Final Project

Final Project

STAT506 Final Project.

Data Import

```
agi_expanded <- read.csv("/Users/garrettpinkston/Desktop/Michigan/STAT506/Final/Data/20zpalla
hc_expanded <- read.csv("/Users/garrettpinkston/Desktop/Michigan/STAT506/Final/Data/Medicare_
```

```
agi = subset(agi_expanded, select = c("zipcode", "agi_stub", "A00100","N1"))

hc = subset(hc_expanded, select = c("Rndrng_Prvdr_Zip5", "Tot_Benes", "Tot_Srvcs", "Tot_Mdcr_
```

```
# agi is listed in thousands
# to find average agi, we need to divide agi by nreturns

colnames(hc)[colnames(hc) == 'Rndrng_Prvdr_Zip5'] <- 'zipcode'

colnames(agi)[colnames(agi) == 'agi_stub'] <- 'agi_bracket'
colnames(agi)[colnames(agi) == 'A00100'] <- 'agi'
colnames(agi)[colnames(agi) == 'N1'] <- 'nreturns'
```

```
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
```

```r
# find average agi and total agi by zipcode
agi <- agi %>%
  group_by(zipcode) %>%
  summarize(
    avg_agi = sum(agi, na.rm = TRUE) / sum(nreturns, na.rm = TRUE),
    total_agi = sum(agi, na.rm = TRUE)
  )

# find total beneficiaries, services and payment by zipcode
hc <- hc %>%
  group_by(zipcode) %>%
  summarize(
    total_beneficiaries = sum(Tot_Benes, na.rm = TRUE),
    total_services = sum(Tot_Srvcs, na.rm = TRUE),
    total_payments = sum(Tot_Mdcr_Pymt_Amt, na.rm = TRUE)
  )


df <- merge(agi, hc, by = "zipcode")

df <- df %>%
  filter(grepl("^\\d{5}$", zipcode) & zipcode != "99999")

df <- na.omit(df)
```

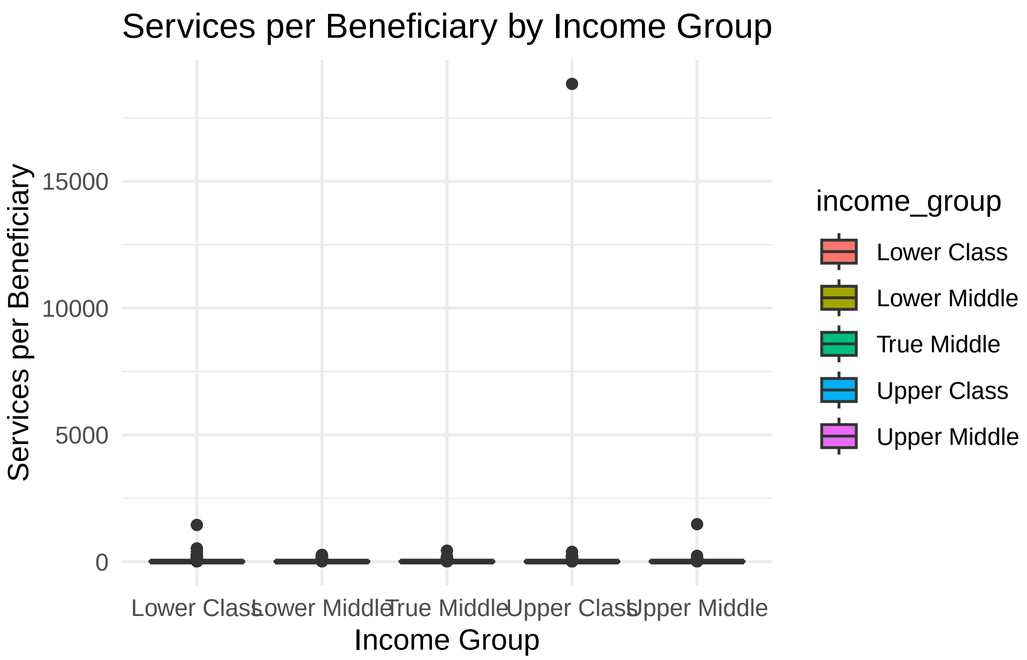
```r
income_quantiles <- quantile(df$avg_agi, probs = seq(0.2, 0.8, by = 0.2), na.rm = TRUE)

df <- df %>%
  mutate(income_group = case_when(
    avg_agi <= income_quantiles[1] ~ "Lower Class",
    avg_agi > income_quantiles[1] & avg_agi <= income_quantiles[2] ~ "Lower Middle",
    avg_agi > income_quantiles[2] & avg_agi <= income_quantiles[3] ~ "True Middle",
    avg_agi > income_quantiles[3] & avg_agi <= income_quantiles[4] ~ "Upper Middle",
    avg_agi > income_quantiles[4] ~ "Upper Class",
    TRUE ~ NA_character_
  ))

income_quantiles
```

```
     20%      40%      60%      80% 
47.51139 55.61107 65.00789 84.57395 
```

