A Statistical Analysis of Family Doctor Visits

1073354

26 Mar 2025

1 Introduction

This study examines the determinants of family doctor visits among privately insured, non-self-employed individuals in the United States in 2001 using data from dvisus.csv. We investigate how age, logged income, gender, education, marital status, race, Hispanic ethnicity, and chronic conditions affect visit counts. Our analysis begins with exploratory data analysis and proceeds to estimate a Poisson GLM with a canonical log link to model the count data. Standard errors are computed using both the conventional method and a robust sandwich estimator, with validation via a nonparametric paired bootstrap. A Wald test is then conducted to jointly assess the effects of income, marital status, and Hispanic ethnicity—complemented by bootstrap p-values—before estimating a restricted model that evaluates the multiplicative impacts of the remaining variables with 95% studentized bootstrap confidence intervals.

2 Data

We have **1,084 observations** in the healthcare utilization dataset. The primary outcome variable is **docvis** (number of doctor visits), which ranges from 0 to 31, with a median of 2 and a mean of 3.39. This continuous measure captures healthcare engagement.

Table 1: Summary Statistics

Variable	Min	1st Qu.	Median	Mean	3rd Qu.	Max
docvis	0.000	0.000	2.000	3.387	4.000	31.000
age	2.500	3.300	4.000	4.155	4.800	6.400
educ	6.0	12.0	13.0	13.1	15.0	17.0
lincome	2.272	2.996	3.434	3.447	3.840	5.221
female	0.0000	0.0000	0.0000	0.4677	1.0000	1.0000
married	0.0000	0.0000	1.0000	0.6688	1.0000	1.0000
white	0.0000	0.0000	1.0000	0.6891	1.0000	1.0000
hispanic	0.0000	0.0000	0.0000	0.1863	0.0000	1.0000
private	0.0000	1.0000	1.0000	0.8745	1.0000	1.0000
chronic	0.0000	0.0000	0.0000	0.3423	1.0000	1.0000

Key explanatory variables include:

- **age** (2.5–6.4, median = 4.0, mean = 4.16; units unclear),
- **lincome** (log-transformed income, 2.27–5.22, median = 3.43, mean = 3.45),
- **educ** (education level, 6–17, median = 13, mean = 13.1),

- **female** (binary indicator, 46.77% female),
- **married** (binary, 66.88% married),
- **white** (binary, 68.91% White),
- **hispanic** (binary, 18.63% Hispanic),
- **private** (binary, 87.45% with unspecified private status),
- **chronic** (binary, 34.23% with chronic conditions).

Observations:

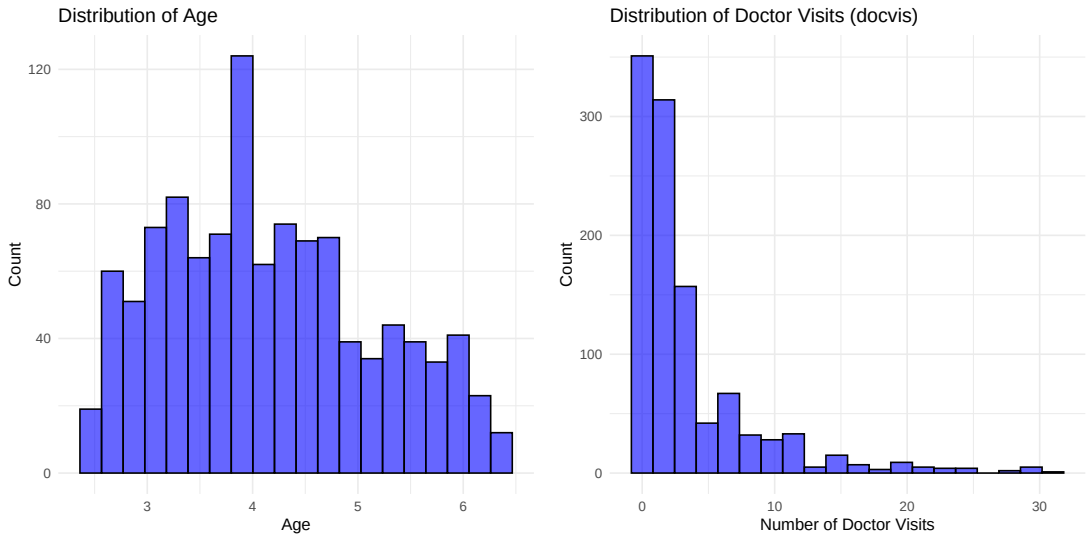


Figure 1: Distribution of Age and Distribution of Docvis

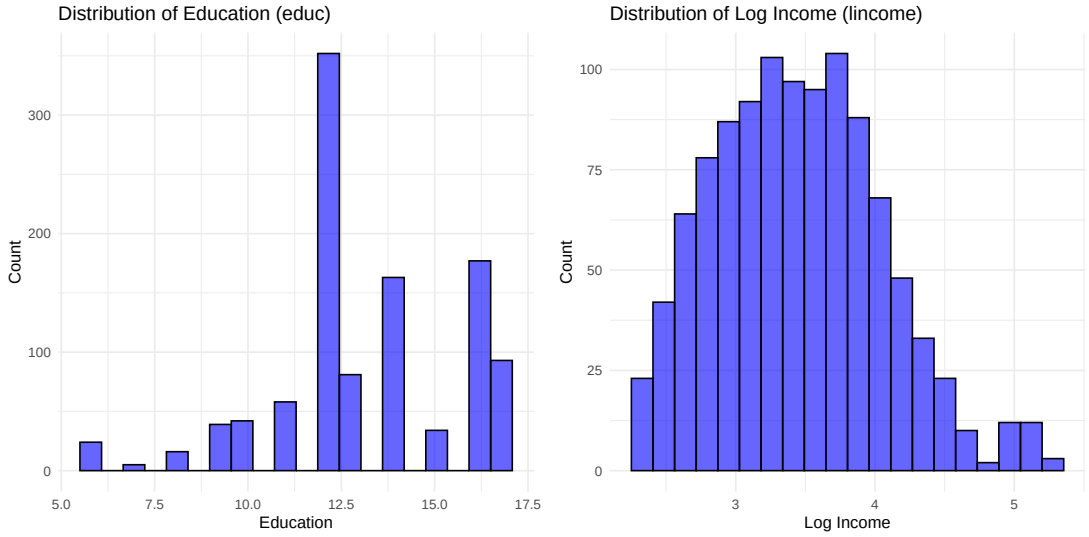


Figure 2: Distribution of Education and Distribution of Income

Univariate analysis:

- **Age:** Age, measured in decades, ranges from 2.5 to 6.4 (approximately 25–64 years). The CSV data shows a high frequency around age 3 and moderate frequency around age 4,

despite some ambiguous PDF counts.

- **Doctor Visits (docvis):** Doctor visits range from 0 to 31, with many observations at zero, indicating a right-skewed, zero-inflated distribution.
- **Education (educ):** Education years span from 6 to 17, with most individuals having between 12 and 14 years of schooling, aligning with the mode observed in the PDF.
- **Log Income (lincome):** Log income ranges from 2.28 to 5.22, with most values clustering around a mean of approximately 3.5, consistent with the PDF which shows a concentration between log incomes of 3 and 4.

