

Debris Univariate Analysis - No Backbarrier

```
debris_df = read.csv("data/debris_data_no_backbarrier.csv")

totals_cols <- c(
  "total_debris_items",
  "total_plastic_items",
  "total_glass_items",
  "total_rubber_items",
  "total_metal_items",
  "total_processed.lumber_items",
  "total_cloth.fabric_items",
  "total_other_items"
)

numeric_columns <- c(
  "total_debris_items",
  "total_plastic_items",
  "total_glass_items",
  "total_rubber_items",
  "total_metal_items",
  "total_processed.lumber_items",
  "total_cloth.fabric_items",
  "total_other_items",
  "plastic.fragments.film",
  "plastic.fragments.foamed",
  "plastic.fragments.hard",
  "plastic.bags",
  "plastic.beverage.bottles",
  "plastic.bottle.or.container.caps",
  "plastic.cups",
  "plastic.food.wrappers",
  "plastic.other.jugs.or.containers",
  "plastic.straws",
  "plastic.utensils",
  "plastic.6.pack.rings",
  "plastic.cigar.tips",
  "plastic.cigarettes",
  "plastic.disposable.cigarette.lighters",
  "plastic.buoys.and.floats",
  "plastic.fishing.lures.and.line",
  "plastic.rope.and.nets",
  "plastic.balloons",
  "plastic.personal.care.products",
  "plastic.shotgun.shells.wads",
  "plastic.other", "metal.fragments",
  "metal.aerosol.cans",
```

```

"metal.aluminum.tin.cans",
"metal.other",
"glass.fragments",
"glass.beverage.bottles",
"glass.jars",
"glass.other",
"rubber.fragments",
"rubber.balloons",
"rubber.flip.flops",
"rubber.gloves",
"rubber.tires",
"rubber.other",
"processed.lumber.building.material",
"processed.lumber.paper.and.cardboard",
"processed.lumber.bags",
"processed.lumber.other",
"cloth.fabric.fragments",
"cloth.fabric.clothing.and.shoes",
"cloth.fabric.face.masks",
"cloth.fabric.gloves",
"cloth.fabric.rope.and.nets",
"cloth.fabric.towels.rags",
"cloth.fabric.other",
"other.other",
"plastic.other.fireworks",
"metal.other.construction.material",
"rubber.other.lobster.claw.rubber.bands",
"processed.lumber.paper.and.cardboard.fireworks",
"other.other.asphalt",
"other.other.brick",
"other.other.wax")

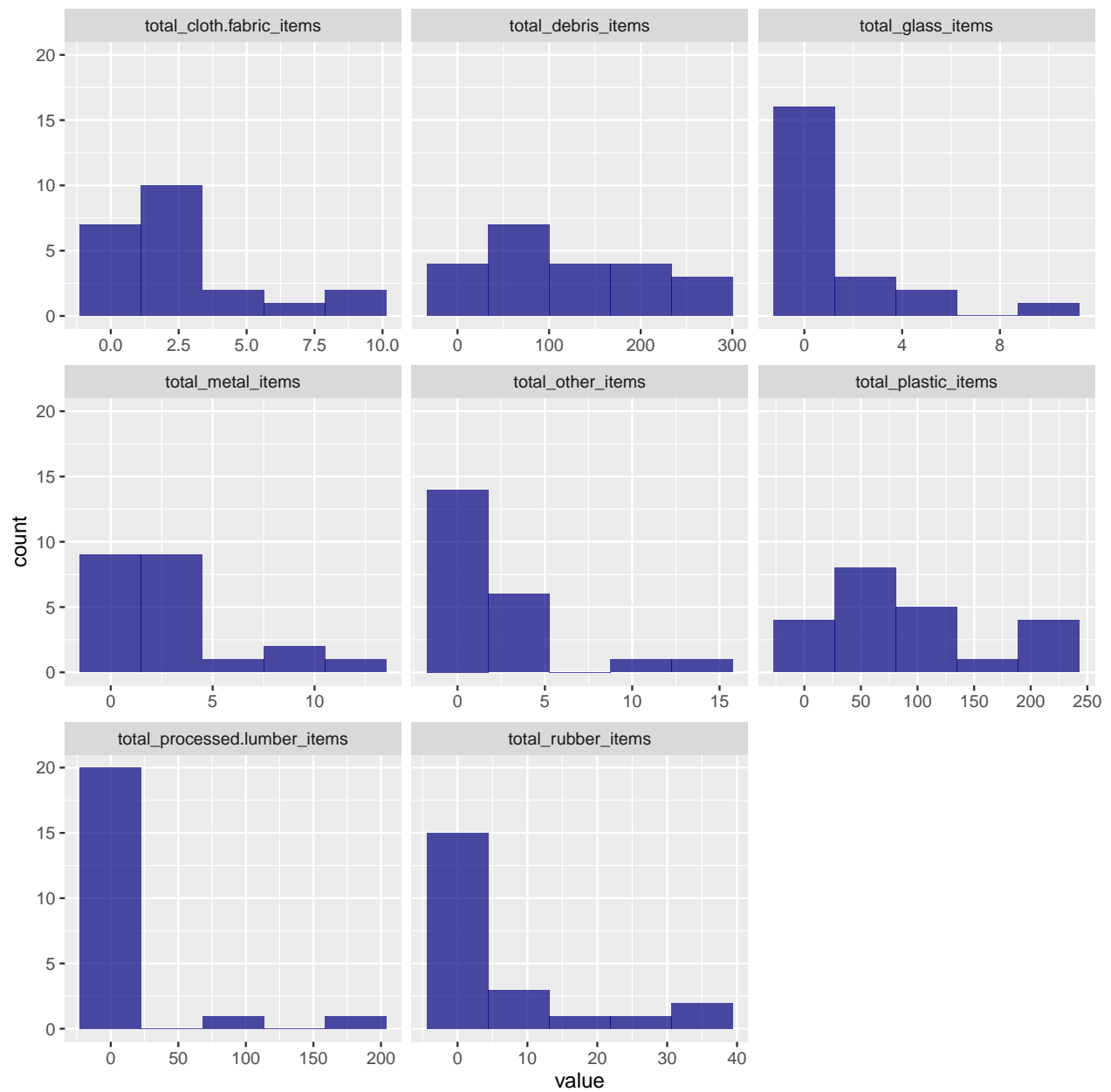
debris_df <- debris_df[, sapply(debris_df, class) != "logical"]
