Analysis of the Mean Global Game Sales

Using Simple Random Sampling and Stratified Random Sampling

Group Members And Role:

Haneul Kim, 58285446, Introduction, Objectives & Conclusion

Youjung Kim 38762639, Data analysis

Seojun Hong, 54678321 Team Lead & Discussion

Minkyung Yun 65258436, Data analysis & Paper Review

PART I

Introduction

Since the popularization of desktop computers and portable gaming devices, video gaming has become a phenomenon. The general public has started to spend more time playing games in their free time, with their friends, and families. From a minor subculture that not many people enjoy, gaming has transformed into one of the mainstream cultures with huge gaming leagues and players. As the population who plays games on a regular basis has increased, the game market has also grown its size exponentially. Thus, game production in this area is highly profitable. There are many platforms where one game can be purchased and played.

This, however, only applies to games that provide what people want. Games that fail to attract people often introduce huge losses to a game production company. There are many factors that could be taken into account when trying to estimate its popularity. Most importantly, genre is one single important factor that decides how many people will buy, and play a game. For example, action and sports games are played by a large population whereas puzzle-solving games are less popular among the general public.

Objectives

In this group project, the group tries to accurately estimate the mean total global sales of video games from a sample size of 1000, from a population of size 16598. For sampling methods, both SRS and stratified random sampling are used. The reason behind using stratified random samples is due to the fact that the number of games sold is different from genre to genre. Since large between-strata variance is expected, the use of stratified random sampling is justified.

By trying to accurately estimate the population mean of interest, this group wishes to answer how much of a population ends up purchasing a video game on average. Estimating the mean total global sales of video games, helps us get a glimpse of how many games that game production companies can expect to sell when they release their games.

Data analysis

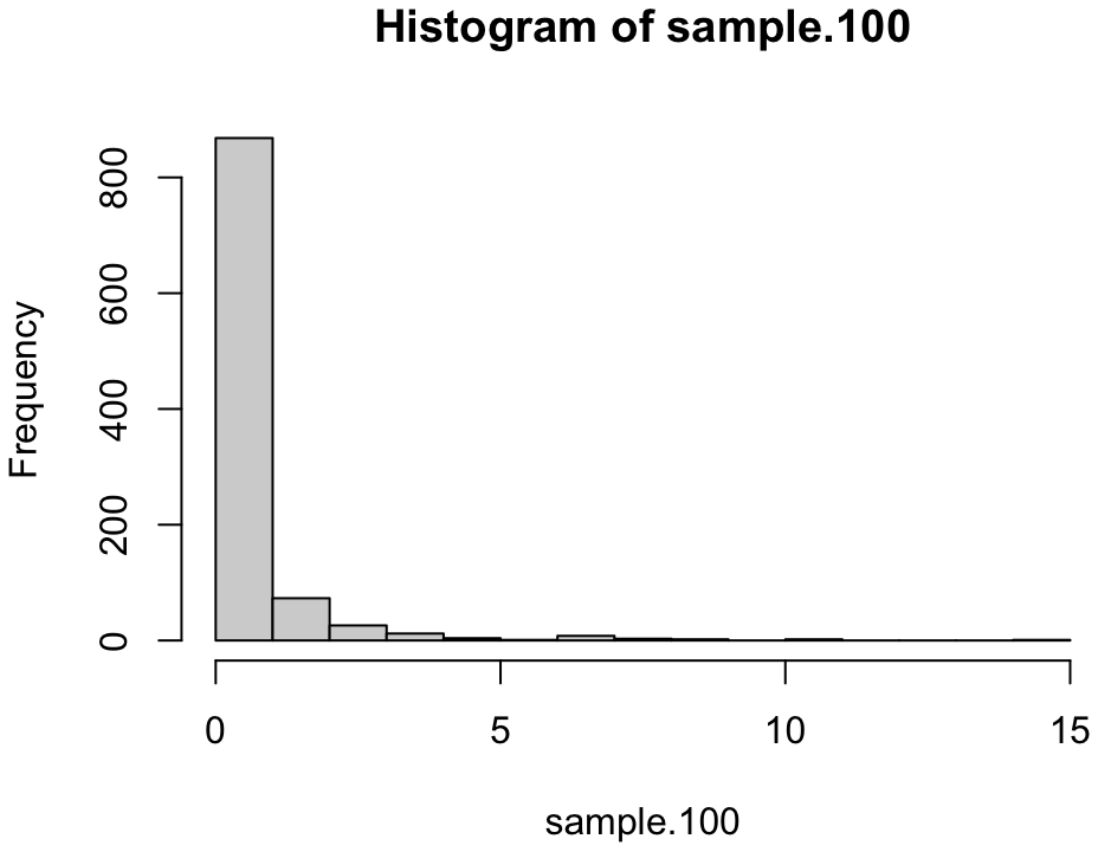
In this study, we conducted both simple random sampling and stratified sampling from a population of 11,493 video games with sales exceeding 100,000 copies. For simple random sampling, we assume that each observation in the sample is independent of others and ensure that each element has an equal chance of being selected. In stratified sampling, our general assumptions were that sampling within each stratum is random and the observations within each stratum are independent from each other. Since we are using proportional allocation, we will be calculating the proportion of each strata in the population and assume that it was known from previous studies.

