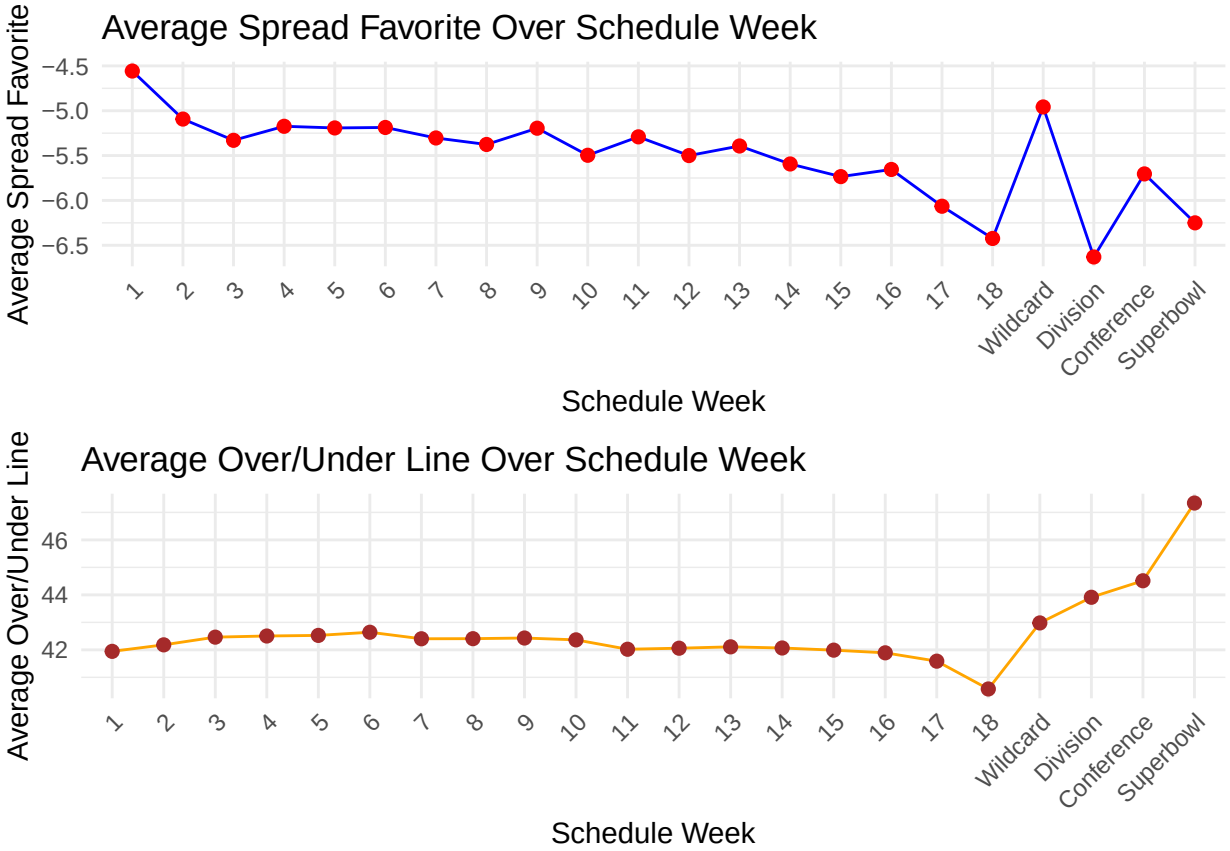
All years are pretty close to 50/50 besides a couple years where at highest it at 55% to the underdogs. Overall the under and underdogs have very slight edges in most years.

```r
# Your existing code for the first plot
custom_order <- c("1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15", "16"
spreadspoke_scores$schedule_week <- fct_relevel(spreadspoke_scores$schedule_week, custom_order)

spread_plot <- ggplot(spreadspoke_scores, aes(x = schedule_week, y = spread_favorite, group = 1)) +
  stat_summary(fun.y = mean, geom = "line", color = "blue") +
  stat_summary(fun.y = mean, geom = "point", color = "red", size = 2) +
  labs(title = "Average Spread Favorite Over Schedule Week",
       x = "Schedule Week",
