final project code

David Moran

2024-12-12

```
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
```

```
file_path <- '/Users/david_m123/Desktop/CRDT Data - CRDT.csv'

data <- read.csv(file_path)

dim(data)
```

```
## [1] 5320   54
```

```
colnames(data)
```

```
##  [1] "Date"                          "State"
##  [3] "Cases_Total"                   "Cases_White"
##  [5] "Cases_Black"                   "Cases_Latinx"
##  [7] "Cases_Asian"                   "Cases_AIAN"
##  [9] "Cases_NHPI"                    "Cases_Multiracial"
## [11] "Cases_Other"                   "Cases_Unknown"
## [13] "Cases_Ethnicity_Hispanic"      "Cases_Ethnicity_NonHispanic"
## [15] "Cases_Ethnicity_Unknown"       "Deaths_Total"
## [17] "Deaths_White"                  "Deaths_Black"
## [19] "Deaths_Latinx"                 "Deaths_Asian"
## [21] "Deaths_AIAN"                   "Deaths_NHPI"
## [23] "Deaths_Multiracial"            "Deaths_Other"
## [25] "Deaths_Unknown"                "Deaths_Ethnicity_Hispanic"
## [27] "Deaths_Ethnicity_NonHispanic" "Deaths_Ethnicity_Unknown"
## [29] "Hosp_Total"                    "Hosp_White"
## [31] "Hosp_Black"                    "Hosp_Latinx"
```

```
## [33] "Hosp_Asian"                   "Hosp_AIAN"
## [35] "Hosp_NHPI"                    "Hosp_Multiracial"
## [37] "Hosp_Other"                   "Hosp_Unknown"
## [39] "Hosp_Ethnicity_Hispanic"      "Hosp_Ethnicity_NonHispanic"
## [41] "Hosp_Ethnicity_Unknown"       "Tests_Total"
## [43] "Tests_White"                  "Tests_Black"
## [45] "Tests_Latinx"                 "Tests_Asian"
## [47] "Tests_AIAN"                   "Tests_NHPI"
## [49] "Tests_Multiracial"            "Tests_Other"
## [51] "Tests_Unknown"                "Tests_Ethnicity_Hispanic"
## [53] "Tests_Ethnicity_NonHispanic"  "Tests_Ethnicity_Unknown"
```

```
data$Date <- as.Date(as.character(data$Date), format = "%Y%m%d")


latest_data <- data %>%
  arrange(Date) %>%
  group_by(State) %>%
  slice_tail(n = 1)

# Calculate mortality rates for racial groups
# Mortality rate = (Deaths for the group / Cases for the group) * 100
latest_data <- latest_data %>%
  mutate(
    Mortality_Rate_White = (Deaths_White / Cases_White) * 100,
    Mortality_Rate_Black = (Deaths_Black / Cases_Black) * 100,
    Mortality_Rate_Latinx = (Deaths_Latinx / Cases_Latinx) * 100,
    Mortality_Rate_Asian = (Deaths_Asian / Cases_Asian) * 100,
    Mortality_Rate_AIAN = (Deaths_AIAN / Cases_AIAN) * 100,
    Mortality_Rate_NHPI = (Deaths_NHPI / Cases_NHPI) * 100,
    Mortality_Rate_Multiracial = (Deaths_Multiracial / Cases_Multiracial) * 100
  )


print(latest_data)
```

```
## # A tibble: 56 x 61
## # Groups:   State [56]
##    Date       State Cases_Total Cases_~1 Cases~2 Cases~3 Cases~4 Cases~5 Cases~6
##    <date>     <chr>       <int>    <int>   <int>   <int>   <int>   <int>   <int>
##  1 2021-03-07 AK          59332    18300    1499      NA    2447   12238    1508
##  2 2021-03-07 AL         499819   160347   82790      NA    2273      NA      NA
##  3 2021-03-07 AR         324818   207596   50842      NA    2913    1070    3358
##  4 2021-03-07 AS             NA       NA      NA      NA      NA      NA      NA
##  5 2021-03-07 AZ         826454   308453   25775  244539   11921   40707      NA
##  6 2021-03-07 CA        3501394   546630  111279 1509103  186562    9025   