In the SRS binary method, we are interested in the proportion of video games with global sales greater than or equal to 1 million dollars. We randomly selected 1000 samples from the population, resulting in an estimated proportion of video game sales greater than or equal to 1 million dollars of approximately 13.400%. The standard error calculated from the sample is approximately 0.01077. The formulas used are presented below:

$$\hat{p}_{srs} = \frac{number\ of\ samples\ with\ one\ million\ sales\ or\ more}{1000\ samples} = 0.1340,$$

$$SE[\hat{p}_{srs}] = \sqrt{(1 - \frac{n}{N})\frac{S^2}{n}} = \text{where } S^2 \text{ is } \hat{p}_{srs}(1\text{-}\ \hat{p}_{srs})\ .$$

If the study is conducted multiple times with corresponding 95% confidence intervals, the true proportion of video games with global sales greater than or equal to 1 million dollars in the entire population falls within the interval of [0.1129, 0.1551] percent. The histogram below illustrates the distribution of the sample.



When estimating the population mean of global sales using SRS, 1000 samples from the populations are selected with equal chances. The estimated sample mean is 0.5334 (in millions) with a standard error of 0.03663 (in millions). The formulas used are presented below.

$$\hat{y} = \sum_{i=1}^{1000} y_i \, / \, 1000 = 0.5334$$

$SE[\hat{y}] = \sqrt{(1 - \frac{n}{N})\frac{s^2}{n}} = 0.010361$ where s^2 is the sample variance.

The 95% confidence interval is given by [0.4616, 0.6052] (in millions). The confidence interval implies that 19 out of 20 times, the true mean lies in the confidence interval calculated using the sample mean and standard error estimated each time.

In stratified binary sampling, we stratified the video game sales data by genre. Each stratum consists of individual genres such as Action, Shooter, Sports, Role-playing, etc. Samples are randomly selected from each stratum, with sample sizes determined using population allocation.

We are interested in the stratified proportion of video game sales over 1 million sales.

$$\hat{p}_{str} = \sum_{h=1}^{H} (\frac{N_h}{N})\hat{p}_{S_h} = 0.136064$$

$$SE[\hat{p}_{str}] = \sqrt{\sum_{h=1}^{H} (\frac{N_h}{N})^2 (1 - \frac{n_h}{N_h}) \frac{S^2_{Sh}}{n_h}} = 0.010361 \text{ where } S^2_{Sh} = \hat{p}_{Sh}(1\text{-}\hat{p}_{Sh}) \text{ for each h, 1 to 12}$$

The estimated stratified proportion based on the stratified sampling is approximately 13.6064%, with a standard error of 0.010361. Using these statistics, a 95% confidence interval can be calculated using the following formula.