debris_df$year <- substr(debris_df$survey_date, 1, 4)

write.csv(debris_df, "cleaned_debris_no_backbarrier.csv")

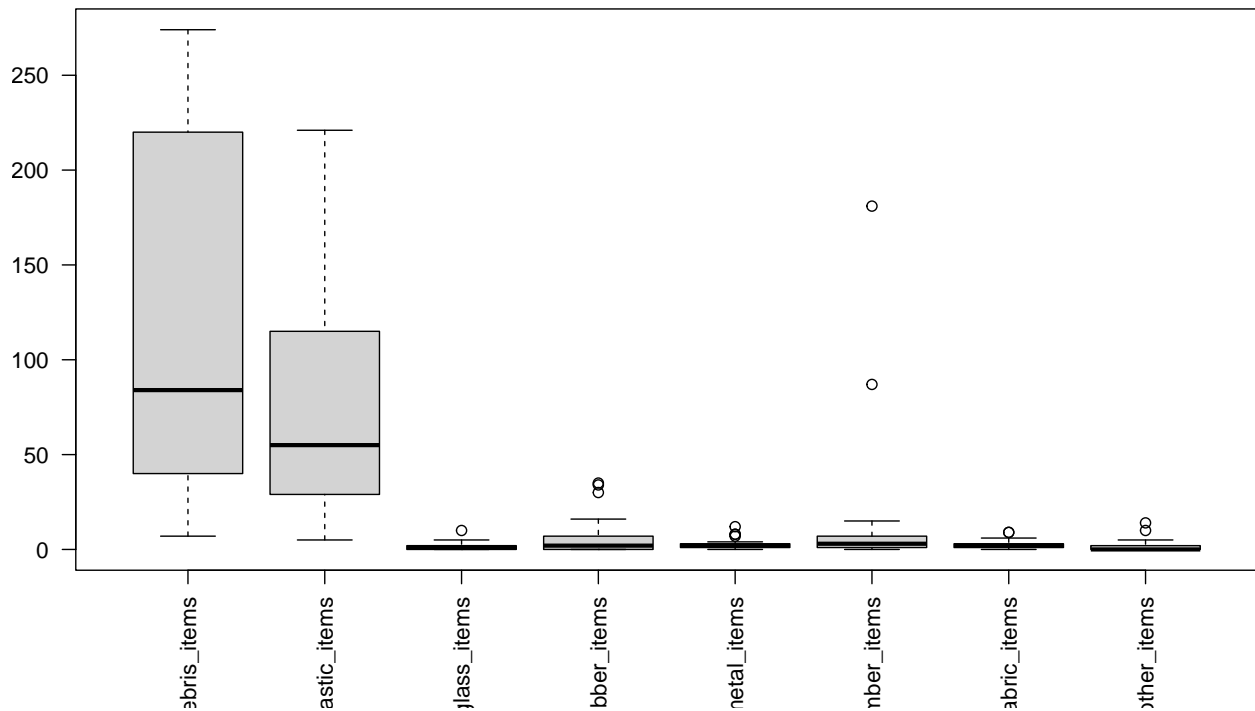
#hist(debris_df[test_cols])
#install.packages("tidyverse")
#install.packages("GGally")

# For ALL DATA
library(tidyr)
library(ggplot2)
ggplot(gather(debris_df[totals_cols]), aes(value)) +
  geom_histogram(bins = 5, fill="navyblue", alpha=.7) +
  facet_wrap(~key, scales = 'free_x')

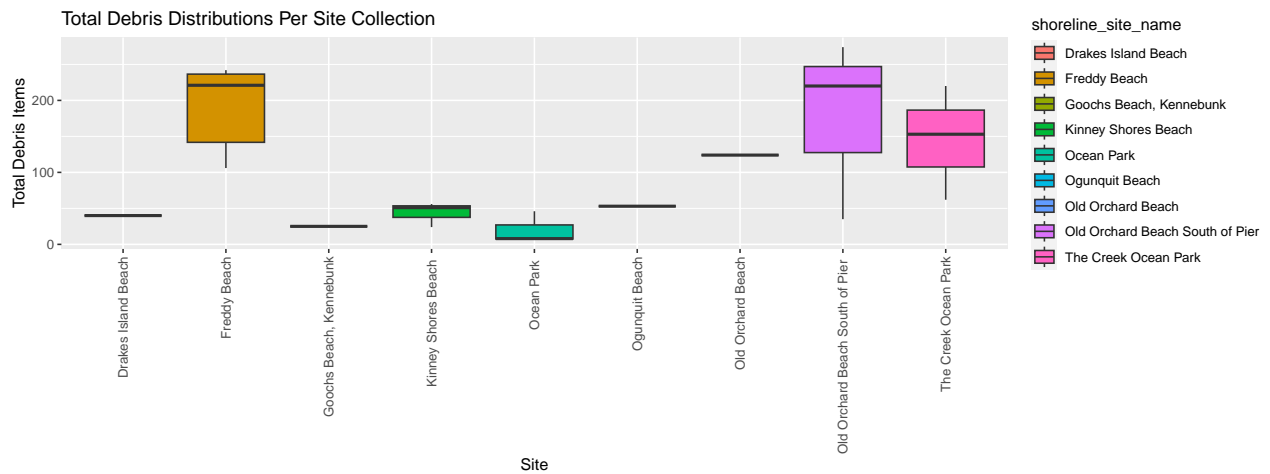
```



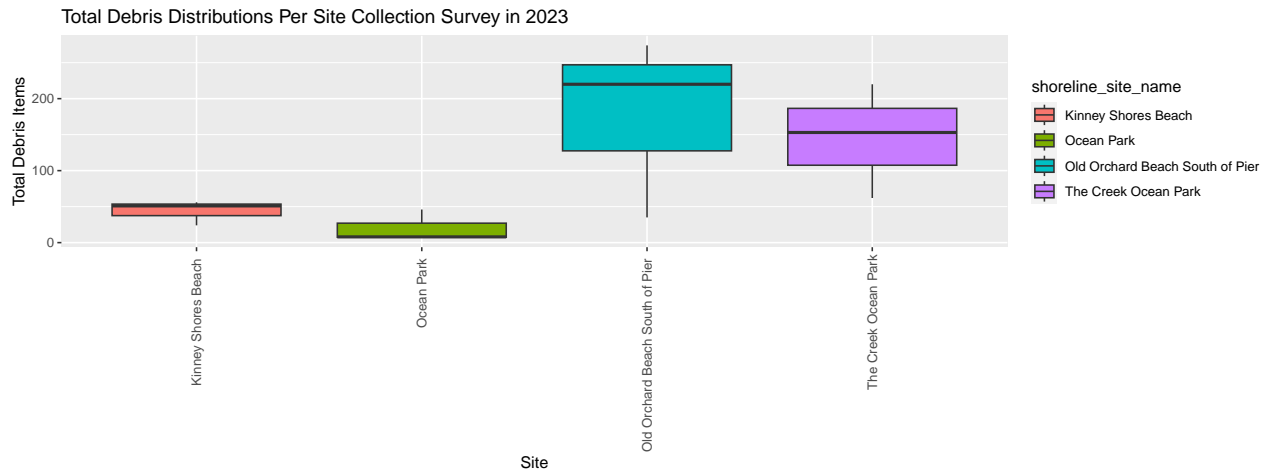
```
# FOR ALL DATA
boxplot(debris_df[totals_cols], las=2)
```



```
# boxplot of debris totals per site
ggplot(debris_df, aes(x=shoreline_site_name, y=total_debris_items, fill=shoreline_site_name)) +
  geom_boxplot() +
  scale_x_discrete(guide = guide_axis(angle = 90)) +
  ggtitle("Total Debris Distributions Per Site Collection") +
  ylab("Total Debris Items") +
  xlab("Site")
```

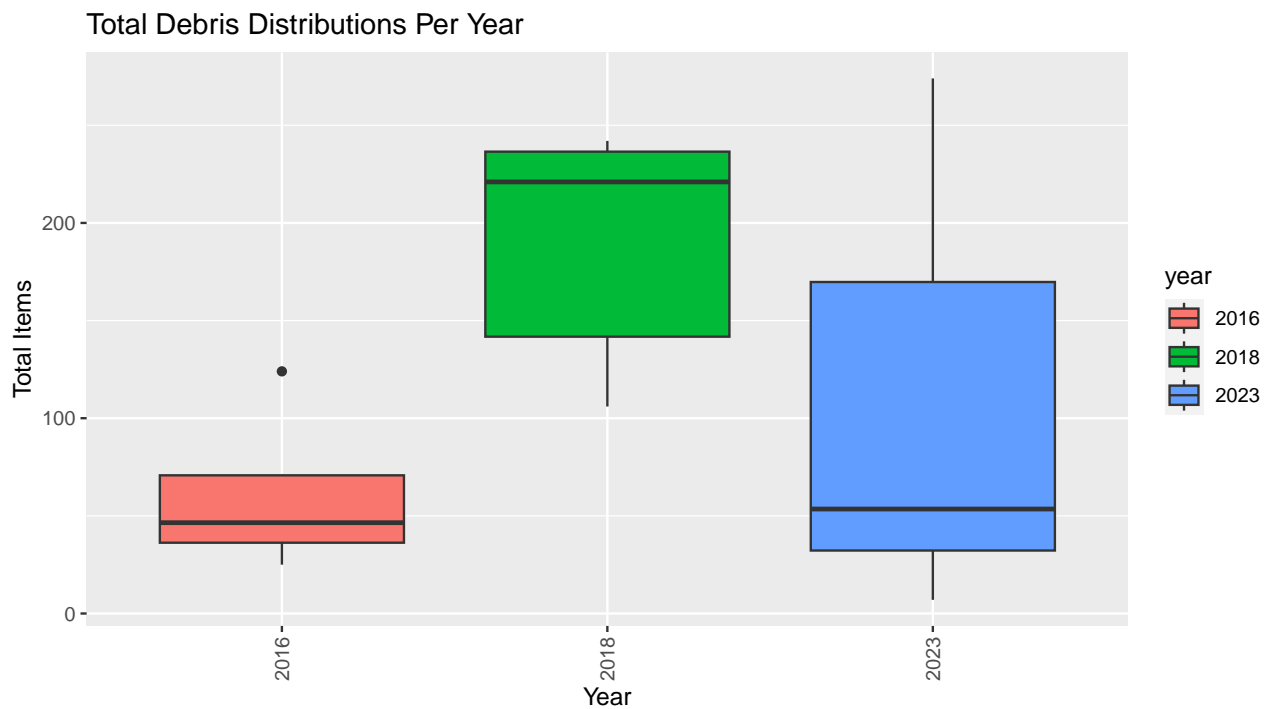


```
# boxplot of debris totals per site
ggplot(subset(debris_df, year == "2023"), aes(x=shoreline_site_name, y=total_debris_items, fill=shoreline_site_name)) +
