stat-321_project

kush manek

2025-04-03

```
library(readr)
library(dplyr)
library(ggplot2)
library(tidyverse)
library(lubridate)
```

```
library(quantmod)
getSymbols("VGT", src = "yahoo", from = "2015-01-01", periodicity = "monthly")
VGT_rounded <- round(VGT, 2)
head(VGT_rounded)
```

```
stock_data <- data.frame(Date = index(VGT_rounded), coredata(VGT_rounded))

# Save as CSV
write.csv(stock_data, "VGT_stock_data.csv", row.names = FALSE)

print("File saved as VGT_stock_data.csv")
```

```
getSymbols("VHT", src = "yahoo", from = "2015-01-01", periodicity = "monthly")
VHT_rounded <- round(VHT, 2)
head(VHT_rounded)
```

```
stock_data <- data.frame(Date = index(VHT_rounded), coredata(VHT_rounded))

# Save as CSV
write.csv(stock_data, "VHT_stock_data.csv", row.names = FALSE)

print("File saved as VHT_stock_data.csv")
```

| For VGT data |

```
library(quantmod)
getSymbols("VGT", src = "yahoo", from = "2015-01-01", periodicity = "monthly")
VGT_rounded <- round(VGT, 2)
```

```
stock_data <- data.frame(Date = index(VGT_rounded), coredata(VGT_rounded))

# Save as CSV
write.csv(stock_data, "VGT_stock_data.csv", row.names = FALSE)

print("File saved as VGT_stock_data.csv")
```

```
VGT <- read_csv("VGT_stock_data.csv");
VGT
```

```
VGT$average_movement <- (VGT$VGT.High+VGT$VGT.Low)/2
VGT
colnames(VGT)
VGT$VGT.Open <- NULL
VGT$VGT.Close <- NULL
VGT$VGT.Adjusted <- NULL
VGT$VGT.Volume <- NULL
VGT_rounded <- VGT %>%
  mutate(across(where(is.numeric), ~ round(.x, 2)))
write_csv(VGT_rounded,"VGT_stock_data_cleaned.csv")
```

```
getSymbols("VHT", src = "yahoo", from = "2015-01-01", periodicity = "monthly")
VHT_rounded <- round(VHT, 2)
```

```
stock_data <- data.frame(Date = index(VHT_rounded), coredata(VHT_rounded))

# Save as CSV
write.csv(stock_data, "VHT_stock_data.csv", row.names = FALSE)

print("File saved as VHT_stock_data.csv")
```

| For VHT data |

```
VHT <- read_csv("VHT_stock_data.csv");

VHT$average_movement <- (VHT$VHT.High+VHT$VHT.Low)/2
VHT

VHT$VHT.Open <- NULL
VHT$VHT.Close <- NULL
VHT$VHT.Adjusted <- NULL
VHT$VHT.Volume <- NULL
VHT_rounded <- VHT %>%
  mutate(across(where(is.numeric), ~ round(.x, 2)))
write_csv(VHT_rounded,"VHT_stock_data_cleaned.csv")
```

| For unemloyment data |

```
unemployment_data <- read_csv("unemployment rate data.csv");
unemployment_data
```

we will not cap the data as the outliers are important to show the unemployment rate during the covid period.

|ICSA data|

```
ICSA_data <- read_csv("ICSA_Initial claims.csv")
ICSA_data
```

```
ICSA_data <- ICSA_data[-c(1:2505), ]
ICSA_data
ICSA_data <- ICSA_data %>%
  mutate(observation_date = as.Date(observation_date, format = case_when(
  grepl("/", observation_date) ~ "%m/%d/%Y",
  grepl("-", observation_date) ~ "%m-%d-%Y",
  grepl(",", observation_date) ~ "%B %d, %Y",
  TRUE ~ NA_character_
  )))
ICSA_data
write_csv(ICSA_data,"ICSA_Initial claims_cleaned.csv")
```

removed the previous data as we are only taking data from last 10 years into consideration for the analysis.