$\hat{p}_{str} \pm 1.96 * SE[\hat{p}_{str}]$.

According to the calculated confidence interval, we are 95% confident that the true proportion of games with 1 million or more global sales in the population is between 11.5756% and 15.6371%.

In addition to simple random sampling (SRS), we conducted stratified continuous sampling as an alternative method. In stratified sampling, the population is divided into subpopulations based on genres of video games. The samples are randomly selected from each stratum, specifically the video games genre and the sample sizes for each genre are determined by population allocation. Assuming equal variance and sample costs across all strata, the population allocation might suggest optimal sample sizes for each genre.

Based on stratified sampling with an identical sample to SRS, our estimate for the average global video game sales and its standard error are given by:

$$\bar{y}_{str} = \sum_{h=1}^{H} (\frac{N_h}{N})\bar{y}_{S_h} = 0.5170 \text{ (in millions),}$$

$$SE[\bar{y}_{str}] = \sqrt{\sum_{h=1}^{H} (\frac{N_h}{N})^2 (1 - \frac{n_h}{N_h}) \frac{S^2_{Sh}}{n_h}} = 0.0318 \text{ (in millions).}$$

The standard error is slightly reduced, being 0.0048, about 13% smaller than that of SRS, which reveals that the stratified sampling performs relatively better than SRS. Due to the smaller standard error, a 95% confidence interval for the average global sales might become narrower compared to that of SRS, indicating greater precision. Using the estimates calculated above, a 95% confidence interval can be generated by the formula:

$$\overline{y}_{str} \pm 1.96 * SE[\overline{y}_{str}] = (0.4546,\ 0.5793).$$

According to the obtained confidence interval, we are 95% confident that the true population mean for global video game sales lies between 0.4546 and 0.5793 (in millions).

Discussion

While being simple to interpret and understand, in general, simple random sampling (SRS) exhibits two primary shortcomings. Firstly, ensuring an equal chance of selection for every individual in the population proves challenging in practice. This necessitates a comprehensive population list, which becomes increasingly difficult to access as the population size grows. Consequently, more time and research costs may be incurred, rendering this approach unviable in some instances. However, if these conditions are met, the resulting sample will be unbiased. Luckily, for this project, this issue did not arise as the selected dataset was treated as the entire population which means full access to the list for fair sampling is ensured.

The second flaw of SRS lies in its potential lack of representativeness for the entire population. For instance, in this project's dataset, action games were nearly six times more prevalent than puzzle games. SRS would result in a significantly lower likelihood of sampling puzzle games, rendering the sample unrepresentative of the overall population.

To address this concern, the project employed stratified sampling as a second method. Stratified sampling ensures the representation of small groups or strata. Proportional allocation is used to establish sample sizes for each stratum, preventing over-sampling or under-sampling. Compared to SRS, stratified sampling yields a smaller standard error when strata exhibit sufficient differences. The project justified using stratified sampling by assuming that games of different genres have varying global sales. The strata were delineated based on the genre of each game. While stratified sampling has its limitations, such as the potential difficulty and time consumption associated with classifying and dividing observations into different strata, the chosen dataset mitigated this concern by already classifying each game into one of twelve genres. Another limitation of this method derives from the fact that proportional allocation was used for sample sizes of the strata. For this method to be usable, the proportion of each strata in the population has to be known or at least a guess from previous studies. Again, this project treated a dataset as a population and therefore assumed that the proportion for each strata in the population is known. In a real-life scenario, using proportional allocation for 12 stratas like this project may be very difficult.

A limitation specific to this project involved the potential ambiguity in genre classification. While the dataset categorized "Tetris" as a puzzle game, others might argue it qualifies as a strategy game. The inherent challenge arises from most games not adhering to a single genre but rather embodying a blend of multiple genres. A change in the classifier could yield different results, adding a layer of subjectivity to the findings.

