Question2

2024-04-06

```
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(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(ggplot2)
library(tidyr)

# Read the CSV file
trashwheel_data <- read.csv("data/trashwheel.csv")
```

To delve deeper into the analysis of waste collection efficiency and effectiveness, we calculate the total weight and volume of waste collected within a year. This step is crucial for understanding the scale of waste management efforts and their year-on-year progression.

- **New variables:**
  - `TotalWeight` and `TotalVolume` calculated by taking sum of weight and volume within a year for each Trash Wheel.
  - `Uses` calculated by number of uses for each Trash Wheel over Years.

```r
#  Data segmentation by Trash Wheel by Year
trashwheel_segments <- trashwheel_data %>%
  group_by(Name, Year) %>%
  summarise(
    Uses = n(),
    AverageWeight = mean(Weight, na.rm = TRUE),
    AverageVolume = mean(Volume, na.rm = TRUE),
    TotalWeight = sum(Weight, na.rm = TRUE),
    TotalVolume = sum(Volume, na.rm = TRUE)
  )
```

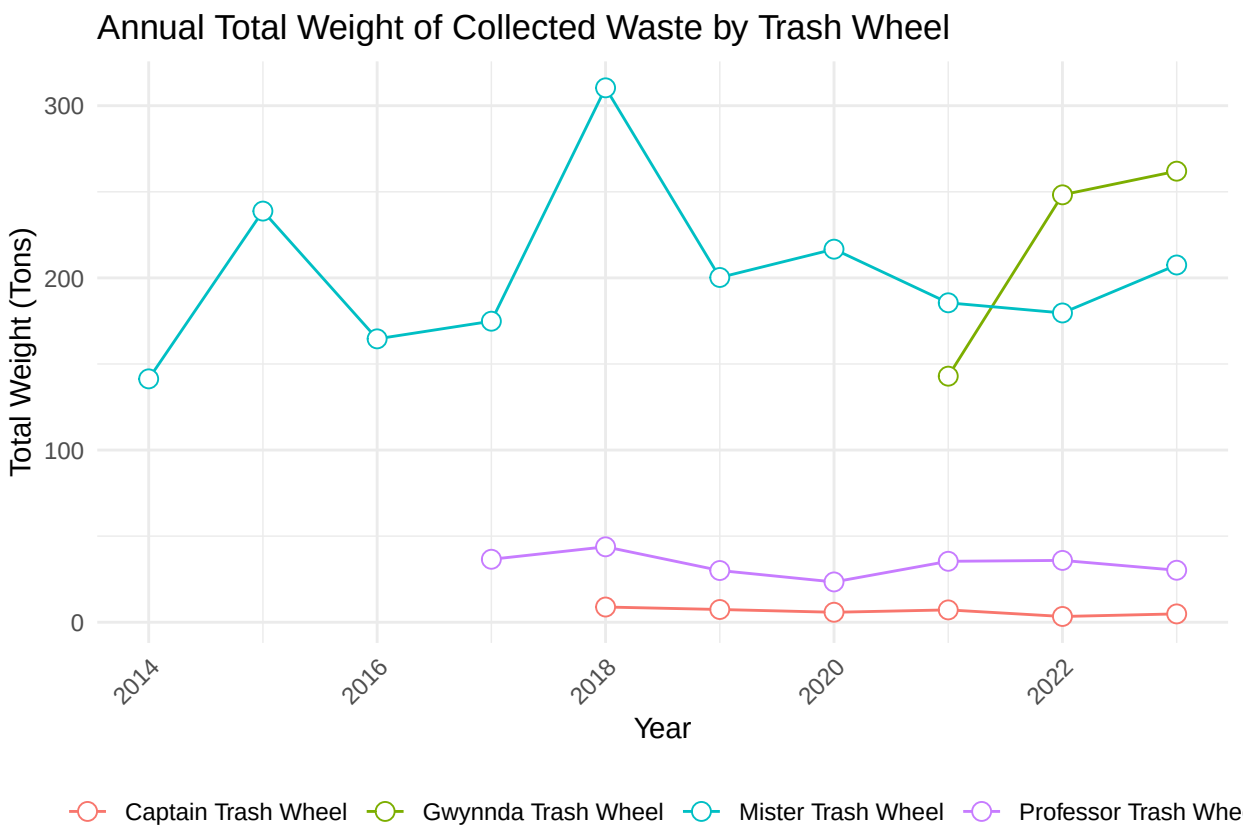
```
## `summarise()` has grouped output by 'Name'. You can override using the
## `.groups` argument.
```

```r
trashwheel_segments
```