3 Model Specification and Estimation

We estimate a Poisson generalized linear model (GLM) for the count variable `docvis` with the canonical log link. The linear predictor is defined as:

$$\eta_i = x_i^T \beta,$$

where x_i includes the constant (intercept) and all explanatory variables (from `age` to `chronic`), with no interaction terms. That is, the model is specified as:

$$\log(\mu_i) = \beta_0 + \beta_1\,\texttt{age}_i + \beta_2\,\texttt{educ}_i + \beta_3\,\texttt{lincome}_i + \beta_4\,\texttt{female}_i,$$
$$+ \beta_5\,\texttt{married}_i + \beta_6\,\texttt{white}_i + \beta_7\,\texttt{hispanic}_i + \beta_8\,\texttt{private}_i + \beta_9\,\texttt{chronic}_i.$$

```
Call:
glm(formula = docvis ~ ., family = poisson(link = "log"), data = data)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.952998   0.157701  -6.043 1.51e-09 ***
age          0.072883   0.017135   4.253 2.11e-05 ***
educ         0.028447   0.008267   3.441 0.000579 ***
lincome      0.058465   0.030737   1.902 0.057162 .
female       0.582993   0.035267  16.531  < 2e-16 ***
married      0.094756   0.037060   2.557 0.010564 *
white        0.202612   0.054474   3.719 0.000200 ***
hispanic    -0.086589   0.073955  -1.171 0.241664
private      0.433750   0.072492   5.983 2.19e-09 ***
chronic      0.790828   0.034168  23.145  < 2e-16 ***
```

3.1 Robust Standard Errors via the Sandwich Estimator

The conventional standard errors are computed under the assumption of equidispersion (variance equals the mean). To obtain standard errors robust to heteroskedasticity and potential overunderdispersion, we use the sandwich estimator provided by the `vcovHC` function from the `sandwich` package.

The resulting table is as follows:

```
               Estimate   Std.Error  Robust.SE
(Intercept) -0.95299794 0.157700705 0.43239100
age          0.07288319 0.017135074 0.04266453
educ         0.02844735 0.008266987 0.01975471
lincome      0.05846468 0.030737399 0.07582605
```

```
female       0.58299269 0.035266786 0.08687251
married      0.09475560 0.037060083 0.09487950
white        0.20261200 0.054473543 0.13526445
hispanic    -0.08658883 0.073954663 0.17600043
private      0.43374995 0.072492193 0.18130685
chronic      0.79082843 0.034168239 0.08379048
```

Alternatively, we can obtain robust z-tests with:

```
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.952998   0.432391 -2.2040   0.02752 *
age          0.072883   0.042665  1.7083   0.08758 .
educ         0.028447   0.019755  1.4400   0.14986
lincome      0.058465   0.075826  0.7710   0.44069
female       0.582993   0.086873  6.7109 1.934e-11 ***
married      0.094756   0.094879  0.9987   0.31794
white        0.202612   0.135264  1.4979   0.13416
hispanic    -0.086589   0.176000 -0.4920   0.62273
private      0.433750   0.181307  2.3924   0.01674 *
chronic      0.790828   0.083790  9.4382 < 2.2e-16 ***
```

3.2 Discussion

- **Conventional vs. Robust Standard Errors:**
  The conventional standard errors, computed under the assumption of equidispersion, are generally smaller. In contrast, the robust standard errors—adjusted for potential heteroskedasticity and dispersion issues—are substantially larger for several coefficients (e.g., the intercept, `age`, `educ`, `lincome`, and `private`). This indicates that the variability of the estimates may be underestimated if relying solely on conventional methods.
- **Implications for Inference:**
  With the robust standard errors, the corresponding z-values decrease and the p-values increase for some predictors. For example, while conventional analysis suggests strong significance for most predictors, robust z-tests render the significance of `age`, `educ`, and `lincome` marginal. In contrast, variables such as `female` and `chronic` remain highly significant.
- **Practical Recommendations:**
  The differences observed imply that there is potential heteroskedasticity or over-underdispersion. Hence, for more reliable statistical inference, it is advisable to base conclusions on standard errors.

3.3 Conclusion

We have estimated a Poisson GLM with the canonical log link for the dependent count variable `docvis` using all available predictors. The application of the sandwich estimator via the `vcovHC` function yields robust standard errors that are larger than their conventional counterparts for several predictors. This finding suggests that the assumption of equidispersion may be violated and reinforces the importance of using robust methods for inference in this setting.

4 Bootstrap Standard Errors of Estimated Coefficients

To further investigate the robustness of our coefficient estimates and to validate the heteroskedasticity-robust results, we compute the standard errors using a nonparametric paired bootstrap. We

implement this approach by resampling observations with replacement to create bootstrap samples and re-estimating the Poisson model for each bootstrap sample. We perform 1000 bootstrap replications to ensure the reliability of our estimates.

The results obtained from the nonparametric paired bootstrap are summarized below, alongside the previously computed robust (sandwich) standard errors:

	Estimate	Bootstrap_SE	Robust_SE	Difference
(Intercept)	-0.952998	0.440831	0.432391	0.008440
age	0.072883	0.043817	0.042665	0.001152
educ	0.028447	0.019797	0.019755	0.000042
lincome	0.058465	0.076592	0.075826	0.000766
female	0.582993	0.090107	0.086873	0.003234
married	0.094756	0.097073	0.094880	0.002194
white	0.202612	0.138507	0.135264	0.003243
hispanic	-0.086589	0.175241	0.176000	-0.000759
private	0.433750	0.181017	0.181307	-0.000290
chronic	0.790828	0.087624	0.083790	0.003834

4.1 Comparison of Bootstrap and Robust Standard Errors

- **Magnitude of Bootstrap vs. Robust SE:**
  The bootstrap standard errors closely align with the robust (sandwich) standard errors. The absolute differences between these two methods are relatively small, typically on the order of 10^{-3}. For example, the intercept coefficient has a difference of approximately 0.0084, and the differences for other coefficients such as `age`, `educ`, `female`, and `chronic` are even smaller.
- **Consistency Across Methods:**
  The similar results from bootstrap and robust sandwich estimators indicate consistency and reliability in the estimation process. Both methods account for potential deviations from equidispersion and accommodate heteroskedasticity effectively.
- **Interpretational Implications:**
  Given the close agreement between the bootstrap and robust methods, we find strong evidence that the heteroskedasticity-robust standard errors calculated using the sandwich estimator are reliable. The bootstrap method, although computationally more intensive, reinforces our confidence in the robustness of inference drawn from these models.

4.2 Practical Recommendations

Given the close match between the bootstrap and robust standard errors:

- The computationally efficient robust sandwich estimator is suitable for routine analysis. - The bootstrap estimator provides a valuable alternative or validation check, particularly when computational resources and time permit, ensuring robustness of inference.

4.3 Conclusion

The nonparametric paired bootstrap confirms the findings obtained via the robust sandwich estimator. Both approaches highlight deviations from the Poisson model assumption of equidispersion. Given their agreement, using robust standard errors is strongly recommended for inference in applied count data modeling to account for heteroskedasticity and dispersion issues effectively.