|Weekly_United_States_COVID-19|

```
covid_data <- read_csv("Weekly_United_States_COVID-19_Cases_and_Deaths_by_State_-_ARCHIVED_20250304.csv
covid_data
covid_data <- covid_data %>%
  mutate(date_updated = as.Date(date_updated, format = case_when(
  grepl("/", date_updated) ~ "%m/%d/%Y",
  grepl("-", date_updated) ~ "%m-%d-%Y",
  grepl(",", date_updated) ~ "%B %d, %Y",
  TRUE ~ NA_character_
  )))

covid_data <- covid_data %>%
  mutate(start_date = as.Date(start_date, format = case_when(
  grepl("/", start_date) ~ "%m/%d/%Y",
  grepl("-", start_date) ~ "%m-%d-%Y",
  grepl(",", start_date) ~ "%B %d, %Y",
  TRUE ~ NA_character_
  )))


covid_data <- covid_data %>%
  mutate(end_date = as.Date(end_date, format = case_when(
  grepl("/", end_date) ~ "%m/%d/%Y",
  grepl("-", end_date) ~ "%m-%d-%Y",
  grepl(",", end_date) ~ "%B %d, %Y",
  TRUE ~ NA_character_
  )))
covid_data
```

```r
covid_data$state <- as.factor(covid_data$state)
covid_data
```

```r
covid_data$new_historic_cases <- NULL
covid_data$new_historic_deaths <- NULL
covid_data
```

```r
covid_data <- covid_data %>%
  mutate(
    circuit = case_when(
      state %in% c("ME", "MA", "NH", "PR", "RI") ~ "Region 1",
      state %in% c("CT", "NY", "VT", "NYC") ~ "Region 2",
      state %in% c("DE", "NJ", "PA", "VI") ~ "Region 3",
      state %in% c("MD", "NC", "SC", "VA", "WV", "AL", "FL", "GA", "DC") ~ "Region 4",
      state %in% c("LA", "MS", "TX","AS") ~ "Region 5",
      state %in% c("KY", "MI", "OH", "TN") ~ "Region 6",
      state %in% c("IL", "IN", "WI") ~ "Region 7",
      state %in% c("AR", "IA", "MN", "MO", "NE", "ND", "SD") ~ "Region 8",
      state %in% c("AK", "AZ", "CA", "GU", "HI", "ID", "MT", "NV", "OR", "WA","FSM", "MP", "PW", "RMI")
      state %in% c("CO", "KS", "NM", "OK", "UT", "WY") ~ "Region 10"
    )
  )

covid_data$jurisdiction_type <- NULL

covid_data <- covid_data %>%
  rename("Jurisidiction" = circuit)

# View the updated dataset
head(covid_data)

write_csv(covid_data,"Weekly_United_States_COVID-19_Cases_and_Deaths_by_State_-_ARCHIVED_20250304_clean
```

```r
covid_19_data <- read_csv("COVID-19_Weekly_Cases_and_Deaths_by_Age__Race_Ethnicity__and_Sex_-_ARCHIVED_
covid_19_data
```

```r
library(tidyr)

covid_19_data <- covid_19_data %>%
  mutate(
    death_count_suppressed = replace_na(death_count_suppressed, 0),
    death_crude_rate_suppressed_per_100k = replace_na(death_crude_rate_suppressed_per_100k, 0)
  )

covid_19_data <- covid_19_data %>%
  mutate(end_of_week = as.Date(end_of_week, format = case_when(
  grepl("/", end_of_week) ~ "%m/%d/%Y",
  grepl("-", end_of_week) ~ "%m-%d-%Y",
  grepl(",", end_of_week) ~ "%B %d, %Y",
  TRUE ~ NA_character_
  )))

covid_19_data
```

```
write.csv(covid_19_data,"COVID-19_Weekly_Cases_and_Deaths_by_Age__Race_Ethnicity__and_Sex_-_ARCHIVED_20
```

| Forecast of the VGT and VHT dataset|

```
library(forecast)
library(tidyverse)

VGT$Date <- as.Date(VGT$Date)
VHT$Date <- as.Date(VHT$Date)

# Create time series objects
vgt_ts <- ts(VGT$average_movement, start = c(year(min(VGT$Date)), month(min(VGT$Date))), frequency = 12)
vht_ts <- ts(VHT$average_movement, start = c(year(min(VHT$Date)), month(min(VHT$Date))), frequency = 12)

# Fit ARIMA models
vgt_arima <- auto.arima(vgt_ts)
vht_arima <- auto.arima(vht_ts)

# Forecast for next 12 months
vgt_forecast <- forecast(vgt_arima, h = 12, level = c(95, 99))
vht_forecast <- forecast(vht_arima, h = 12, level = c(95, 99))

# Plot the forecasts
autoplot(vgt_forecast) + ggtitle("VGT Stock Price Forecast") + xlab("Time") + ylab("Average Movement")
autoplot(vht_forecast) + ggtitle("VHT Stock Price Forecast") + xlab("Time") + ylab("Average Movement")

# Print forecast summaries
summary(vgt_forecast)
summary(vht_forecast)
```