PHE States EDA

Ryan Mersereau

2026-02-27

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.5.2

## Warning: package 'ggplot2' was built under R version 4.5.1

## Warning: package 'tidyr' was built under R version 4.5.2

## Warning: package 'readr' was built under R version 4.5.1

## Warning: package 'purrr' was built under R version 4.5.2

## Warning: package 'stringr' was built under R version 4.5.2

## Warning: package 'forcats' was built under R version 4.5.2

## Warning: package 'lubridate' was built under R version 4.5.2

## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.1     v stringr   1.6.0
## v ggplot2   3.5.2     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.2
## v purrr     1.2.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(dplyr)
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.5.1
```

```
# Reading in data
orders <- read_excel("C:/Users/gemer/Downloads/Finalordersstates.xlsx")
```

```r
#Converting date objects and adding time to delivery variable
orders$dateAndTime <- ymd_hms(orders$dateAndTime, tz = "UTC")
```

```
## Warning: 227 failed to parse.
```

```r
orders$dateAndTime <- as.Date(orders$dateAndTime)
orders$dateAndTime <- format(orders$dateAndTime, format = "%m/%d/%Y")

orders <- orders %>%
  rename(arrival_date = dateAndTime)

orders$arrival_date <- as.Date(orders$arrival_date, format = "%m/%d/%Y")
orders$shipping_date <- as.Date(orders$shipping_date, format = "%m/%d/%Y")

orders <- orders %>%
  mutate(time_to_delivery = difftime(orders$arrival_date, orders$shipping_date,
                                     units = "days"))

orders$time_to_delivery <- as.integer(orders$time_to_delivery)

orders <- drop_na(orders)
```

```r
#Creating table for state data
statesfreq <- orders %>%
  group_by(Ship_State, Ship_Country) %>%
  rename(state = Ship_State) %>%
  summarise(count = n(),
            Mean_delivery_time = mean(time_to_delivery))
```

```
## `summarise()` has grouped output by 'state'. You can override using the
## `.groups` argument.
```

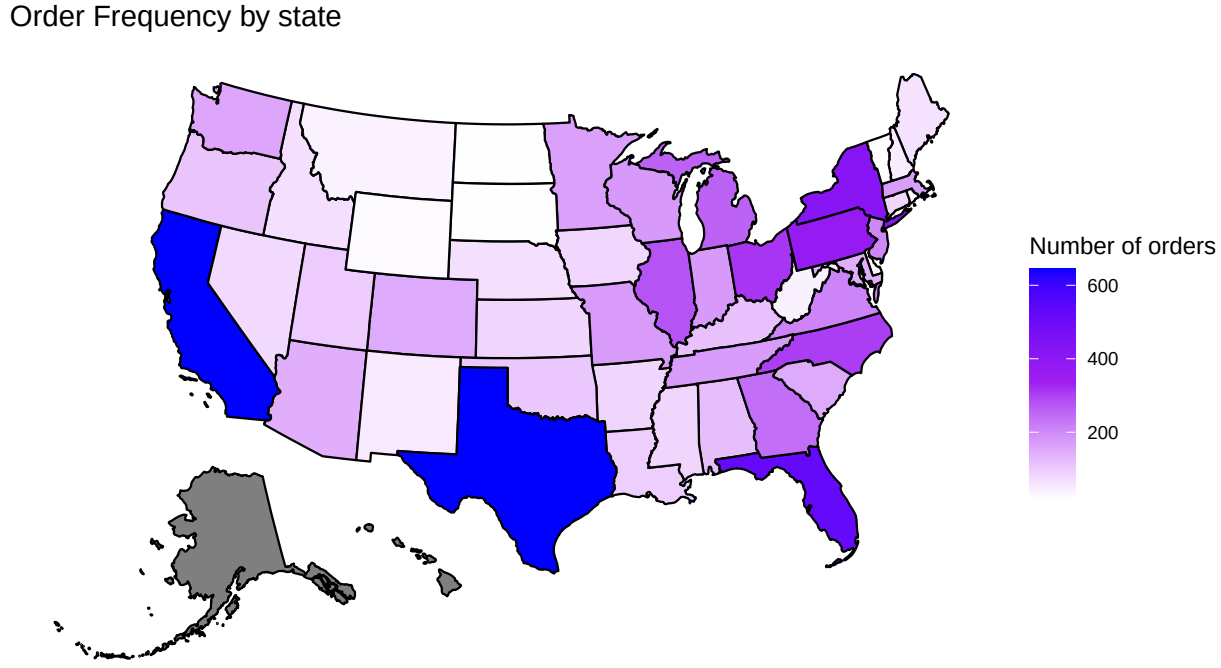
```r
statesfreq
```

