Virat Kohli Test Career Analysis

Deepak Kumar

2025-05-13

#Load neccessary libraries

```
library(readxl)
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(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.4.3

library(lubridate)

##
## Attaching package: 'lubridate'

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

###level-1 data cleaning and preprocessing #load the dataset

```
data<-read_excel("C:/virat kohli analysis/testcareermatchwise.xlsx")

## New names:
## • `` -> `...8`
## • `` -> `...12`

View(data)
```

Rename columns for readability

```
colnames(data) <- c("Bat1", "Bat2", "Runs", "BallsFaced", "StrikeRate",
"Fours", "Sixes",
                    "Unused", "Opposition", "Ground", "StartDate",
"TestNumber")
```

Drop unused column

```
data <- data %>% select(-Unused)
```

Remove “v” prefix from Opposition

```
data$Opposition <- gsub("^v\\s+", "", data$Opposition)
```

Convert StartDate to Date format

```
library(lubridate)

data$StartDate <- dmy(data$StartDate)

## Warning: All formats failed to parse. No formats found.
```

Handle missing values and extract not out flag

```
clean_innings <- function(score) {
  score <- as.character(score)
  score[score %in% c("DNB", "TDNB", "absent")] <- NA
  score <- gsub("\\*", "", score)  # Remove not-out asterisk
  return(as.numeric(score))
}
not_out_flag <- function(score) {
  score <- as.character(score)
  return(grepl("\\*", score))
}
```

Apply cleaning to Bat1 and Bat2

```
data <- data %>%
  mutate(
    Bat1_NotOut = not_out_flag(Bat1),
    Bat2_NotOut = not_out_flag(Bat2),
    Bat1 = clean_innings(Bat1),
    Bat2 = clean_innings(Bat2)
  )

## Warning: There was 1 warning in `mutate()`.
## ℹ In argument: `Bat2 = clean_innings(Bat2)`.
## Caused by warning in `clean_innings()`:
## ! NAs introduced by coercion

# Combine total runs in both innings
data <- data %>%
  mutate(
    TotalRuns = rowSums(cbind(Bat1, Bat2), na.rm = TRUE),
```

```
    TotalNotOuts = Bat1_NotOut + Bat2_NotOut,
    PlayedInnings = (!is.na(Bat1)) + (!is.na(Bat2)),
    Year = year(StartDate)
  )
```

Clean other numeric columns

```
data <- data %>%
  mutate(
    Runs = as.numeric(Runs),
    BallsFaced = as.numeric(BallsFaced),
    StrikeRate = as.numeric(StrikeRate),
    Fours = as.numeric(Fours),
    Sixes = as.numeric(Sixes)
  )
```

```
## Warning: There were 5 warnings in `mutate()`.
## The first warning was:
## ℹ In argument: `Runs = as.numeric(Runs)`.
## Caused by warning:
## ! NAs introduced by coercion
## ℹ Run `dplyr::last_dplyr_warnings()` to see the 4 remaining warnings.
```

Add placeholder for match duration

```
data$MatchDuration <- 5  # By default, Test match = 5 days
```

Preview the cleaned data

```
str(data)
```

```
## tibble [123 × 18] (S3: tbl_df/tbl/data.frame)
##  $ Bat1         : num [1:123] 4 0 30 52 11 23 44 116 58 103 ...
##  $ Bat2         : num [1:123] 15 27 NA 63 0 9 75 22 NA 51 ...
##  $ Runs         : num [1:123] 19 27 30 115 11 32 119 138 58 154 ...
##  $ BallsFaced   : num [1:123] 64 109 53 225 22 65 217 275 107 275 ...
##  $ StrikeRate   : num [1:123] 29.7 24.8 56.6 51.1 50 ...
##  $ Fours        : num [1:123] 3 1 2 8 1 5 15 13 8 23 ...
##  $ Sixes        : num [1:123] 0 1 0 1 0 0 0 1 0 1 ...
##  $ Opposition   : chr [1:123] "v West Indies" "v West Indies" "v West
Indies" "v West Indies" ...
##  $ Ground       : chr [1:123] "Kingston" "Bridgetown" "Roseau" "Wankhede"
...
##  $ StartDate    : Date[1:123], format: NA NA ...
##  $ TestNumber   : chr [1:123] "Test # 1997" "Test # 1998" "Test # 1999"
"Test # 2019" ...
##  $ Bat1_NotOut  : logi [1:123] FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Bat2_NotOut  : logi [1:123] FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ TotalRuns    : num [1:123] 19 27 30 115 11 32 119 138 58 154 ...
```

