Lin_Plotting

Frances Lin

3/10/2021

```
# Load packages
library(tidyverse)
```

```
## -- Attaching packages --------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   1.0.2
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.4.0

## Warning: package 'readr' was built under R version 3.6.2

## Warning: package 'dplyr' was built under R version 3.6.2

## -- Conflicts ------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

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

```
# Read data for Q1 avg fare amount
yellow_20 <- read.csv("2020_yellow_mon_amount.csv")
yellow_19 <- read.csv("2019_yellow_mon_amount.csv")
green_20 <- read.csv("2020_green_mon_amount.csv")
green_19 <- read.csv("2019_green_mon_amount.csv")

# Read data for Q2 total count of license
hfhv_20 <- read.csv("2020_hfhv_license.csv")
hfhv_19 <- read.csv("2019_hfhv_license.csv")

# Read data for Q3 top 6 drop-off location
yellow_20_DO <- read.csv("2020_yellow_DOLocation.csv")
yellow_19_DO <- read.csv("2019_yellow_DOLocation.csv")
green_20_DO <- read.csv("2020_green_DOLocation.csv")
green_19_DO <- read.csv("2019_green_DOLocation.csv")
fhv_20_DO <- read.csv("2020_fhv_DOLocation.csv")
fhv_19_DO <- read.csv("2019_fhv_DOLocation.csv")
hfhv_20_DO <- read.csv("2020_hfhv_DOLocation.csv")
hfhv_19_DO <- read.csv("2019_hfhv_DOLocation.csv")
```

```
# View data for Q1
head(yellow_20)
```

```
##   pickup_mon mon_amount
## 1          1   18.60099
## 2          2   18.55870
## 3          3   18.50115
## 4          4   16.42109
## 5          5   18.45184
## 6          6   18.76602
```

```
# View data for Q2
head(hfhv_20)
```

```
##   license count_license
## 1    Uber     103112054
## 2    Lyft      37250101
## 3     Via       2872556
```

```
# View data for Q3
head(yellow_20_DO)
```

```
##   DOLocationID DOLocationID_count
## 1          236            1119163
## 2          237            1008712
## 3          161             837123
## 4          170             731856
## 5          141             681651
## 6          142             665673
```

```
# Create a dataframe for Q1
df <- tibble(
  pickup_mon = rep(yellow_20$pickup_mon, 4),
  mon_amount = c(yellow_20$mon_amount, yellow_19$mon_amount, green_20$mon_amount, green_19$mon_amount),
  year = c(rep(2020, 12), rep(2019, 12), rep(2020, 12), rep(2019,12)),
  type = c(rep("yellow", 12), rep("yellow", 12), rep("green", 12), rep("green",12))
  )
head(df)
```

```
## # A tibble: 6 x 4
##   pickup_mon mon_amount  year type
##        <int>      <dbl> <dbl> <chr>
## 1          1       18.6  2020 yellow
## 2          2       18.6  2020 yellow
## 3          3       18.5  2020 yellow
## 4          4       16.4  2020 yellow
## 5          5       18.5  2020 yellow
## 6          6       18.8  2020 yellow
```