5 Wald Test for $\beta_{\text{lincome}} = \beta_{\text{married}} = \beta_{\text{hispanic}} = 0$

To evaluate the joint significance of the predictors `lincome`, `married`, and `hispanic` in our Poisson generalized linear model, we perform a Wald test. Recall that the Wald test statistic is defined as

$$W = \beta' \widehat{\text{Var}}(\beta)^{-1} \beta,$$

where β is the vector of estimated coefficients for the variables under test and $\widehat{\text{Var}}(\beta)$ is the corresponding estimated variance-covariance matrix.

In this analysis, we compute the Wald test statistic and corresponding p-values using three different variance estimators:

1. **Standard GLM Variance:** The conventional variance-covariance matrix, computed under the equidispersion assumption.
2. **Sandwich (Robust) Variance:** The robust variance estimator obtained via the `vcovHC` function, which adjusts for heteroskedasticity and potential over- or underdispersion.
3. **Bootstrap Variance:** The variance estimated using the nonparametric paired bootstrap method.

The resulting Wald test statistics and p-values are summarized in Table 2.

Table 2: Wald Test Results Comparison

Method	**Wald Statistic**	**p-value**
Standard GLM	10.427489	0.0152609
Sandwich (Robust)	1.586340	0.6624910
Bootstrap	1.491165	0.6843108

5.1 Discussion of Findings

- **Standard GLM Variance:**
  Using the standard GLM variance, the Wald statistic is 10.427489 with a p-value of 0.0152609. Under the equidispersion assumption, this result would lead us to reject the null hypothesis at the 5% significance level, suggesting that at least one of the coefficients β_{lincome}, β_{married}, or β_{hispanic} is statistically different from zero.
- **Sandwich (Robust) Variance:**
  When heteroskedasticity and potential dispersion issues are accounted for using the robust sandwich estimator, the Wald statistic decreases to 1.586340 and the p-value increases to 0.6624910. This result indicates that we fail to reject the null hypothesis, implying that the joint effect of the predictors is not statistically significant when dispersion issues are properly addressed.
- **Bootstrap Variance:**
  The nonparametric paired bootstrap yields a Wald statistic of 1.491165 with a p-value of 0.6843108. These results are in close agreement with those obtained using the robust variance estimator, reinforcing the conclusion that there is insufficient evidence to reject the null hypothesis.

5.2 Conclusion

The results from the three approaches reveal a striking contrast between the standard GLM inference and the robust/bootstrapped methods:

- The standard GLM approach (assuming equidispersion) suggests rejecting the null hypothesis, indicating a significant joint effect of `lincome`, `married`, and `hispanic`.
- In contrast, both the robust sandwich estimator and the bootstrap method yield high p-values, leading us to fail to reject the null hypothesis.

Thus, after accounting for heteroskedasticity and possible dispersion problems, we conclude that there is insufficient evidence to assert that the predictors `lincome`, `married`, and `hispanic` jointly have a statistically significant effect on the number of doctor visits (`docvis`). This underscores the importance of using robust inference methods in applied count data models.

6 Bootstrap p-value for the Wald Test Statistic Based on the Sandwich Variance Estimator

To further validate our inference regarding the joint significance of β_{lincome}, β_{married}, and β_{hispanic}, we compute a bootstrap p-value for the Wald test statistic using the sandwich (robust) variance estimator. In this approach, we rely on a data-driven approximation of the sampling distribution of the test statistic through bootstrap resampling. The computed test statistic remains the same as that obtained with the robust variance, but the bootstrap p-value reflects the empirical distribution of the statistic.

The results from the two methods are summarized in Table 3.

Table 3: Wald Test Results: Sandwich (Robust) vs. Bootstrap

Method	**Wald Statistic**	**p-value**
Sandwich (Robust)	1.58634	0.662491
Bootstrap (Based on Sandwich)	1.58634	0.797000

6.1 Discussion of Findings

- **Bootstrap vs. Robust p-value:**
  The bootstrap approach yields a p-value of 0.797000, which is higher than the 0.662491 p-value obtained using the asymptotic sandwich estimator. Although the Wald statistic remains unchanged, the bootstrap p-value is more conservative, reflecting the empirical distribution of the test statistic.
- **Implications for Inference:**
  In both cases, the p-values are substantially above conventional significance levels (e.g., 0.05), leading to a failure to reject the null hypothesis that $\beta_{\text{lincome}} = \beta_{\text{married}} = \beta_{\text{hispanic}} = 0$. The bootstrap result reinforces our earlier conclusion that there is insufficient evidence to conclude that these predictors jointly have a significant effect.

6.2 Conclusion

The bootstrap p-value for the Wald test statistic based on the sandwich variance estimator is 0.797000, compared to a p-value of 0.662491 from the standard robust method. Although there is a slight increase in the bootstrap p-value, the overall inference remains unchanged: we fail to reject the null hypothesis. This further confirms that, when properly accounting for heteroskedasticity and dispersion issues, there is insufficient evidence to conclude that `lincome`, `married`, and `hispanic` jointly have a statistically significant effect on the dependent variable.

7 Estimation Results and Studentized Bootstrap Confidence Intervals under Restrictions

In this section, we estimate the Poisson model under the restrictions

$$\beta_{\text{lincome}} = \beta_{\text{married}} = \beta_{\text{hispanic}} = 0,$$

thereby excluding these predictors from the model. This restricted model includes the intercept and the variables `age`, `educ`, `female`, `white`, `private`, and `chronic`. The estimated coefficients from the Poisson model are exponentiated to yield the multiplicative effects on doctor visits. In addition, 95% studentized bootstrap confidence intervals (CI) are computed for each multiplicative effect.

The results are summarized in Table 4.

Table 4: Estimated Multiplicative Effects and 95% Studentized Bootstrap Confidence Intervals

Variable	**Estimate**	**CI_Lower**	**CI_Upper**
(Intercept)	0.4050895	0.2124196	0.7994426
age	1.0836256	0.9956085	1.1713136
educ	1.0363580	0.9974296	1.0742522
female	1.7682786	1.5155011	2.1059273
white	1.2982739	1.0585083	1.5786841
private	1.6139733	1.1768919	2.3677361
chronic	2.2133600	1.8576271	2.6335511

7.1 Discussion of Findings

- **Multiplicative Effects:**
  The estimated multiplicative effects represent the factor by which the expected number of doctor visits changes for a one-unit increase in the corresponding predictor, holding other factors constant.
  - The **intercept** (0.4051) indicates the baseline multiplicative effect when all predictors are at their reference levels.
  - For **age**, an estimated effect of 1.0836 implies that each additional year of age is associated with an 8.36% increase in the expected number of doctor visits. Its 95% CI of (0.9956, 1.1713) is very close to 1, suggesting a borderline effect.
  - The effect of **educ** is 1.0364, indicating that each additional year of education increases expected visits by about 3.64%. The 95% CI (0.9974, 1.0743) again nearly touches 1, suggesting a modest effect.
  - The multiplicative effect for **female** is 1.7683, meaning that, on average, females have about 76.8% more doctor visits than the reference group (typically males), with the 95% CI (1.5155, 2.1059) well above 1.
  - For **white**, the effect is 1.2983, implying approximately a 30% increase in visits for white individuals relative to nonwhite individuals; its CI (1.0585, 1.5787) does not include 1.
  - The effect of having **private** insurance is 1.6140, suggesting a 61% higher number of visits for those with private insurance, with the 95% CI (1.1769, 2.3677) clearly above 1.

