```r
# Set working directory to project root
setwd("C:/Users/mohi2/OneDrive/Documents/GIT Projects/E-Cigarette_Regulations_Analysis/")

# Load Dataset from the Same Directory as the Code
file_name <- "CDC_STATE_System_E-Cigarette_Legislation.csv"
file_path <- paste0(getwd(), "/data/", file_name)
data <- read.csv(file_path)

# Data Preprocessing
data <- data %>%
  mutate(
    Effective_Date = as.Date(Effective_Date, format = "%m/%d/%Y"),
    Year_Quarter = paste(YEAR, "Q", Quarter, sep = "")
  ) %>%
  filter(!is.na(Effective_Date))
```

```r
# Calculate Stringency Score Based on Policy Provisions
data <- data %>%
  mutate(stringency_score = case_when(
    ProvisionGroupDesc == "Enforcement" & ProvisionDesc == "Enforcement (Type)" ~
      case_when(
        grepl("police|sheriff|peace officer", tolower(ProvisionValue)) ~ 7,
        grepl("alcoholic beverage|liquor", tolower(ProvisionValue)) ~ 6,
        grepl("department|commission", tolower(ProvisionValue)) ~ 5,
        grepl("local|mayor", tolower(ProvisionValue)) ~ 3,
        ProvisionValue == "Yes" ~ 2,
        ProvisionValue == "No Provision" ~ 1,
        TRUE ~ 1
      ),
    ProvisionGroupDesc == "Penalties" & ProvisionDesc == "License Suspension or Revocation" ~
      case_when(
        ProvisionValue == "Both" ~ 6,
        ProvisionValue == "Revocation" ~ 4,
        ProvisionValue == "Suspension" ~ 3,
        TRUE ~ 1
      ),
    ProvisionGroupDesc == "Restrictions" & ProvisionDesc %in% c("Possession Prohibited", "Purcha
se Prohibited") & ProvisionValue == "Yes" ~ 8,
    ProvisionGroupDesc == "Restrictions" & ProvisionDesc == "Use Prohibited" & ProvisionValue ==
"Yes" ~ 7,
    ProvisionGroupDesc == "Restrictions" & ProvisionDesc %in% c("Banned from Location", "Restric
tion on Access") & ProvisionValue == "Yes" ~ 6,
    ProvisionGroupDesc == "Restrictions" & ProvisionDesc == "Minimum Age (Years)" ~
      case_when(
        as.numeric(ProvisionValue) == 21 ~ 5,
        as.numeric(ProvisionValue) == 20 ~ 4,
        as.numeric(ProvisionValue) == 19 ~ 3,
        as.numeric(ProvisionValue) == 18 ~ 2,
        TRUE ~ 1
      ),
    TRUE ~ 1
  ))
```