```
## # A tibble: 49 x 4
## # Groups:   state [49]
##    state Ship_Country count Mean_delivery_time
##    <chr> <chr>        <int>              <dbl>
##  1 AL    US             122               4.30
##  2 AR    US              84               5.95
##  3 AZ    US             146               6.08
##  4 CA    US             645               6.89
##  5 CO    US             153               6.75
##  6 CT    US              84               5.01
##  7 DC    US              20               3.75
##  8 DE    US              35               3.71
##  9 FL    US             523               3.84
## 10 GA    US             243               4.13
## # i 39 more rows
```

```
#US Delivery frequency heatmap
library(usmap)
```

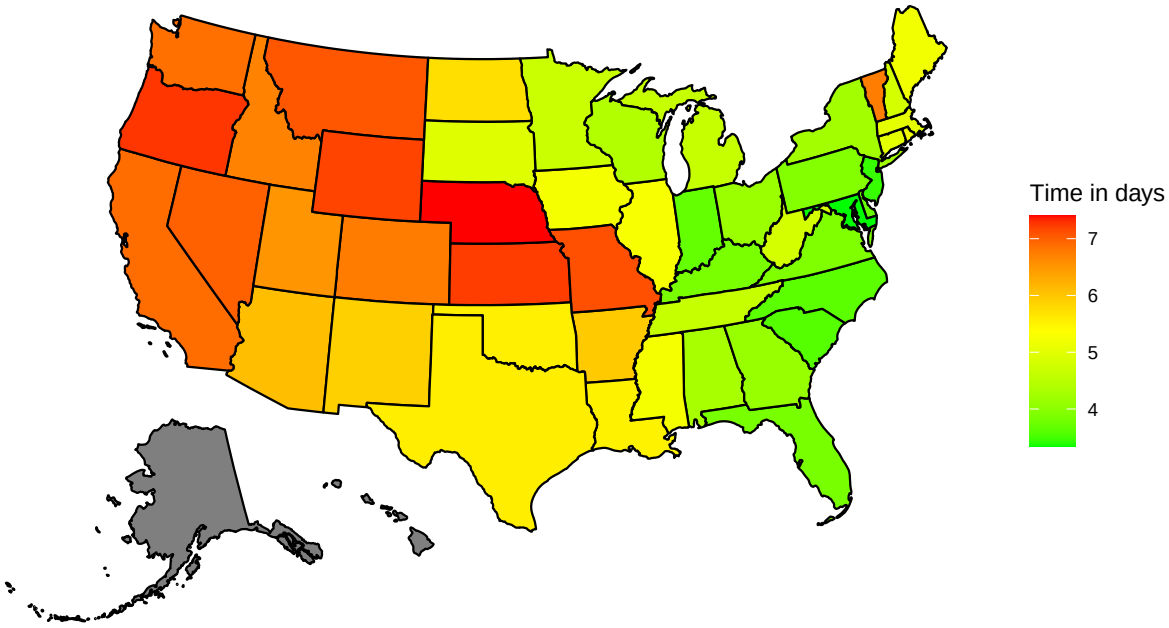
```
## Warning: package 'usmap' was built under R version 4.5.2
```

```
library(ggplot2)

plot_usmap(data = statesfreq, values = "count", regions = "states") +
  scale_fill_gradientn(colours = c("white", "purple", "blue"),
                       name = "Number of orders", label = scales::comma) +
  labs(title = "Order Frequency by state") +
  theme(legend.position = "right")
```




```
#US Mean delivery time heatmap

plot_usmap(data = statesfreq, values = "Mean_delivery_time", regions = "states") +
  scale_fill_gradientn(colours = c("green", "yellow", "red"),
                       name = "Time in days", label = scales::comma) +
  labs(title = "Mean Delivery Time of US Orders") +
  theme(legend.position = "right")
```

Mean Delivery Time of US Orders




```
view(countypov)
```