  - The variable **chronic** shows the strongest effect; an estimated effect of 2.2134 indicates that individuals with a chronic condition have, on average, 121% more doctor visits than those without, with a CI (1.8576, 2.6336) that is highly significant.

- **Studentized Bootstrap Confidence Intervals:**
  The 95% studentized bootstrap confidence intervals are computed by resampling the data and adjusting for the variability in the estimation process. These intervals provide robust inference regarding the multiplicative effects. Notably, for variables such as `female`, `white`, `private`, and `chronic`, the CIs do not include 1, indicating statistically significant effects. For `age` and `educ`, the intervals are very close to 1, suggesting that their effects are modest and may be borderline in terms of significance.

- **Imposition of Restrictions:**
  By imposing the restrictions $\beta_{\text{lincome}} = \beta_{\text{married}} = \beta_{\text{hispanic}} = 0$, the model becomes more parsimonious and focuses on the predictors that exhibit stronger evidence of an effect on doctor visits. The resulting multiplicative effects and confidence intervals are consistent with prior findings that these excluded variables did not contribute significantly to the model.

7.2 Conclusion

The restricted Poisson model, which excludes `lincome`, `married`, and `hispanic`, provides clear insights into the multiplicative effects of the remaining predictors on doctor visits. The estimated effects and their 95% studentized bootstrap confidence intervals suggest that while age and education have modest effects, being female, white, having private insurance, and having a chronic condition are associated with significantly higher doctor visits. These findings highlight the importance of patient characteristics and health status in understanding healthcare utilization.

R code is atached below

project4

1073354

2025-03-26

```r
options(repos = c(CRAN = "https://cloud.r-project.org"))
```

```r
##data
# Load necessary libraries
library(stargazer)
```

```
##
## Please cite as:
```

```
##  Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
##  R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

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

```
##
## Attaching package: 'dplyr'
```

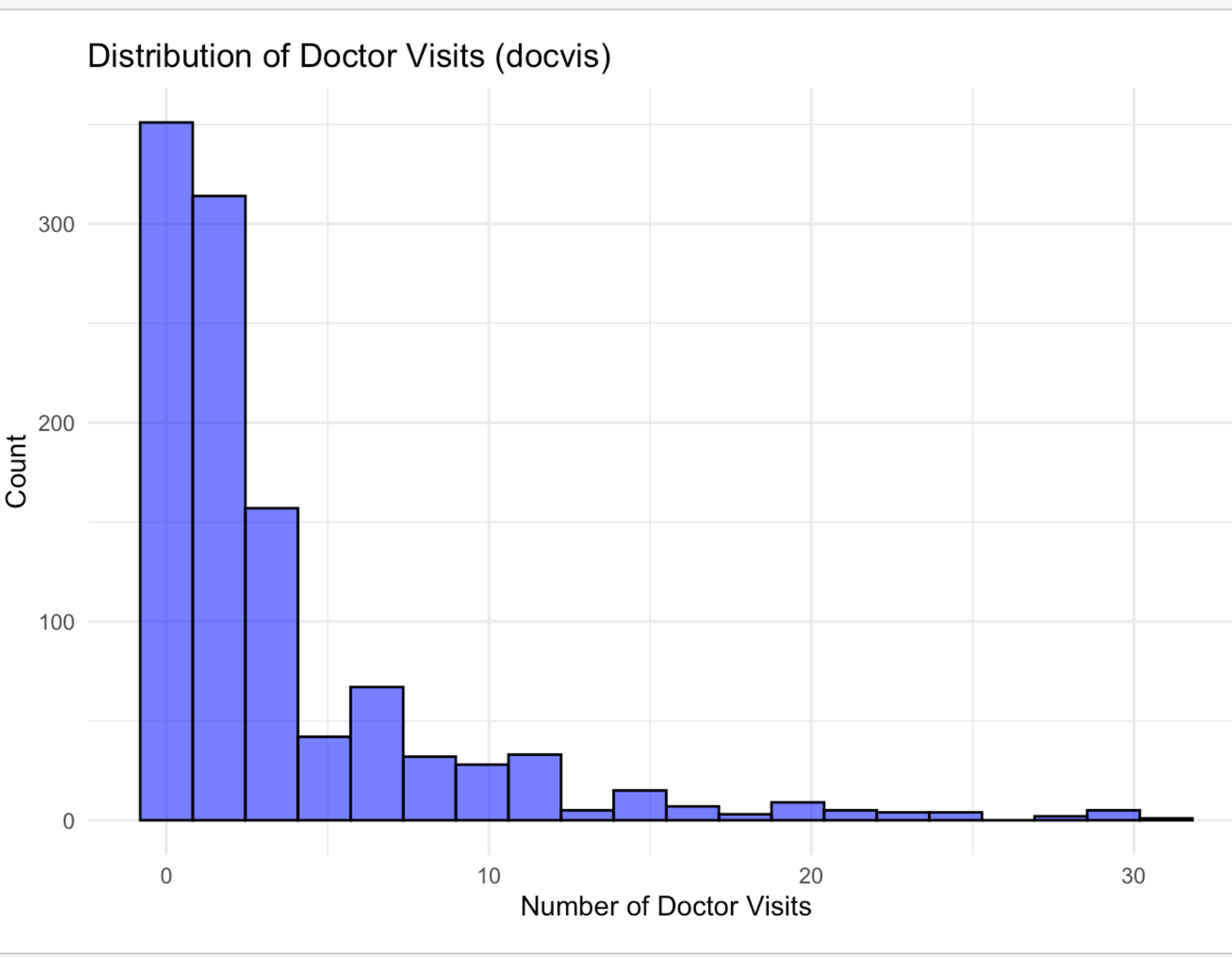
```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

