Kia Thefts Week 5-6

Sam Kluthe DSC640

Audience

The primary audience for these visuals consists of police leadership and city officials across the Lower Mainland. This audience is composed of senior decision-makers who are generally familiar with crime and public safety data but may not be directly involved in day-to-day data analysis. The visuals are intended to be presented at cross-jurisdictional meetings or professional seminars, where officials from multiple police departments and municipalities convene to share information and best practices. As a result, the visualizations are designed to communicate insights clearly and efficiently, minimizing technical jargon while emphasizing actionable trends and comparative outcomes. The focus is on highlighting successful strategies employed by jurisdictions that have effectively reduced vehicle thefts, enabling informed discussion, strategic alignment, and data-driven decision-making among experienced stakeholders.

Purpose

The purpose of this data story is to move beyond describing national vehicle theft trends and instead support informed, coordinated action among police leadership and city officials. While the visuals show that vehicle thefts are increasing nationwide, they also reveal that the rate of increase is significantly slower in eastern states, despite those regions experiencing high overall theft volumes. In addition, the data highlights a distinct upward trend in Kia and Hyundai thefts, underscoring the need for targeted intervention. The intent of this story is to encourage cross-jurisdictional dialogue and knowledge sharing by examining why certain regions have been more successful at slowing growth in theft rates. The call to action is for decision-makers to evaluate and adapt proven prevention strategies from these jurisdictions, prioritize manufacturer-specific theft mitigation efforts,

and support coordinated policy, enforcement, and public-awareness initiatives aimed at reducing vehicle theft growth rather than merely responding to its consequences.

Medium

PowerPoint presentation is the most effective medium for conveying this data to the intended audience. The presentation format allows the visuals to be clearly explained and contextualized during in-person or virtual meetings with police leadership and city officials, supporting discussion and collaborative analysis. Additionally, the slide-based format enables cities and police departments that are unable to attend the presentation to review the material independently later. This approach ensures consistent messaging, allows for flexible dissemination, and supports both guided interpretation and self-paced review, making it well suited for an audience of senior decision-makers with varying levels of availability and data familiarity.

Call to Action

The data indicates that Kia and Hyundai thefts increase in the final months of the year, particularly from October through January. To address this seasonal surge, cities should implement a focused Car Theft Program that prioritizes Kia and Hyundai vehicles *before* thefts peak. By launching targeted prevention efforts in early agencies can reduce thefts during the highest-risk months rather than responding after losses have already occurred.

Design

Gestalt principles guided the overall design to ensure the visuals were easy to interpret and visually consistent. Similar colors were used to represent Kia thefts across all applicable charts, reinforcing grouping and recognition through similarity. A limited, consistent color palette was used wherever possible to reduce visual noise and keep attention on key patterns rather than decorative elements. Careful attention was paid to alignment of titles,

axes, and legends to support visual order and readability. Text sizing and spacing were applied consistently so that headings clearly communicate the main takeaway, often framed as a call to action, while supporting labels remain legible without competing for attention.

Ethical Considerations

Several ethical considerations were taken into account when working with these vehicle theft datasets. The analysis uses all four data sources exactly as provided and does not attempt to evaluate or reconcile discrepancies between them. While differences across datasets are present, no assumptions were made about bias, accuracy, or intent behind the data collection methods. The focus of the analysis is on identifying broad trends rather than drawing definitive conclusions from any single source.

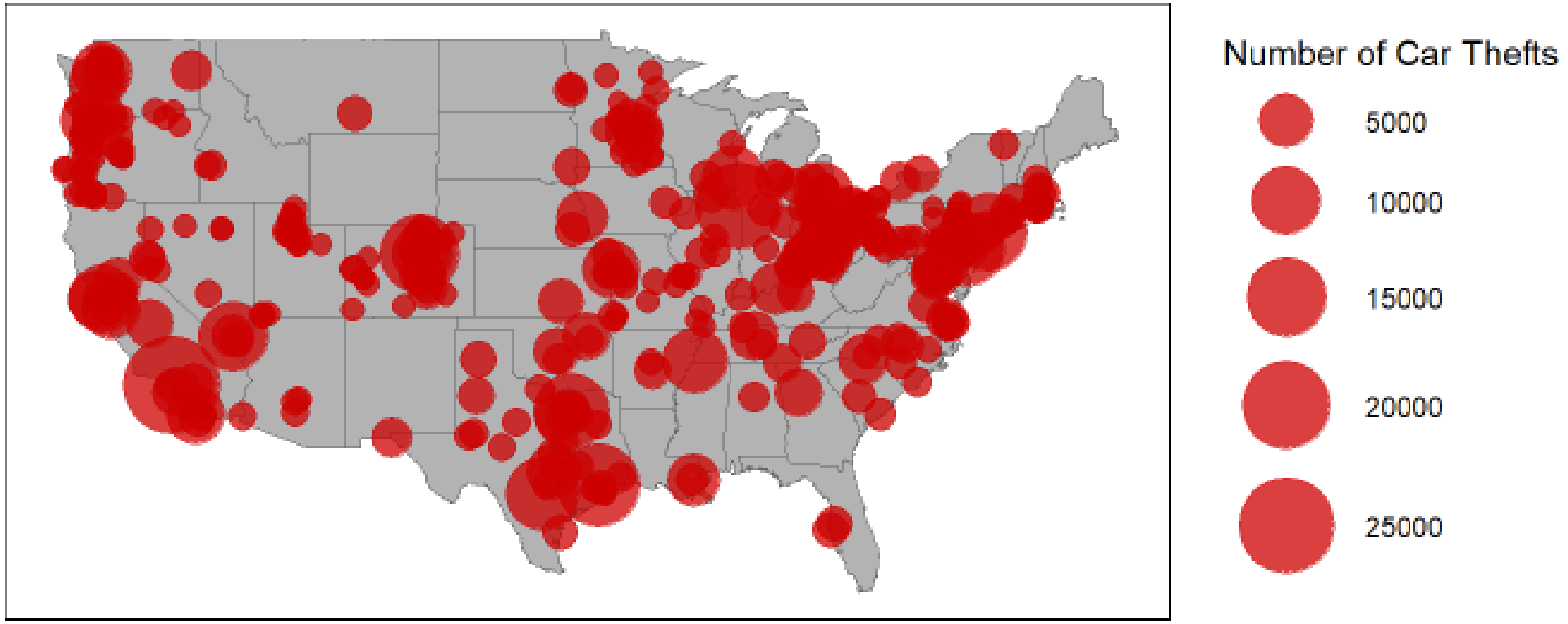
Care was also taken to avoid highlighting or segregating specific neighborhoods, demographic groups, or communities. All results are presented at higher geographic levels, such as city or state, to reduce the risk of stigmatization or misinterpretation. The goal is to inform policy discussions and prevention strategies, not to label certain areas or populations.