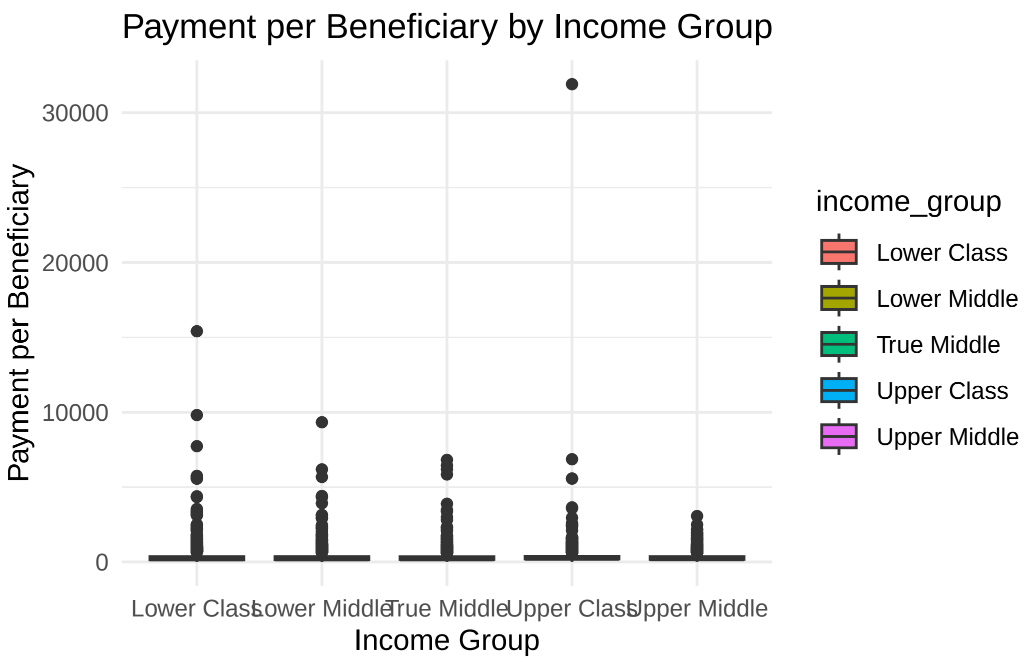
```r
df <- df %>%
  mutate(
    services_per_beneficiary = total_services / total_beneficiaries,
    payment_per_beneficiary = total_payments / total_beneficiaries
  )

summary_metrics <- df %>%
  group_by(income_group) %>%
  summarize(
    mean_services_per_beneficiary = mean(services_per_beneficiary, na.rm = TRUE),
    mean_payment_per_beneficiary = mean(payment_per_beneficiary, na.rm = TRUE)
  )

summary_metrics
```

```
# A tibble: 5 x 3
  income_group mean_services_per_beneficiary mean_payment_per_beneficiary
  <chr>                                <dbl>                        <dbl>
1 Lower Class                           10.9                         316.
2 Lower Middle                          11.2                         310.
3 True Middle                           10.5                         304.
4 Upper Class                           14.8                         334.
5 Upper Middle                          10.5                         298.
```

```r
library(ggplot2)

# visualization across income groups
# boxplot services per beneficary
ggplot(df, aes(x = income_group, y = services_per_beneficiary, fill = income_group)) +
  geom_boxplot() +
  theme_minimal() +
  labs(
    title = "Services per Beneficiary by Income Group",
    x = "Income Group",
    y = "Services per Beneficiary"
  )
```