       y = "Average Spread Favorite") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
## Warning: The ‘fun.y‘ argument of ‘stat_summary()‘ is deprecated as of ggplot2 3.3.0.
## i Please use the ‘fun‘ argument instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

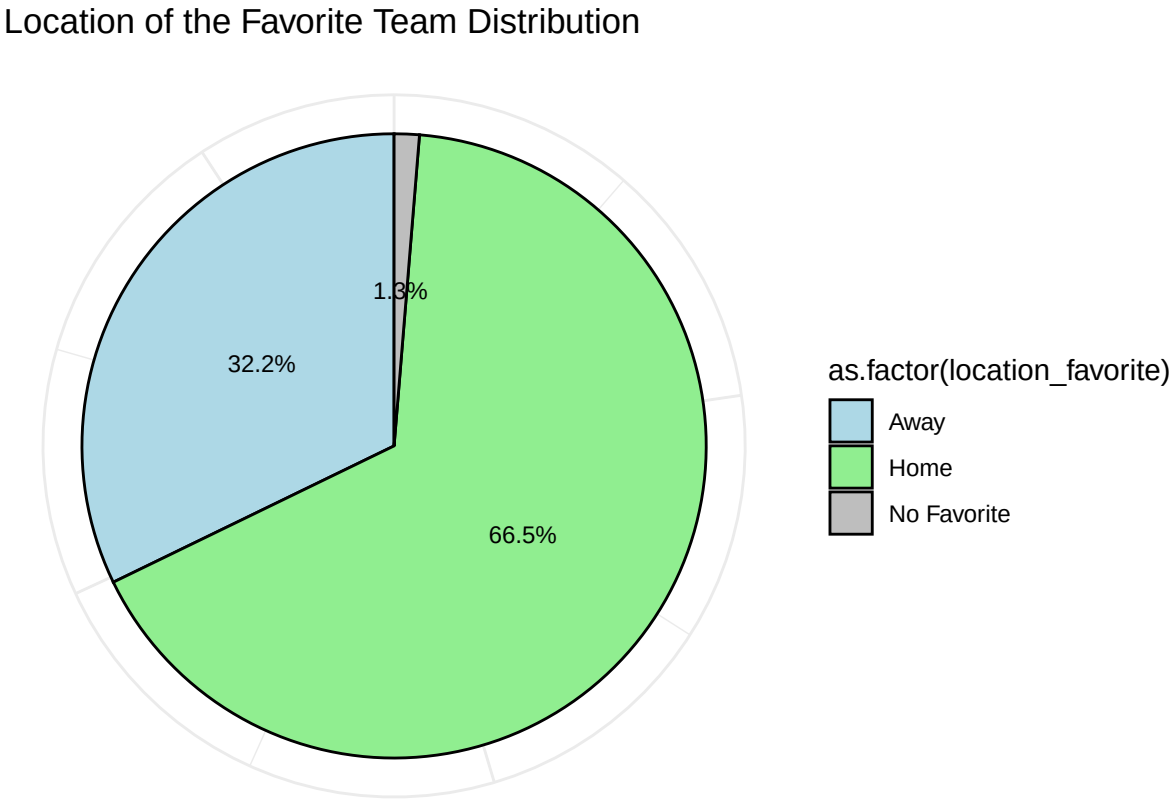
```r
# Your existing code for the second plot
over_under_plot <- ggplot(spreadspoke_scores, aes(x = schedule_week, y = over_under_line, group = 1)) +
  stat_summary(fun.y = mean, geom = "line", color = "orange") +
  stat_summary(fun.y = mean, geom = "point", color = "brown", size = 2) +
  labs(title = "Average Over/Under Line Over Schedule Week",
       x = "Schedule Week",
       y = "Average Over/Under Line") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