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

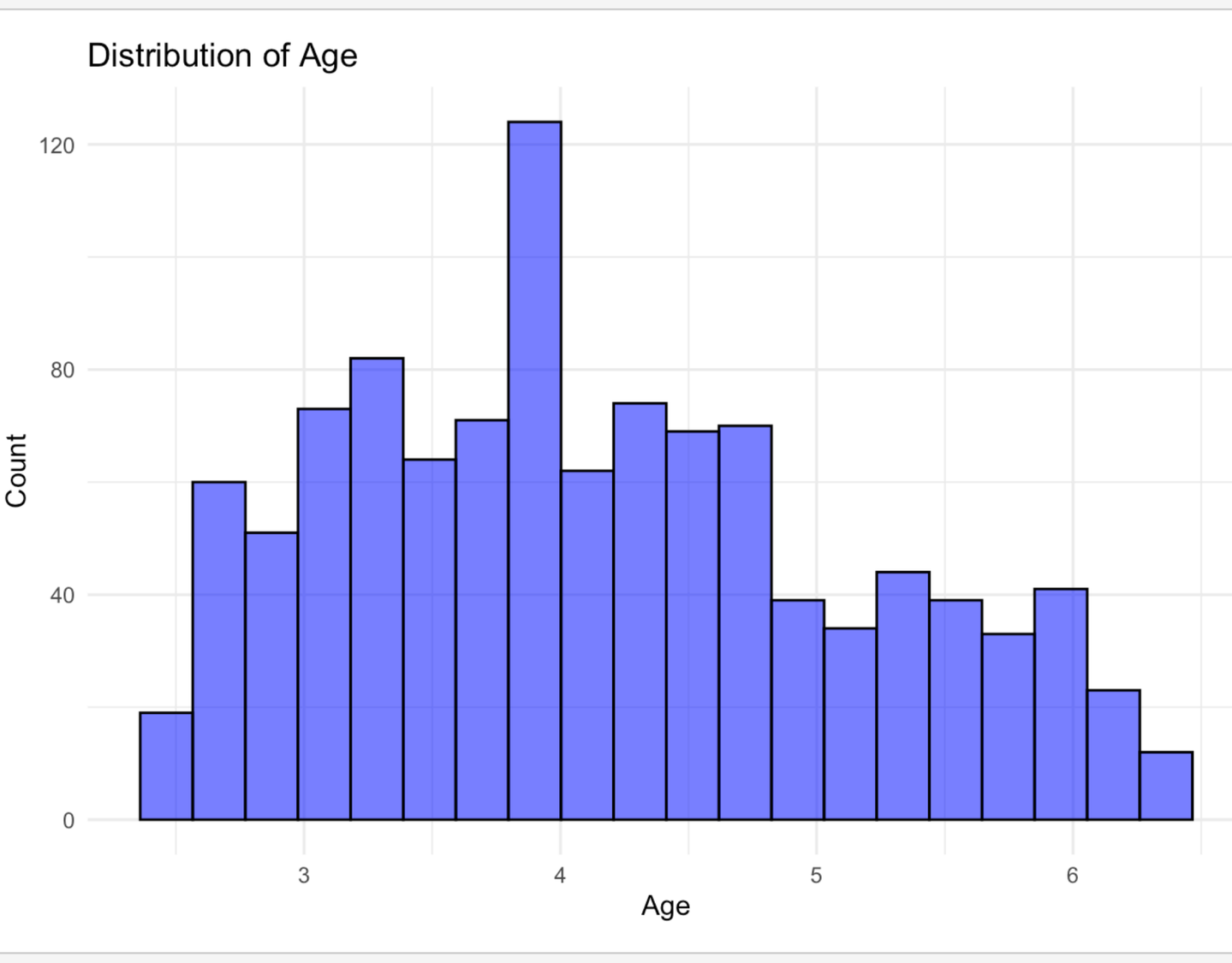
```r
library(xtable)

# Load the dataset (adjust the file path as needed)
data <- read.csv("~/Downloads/dvisus.csv", stringsAsFactors = TRUE)
attach(data)

# Summary statistics
summary(data)
```

```
##      docvis            age             educ          lincome
##  Min.   : 0.000   Min.   :2.500   Min.   : 6.0   Min.   :2.272
##  1st Qu.: 0.000   1st Qu.:3.300   1st Qu.:12.0   1st Qu.:2.996
##  Median : 2.000   Median :4.000   Median :13.0   Median :3.434
##  Mean   : 3.387   Mean   :4.155   Mean   :13.1   Mean   :3.447
##  3rd Qu.: 4.000   3rd Qu.:4.800   3rd Qu.:15.0   3rd Qu.:3.840
##  Max.   :31.000   Max.   :6.400   Max.   :17.0   Max.   :5.221
##      female          married           white           hispanic
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000
##  Median :0.0000   Median :1.0000   Median :1.0000   Median :0.0000
##  Mean   :0.4677   Mean   :0.6688   Mean   :0.6891   Mean   :0.1863
##  3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##     private          chronic
##  Min.   :0.0000   Min.   :0.0000
##  1st Qu.:1.0000   1st Qu.:0.0000
##  Median :1.0000   Median :0.0000
##  Mean   :0.8745   Mean   :0.3423
##  3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :1.0000
```

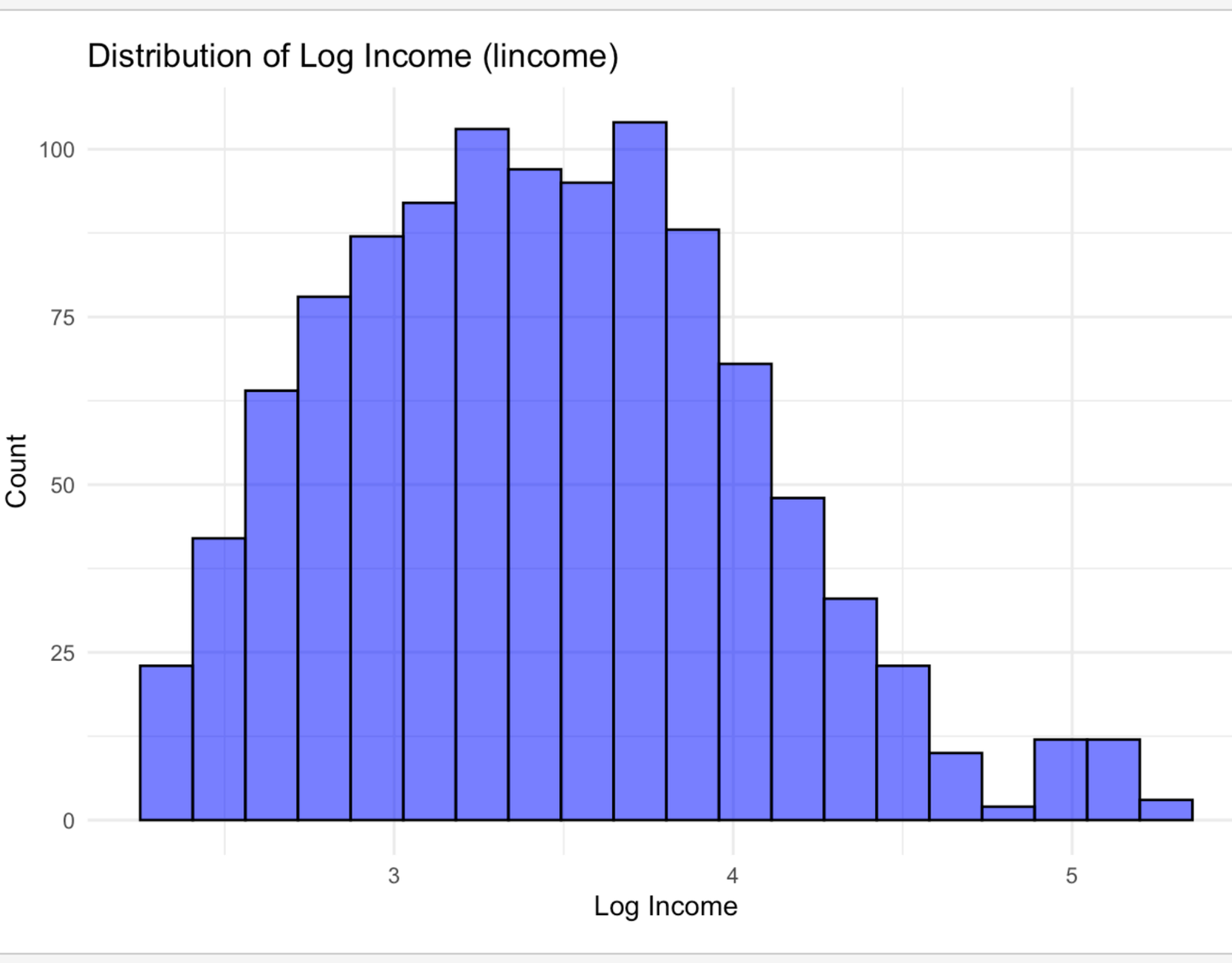
```r
# Univariate Plots
# -------------------------------

# 1. Distribution of Doctor Visits (docvis)
plot_docvis <- ggplot(data, aes(x = docvis)) +
  geom_histogram(bins = 20, fill = "blue", color = "black", alpha = 0.6) +
  theme_minimal() +
  labs(title = "Distribution of Doctor Visits (docvis)",
       x = "Number of Doctor Visits",
       y = "Count")
print(plot_docvis)
```