```
##  $ TotalNotOuts : int [1:123] 0 0 0 0 0 0 0 0 0 1 ...
##  $ PlayedInnings: int [1:123] 2 2 1 2 2 2 2 2 1 2 ...
##  $ Year         : num [1:123] NA NA NA NA NA NA NA NA NA NA ...
##  $ MatchDuration: num [1:123] 5 5 5 5 5 5 5 5 5 5 ...
```

```
head(data)
```

```
## # A tibble: 6 × 18
##    Bat1  Bat2  Runs BallsFaced StrikeRate Fours Sixes Opposition    Ground
##   <dbl> <dbl> <dbl>      <dbl>      <dbl> <dbl> <dbl> <chr>         <chr>
## 1     4    15    19         64       29.7     3     0 v West Indies
Kingston
## 2     0    27    27        109       24.8     1     1 v West Indies
Bridgetown
## 3    30    NA    30         53       56.6     2     0 v West Indies Roseau
## 4    52    63   115        225       51.1     8     1 v West Indies
Wankhede
## 5    11     0    11         22       50       1     0 v Australia
Melbourne
## 6    23     9    32         65       49.2     5     0 v Australia   Sydney
## # ℹ 9 more variables: StartDate <date>, TestNumber <chr>, Bat1_NotOut
<lgl>,
## #   Bat2_NotOut <lgl>, TotalRuns <dbl>, TotalNotOuts <int>,
## #   PlayedInnings <int>, Year <dbl>, MatchDuration <dbl>
```

*after performing all cleanig check the data type of columns and retrieve first 8 data using head function

###Level -2 stastical analysis of my dataset # 1. Overall career batting average (runs per innings)

```
career_average <- sum(data$TotalRuns, na.rm = TRUE) / sum(data$PlayedInnings,
na.rm = TRUE)
```

this average may differ from real average because in real average for cricket matches consider not out inning then calculate

2. Career not-out percentage

```
not_out_percentage <- mean(data$TotalNotOuts > 0) * 100
```

3. Highest score in a single innings

```
highest_score <- max(c(data$Bat1, data$Bat2), na.rm = TRUE)
```

this score was made by kohli against south africa

4. Total career runs

```
total_runs <- sum(data$TotalRuns, na.rm = TRUE)
```

in test matches virat has scored total 9230 runs

5. Total centuries (100+ in any single innings)

```
centuries <- sum(data$Bat1 >= 100, na.rm = TRUE) + sum(data$Bat2 >= 100,
na.rm = TRUE)
```

virat has scroed total 30 test centuries

6. Total fifties (50–99 in any innings)

```
fifties <- sum(data$Bat1 >= 50 & data$Bat1 < 100, na.rm = TRUE) +
           sum(data$Bat2 >= 50 & data$Bat2 < 100, na.rm = TRUE)
```

virat has scored 31 total fifites

7. Average runs per year

```
avg_runs_year <- data %>%
  group_by(Year) %>%
  summarise(RunsInYear = sum(TotalRuns, na.rm = TRUE)) %>%
  mutate(AveragePerYear = round(RunsInYear / n(), 2))
```

8. Average against each opposition

```
avg_opposition <- data %>%
  group_by(Opposition) %>%
  summarise(Average = round(sum(TotalRuns, na.rm = TRUE) / sum(PlayedInnings,
na.rm = TRUE), 2),
            Matches = n())
```

this table shows batting average of virat kohli against each teams

9. Ground-wise batting average

```
avg_ground <- data %>%
  group_by(Ground) %>%
  summarise(Average = round(sum(TotalRuns, na.rm = TRUE) / sum(PlayedInnings,
na.rm = TRUE), 2),
            Matches = n())
```

this table shows average of different grounds

10. Strike Rate analysis: mean and max

```
mean_strike_rate <- mean(data$StrikeRate, na.rm = TRUE)
max_strike_rate <- max(data$StrikeRate, na.rm = TRUE)
print(mean_strike_rate)

## [1] 51.20223

print(max_strike_rate)