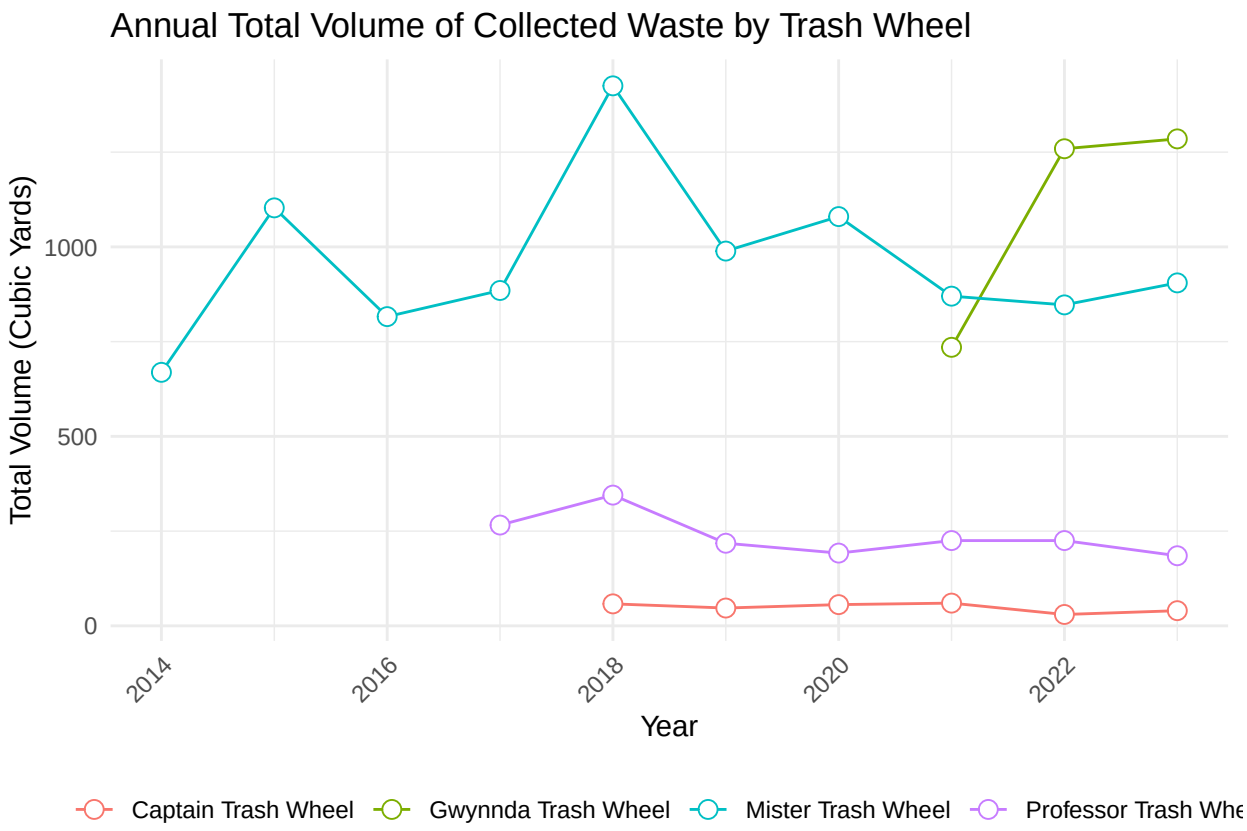
```
## # A tibble: 26 x 7
## # Groups:   Name [4]
##    Name           Year  Uses AverageWeight AverageVolume TotalWeight TotalVolume
##    <chr>         <int> <int>         <dbl>         <dbl>       <dbl>       <int>
##  1 Captain Tras~  2018     6         1.47           9.67        8.83          58
##  2 Captain Tras~  2019     5         1.48           9.4         7.39          47
##  3 Captain Tras~  2020     6         0.965          9.33        5.79          56
##  4 Captain Tras~  2021     6         1.19          10           7.16          60
##  5 Captain Tras~  2022     3         1.11          10           3.34          30
##  6 Captain Tras~  2023     4         1.21          10           4.84          40
##  7 Gwynnda Tras~  2021    49         2.92          15         143.           735
##  8 Gwynnda Tras~  2022    85         2.92          14.8       248.          1259
##  9 Gwynnda Tras~  2023    86         3.05          14.9       262.          1285
## 10 Mister Trash~  2014    44         3.21          15.2       141.           669
## # i 16 more rows
```