  geom_boxplot() +
  scale_x_discrete(guide = guide_axis(angle = 90)) +
  ggtitle("Total Debris Distributions Per Site Collection Survey in 2023") +
  ylab("Total Debris Items") +
  xlab("Site")
```



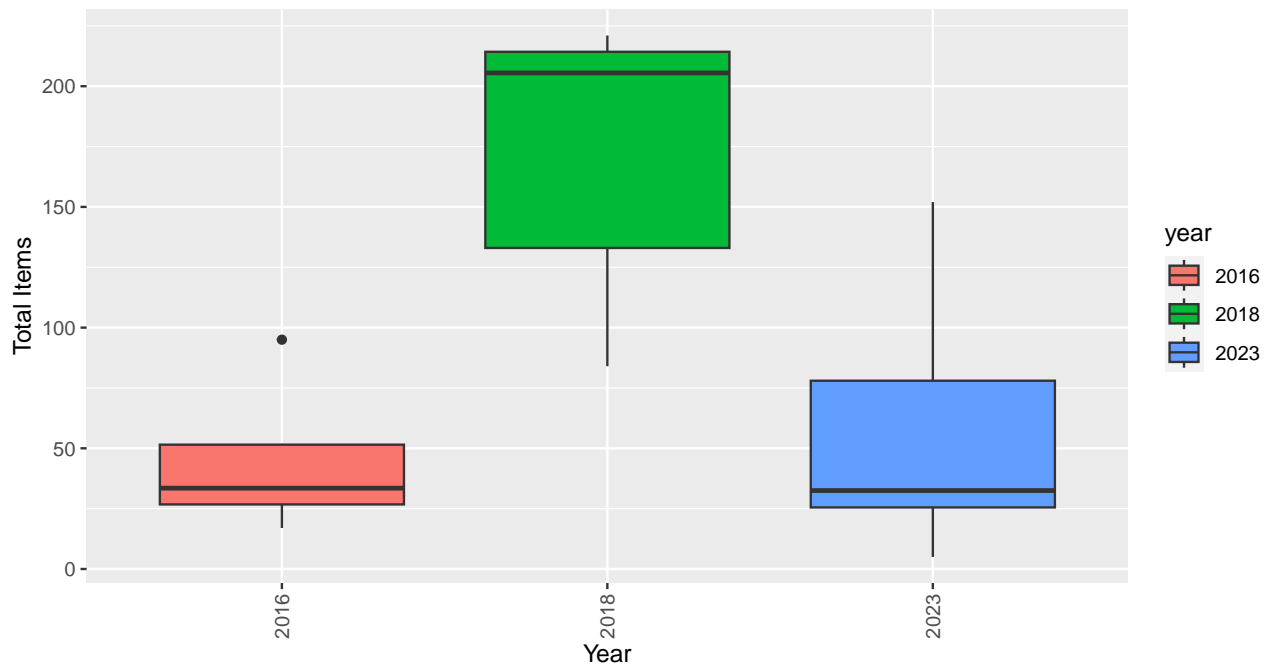
boxplot of debris totals per year

```
ggplot(debris_df, aes(x=year, y=total_debris_items, fill=year)) +
  geom_boxplot() +
  scale_x_discrete(guide = guide_axis(angle = 90)) +
  ggtitle("Total Debris Distributions Per Year") +
  ylab("Total Items") +
  xlab("Year")
```



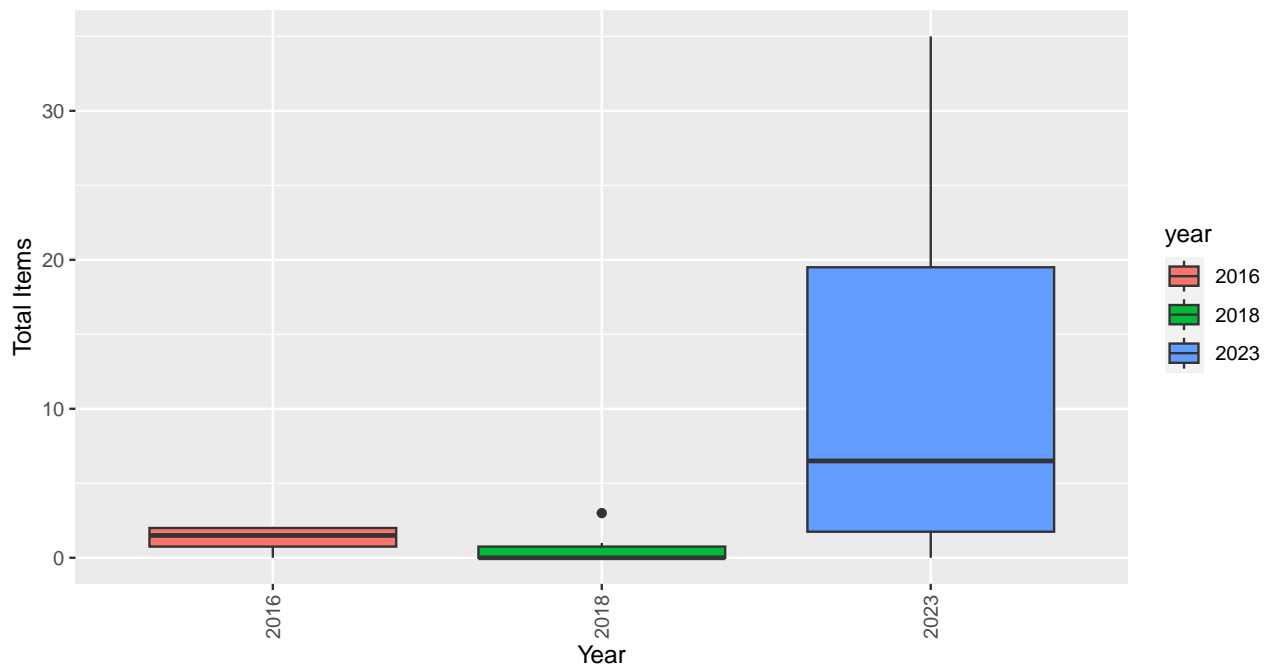
```
ggplot(debris_df, aes(x=year, y=total_plastic_items, fill=year)) +
  geom_boxplot() +
  scale_x_discrete(guide = guide_axis(angle = 90)) +
  ggtitle("Total Plastic Distributions Per Year") +
  ylab("Total Items") +
  xlab("Year")
```

Total Plastic Distributions Per Year



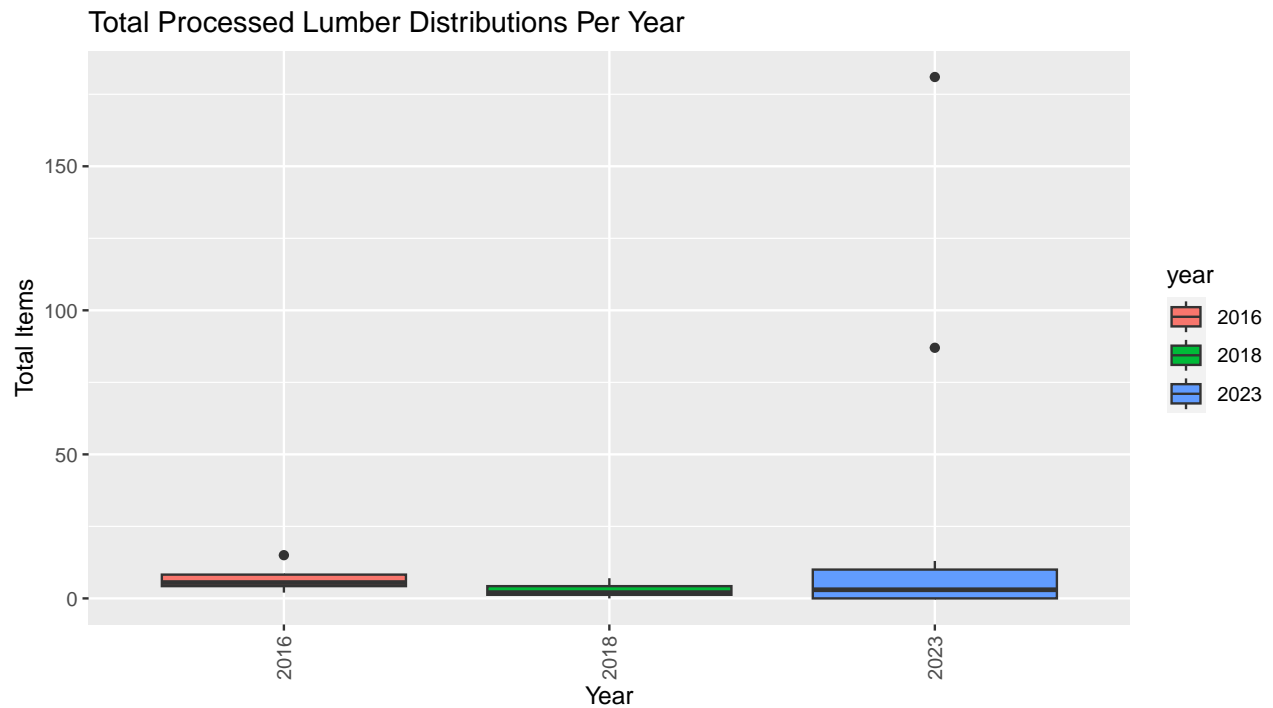
```
ggplot(debris_df, aes(x=year, y=total_rubber_items, fill=year)) +
  geom_boxplot() +
  scale_x_discrete(guide = guide_axis(angle = 90)) +