## [1] 105.55
```

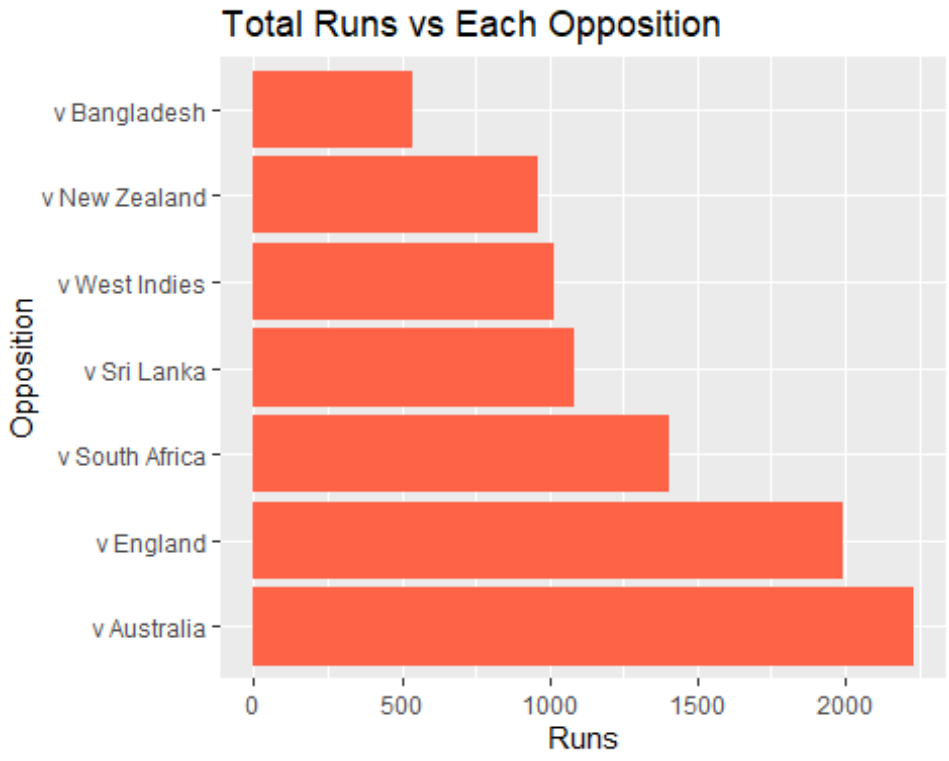
virat mean strike rate is 51.20223 virat max strike rate in test is 105.55

###Level-3 visualization of my dataset #loading required libraries visualization

```
library(ggplot2)
library(dplyr)
library(tidyr)
```

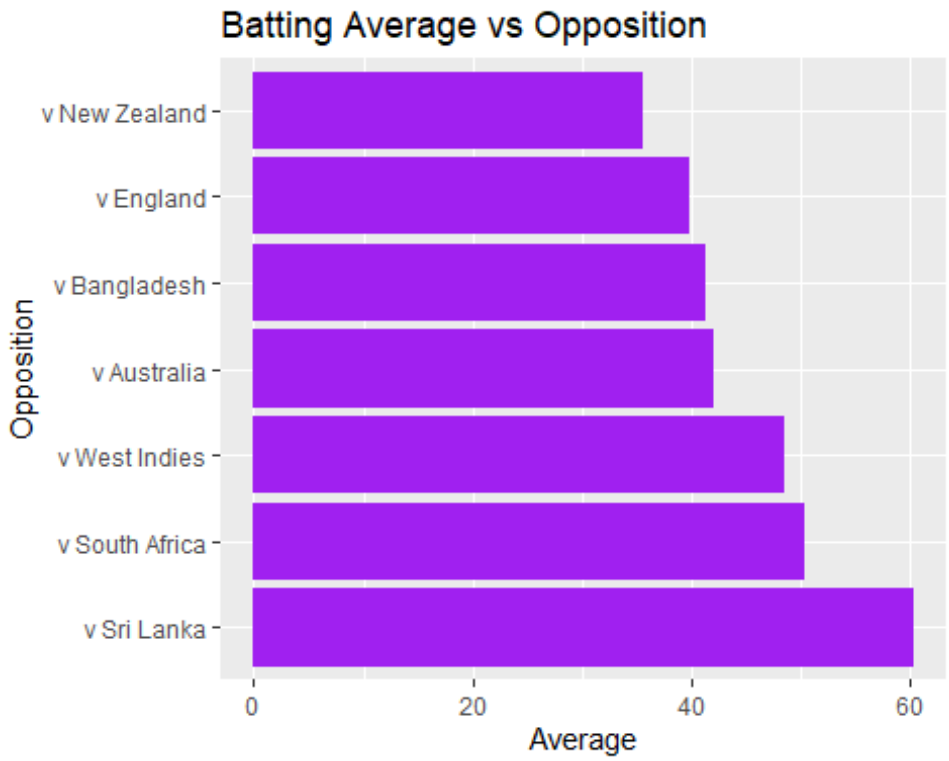
#virat kohli runs against opponents