Now we plot the line graph to visualize the performance in terms of `TotalWeight` and `TotalVolume` of trash wheel over years

```r
# Time-Series Analysis: Performance of each Trash Wheel over time
ggplot(trashwheel_segments, aes(x = Year, y = TotalWeight, group = Name, color = Name)) +
  geom_line() +
  geom_point(size = 3, shape = 21, fill = "white") + # Using filled points for better visibility
  theme_minimal() +
  labs(title = "Annual Total Weight of Collected Waste by Trash Wheel",
       x = "Year", y = "Total Weight (Tons)") +
  theme(legend.title = element_blank(), # Removing the legend title for clarity
        legend.position = "bottom", # Positioning legend at the bottom for easy comparison
        axis.text.x = element_text(angle = 45, hjust = 1))
```




```r
ggplot(trashwheel_segments, aes(x = Year, y = TotalVolume, group = Name, color = Name)) +
  geom_line() +
  geom_point(size = 3, shape = 21, fill = "white") + # Using filled points for better visibility
  theme_minimal() +
  labs(title = "Annual Total Volume of Collected Waste by Trash Wheel",
       x = "Year", y = "Total Volume (Cubic Yards)") +
  theme(legend.title = element_blank(), # Removing the legend title for clarity
        legend.position = "bottom", # Positioning legend at the bottom for easy comparison
        axis.text.x = element_text(angle = 45, hjust = 1))
```



Now, in order to plot a line graph to illustrate the performance of each Trash Wheel over time, highlighting trends and year-on-year changes in waste collection, we will need to calculate the rate of change in both weight and volume.

- **New Variables:** `Yearly Performance Growth Rate`, calculated as the percentage change in `Weight` and `Volume` from one year to the next, to quantify performance improvements or declines.