```
# Juno is no longer there in 2020
# Add to dataframe for plotting purposes
hfhv_20 <- hfhv_20 %>% add_row(license = "Juno", count_license = 0)
hfhv_20
```

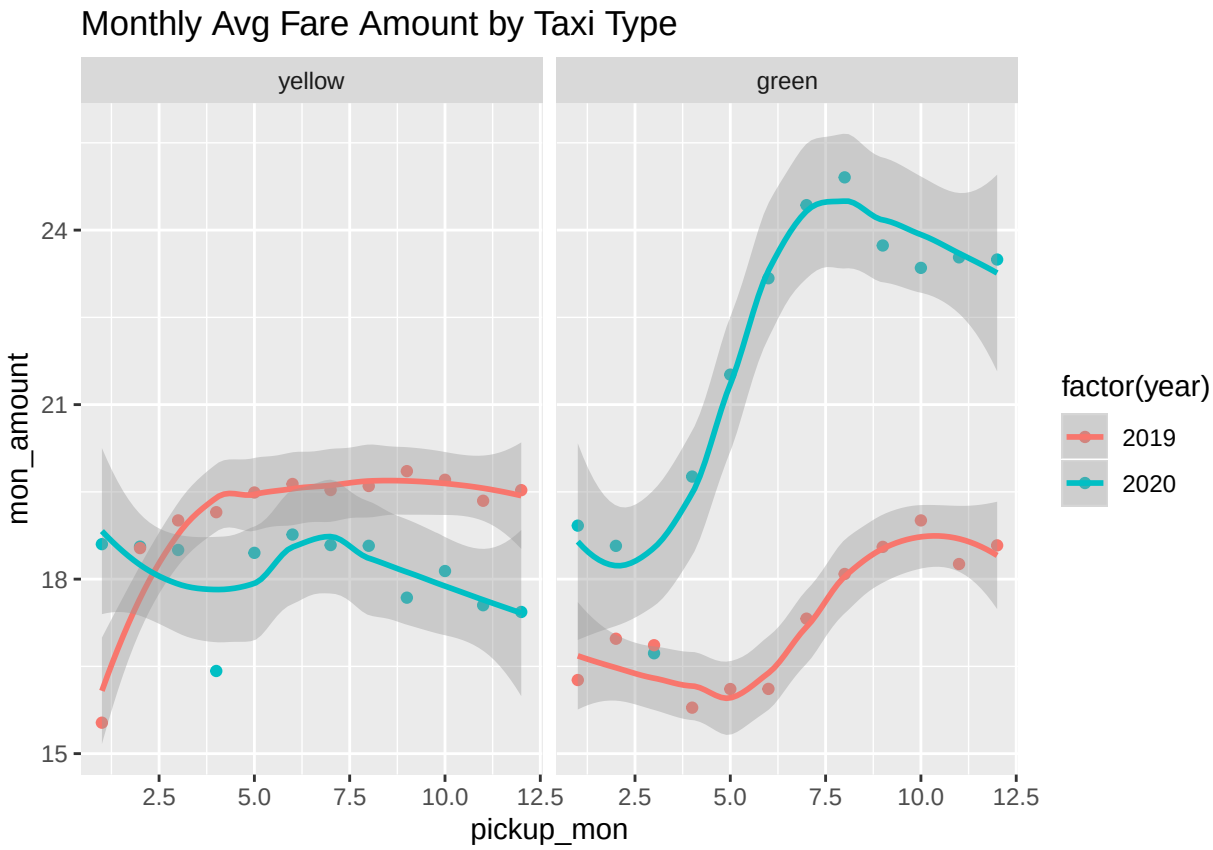
```
##   license count_license
## 1    Uber     103112054
## 2    Lyft      37250101
## 3     Via       2872556
## 4    Juno             0
```

```
# Create a dataframe for Q2
df_license <- tibble(
  license = rep(hfhv_20$license, 2),
  count_license = c(hfhv_20$count_license, hfhv_19$count_license),
  year = c(rep("2020", 4), rep("2019", 4))
)
head(df_license)
```