```
ggplot(data %>% group_by(Opposition) %>% summarise(TotalRuns = sum(TotalRuns,
na.rm = TRUE)),
       aes(x = reorder(Opposition, -TotalRuns), y = TotalRuns)) +
  geom_col(fill = "tomato") +
  coord_flip() +
  labs(title = "Total Runs vs Each Opposition", x = "Opposition", y = "Runs")
```



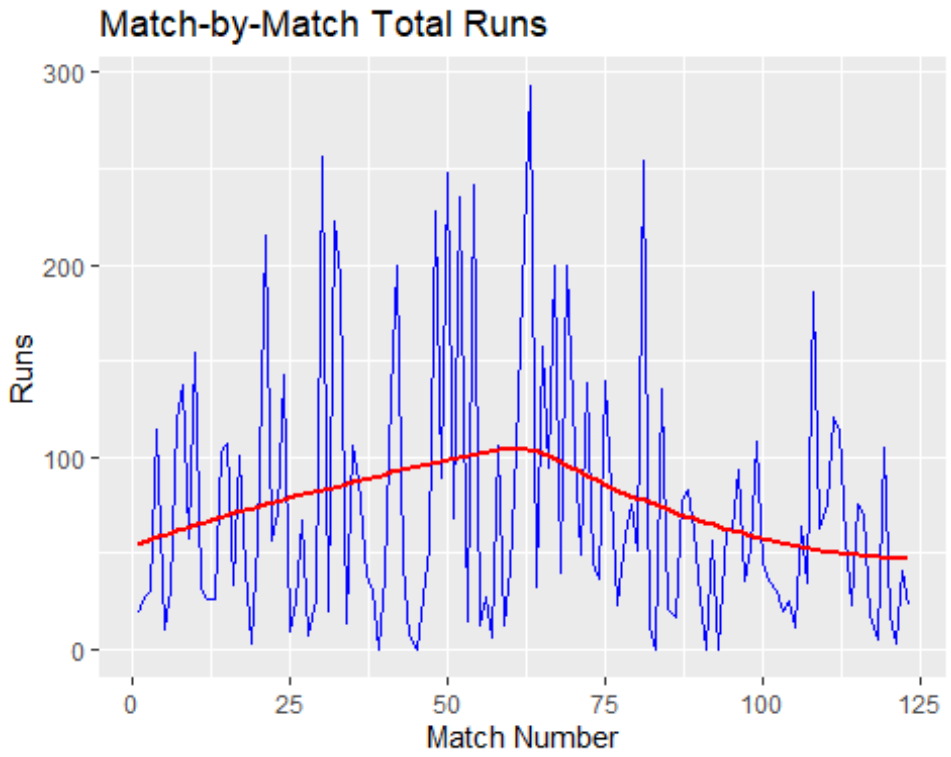
#vira kohli batting average per oppositon

```
data %>%
  group_by(Opposition) %>%
  summarise(Average = sum(TotalRuns, na.rm = TRUE) / sum(PlayedInnings, na.rm
= TRUE)) %>%
  ggplot(aes(x = reorder(Opposition, -Average), y = Average)) +
  geom_col(fill = "purple") +
  coord_flip() +
  labs(title = "Batting Average vs Opposition", x = "Opposition", y =
"Average")
```



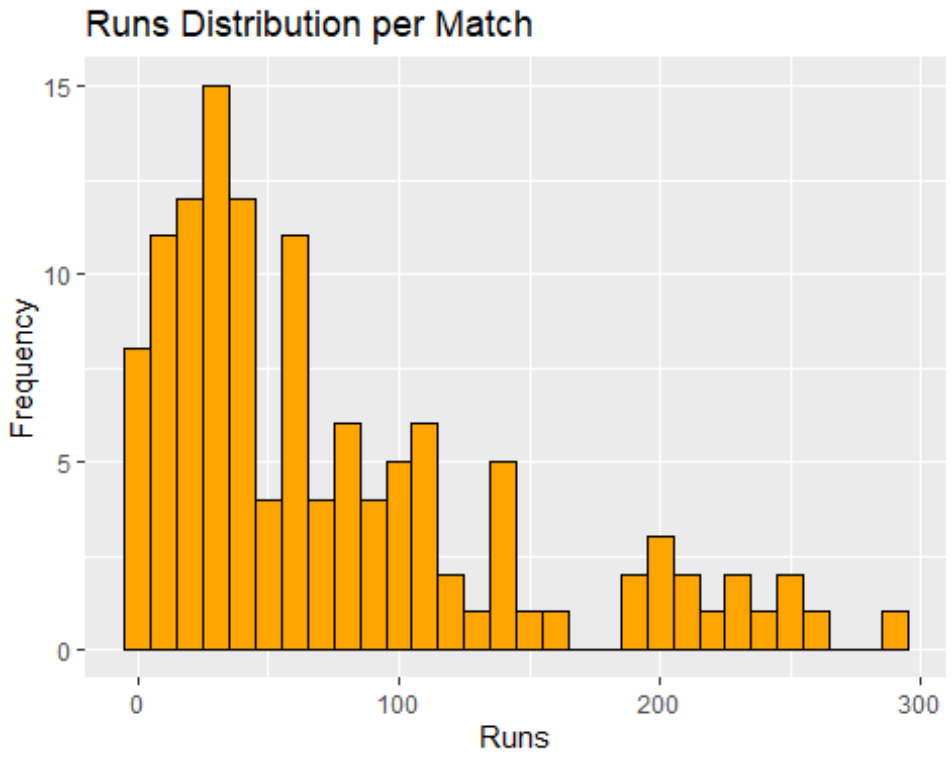
virat kohli performence over time innings-by-innings

