MiniProject #01 Analysis

Siddhi Kataria

Mini Project #01

Fiscal Characterisitcs of Major US Public Transit Systems

Author: Siddhi Kataria

#Code:

if(!require("quarto")){ install.packages("quarto") } library(quarto) if(!quarto::quarto_binary_sitrep()){ stop("Something is wrong with your quarto installation.") } quarto::quarto_render(".") system("git add docs/*") if(!any(grepl("rstudio", search()))){q("no")}

Load necessary libraries

library(tidyverse) library(readxl) # Ensure readxl is loaded for reading Excel files

Set the correct file path

file_path <- "C:/Users/kisho/Downloads/ridership.xlsx.xlsx"

Check if the file exists at the specified path

if (!file.exists(file_path)) { stop(paste("Error: The file does not exist at the specified path:", file_path)) }

Read and process the 'UPT' sheet

TRIPS <- read_xlsx(file_path, sheet = "UPT") |> filter(`Mode/Type of Service Status` == "Active") |> select(-`Legacy NTD ID`, -`Reporter Type`, -`Mode/Type of Service Status`, -`UACE CD`, -`TOS`) |> pivot_longer(-c(`NTD ID`:3 `Mode`), names_to = "month", values_to = "UPT") |> drop_na() |> mutate(month = lubridate::my(month)) # Parse month-year date specs

Read and process the 'VRM' sheet

MILES <- read_xlsx(file_path, sheet = "VRM") |> filter(`Mode/Type of Service Status` == "Active") |> select(-`Legacy NTD ID`, -`Reporter Type`, -`Mode/Type of Service Status`, -`UACE CD`, -`TOS`) |> pivot_longer(-c(`NTD ID`:3 `Mode`), names_to = "month", values_to = "VRM") |> drop_na() |> group_by(`NTD ID`, `Agency`, `UZA Name`, `Mode`, `3 Mode`, month) |> summarize(VRM = sum(VRM)) |> ungroup() |> mutate(month = lubridate::my(month)) # Parse month-year date specs

Combine TRIPS and MILES data

USAGE <- inner_join(TRIPS, MILES) |> mutate(`NTD ID` = as.integer(`NTD ID`))

Display the resulting data

print(USAGE)

if(!require("DT")) install.packages("DT") library(DT)

sample_n(USAGE, 1000) |> mutate(month=as.character(month)) |> DT::datatable()

#Task 1 USAGE <- USAGE |> rename(metro_area = `UZA Name`)

#Task 2 unique_modes <- USAGE |> distinct(Mode) print(unique_modes)

USAGE <- USAGE |> mutate(Mode = case_when(Mode == "HR" ~ "Heavy Rail", Mode == "LR" ~ "Light Rail", Mode == "MB" ~ "Motor Bus", Mode == "DR" ~ "Demand Response", Mode == "CR" ~ "Commuter Rail", Mode == "TB" ~ "Trolley Bus", TRUE ~ "Unknown"))

#Task 3: A. Transit Agency with Most Total VRM: most_total_vrm <- USAGE |> group_by(`Agency`) |> summarize(total_VRM = sum(VRM, na.rm = TRUE)) |> arrange(desc(total_VRM)) |> slice(1) print(most_total_vrm)

#Task 3: B. Transit Mode with Most Total VRM: most_mode_vrm <- USAGE |> group_by(Mode) |> summarize(total_VRM = sum(VRM, na.rm = TRUE)) |> arrange(desc(total_VRM)) |> slice(1) print(most_mode_vrm)

#Task 3: C. Trips Taken on the NYC Subway in May 2024: nyc_subway_trips <- USAGE |> filter(Mode == "Heavy Rail", month == "2024-05-01") |> summarize(total_trips = sum(UPT, na.rm = TRUE)) print(nyc_subway_trips)

#Task 3: D. Mode with Longest Average Trip in May 202 average_trip_length <- USAGE |> filter(month == "2024-05-01") |> group_by(Mode) |> summarize(total_VRM = sum(VRM, na.rm = TRUE), # Total Vehicle Revenue Miles total_trips = sum(UPT, na.rm = TRUE), # Total trips average_trip_length = total_VRM / total_trips # Average trip length estimation) |> arrange(desc(average_trip_length)) |> slice(1)

print(average_trip_length)

#Task 3: E. NYC Subway Ridership Fall Between April 2019 and April 2020 ridership_fall <- USAGE |> filter(Mode == "Heavy Rail", month %in% c("2019-04-01", "2020-04-01")) |> group_by(month) |> summarize(total_ridership = sum(UPT, na.rm = TRUE)) |> mutate(difference = total_ridership[month == "2020-04-01"] - total_ridership[month == "2019-04-01"]) print(ridership_fall)

#Task 4: Interesting fact #1 total_ridership <- USAGE |> group_by(Agency) |> summarize(total_UPT = sum(UPT, na.rm = TRUE)) |> arrange(desc(total_UPT)) |> slice(1) print(total_ridership)

#Task 4: Interesting fact #2 average_VRM_by_mode <- USAGE |> group_by(Mode) |> summarize(average_VRM = mean(VRM, na.rm = TRUE)) |> arrange(desc(average_VRM)) |> slice(1) print(average_VRM_by_mode)

#Task 4: Interesting fact #3 active_agencies_count <- USAGE |> filter(month == "2024-05-01") |> distinct(Agency) |> count() print(active_agencies_count)

#Task 5: Create Annual Totals for UPT and VRM for 2022 library(lubridate)

USAGE_2022_ANNUAL <- USAGE |> filter(year(month) == 2022) |> group_by(`NTD ID`, Agency, metro_area, Mode) |> summarize(UPT = sum(UPT, na.rm = TRUE), VRM = sum(VRM, na.rm = TRUE)) |> ungroup()

USAGE_AND_FINANCIALS <- left_join(USAGE_2022_ANNUAL, FINANCIALS, join_by(`NTD ID`, Mode)) |> drop_na()

#Task 6 I: Which transit system (agency and mode) had the most UPT in 2022? USAGE_AND_FINANCIALS |> arrange(desc(UPT)) |> slice(1) |> select(Agency, Mode, UPT)

#Task 6 II: Which transit system (agency and mode) had the highest farebox recovery, defined as the highest ratio of Total Fares to Expenses? USAGE_AND_FINANCIALS |> mutate(farebox_recovery = `Total Fares` / Expenses) |> arrange(desc(farebox_recovery)) |> slice(1)

#Task 6 III:Which transit system (agency and mode) has the lowest expenses per UPT? USAGE_AND_FINANCIALS |> mutate(expenses_per_UPT = Expenses / UPT) |> arrange(expenses_per_UPT) |> slice(1)

#Task 6 IV: Which transit system (agency and mode) has the highest total fares per UPT? USAGE_AND_FINANCIALS |> mutate(fares_per_UPT = `Total Fares` / UPT) |> arrange(desc(fares_per_UPT)) |> slice(1)

#Task 6 V: Which transit system (agency and mode) has the lowest expenses per VRM? USAGE_AND_FINANCIALS |> mutate(expenses_per_VRM = Expenses / VRM) |> arrange(expenses_per_VRM) |> slice(1)

#Task 6 VI: Which transit system (agency and mode) has the highest total fares per VRM? USAGE_AND_FINANCIALS |> mutate(fares_per_VRM = `Total Fares` / VRM) |> arrange(desc(fares_per_VRM)) |> slice(1)