Car Theft Program

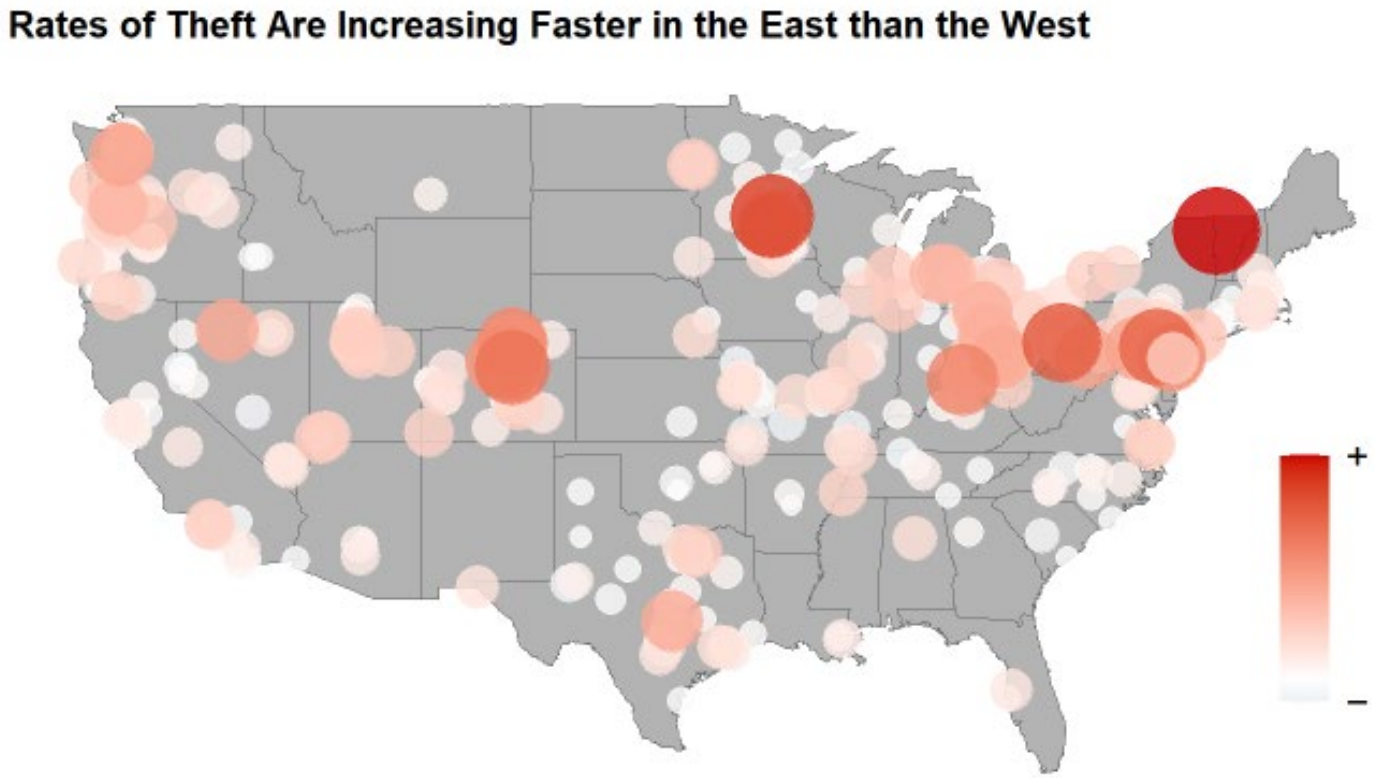
Lower Mainload Focus

Car Thefts are a Problem Nationwide

Car Thefts in 2022 by Location

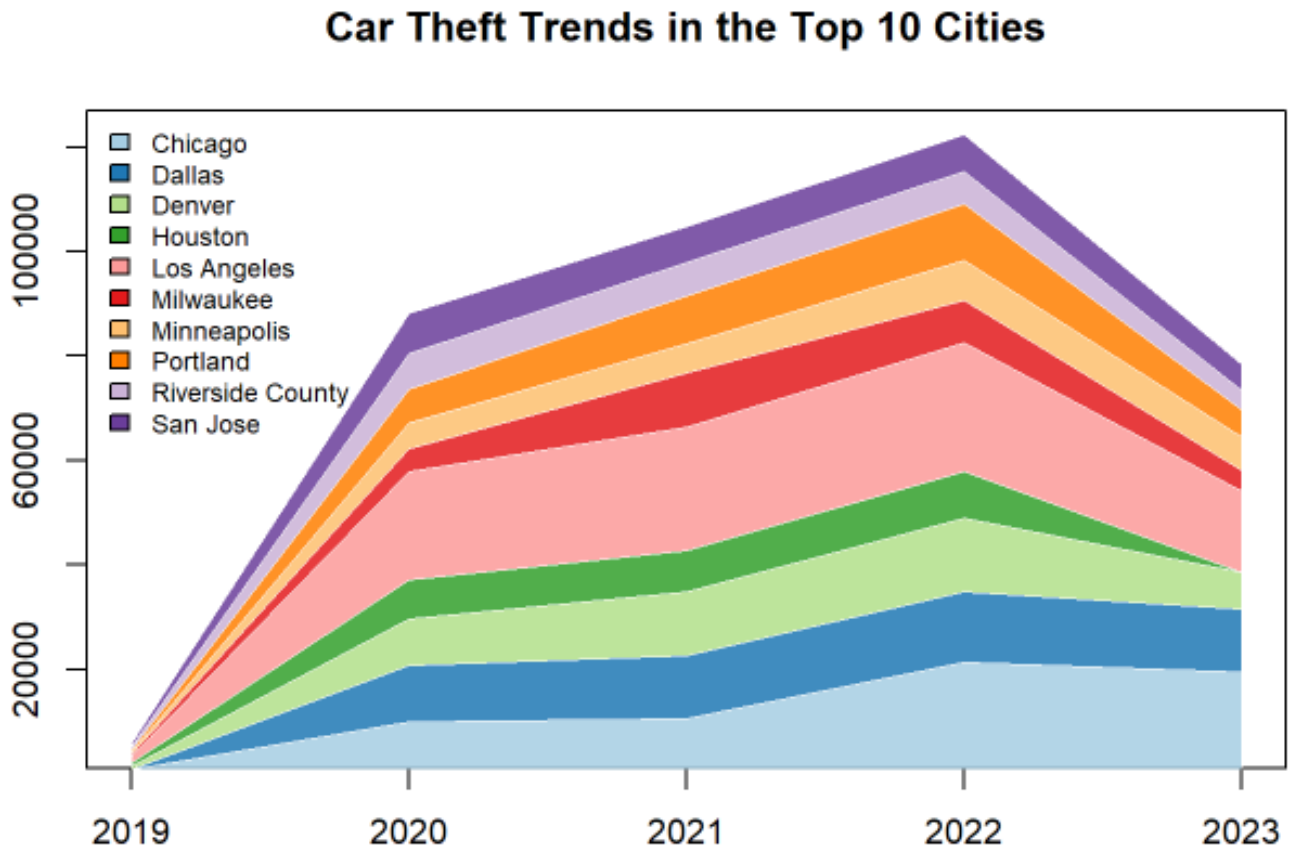


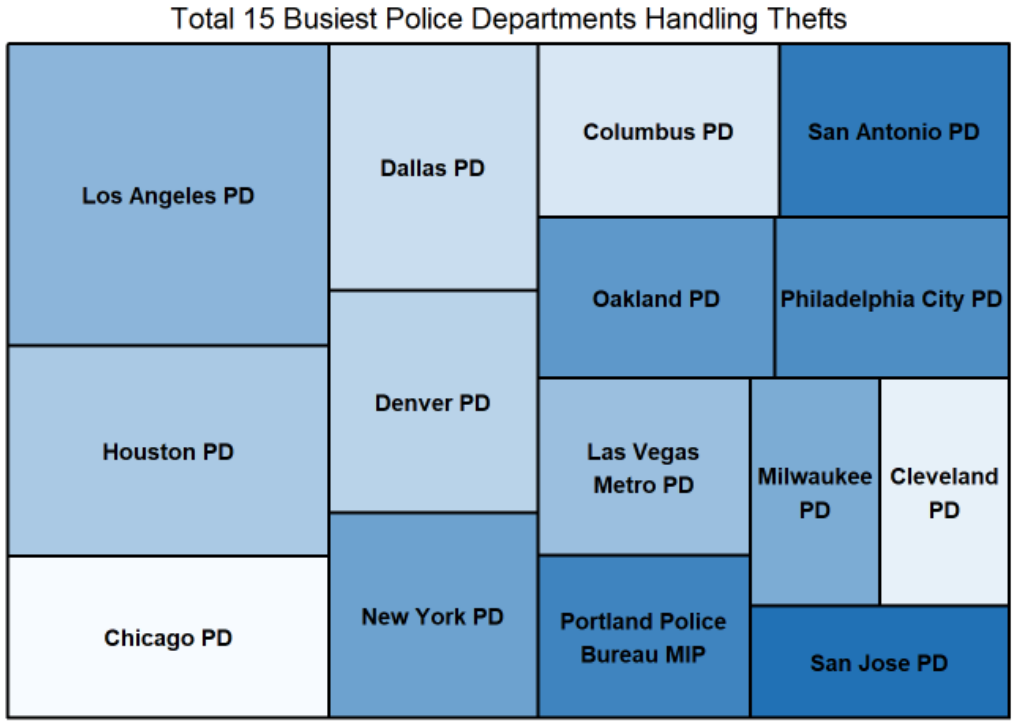
Prevention Efforts have been Successful in Eastern States



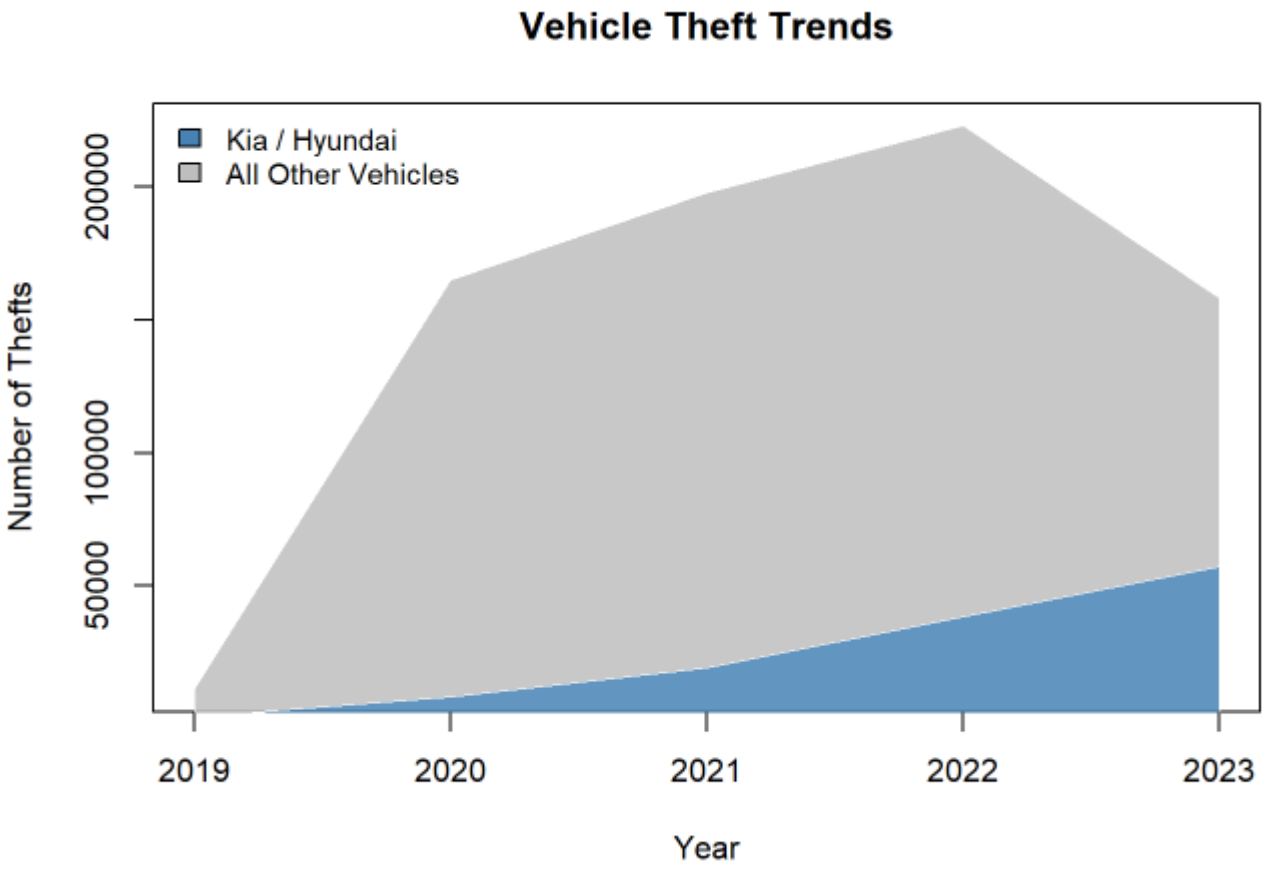
Overall deployment of our Theft Program has been a success, but more efforts are necessary.

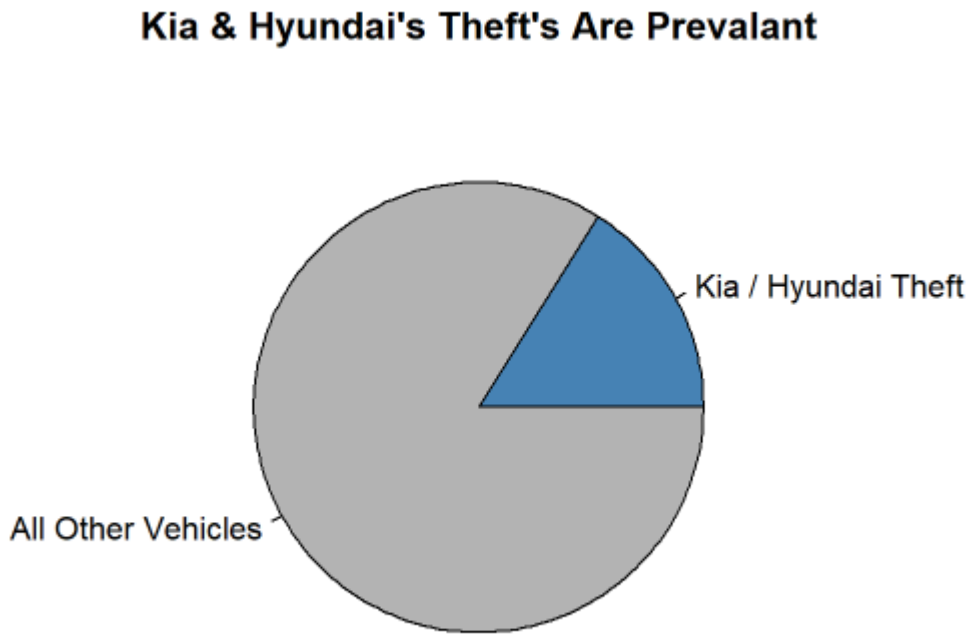
The Busiest Metro PD Departments have had success