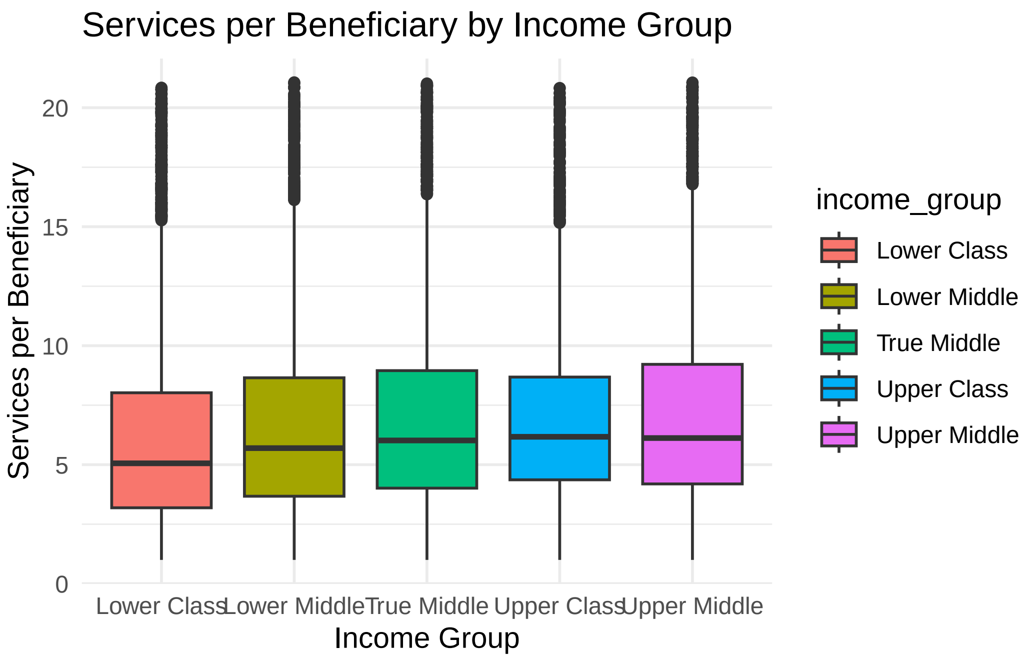
```
# payment per beneficiary
ggplot(df, aes(x = income_group, y = payment_per_beneficiary, fill = income_group)) +
  geom_boxplot() +
  theme_minimal() +
  labs(
    title = "Payment per Beneficiary by Income Group",
    x = "Income Group",
    y = "Payment per Beneficiary"
  )
```