Conclusion

Overall, using stratified sampling decreased the standard error noticeably in both binary and continuous mean estimation cases compared to SRS. This shows that our expectation about the population was valid: that between strata variance is large. The estimation of population means is 0.5170 million with a standard error of 0.0318 million for the continuous case. The estimation of the proportion of games that are sold more than 1 million times is 13.6064% and 0.010361%. The confidence intervals for both cases are [0.4546, 0.5793] in millions, and [11.5756, 15.6371] in percentage, respectively. Although the data and analysis used in this project conclude that the use of stratified random sampling is better in terms of resulting confidence intervals that are narrower, this conclusion cannot be generalized. When a larger population or other population is of interest, accurate prior analysis of the population about whether between strata variance is large or not, would be required.

PART II

The likelihood ratio tests (LRT) regarded as inferior have emerged in the realm of statistical literature, over the last two decades. Especially in multiparameter hypothesis testing problems, there is an assertion that the likelihood ratio criterion yields faulty statistical procedures, thus suggesting alternative tests. Even if the size alpha tests are considered superior due to increased power and reduced biases, the authors claimed that the new tests are defective with implausible and statistically unacceptable inferential results. Some criticize the LRT while endorsing the new tests, emphasizing the significance of power, unbiasedness, and size over intuition. However, there are some concerns that neglecting intuition could ruin the reliability of statistical science. In this regard, the authors reappraised the LRT as a primary method for non-Bayesian parametric hypothesis-testing problems.

