Vision Zero - Testing

```
# Global options
knitr::opts_chunk$set(
  warning = FALSE,  # keine Warnungen
  message = FALSE,  # keine Nachrichten wie "Package xy loaded"
  error = FALSE     # keine Fehlermeldungen anzeigen
)
```

Packages:

```
library(sf)
library(ggplot2)
library(dplyr)
library(tidyr)
library(jsonlite)
library(lwgeom)
library(readxl)
library(scales)
library(here)
library(zoo)
library(viridis)
```

Loading and Cleaning Data

Loading dataframes (GeoJSON Format)

```
strassen_wege <- st_read("strassen_wege.geojson")
```

```
Reading layer `strassen_wege' from data source
  `C:\Users\micha\OneDrive\Dokumente\GitHub\Kerstin_Michael\Test_M\strassen_wege.geojson'
  using driver `GeoJSON'
Simple feature collection with 7688 features and 8 fields
Geometry type: LINESTRING
Dimension:     XY
Bounding box:  xmin: 7.555919 ymin: 47.51977 xmax: 7.693801 ymax: 47.60093
Geodetic CRS:  WGS 84
```

```r
velorouten <- st_read("velorouten.geojson")
```

```
Reading layer `velorouten' from data source
  `C:\Users\micha\OneDrive\Dokumente\GitHub\Kerstin_Michael\Test_M\velorouten.geojson'
  using driver `GeoJSON'
Simple feature collection with 21 features and 9 fields
Geometry type: LINESTRING
Dimension:     XY
Bounding box:  xmin: 7.55692 ymin: 47.53042 xmax: 7.657115 ymax: 47.59597
Geodetic CRS:  WGS 84
```

```r
unfaelle <- st_read("unfaelle.geojson")
```

```
Reading layer `unfaelle' from data source
  `C:\Users\micha\OneDrive\Dokumente\GitHub\Kerstin_Michael\Test_M\unfaelle.geojson'
  using driver `GeoJSON'
Simple feature collection with 10848 features and 12 fields
Geometry type: POINT
Dimension:     XY
Bounding box:  xmin: 7.556191 ymin: 47.52373 xmax: 7.692592 ymax: 47.60021
Geodetic CRS:  WGS 84
```

```r
velostadtplan <- st_read("velostadtplan.geojson")
```

```
Reading layer `velostadtplan' from data source
  `C:\Users\micha\OneDrive\Dokumente\GitHub\Kerstin_Michael\Test_M\velostadtplan.geojson'
  using driver `GeoJSON'
Simple feature collection with 5794 features and 8 fields
Geometry type: GEOMETRY
Dimension:     XY
Bounding box:  xmin: 7.481763 ymin: 47.50789 xmax: 7.716498 ymax: 47.60766
Geodetic CRS:  WGS 84
```

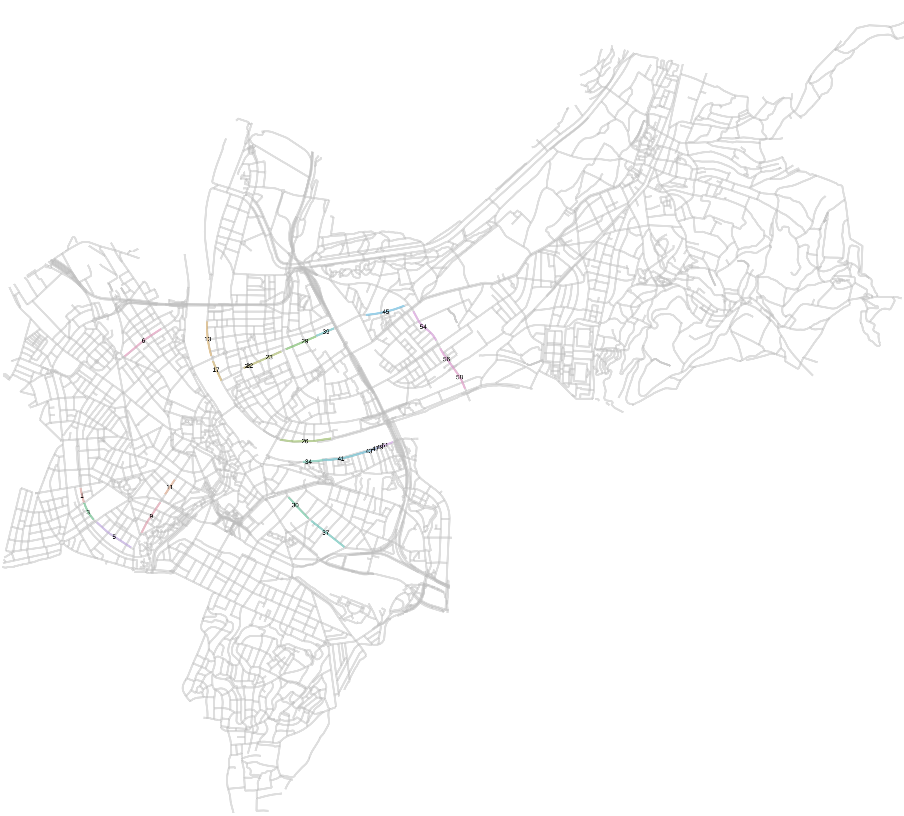
Cleaning “velostrassen”

```
velostrassen <- velostadtplan %>% filter(gml_id == "Velostrasse")
```

```
velostrassen_union <- velostrassen %>%
  st_union() %>%
  st_line_merge()

# Schritt 2: Wieder aufteilen in einzelne Linien
velostrassen_cleaned <- st_cast(velostrassen_union, "LINESTRING") %>%
  st_as_sf() %>%
  mutate(temp_ID = paste0(row_number()))
```

Clean copies

```
velostrassen_cleaned_selection <- velostrassen_cleaned %>%
  filter(temp_ID %in% c(1, 3, 5, 6, 9, 11, 13, 17, 21, 22, 23, 29,
                        39, 45, 54, 58, 41, 30 , 37, 34, 43, 23, 56,
                        47, 49, 51, 26))

p <- ggplot() +
  geom_sf(data = strassen_wege, color = "grey", size = 1, alpha = 0.5) +
  geom_sf(data = velostrassen_cleaned_selection, aes(color = temp_ID),
          size = 1, alpha = 0.3) +
  geom_sf_text(data = velostrassen_cleaned_selection, aes(label = temp_ID),
               size = 1, color = "black", check_overlap = FALSE) +
  theme_void() +
  theme(legend.position = "none")

print(p)
```



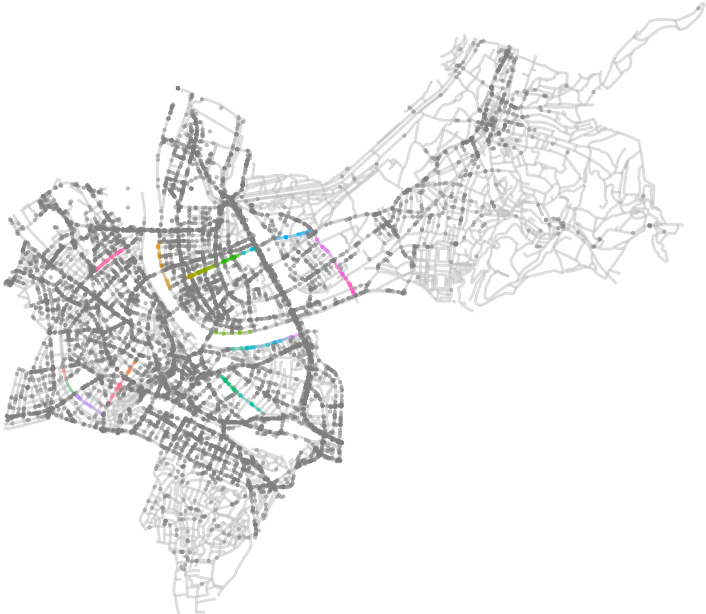
```r
ggsave("velorouten_grouped_selection.jpeg", plot = p, width = 12, height = 8, dpi = 600)

remove(velostrassen_cleaned, velostrassen_union)
```