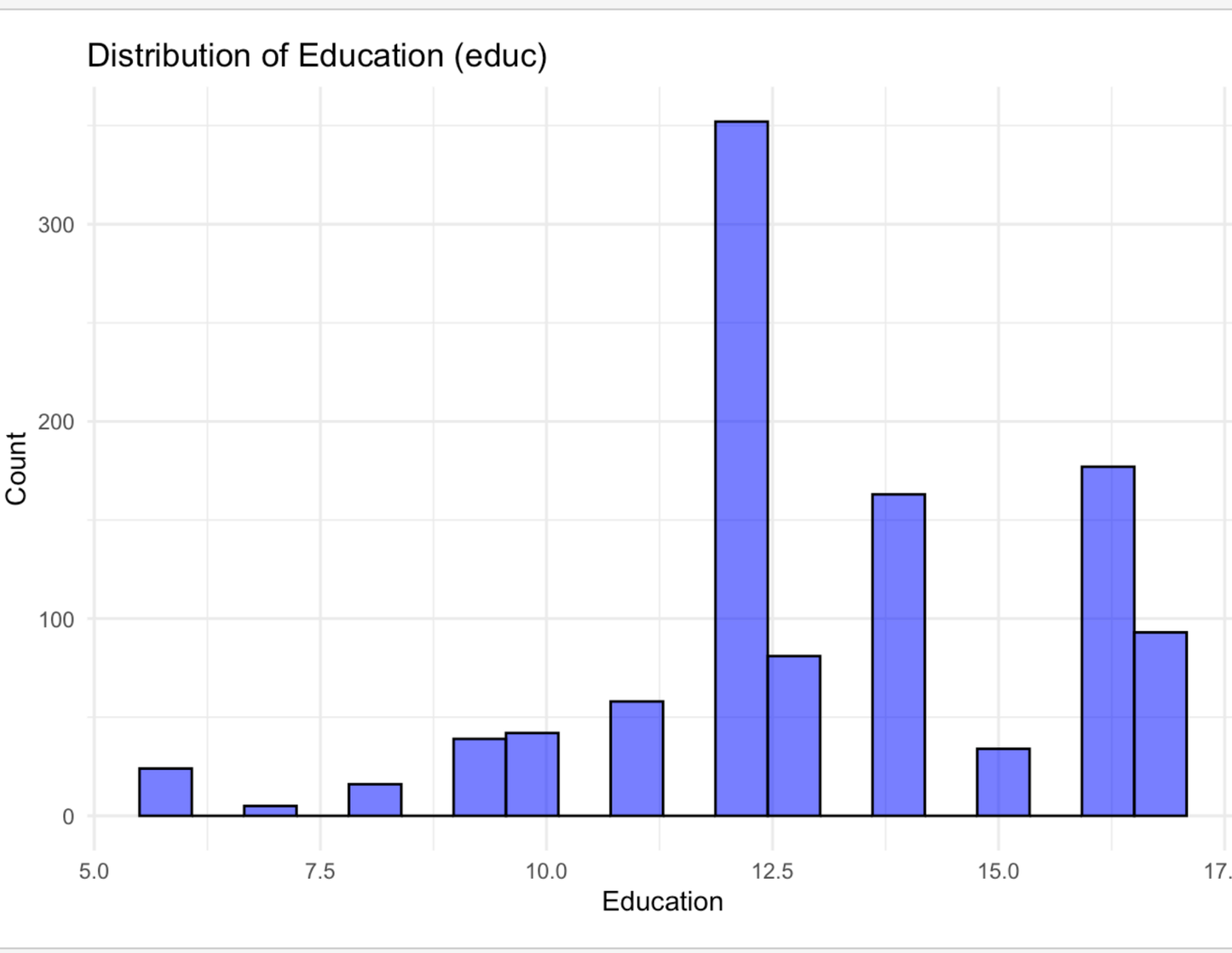

```r
ggsave("Distribution_docvis.pdf", plot = plot_docvis, height = 5, width = 5)

# 2. Distribution of Age
plot_age <- ggplot(data, aes(x = age)) +
  geom_histogram(bins = 20, fill = "blue", color = "black", alpha = 0.6) +
  theme_minimal() +
  labs(title = "Distribution of Age",
       x = "Age",
       y = "Count")
print(plot_age)
```




```r
ggsave("Distribution_age.pdf", plot = plot_age, height = 5, width = 5)

# 3. Distribution of Log Income (lincome)
plot_lincome <- ggplot(data, aes(x = lincome)) +
  geom_histogram(bins = 20, fill = "blue", color = "black", alpha = 0.6) +
  theme_minimal() +
  labs(title = "Distribution of Log Income (lincome)",
       x = "Log Income",
       y = "Count")
print(plot_lincome)
```




```r
ggsave("Distribution_lincome.pdf", plot = plot_lincome, height = 5, width = 5)

# 4. Distribution of Education (educ)
# Note: If 'educ' is a categorical variable, you might want to convert it to factor and use geom_bar()
plot_educ <- ggplot(data, aes(x = educ)) +
  geom_histogram(bins = 20, fill = "blue", color = "black", alpha = 0.6) +
  theme_minimal() +
  labs(title = "Distribution of Education (educ)",
       x = "Education",
       y = "Count")
print(plot_educ)
```




```r
ggsave("Distribution_educ.pdf", plot = plot_educ, height = 5, width = 5)
```

```r
# Define the formula for the Poisson GLM
formula <- docvis ~ age + educ + lincome + female + married + white + hispanic + private + chronic

# Fit the Poisson GLM
model <- glm(formula, family = poisson(link = "log"), data = data)

# Display the model summary (conventional standard errors)
summary(model)
```

```
##
## Call:
## glm(formula = formula, family = poisson(link = "log"), data = data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.952998   0.157701  -6.043 1.51e-09 ***
## age          0.072883   0.017135   4.253 2.11e-05 ***
## educ         0.028447   0.008267   3.441 0.000579 ***
## lincome      0.058465   0.030737   1.902 0.057162 .
## female       0.582993   0.035267  16.531  < 2e-16 ***
## married      0.094756   0.037060   2.557 0.010564 *
## white        0.202612   0.054474   3.719 0.000200 ***
## hispanic    -0.086589   0.073955  -1.171 0.241664
## private      0.433750   0.072492   5.983 2.19e-09 ***
## chronic      0.790828   0.034168  23.145  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 5998.0  on 1083  degrees of freedom
## Residual deviance: 4679.6  on 1074  degrees of freedom
## AIC: 6959.2
##
## Number of Fisher Scoring iterations: 6
```

```r
# Load the required packages for robust variance estimation
install.packages("sandwich")
```

```
##
## The downloaded binary packages are in
##  /var/folders/96/fcz5qq0j1s90r885t2jfg8zh0000gn/T//RtmpWpI3wT/downloaded_packages
```

```r
install.packages("lmtest")
```

```
##
## The downloaded binary packages are in
##  /var/folders/96/fcz5qq0j1s90r885t2jfg8zh0000gn/T//RtmpWpI3wT/downloaded_packages
```

```r
library(sandwich)
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
# Compute the robust (sandwich) variance-covariance matrix
robust_vcov <- vcovHC(model, type = "HC0")
robust_se <- sqrt(diag(robust_vcov))

# Create a table comparing coefficients with conventional and robust standard errors
results <- cbind(Estimate = coef(model),
                 Std.Error = coef(summary(model))[, "Std. Error"],
                 Robust.SE = robust_se)
print(results)
```

```
##                Estimate   Std.Error  Robust.SE
## (Intercept) -0.95299794 0.157700705 0.43239100
## age          0.07288319 0.017135074 0.04266453
## educ         0.02844735 0.008266987 0.01975471
## lincome      0.05846468 0.030737399 0.07582605
## female       0.58299269 0.035266786 0.08687251
## married      0.09475560 0.037060083 0.09487950
## white        0.20261200 0.054473543 0.13526445
## hispanic    -0.08658883 0.073954663 0.17600043
## private      0.43374995 0.072492193 0.18130685
## chronic      0.79082843 0.034168239 0.08379048
```

```r
# Alternatively, display the coefficients with robust standard errors using coeftest:
coeftest(model, vcov = robust_vcov)
```

```