```
## # A tibble: 6 x 3
##   license count_license year
##   <fct>           <dbl> <chr>
## 1 Uber        103112054 2020
## 2 Lyft         37250101 2020
## 3 Via           2872556 2020
## 4 Juno                0 2020
## 5 Uber        164844505 2019
## 6 Lyft         53275098 2019
```

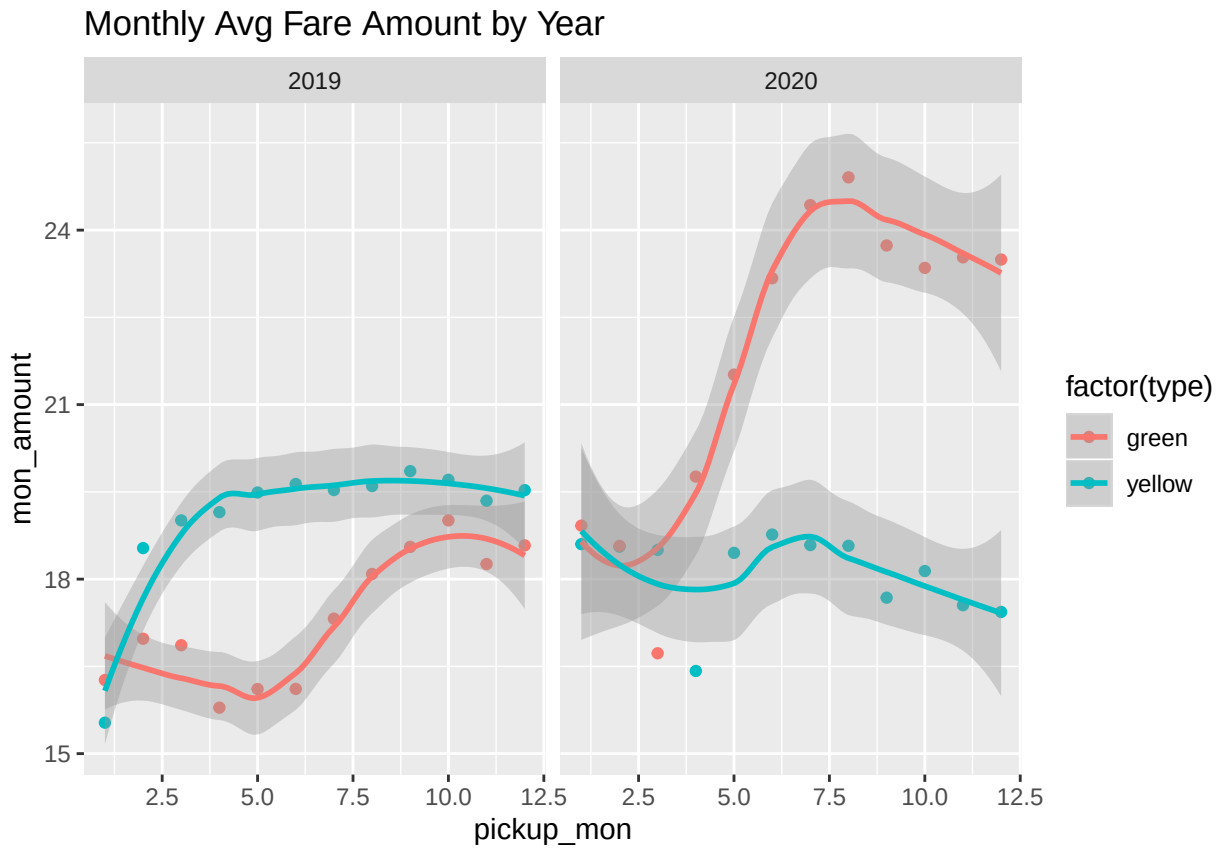
```
# Plot data for Q1
p <- ggplot(df, aes(x=pickup_mon, y=mon_amount, color = factor(year))) + geom_point() + geom_smooth()
p + facet_wrap(~factor(type, levels = c("yellow", "green")), ncol = 2) +
  ggtitle("Monthly Avg Fare Amount by Taxi Type")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

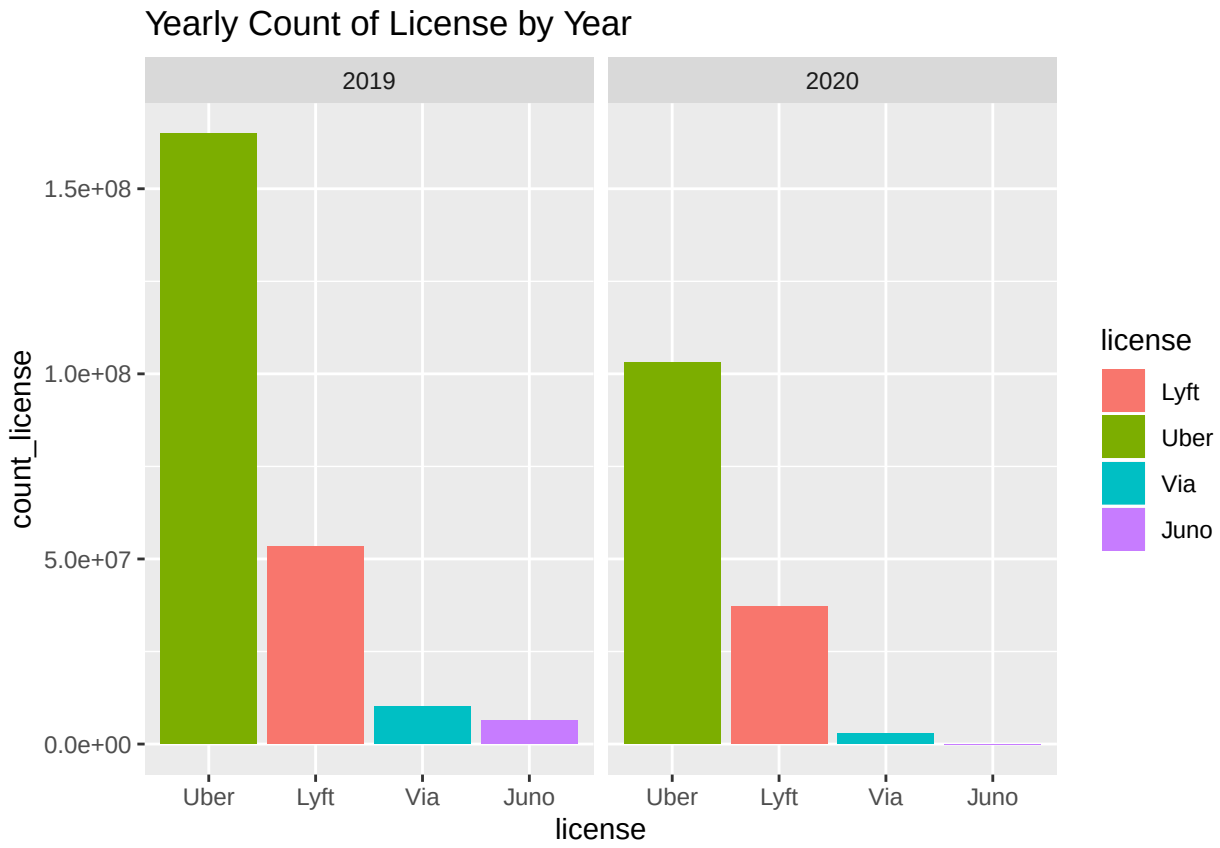



```
# Plot data for Q1
p <- ggplot(df, aes(x=pickup_mon, y=mon_amount, color = factor(type))) + geom_point() + geom_smooth()
p + facet_wrap(~factor(year), ncol = 2) +
  ggtitle("Monthly Avg Fare Amount by Year")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```