```
data %>%
  mutate(MatchNumber = row_number()) %>%
  ggplot(aes(x = MatchNumber, y = TotalRuns)) +
  geom_line(color = "blue") +
  geom_smooth(se = FALSE, color = "red") +
  labs(title = "Match-by-Match Total Runs", x = "Match Number", y = "Runs")

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



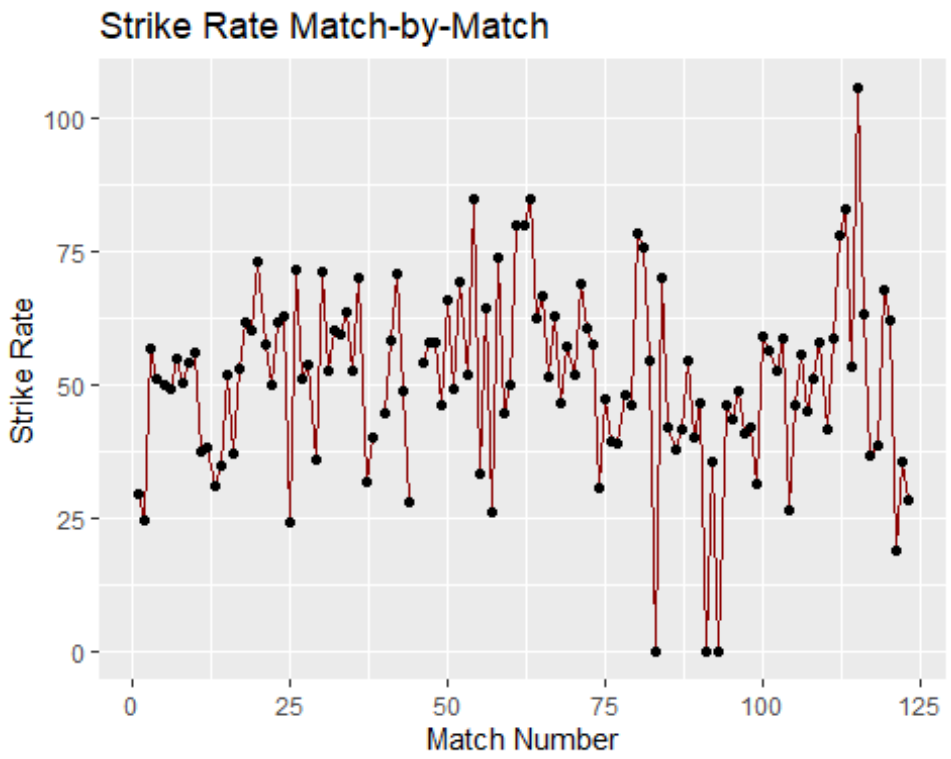
virat kohli runs distributions using histogram

```
ggplot(data, aes(x = TotalRuns)) +
  geom_histogram(binwidth = 10, fill = "orange", color = "black") +
  labs(title = "Runs Distribution per Match", x = "Runs", y = "Frequency")
```