Combine with "unfaelle" Dataset

```r
# Sicherheitscheck: gleiche CRS (Koordinatensystem)
unfaelle <- st_transform(unfaelle, st_crs(velostrassen_cleaned_selection))

# Optional: etwas Toleranzradius, falls Punkte nicht exakt auf Linie liegen (z.B. 10 Meter)
buffered_velo <- st_buffer(velostrassen_cleaned_selection, dist = 15)

# Räumlicher Join
unfaelle_with_match <- st_join(unfaelle, buffered_velo["temp_ID"])

# Neue Variable: TRUE/FALSE
```

```
unfaelle_with_match <- unfaelle_with_match %>%
  mutate(
    on_velostrasse = !is.na(temp_ID)  # TRUE wenn er zugewiesen wurde
  )
```

```
p <- ggplot() +
  geom_sf(data = strassen_wege, color = "grey", size = 1, alpha = 0.5) +
  geom_sf(data = velostrassen_cleaned_selection, aes(color = temp_ID), size = 1, alpha = 0.3
  geom_sf(data = unfaelle_with_match, aes(color = temp_ID), size = 0.1, alpha = 0.5) +
  theme_void() +
  theme(legend.position = "none")

print(p)
```



```
ggsave("accident_matching.jpeg", plot = p, width = 12, height = 8, dpi = 600)
```

Add street name

```
velostrassen_ID_sheet <- read_excel("../Test_M/velostrassen_ID_sheet.xlsx")

unfaelle_with_match$temp_ID <- as.numeric(unfaelle_with_match$temp_ID)
```

```
# left-join
unfaelle_clean <- unfaelle_with_match %>%
  left_join(velostrassen_ID_sheet, by = "temp_ID")
```

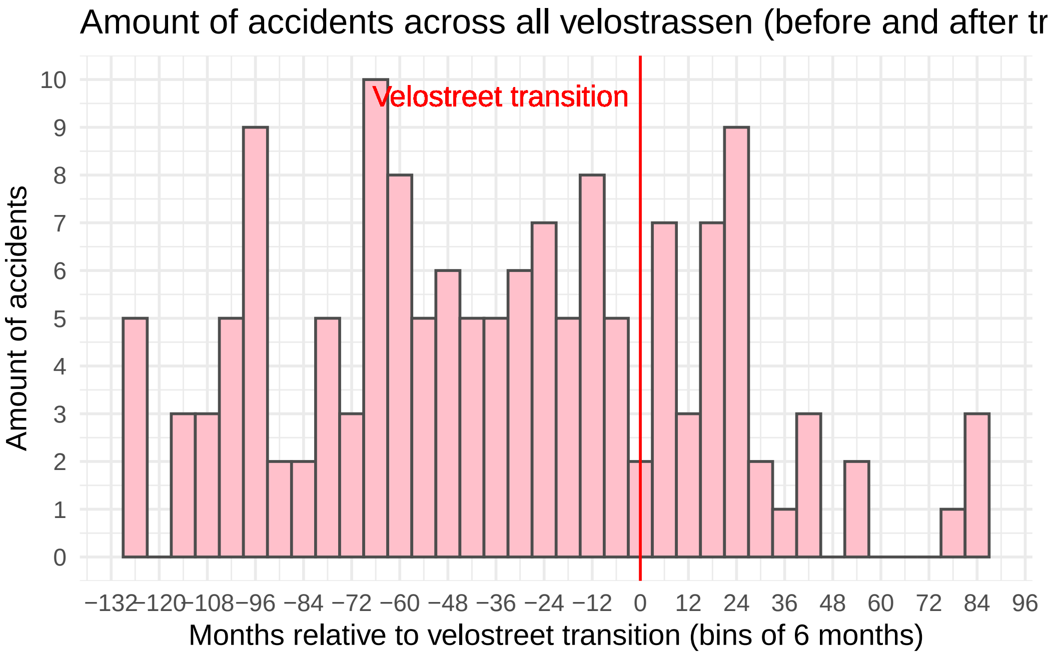
Backtrack Transition

```
unfaelle_clean <- unfaelle_clean %>%
  mutate(
    date_accident = as.Date(paste0(jahr, "-", monat, "-01")),
    ym_accident = as.yearmon(date_accident))


unfaelle_clean_velostrasse <- unfaelle_clean %>%
  filter(on_velostrasse == TRUE &
           typ != "Parkierunfall") %>%
  mutate(date_reference = as.Date(paste0(year, "-", month, "-01")),
    ym_reference = as.yearmon(date_reference),
    diff = ym_accident - ym_reference,
    diff_months = round((ym_accident - ym_reference) * 12))

remove(unfaelle_with_match, unfaelle, buffered_velo)
```

Plotting Data

```
p <- unfaelle_clean_velostrasse %>%
ggplot(aes(diff_months)) +
  geom_histogram(binwidth = 6, fill = "pink", color = "grey30") +
  geom_vline(xintercept = 0, color = "red", linetype = 1, linewidth = 0.5) +
  geom_text(aes(x = 10, y = 9, label = "Velostreet transition"),
            vjust = -1, hjust = 1.2, color = "red", size = 4) +
  scale_x_continuous(breaks = seq(floor(min(unfaelle_clean_velostrasse$diff_months) / 12) * 1
                                  ceiling(max(unfaelle_clean_velostrasse$diff_months) / 12) *
                                    12,
                                  by = 12)) +
  scale_y_continuous(breaks = c(0:10)) +
  ggtitle("Amount of accidents across all velostrassen (before and after transition)") +
  xlab("Months relative to velostreet transition (bins of 6 months)") +
  ylab("Amount of accidents") +
  theme_minimal()
```

```
print(p)
```

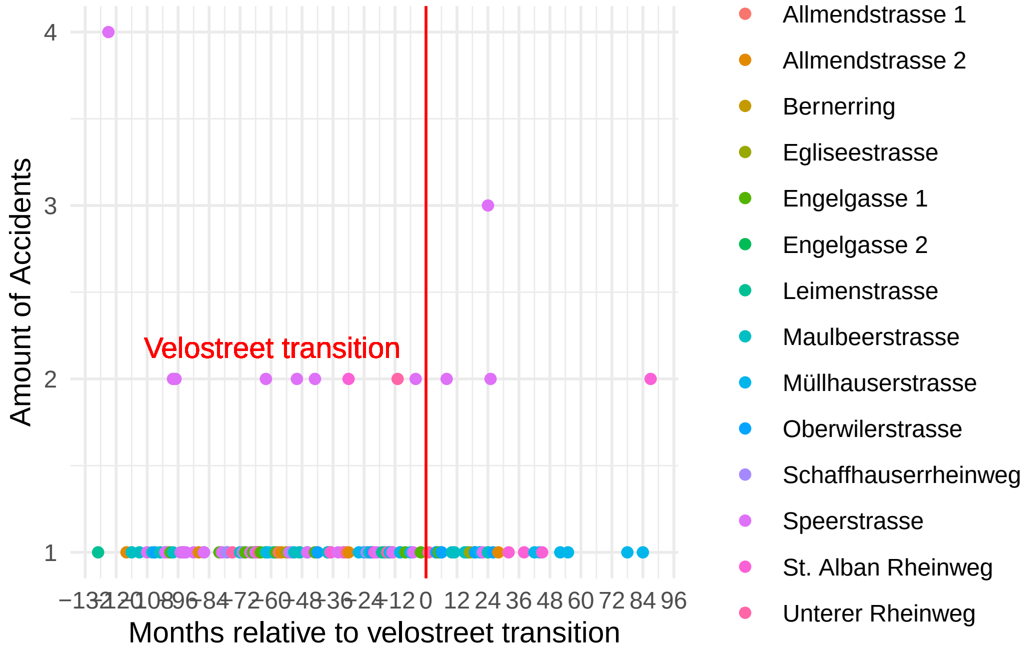


```
ggsave("Plot_1.jpeg", plot = p, width = 12, height = 8, dpi = 600)
```

This plot is not yet useful because the outer spans of the x-axis contain fewer streets.