15281
##  7 2021-03-07 CO         435762   181669   12637  119224    6406    2527    1264
##  8 2021-03-07 CT         285330    85469   19651   41523    3019     393      NA
##  9 2021-03-07 DC          41419    10708   20164      NA     914      86      82
## 10 2021-03-07 DE          88354    42730   19768   14532    1842      NA      NA
## # ... with 46 more rows, 52 more variables: Cases_Multiracial <int>,
## #   Cases_Other <int>, Cases_Unknown <int>, Cases_Ethnicity_Hispanic <int>,
## #   Cases_Ethnicity_NonHispanic <int>, Cases_Ethnicity_Unknown <int>,
## #   Deaths_Total <int>, Deaths_White <int>, Deaths_Black <int>,
```

```
## #   Deaths_Latinx <int>, Deaths_Asian <int>, Deaths_AIAN <int>,
## #   Deaths_NHPI <int>, Deaths_Multiracial <int>, Deaths_Other <int>,
## #   Deaths_Unknown <int>, Deaths_Ethnicity_Hispanic <int>, ...
```

```
write.csv(latest_data, "ManipCRDT.csv")
```

```
library(tidyr)    # For data manipulation and melting
library(dplyr)    # For data manipulation
library(stats)    # For linear regression


columns_needed <- c("Mortality_Rate_White", "Mortality_Rate_Black", "Mortality_Rate_Latinx",
                    "Mortality_Rate_Asian", "Mortality_Rate_AIAN", "Cases_Total", "State")
clean_data <- latest_data %>%
  select(all_of(columns_needed)) %>%
  na.omit()


head(clean_data)
```

```
## # A tibble: 6 x 7
## # Groups:   State [6]
##   Mortality_Rate_White Mortality_Rate_Bl~1 Morta~2 Morta~3 Morta~4 Cases~5 State
##                  <dbl>               <dbl>   <dbl>   <dbl>   <dbl>   <int> <chr>
## 1                 2.61                1.68   1.92     1.85   3.34   826454 AZ
## 2                 3.03                2.94   1.62     3.27   2.04  3501394 CA
## 3                 2.13                1.51   1.01     1.80   1.46   435762 CO
## 4                 6.33                4.61   1.88     2.75   0.763  285330 CT
## 5                 2.80                2.44   0.997    1.67   1.98  1023487 GA
## 6                 2.59                3.31   1.50     2.72   1.74  1198335 IL
## # ... with abbreviated variable names 1: Mortality_Rate_Black,
## #   2: Mortality_Rate_Latinx, 3: Mortality_Rate_Asian, 4: Mortality_Rate_AIAN,
## #   5: Cases_Total
```

```
melted_data <- clean_data %>%
  pivot_longer(
    cols = starts_with("Mortality_Rate_"),
    names_to = "Race",
    values_to = "Mortality_Rate"
  )


melted_data <- melted_data %>%
  mutate(Race = gsub("Mortality_Rate_", "", Race))

# Run a linear regression model with mortality rate as the dependent variable and race as the independe
model <- lm(Mortality_Rate ~ Race, data = melted_data)

# Display the regression results
summary(model)
```

```
##
```

```
## Call:
## lm(formula = Mortality_Rate ~ Race, data = melted_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.3181 -0.5770 -0.1542  0.5245  3.4578
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.2646     0.2385   9.494  5.4e-15 ***
## RaceAsian    -0.3372     0.3373  -1.000 0.320281
## RaceBlack    -0.3370     0.3373  -0.999 0.320596
## RaceLatinx   -1.1742     0.3373  -3.481 0.000792 ***
## RaceWhite     0.6109     0.3373   1.811 0.073688 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.012 on 85 degrees of freedom
## Multiple R-squared:  0.257,  Adjusted R-squared:  0.222
## F-statistic:  7.35 on 4 and 85 DF,  p-value: 3.92e-05
```