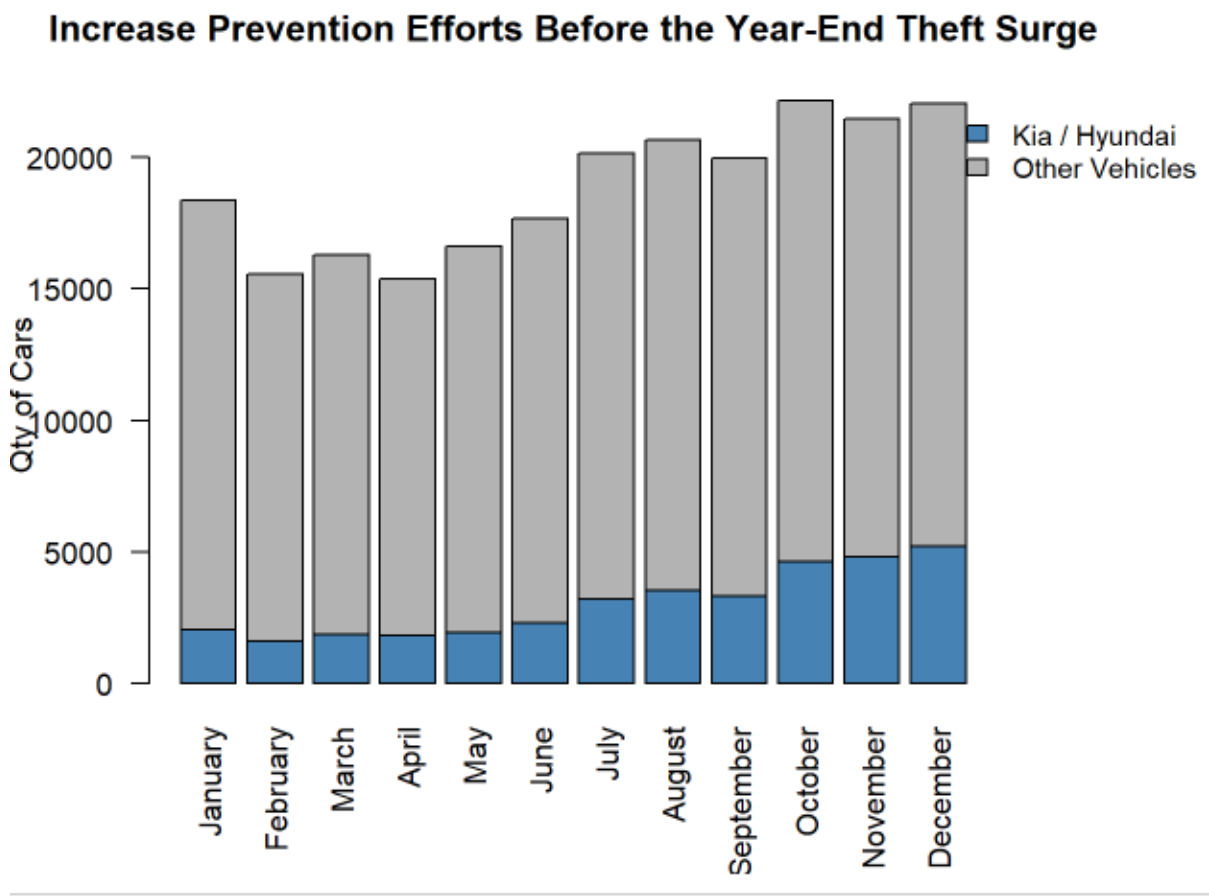


Target Your Local Theft Program Towards Kia/Hyundai



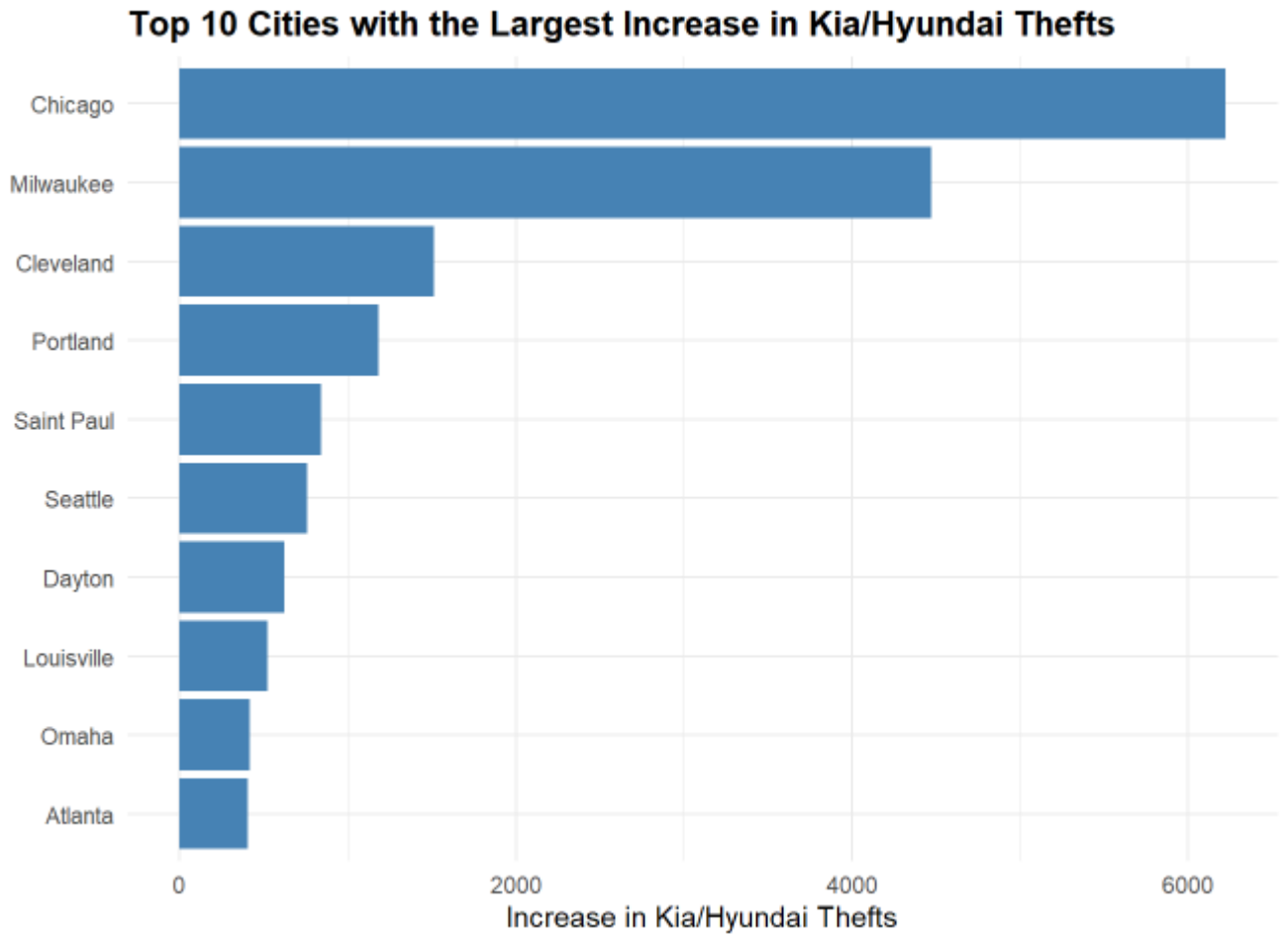


Prepare for the Surge Across all Car Types



Immediate Intervention Required

These cities should prioritize targeted vehicle theft prevention efforts



DSC640 Kia Thefts

2026-01-03

#Graphs I have to use for this assignment #Pie Charts #Donut Charts #Stacked Bars with Categorical Data #Tree Map #Area Chart #Stacked Area Chart

Import data

```
library(readxl)
setwd("C:/Users/samkl/Week 5-6"
)
getwd()
```

```
## [1] "C:/Users/samkl/Week 5-6"
```

```
list.files()
```

```
## [1] "~$Motherboard VICE News Kia Hyundai Theft Data.xlsx"
## [2] "carTheftsMap.csv"
## [3] "KiaHyundaiMilwaukeeData.csv"
## [4] "kiaHyundaiThefts.csv"
## [5] "Motherboard VICE News Kia Hyundai Theft Data.xlsx"
```

```
Milwaukee <- read.csv("KiaHyundaiMilwaukeeData.csv")
OtherCities <- read.csv("kiaHyundaiThefts.csv")
Theft_Map <- read.csv("carTheftsMap.csv")
News_Data <-read_excel(
  "Motherboard VICE News Kia Hyundai Theft Data.xlsx",
  col_names = FALSE
)
```

```
## New names:
## • `` -> `...1`
## • `` -> `...2`
## • `` -> `...3`
## • `` -> `...4`
## • `` -> `...5`
## • `` -> `...6`
## • `` -> `...7`
## • `` -> `...8`
## • `` -> `...9`
## • `` -> `...10`
## • `` -> `...11`
## • `` -> `...12`
## • `` -> `...13`
## • `` -> `...14`
## • `` -> `...15`
```

```
## • `` -> `...16`
## • `` -> `...17`
## • `` -> `...18`
## • `` -> `...19`
## • `` -> `...20`
## • `` -> `...21`
## • `` -> `...22`
## • `` -> `...23`
## • `` -> `...24`
## • `` -> `...25`
## • `` -> `...26`
## • `` -> `...27`
## • `` -> `...28`
## • `` -> `...29`
## • `` -> `...30`
## • `` -> `...31`
## • `` -> `...32`
## • `` -> `...33`
## • `` -> `...34`
## • `` -> `...35`
## • `` -> `...36`
## • `` -> `...37`
## • `` -> `...38`
## • `` -> `...39`
## • `` -> `...40`
## • `` -> `...41`
## • `` -> `...42`
## • `` -> `...43`
## • `` -> `...44`
## • `` -> `...45`
## • `` -> `...46`
## • `` -> `...47`
## • `` -> `...48`
## • `` -> `...49`
## • `` -> `...50`
## • `` -> `...51`
## • `` -> `...52`
## • `` -> `...53`
## • `` -> `...54`
## • `` -> `...55`
## • `` -> `...56`
## • `` -> `...57`
## • `` -> `...58`
## • `` -> `...59`
## • `` -> `...60`
## • `` -> `...61`
## • `` -> `...62`
## • `` -> `...63`
## • `` -> `...64`
## • `` -> `...65`
## • `` -> `...66`
## • `` -> `...67`
## • `` -> `...68`
## • `` -> `...69`
```

```
## • `` -> `...70`
## • `` -> `...71`
## • `` -> `...72`
## • `` -> `...73`
## • `` -> `...74`
## • `` -> `...75`
## • `` -> `...76`
## • `` -> `...77`
## • `` -> `...78`
## • `` -> `...79`
## • `` -> `...80`
## • `` -> `...81`
## • `` -> `...82`
## • `` -> `...83`
## • `` -> `...84`
## • `` -> `...85`
## • `` -> `...86`
## • `` -> `...87`
## • `` -> `...88`
## • `` -> `...89`
## • `` -> `...90`
## • `` -> `...91`
## • `` -> `...92`
## • `` -> `...93`
## • `` -> `...94`
## • `` -> `...95`
## • `` -> `...96`
## • `` -> `...97`
## • `` -> `...98`
## • `` -> `...99`
## • `` -> `...100`
## • `` -> `...101`
## • `` -> `...102`
## • `` -> `...103`
## • `` -> `...104`
## • `` -> `...105`
## • `` -> `...106`
## • `` -> `...107`
## • `` -> `...108`
## • `` -> `...109`
## • `` -> `...110`
## • `` -> `...111`
## • `` -> `...112`
## • `` -> `...113`
## • `` -> `...114`
## • `` -> `...115`
## • `` -> `...116`
## • `` -> `...117`
## • `` -> `...118`
## • `` -> `...119`
## • `` -> `...120`
## • `` -> `...121`
## • `` -> `...122`
## • `` -> `...123`
```

```
## • `` -> `...124`
## • `` -> `...125`
## • `` -> `...126`
## • `` -> `...127`
## • `` -> `...128`
## • `` -> `...129`
## • `` -> `...130`
## • `` -> `...131`
## • `` -> `...132`
## • `` -> `...133`
## • `` -> `...134`
## • `` -> `...135`
## • `` -> `...136`
## • `` -> `...137`
## • `` -> `...138`
## • `` -> `...139`
## • `` -> `...140`
## • `` -> `...141`
## • `` -> `...142`
## • `` -> `...143`
## • `` -> `...144`
## • `` -> `...145`
## • `` -> `...146`
## • `` -> `...147`
## • `` -> `...148`
## • `` -> `...149`
## • `` -> `...150`
## • `` -> `...151`
## • `` -> `...152`
## • `` -> `...153`
## • `` -> `...154`
## • `` -> `...155`
## • `` -> `...156`
## • `` -> `...157`
## • `` -> `...158`
## • `` -> `...159`
## • `` -> `...160`
## • `` -> `...161`
## • `` -> `...162`
## • `` -> `...163`
## • `` -> `...164`
## • `` -> `...165`
## • `` -> `...166`
## • `` -> `...167`
## • `` -> `...168`
## • `` -> `...169`
## • `` -> `...170`
## • `` -> `...171`
## • `` -> `...172`
## • `` -> `...173`
## • `` -> `...174`
## • `` -> `...175`
## • `` -> `...176`
## • `` -> `...177`
```

```
## • `` -> `...178`
## • `` -> `...179`
## • `` -> `...180`
## • `` -> `...181`
## • `` -> `...182`
## • `` -> `...183`
## • `` -> `...184`
## • `` -> `...185`
## • `` -> `...186`
## • `` -> `...187`
## • `` -> `...188`
## • `` -> `...189`
## • `` -> `...190`
## • `` -> `...191`
## • `` -> `...192`
## • `` -> `...193`
## • `` -> `...194`
## • `` -> `...195`
## • `` -> `...196`
## • `` -> `...197`
## • `` -> `...198`
## • `` -> `...199`
## • `` -> `...200`
## • `` -> `...201`
## • `` -> `...202`
## • `` -> `...203`
## • `` -> `...204`
## • `` -> `...205`
## • `` -> `...206`
## • `` -> `...207`
## • `` -> `...208`
## • `` -> `...209`
## • `` -> `...210`
## • `` -> `...211`
```

Append Milwaukee with the other City file to make one file

```
Theft_by_city <- rbind(Milwaukee, OtherCities)

Theft_Map <- read.csv("carTheftsMap.csv")
News_Data <-read_excel(
  "Motherboard VICE News Kia Hyundai Theft Data.xlsx",
  col_names = FALSE
)
```

```
## New names:
## • `` -> `...1`
## • `` -> `...2`
## • `` -> `...3`
## • `` -> `...4`
```

```
## • `` -> `...5`
## • `` -> `...6`
## • `` -> `...7`
## • `` -> `...8`
## • `` -> `...9`
## • `` -> `...10`
## • `` -> `...11`
## • `` -> `...12`
## • `` -> `...13`
## • `` -> `...14`
## • `` -> `...15`
## • `` -> `...16`
## • `` -> `...17`
## • `` -> `...18`
## • `` -> `...19`
## • `` -> `...20`
## • `` -> `...21`
## • `` -> `...22`
## • `` -> `...23`
## • `` -> `...24`
## • `` -> `...25`
## • `` -> `...26`
## • `` -> `...27`
## • `` -> `...28`
## • `` -> `...29`
## • `` -> `...30`
## • `` -> `...31`
## • `` -> `...32`
## • `` -> `...33`
## • `` -> `...34`
## • `` -> `...35`
## • `` -> `...36`
## • `` -> `...37`
## • `` -> `...38`
## • `` -> `...39`
## • `` -> `...40`
## • `` -> `...41`
## • `` -> `...42`
## • `` -> `...43`
## • `` -> `...44`
## • `` -> `...45`
## • `` -> `...46`
## • `` -> `...47`
## • `` -> `...48`
## • `` -> `...49`
## • `` -> `...50`
## • `` -> `...51`
## • `` -> `...52`
## • `` -> `...53`
## • `` -> `...54`
## • `` -> `...55`
## • `` -> `...56`
## • `` -> `...57`
## • `` -> `...58`
```

```
## • `` -> `...59`
## • `` -> `...60`
## • `` -> `...61`
## • `` -> `...62`
## • `` -> `...63`
## • `` -> `...64`
## • `` -> `...65`
## • `` -> `...66`
## • `` -> `...67`
## • `` -> `...68`
## • `` -> `...69`
## • `` -> `...70`
## • `` -> `...71`
## • `` -> `...72`
## • `` -> `...73`
## • `` -> `...74`
## • `` -> `...75`
## • `` -> `...76`
## • `` -> `...77`
## • `` -> `...78`
## • `` -> `...79`
## • `` -> `...80`
## • `` -> `...81`
## • `` -> `...82`
## • `` -> `...83`
## • `` -> `...84`
## • `` -> `...85`
## • `` -> `...86`
## • `` -> `...87`
## • `` -> `...88`
## • `` -> `...89`
## • `` -> `...90`
## • `` -> `...91`
## • `` -> `...92`
## • `` -> `...93`
## • `` -> `...94`
## • `` -> `...95`
## • `` -> `...96`
## • `` -> `...97`
## • `` -> `...98`
## • `` -> `...99`
## • `` -> `...100`
## • `` -> `...101`
## • `` -> `...102`
## • `` -> `...103`
## • `` -> `...104`
## • `` -> `...105`
## • `` -> `...106`
## • `` -> `...107`
## • `` -> `...108`
## • `` -> `...109`
## • `` -> `...110`
## • `` -> `...111`
## • `` -> `...112`
```

```
## • `` -> `...113`
## • `` -> `...114`
## • `` -> `...115`
## • `` -> `...116`
## • `` -> `...117`
## • `` -> `...118`
## • `` -> `...119`
## • `` -> `...120`
## • `` -> `...121`
## • `` -> `...122`
## • `` -> `...123`
## • `` -> `...124`
## • `` -> `...125`
## • `` -> `...126`
## • `` -> `...127`
## • `` -> `...128`
## • `` -> `...129`
## • `` -> `...130`
## • `` -> `...131`
## • `` -> `...132`
## • `` -> `...133`
## • `` -> `...134`
## • `` -> `...135`
## • `` -> `...136`
## • `` -> `...137`
## • `` -> `...138`
## • `` -> `...139`
## • `` -> `...140`
## • `` -> `...141`
## • `` -> `...142`
## • `` -> `...143`
## • `` -> `...144`
## • `` -> `...145`
## • `` -> `...146`
## • `` -> `...147`
## • `` -> `...148`
## • `` -> `...149`
## • `` -> `...150`
## • `` -> `...151`
## • `` -> `...152`
## • `` -> `...153`
## • `` -> `...154`
## • `` -> `...155`
## • `` -> `...156`
## • `` -> `...157`
## • `` -> `...158`
## • `` -> `...159`
## • `` -> `...160`
## • `` -> `...161`
## • `` -> `...162`
## • `` -> `...163`
## • `` -> `...164`
## • `` -> `...165`
## • `` -> `...166`
```

```
## • `` -> `...167`
## • `` -> `...168`
## • `` -> `...169`
## • `` -> `...170`
## • `` -> `...171`
## • `` -> `...172`
## • `` -> `...173`
## • `` -> `...174`
## • `` -> `...175`
## • `` -> `...176`
## • `` -> `...177`
## • `` -> `...178`
## • `` -> `...179`
## • `` -> `...180`
## • `` -> `...181`
## • `` -> `...182`
## • `` -> `...183`
## • `` -> `...184`
## • `` -> `...185`
## • `` -> `...186`
## • `` -> `...187`
## • `` -> `...188`
## • `` -> `...189`
## • `` -> `...190`
## • `` -> `...191`
## • `` -> `...192`
## • `` -> `...193`
## • `` -> `...194`
## • `` -> `...195`
## • `` -> `...196`
## • `` -> `...197`
## • `` -> `...198`
## • `` -> `...199`
## • `` -> `...200`
## • `` -> `...201`
## • `` -> `...202`
## • `` -> `...203`
## • `` -> `...204`
## • `` -> `...205`
## • `` -> `...206`
## • `` -> `...207`
## • `` -> `...208`
## • `` -> `...209`
## • `` -> `...210`
## • `` -> `...211`
```