```
# Plot data for Q2
p_license <- ggplot(df_license, aes(x=factor(license, levels = c("Uber", "Lyft", "Via", "Juno")), y=cou
  geom_bar(stat = "identity") +
  xlab("license")
p_license + facet_wrap(~factor(year), ncol = 2) +
  ggtitle("Yearly Count of License by Year")
```




```
# Join dataframes by year for Q3
df_location_20 <- rbind(yellow_20_DO, green_20_DO, fhv_20_DO, hfhv_20_DO)
df_location_20 <- df_location_20 %>% add_column(
  type = c(rep("yellow", 6), rep("green", 6), rep("fhv", 6), rep("hfhv", 6)),
  year = c(rep(2020, length(df_location_20$DOLocationID)))
)
df_location_19 <- rbind(yellow_19_DO, green_19_DO, fhv_19_DO, hfhv_19_DO)
df_location_19 <- df_location_19 %>% add_column(
  type = c(rep("yellow", 6), rep("green", 6), rep("fhv", 6), rep("hfhv", 6)),
  year = c(rep(2019, length(df_location_19$DOLocationID)))
)

# Join all dataframes
df_location <- rbind(df_location_20, df_location_19)
head(df_location)
```

```
##   DOLocationID DOLocationID_count   type year
## 1          236            1119163 yellow 2020
## 2          237            1008712 yellow 2020
## 3          161             837123 yellow 2020
## 4          170             731856 yellow 2020
## 5          141             681651 yellow 2020
## 6          142             665673 yellow 2020
```

```
# Read data for Q3
zone <- read.csv("zone_lookup.csv")
```

```r
#rename().. doesnt work for some reasons
names(zone)[names(zone) == "LocationID"] <- "DOLocationID" #rename col name
zone <- as.tibble(zone)
```