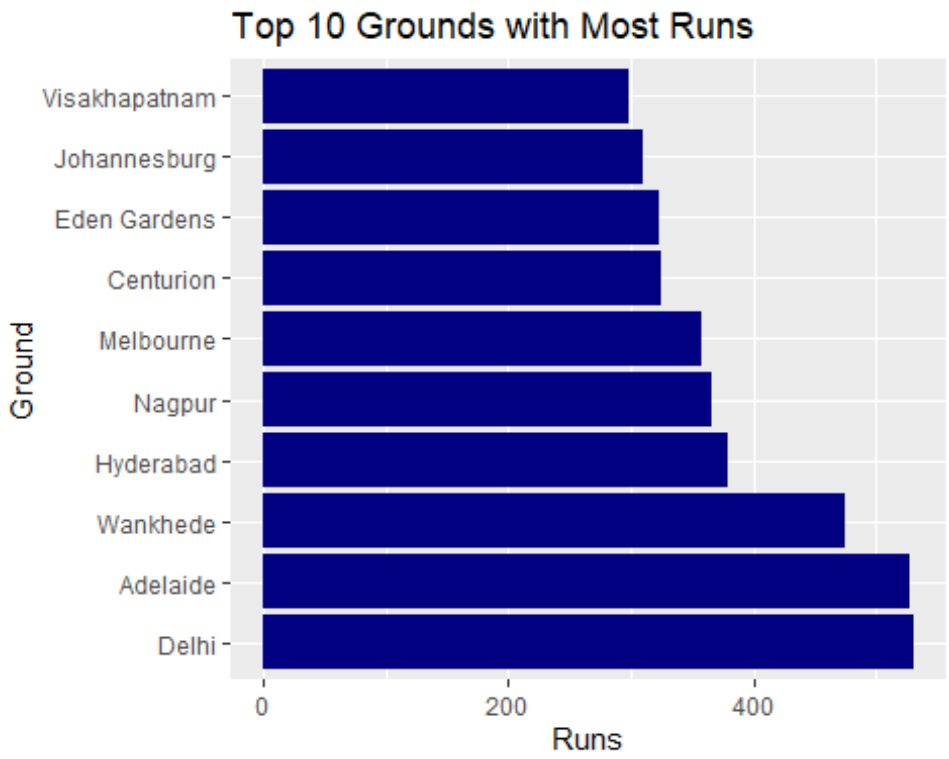
virat kohli strike rate over time

```
data %>%
  mutate(MatchNumber = row_number()) %>%
  ggplot(aes(x = MatchNumber, y = StrikeRate)) +
  geom_line(color = "darkred") +
  geom_point() +
  labs(title = "Strike Rate Match-by-Match", x = "Match Number", y = "Strike
Rate")

## Warning: Removed 2 rows containing missing values or values outside the
scale range
## (`geom_point()`).
```



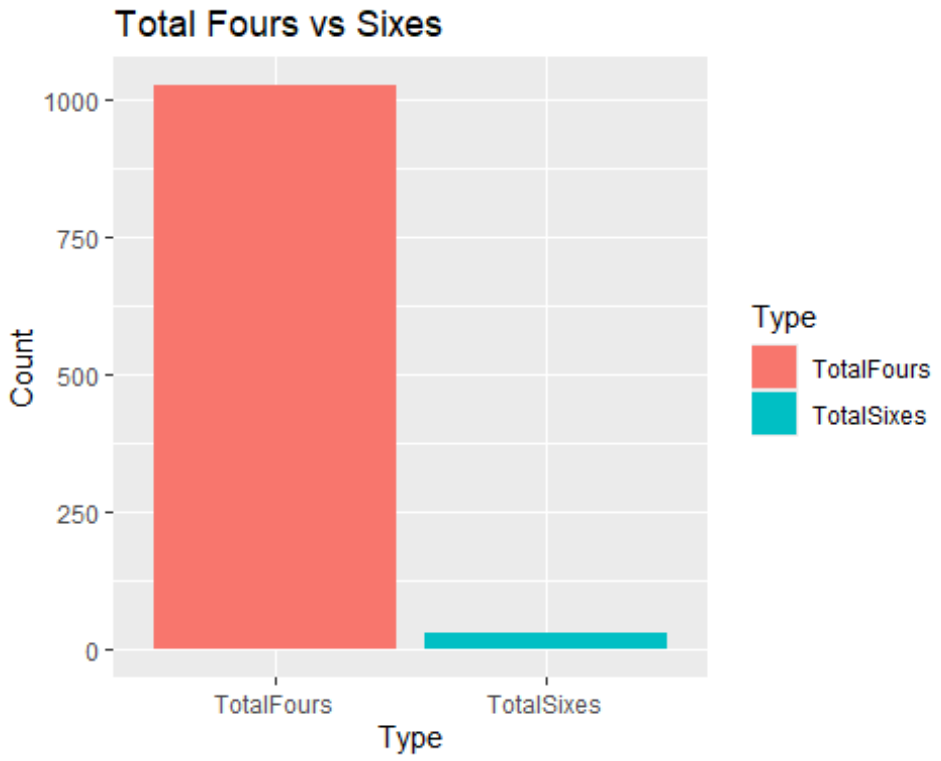
#virat kohli runs by ground

```
data %>%
  group_by(Ground) %>%
  summarise(Runs = sum(TotalRuns, na.rm = TRUE)) %>%
  arrange(desc(Runs)) %>%
  head(10) %>%
  ggplot(aes(x = reorder(Ground, -Runs), y = Runs)) +
  geom_col(fill = "navy") +
  coord_flip() +
  labs(title = "Top 10 Grounds with Most Runs", x = "Ground", y = "Runs")
```