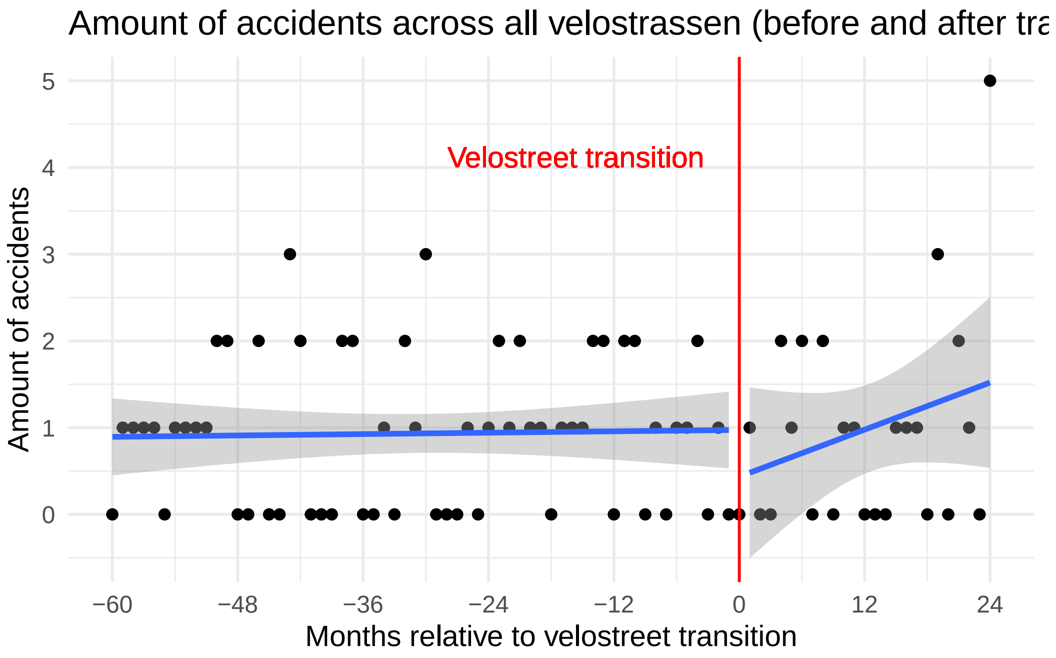
```
p <- unfaelle_clean_velostrasse %>%
  group_by(diff_months, name) %>%
  summarise(count = n()) %>%
ggplot(aes(diff_months, count, color = name)) +
  geom_point() +
  geom_vline(xintercept = 0, color = "red", linetype = 1, size = 0.5) +
  geom_text(aes(x = 10, y = 2, label = "Velostreet transition"),
            vjust = -1, hjust = 1.2, color = "red", size = 4) +
  scale_x_continuous(breaks = seq(floor(min(unfaelle_clean_velostrasse$diff_months) / 12) * 1
                  ceiling(max(unfaelle_clean_velostrasse$diff_months) / 12) * 12,
                  by = 12)) +
  scale_y_continuous(breaks = c(0:10)) +
  xlab("Months relative to velostreet transition") +
  ylab("Amount of Accidents") +
  theme_minimal()
```

```r
print(p)
```



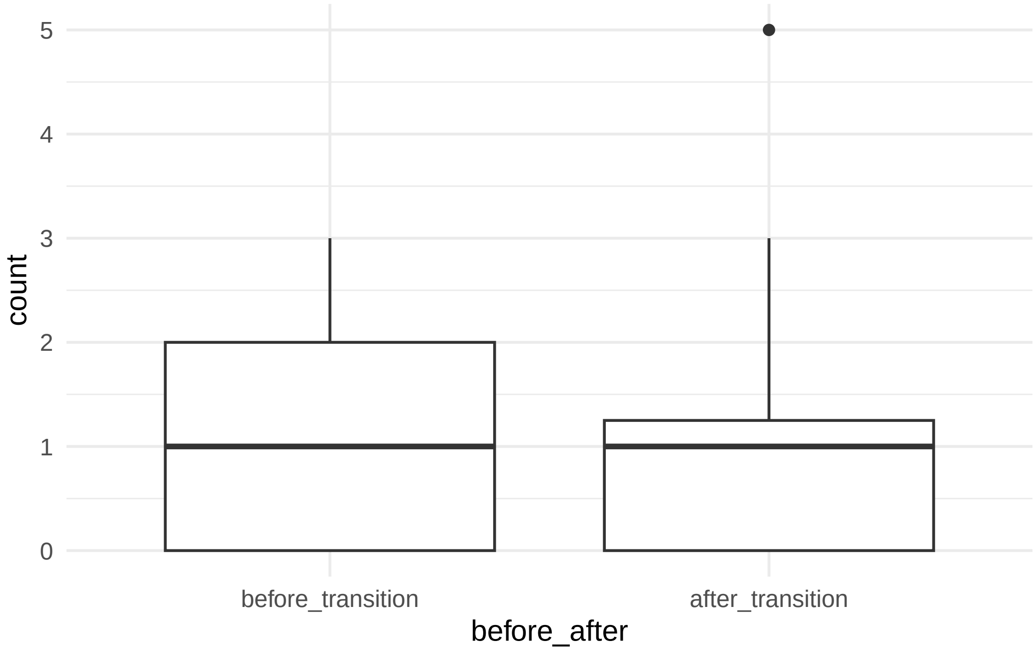
```r
ggsave("Plot_1.1.jpeg", plot = p, width = 12, height = 8, dpi = 600)
```

```r
x <- 1 # choose size of bins

unfaelle_clean_velostrasse_grouped <- unfaelle_clean_velostrasse %>%
  filter(diff_months >= -60 & diff_months <= 24) %>%
  mutate(diff_months_group = floor(diff_months / x) * x) %>%
  group_by(diff_months_group) %>%
  summarise(count = n(), .groups = "drop")

full_months <- tibble(diff_months_group = seq(-60, 24, by = x))

unfaelle_prep <- full_months %>%
  left_join(unfaelle_clean_velostrasse_grouped, by = "diff_months_group") %>%
  mutate(count = replace_na(count, 0)) %>%
  mutate(after_transition = diff_months_group > 0,
         before_transition = diff_months_group < 0)

p <- ggplot(unfaelle_prep, aes(diff_months_group, count)) +
  geom_point() +
  geom_smooth(data = subset(unfaelle_prep, after_transition == TRUE),
              aes(diff_months_group, count), method = "lm") +
  geom_smooth(data = subset(unfaelle_prep, before_transition == TRUE),
              aes(diff_months_group, count), method = "lm") +
  geom_vline(xintercept = 0, color = "red", linetype = 1, size = 0.5) +
  geom_text(aes(x = 4, y = 4, label = "Velostreet transition"),
            vjust = 0, hjust = 1.3, color = "red", size = 4) +
  scale_x_continuous(breaks = seq(
      floor(min(unfaelle_clean_velostrasse$diff_months) / 12) * 12,
      ceiling(max(unfaelle_clean_velostrasse$diff_months) / 12) * 12,
      by = 12
    )
  ) +
  scale_y_continuous(breaks = c(0:5)) +
  ggtitle("Amount of accidents across all velostrassen (before and after transition)") +
  xlab("Months relative to velostreet transition") +
  ylab("Amount of accidents") +
  theme_minimal()

print(p)
```



```r
ggsave("Plot_1.2.jpeg", plot = p, width = 12, height = 8, dpi = 600)
```

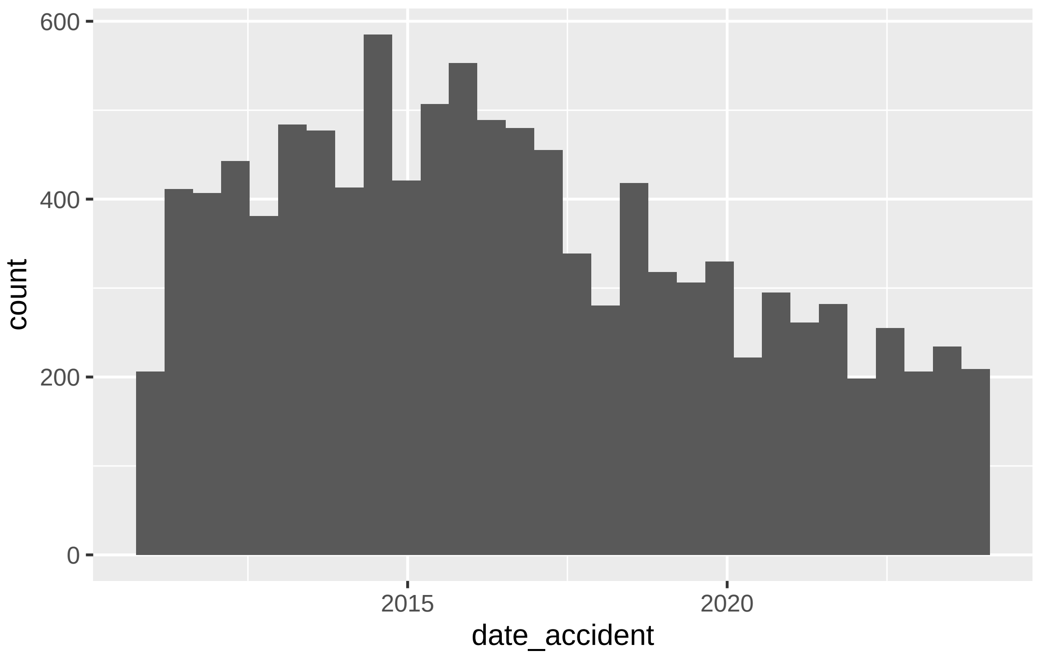
```r
p <- unfaelle_prep %>%
  pivot_longer(cols = c(before_transition, after_transition),
               names_to = 'before_after',
               values_to = "T_F") %>%
  subset(T_F == TRUE) %>%
  mutate(before_after = factor(before_after, levels = c("before_transition", "after_transitic
ggplot(aes(before_after, count)) +
  geom_boxplot() +
  theme_minimal()

print(p)
```