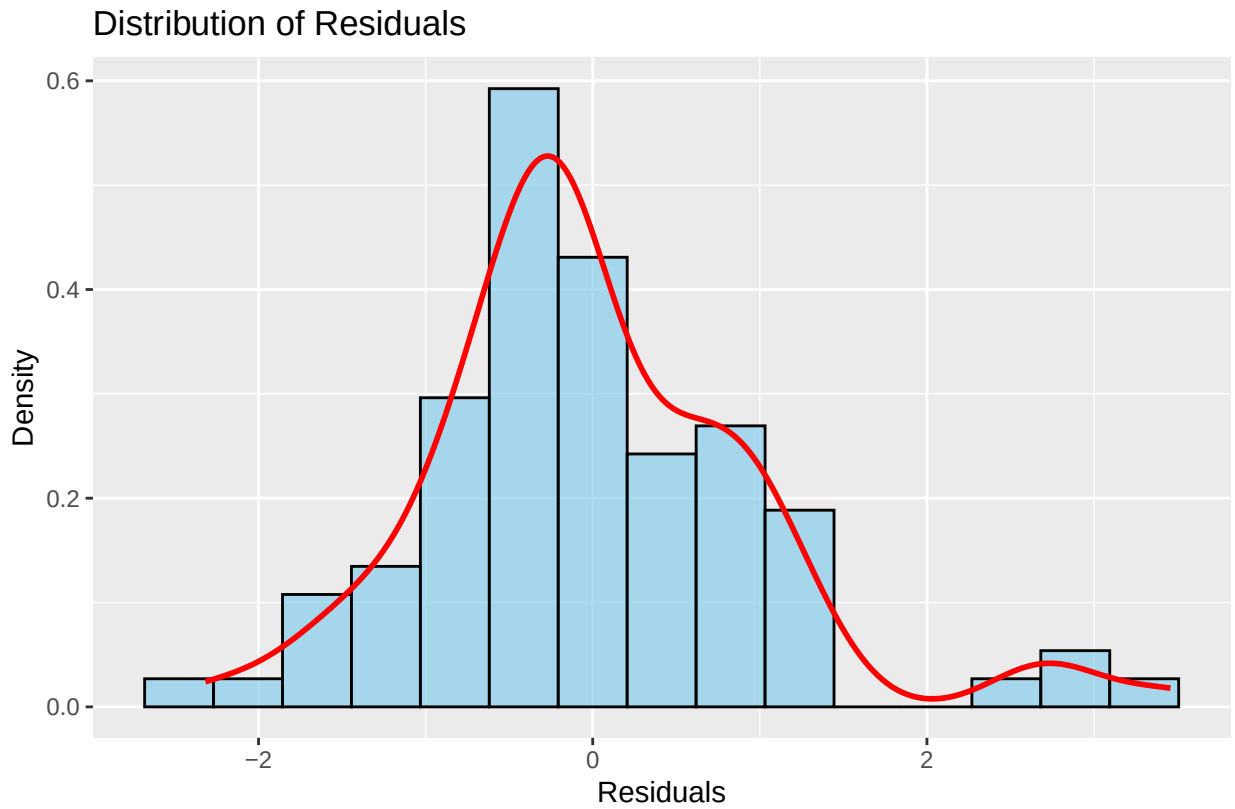
```r
# Load required libraries
library(ggplot2)
library(dplyr)

# Residual Analysis
residuals <- residuals(model)  # Get residuals from the model
fitted_values <- fitted(model)  # Get fitted values from the model

# Plot residual distribution
ggplot(data.frame(residuals), aes(x = residuals)) +
  geom_histogram(aes(y = ..density..), bins = 15, fill = "skyblue", color = "black", alpha = 0.7) +
  geom_density(color = "red", size = 1) +
  labs(title = "Distribution of Residuals", x = "Residuals", y = "Density", caption = "The histogram sh
```

```
## 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.
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



ıodel. The residuals appear approximately normally distributed, supporting the assumption of normality in the model.

