

# STAT380\_MP1

true true

February 22, 2026

## Front Matter

```
# Libraries
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)
library(readxl)
library(lubridate)

##
## Attaching package: 'lubridate'

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

library(knitr)

# Functions
countNA <- function(col){
  return(sum(is.na(col)))
}
```

## Reading in Data Set

```
games_url <- "https://raw.githubusercontent.com/Swopei/STAT380-MP1_repo/main/data/CODGames2_mp.xlsx"
games_tmp <- tempfile(fileext = ".xlsx")
download.file(games_url, destfile = games_tmp, mode = "wb")
Games <- read_excel(games_tmp)
```

This solution was found using Google Gemini, the prompt used was, “How can a .xlsx file be read into R from an url?”

Gemini generation the following as one possible solution:

```
library(readxl)

# 1. Define the URL url <- "https://example.com/data.xlsx"
# 2. Create a temporary file path dest_file <- tempfile(fileext = ".xlsx")
# 3. Download the file (CRITICAL: use mode = "wb") download.file(url, destfile = dest_file, mode = "wb")
# 4. Read the file df <- read_excel(dest_file)
# Optional: clean up the temp file unlink(dest_file)
```

## Data Exploration

This data set contains 27 columns and 211 rows.

The variable FullPartial is a categorical variable

```
levels(as.factor(Games$FullPartial)) #gets categories
```

```
## [1] "Full"      "Partial"
```

FullPartial has two categories: “Full” and “Partial”

```
countNA(as.factor(Games$FullPartial)) #gets NA count
```

```
## [1] 0
```

FullPartial has no NAs

```
Games_FP <- Games %>%
  group_by(FullPartial) %>%
  summarize(
    count = sum(FullPartial == FullPartial),
    percent = paste0(round(sum(FullPartial == FullPartial)/nrow(Games)*100, 2), "%")
  )

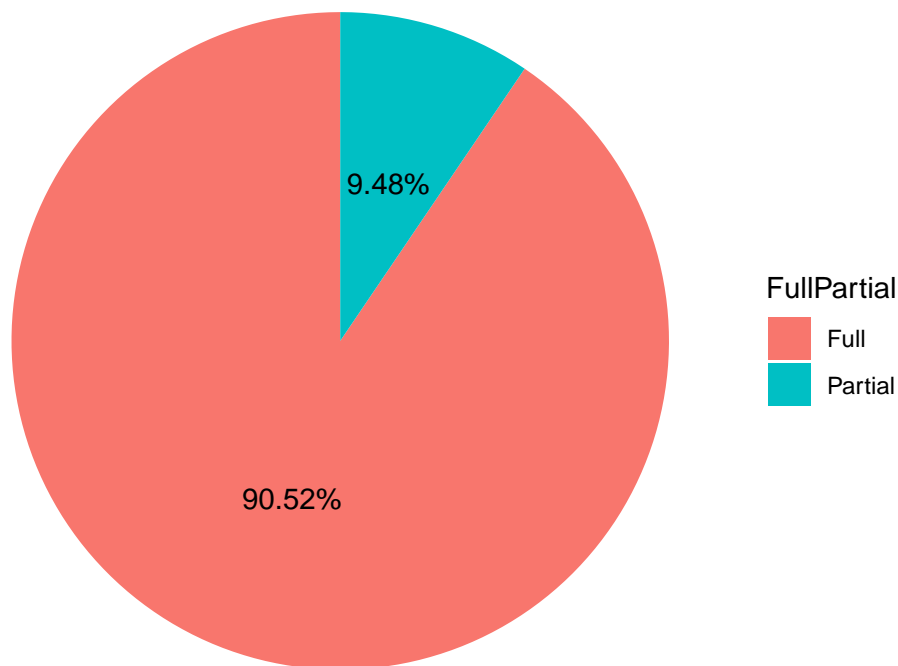
kable(Games_FP)
```

FullPartial	count	percent
Full	191	90.52%

FullPartial	count	percent
Partial	20	9.48%

```
ggplot(data = Games_FP,
       mapping = aes(x = "", y = count, fill = FullPartial))
) +
geom_bar(stat = "identity", width = 1) +
coord_polar("y", start = 0) +
theme_void() +
geom_text(aes(label = percent), position = position_stack(vjust = 0.5)) +
labs(subtitle = "n = 211")
```

n = 211



From the above table and pie chart, we can see that Full is much more likely to appear than partial.

## Proposed Questions

**Question 1: Does Player Performance Change Over Time?**

**Question 2: Our Question**

##Code Appendix

```

knitr::opts_chunk$set(echo = TRUE)
# Libraries
library(dplyr)
library(ggplot2)
library(readxl)
library(lubridate)
library(knitr)

# Functions
countNA <- function(col){
  return(sum(is.na(col)))
}

games_url <- "https://raw.githubusercontent.com/Swopei/STAT380-MP1_repo/main/data/CODGames2_mp.xlsx"
games_tmp <- tempfile(fileext = ".xlsx")
download.file(games_url, destfile = games_tmp, mode = "wb")
Games <- read_excel(games_tmp)
levels(as.factor(Games$FullPartial)) #gets categories
countNA(as.factor(Games$FullPartial)) #gets NA count
Games_FP <- Games %>%
  group_by(FullPartial) %>%
  summarize(
    count = sum(FullPartial == FullPartial),
    percent = paste0(round(sum(FullPartial == FullPartial)/nrow(Games)*100, 2), "%")
  )

kable(Games_FP)

ggplot(data = Games_FP,
       mapping = aes(x = "", y = count, fill = FullPartial))
  ) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y", start = 0) +
  theme_void() +
  geom_text(aes(label = percent), position = position_stack(vjust = 0.5)) +
  labs(subtitle = "n = 211")

```