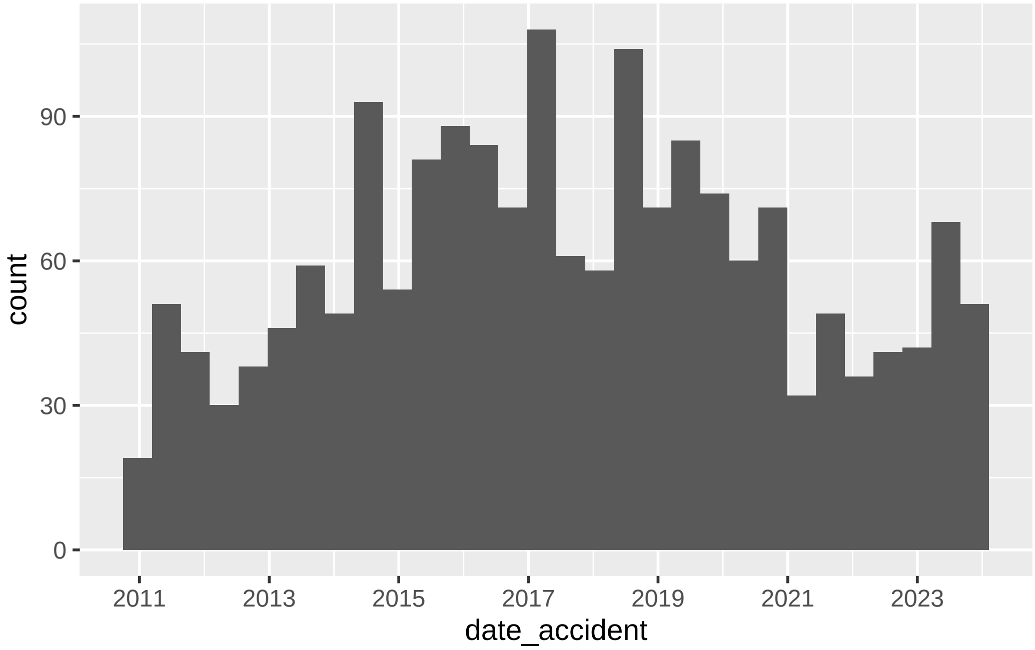
Further Analysis: Amount of Accidents each year

Confounding: Did it decrease overall? or maybe increase?

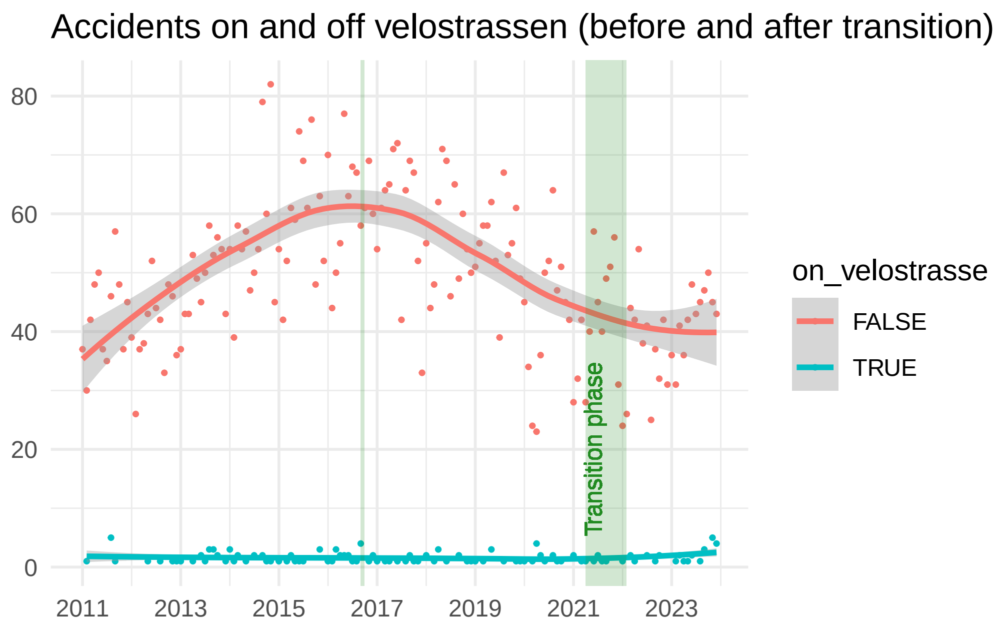
```r
p <- ggplot(unfaelle_clean, aes(date_accident)) +
  geom_histogram()

print(p)
```



```
p <- unfaelle_clean %>%
  filter(fahrrd_bet == "true") %>%
  ggplot(aes(date_accident)) +
  geom_histogram() +
  scale_x_date(
    breaks = seq(from = min(unfaelle_clean$date_accident),
                 to = max(unfaelle_clean$date_accident),
                 by = "2 years"),  # Set breaks every 2 years
    date_labels = "%Y"  # Format the labels as years
)

print(p)
```

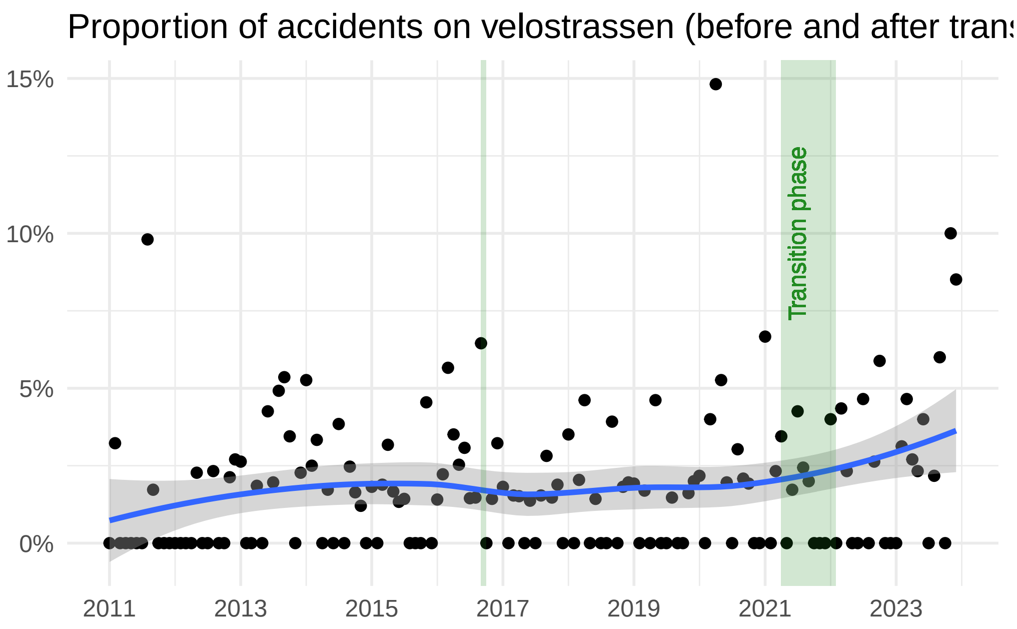


```
p <- unfaelle_clean %>%
  filter(typ != "Parkierunfall") %>%
  group_by(date_accident, on_velostrasse) %>%
  summarise(count = n()) %>%
ggplot(aes(date_accident, count, color = on_velostrasse)) +
  geom_point(size = 0.5) +
  geom_smooth() +
  annotate("rect",
           xmin = as.Date("2016-09-01"),  # your starting x value
           xmax = as.Date("2016-09-30"),    # your ending x value
           ymin = -Inf,     # from bottom of plot
           ymax = Inf,      # to top of plot
           alpha = 0.2,     # transparency
           fill = "forestgreen"     # color
  ) +
  annotate("rect",
           xmin = as.Date("2021-04-01"),  # your starting x value
           xmax = as.Date("2022-01-30"),    # your ending x value
           ymin = -Inf,     # from bottom of plot
           ymax = Inf,      # to top of plot
           alpha = 0.2,     # transparency
           fill = "forestgreen"     # color
  ) +
  geom_text(aes(x = as.Date("2021-10-01"), y = 20, label = "Transition phase"), color = "fore
  scale_x_date(
    breaks = seq(from = min(unfaelle_clean$date_accident),
                 to = max(unfaelle_clean$date_accident),
                 by = "2 years"),  # Set breaks every 2 years
    date_labels = "%Y"  # Format the labels as years
) +
  scale_y_continuous() +
  ggtitle("Accidents on and off velostrassen (before and after transition)") +
  xlab("") +
  ylab("") +
  theme_minimal()

print(p)
```



```
ggsave("Plot_2.1.jpeg", plot = p, width = 12, height = 8, dpi = 600)
```

```
p <- unfaelle_clean %>%
  filter(typ != "Parkierunfall") %>%
  group_by(date_accident) %>%
  summarise(relative_ov = sum(on_velostrasse == TRUE) / n()) %>%
ggplot(aes(date_accident, relative_ov)) +
  geom_point() +
  geom_smooth() +
  annotate("rect",
           xmin = as.Date("2016-09-01"),  # your starting x value
           xmax = as.Date("2016-09-30"),    # your ending x value
           ymin = -Inf,     # from bottom of plot
           ymax = Inf,      # to top of plot
           alpha = 0.2,     # transparency
           fill = "forestgreen"     # color
  ) +
  annotate("rect",
           xmin = as.Date("2021-04-01"),  # your starting x value
           xmax = as.Date("2022-01-30"),    # your ending x value
           ymin = -Inf,     # from bottom of plot
           ymax = Inf,      # to top of plot
           alpha = 0.2,     # transparency
           fill = "forestgreen"     # color
  ) +
  geom_text(aes(x = as.Date("2021-10-01"), y = 0.1, label = "Transition phase"), color = "fo
  scale_x_date(
    breaks = seq(from = min(unfaelle_clean$date_accident),
                 to = max(unfaelle_clean$date_accident),
                 by = "2 years"),  # Set breaks every 2 years
    date_labels = "%Y"  # Format the labels as years
) +
  scale_y_continuous(labels = scales::percent) +
  ggtitle("Proportion of accidents on velostrassen (before and after transition)") +
  xlab("") +
  ylab("") +
  theme_minimal()

print(p)
```



```r
ggsave("Plot_2.2.jpeg", plot = p, width = 12, height = 8, dpi = 600)
```