```
  theme_minimal()
```

```
## List of 97
##  $ line                        :List of 6
##   ..$ colour       : chr "black"
##   ..$ linewidth    : num 0.5
##   ..$ linetype     : num 1
##   ..$ lineend      : chr "butt"
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ rect                        :List of 5
##   ..$ fill         : chr "white"
##   ..$ colour       : chr "black"
##   ..$ linewidth    : num 0.5
##   ..$ linetype     : num 1
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ text                        :List of 11
##   ..$ family       : chr ""
##   ..$ face         : chr "plain"
##   ..$ colour       : chr "black"
##   ..$ size         : num 11
##   ..$ hjust        : num 0.5
##   ..$ vjust        : num 0.5
```

```
##   ..$ angle        : num 0
##   ..$ lineheight   : num 0.9
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ title                     : NULL
##  $ aspect.ratio              : NULL
##  $ axis.title                : NULL
##  $ axis.title.x              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 2.75points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x.top          :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 2.75points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x.bottom       : NULL
##  $ axis.title.y              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : num 90
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 2.75points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.y.left         : NULL
```

```
##  $ axis.title.y.right        :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : num -90
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 2.75points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text                 :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : chr "grey30"
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x               :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 2.2points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x.top           :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 2.2points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
```

```
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x.bottom        : NULL
##  $ axis.text.y               :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 1
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 2.2points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.y.left          : NULL