# Arrange the plots using gridExtra
grid.arrange(spread_plot, over_under_plot, ncol = 1)
```

```
## Warning: Removed 62 rows containing non-finite values (‘stat_summary()‘).

## Warning: Removed 62 rows containing non-finite values (‘stat_summary()‘).
```



Looking at the spread it is lowest on week one and trends to a larger spread throughout the season, likely because it is more clear what teams are capable of. Over/under is much more consistent throughout the year until week 18 which is when lots of teams will sit thier starter if they clinch playoffs.

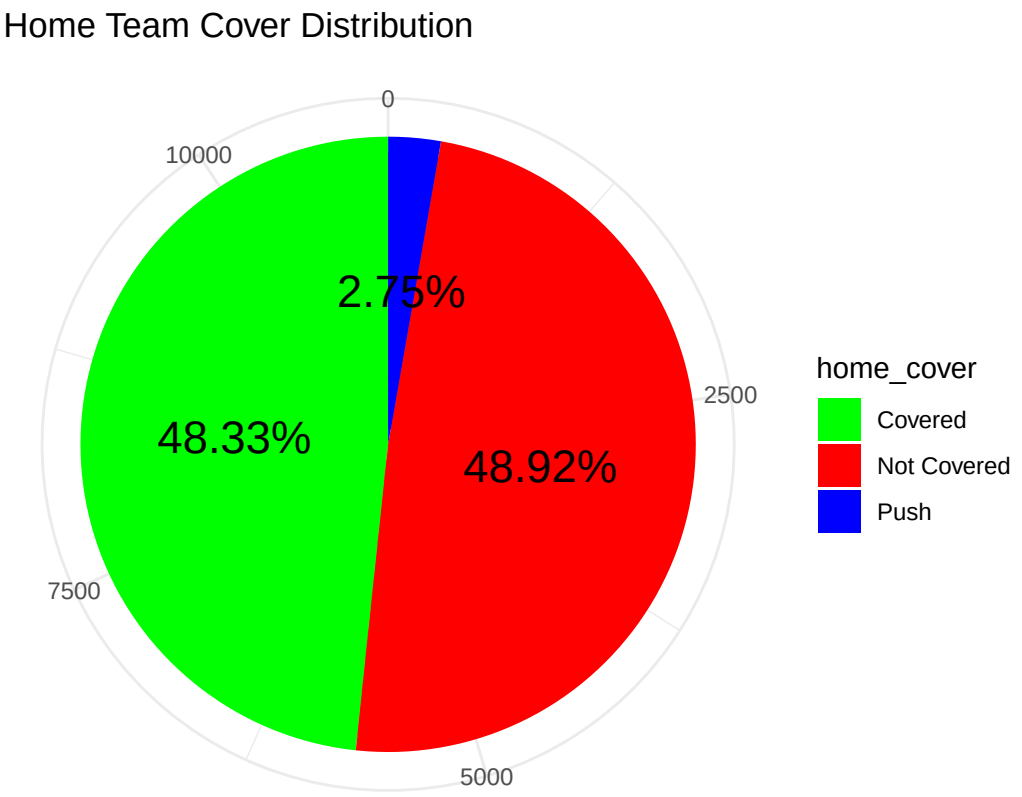
```
# Create a ggplot pie chart for 'location_favorite'
ggplot(spreadspoke_scores, aes(x = "", fill = as.factor(location_favorite))) +
  geom_bar(width = 1, color = "black") +
  coord_polar(theta = "y") +
  labs(title = "Location of the Favorite Team Distribution",
       x = NULL,
       y = NULL) +
  theme_minimal() +
  theme(axis.text.x = element_blank()) +  # Hide x-axis labels
  scale_fill_manual(values = c("Home" = "lightgreen", "Away" = "lightblue","No Favorite"="grey")) +
  geom_text(stat = "count", aes(label = paste0(sprintf("%.1f%%", prop.table(..count..) * 100))),
            position = position_stack(vjust = 0.5), color = "black", size = 3)
```



Home teams are the favorited team 66.5% of the time.