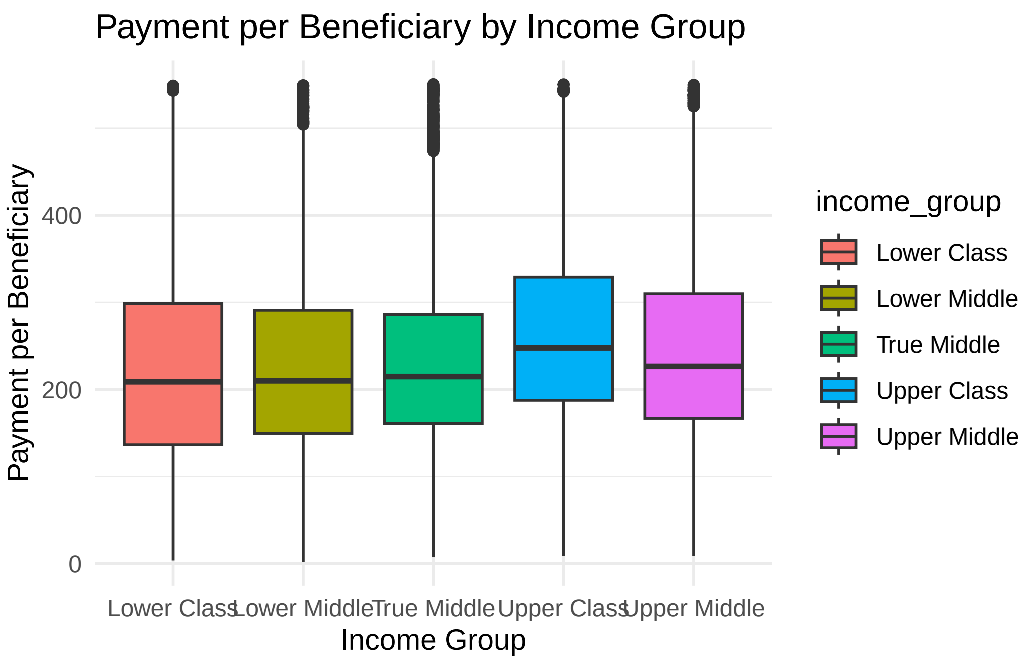
```
#Deal with outliers
remove_outliers <- function(df, col) {
  Q1 <- quantile(df[[col]], 0.25, na.rm = TRUE)
  Q3 <- quantile(df[[col]], 0.75, na.rm = TRUE)
  IQR <- Q3 - Q1  # Interquartile range

  lower_bound <- Q1 - 1.5 * IQR
  upper_bound <- Q3 + 1.5 * IQR

  df <- df %>%
    filter(df[[col]] >= lower_bound & df[[col]] <= upper_bound)

  return(df)
}

df <- remove_outliers(df, "services_per_beneficiary")
df <- remove_outliers(df, "payment_per_beneficiary")
df <- remove_outliers(df, "avg_agi")
```

```
library(ggplot2)

# visualization across income groups
# boxplot services per beneficary
```

```r
ggplot(df, aes(x = income_group, y = services_per_beneficiary, fill = income_group)) +
  geom_boxplot() +
  theme_minimal() +
  labs(
    title = "Services per Beneficiary by Income Group",
    x = "Income Group",
    y = "Services per Beneficiary"
  )
```



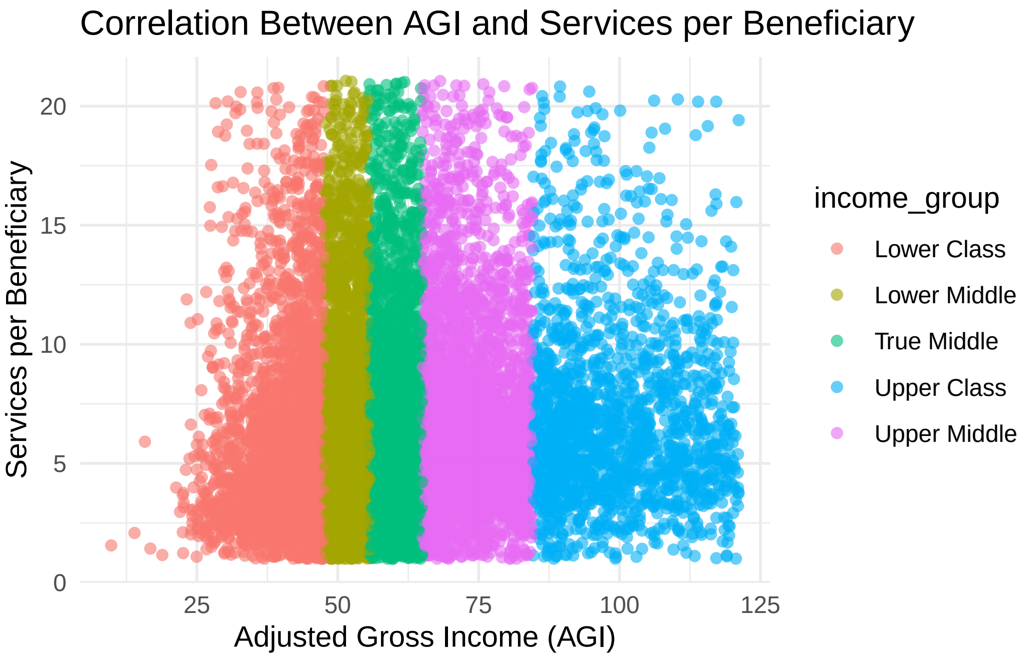
```r
# payment per beneficiary
ggplot(df, aes(x = income_group, y = payment_per_beneficiary, fill = income_group)) +
  geom_boxplot() +
  theme_minimal() +
  labs(
    title = "Payment per Beneficiary by Income Group",
    x = "Income Group",
    y = "Payment per Beneficiary"
  )
```