##  $ axis.text.y.right         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 2.2points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.ticks                : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ axis.ticks.x              : NULL
##  $ axis.ticks.x.top          : NULL
##  $ axis.ticks.x.bottom       : NULL
##  $ axis.ticks.y              : NULL
##  $ axis.ticks.y.left         : NULL
##  $ axis.ticks.y.right        : NULL
##  $ axis.ticks.length         : 'simpleUnit' num 2.75points
##   ..- attr(*, "unit")= int 8
##  $ axis.ticks.length.x       : NULL
##  $ axis.ticks.length.x.top   : NULL
##  $ axis.ticks.length.x.bottom: NULL
##  $ axis.ticks.length.y       : NULL
##  $ axis.ticks.length.y.left  : NULL
##  $ axis.ticks.length.y.right : NULL
##  $ axis.line                 : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ axis.line.x               : NULL
##  $ axis.line.x.top           : NULL
##  $ axis.line.x.bottom        : NULL
##  $ axis.line.y               : NULL
##  $ axis.line.y.left          : NULL
```

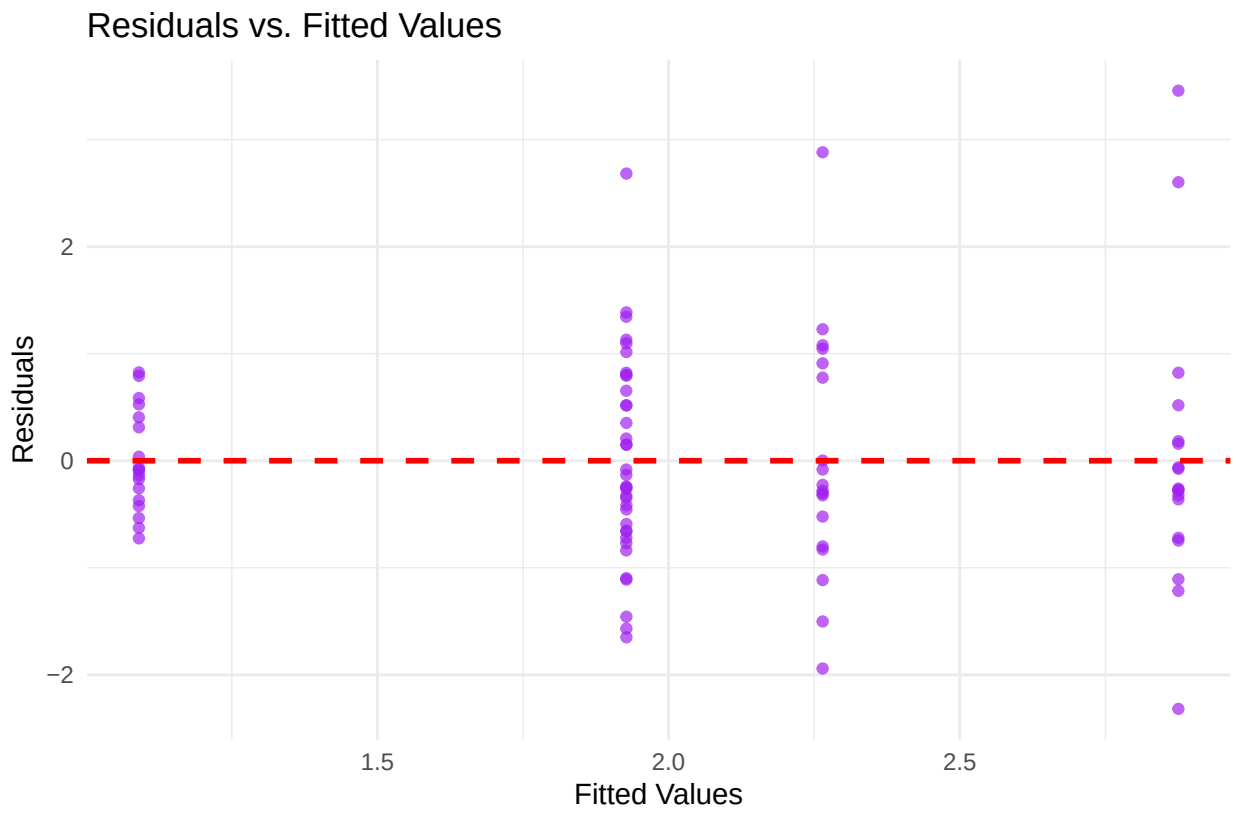
```
##  $ axis.line.y.right         : NULL
##  $ legend.background         : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ legend.margin             : 'margin' num [1:4] 5.5points 5.5points 5.5points 5.5points
##   ..- attr(*, "unit")= int 8
##  $ legend.spacing            : 'simpleUnit' num 11points
##   ..- attr(*, "unit")= int 8
##  $ legend.spacing.x          : NULL
##  $ legend.spacing.y          : NULL
##  $ legend.key                : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ legend.key.size           : 'simpleUnit' num 1.2lines
##   ..- attr(*, "unit")= int 3
##  $ legend.key.height         : NULL
##  $ legend.key.width          : NULL
##  $ legend.text               :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.text.align         : NULL
##  $ legend.title              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.title.align        : NULL
##  $ legend.position           : chr "right"
##  $ legend.direction          : NULL
##  $ legend.justification      : chr "center"
##  $ legend.box                : NULL
##  $ legend.box.just           : NULL
##  $ legend.box.margin         : 'margin' num [1:4] 0cm 0cm 0cm 0cm
##   ..- attr(*, "unit")= int 1
##  $ legend.box.background     : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ legend.box.spacing        : 'simpleUnit' num 11points
##   ..- attr(*, "unit")= int 8
```