Clean up the News_Data XLX

```
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
```

```
library(tidyr)
library(readr)
library(zoo)
```

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

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

```
library(ggplot2)
library(dplyr)
library(tidyr)

# ---- 0) Make News_Data a plain data.frame of characters (prevents tibble row-assign headaches) ----
News_Data <- as.data.frame(lapply(News_Data, as.character), stringsAsFactors = FALSE)

stopifnot(nrow(News_Data) >= 2)

# ---- 1) Build a cleaned "first header row" (fill-down) from News_Data[1, ] ----
first_row <- as.character(unlist(News_Data[1, ], use.names = FALSE))

# fill-down blanks/NA
first_row[first_row == ""] <- NA
for (i in seq_along(first_row)) {
 if (is.na(first_row[i]) && i > 1) first_row[i] <- first_row[i - 1]
}

# force first column name
first_row[1] <- "Date"

# ---- 2) Combine row 1 + row 2 into column names ----
second_row <- as.character(unlist(News_Data[2, ], use.names = FALSE))

new_names <- paste(first_row, second_row, sep = "__")
new_names[1] <- "Date"
new_names <- make.unique(new_names)

colnames(News_Data) <- new_names
```

```r
# ---- 3) Drop the two header rows ----
News_Data <- News_Data[-c(1, 2), , drop = FALSE]

# ---- 4) Unpivot then repivot to final dataset ----
long_data <- pivot_longer(
  News_Data,
  cols = -Date,
  names_to = c("City", "Metric"),
  names_sep = "__",
  values_to = "Value"
)

News_Data_Final <- pivot_wider(
  long_data,
  names_from = Metric,
  values_from = Value
)

# ---- 5) Fix Date + Year (now that News_Data_Final exists) ----
News_Data_Final$Date <- as.Date(News_Data_Final$Date)
News_Data_Final$Year <- format(News_Data_Final$Date, "%Y")

# ---- 6) Split City into City / State ----
News_Data_Final <- News_Data_Final %>%
  separate(
    City,
    into = c("City", "State"),
    sep = ",\\s*",
    fill = "right",
    remove = TRUE
  )

# sanity checks
str(News_Data_Final$Date)
```

```
##  Date[1:3150], format: "2019-12-01" "2019-12-01" "2019-12-01" "2019-12-01" "2019-12-01" ...
```

```r
unique(News_Data_Final$City)
```

```
##  [1] "Denver"                "El Paso"               "Portland"
##  [4] "Atlanta"               "Chicago"               "Virginia Beach"
##  [7] "Louisville"            "San Diego"             "Sacramento"
## [10] "Dallas"                "Fort Worth"            "Bakersfield"
## [13] "Omaha"                 "Lubbock"               "Fremont"
## [16] "San Antonio"           "Irving"                "Madison"
## [19] "Frisco"                "Lincoln"               "Wichita"
## [22] "Plano"                 "Akron"                 "Buffalo"
## [25] "Chula Vista"           "Reno"                  "Oxnard"
## [28] "San Francisco"         "McKinney"              "Arlington"
## [31] "Garland"               "Riverside County"      "Stockton"
## [34] "Corpus Cristi"         "Cincinatti"            "Montgomery County"
```

```
## [37] "Henderson"              "San Bernardino"         "Newport News"
## [40] "Amarillo"               "New Haven"              "Washington"
## [43] "Boise"                  "Austin"                 "Chandler"
## [46] "Modesto"                "San Jose"               "Aurora"
## [49] "Vancouver"              "Raleigh"                "St. Petersburg"
## [52] "Milwaukee"              "Los Angeles"            "Glendale"
## [55] "Orlando"                "Eugene"                 "Rochester"
## [58] "Durham"                 "Norfolk"                "Pittsburgh"
## [61] "Miami"                  "Spokane"                "Salt Lake City"
## [64] "Minneapolis"            "Anaheim"                "Providence"
## [67] "Syracuse"               "Peoria"                 "Houston"
## [70] "Prince George's County"
```

```
#Calculate Percent of Kia in News data

# Force numeric (remove commas just in case)
News_Data_Final$`Kia/Hyundais` <- as.numeric(
  gsub(",", "", News_Data_Final$`Kia/Hyundais`)
)

News_Data_Final$All <- as.numeric(
  gsub(",", "", News_Data_Final$All)
)
News_Data_Final$`Kia Share` <- ifelse(
  News_Data_Final$All == 0,
  NA,
  (News_Data_Final$`Kia/Hyundais` / News_Data_Final$All) * 100
)

#Groupby to graph theft by city as reported on news

city_summary_news <- News_Data_Final %>%
  group_by(City) %>%
  summarise(
    Kias_Stolen = sum(`Kia/Hyundais`, na.rm = TRUE),
    Count = n(),
    .groups = "drop"
  )

#Groupby to graph theft by city

city_summary <- Theft_by_city %>%
  group_by(city) %>%
  summarise(
    countKiaHyundaiThefts = sum(countKiaHyundaiThefts, na.rm = TRUE),
    .groups = "drop"
  )
```

##Create TOtal Thefts in Theft by City

```
Theft_by_city$Total_Thefts <-
  Theft_by_city$countKiaHyundaiThefts + Theft_by_city$countOtherThefts
```

```
#Calculate Percent of Kia in Theft by City
Theft_by_city$Kia_Share <- ifelse(
  Theft_by_city$Total_Thefts == 0,
  NA,
  (Theft_by_city$countKiaHyundaiThefts / Theft_by_city$Total_Thefts) * 100
)
```

##Look at Value in News Data

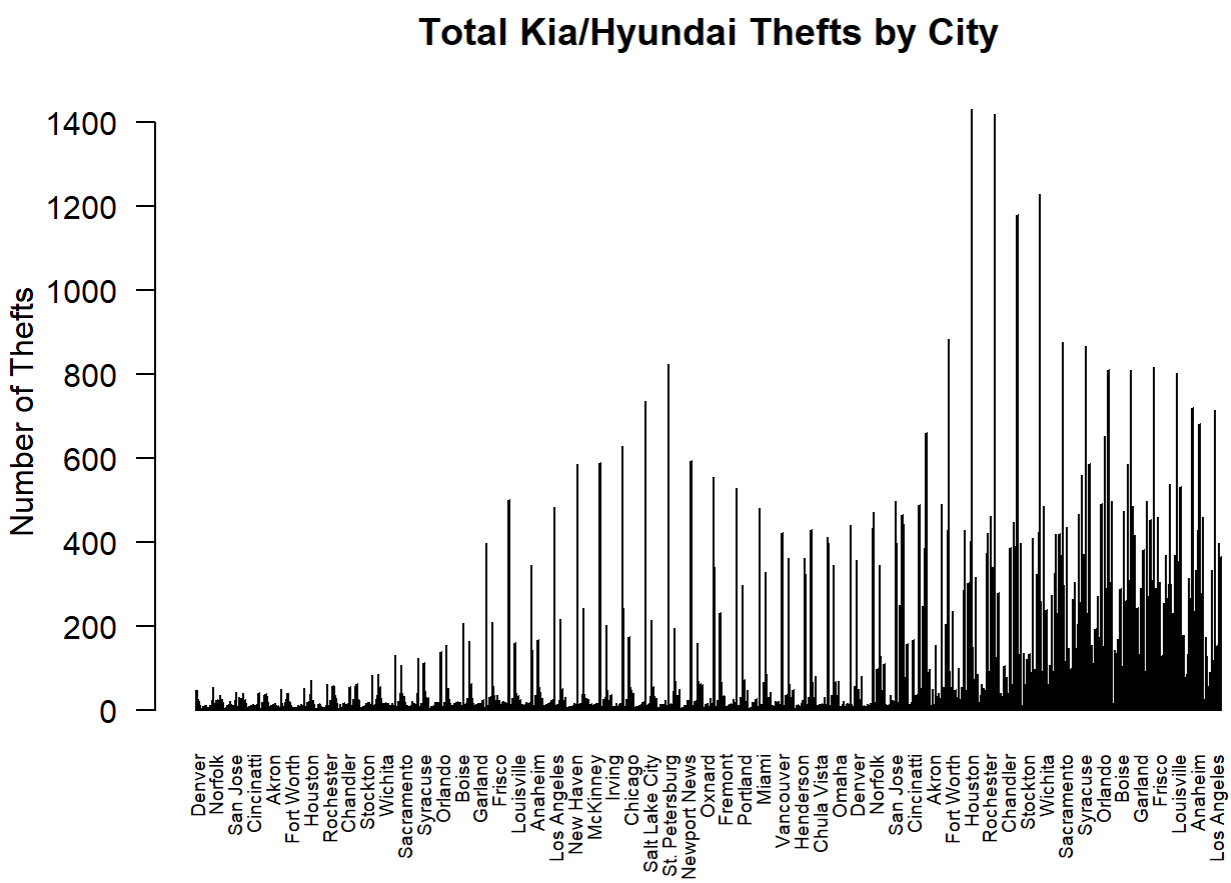
```
#Look at the values in News_data

str(News_Data_Final$`Kia/Hyundais`)
```

```
##  num [1:3150] 48 13 13 NA 46 NA 14 25 2 21 ...
```