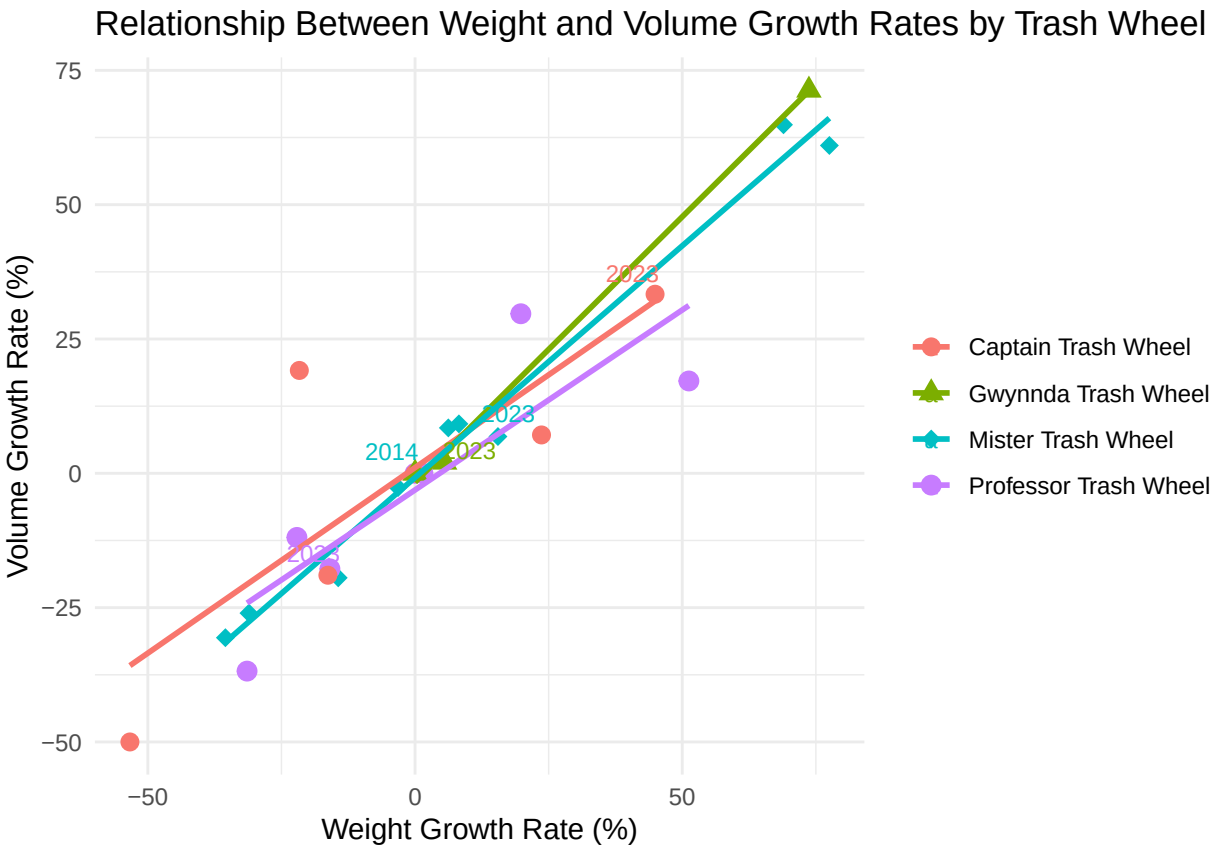
```
# Calculating Yearly Performance Growth Rate
trashwheel_segments <- trashwheel_segments %>%
  group_by(Name) %>%
  arrange(Year) %>%
  mutate(
    WeightGrowthRate = (TotalWeight / lag(TotalWeight) - 1) * 100, # Growth weight rate
    VolumeGrowthRate = (TotalVolume / lag(TotalVolume) - 1) * 100 # Growth volume rate
  ) %>%
  replace_na(list(WeightGrowthRate = 0, VolumeGrowthRate = 0)) %>%  # means that at the time when a tra
  arrange(desc(WeightGrowthRate), desc(VolumeGrowthRate))
trashwheel_segments
```

```
## # A tibble: 26 x 9
## # Groups:   Name [4]
##    Name           Year  Uses AverageWeight AverageVolume TotalWeight TotalVolume
##    <chr>         <int> <int>         <dbl>         <dbl>       <dbl>       <int>
##  1 Mister Trash~  2018    93          3.34          15.3      310.          1425
##  2 Gwynnda Tras~  2022    85          2.92          14.8      248.          1259
##  3 Mister Trash~  2015    71          3.36          15.5      239.          1103
##  4 Professor Tr~  2021    15          2.36          15         35.4          225
```

```
