

# HW 03 - Road traffic accidents

Tina Huynh

## Contents

<b>1</b>	<b>Getting started</b>	<b>1</b>
1.1	Warm up . . . . .	2
1.2	Packages . . . . .	2
1.3	Data . . . . .	2
<b>2</b>	<b>Exercises</b>	<b>2</b>



Figure 1: Photo by Clark Van Der Beken on Unsplash

In this assignment we'll look at traffic accidents in Edinburgh. The data are made available online by the UK Government. It covers all recorded accidents in Edinburgh in 2018 and some of the variables were modified for the purposes of this assignment.

## 1 Getting started

**IMPORTANT:** If there is no GitHub repo created for you for this assignment, it means I didn't have your GitHub username as of when I assigned the homework. Please let me know your GitHub username asap,

and I can create your repo.

Go to the course GitHub organization and locate your homework repo, which should be named `hw-03-YOUR_GITHUB_USERNAME`. Grab the URL of the repo, and clone it in RStudio. First, open the R Markdown document `hw-03.Rmd` and Knit it. Make sure it compiles without errors. The output will be in the file markdown `.md` file with the same name.

## 1.1 Warm up

Before we introduce the data, let's warm up with some simple exercises.

- Update the YAML, changing the author name to your name, and **knit** the document.
- Commit your changes with a meaningful commit message.
- Push your changes to GitHub.
- Go to your repo on GitHub and confirm that your changes are visible in your Rmd **and** md files. If anything is missing, commit and push again.

## 1.2 Packages

We'll use the **tidyverse** package for much of the data wrangling and visualization and the data lives in the **dsbox** package. These packages are already installed for you. You can load them by running the following in your Console:

```
library(tidyverse)
library(dsbox)
```

## 1.3 Data

The data can be found in the **dsbox** package, and it's called **accidents**. Since the dataset is distributed with the package, we don't need to load it separately; it becomes available to us when we load the package. You can find out more about the dataset by inspecting its documentation, which you can access by running `?accidents` in the Console or using the Help menu in RStudio to search for **accidents**. You can also find this information [here](#).

## 2 Exercises

1. How many observations (rows) does the dataset have? Instead of hard coding the number in your answer, use inline code.

The dataset contains 768 observations.

**Tired of typing your password?** Chances are your browser has already saved your password, but if not, you can ask Git to save (cache) your password for a period of time, where you indicate the period of time in seconds. For example, if you want it to cache your password for 1 hour, that would be 3,600 seconds. To do so, run the following *in the Console*: `usethis::use_git_config(credential.helper = "cache --timeout=3600")`. If you want to cache it for a longer time, you can adjust the number of seconds in the code.

2. Run `View(accidents)` in your Console to view the data in the data viewer. What does each row in the dataset represent?