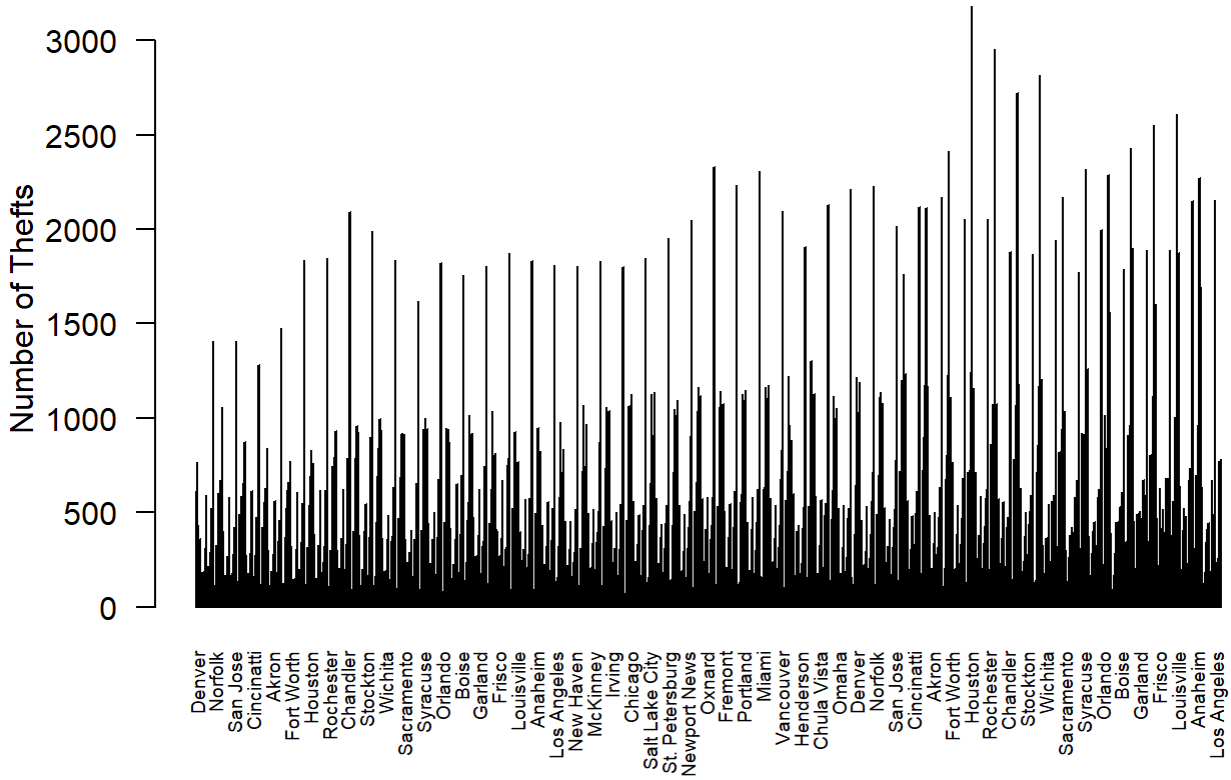
```
News_Data_Final$`Kia/Hyundais` <-
  as.numeric(gsub(",", "", News_Data_Final$`Kia/Hyundais`))

News_Data_Final$All <-
  as.numeric(gsub(",", "", News_Data_Final$All))
```

##Barplot of Total Thefts By City from News Data

```
bar_kia_hyundai_by_city <-
barplot(
  height = News_Data_Final$`Kia/Hyundais`,
  names.arg = News_Data_Final$City,
  las = 2,              # rotate labels
  cex.names = 0.6,      # make labels smaller
  main = "Total Kia/Hyundai Thefts by City",
  ylab = "Number of Thefts"
)
```

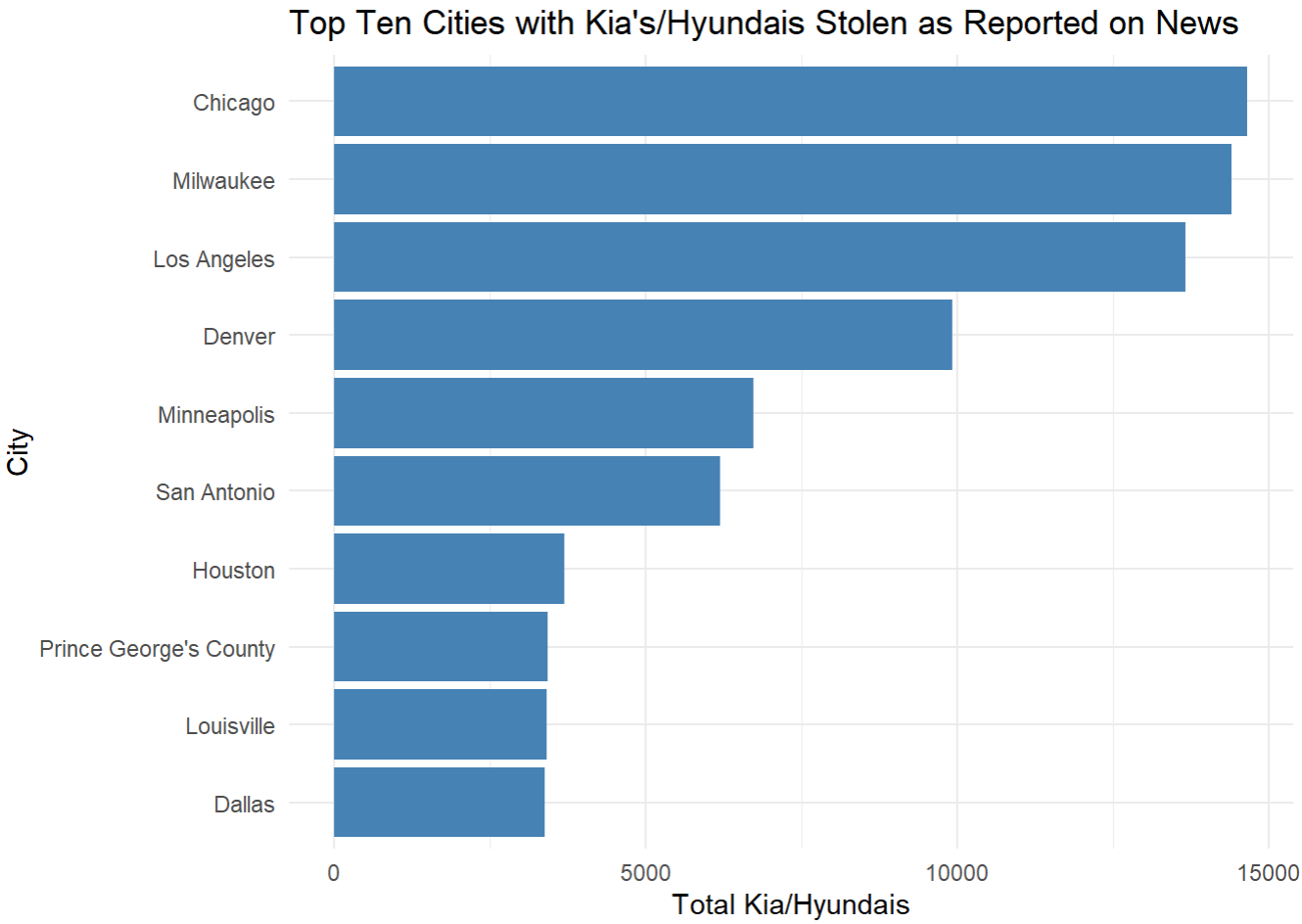



```
bar_totalthefts_by_city <-
barplot(
  height = News_Data_Final$All,
  names.arg = News_Data_Final$City,
  las = 2,              # rotate labels
  cex.names = 0.6,      # make labels smaller
  main = "Total Thefts by City",
  ylab = "Number of Thefts"
)
```

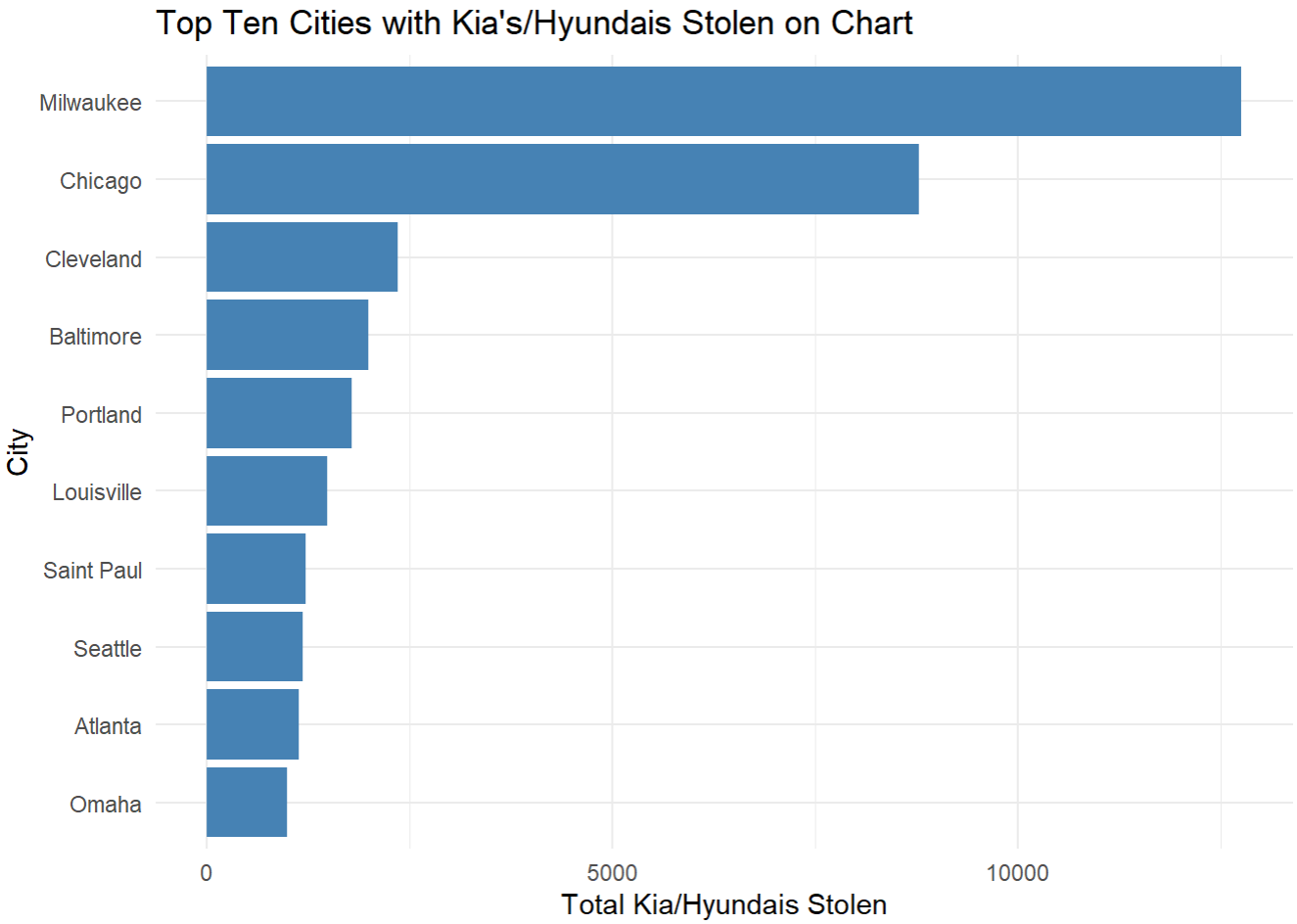


##Bargraph Graph Top 10 Cities as reported by news

```
#graph top ten cities
top10_cities_news <- city_summary_news %>%
  arrange(desc(Kias_Stolen)) %>%
  slice_head(n = 10)

library(ggplot2)

Top_ten_cities_bargraph <-
ggplot(top10_cities_news, aes(x = reorder(City, Kias_Stolen), y = Kias_Stolen)) +
  geom_col(fill = "steelblue") +
  coord_flip() +
  theme_minimal() +
  labs(
    title = "Top Ten Cities with Kia's/Hyundais Stolen as Reported on News",
    x = "City",
    y = "Total Kia/Hyundais"
  )

Top_ten_cities_bargraph
```

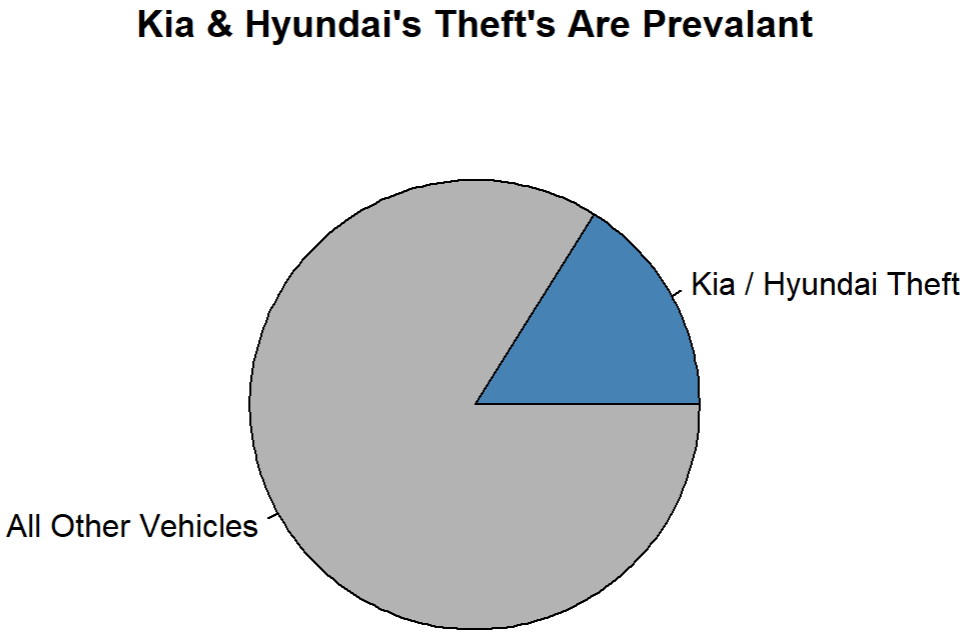



```
#graph top ten cities from Theft_city_Data
top10_cities <- city_summary %>%
 arrange(desc(countKiaHyundaiThefts)) %>%
 slice_head(n = 10)

library(ggplot2)

top10_cities_kias_Chartdata <-
ggplot(top10_cities, aes(x = reorder(city, countKiaHyundaiThefts), y = countKiaHyundaiThefts)) +
 geom_col(fill = "steelblue") +
 coord_flip() +
 theme_minimal() +
 labs(
  title = "Top Ten Cities with Kia's/Hyundais Stolen on Chart",
  x = "City",
  y = "Total Kia/Hyundais Stolen"
 )

top10_cities_kias_Chartdata
```