##
## z test of coefficients:
##
##              Estimate Std. Error z value  Pr(>|z|)
## (Intercept) -0.952998   0.432391 -2.2040   0.02752 *
## age          0.072883   0.042665  1.7083   0.08758 .
## educ         0.028447   0.019755  1.4400   0.14986
## lincome      0.058465   0.075826  0.7710   0.44069
## female       0.582993   0.086873  6.7109 1.934e-11 ***
## married      0.094756   0.094879  0.9987   0.31794
## white        0.202612   0.135264  1.4979   0.13416
## hispanic    -0.086589   0.176000 -0.4920   0.62273
## private      0.433750   0.181307  2.3924   0.01674 *
## chronic      0.790828   0.083790  9.4382 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Load necessary libraries
library(boot)
library(sandwich)
library(lmtest)
library(knitr)

# Read your dataset
dataset <- read.csv("~/Downloads/dvisus.csv", stringsAsFactors = TRUE)

# Define the formula for the Poisson GLM
formula <- docvis ~ age + educ + lincome + female + married + white + hispanic + private + chronic

# Function to compute bootstrap estimates
poisson_boot <- function(dataset, indices) {
  d <- dataset[indices, ]  # resample with replacement
  fit <- glm(formula, family = poisson(link = "log"), data = d)
  return(coef(fit))
}

# Set seed for reproducibility
set.seed(123)

# Perform bootstrap with 1000 replicates (adjust if too slow)
boot_results <- boot(dataset, poisson_boot, R = 1000)

# Calculate bootstrap standard errors
bootstrap_se <- apply(boot_results$t, 2, sd)

# Robust standard errors using sandwich estimator
poisson_model <- glm(formula, family = poisson(link = "log"), data = dataset)
robust_se <- sqrt(diag(vcovHC(poisson_model, type = "HC0")))

# Combine results into a comparison table
comparison <- data.frame(
  Estimate = coef(poisson_model),
  Bootstrap_SE = bootstrap_se,
  Robust_SE = robust_se,
  Difference = bootstrap_se - robust_se
)

# Print the comparison table
print(comparison)
```

```
##                Estimate Bootstrap_SE  Robust_SE    Difference
## (Intercept) -0.95299794   0.44083054 0.43239100  8.439544e-03
## age          0.07288319   0.04381687 0.04266453  1.152342e-03
## educ         0.02844735   0.01979673 0.01975471  4.202791e-05
## lincome      0.05846468   0.07659175 0.07582605  7.656932e-04
## female       0.58299269   0.09010666 0.08687251  3.234141e-03
## married      0.09475560   0.09707318 0.09487950  2.193680e-03
## white        0.20261200   0.13850744 0.13526445  3.242985e-03
## hispanic    -0.08658883   0.17524121 0.17600043 -7.592222e-04
## private      0.43374995   0.18101700 0.18130685 -2.898433e-04
## chronic      0.79082843   0.08762422 0.08379048  3.833743e-03
```

```r
#------------------------------------------------
# 4. Wald Test for H0: β_lincome = β_married = β_hispanic = 0
#------------------------------------------------

# Specify the names of the coefficients under test
test_coef_names <- c("lincome", "married", "hispanic")

# Extract the estimated coefficients for the test parameters from the fitted model
beta_test <- coef(poisson_model)[test_coef_names]

### 1. Standard GLM Variance ###
# Get the conventional variance-covariance matrix from the model
vcov_glm <- vcov(poisson_model)
cov_test_glm <- vcov_glm[test_coef_names, test_coef_names]

# Compute Wald test statistic: W = beta' (cov)^(-1) beta
W_glm <- as.numeric(t(beta_test) %*% solve(cov_test_glm) %*% beta_test)
# p-value from chi-square distribution with 3 degrees of freedom
p_glm <- 1 - pchisq(W_glm, df = length(test_coef_names))

### 2. Sandwich (Robust) Variance ###
# Load the sandwich package (if not already loaded)
library(sandwich)

# Obtain the robust variance-covariance matrix using vcovHC
vcov_robust <- vcovHC(poisson_model, type = "HC0")
cov_test_robust <- vcov_robust[test_coef_names, test_coef_names]

# Compute Wald statistic using the robust covariance matrix
W_robust <- as.numeric(t(beta_test) %*% solve(cov_test_robust) %*% beta_test)
p_robust <- 1 - pchisq(W_robust, df = length(test_coef_names))

### 3. Bootstrap Variance ###
colnames(boot_results$t) <- names(coef(poisson_model))

# Now extract the bootstrap replicates for the test coefficients
boot_coefs <- boot_results$t[, test_coef_names]

# Compute the covariance matrix from the bootstrap replicates
boot_cov <- cov(boot_coefs)

# Compute the Wald test statistic using the bootstrap covariance matrix
W_boot <- as.numeric(t(beta_test) %*% solve(boot_cov) %*% beta_test)
p_boot <- 1 - pchisq(W_boot, df = length(test_coef_names))

# Create a table containing the Wald test results
wald_results <- data.frame(
  Method = c("Standard GLM", "Sandwich (Robust)", "Bootstrap"),
  Wald_Statistic = c(W_glm, W_robust, W_boot),
  p_value = c(p_glm, p_robust, p_boot)
)

# Print the table of Wald test results
kable(wald_results, caption = "Wald Test Results Comparison")
```