```
# Create a ggplot pie chart with counts and percentages for 'home_cover'
ggplot(spreadspoke_scores, aes(x = "", fill = home_cover)) +
  geom_bar(width = 1) +
  coord_polar(theta = "y") +
  stat_count(aes(label = sprintf("%d\n%.1f%%", ..count.., 100 * ..count../sum(..count..))),
             position = position_stack(vjust = 0)) +
  geom_text(stat = "count", aes(label = paste0(sprintf("%.2f%%", prop.table(..count..) * 100))),
            position = position_stack(vjust = 0.5), color = "black", size = 6) +  # Adjust size
  labs(title = "Home Team Cover Distribution",
       x = NULL,
       y = NULL) +
  theme_minimal() +
  scale_fill_manual(values = c("Covered" = "green", "Not Covered" = "red", "Push" = "blue"))
```

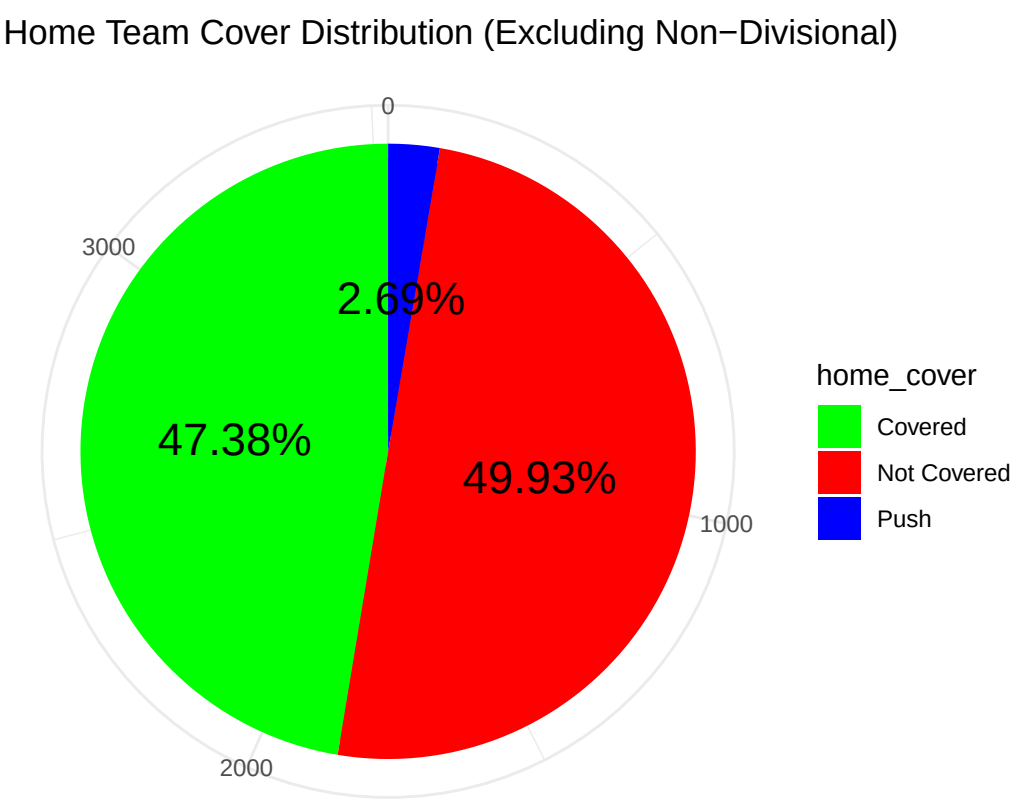
```
## Warning in stat_count(aes(label = sprintf("%d\n%.1f%%", ..count.., 100 * :
## Ignoring unknown aesthetics: label
```



With the different distribution from the graph before I thought there might be some differences in the home teams and if they covered or not but there was nearly no difference.

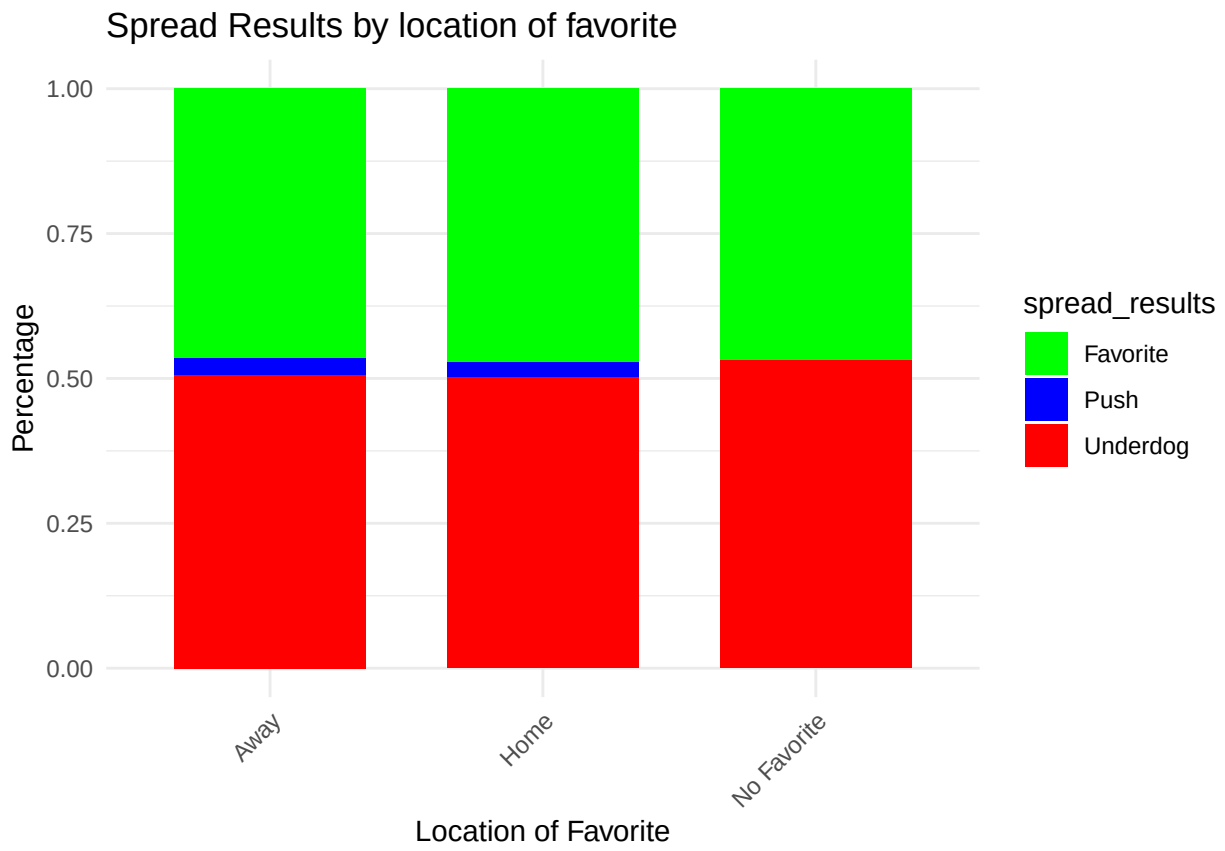
```
# Subset the data to exclude Non-divisional matchups
subset_data <- spreadspoke_scores[spreadspoke_scores$matchup_type != "Non-Divisional", ]