##Pie Chart

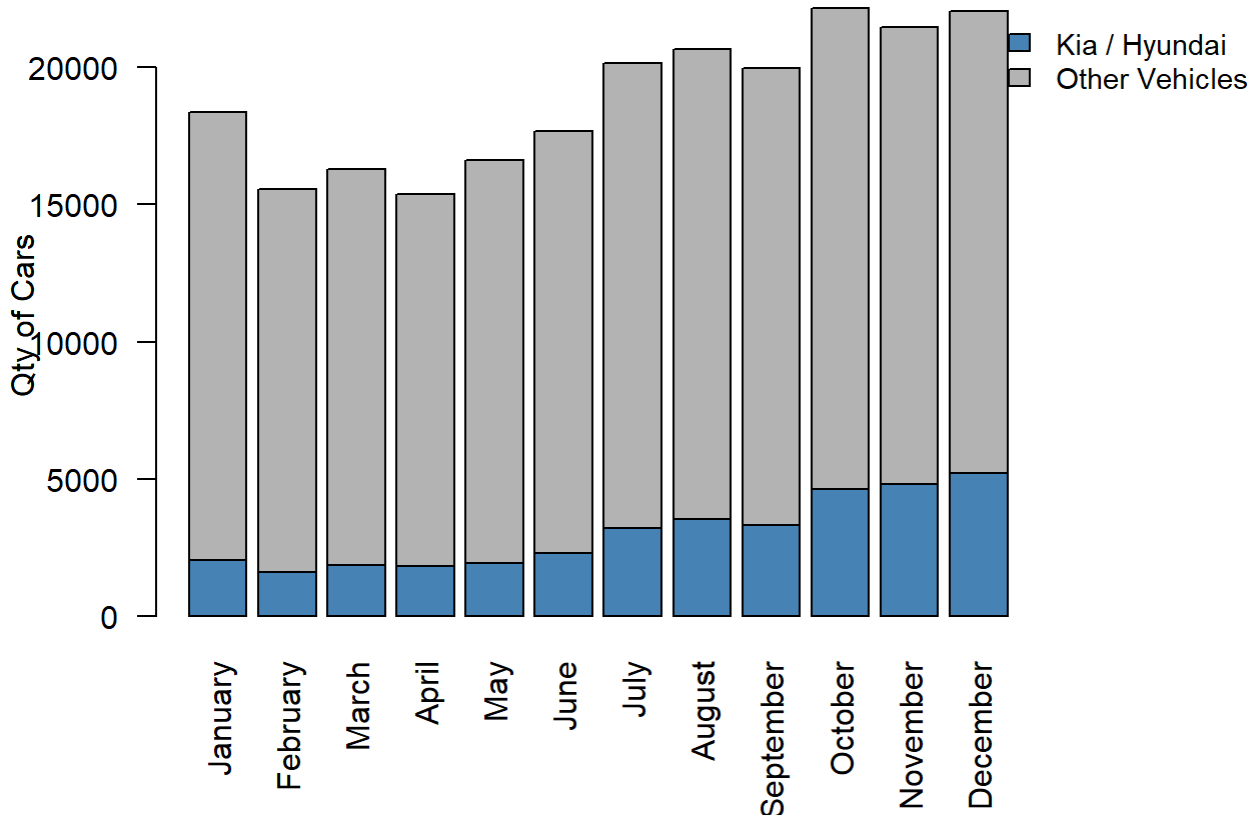
```
#Pie Chart of Kia Thefts vs All Thefts (using Theft by City Data)

# Aggregate totals across the dataset
kia_total <- sum(Theft_by_city$countKiaHyundaiThefts, na.rm = TRUE)
other_total <- sum(Theft_by_city$countOtherThefts, na.rm = TRUE)

# Create vector for pie chart
theft_counts <- c(
  "Kia / Hyundai Theft" = kia_total,
  "All Other Vehicles" = other_total
)

# Pie chart
kia_alltheft_piechart<-
pie(
  theft_counts,
  main = "Kia & Hyundai's Theft's Are Prevalant",
  col = c("steelblue", "gray70")
)
```



##Monthly Vehicle Theft by Type (stacked bar graph)

```
# Create a proper Date from year + month
# Create a proper Date from year + month
Theft_by_city$Date <- as.Date(
  paste(Theft_by_city$year, Theft_by_city$month, "01"),
  format = "%Y %b %d"
)

Theft_by_city$Date <- as.Date(Theft_by_city$Date)

# Create Month factor
Theft_by_city$Month <- factor(
  format(Theft_by_city$Date, "%B"),
  levels = month.name
)

# Aggregate by Month
monthly_type <- aggregate(
  cbind(countKiaHyundaiThefts, countOtherThefts) ~ Month,
  data = Theft_by_city,
  sum,
  na.rm = TRUE
)

# ---- GRAPHICS SETTINGS (THIS IS THE KEY FIX) ----
```

```r
op <- par(no.readonly = TRUE)

# Increase right margin significantly
par(
  mar = c(5, 4, 4, 8),  # bottom, left, top, RIGHT
  xpd = NA              # allow drawing outside plot region
)

# Plot stacked bar chart
barplot(
  t(as.matrix(monthly_type[, -1])),
  names.arg = monthly_type$Month,
  las = 2,
  col = c("steelblue", "gray70"),
  main = "Increase Prevention Efforts Before the Year-End Theft Surge",
  ylab = "Qty of Cars"
)

# Legend — now guaranteed visible
legend(
  "topright",
  inset = c(-0.25, 0),
  legend = c("Kia / Hyundai", "Other Vehicles"),
  fill = c("steelblue", "gray70"),
  bty = "n",
  cex = 0.9
)
```




```
# Reset graphics parameters
par(op)
```

##Tree Map

```
#reshape theft map data

library(tidyr)

Theft_Map_long <- pivot_longer(
  Theft_Map,
  cols = starts_with("countCarThefts"),
  names_to = "Year",
  values_to = "Thefts"
)

Theft_Map_long$Thefts <- as.numeric(gsub(",", "", Theft_Map_long$Thefts))
```

```
## Warning: NAs introduced by coercion
```

```
Theft_Map_long$Thefts <- as.numeric(
  gsub("[^0-9.]", "", Theft_Map_long$Thefts)
)
```

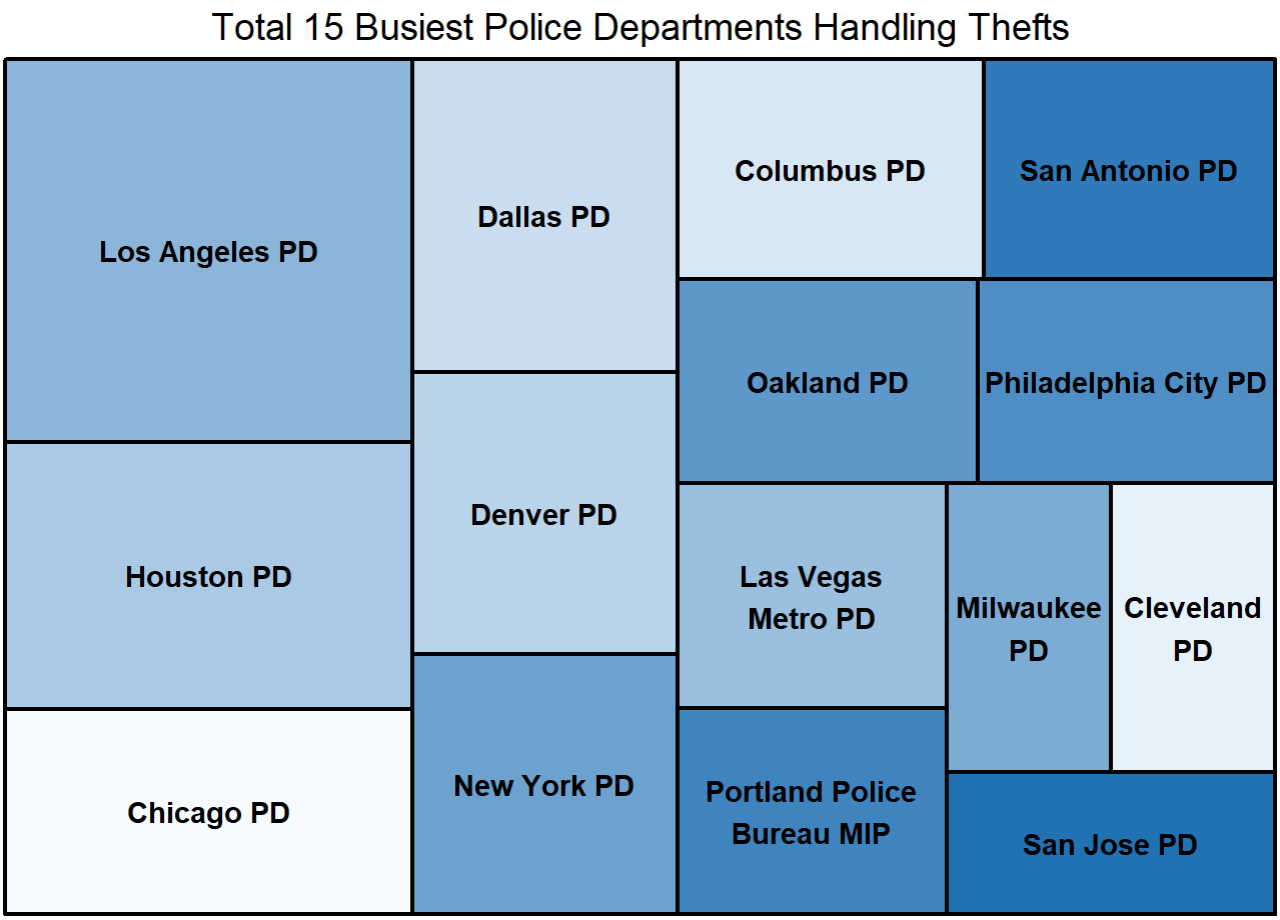
```r
# Clean up Year
Theft_Map_long$Year <- gsub("countCarThefts", "", Theft_Map_long$Year)
Theft_Map_long$Year <- as.integer(Theft_Map_long$Year)

names(Theft_Map_long)
```

```
## [1] "agency_ori"              "geo_name"
## [3] "latitude"                "longitude"
## [5] "percentChange2019to2022" "Year"
## [7] "Thefts"
```

```r
# 1) Aggregate total thefts by agency
agency_totals <- aggregate(
  Thefts ~ geo_name,
  data = Theft_Map_long,
  sum,
  na.rm = TRUE
)

# 2) Keep top 15 agencies (treemap gets unreadable otherwise)
agency_totals <- agency_totals[order(-agency_totals$Thefts), ]
top_agencies <- head(agency_totals, 15)

library(RColorBrewer)
library(viridis)
```

```
## Loading required package: viridisLite
```