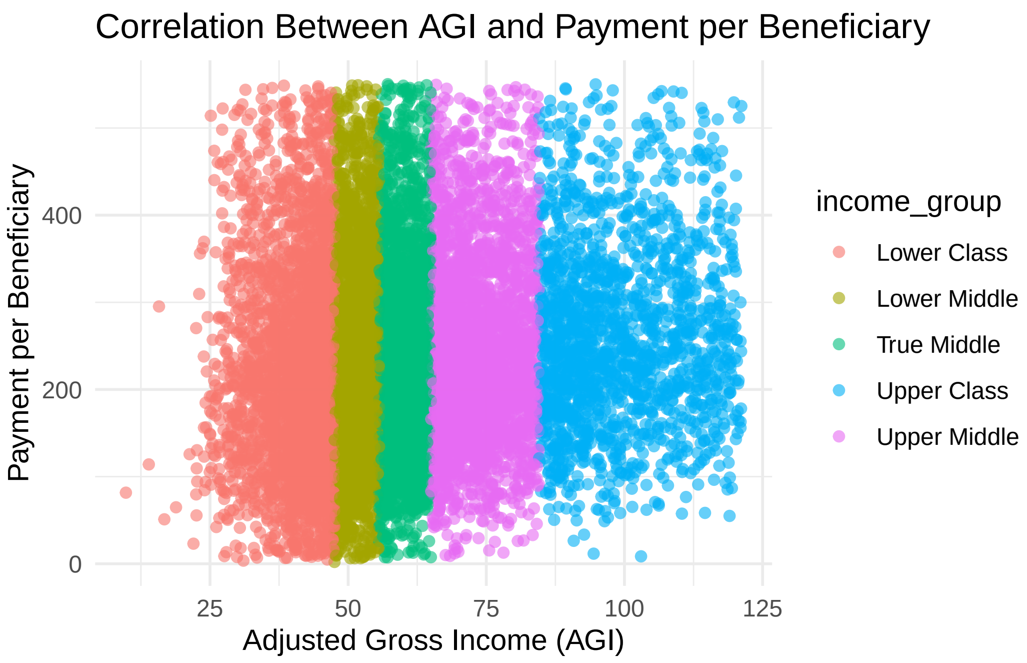
```
# correlation across groups
cor_services_income <- cor(df$avg_agi, df$services_per_beneficiary, use = "complete.obs")
cor_payments_income <- cor(df$avg_agi, df$payment_per_beneficiary, use = "complete.obs")

# list correlation results
list(
  correlation_services_income = cor_services_income,
  correlation_payments_income = cor_payments_income
)
```

```
$correlation_services_income
[1] 0.06589985

$correlation_payments_income
[1] 0.1152638
```

```
# correlation scatterplot services per beneficiary
ggplot(df, aes(x = avg_agi, y = services_per_beneficiary, color = income_group)) +
  geom_point(alpha = 0.6) +
  theme_minimal() +
  labs(
    title = "Correlation Between AGI and Services per Beneficiary",
    x = "Adjusted Gross Income (AGI)",
    y = "Services per Beneficiary"
  )
```



```
ggplot(df, aes(x = avg_agi, y = payment_per_beneficiary, color = income_group)) +
  geom_point(alpha = 0.6) +
  theme_minimal() +
  labs(