Each row in the dataset represents a distinct traffic accident that occurred in Edinburgh in 2018. Each accident entry includes detailed information about the time and location of the incident, weather and road conditions at the time, the severity of the accident, and various characteristics of the vehicles and casualties involved.

```
## Rows: 768
## Columns: 31
## $ id                <chr> "20189500000002", "20189500000006", "20189500000012", ~
## $ easting           <dbl> 327174, 324874, 330500, 321890, 320120, 331752, 325~
## $ northing          <dbl> 670941, 672457, 671750, 671640, 669330, 667988, 674~
## $ longitude         <dbl> -3.167032, -3.204252, -3.114026, -3.251772, -3.2794~
## $ latitude          <dbl> 55.92600, 55.93926, 55.93376, 55.93145, 55.91041, 5~
## $ police_force      <dbl> 95, 95, 95, 95, 95, 95, 95, 95, 95, 95, 95, 95, 95, ~
## $ severity          <fct> Slight, Slight, Slight, Slight, Slight, Slight, Sli~
## $ vehicles          <dbl> 1, 1, 2, 3, 2, 3, 1, 1, 1, 2, 2, 1, 1, 1, 2, 1, 2, ~
## $ casualties        <dbl> 1, 1, 2, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 4, 1, 1, 1, ~
## $ date              <chr> "31/12/2018", "30/12/2018", "03/01/2018", "01/01/20~
## $ day_of_week       <fct> Monday, Sunday, Wednesday, Monday, Thursday, Wednes~
## $ time              <time> 14:59:00, 12:50:00, 14:34:00, 02:25:00, 09:00:00, ~
## $ district          <dbl> 923, 923, 923, 923, 923, 923, 923, 923, 923, 923, 9~
## $ highway           <chr> "S12000036", "S12000036", "S12000036", "S12000036", ~
## $ first_road_class  <fct> Unclassified, Unclassified, A(M) road, A(M) road, U~
## $ first_road_number <chr> "0", "0", "6095", "71", "0", "720", "0", "0", "1", ~
## $ road_type         <fct> Single carriageway, Single carriageway, Single carr~
## $ speed_limit       <dbl> 20, 20, 20, 30, 30, 70, 20, 30, 20, 20, 20, 20, 20, ~
## $ junction_detail  <chr> "Other junction", "Other junction", "Crossroads", "~
## $ junction_control  <chr> "Give way or uncontrolled", "Give way or uncontroll~
## $ second_road_class <chr> "Unclassified", "Missing / Out of range", "A-road", ~
## $ second_road_number <chr> "0", "-1", "6106", "0", "0", "720", "0", "0", "0", ~
## $ ped_cross_human   <chr> "None within 50 metres", "None within 50 metres", "~
## $ ped_cross_physical <chr> "Pedestrian phase at traffic signal junction", "No ~
## $ light             <fct> Daylight, Daylight, Daylight, Darkness - lights lit~
## $ weather           <fct> Fine + no high winds, Fine + no high winds, Fine + ~
## $ road_surface      <fct> Dry, Dry, Wet or damp, Wet or damp, Wet or damp, We~
## $ special_condition <fct> None, None, None, None, None, None, None, None, Non~
## $ hazard            <fct> None, None, None, None, None, None, None, None, Non~
## $ urban_rural       <dbl> 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ police            <chr> "Yes", "Yes", "Yes", "Yes", "Yes", "Yes", "Yes", "Y~
```

```