# Create a ggplot pie chart with counts and percentages for 'home_cover'
ggplot(subset_data, aes(x = "", fill = home_cover)) +
  geom_bar(width = 1) +
  coord_polar(theta = "y") +
  stat_count(aes(label = sprintf("%d\n%.1f%%", ..count.., 100 * ..count../sum(..count..))),
             position = position_stack(vjust = 0)) +
  geom_text(stat = "count", aes(label = paste0(sprintf("%.2f%%", prop.table(..count..) * 100))),
            position = position_stack(vjust = 0.5), color = "black", size = 6) +  # Adjust size
  labs(title = "Home Team Cover Distribution (Excluding Non-Divisional)",
       x = NULL,
       y = NULL) +
  theme_minimal() +
  scale_fill_manual(values = c("Covered" = "green", "Not Covered" = "red", "Push" = "blue"))
```

```
## Warning in stat_count(aes(label = sprintf("%d\n%.1f%%", ..count.., 100 * :
## Ignoring unknown aesthetics: label
```



In this chart I just looked at divisional opponents and surprisingly home team covers less often, but there is not much of a difference and not big enough to say it is significant.

```
ggplot(spreadspoke_scores, aes(x = location_favorite, fill = spread_results)) +
  geom_bar(position = "fill", width = 0.7) +
  labs(title = "Spread Results by location of favorite",
       x = "Location of Favorite",
       y = "Percentage") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_manual(values = c("Favorite" = "green", "Underdog" = "red", "Push" = "blue"))
```



There is no difference between the home and away with who covers the spread. In no favorite Home is the favorite and Away is Underdog.

```
# Your first plot
spread_plot <- ggplot(spreadspoke_scores, aes(x = spread_category, fill = spread_results)) +
  geom_bar(position = "fill", width = 0.7) +
  labs(title = "Spread Results by Spread Category",
       x = "Spread Category",
       y = "Percentage") +
  theme_minimal() +
  scale_fill_manual(values = c("Favorite" = "green", "Underdog" = "red", "Push" = "blue"))

# Assuming your data frame is named 'spreadspoke_scores'
over_under_category_filtered <- spreadspoke_scores %>%
  filter(!is.na(over_under_category) & over_under_category != "Low 36.5 and below")

# Your second plot
over_under_plot <- ggplot(over_under_category_filtered, aes(x = over_under_category, fill = over_under_
  geom_bar(position = "fill", width = 0.7) +
  labs(title = "Spread Results by Over/Under Category",
       x = "Over/Under Category",
       y = "Percentage") +
  theme_minimal() +
  scale_fill_manual(values = c("Over" = "orange", "Under" = "yellow", "PUSH" = "brown"))

# Arrange the plots using gridExtra
```

```
grid.arrange(spread_plot, over_under_plot, ncol = 1)
```

There is nearly no difference in the spread categories except that the small spread has pushes more often. Underdogs cover slightly more with the medium and large spreads. High over/unders the under covers slightly more and when they are set low the over hits a little more often.

```
#test to see if spread category is independent from spread results
contingency_table <- table(spreadspoke_scores$spread_results,
                           spreadspoke_scores$spread_category)