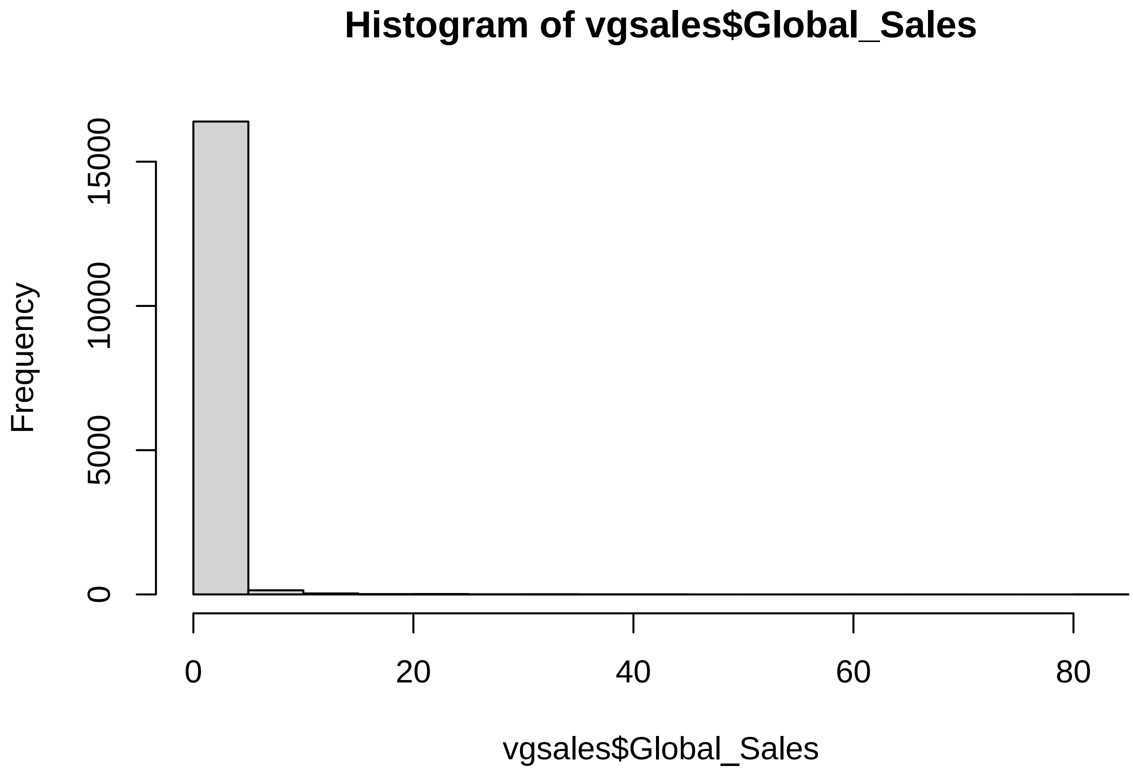
Binary with SRS and Stratified Sampling

Haneul Kim, Youjung Kim, Seojun Hong, Minkyung Yun

Nov 15th, 2023

```r
setwd("~/Desktop/STAT 344/project")
set.seed(1)
vgsales <- read.csv("vgsales.csv",sep = ",", header = TRUE)

# Histogram of the a population
hist(vgsales$Global_Sales)
```