##      id            easting      northing      longitude
## Length:768      Min.   :311270    Min.   :662344    Min.   : -3.422
## Class :character 1st Qu.:321858    1st Qu.:671458    1st Qu.: -3.252
## Mode  :character Median :324814    Median :673160    Median : -3.204
##                Mean  :323994    Mean  :672887    Mean   : -3.218
##                3rd Qu.:326882    3rd Qu.:674489    3rd Qu.: -3.173
##                Max.   :332720    Max.   :678755    Max.   : -3.079
##                NA's   :8         NA's   :8         NA's   :8
##      latitude    police_force    severity      vehicles      casualties
## Min.   :55.85    Min.   :95    Fatal   : 5    Min.    : 1.000    Min.    :1.000
## 1st Qu.:55.93    1st Qu.:95    Serious:115    1st Qu.: 1.000    1st Qu.:1.000
## Median :55.95    Median :95    Slight  :648    Median : 2.000    Median :1.000
## Mean   :55.94    Mean   :95                      Mean   : 1.743    Mean   :1.224
## 3rd Qu.:55.96    3rd Qu.:95                      3rd Qu.: 2.000    3rd Qu.:1.000
## Max.   :55.99    Max.   :95                      Max.   :10.000    Max.   :8.000
```

```

## NA's :8
##      date          day_of_week      time          district
## Length:768      Monday :136      Length:768      Min. :923
## Class :character Tuesday :127      Class1:hms      1st Qu.:923
## Mode :character Wednesday:110      Class2:difftime Median :923
##      Thursday :116      Mode :numeric      Mean :923
##      Friday :134      3rd Qu.:923
##      Saturday : 86      Max. :923
##      Sunday : 59
##      highway      first_road_class first_road_number
## Length:768      Motorway : 16      Length:768
## Class :character A(M) road :320      Class :character
## Mode :character B-road : 32      Mode :character
##      C-road : 98
##      Unclassified:302
##
##
##      road_type      speed_limit      junction_detail      junction_control
## Roundabout : 34      Min. :20.00      Length:768      Length:768
## One way street : 9      1st Qu.:20.00      Class :character      Class :character
## Dual carriageway : 75      Median :30.00      Mode :character      Mode :character
## Single carriageway:643      Mean :29.35
## Slip road : 7      3rd Qu.:30.00
##      Max. :70.00
##
## second_road_class second_road_number ped_cross_human      ped_cross_physical
## Length:768      Length:768      Length:768      Length:768
## Class :character      Class :character      Class :character      Class :character
## Mode :character      Mode :character      Mode :character      Mode :character
##
##
##
##      light          weather
## Daylight :590      Fine + no high winds :618
## Darkness - lights lit :147      Raining + no high winds: 65
## Darkness - lights unlit : 10      Unknown : 29
## Darkness - no lighting : 18      Other : 15
## Darkness - lighting unknown: 3      Fine + high winds : 14
##      Raining + high winds : 13
##      (Other) : 14
##      road_surface      special_condition
## Dry :586      None :753
## Wet or damp :164      Road sign or marking defective or obscured: 1
## Snow : 7      Roadworks : 11
## Frost or ice : 10      Road surface defective : 3
## Flood over 3cm deep: 1
##
##
##      hazard      urban_rural
## None :762      Min. :1.000
## Other object on road : 4      1st Qu.:1.000
## Previous accident : 1      Median :1.000
## Pedestrian in carriageway - not injured: 1      Mean :1.135

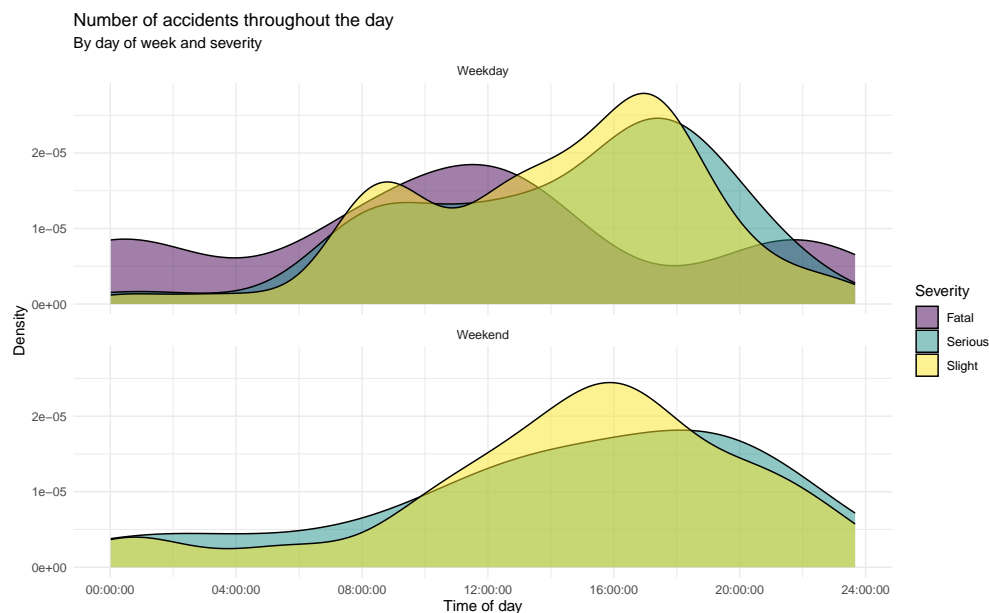
```

```
##                               3rd Qu.:1.000
##                               Max.    :3.000
##
##      police
## Length:768
## Class :character
## Mode  :character
##
##
##
```