# Perform chi-squared test
chi_square_result <- chisq.test(contingency_table)

# Print the result
print(chi_square_result)
```

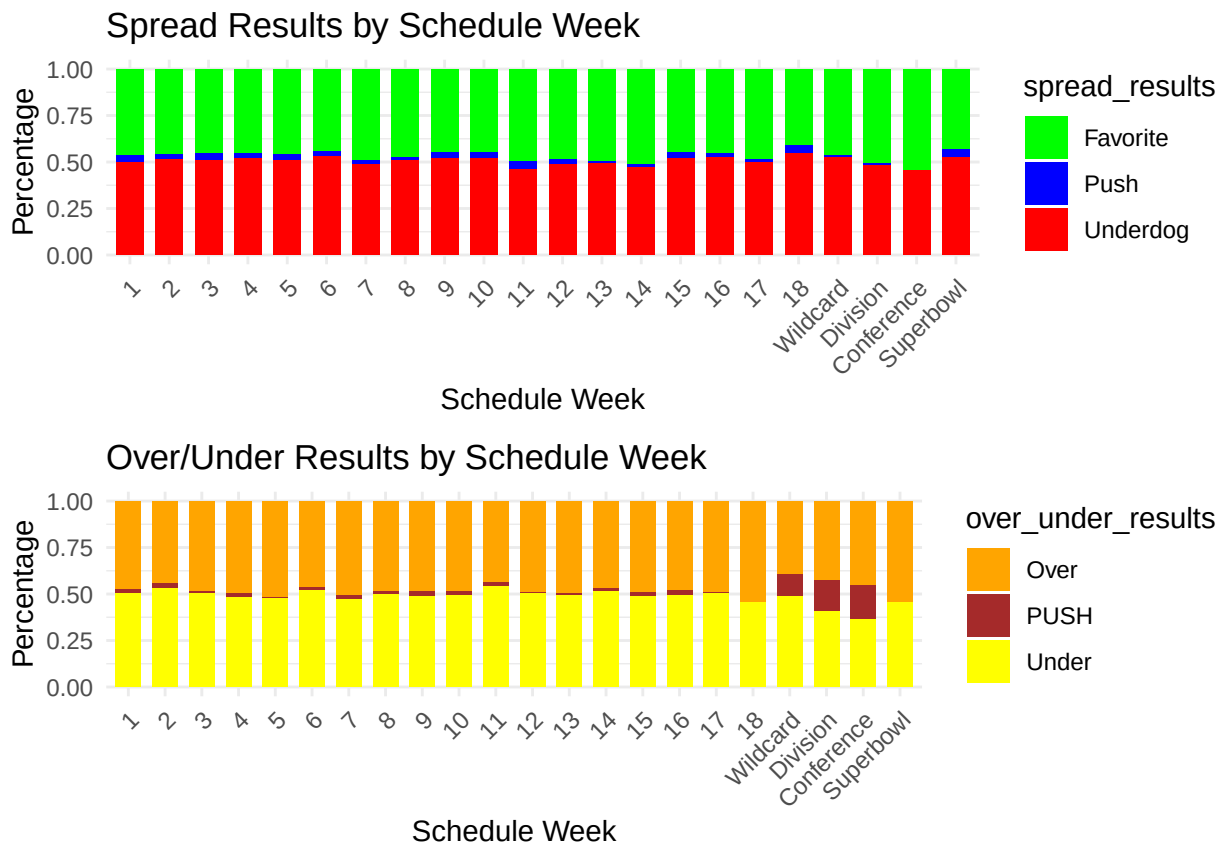
```
##
##  Pearson's Chi-squared test
##
## data:  contingency_table
## X-squared = 85.434, df = 4, p-value < 2.2e-16
```

```r
print(contingency_table)
```

```
##
##            Large Spread Medium Spread Small Spread
##   Favorite         1224          