    title = "Correlation Between AGI and Payment per Beneficiary",
    x = "Adjusted Gross Income (AGI)",
    y = "Payment per Beneficiary"
  )
```



```r
df$income_group <- factor(df$income_group, levels = c("Lower Class", "Lower Middle", "True Mi

anova_services <- aov(services_per_beneficiary ~ income_group, data = df)
anova_payments <- aov(payment_per_beneficiary ~ income_group, data = df)

summary(anova_services)
```

```
                Df Sum Sq Mean Sq F value Pr(>F)    
income_group     4   1619   404.8    23.7 <2e-16 ***
Residuals    13474 230178    17.1                   
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(anova_payments)
```

```
                Df    Sum Sq Mean Sq F value Pr(>F)    
income_group     4   2121535  530384   44.38 <2e-16 ***
Residuals    13474 161009725   11950                   
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

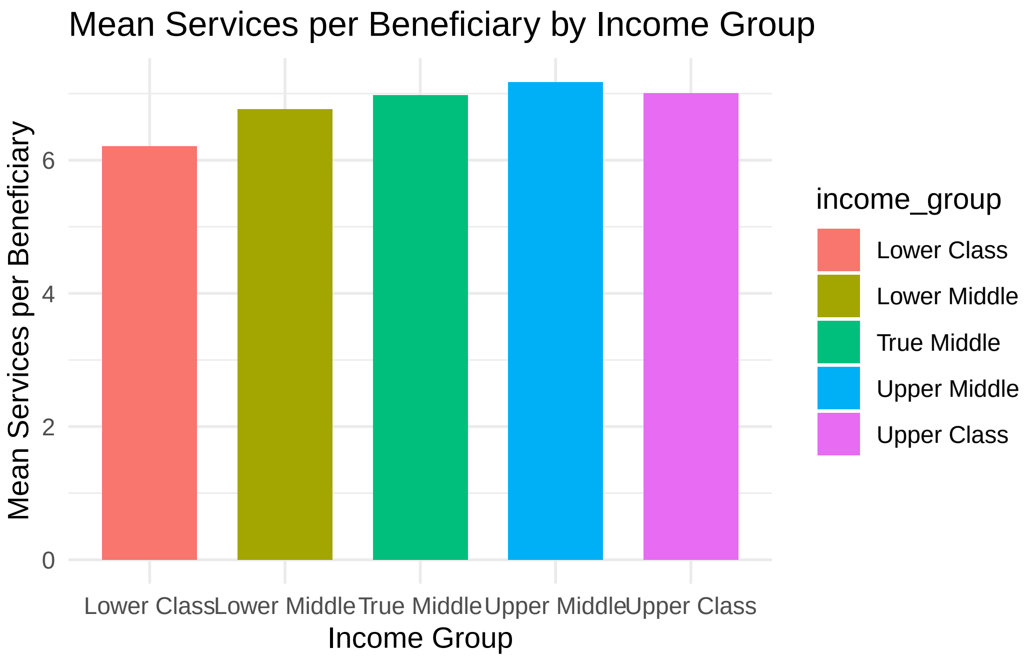
```r
tukey_services <- TukeyHSD(anova_services)
tukey_payments <- TukeyHSD(anova_payments)