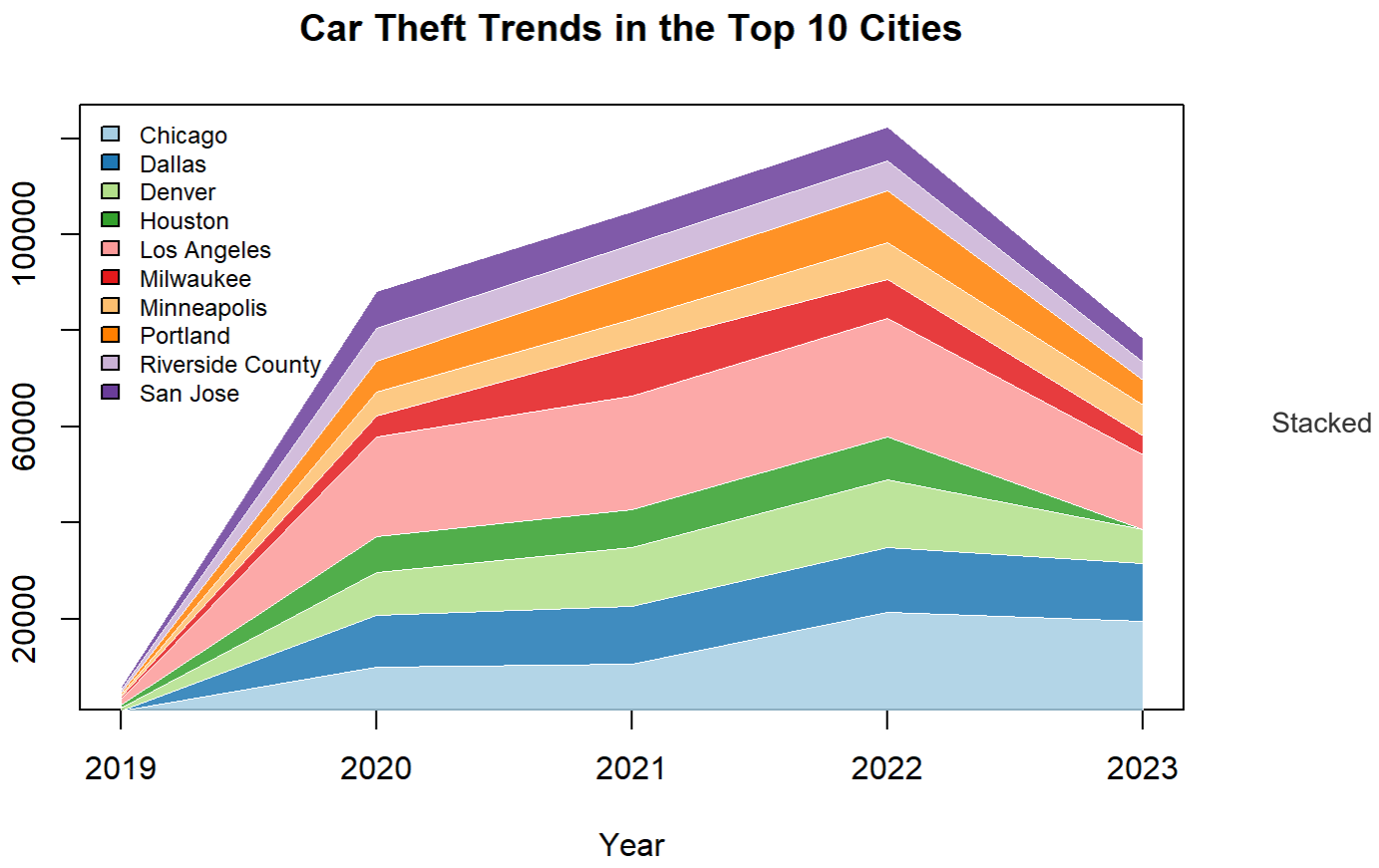
```r
city_palette <- colorRampPalette(
  c("#f7fbff", "#2171b5")   # lighter dark-end so text stays black
)(nrow(top_agencies))

# 3) Treemap
library(treemap)

Busy_PD_Treemap<-
treemap(
  top_agencies,
  index = "geo_name",
  vSize = "Thefts",
  palette = city_palette,
  fontcolor = "black",
  title = "Total 15 Busiest Police Departments Handling Thefts"
)
```



##Stacked Area Graph

```
#look for trends by year & city (using News_data)
# 1) Create numeric Year
News_Data_Final$Year <- as.integer(format(News_Data_Final$Date, "%Y"))

# 2) Aggregate thefts by Year + City
year_city <- aggregate(
  All ~ Year + City,
  data = News_Data_Final,
  sum,
  na.rm = TRUE
)

# 3) Find top 10 cities overall
city_totals <- aggregate(All ~ City, data = year_city, sum, na.rm = TRUE)
top_cities <- head(city_totals[order(-city_totals$All), "City"], 10)

# 4) Filter to top cities and reshape wide
library(tidyr)

year_city_top <- year_city[year_city$City %in% top_cities, ]

year_city_wide <- pivot_wider(
  year_city_top,
  names_from = City,
```

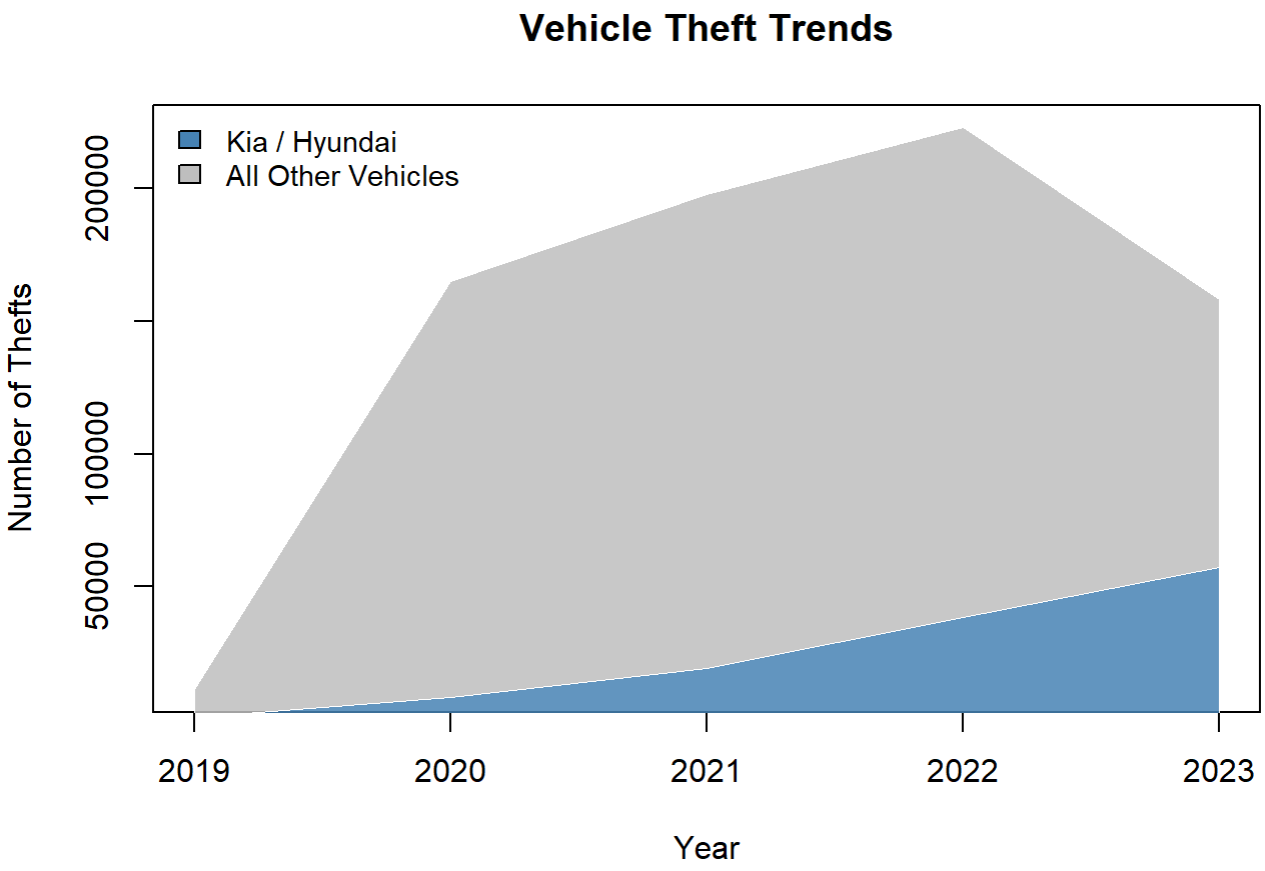
```
  values_from = All,
  values_fill = 0
)

year_city_wide <- year_city_wide[order(year_city_wide$Year), ]

# 5) Plot stacked area (base R) with DISTINCT colors
theft_matrix <- as.matrix(year_city_wide[, -1])
years <- year_city_wide$Year

library(RColorBrewer)

#    High-contrast categorical colors (10 cities)
colors <- brewer.pal(ncol(theft_matrix), "Paired")

cumulative <- rep(0, nrow(theft_matrix))

plot(
  years,
  rowSums(theft_matrix),
  type = "n",
  xlab = "Year",
  ylab = "",
  main = "Car Theft Trends in the Top 10 Cities"
)

for (i in seq_len(ncol(theft_matrix))) {
  polygon(
    c(years, rev(years)),
    c(cumulative + theft_matrix[, i], rev(cumulative)),
    col = adjustcolor(colors[i], alpha.f = 0.85),
    border = "white",
    lwd = 0.6
  )
  cumulative <- cumulative + theft_matrix[, i]
}

legend(
  "topleft",
  legend = colnames(theft_matrix),
  fill = colors,
  cex = 0.75,
  bty = "n"
)
```



Area Bar of KIA vs ALL

```
# --- Stacked area: Kia/Hyundai vs All Other (News data) ---

library(dplyr)

# 1) Ensure Year exists
News_Data_Final$Year <- as.integer(format(News_Data_Final$Date, "%Y"))

# 2) Make sure the columns are numeric (handles commas)
News_Data_Final$`Kia/Hyundais` <- as.numeric(gsub(",", "", News_Data_Final$`Kia/Hyundais`))
News_Data_Final$All <- as.numeric(gsub(",", "", News_Data_Final$All))

# 3) Aggregate totals by Year (across all cities)
year_totals <- News_Data_Final %>%
  group_by(Year) %>%
  summarise(
    KiaHyundai = sum(`Kia/Hyundais`, na.rm = TRUE),
    AllThefts  = sum(All, na.rm = TRUE),
    .groups = "drop"
  ) %>%
  mutate(
    OtherVehicles = pmax(AllThefts - KiaHyundai, 0)  # avoid negatives if any data weirdness
  ) %>%
  arrange(Year)

# 4) Build matrix for stacked area
```

```r
years <- year_totals$Year
theft_matrix <- as.matrix(year_totals[, c("KiaHyundai", "OtherVehicles")])

# 5) Plot (2 clean, high-contrast colors)
colors <- c("steelblue", "grey")

cumulative <- rep(0, nrow(theft_matrix))

plot(
  years,
  rowSums(theft_matrix),
  type = "n",
  xlab = "Year",
  ylab = "Number of Thefts",
  main = "Vehicle Theft Trends"
)

for (i in seq_len(ncol(theft_matrix))) {
  polygon(
    c(years, rev(years)),
    c(cumulative + theft_matrix[, i], rev(cumulative)),
    col = adjustcolor(colors[i], alpha.f = 0.85),
    border = "white",
    lwd = 0.8
  )
  cumulative <- cumulative + theft_matrix[, i]
}

legend(
  "topleft",
  legend = c("Kia / Hyundai", "All Other Vehicles"),
  fill = colors,
  cex = 0.9,
  bty = "n"
)
```



Compare Top agencies

```
#print top agencies to ensure they are the same color as my stacked area graph
top_agencies
```

```
##                       geo_name Thefts
## 264              Los Angeles PD  87440
## 212                  Houston PD  60740
## 88                   Chicago PD  46701
## 115                   Dallas PD  46586
## 124                   Denver PD  41863
## 321                 New York PD  38631
## 103                 Columbus PD  37770
## 417              San Antonio PD  35865
## 339                  Oakland PD  34422
## 377       Philadelphia City PD  34075
## 247         Las Vegas Metro PD  33959
## 386 Portland Police Bureau MIP  30962
## 298                Milwaukee PD  26566
## 97                 Cleveland PD  26545
## 421                 San Jose PD  26079
```

```
top_cities
```

```
##  [1] "Los Angeles"      "Chicago"          "Dallas"           "Denver"
##  [5] "Portland"         "Milwaukee"        "San Jose"         "Minneapolis"
##  [9] "Houston"          "Riverside County"
```

##Physical Map of US and Theft Rates

```
library(ggplot2)
library(maps)
```

```
##
## Attaching package: 'maps'
```

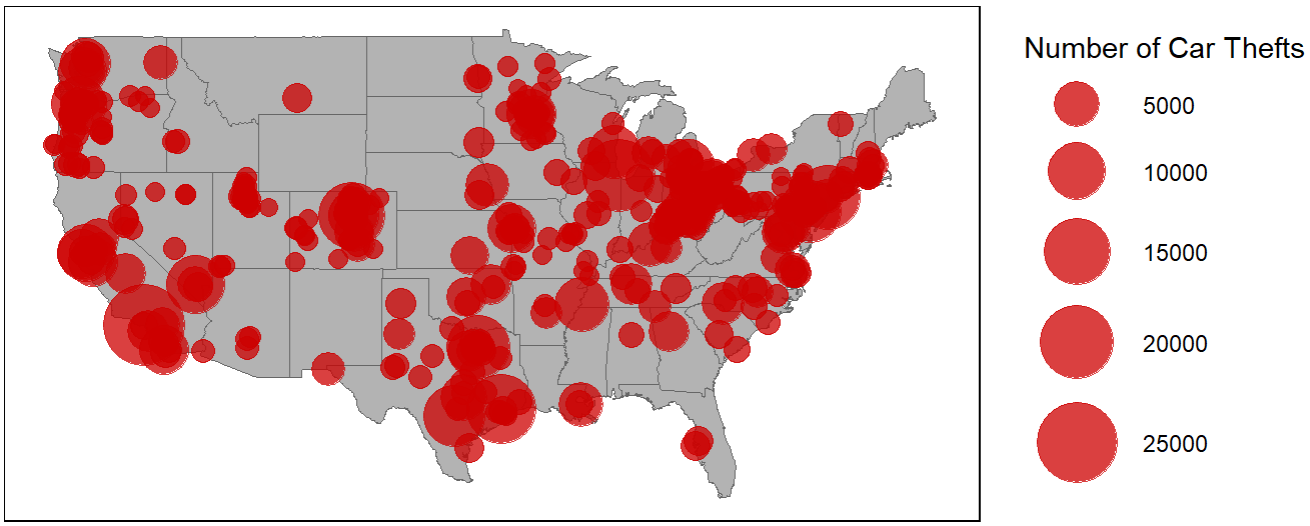
```
## The following object is masked from 'package:viridis':
##
##     unemp
```

```
library(ggplot2)
library(maps)

# Ensure numeric
Theft_Map$longitude <- as.numeric(Theft_Map$longitude)
Theft_Map$latitude  <- as.numeric(Theft_Map$latitude)
Theft_Map$countCarThefts2022 <- as.numeric(Theft_Map$countCarThefts2022)
```

```
## Warning: NAs introduced by coercion
```

```
Theft_Map <- Theft_Map[
  !is.na(Theft_Map$longitude) &
  !is.na(Theft_Map$latitude) &
  !is.na(Theft_Map$countCarThefts2022),
]

# Remove Hawaii / zoom to lower 48
Theft_Map <- Theft_Map[
  Theft_Map$longitude > -130 & Theft_Map$longitude < -60 &
  Theft_Map$latitude  > 24   & Theft_Map$latitude  < 50,
]

us_map <- map_data("state")

ggplot() +
  geom_polygon(
    data = us_map,
    aes(long, lat, group = group),
    fill = "gray70",
    color = "gray40",
    linewidth = 0.2
  ) +
  geom_point(
    data = Theft_Map,
```