```
## Warning: There were 4 warnings in `mutate()`.
## The first warning was:
## ℹ In argument: `stringency_score = case_when(...)`.
## Caused by warning:
## ! NAs introduced by coercion
## ℹ Run `dplyr::last_dplyr_warnings()` to see the 3 remaining warnings.
```

```
# Aggregate Stringency Scores by State and Quarter
aggregated_data <- data %>%
  group_by(LocationDesc, Year_Quarter) %>%
  summarise(Total_Stringency = sum(stringency_score, na.rm = TRUE), .groups = "drop") %>%
  arrange(LocationDesc, Year_Quarter)
```

```
# ARIMA Forecasting for Each State
state_results <- list()
future_stringency <- data.frame(State = character(), Future_Stringency = numeric())
metrics_table <- data.frame(State = character(), MAPE = numeric(), RMSE = numeric())

for (state in unique(aggregated_data$LocationDesc)) {
  state_data <- aggregated_data %>% filter(LocationDesc == state)
  if (nrow(state_data) > 5) {
    ts_data <- ts(state_data$Total_Stringency, frequency = 4)
    model <- auto.arima(ts_data)
    forecasted <- forecast(model, h = 4)
    state_results[[state]] <- list(model = model, forecast = forecasted)
    future_stringency <- rbind(
      future_stringency,
      data.frame(
        State = state,
        Future_Stringency = mean(forecasted$mean)
      )
    )

    # Collect Error Metrics
    mape <- mean(abs(residuals(model) / ts_data) * 100)
    metrics_table <- rbind(
      metrics_table,
      data.frame(
        State = state,
        MAPE = mape,
        RMSE = sqrt(mean(residuals(model)^2))
      )