2134         1811
##   Push               37            82          184
##   Underdog         1319          2279         1945
```

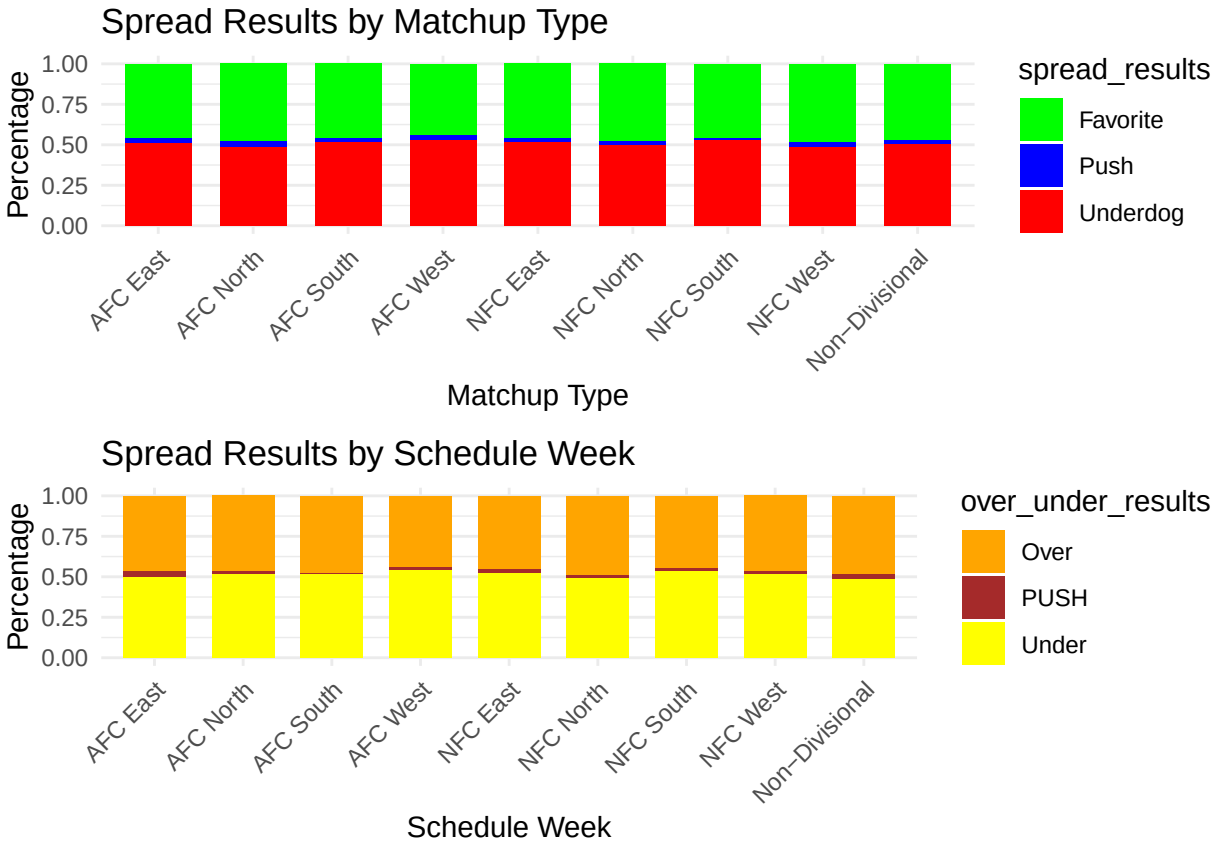
H0: no association or relationship between the two variables Ha: Thereis significant relationship between the two variables Given the very small p-value we would reject the null so there likely is a significant relationship between the distributions given.