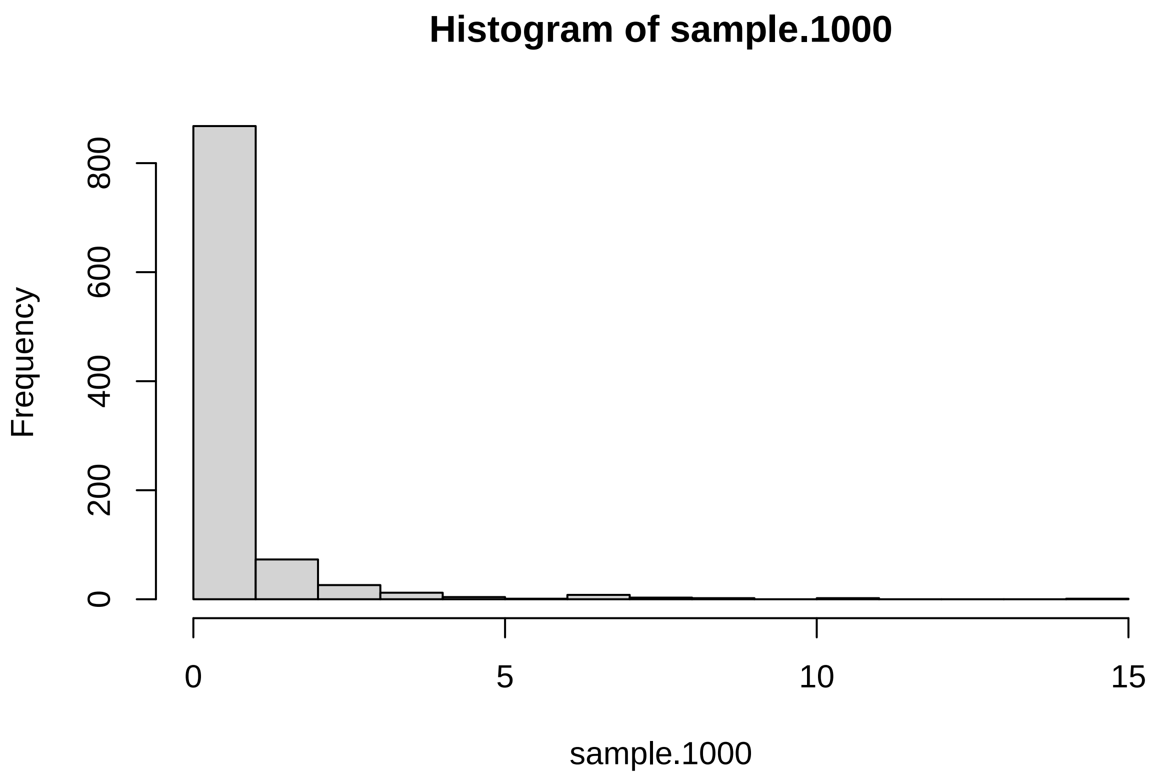


```r
N <- nrow(vgsales)
n <- 1000

# A simple randome samples of 1000

sample.1000 <- sample(vgsales$Global_Sales, n, replace=F)
```

```r
# Histogram of the a sample
hist(sample.1000)
```



```r
# sample parameter values
s.G1 <- sum(sample.1000 >= 1)/n
s.L1 <- sum(sample.1000 < 1)/n
print(c(s.G1,s.L1))
```

```
## [1] 0.134 0.866
```

```r
# Compute sample proportion for sample
phat <- s.G1
phat # 0.13400000
```

```
## [1] 0.134
```

```r
# Variance of the a sample
SE <- sqrt(phat*(1-phat)/n)
SE # 0.01077237
```

```
## [1] 0.01077237
```

```
# Confidence interval
CI.lower <- phat - 1.96*SE
CI.upper <- phat + 1.96*SE
print(c(CI.lower, CI.upper))
```

```
## [1] 0.1128862 0.1551138
```

```
# Stratified binary
set.seed(1)
vgsales$Genre <- as.factor(vgsales$Genre)
head(vgsales)
```

```
##   Rank                     Name Platform Year        Genre Publisher NA_Sales
## 1    1               Wii Sports      Wii 2006       Sports  Nintendo    41.49
## 2    2        Super Mario Bros.      NES 1985     Platform  Nintendo    29.08
## 3    3           Mario Kart Wii      Wii 2008       Racing  Nintendo    15.85
## 4    4        Wii Sports Resort      Wii 2009       Sports  Nintendo    15.75
## 5    5 Pokemon Red/Pokemon Blue       GB 1996 Role-Playing  Nintendo    11.27
## 6    6                   Tetris       GB 1989       Puzzle  Nintendo    23.20
##   EU_Sales JP_Sales Other_Sales Global_Sales
## 1    29.02     3.77        8.46        82.74
## 2     3.58     6.81        0.77        40.24
## 3    12.88     3.79        3.31        35.82
## 4    11.01     3.28        2.96        33.00
## 5     8.89    10.22        1.00        31.37
## 6     2.26     4.22        0.58        30.26
```

```
max(vgsales$Global_Sales) # 82.74
```

```
## [1] 82.74
```

```
sample <- NULL
for (i in levels(vgsales$Genre)) {
  row_i <- which(vgsales$Genre == i)
  if (i == "Adventure") {
    n_h <- 78
  } else {n_h <- round((length(row_i)/nrow(vgsales) * n))}

  row_i_sample <- sample(row_i, n_h, replace=F)
  sample <- rbind(sample, vgsales[row_i_sample,])
}

Best_Seller <- (sample$Global_Sales >= 1)
Best_Seller_x <- (sample$NA_Sales >= 1)
mod_sample <- cbind(sample,Best_Seller,Best_Seller_x)

y_str <- 0

for (i in levels(vgsales$Genre)) {
  N_h <- length(which(vgsales$Genre == i))
  n_h <- length(which(mod_sample$Genre == i))
  p <- sum(subset(mod_sample, Genre == i)$Best_Seller)/n_h
  y_s <- N_h/N * p
  y_str <- y_str + y_s
}

y_str_se_sq <- 0

for (i in levels(vgsales$Genre)) {
  N_h <- length(which(vgsales$Genre == i))
  n_h <- length(which(mod_sample$Genre == i))
  p <- sum(subset(mod_sample, Genre == i)$Best_Seller)/n_h
  s <- (N_h/N)^2 * (1 - n_h/N_h) * (p * (1-p))/n_h
  y_str_se_sq <- y_str_se_sq + s
}

y_str # 0.1360635
```

```
## [1] 0.1360635
```

```
sqrt(y_str_se_sq) # 0.01036072
```

```
## [1] 0.01036072
```

```
CI.lower <- y_str - 1.96*sqrt(y_str_se_sq)
CI.upper <- y_str + 1.96*sqrt(y_str_se_sq)
print(c(CI.lower, CI.upper)) # 0.1157564 0.1563705
```

```
## [1] 0.1157564 0.1563705
```