```
##  $ panel.background          : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ panel.border              : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ panel.spacing             : 'simpleUnit' num 5.5points
##   ..- attr(*, "unit")= int 8
##  $ panel.spacing.x           : NULL
##  $ panel.spacing.y           : NULL
##  $ panel.grid                :List of 6
##   ..$ colour       : chr "grey92"
##   ..$ linewidth    : NULL
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ panel.grid.major          : NULL
##  $ panel.grid.minor          :List of 6
##   ..$ colour       : NULL
##   ..$ linewidth    : 'rel' num 0.5
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ panel.grid.major.x        : NULL
##  $ panel.grid.major.y        : NULL
##  $ panel.grid.minor.x        : NULL
##  $ panel.grid.minor.y        : NULL
##  $ panel.ontop               : logi FALSE
##  $ plot.background           : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ plot.title                :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 1.2
##   ..$ hjust        : num 0
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 5.5points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.title.position       : chr "panel"
##  $ plot.subtitle             :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : num 1
```

```
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 5.5points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.caption              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : num 1
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 5.5points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.caption.position     : chr "panel"
##  $ plot.tag                  :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 1.2
##   ..$ hjust        : num 0.5
##   ..$ vjust        : num 0.5
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.tag.position         : chr "topleft"
##  $ plot.margin               : 'margin' num [1:4] 5.5points 5.5points 5.5points 5.5points
##   ..- attr(*, "unit")= int 8
##  $ strip.background          : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ strip.background.x        : NULL
##  $ strip.background.y        : NULL
##  $ strip.clip                : chr "inherit"
##  $ strip.placement           : chr "inside"
##  $ strip.text                :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : chr "grey10"
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 4.4points 4.4points 4.4points 4.4points
```

```
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.text.x              : NULL
##  $ strip.text.x.bottom       : NULL
##  $ strip.text.x.top          : NULL
##  $ strip.text.y              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : num -90
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.text.y.left         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : num 90
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.text.y.right        : NULL
##  $ strip.switch.pad.grid     : 'simpleUnit' num 2.75points
##   ..- attr(*, "unit")= int 8
##  $ strip.switch.pad.wrap     : 'simpleUnit' num 2.75points
##   ..- attr(*, "unit")= int 8
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi TRUE
##  - attr(*, "validate")= logi TRUE
```

```r
# Plot residuals vs. fitted values
ggplot(data.frame(fitted_values, residuals), aes(x = fitted_values, y = residuals)) +
  geom_point(alpha = 0.7, color = "purple") +
  geom_hline(yintercept = 0, color = "red", linetype = "dashed", size = 1) +
  labs(title = "Residuals vs. Fitted Values", x = "Fitted Values", y = "Residuals", caption = "The scat
  ) +
  theme_minimal()
```



ear pattern, suggesting that the model satisfies the assumption of homoscedasticity (constant variance of residuals).

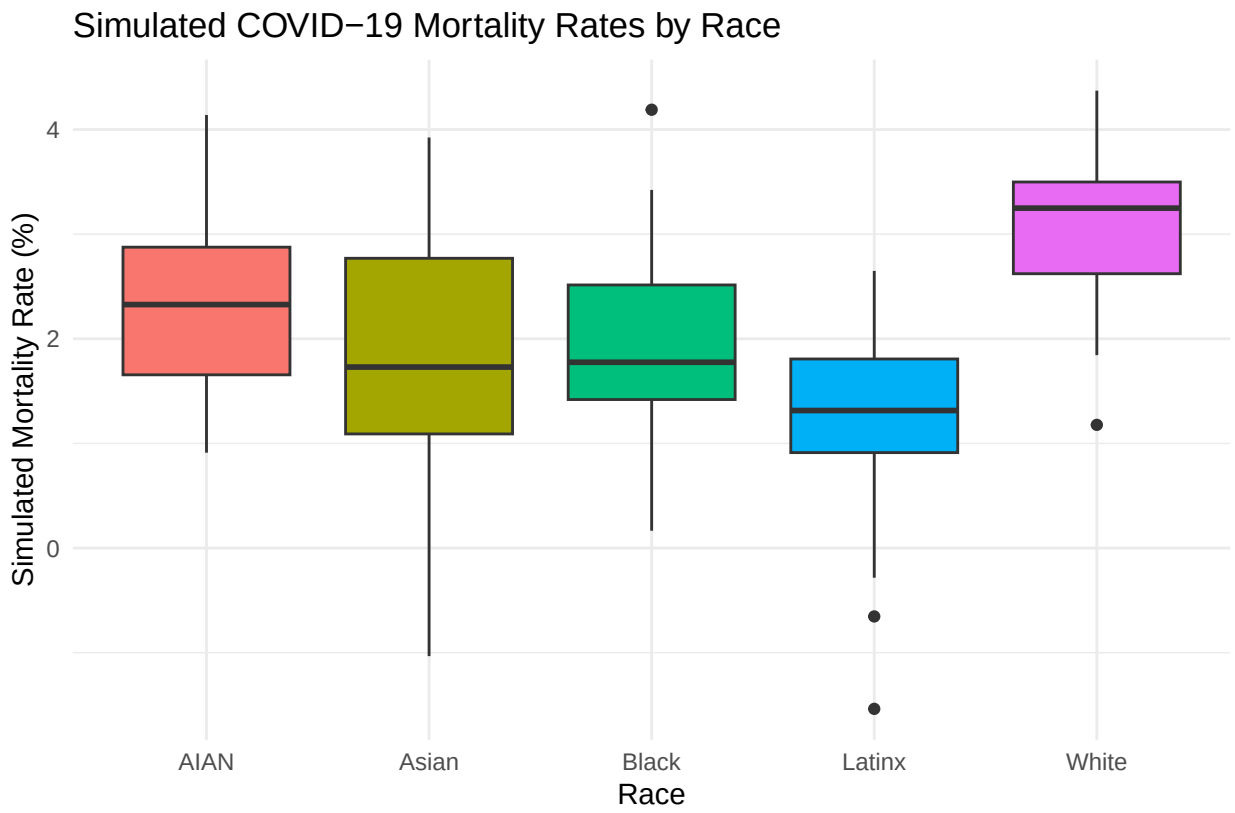
```
# Simulate data using the model's parameters and random noise
# Extract model coefficients
coefficients <- coef(model)
intercept <- coefficients["(Intercept)"]