```r
# Your first plot
spread_results_plot <- ggplot(spreadspoke_scores, aes(x = schedule_week, fill = spread_results)) +
  geom_bar(position = "fill", width = 0.7) +
  labs(title = "Spread Results by Schedule Week",
       x = "Schedule Week",
       y = "Percentage") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_manual(values = c("Favorite" = "green", "Underdog" = "red", "Push" = "blue"))

# Your second plot
over_under_results_plot <- ggplot(spreadspoke_scores, aes(x = schedule_week, fill = over_under_results))
  geom_bar(position = "fill", width = 0.7) +
  labs(title = "Over/Under Results by Schedule Week",
       x = "Schedule Week",
       y = "Percentage") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_manual(values = c("Over" = "orange", "Under" = "yellow", "PUSH" = "brown"))

# Arrange the plots using gridExtra
grid.arrange(spread_results_plot, over_under_results_plot, ncol = 1)
```



Most weeks are nearly all 50/50 but in weeks 15, 18, and the Superbowl the underdog seems to have a slight edge Throughout the season

```
# Create the first plot
spread_results_plot <-ggplot(spreadspoke_scores, aes(x = matchup_type, fill = spread_results)) +
  geom_bar(position = "fill", width = 0.7) +
  labs(title = "Spread Results by Matchup Type",
       x = "Matchup Type",
       y = "Percentage") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_manual(values = c("Favorite" = "green", "Underdog" = "red", "Push" = "blue"))
# Create the second plot
over_under_results_plot <- ggplot(spreadspoke_scores, aes(x = matchup_type, fill = over_under_results))
  geom_bar(position = "fill", width = 0.7) +
  labs(title = "Spread Results by Schedule Week",
       x = "Schedule Week",
       y = "Percentage") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_manual(values = c("Over" = "orange", "Under" = "yellow", "PUSH" = "brown"))

# Arrange the plots using gridExtra
grid.arrange(spread_results_plot, over_under_results_plot, ncol = 1)
```



With all of the divisions the under and the underdog is slightly favored but nothing that is very significant.

Conclusion: When looking at aspects like week of the season, home vs. away, divisional games, spread category, ect. There is no edge when trying to look at these aspects because linesmakers likely input these factors into their models to make these spreads and over/unders. So trying to predict who will cover using these methods of sorting/categorizing is still a coin flip. To make better predicitons I am guessing you will have to look at more advanced offensive and defensive performance metrics to get an edge on the linesmakers.