Notebook

2025-01-18

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.2
## -- 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(ggplot2)
library(dplyr)
```

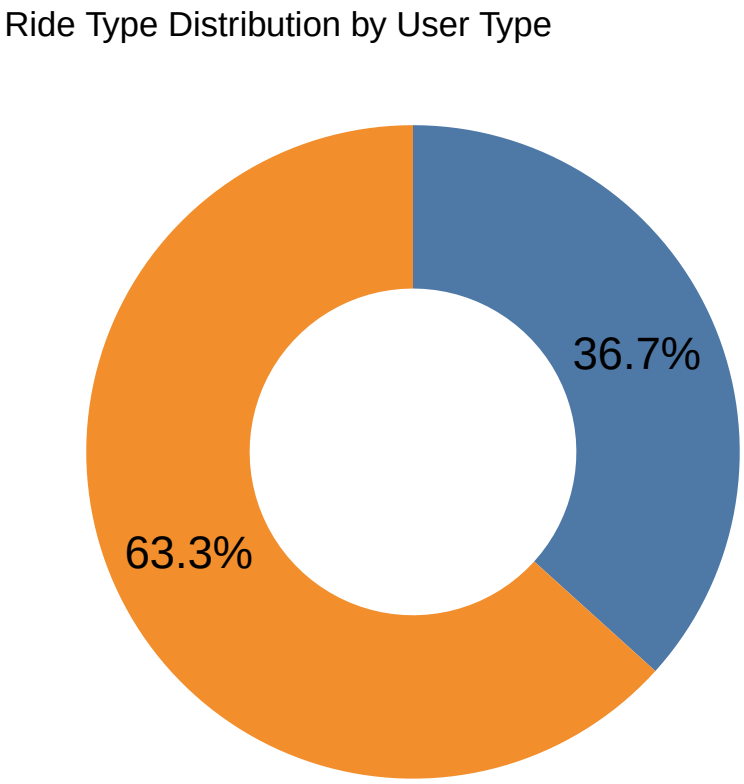
```
january <- read.csv("datasets/202401-divvy-tripdata.csv")
february <- read.csv("datasets/202402-divvy-tripdata.csv")
march <- read.csv("datasets/202403-divvy-tripdata.csv")
april <- read.csv("datasets/202404-divvy-tripdata.csv")
may <- read.csv("datasets/202405-divvy-tripdata.csv")
june <- read.csv("datasets/202406-divvy-tripdata.csv")
july <- read.csv("datasets/202407-divvy-tripdata.csv")
august <- read.csv("datasets/202408-divvy-tripdata.csv")
september <- read.csv("datasets/202409-divvy-tripdata.csv")
october <- read.csv("datasets/202410-divvy-tripdata.csv")
november <- read.csv("datasets/202411-divvy-tripdata.csv")
december <- read.csv("datasets/202412-divvy-tripdata.csv")
```

```
all_data <- bind_rows(
    january |> mutate(month = 1),
    february |> mutate(month = 2),
    march |> mutate(month = 3),
    april |> mutate(month = 4),
    may |> mutate(month = 5),
    june |> mutate(month = 6),
    july |> mutate(month = 7),
    august |> mutate(month = 8),
    september |> mutate(month = 9),
    october |> mutate(month = 10),
    november |> mutate(month = 11),
    december |> mutate(month = 12)
)
```

```
sample_data <- all_data |> sample_n(1000)
```

```r
ride_type_by_user <- sample_data |>
    group_by(member_casual) |>
    summarise(count = n()) |>
    mutate(percentage = count / sum(count) * 100,
           label = paste0(round(percentage, 2), "%"))

# Create the donut chart
ride_type_by_user |>
    ggplot(aes(ymax = cumsum(percentage / 100), ymin = c(0, head(cumsum(percentage / 100), -1)), xmax =
    geom_rect() +
    geom_label(aes(y = (cumsum(percentage / 100) + c(0, head(cumsum(percentage / 100), -1))) / 2, label
    coord_polar(theta = "y") +
    xlim(c(2, 4)) +
    scale_fill_manual(values = c("member" = "#F28E2B", "casual" = "#4E79A7")) +
    labs(title = "Ride Type Distribution by User Type", fill = "User Type") +
    theme_void() +
    theme(legend.position = "none")
```




```r
ride_type_by_rideable <- sample_data |>
    group_by(rideable_type) |>
    summarise(count = n())
ride_type_by_rideable
```

```
## # A tibble: 3 x 2
##   rideable_type    count
##   <chr>            <int>
```

```
## 1 classic_bike       472
## 2 electric_bike      500
## 3 electric_scooter    28
```

```
average_ride_length <- sample_data |>
    mutate(ride_length = as.numeric(difftime(ended_at, started_at, units = "mins"))) |>
    group_by(member_casual) |>
    summarise(average_ride_length = mean(ride_length, na.rm = TRUE))
average_ride_length
```

```
## # A tibble: 2 x 2
##   member_casual average_ride_length
##   <chr>                       <dbl>
## 1 casual                       20.2
## 2 member                       11.4
```

```
ride_length_by_weekday <- sample_data |>
    mutate(ride_length = as.numeric(difftime(ended_at, started_at, units = "mins")),
           day_of_week = weekdays(as.Date(started_at))) |>
    group_by(day_of_week) |>
    summarise(average_ride_length = mean(ride_length, na.rm = TRUE))
ride_length_by_weekday
```

```
## # A tibble: 7 x 2
##   day_of_week average_ride_length
##   <chr>                     <dbl>
## 1 Friday                     13.6
## 2 Monday                     14.2
## 3 Saturday                   16.8
## 4 Sunday                     16.6
## 5 Thursday                   18.6
## 6 Tuesday                    11.1
## 7 Wednesday                  11.7
```

```
total_rides_by_weekday <- sample_data |>
    mutate(day_of_week = weekdays(as.Date(started_at))) |>
    group_by(day_of_week) |>
    summarise(total_rides = n())
total_rides_by_weekday
```

```
## # A tibble: 7 x 2
##   day_of_week total_rides
##   <chr>             <int>
## 1 Friday              146
## 2 Monday              136
## 3 Saturday            154
## 4 Sunday              139
## 5 Thursday            138
## 6 Tuesday             135
## 7 Wednesday           152
```

```
total_rides_by_hour <- sample_data |>
    mutate(hour = hour(started_at)) |>
    group_by(hour) |>
    summarise(total_rides = n())
total_rides_by_hour
```

```
## # A tibble: 24 x 2
##     hour total_rides
##    <int>       <int>
##  1     0          13
##  2     1           6
##  3     2           1
##  4     3           4
##  5     4           2
##  6     5           6
##  7     6          20
##  8     7          42
##  9     8          59
## 10     9          49
## # i 14 more rows
```

```
total_rides_by_month <- sample_data |>
    mutate(month = month(started_at, label = TRUE)) |>
    group_by(month) |>
    summarise(total_rides = n())
total_rides_by_month
```

```
## # A tibble: 12 x 2
##    month total_rides
##    <ord>       <int>
##  1 Jan            21
##  2 Feb            41
##  3 Mar            64
##  4 Apr            71
##  5 May            98
##  6 Jun           109
##  7 Jul           136
##  8 Aug           133
##  9 Sep           156
## 10 Oct            93
## 11 Nov            47
## 12 Dec            31
```