set.seed(42)
simulated_data <- melted_data %>%
  mutate(
    Simulated_Mortality_Rate = intercept +
      rnorm(n(), mean = 0, sd = sd(residuals)) +
      sapply(Race, function(x) ifelse(is.na(coefficients[paste0("Race", x)]), 0, coefficients[paste0("Ra
  )


ggplot(simulated_data, aes(x = Race, y = Simulated_Mortality_Rate, fill = Race)) +
  geom_boxplot() +
  labs(
    title = "Simulated COVID-19 Mortality Rates by Race",
    x = "Race", y = "Simulated Mortality Rate (%)", caption = "The boxplot illustrates simulated COVID-

  ) +
  theme_minimal() +
  theme(legend.position = "none")
```



eproduces similar patterns and variability observed in the actual data, supporting the validity of the model's structure.

```
# Baseline prediction: Mean mortality rate across all races
baseline_prediction <- mean(melted_data$Mortality_Rate)

# Calculate MSE and MAE for baseline
baseline_mse <- mean((melted_data$Mortality_Rate - baseline_prediction)^2)
baseline_mae <- mean(abs(melted_data$Mortality_Rate - baseline_prediction))

# Calculate MSE and MAE for the linear regression model
model_mse <- mean((melted_data$Mortality_Rate - predict(model))^2)
model_mae <- mean(abs(melted_data$Mortality_Rate - predict(model)))

# Baseline comparison results
baseline_comparison <- data.frame(
  Metric = c("Baseline MSE", "Model MSE", "Baseline MAE", "Model MAE"),
  Value = c(baseline_mse, model_mse, baseline_mae, model_mae)
)

print("Baseline vs. Model Metrics")
```

```
## [1] "Baseline vs. Model Metrics"
```

```
print(baseline_comparison)
```

```
##         Metric     Value
## 1 Baseline MSE 1.3018602
```

```
## 2    Model MSE 0.9672783
## 3 Baseline MAE 0.8789645
## 4    Model MAE 0.7284141
```

```
# Simulating spatial correlation
set.seed(42)
unique_states <- unique(melted_data$State)
spatial_effects <- data.frame(
  State = unique_states,
  Spatial_Effect = runif(length(unique_states), min = -0.5, max = 0.5)
)


melted_data <- melted_data %>%
  left_join(spatial_effects, by = "State") %>%
  mutate(Simulated_Mortality_Rate_Spatial = Mortality_Rate + Spatial_Effect)


simulated_model <- lm(Simulated_Mortality_Rate_Spatial ~ Race, data = melted_data)
simulated_model_summary <- summary(simulated_model)