```
## Warning: `as.tibble()` is deprecated, use `as_tibble()` (but mind the new semantics).
## This warning is displayed once per session.
```

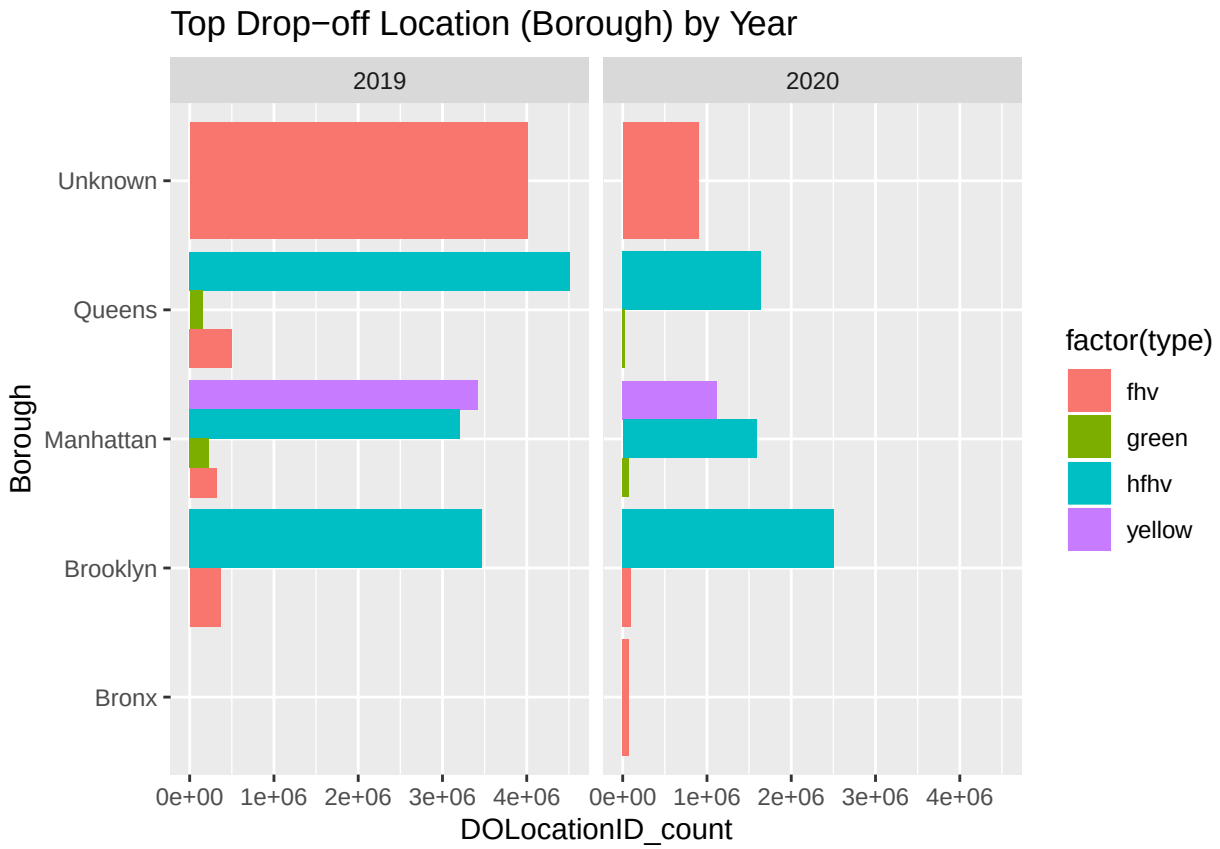
```r
head(zone)
```