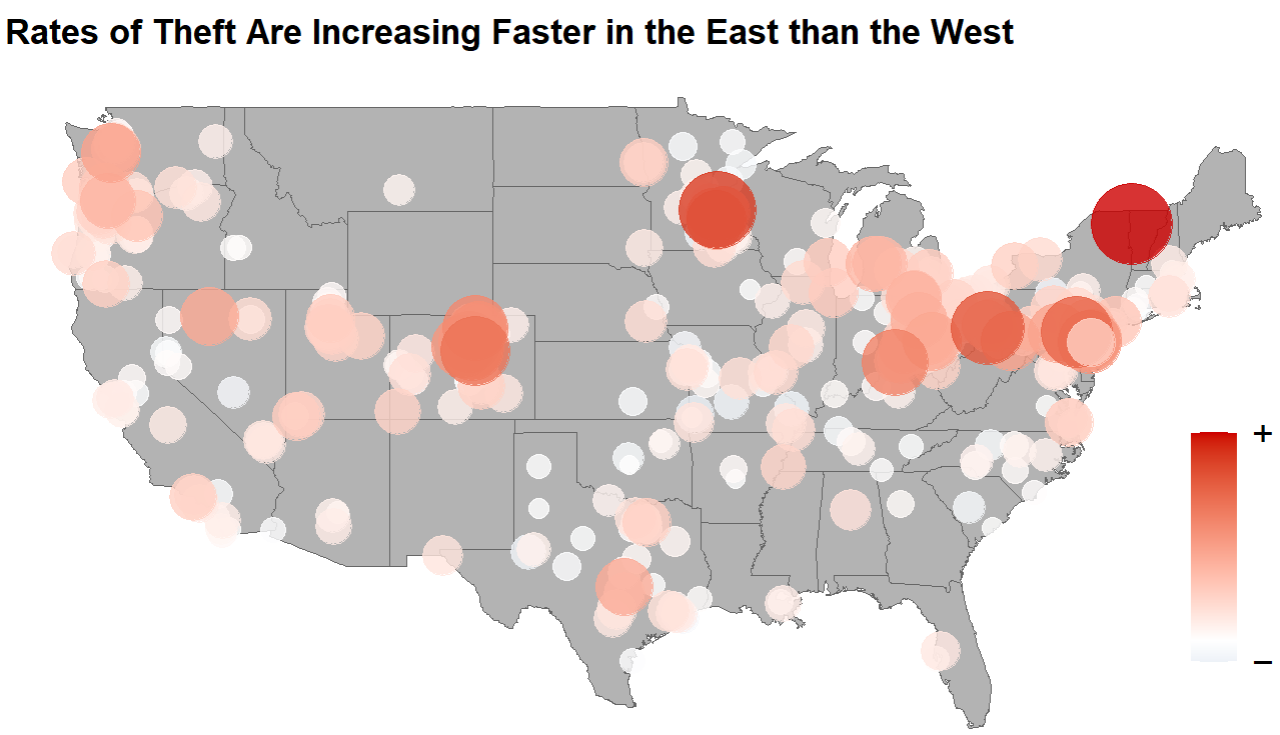
```
    aes(
      x = longitude,
      y = latitude,
      size = countCarThefts2022
    ),
    color = "red3",
    alpha = 0.75
  ) +
  scale_size_continuous(range = c(3, 14)) +
  coord_quickmap() +
  labs(
    title = "Car Thefts in 2022 by Location",
    size = "Number of Car Thefts"
  ) +
  theme_minimal() +
  theme(
axis.title = element_blank(),   # remove x/y titles
axis.text  = element_blank(),   # remove x/y numbers
axis.ticks = element_blank(),   # remove tick marks
    panel.grid = element_blank(),
    panel.background = element_rect(fill = "white"),
    plot.title = element_text(face = "bold")
  )
```




```
library(ggplot2)
library(maps)
```

```r
# Ensure numeric
Theft_Map$longitude <- as.numeric(Theft_Map$longitude)
Theft_Map$latitude  <- as.numeric(Theft_Map$latitude)
Theft_Map$percentChange2019to2022 <- as.numeric(Theft_Map$percentChange2019to2022)

# Remove missing values
Theft_Map <- Theft_Map[
  !is.na(Theft_Map$longitude) &
  !is.na(Theft_Map$latitude) &
  !is.na(Theft_Map$percentChange2019to2022),
]

# Zoom to lower 48 (removes HI / AK)
Theft_Map <- Theft_Map[
  Theft_Map$longitude > -130 & Theft_Map$longitude < -60 &
  Theft_Map$latitude  > 24   & Theft_Map$latitude  < 50,
]

# Map background
us_map <- map_data("state")

# Plot
ggplot() +
  geom_polygon(
    data = us_map,
    aes(long, lat, group = group),
    fill = "gray70",
    color = "gray40",
    linewidth = 0.2
  ) +
  geom_point(
    data = Theft_Map,
    aes(
      x = longitude,
      y = latitude,
      size = abs(percentChange2019to2022),
      color = percentChange2019to2022
    ),
    alpha = 0.8
  ) +
  scale_size_continuous(range = c(3, 14)) +
  scale_color_gradient2(
    low = "steelblue",
    mid = "white",
    high = "red3",
    midpoint = 0,
    name = "",
    breaks = c(
      min(Theft_Map$percentChange2019to2022, na.rm = TRUE),
      max(Theft_Map$percentChange2019to2022, na.rm = TRUE)
    ),
    labels = c("−", "+")
```

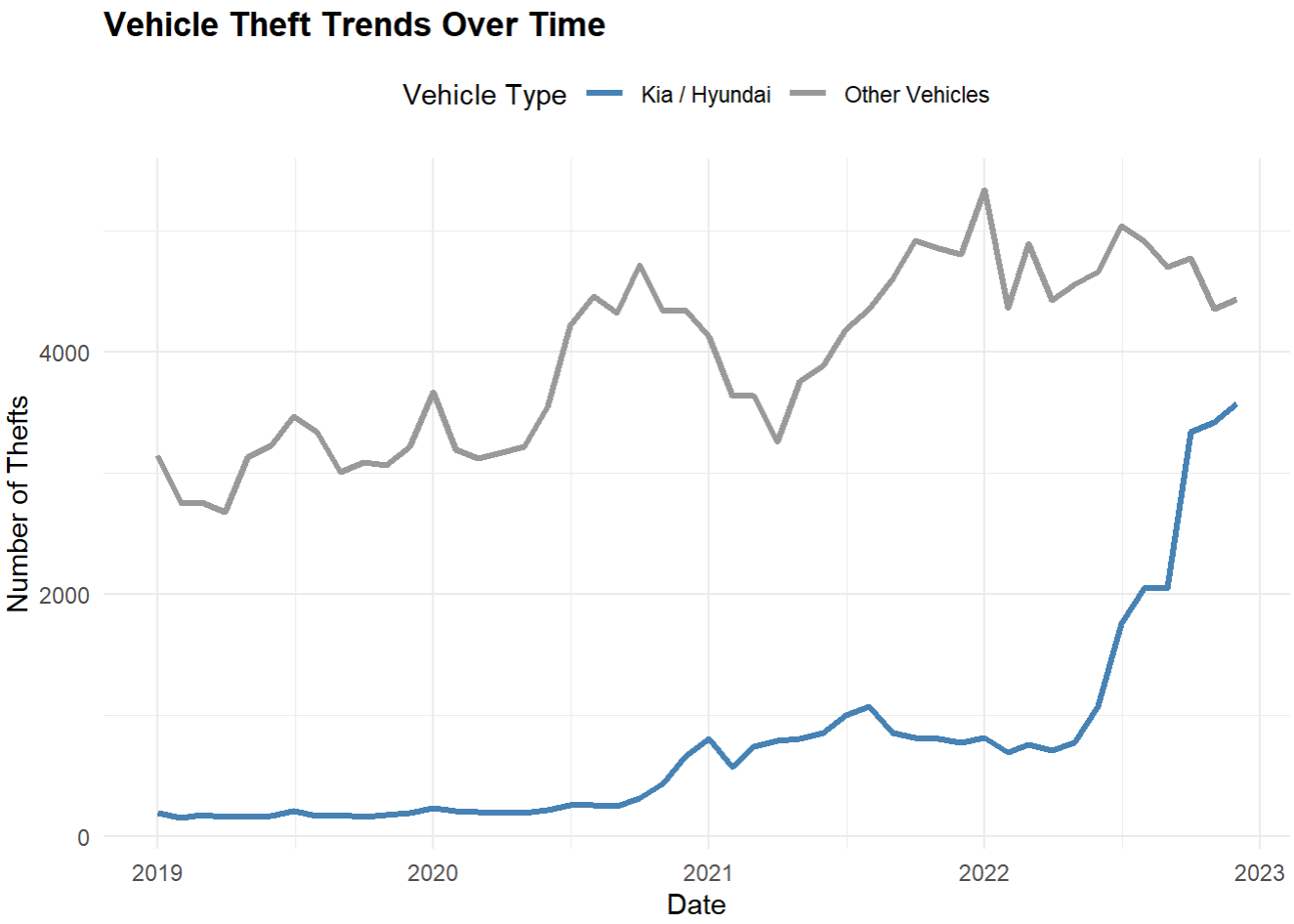
```
) +
coord_quickmap() +
labs(
  title = "Rates of Theft Are Increasing Faster in the East than the West"
) +
guides(size = "none") +
theme_minimal() +
theme(
  axis.title = element_blank(),
  axis.text  = element_blank(),
  axis.ticks = element_blank(),
  panel.grid = element_blank(),
  plot.title = element_text(face = "bold"),
  legend.position = c(0.89, 0.12),
  legend.text = element_text(size = 14),
  legend.justification = c("left", "bottom")
)
```



Line Graph of not sure what yet

```
library(dplyr)
library(ggplot2)

# Ensure Date is Date type
Theft_by_city$Date <- as.Date(Theft_by_city$Date)
```

```r
# Aggregate totals by Date
theft_time_series <- Theft_by_city %>%
  group_by(Date) %>%
  summarise(
    KiaHyundai = sum(countKiaHyundaiThefts, na.rm = TRUE),
    OtherVehicles = sum(countOtherThefts, na.rm = TRUE),
    .groups = "drop"
  )

# Plot line graph
ggplot(theft_time_series, aes(x = Date)) +
  geom_line(aes(y = KiaHyundai, color = "Kia / Hyundai"), linewidth = 1.2) +
  geom_line(aes(y = OtherVehicles, color = "Other Vehicles"), linewidth = 1.2) +
  scale_color_manual(
    values = c(
      "Kia / Hyundai" = "steelblue",
      "Other Vehicles" = "gray60"
    )
  ) +
  labs(
    title = "Vehicle Theft Trends Over Time",
    y = "Number of Thefts",
    color = "Vehicle Type"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(face = "bold"),
    legend.position = "top"
  )
```



Which Cities increased the most?

```
library(dplyr)
library(ggplot2)

# Ensure Date is Date type
Theft_by_city$Date <- as.Date(Theft_by_city$Date)

# Create Year
Theft_by_city$Year <- as.integer(format(Theft_by_city$Date, "%Y"))

# Aggregate Kia thefts by City + Year
city_year_kia <- Theft_by_city %>%
  group_by(city, Year) %>%
  summarise(
    KiaThefts = sum(countKiaHyundaiThefts, na.rm = TRUE),
    .groups = "drop"
  )

# Calculate change (last year - first year) for each city
city_change <- city_year_kia %>%
  group_by(city) %>%
  summarise(
    Kia_Change = KiaThefts[Year == max(Year)] - KiaThefts[Year == min(Year)],
    .groups = "drop"
  ) %>%
```

```
  filter(Kia_Change > 0) %>%          # keep only increases
  arrange(desc(Kia_Change)) %>%
  slice_head(n = 10)

# Plot
ggplot(city_change, aes(x = reorder(city, Kia_Change), y = Kia_Change)) +
  geom_col(fill = "steelblue") +
  coord_flip() +
  labs(
    title = "Top 10 Cities with the Largest Increase in Kia/Hyundai Thefts",
    x = "",
    y = "Increase in Kia/Hyundai Thefts"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(face = "bold")
  )
```