Continuous with SRS and Stratified Sampling

Haneul Kim, Youjung Kim, Seojun Hong, Minkyung Yun

Nov 15th, 2023

```r
setwd("~/Desktop/STAT 344/project")
set.seed(1)
library(ggplot2)

# Read the data
vgsales <- read.csv("vgsales.csv",sep = ",", header = TRUE)

# SRS MEAN
SRS_sample = sample(vgsales$Global_Sales, 1000, replace = FALSE)
mean(SRS_sample) # 0.53343
```

```
## [1] 0.53343
```

```r
# SRS SE
se = sqrt((1-1000/nrow(vgsales))*var(SRS_sample)/1000)
se # 0.03662889
```

```
## [1] 0.03662889
```

```r
#Confidence Interval

ub = mean(SRS_sample) + 1.96 * se
lb = mean(SRS_sample) - 1.96 * se

# Stratified continous

dat.fil <- vgsales[c("Genre", "Global_Sales")]
N <- nrow(dat.fil)
n <- 1000

N.h <- table(dat.fil$Genre)

# population proportion for each strata
N.h/N
```

```
##
##       Action    Adventure     Fighting         Misc     Platform       Puzzle
##   0.19978311   0.07747921   0.05109049   0.10477166   0.05337993   0.03506447
##       Racing Role-Playing      Shooter   Simulation       Sports     Strategy
##   0.07525003   0.08964936   0.07892517   0.05223521   0.14134233   0.04102904
```

```
# Determine sample sizes for each strata using population allocation
(N.h/N)*n
```

```
##
##       Action    Adventure     Fighting         Misc     Platform       Puzzle
##    199.78311     77.47921     51.09049    104.77166     53.37993     35.06447
##       Racing Role-Playing      Shooter   Simulation       Sports     Strategy
##     75.25003     89.64936     78.92517     52.23521    141.34233     41.02904
```

```
set.seed(1)

## Extract random samples from each strata
genres <- c("Action", "Adventure", "Fighting", "Misc", "Platform",
            "Puzzle", "Racing", "Role-Playing", "Shooter",
            "Simulation", "Sports", "Strategy")
samples <- list()

# sample size = 1000 using population allocation
sample_sizes <- c(200, 78, 51, 105, 53, 35, 75, 90, 79, 52, 141, 41)

for (i in seq_along(genres)) {
  genre <- genres[i]
  sample_size <- sample_sizes[i]
  genre_sample <- dat.fil[sample(which(dat.fil$Genre == genre), sample_size, replace = FALSE), ]
  samples[[genre]] <- genre_sample
}

# Sample means
str_means <- numeric(length(genres))
variable <- "Global_Sales"

for (i in seq_along(genres)) {
  genre <- genres[i]
  genre_sample <- samples[[genre]]
  str_means[i] <- mean(genre_sample[[variable]], na.rm = TRUE)
}

str_means
```

```
##  [1] 0.5536000 0.1700000 0.3225490 0.5199048 0.5026415 0.2774286 0.5849333
##  [8] 0.6146667 1.1177215 0.4367308 0.4703546 0.2204878
```

```
## sample strata variance
str_variances <- numeric(length(genres))
variable <- "Global_Sales"

for (i in seq_along(genres)) {
  genre <- genres[i]
  genre_sample <- samples[[genre]]
  str_variances[i] <- var(genre_sample[[variable]], na.rm = TRUE)
}

str_variances
```

```
##  [1] 0.9518784 0.1001740 0.1336354 1.2482240 0.5055852 0.1204785 0.6822524
##  [8] 2.0952701 4.9505588 0.7286656 0.2711320 0.1208548
```

```
wt.strata = N.h/N

## overall stratified estimate and its SE
Str.est = sum(wt.strata*str_means)
Str.se = sqrt(sum((wt.strata^2)*(1-(sample_sizes/N.h))*(str_variances/sample_sizes)))

print(c(Str.est, Str.se))
```

```
## [1] 0.51695794 0.03181284
```

```
# [1] 0.51695794 0.03181284


# Confidence Interval
lower.b = Str.est - 1.96*Str.se
upper.b = Str.est + 1.96*Str.se

print(c(lower.b, upper.b))
```

```
## [1] 0.4546048 0.5793111
```