virat kohli total fours and sixes

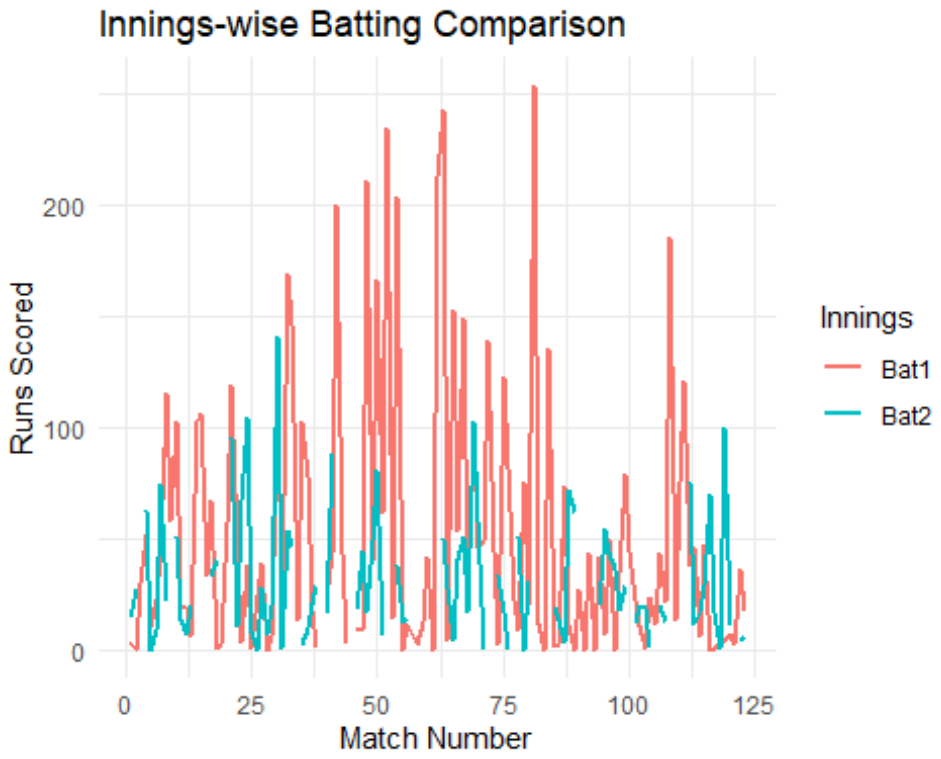
```
data %>%
  summarise(TotalFours = sum(Fours, na.rm = TRUE),
            TotalSixes = sum(Sixes, na.rm = TRUE)) %>%
  pivot_longer(cols = everything(), names_to = "Type", values_to = "Count")
%>%
  ggplot(aes(x = Type, y = Count, fill = Type)) +
  geom_col() +
  labs(title = "Total Fours vs Sixes", x = "Type", y = "Count")
```



#virat kohli bat1 vs bat2 comparison using bar plot

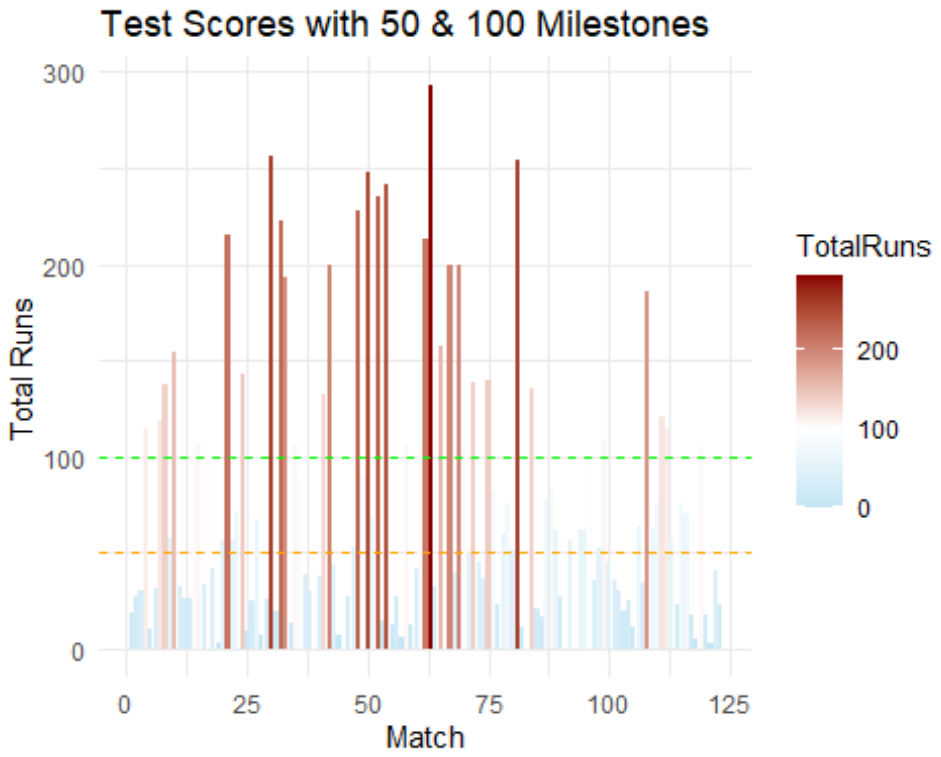
```
# Step 1: Add Match number
data <- data %>%
  mutate(Match = row_number())

# Step 2: Reshape data into long format
data_long <- data %>%
  select(Match, Bat1, Bat2) %>%
  pivot_longer(cols = c(Bat1, Bat2), names_to = "Innings", values_to =
"Runs")

# Step 3: Ensure Runs is numeric
data_long$Runs <- as.numeric(data_long$Runs)

# Step 4: Create the plot
ggplot(data_long, aes(x = Match, y = Runs, color = Innings)) +
  geom_line(size = 1) +
  labs(title = "Innings-wise Batting Comparison", x = "Match Number", y =
"Runs Scored") +
  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.
```