##  5 Captain Tras~  2023     4          1.21          10          4.84          40
##  6 Captain Tras~  2021     6          1.19          10          7.16          60
##  7 Professor Tr~  2018    24          1.83          14.4       43.8          345
##  8 Mister Trash~  2023    62          3.35          14.6      207.           905
##  9 Mister Trash~  2020    72          3.01          15        217.          1080
## 10 Mister Trash~  2017    55          3.18          16.1      175.           885
## # i 16 more rows
## # i 2 more variables: WeightGrowthRate <dbl>, VolumeGrowthRate <dbl>
```

```r
library(ggplot2)
library(dplyr)
library(ggrepel)

# Assigning unique shapes to each Trash Wheel
shapes <- c(16, 17, 18, 19) # Circle, Triangle, Square, Diamond

ggplot(trashwheel_segments, aes(x = WeightGrowthRate, y = VolumeGrowthRate, color = Name)) +
  geom_point(aes(shape = Name), size = 3) +
  geom_smooth(method = "lm", se = FALSE) +
  geom_text_repel(aes(label = if_else(Year == max(Year) | Year == min(Year), as.character(Year), "")),
                  nudge_y = 0.05, size = 3,
                  segment.color = 'grey50') + # Add labels to first and last years for clarity
  scale_shape_manual(values = shapes) + # Apply the shapes
  theme_minimal() +
  labs(title = "Relationship Between Weight and Volume Growth Rates by Trash Wheel",
       x = "Weight Growth Rate (%)", y = "Volume Growth Rate (%)") +
  theme(legend.title = element_blank())
```