  ggtitle("Total Rubber Distributions Per Year") +
  ylab("Total Items") +
  xlab("Year")
```

Total Rubber Distributions Per Year

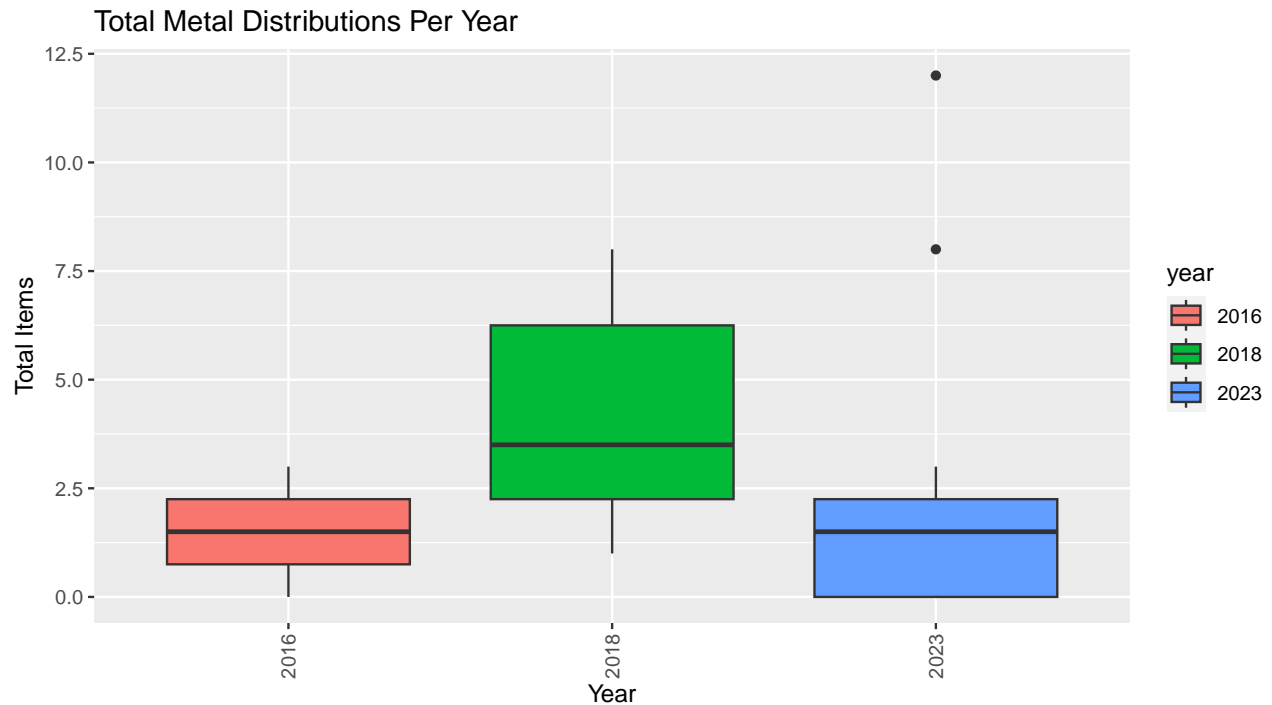


```
ggplot(debris_df, aes(x=year, y=total_processed.lumber_items, fill=year)) +
  geom_boxplot() +
```

```
scale_x_discrete(guide = guide_axis(angle = 90)) +
ggtitle("Total Processed Lumber Distributions Per Year") +
ylab("Total Items") +
xlab("Year")
```



```
ggplot(debris_df, aes(x=year, y=total_metal_items, fill=year)) +
geom_boxplot() +
scale_x_discrete(guide = guide_axis(angle = 90)) +
ggtitle("Total Metal Distributions Per Year") +
ylab("Total Items") +
xlab("Year")
```



```
t.test(debris_df[debris_df$year == "2018"]$total_debris_items, debris_df[debris_df$year == "2023"]$total_debris_items)
```

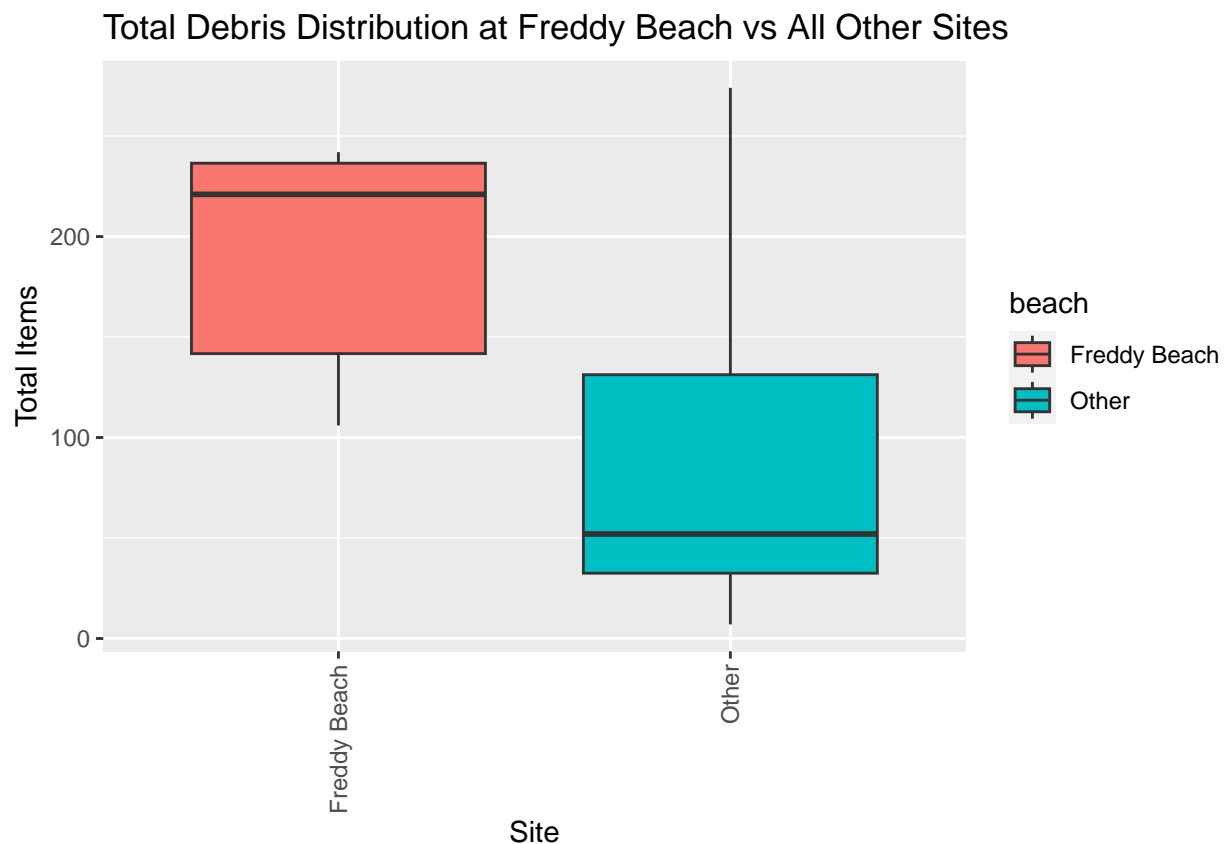
```
##
## One Sample t-test
##
## data: debris_df[debris_df$year == "2018"]$total_debris_items
## t = 5.9658, df = 21, p-value = 6.388e-06
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 75.35606 156.00758