Traffic count data

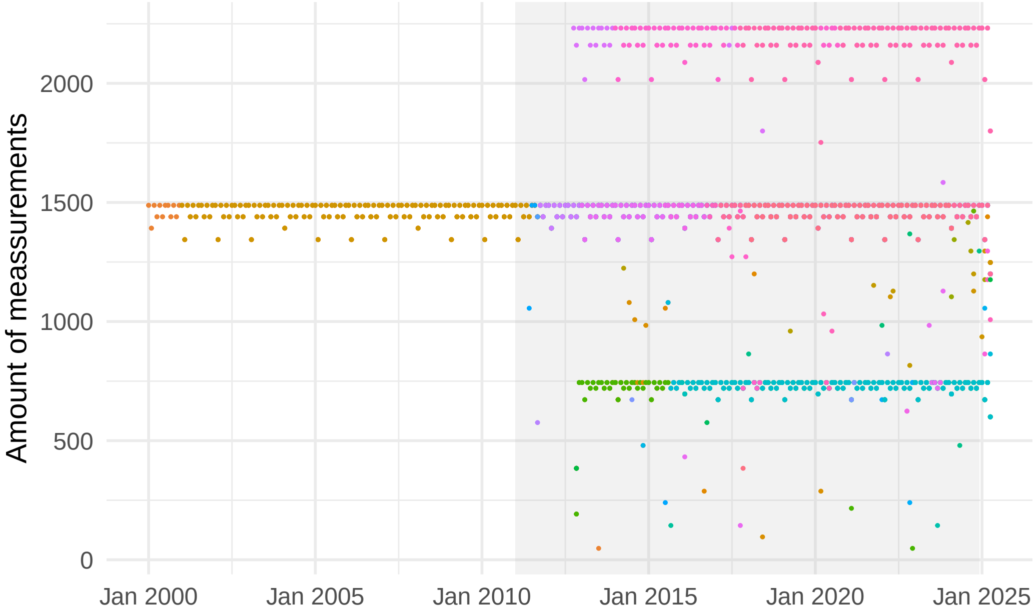
Confounding: Is the increase in accidents due to more cyclists around Basel?

```r
zaehlung <- read.csv2("../../../zaehlung.csv")
```

```r
zaehlung <- zaehlung %>%
  mutate(ym = as.yearmon(paste0(Year, "-", Month)))
```

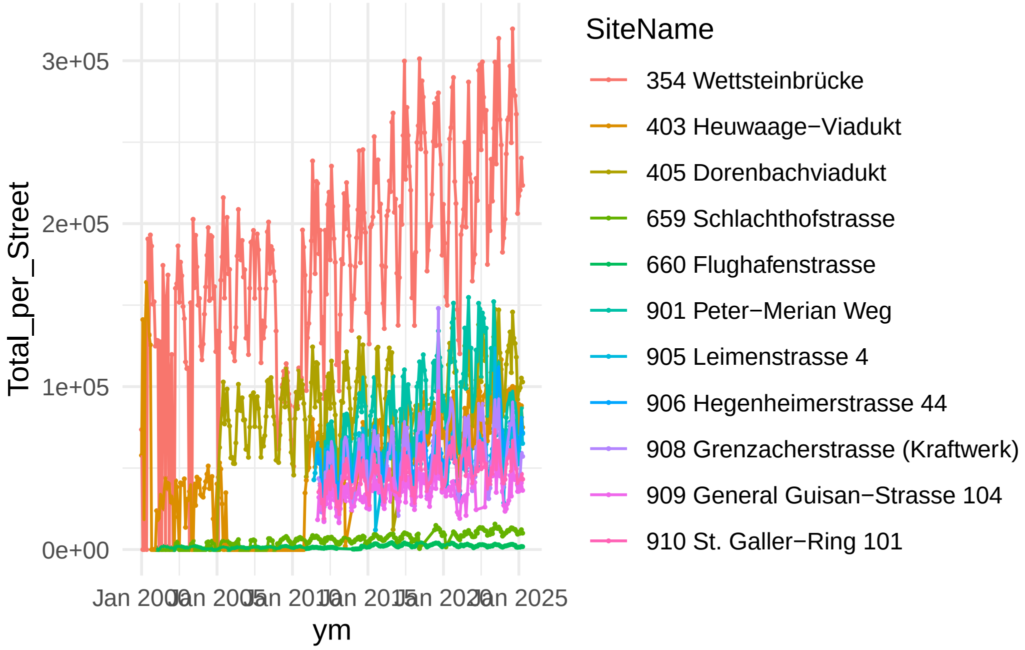
```r
p <- zaehlung %>%
  group_by(ym, SiteName) %>%
  summarise(count = n()) %>%
ggplot(aes(ym, count, color = SiteName)) +
  annotate("rect",
           xmin = as.yearmon("2011-01-01"),  # your starting x value
           xmax = as.yearmon("2024-12-31"),    # your ending x value
           ymin = -Inf,      # from bottom of plot
           ymax = Inf,       # to top of plot
           alpha = 0.2,      # transparency
```

```
            fill = "grey"      # color
  ) +
  xlab("") +
  ylab("Amount of meassurements") +
  geom_point(size = 0.2) +
  theme_minimal() +
  theme(legend.position = "none")

print(p)
```