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



The graph demonstrates a largely linear correlation between the changes in weight and volume, indicating that an increase in weight tends to correspond with an increase in volume.

Given this observation, focusing on a single attribute could simplify our analysis. Nevertheless, we will continue to examine both attributes to uncover any potential interesting phenomena that might emerge.

We plot a line graph for comparison between growth rate in both `Weight` and `Volume` as follow:

```
library(ggplot2)

# Plotting Weight Growth Rate over Time for each Trash Wheel
ggplot(trashwheel_segments, aes(x = Year, y = WeightGrowthRate, group = Name)) +
  geom_line(aes(color = Name)) +
  geom_point(aes(color = Name)) +
  geom_smooth(se = FALSE, method = "loess", color = "gray40") + # Add a smooth trend line
  facet_wrap(~Name, scales = "free_y") + # Faceting by Trash Wheel
  theme_minimal() +
  labs(title = "Weight Growth Rate Over Time by Trash Wheel",
       x = "Year", y = "Weight Growth Rate (%)") +
  theme(legend.title = element_blank() ,
        axis.text.x = element_text(angle = 45, hjust = 1))
```

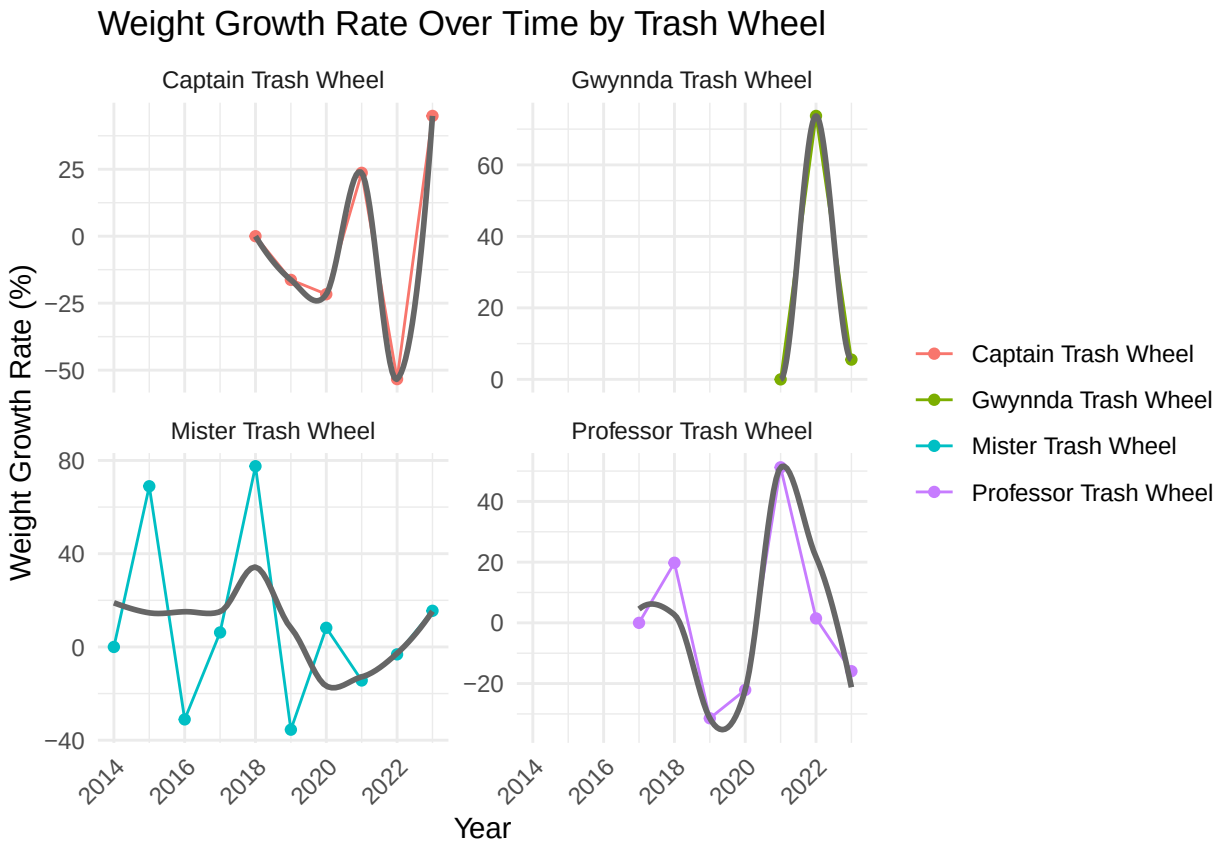
```
## `geom_smooth()` using formula = 'y ~ x'

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : span too small.  fewer data values than degrees of freedom.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
```

```
## : pseudoinverse used at 2021
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : neighborhood radius 1.01
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : reciprocal condition number 0
```