## sample estimates:
## mean of x
## 115.6818
```

```
# boxplot of debris totals per site vs freddy beach
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
freddy_vs_all_df <- debris_df
freddy_vs_all_df <- freddy_vs_all_df %>%
  mutate(beach = if_else(shoreline_site_name == "Freddy Beach", "Freddy Beach", "Other"))
ggplot(freddy_vs_all_df, aes(x=beach, y=total_debris_items, fill=beach)) +
  geom_boxplot() +
  scale_x_discrete(guide = guide_axis(angle = 90)) +
```



```
ggtitle("Total Debris Distribution at Freddy Beach vs All Other Sites") +
ylab("Total Items") +
xlab("Site")
```



```
# means of the two groups vs each other.
print(mean(freddy_vs_all_df[freddy_vs_all_df$beach == "Freddy Beach"]$total_debris_items))
```

```
## [1] 115.6818
```

```
not_freddy <- subset(freddy_vs_all_df, subset=beach != "Freddy Beach")
print(mean(not_freddy$total_debris_items))
```

```
## [1] 87.375
```

```
# On Average, there is 11% more debris at Freddy Beach than all other sites.
(116 - 87)/(116 + 87)
```

```
## [1] 0.1428571
```

```
# T test of Freddy beach total debris vs every other location.
```

```
# We have a p-value of 1e-6, and a confidence interval of 86 to 162 with the mean sitting at 124 debris
```

```
t.test(debris_df[debris_df$shoreline_site_name == "Freddy Beach"]$total_debris_items, debris_df[debris_
```

```
##
```

```
## One Sample t-test
```

```
##
```

```
## data: debris_df[debris_df$shoreline_site_name == "Freddy Beach"]$total_debris_items
```

```
## t = 5.9658, df = 21, p-value = 6.388e-06
```

```
## alternative hypothesis: true mean is not equal to 0
```

```

## 95 percent confidence interval:
##    75.35606 156.00758
## sample estimates:
## mean of x
##    115.6818

anovatest <- aov(total_debris_items ~ year, data=debris_df)
summary(anovatest)

##              Df Sum Sq Mean Sq F value Pr(>F)
## year          2  50860   25430   3.933 0.0372 *
## Residuals    19 122856    6466
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

trimmed_data <- subset(debris_df, select=c(
  'total_plastic_items',
  'shoreline_site_id',
  'shoreline_site_name',