    )
  }
}

# Print Metrics Table
knitr::kable(metrics_table, caption = "ARIMA Model Metrics for Each State")
```

ARIMA Model Metrics for Each State

State	MAPE	RMSE
Alabama	0.9668572	2.7701885
Alaska	1.7001132	6.3639611
Arizona	0.0000000	0.0000000
Arkansas	0.8260275	1.0554952
California	2.0393417	3.4963036
Colorado	0.8251871	2.2724746
Connecticut	0.8434819	1.5617404
Delaware	0.2002474	0.6100439
District of Columbia	0.0000000	0.0000000
Florida	0.1276942	0.4629850
Georgia	0.1647186	0.4629533
Guam	0.1113372	0.4575903

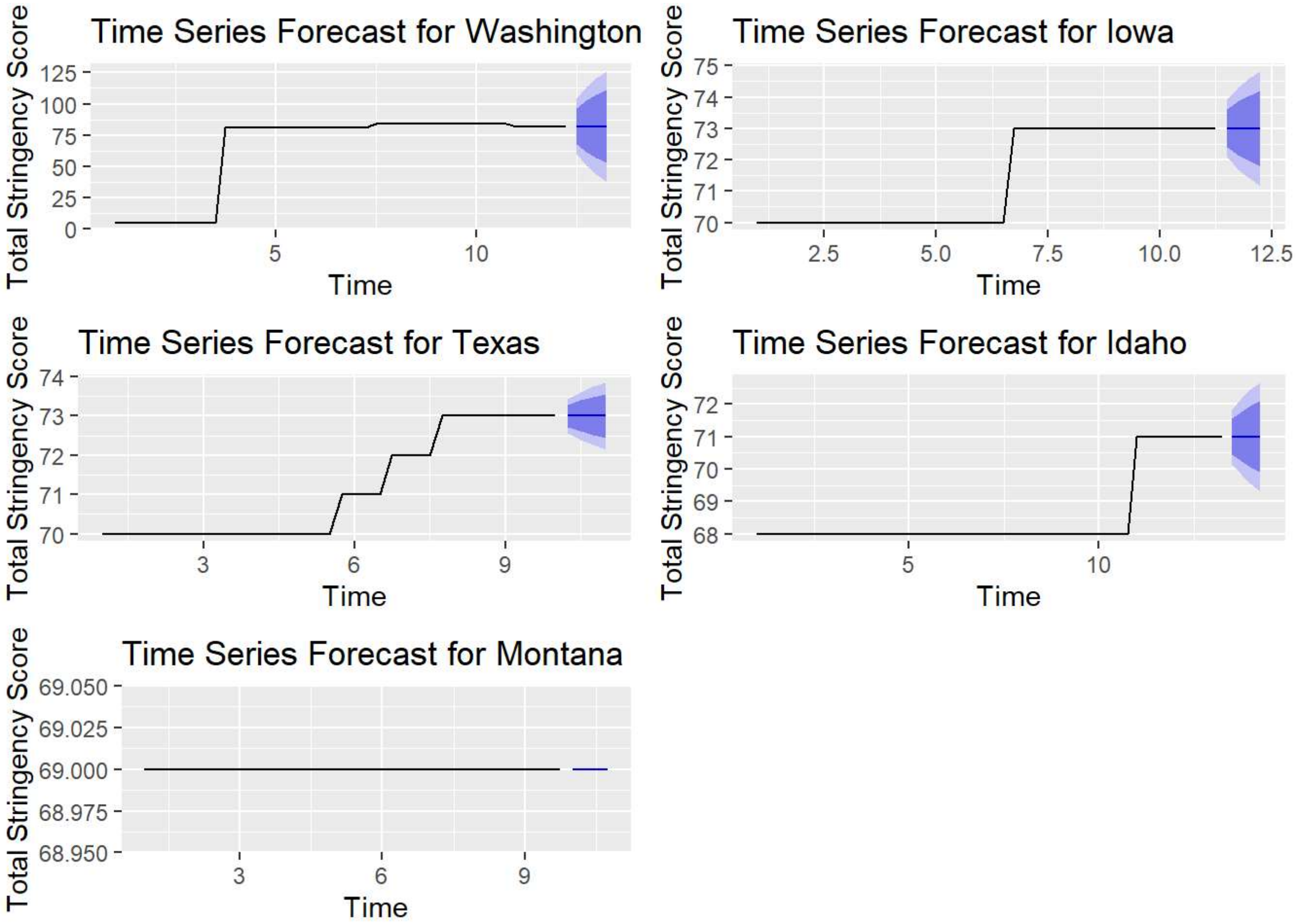
State	MAPE	RMSE
Hawaii	1.2237271	3.1555455
Idaho	0.0865070	0.4243730
Illinois	1.4814918	1.9306159
Indiana	0.1056936	0.4424143
Iowa	0.1002283	0.4630360
Kansas	0.2535039	0.4430088
Kentucky	0.0410853	0.1527820
Louisiana	0.2063239	0.7625515
Maine	0.1571788	0.2410835
Maryland	1.8636516	2.5924963
Massachusetts	0.3750091	0.2000232
Michigan	0.2356703	0.6397136
Minnesota	0.9301061	1.8382907
Mississippi	0.6664251	1.6218657
Missouri	0.0000000	0.0000000
Montana	0.0000000	0.0000000
Nebraska	0.1467277	0.3410693
Nevada	0.1716216	0.4932524
New Hampshire	0.1159402	0.3945932
New Jersey	0.0774242	0.2582558
New Mexico	0.1773893	0.4804293
New York	0.1379529	0.4330572
North Carolina	0.0000000	0.0000000
North Dakota	0.1515889	0.4867319
Northern Mariana Islands	0.2164530	0.8333933
Ohio	0.1371518	0.4629743
Oklahoma	0.2084945	0.5631629
Oregon	1.0986312	2.6067013
Palau	1.9738482	3.8680150
Pennsylvania	20.5663295	3.9513876
Puerto Rico	0.0000000	0.0000000
Rhode Island	0.9955276	3.5790446