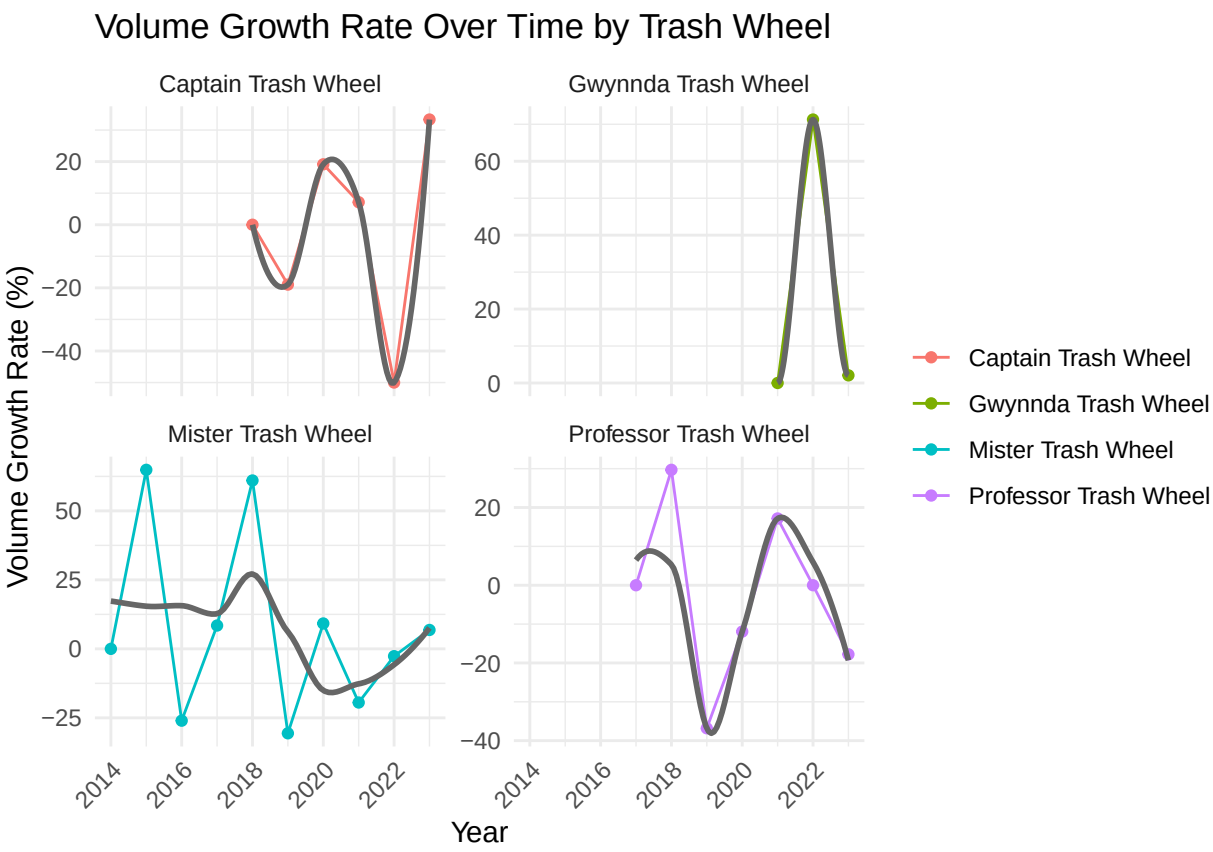
```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : There are other near singularities as well. 1.0201
```




```
# Plotting Volume Growth Rate over Time for each Trash Wheel
ggplot(trashwheel_segments, aes(x = Year, y = VolumeGrowthRate, group = Name)) +
  geom_line(aes(color = Name)) +
  geom_point(aes(color = Name)) +
  geom_smooth(se = FALSE, method = "loess", color = "gray40") + # Add a smooth trend line
  facet_wrap(~Name, scales = "free_y") + # Faceting by Trash Wheel
  theme_minimal() +
  labs(title = "Volume Growth Rate Over Time by Trash Wheel",
       x = "Year", y = "Volume Growth Rate (%)") +
  theme(legend.title = element_blank(),
        axis.text.x = element_text(angle = 45, hjust = 1))
```