  'total_glass_items',
  'total_rubber_items',
  'total_metal_items',
  'total_processed.lumber_items',
  'total_cloth.fabric_items',
  'total_other_items'))

total_columns <- c('total_plastic_items',
  'total_glass_items',
  'total_rubber_items',
  'total_metal_items',
  'total_processed.lumber_items',
  'total_cloth.fabric_items',
  'total_other_items')

sum_df <- trimmed_data %>% group_by(shoreline_site_name) %>% summarise_at(total_columns, sum)
sum_df <- sum_df[order(sum_df$total_plastic_items),]
sum_df$shoreline_site_name <- factor(sum_df$shoreline_site_name, levels=rev(c(
  'Freddy Beach', 'Old Orchard Beach South of Pier ', 'The Creek Ocean Park', 'Ocean Park ',
  'Kinney Shores Beach ', 'Old Orchard Beach', 'Ogunquit Beach', 'Drakes Island Beach', 'Goochs Beach, I

sum_df |> pivot_longer(-shoreline_site_name) |> ggplot(aes(x=shoreline_site_name, y=value, fill=name)) +
  geom_bar(position="stack", stat="identity") + coord_flip() +
  ggtitle("Total Debris Type Breakdown Per Site") +
  xlab("Site Locations") +
  ylab("Total Debris (#)")

```