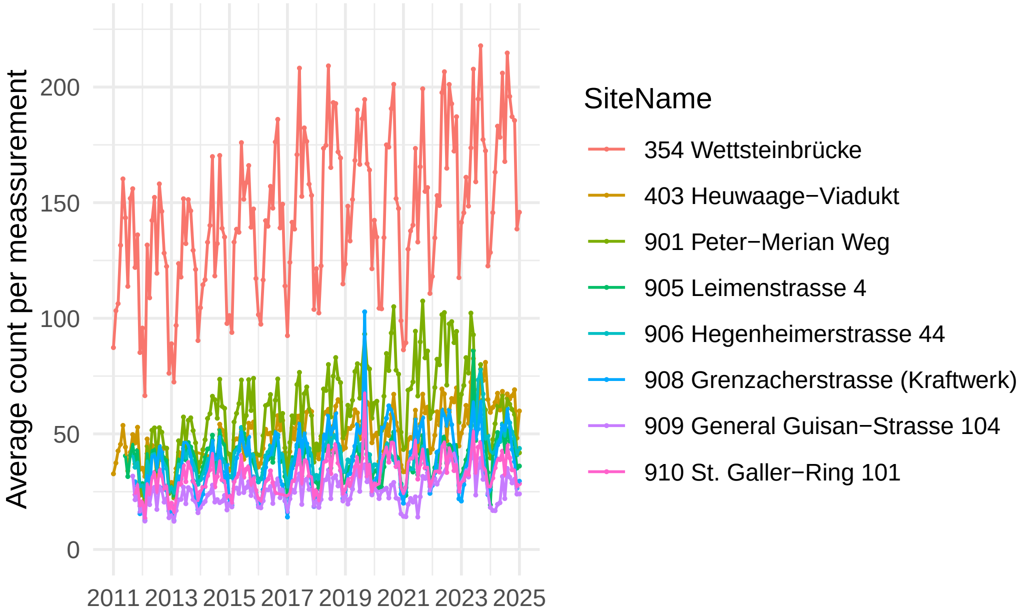
```
# Sites with at least one measurement 2011
sites_with_multiple_measurements <- zaehlung %>%
  filter(Year == 2011) %>%
  group_by(SiteName) %>%
  summarise(count = n()) %>%
  filter(count > 1) %>%
  pull(SiteName)

# Gesamten Datensatz auf diese Sites filtern
zaehlung_since_2011 <- zaehlung %>%
  filter(SiteName %in% sites_with_multiple_measurements)
```

```r
p <- zaehlung_since_2011 %>%
  filter(TrafficType == "Velo") %>%
  group_by(ym, SiteName) %>%
  summarise(Total_per_Street = sum(Total)) %>%
ggplot(aes(ym, Total_per_Street, group = SiteName, color = SiteName)) +
  geom_point(size = 0.2) +
  geom_line() +
  theme_minimal()

ggsave("Plot_3.jpeg", plot = p, width = 20, height = 8, dpi = 600)

print(p)
```



```r
zaehlung_selection <- zaehlung_since_2011 %>%
  filter(!(SiteCode %in% c(405, 659, 660)))
```

```r
p <- zaehlung_selection %>%
  filter(TrafficType == "Velo" &
           ym >= as.yearmon("2011-01")) %>%
  group_by(ym, SiteName) %>%
  summarise(Average_per_Meassurement = sum(Total) / n()) %>%
  mutate(ym = as.Date(ym)) %>%
ggplot(aes(ym, Average_per_Meassurement, color = SiteName)) +
  geom_point(size = 0.2) +
  geom_line() +
  xlab("") +
  ylab("Average count per meassurement") +
  scale_x_date(
    limits = as.Date(c("2011-01-01", "2025-01-01")),
    breaks = seq(as.Date("2011-01-01"), as.Date("2025-01-01"), by = "2 years"),
    date_labels = "%Y") +
  scale_y_continuous(limits = c(0,225),
                     breaks = c(0,50,100,150,200)) +
  theme_minimal()

print(p)
```



```
ggsave("Plot_4.jpeg", plot = p, width = 20, height = 8, dpi = 600)
```

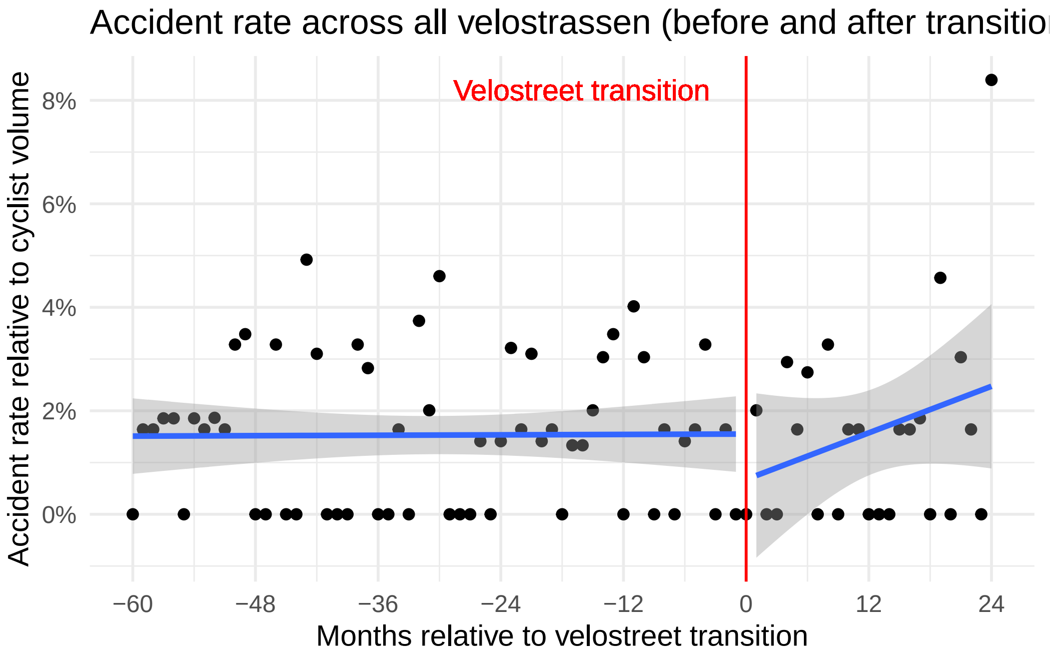
```
average_zaehlung <- zaehlung_selection %>%
  filter(TrafficType == "Velo", ym >= as.yearmon("2011-01")) %>%
  group_by(ym, SiteName) %>%
  summarise(
    Average_per_Measurement = sum(Total) / n(),
    .groups = "drop"
  ) %>%
  group_by(ym) %>%
  summarise(
    mean_average = mean(Average_per_Measurement),
    .groups = "drop"
  )
```

Plotting Data (Inclusion of Traffic Count Data)

```
unfaelle_clean_velostrasse_relative <- unfaelle_clean %>%
  filter(on_velostrasse == TRUE &
           typ != "Parkierunfall") %>%
  mutate(date_reference = as.Date(paste0(year, "-", month, "-01")),
    ym = as.yearmon(date_reference)) %>%
  left_join(average_zaehlung, by = "ym") %>%
  mutate(diff = ym_accident - ym,
    diff_months = round((ym_accident - ym) * 12))
```