print(tukey_services)
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = services_per_beneficiary ~ income_group, data = df)

$income_group
                                diff         lwr       upr     p adj
Lower Middle-Lower Class   0.5586785  0.26355949 0.8537975 0.0000024
True Middle-Lower Class    0.7709406  0.47658841 1.0652929 0.0000000
Upper Middle-Lower Class   0.9662782  0.67337006 1.2591864 0.0000000
Upper Class-Lower Class    0.7969163  0.45613717 1.1376955 0.0000000
True Middle-Lower Middle   0.2122621 -0.08360618 0.5081304 0.2872618
Upper Middle-Lower Middle  0.4075997  0.11316804 0.7020314 0.0014996
Upper Class-Lower Middle   0.2382378 -0.10385173 0.5803274 0.3173119
Upper Middle-True Middle   0.1953376 -0.09832549 0.4890006 0.3649774
Upper Class-True Middle    0.0259757 -0.31545256 0.3674039 0.9995878
Upper Class-Upper Middle  -0.1693619 -0.50954595 0.1708222 0.6545292
```

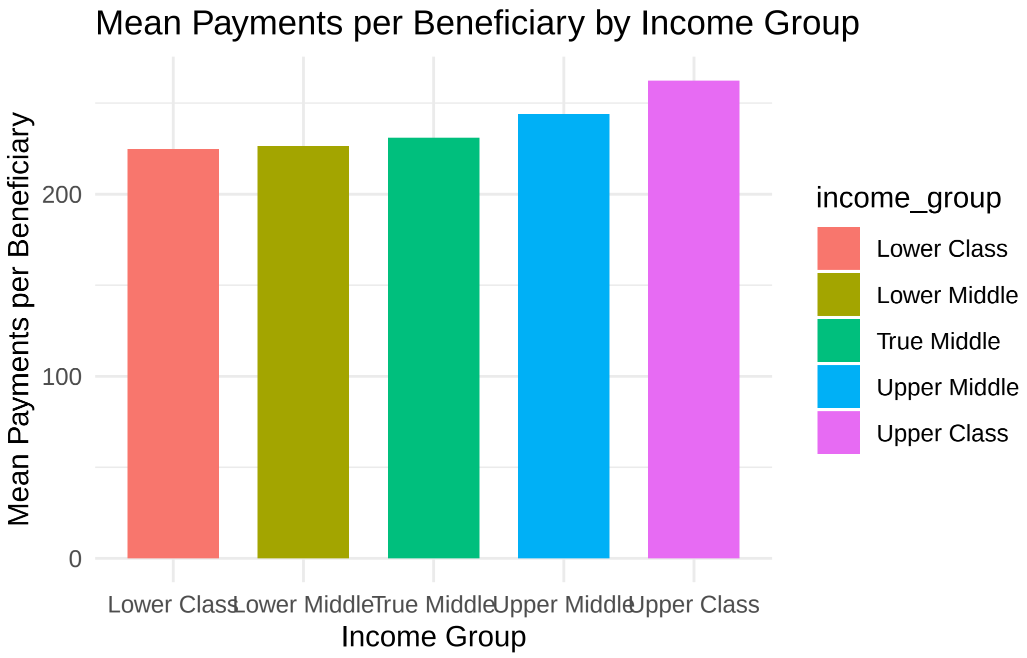
```r
print(tukey_payments)
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = payment_per_beneficiary ~ income_group, data = df)

$income_group
                               diff       lwr       upr     p adj
Lower Middle-Lower Class   1.544036 -6.261299  9.349371 0.9831977
True Middle-Lower Class    6.233793 -1.551262 14.018848 0.1855390
Upper Middle-Lower Class  19.105242 11.358380 26.852104 0.0000000
Upper Class-Lower Class   37.740088 28.727129 46.753047 0.0000000
True Middle-Lower Middle   4.689757 -3.135395 12.514910 0.4748652
Upper Middle-Lower Middle 17.561206  9.774050 25.348362 0.0000000
Upper Class-Lower Middle  36.196052 27.148436 45.243668 0.0000000
Upper Middle-True Middle  12.871449  5.104621 20.638277 0.0000608
Upper Class-True Middle   31.506295 22.476169 40.536421 0.0000000
Upper Class-Upper Middle  18.634846  9.637626 27.632066 0.0000002
```

```r
library(ggplot2)
library(dplyr)

summary_df <- df %>%
  group_by(income_group) %>%
  summarize(
    mean_services = mean(services_per_beneficiary, na.rm = TRUE),
    mean_payments = mean(payment_per_beneficiary, na.rm = TRUE),

  )

ggplot(summary_df, aes(x = income_group, y = mean_services, fill = income_group)) +
  geom_bar(stat = "identity", position = position_dodge(), width = 0.7) +
  theme_minimal() +
  labs(
    title = "Mean Services per Beneficiary by Income Group",
    x = "Income Group",
    y = "Mean Services per Beneficiary"
  )
```



```r
ggplot(summary_df, aes(x = income_group, y = mean_payments, fill = income_group)) +
  geom_bar(stat = "identity", position = position_dodge(), width = 0.7) +
  theme_minimal() +
  labs(
    title = "Mean Payments per Beneficiary by Income Group",
    x = "Income Group",
    y = "Mean Payments per Beneficiary"
  )
```


Mean Payments per Beneficiary by Income Group
Mean Payments per Beneficiary
Lower Class
Lower Middle
True Middle
Upper Middle
Upper Class
Income Group
income_group
Lower Class
Lower Middle
True Middle
Upper Middle
Upper Class