```
## # A tibble: 6 x 4
##   DOLocationID Borough       Zone                     service_zone
##          <int> <fct>         <fct>                    <fct>
## 1            1 EWR           Newark Airport           EWR
## 2            2 Queens        Jamaica Bay              Boro Zone
## 3            3 Bronx         Allerton/Pelham Gardens  Boro Zone
## 4            4 Manhattan     Alphabet City            Yellow Zone
## 5            5 Staten Island Arden Heights            Boro Zone
## 6            6 Staten Island Arrochar/Fort Wadsworth  Boro Zone
```

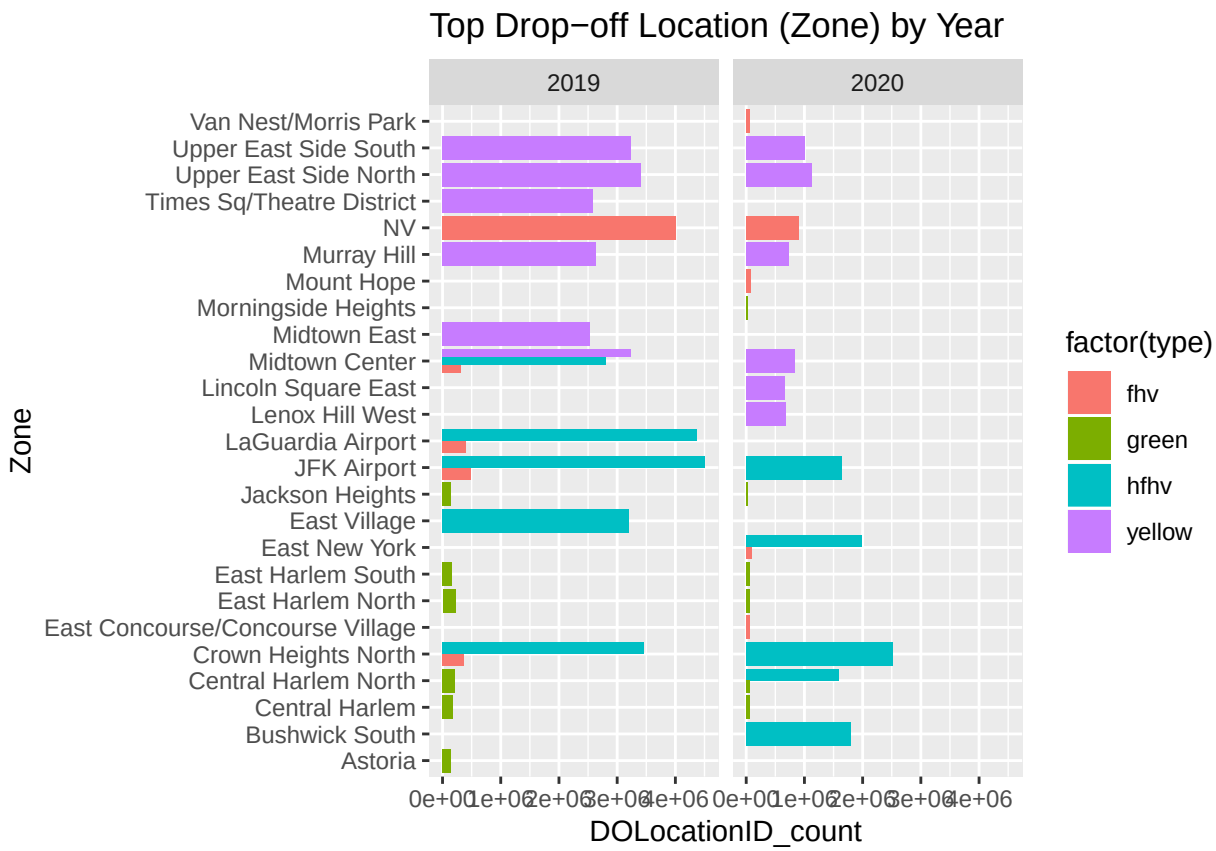
```r
# Join dataframes using "DOLocationID" as key
df_join <- full_join(df_location, zone, by = "DOLocationID")
df_join <- df_join %>% drop_na()
head(df_join)
```

```
##   DOLocationID DOLocationID_count   type year   Borough
## 1          236            1119163 yellow 2020 Manhattan
## 2          237            1008712 yellow 2020 Manhattan
## 3          161             837123 yellow 2020 Manhattan
## 4          170             731856 yellow 2020 Manhattan
## 5          141             681651 yellow 2020 Manhattan
## 6          142             665673 yellow 2020 Manhattan
##                     Zone service_zone
## 1 Upper East Side North  Yellow Zone
## 2 Upper East Side South  Yellow Zone
## 3        Midtown Center  Yellow Zone
## 4           Murray Hill  Yellow Zone
## 5       Lenox Hill West  Yellow Zone
## 6   Lincoln Square East  Yellow Zone
```

```r
# Plot data for Q3
p_Borough <- ggplot(df_join, aes(x = Borough, y = DOLocationID_count, fill = factor(type))) +
  geom_bar(stat = "identity", position=position_dodge()) +
  coord_flip()
p_Borough + facet_wrap(~factor(year), ncol = 2) +
  ggtitle("Top Drop-off Location (Borough) by Year")
```




```
# Plot data for Q3
p_Zone <- ggplot(df_join, aes(x = Zone, y = DOLocationID_count, fill = factor(type))) +
  geom_bar(stat = "identity", position=position_dodge()) +
  coord_flip()
p_Zone + facet_wrap(~factor(year), ncol = 2) +
  ggtitle("Top Drop-off Location (Zone) by Year")
```




```
# Analysis for Q1
df_green <- df[which(df$type == "green"), ]
head(df_green)
```

```
## # A tibble: 6 x 4
##   pickup_mon mon_amount  year type
##        <int>      <dbl> <dbl> <chr>
## 1          1       18.9  2020 green
## 2          2       18.6  2020 green
## 3          3       16.7  2020 green
## 4          4       19.8  2020 green
## 5          5       21.5  2020 green
## 6          6       23.2  2020 green
```

```
max(df_green$mon_amount) - min(df_green$mon_amount)
```

```
## [1] 9.116099
```