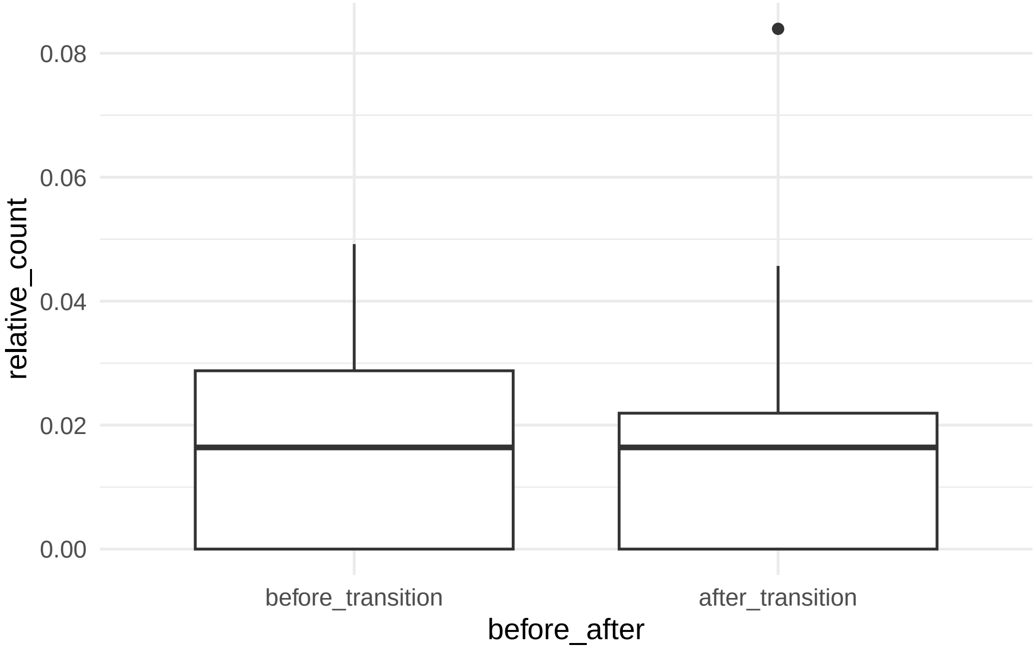
```
x <- 1 # choose size of bins

unfaelle_clean_velostrasse_grouped <- unfaelle_clean_velostrasse_relative %>%
  filter(diff_months >= -60 & diff_months <= 24) %>%
  mutate(diff_months_group = floor(diff_months / x) * x) %>%
  group_by(diff_months_group) %>%
  summarise(
    count = n(),
    relative_count = count / mean(mean_average, na.rm = TRUE),
    .groups = "drop"
  )

full_months <- tibble(diff_months_group = seq(-60, 24, by = x))

unfaelle_prep <- full_months %>%
  left_join(unfaelle_clean_velostrasse_grouped, by = "diff_months_group") %>%
  mutate(count = replace_na(count, 0),
         relative_count = replace_na(relative_count, 0)) %>%
  mutate(after_transition = diff_months_group > 0,
         before_transition = diff_months_group < 0)
```

```r
p <- ggplot(unfaelle_prep, aes(diff_months_group, relative_count)) +
  geom_point() +
  geom_smooth(data = subset(unfaelle_prep, after_transition == TRUE),
              aes(diff_months_group, relative_count), method = "lm") +
  geom_smooth(data = subset(unfaelle_prep, before_transition == TRUE),
              aes(diff_months_group, relative_count), method = "lm") +
  geom_vline(xintercept = 0, color = "red", linetype = 1, size = 0.5) +
  geom_text(aes(x = 4, y = 0.08, label = "Velostreet transition"),
            vjust = 0, hjust = 1.3, color = "red", size = 4) +
  scale_x_continuous(breaks = seq(
      floor(min(unfaelle_clean_velostrasse$diff_months) / 12) * 12,
      ceiling(max(unfaelle_clean_velostrasse$diff_months) / 12) * 12,
      by = 12
    )
  ) +
  scale_y_continuous(labels = percent_format(accuracy = 1)) +
  ggtitle("Accident rate across all velostrassen (before and after transition)") +
  xlab("Months relative to velostreet transition") +
  ylab("Accident rate relative to cyclist volume") +
  theme_minimal()

print(p)
```



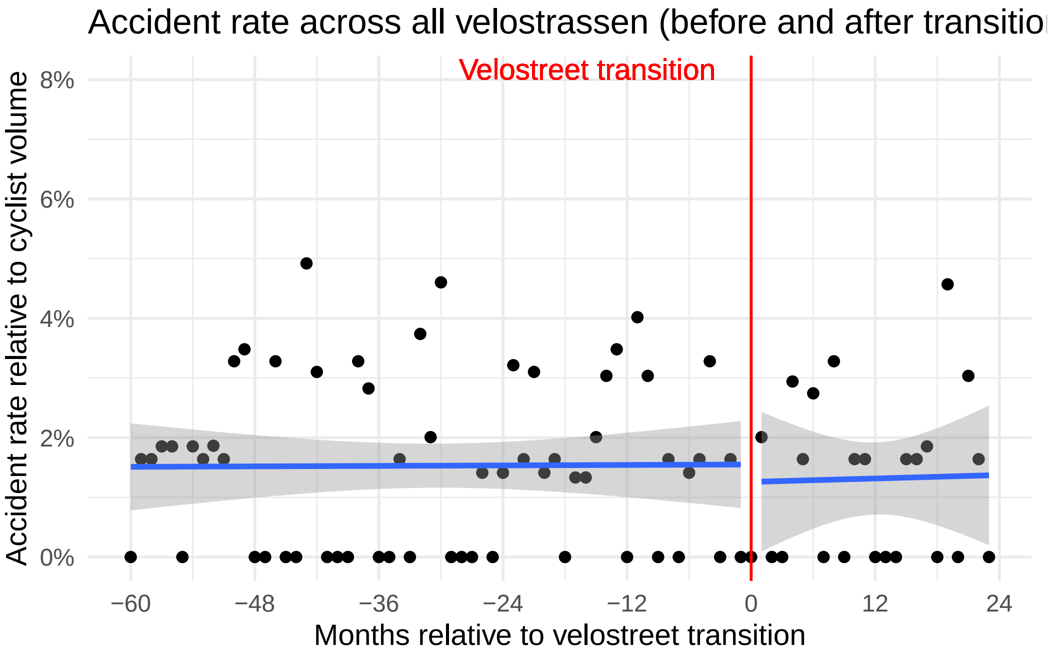
```
ggsave("Plot_6.jpeg", plot = p, width = 12, height = 8, dpi = 600)
```

```
p <- unfaelle_prep %>%
  pivot_longer(cols = c(before_transition, after_transition),
               names_to = 'before_after',
               values_to = "T_F") %>%
  subset(T_F == TRUE) %>%
  mutate(before_after = factor(before_after, levels = c("before_transition", "after_transiti
ggplot(aes(before_after, relative_count)) +
  geom_boxplot() +
  theme_minimal()

print(p)
```



Remove Outlier

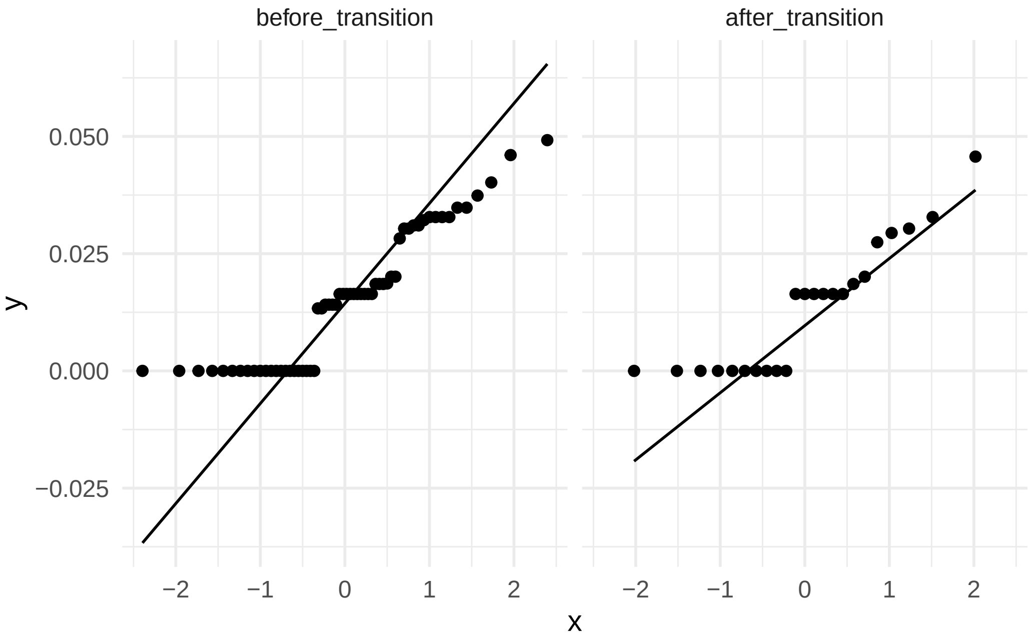
```
x <- 1 # choose size of bins

unfaelle_clean_velostrasse_grouped <- unfaelle_clean_velostrasse_relative %>%
  filter(diff_months >= -60 & diff_months <= 23) %>%
  mutate(diff_months_group = floor(diff_months / x) * x) %>%
  group_by(diff_months_group) %>%
  summarise(
    count = n(),
    relative_count = count / mean(mean_average, na.rm = TRUE),
    .groups = "drop"
  )

full_months <- tibble(diff_months_group = seq(-60, 23, by = x))

unfaelle_prep <- full_months %>%
  left_join(unfaelle_clean_velostrasse_grouped, by = "diff_months_group") %>%
  mutate(count = replace_na(count, 0),
         relative_count = replace_na(relative_count, 0)) %>%
  mutate(after_transition = diff_months_group > 0,
         before_transition = diff_months_group < 0)

p <- ggplot(unfaelle_prep, aes(diff_months_group, relative_count)) +
  geom_point() +
  geom_smooth(data = subset(unfaelle_prep, after_transition == TRUE),
              aes(diff_months_group, relative_count), method = "lm") +
  geom_smooth(data = subset(unfaelle_prep, before_transition == TRUE),
              aes(diff_months_group, relative_count), method = "lm") +
  geom_vline(xintercept = 0, color = "red", linetype = 1, size = 0.5) +
  geom_text(aes(x = 4, y = 0.08, label = "Velostreet transition"),
            vjust = 0, hjust = 1.3, color = "red", size = 4) +
  scale_x_continuous(breaks = seq(
      floor(min(unfaelle_clean_velostrasse$diff_months) / 12) * 12,
      ceiling(max(unfaelle_clean_velostrasse$diff_months) / 12) * 12,
      by = 12
    )
  ) +
  scale_y_continuous(labels = percent_format(accuracy = 1)) +
  ggtitle("Accident rate across all velostrassen (before and after transition)") +
  xlab("Months relative to velostreet transition") +
  ylab("Accident rate relative to cyclist volume") +
  theme_minimal()

print(p)
```



```r
ggsave("Plot_6.jpeg", plot = p, width = 12, height = 8, dpi = 600)
```