print("Simulated Model Summary")
```

```
## [1] "Simulated Model Summary"
```

```
print(simulated_model_summary)
```

```
##
## Call:
## lm(formula = Simulated_Mortality_Rate_Spatial ~ Race, data = melted_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4795 -0.6840 -0.0469  0.6557  3.6645
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.3884     0.2698   8.854 1.07e-13 ***
## RaceAsian    -0.3372     0.3815  -0.884  0.37919
## RaceBlack    -0.3370     0.3815  -0.883  0.37950
## RaceLatinx   -1.1742     0.3815  -3.078  0.00281 **
## RaceWhite     0.6109     0.3815   1.601  0.11303
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.145 on 85 degrees of freedom
## Multiple R-squared:  0.2129, Adjusted R-squared:  0.1758
## F-statistic: 5.747 on 4 and 85 DF,  p-value: 0.0003836
```

```
library(dplyr)
library(broom)
```

```r
# Define a function to perform bootstrapping
bootstrap_model <- function(data, n_iterations = 1000) {
  coefficients <- list()

  for (i in 1:n_iterations) {
    # Resample the data with replacement
    boot_data <- data[sample(nrow(data), replace = TRUE), ]
    # Fit the model
    boot_model <- lm(Mortality_Rate ~ Race, data = boot_data)
    # Store the coefficients
    coefficients[[i]] <- coef(boot_model)
  }

  # Convert the list of coefficients to a data frame
  coef_df <- do.call(rbind, coefficients)
  colnames(coef_df) <- names(coefficients[[1]])
  return(as.data.frame(coef_df))
}

# Perform bootstrapping
set.seed(42)
n_iterations <- 1000
boot_results <- bootstrap_model(melted_data, n_iterations)

# Summarize the bootstrapped coefficients
boot_summary <- boot_results %>%
  summarise(across(everything(), list(
    mean = mean,
    sd = sd,
    `2.5%` = ~ quantile(., 0.025),
    `97.5%` = ~ quantile(., 0.975)
  )))

# Print the summary of bootstrap results
print("Bootstrap Summary:")
```

```
## [1] "Bootstrap Summary:"
```

```r
print(boot_summary)
```

```
##   (Intercept)_mean (Intercept)_sd (Intercept)_2.5% (Intercept)_97.5%
## 1         2.268754      0.2792075          1.76864          2.856232
##   RaceAsian_mean RaceAsian_sd RaceAsian_2.5% RaceAsian_97.5% RaceBlack_mean
## 1     -0.3381237    0.3371661      -1.017823       0.2865351       -0.35137
##   RaceBlack_sd RaceBlack_2.5% RaceBlack_97.5% RaceLatinx_mean RaceLatinx_sd
## 1    0.3759984      -1.079586       0.3878012       -1.176458     0.3042226
##   RaceLatinx_2.5% RaceLatinx_97.5% RaceWhite_mean RaceWhite_sd RaceWhite_2.5%
## 1       -1.813754       -0.6084173      0.6137887    0.4283897     -0.1872425
##   RaceWhite_97.5%
## 1        1.470885
```

```r
# Interpretation of statistical significance
# If the confidence interval (2.5%, 97.5%) for a coefficient does not include zero,
# it is statistically significant at the 95% confidence level.

# Combine the bootstrap summary into a single tidy dataframe
boot_summary_tidy <- boot_summary %>%
  pivot_longer(
    cols = everything(),
    names_to = c("Coefficient", ".value"),
    names_sep = "_"
  ) %>%
  rename(
    Mean = mean,
    SD = sd,
    `Lower CI (2.5%)` = `2.5%`,
    `Upper CI (97.5%)` = `97.5%`
  )


boot_summary_tidy
```

```
## # A tibble: 5 x 5
##   Coefficient   Mean    SD `Lower CI (2.5%)` `Upper CI (97.5%)`
##   <chr>        <dbl> <dbl>             <dbl>              <dbl>
## 1 (Intercept)  2.27  0.279             1.77               2.86
## 2 RaceAsian   -0.338 0.337            -1.02               0.287
## 3 RaceBlack   -0.351 0.376            -1.08               0.388
## 4 RaceLatinx  -1.18  0.304            -1.81              -0.608
## 5 RaceWhite    0.614 0.428            -0.187              1.47
```