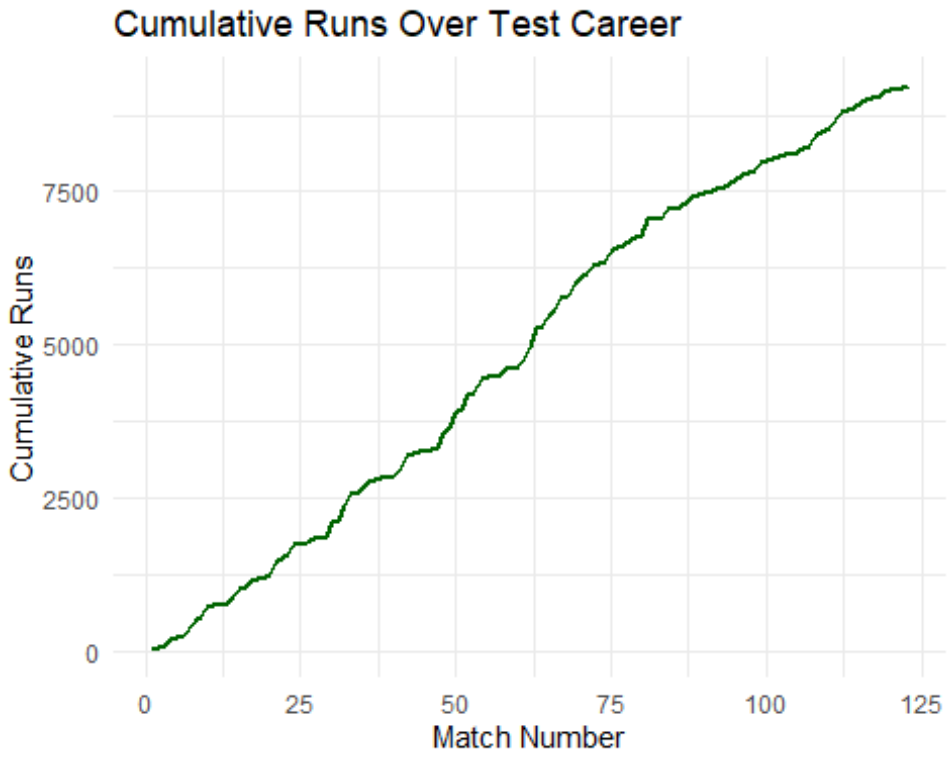
100s and 50s highlight chart

```
ggplot(data, aes(x = Match, y = TotalRuns, fill = TotalRuns)) +
  geom_col() +
  scale_fill_gradient2(low = "skyblue", high = "darkred", midpoint = 100) +
  geom_hline(yintercept = 50, linetype = "dashed", color = "orange") +
  geom_hline(yintercept = 100, linetype = "dashed", color = "green") +
  labs(title = "Test Scores with 50 & 100 Milestones", x = "Match", y =
"Total Runs") +
  theme_minimal()
```



##Runs Over Time

```
data <- data %>%
  mutate(CumulativeRuns = cumsum(TotalRuns))

ggplot(data, aes(x = Match, y = CumulativeRuns)) +
  geom_line(color = "darkgreen", size = 1) +
  labs(title = "Cumulative Runs Over Test Career", x = "Match Number", y =
"Cumulative Runs") +
  theme_minimal()
```



this graph shows total runs have accumulayed across his career