T-Test

```r
p <- unfaelle_prep %>%
  pivot_longer(cols = c(before_transition, after_transition),
               names_to = 'before_after',
               values_to = "T_F") %>%
  subset(T_F == TRUE) %>%
  mutate(before_after = factor(before_after, levels = c("before_transition", "after_transitio
ggplot(aes(sample = relative_count)) +
  geom_qq() +
  geom_qq_line() +
  facet_wrap(~before_after) +
  theme_minimal()

print(p)
```



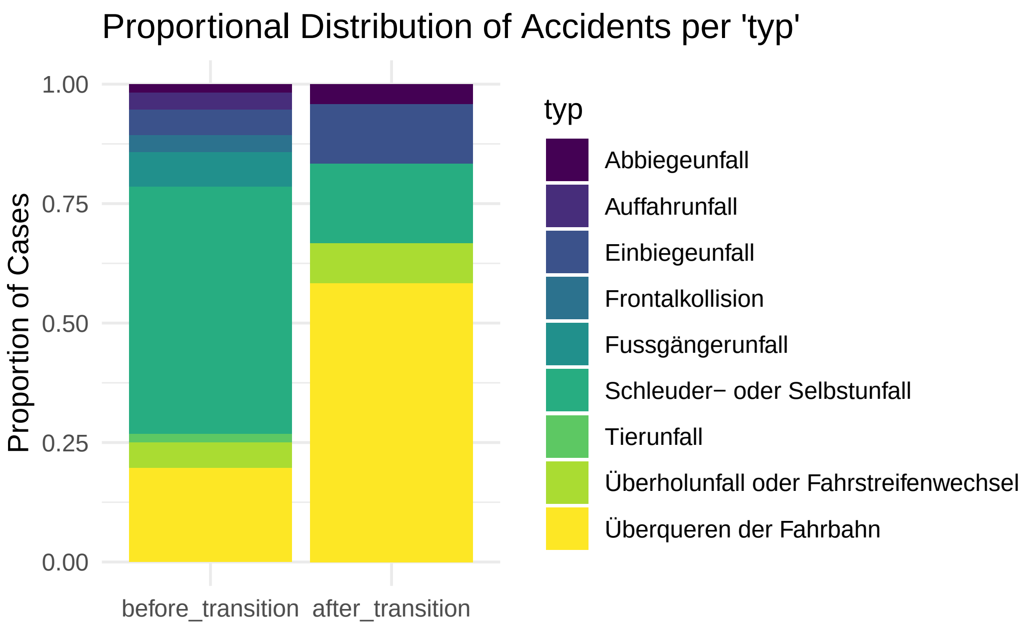
```
unfaelle_testing <- unfaelle_prep %>%
  pivot_longer(cols = c(before_transition, after_transition),
               names_to = 'before_after',
               values_to = "T_F") %>%
  subset(T_F == TRUE) %>%
  mutate(before_after = factor(before_after, levels = c("before_transition", "after_transiti

results <- t.test(relative_count ~ before_after, data = unfaelle_testing)

print(results)
```

```

    Welch Two Sample t-test

data:  relative_count by before_after
t = 0.63466, df = 41.399, p-value = 0.5291
alternative hypothesis: true difference in means between group before_transition and group a
95 percent confidence interval:
 -0.004675983  0.008963599
sample estimates:
mean in group before_transition  mean in group after_transition
                     0.01530627                      0.01316246
```

Absolute Plot

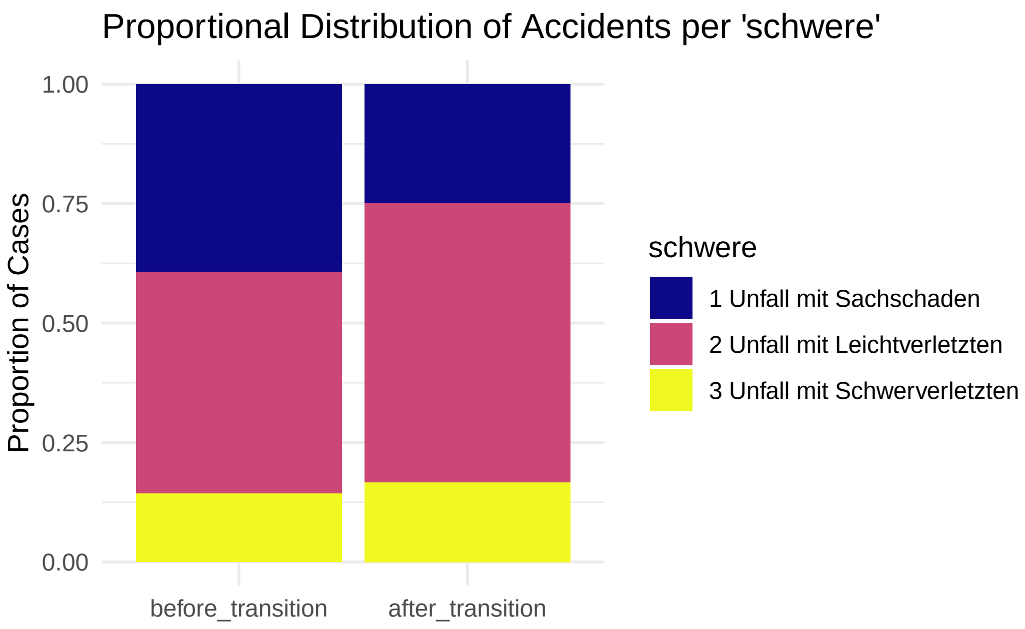
```r
unfaelle_clean_velostrasse_relative <- unfaelle_clean %>%
  filter(on_velostrasse == TRUE &
           typ != "Parkierunfall") %>%
  mutate(date_reference = as.Date(paste0(year, "-", month, "-01")),
    ym = as.yearmon(date_reference)) %>%
  left_join(average_zaehlung, by = "ym") %>%
  mutate(diff = ym_accident - ym,
    diff_months = round((ym_accident - ym) * 12))

unfaelle_clean_velostrasse_absolute <- unfaelle_clean_velostrasse_relative %>%
  filter(diff_months >= -60 & diff_months <= 24) %>%
  mutate(diff_months_group = floor(diff_months / x) * x) %>%
  mutate(after_transition = diff_months_group > 0,
         before_transition = diff_months_group < 0) %>%
  pivot_longer(cols = c(before_transition, after_transition),
               names_to = 'before_after',
               values_to = "T_F") %>%
  subset(T_F == TRUE) %>%
  mutate(before_after = factor(before_after, levels = c("before_transition", "after_transiti
```

```r
p <- unfaelle_clean_velostrasse_absolute %>%
  group_by(before_after, typ) %>%
  summarise(count = n()) %>%
ggplot(aes(before_after, count, fill = typ)) +
  geom_col(position = "fill") +
  scale_fill_viridis_d(option = "D") +
  ylab("Proportion of Cases") +
  xlab("") +
  ggtitle("Proportional Distribution of Accidents per 'typ'") +
  theme_minimal()

print(p)
```



```
ggsave("Plot_7.jpeg", plot = p, width = 12, height = 8, dpi = 600)
```

```
p <- unfaelle_clean_velostrasse_absolute %>%
  group_by(before_after, schwere) %>%
  summarise(count = n()) %>%
ggplot(aes(before_after, count, fill = schwere)) +
  geom_col(position = "fill") +
  scale_fill_viridis_d(option = "C") +
  ylab("Proportion of Cases") +
  xlab("") +
  ggtitle("Proportional Distribution of Accidents per 'schwere'") +
  theme_minimal()

print(p)
```



```r
ggsave("Plot_8.jpeg", plot = p, width = 12, height = 8, dpi = 600)
```