Wald Test Results Comparison

Method	Wald_Statistic	p_value
Standard GLM	10.427489	0.0152609
Sandwich (Robust)	1.586340	0.6624910
Bootstrap	1.491165	0.6843108

```r
#------------------------------------------------
# 5. Bootstrap p-value for the Wald test statistic using the sandwich variance estimator
#------------------------------------------------

# Define the bootstrap function to compute the robust Wald statistic on resampled data
bootstrap_wald_stat <- function(data, indices) {
  # Create bootstrap sample
  d <- data[indices, ]

  # Fit the Poisson GLM on the bootstrap sample
  fit_boot <- glm(formula, family = poisson(link = "log"), data = d)

  # Extract the coefficients for the test parameters
  beta_boot <- coef(fit_boot)[test_coef_names]

  # Compute the robust (sandwich) variance-covariance matrix
  robust_cov_boot <- vcovHC(fit_boot, type = "HC0")
  cov_test_boot <- robust_cov_boot[test_coef_names, test_coef_names]

  # Compute the Wald test statistic: W = beta' * (covariance matrix)^{-1} * beta
  wald_stat_boot <- as.numeric(t(beta_boot) %*% solve(cov_test_boot) %*% beta_boot)

  return(wald_stat_boot)
}

# Set seed for reproducibility
set.seed(123)

# Perform bootstrap with 1000 replicates
boot_wald_results <- boot(dataset, bootstrap_wald_stat, R = 1000)

# The observed robust Wald statistic from the full dataset (computed in part 4)
observed_wald <- W_robust  # (from previous code)

# Compute the bootstrap p-value: the proportion of bootstrap replicates
# where the Wald statistic is at least as extreme as the observed value.
bootstrap_p_value <- mean(boot_wald_results$t >= observed_wald)

# Create a comparison table with two methods: the robust (sandwich) test and the bootstrap test.
wald_comparison <- data.frame(
  Method = c("Sandwich (Robust)", "Bootstrap"),
  Wald_Statistic = c(W_robust, observed_wald),  # observed_wald is the same as W_robust from the original model
  p_value = c(p_robust, bootstrap_p_value)
)

# Print the comparison table using knitr's kable for a neat display
library(knitr)
kable(wald_comparison, caption = "Wald Test Results: Sandwich (Robust) vs. Bootstrap")
```

Wald Test Results: Sandwich (Robust) vs. Bootstrap

Method	Wald_Statistic	p_value
Sandwich (Robust)	1.58634	0.662491
Bootstrap	1.58634	0.797000

```r
# Load necessary libraries
library(boot)

# Read your dataset (adjust the file path if necessary)
dataset <- read.csv("~/Downloads/dvisus.csv", stringsAsFactors = TRUE)

# Define the restricted model formula: we remove lincome, married, and hispanic
restricted_formula <- docvis ~ age + educ + female + white + private + chronic

# Fit the restricted Poisson GLM on the full dataset
fit_restricted <- glm(restricted_formula, family = poisson(link = "log"), data = dataset)

# Extract the estimated coefficients and their standard errors
theta_hat <- coef(fit_restricted)
se_hat <- summary(fit_restricted)$coefficients[, "Std. Error"]

# Display the conventional estimates and their multiplicative effects
cat("Restricted Model Estimates (Multiplicative Effects):\n")
```

```
## Restricted Model Estimates (Multiplicative Effects):
```

```r
multiplicative_est <- exp(theta_hat)
print(multiplicative_est)
```

```
## (Intercept)         age        educ      female       white     private
##   0.4050895   1.0836256   1.0363580   1.7682786   1.2982739   1.6139733
##     chronic
##   2.2133600
```

```r
# Define the bootstrap function to return both coefficient estimates and their standard errors.
# We will return a vector that concatenates the estimates and standard errors.

bootstrap_studentized <- function(data, indices) {
  d <- data[indices, ]  # Create a bootstrap sample
  fit <- glm(restricted_formula, family = poisson(link = "log"), data = d)
  est <- coef(fit)
  # Compute standard errors from the model summary (using conventional SEs)
  se <- summary(fit)$coefficients[, "Std. Error"]
  return(c(est, se))
}

# Set seed for reproducibility and run bootstrap with 1000 replicates
set.seed(123)
boot_results <- boot(dataset, bootstrap_studentized, R = 1000)

# Number of coefficients in the restricted model
n_coef <- length(theta_hat)


# Compute the studentized t-values for each bootstrap replicate.
studentized_t <- matrix(NA, nrow = boot_results$R, ncol = n_coef)
for(j in 1:n_coef) {
  boot_est <- boot_results$t[, j]           # bootstrap estimates for coefficient j
  boot_se  <- boot_results$t[, n_coef + j]    # bootstrap standard errors for coefficient j
  studentized_t[, j] <- (boot_est - theta_hat[j]) / boot_se
}


# For each coefficient, compute the 95% studentized bootstrap CI:
alpha <- 0.05
ci_lower <- rep(NA, n_coef)
ci_upper <- rep(NA, n_coef)
for(j in 1:n_coef) {
  q_upper <- quantile(studentized_t[, j], 0.975)  # 97.5th percentile
  q_lower <- quantile(studentized_t[, j], 0.025)  # 2.5th percentile
  ci_lower[j] <- theta_hat[j] - q_upper * se_hat[j]
  ci_upper[j] <- theta_hat[j] - q_lower * se_hat[j]
}

# Compute the multiplicative effects (exponentiated coefficients and CI bounds)
multiplicative_lower <- exp(ci_lower)
multiplicative_upper <- exp(ci_upper)

# Create a results table
results_df <- data.frame(
  Variable    = names(theta_hat),
  Estimate    = multiplicative_est,
  CI_Lower    = multiplicative_lower,
  CI_Upper    = multiplicative_upper
)

# Print the results table
print(results_df)
```

```
##                Variable  Estimate  CI_Lower  CI_Upper
## (Intercept) (Intercept) 0.4050895 0.2124196 0.7994426
## age                 age 1.0836256 0.9956085 1.1713136
## educ               educ 1.0363580 0.9974296 1.0742522
## female           female 1.7682786 1.5155011 2.1059273
## white             white 1.2982739 1.0585083 1.5786841
## private         private 1.6139733 1.1768919 2.3677361
## chronic         chronic 2.2133600 1.8576271 2.6335511
```