South Carolina	0.0394550	0.1460935
South Dakota	0.1665846	0.6172696
Tennessee	1.2392644	2.2152675
Texas	0.0925473	0.2132283
Utah	0.2119543	0.8234216
Vermont	1.6282675	6.2874415
Virginia	2.4978022	4.5877708

State	MAPE	RMSE
Washington	2.1725568	11.2181918
West Virginia	0.8882613	2.4399932
Wisconsin	0.0000000	0.0000000
Wyoming	0.8633595	3.4671115

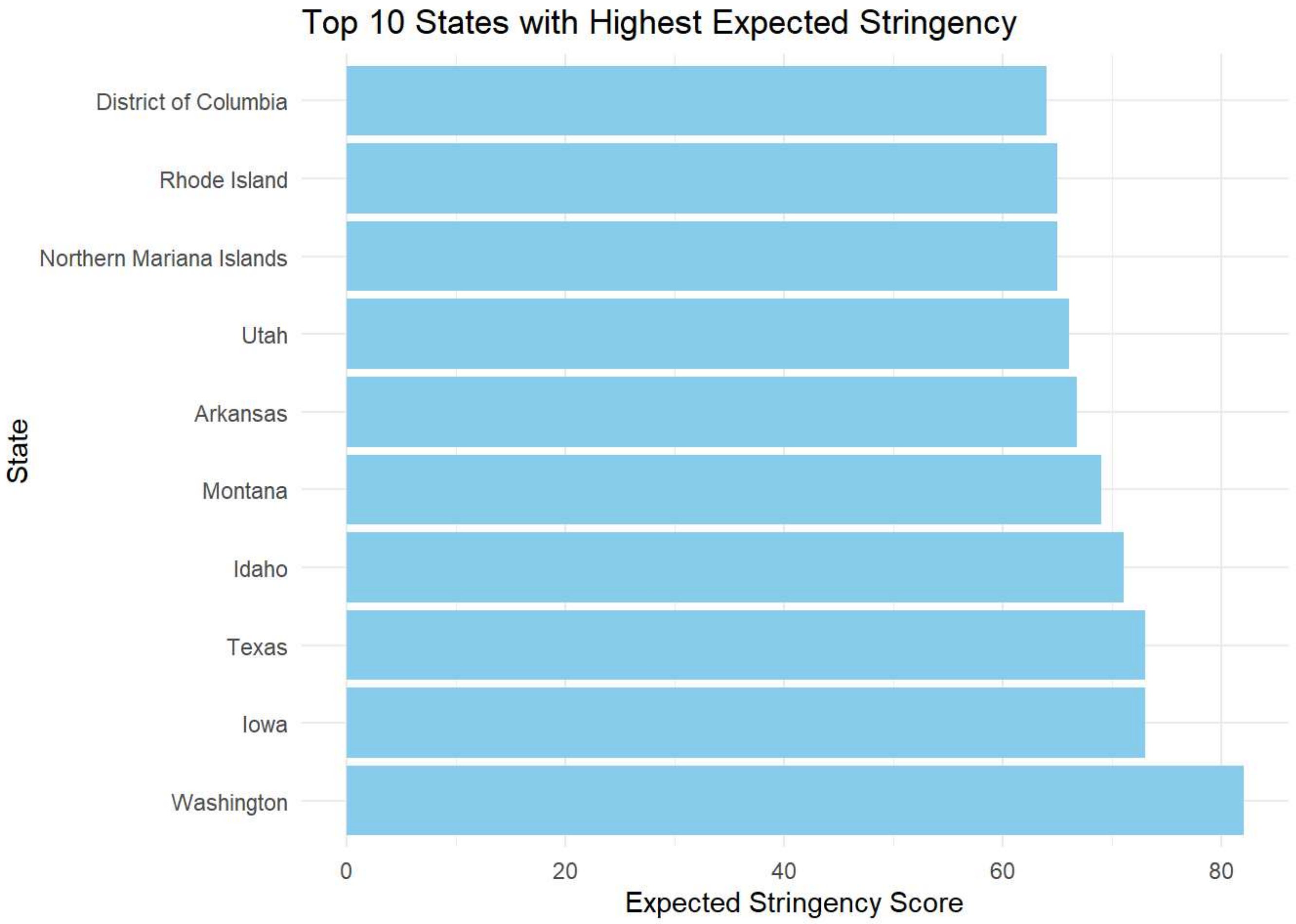
```r
# Plot Forecast for Top 5 States with Highest Future Stringency
top_10_states <- future_stringency %>%
  arrange(desc(Future_Stringency)) %>%
  head(10)

top_5_states <- head(top_10_states, 5)
forecast_plots <- list()
for (state in top_5_states$State) {
  forecast_result <- state_results[[state]]
  forecast_plots[[state]] <- autoplot(forecast_result$forecast) +
    ggtitle(paste("Time Series Forecast for", state)) +
    xlab("Time") +
    ylab("Total Stringency Score")
}

gridExtra::grid.arrange(grobs = forecast_plots, ncol = 2)
```



```r
# Bar Plot of Top 10 States with Highest Expected Stringency
ggplot(top_10_states, aes(x = reorder(State, -Future_Stringency), y = Future_Stringency)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  theme_minimal() +
  labs(
    title = "Top 10 States with Highest Expected Stringency",
    x = "State",
    y = "Expected Stringency Score"
  ) +
  coord_flip()
```


Top 10 States with Highest Expected Stringency
District of Columbia
Rhode Island
Northern Mariana Islands
Utah
Arkansas
Montana
Idaho
Texas
Iowa
Washington
State
0
20
40
60
80
Expected Stringency Score