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

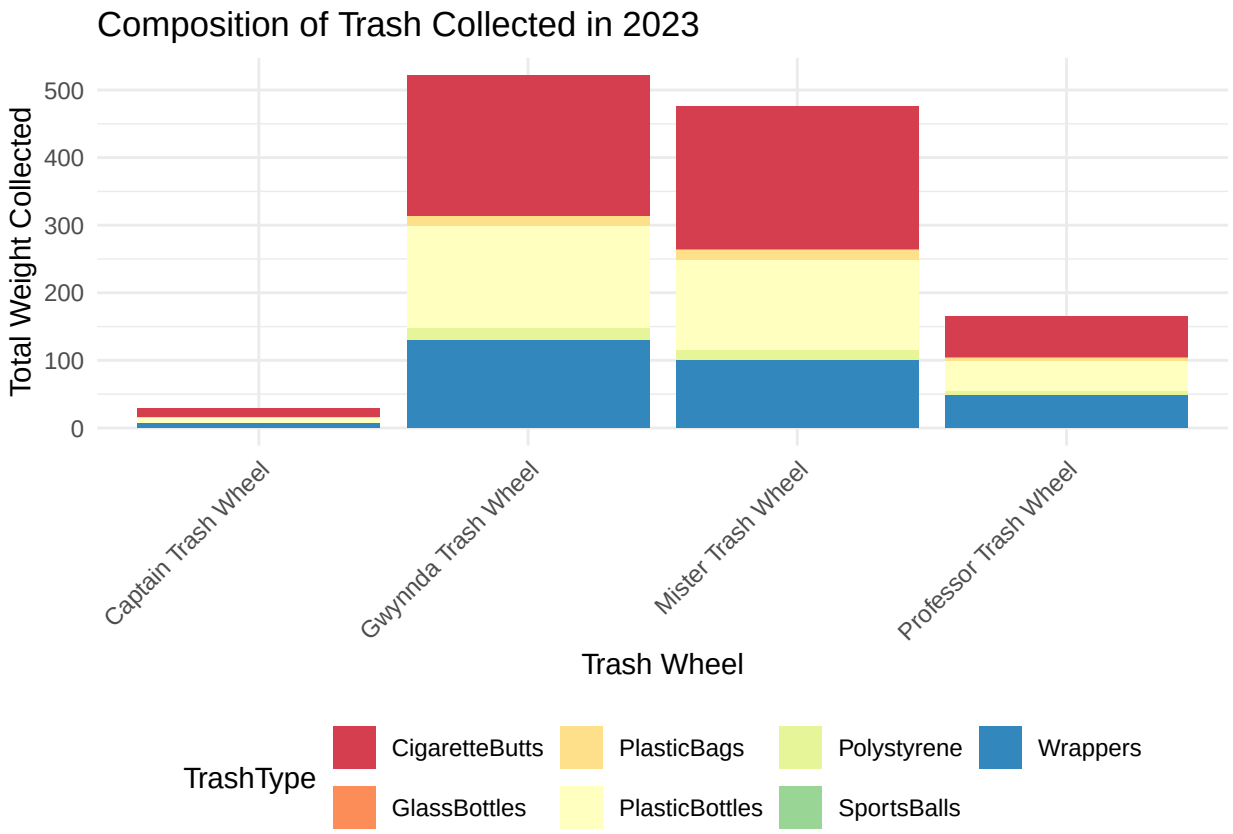
```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : span too small.  fewer data values than degrees of freedom.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : pseudoinverse used at 2021

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : neighborhood radius 1.01

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : There are other near singularities as well. 1.0201
```




```r
# Filter data for the most recent year
latest_year <- max(trashwheel_data$Year)
latest_year_data <- trashwheel_data %>%
  filter(Year == latest_year)

# Melt the data to long format for ggplot
composition_data_long <- latest_year_data %>%
  select(Name, PlasticBottles, Polystyrene, CigaretteButts, GlassBottles, PlasticBags, Wrappers, Sportsl
  gather(key = "TrashType", value = "Quantity", -Name) %>%
  group_by(Name, TrashType) %>%
  summarise(Quantity = sum(Quantity, na.rm = TRUE)) %>%
  ungroup()
```

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

```
ggplot(composition_data_long, aes(x = Name, y = Quantity, fill = TrashType)) +
  geom_bar(stat = "identity", position = "stack") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    legend.position = "bottom"
  ) +
  labs(title = paste("Composition of Trash Collected in", latest_year),
       x = "Trash Wheel",
       y = "Total Weight Collected") +
  scale_fill_brewer(palette = "Spectral") +
  scale_y_continuous(labels = scales::comma_format(scale = 1e-3))
```




```
library(ggplot2)

min_uses <- min(trashwheel_segments$Uses) - 10
max_uses <- max(trashwheel_segments$Uses) + 10

# Plotting with extended regression lines
ggplot(trashwheel_segments, aes(x = Uses, y = TotalWeight, label = Name)) +
  geom_point(aes(color = Name), size = 4) +
  geom_smooth(aes(color = Name), method = "lm", se = FALSE, fullrange = TRUE) +
  scale_x_continuous(name = "Number of Uses", limits = c(min_uses, max_uses)) +
  scale_y_continuous(name = "Total Weight Collected (kg)") +
```

```
  theme_minimal() +
  labs(title = "Weight Collected vs. Number of Uses for Each Trash Wheel",
       subtitle = "Linear regression lines extended for prediction") +
  theme(plot.title = element_text(hjust = 0.5))
```

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