Knit, commit, and push your changes to GitHub with an appropriate commit message. Make sure to commit and push all changed files so that your Git pane is cleared up afterwards.

3. Recreate the following plot, and describe in context of the data. In your answer, don't forget to label your R chunk as well (where it says `label-me-1`). Your label should be short, informative, shouldn't include spaces, and shouldn't shouldn't repeat a previous label.

```
accidents %>%
  mutate(day_of_week_type = if_else(day_of_week %in% c("Saturday", "Sunday"), "Weekend", "Weekday")) %>%
  ggplot(aes(x = time, fill = severity)) +
  geom_density(alpha = 0.5) +
  facet_wrap(~day_of_week_type, ncol = 1) +
  theme_minimal() +
  scale_fill_viridis_d() +
  labs(
    title = "Number of accidents throughout the day",
    subtitle = "By day of week and severity",
    x = "Time of day",
    y = "Density",
    fill = "Severity"
  )
)
```



The density plot above reveals critical temporal patterns in Edinburgh’s traffic accidents, comparing weekday and weekend incidents by time of day and severity.

On weekdays, the data shows two prominent accident peaks: one during the morning rush hour (approximately 8-9 AM) and a larger, more sustained peak during the afternoon/evening commute (3-6 PM). This bimodal distribution strongly correlates with typical commuting patterns, suggesting that traffic congestion during work commutes significantly contributes to accident frequency. Notably, the afternoon peak is more pronounced and extends over a longer period, potentially reflecting the less synchronized nature of evening commutes compared to morning ones.

Weekend accident patterns differ substantially, with accidents more evenly distributed throughout the day and reaching their highest frequency in the afternoon and early evening hours. The absence of a morning peak on weekends reflects the reduced early-day traffic when commuting pressures are absent. Instead, the gradual increase in accidents throughout the day likely corresponds to increased leisure travel, shopping activities, and potentially social events later in the day.

Regarding severity, “Slight” accidents (indicated in purple) constitute the majority of incidents across both weekday and weekend periods. However, “Serious” accidents (shown in green-blue) maintain a relatively consistent proportion throughout the day. Fatal accidents, while rare, do not show a clear temporal pattern in this visualization, which is expected given their fortunately infrequent occurrence.

The overall lower density of weekend accidents compared to weekday incidents suggests that commuter traffic significantly influences accident rates in Edinburgh. This comprehensive temporal analysis provides valuable insights for traffic management authorities, who might consider targeted safety measures during high-risk periods, particularly weekday commuting hours when accident densities reach their peak.

*Knit, commit, and push your changes to GitHub with an appropriate commit message. Make sure to commit and push all changed files so that your Git pane is cleared up afterwards.*

4. Create another data visualization based on these data and interpret it. You can choose any variables and any type of visualization you like, but it must have at least three variables, e.g. a scatterplot of x vs. y isn’t enough, but if points are colored by z, that’s fine. In your answer, don’t forget to label your R chunk as well (where it says `label-me-2`).

```
# First, examine the speed limit and weather distributions
accidents %>%
  count(speed_limit) %>%
  arrange(desc(n))
```

```
## # A tibble: 6 x 2
##   speed_limit     n
##   <dbl> <int>
## 1      20    379
## 2      30    246
## 3      40     64
## 4      70     47
## 5      50     19
## 6      60     13
```

```
accidents %>%
  count(weather) %>%
  arrange(desc(n))
```

```
## # A tibble: 9 x 2
##   weather     n
```

```
##    <fct>                <int>
## 1 Fine + no high winds    618
## 2 Raining + no high winds  65
## 3 Unknown                 29
## 4 Other                   15
## 5 Fine + high winds       14
## 6 Raining + high winds    13
## 7 Snowing + no high winds  11
## 8 Snowing + high winds     2
## 9 Fog or mist             1
```

```
# Filter out the very infrequent categories for clearer visualization
```

```
accidents_filtered <- accidents %>%
```

```
  filter(weather %in% c("Fine + no high winds", "Raining + no high winds", "Fine + high winds", "Raining + high winds"))
  mutate(
```

```
    weather_type = case_when(
      str_detect(weather, "Fine") ~ "Fine",
      str_detect(weather, "Rain") ~ "Rainy",
      TRUE ~ weather
    ),
```

```
    wind_condition = if_else(str_detect(weather, "high winds"), "High winds", "No high winds"),
```

```
    speed_category = case_when(
      speed_limit <= 20 ~ "20 mph or less",
      speed_limit <= 30 ~ "30 mph",
      speed_limit <= 40 ~ "40 mph",
      TRUE ~ ">40 mph"
    ),
```

```
    speed_category = factor(speed_category, levels = c("20 mph or less", "30 mph", "40 mph", ">40 mph"))
  )
```

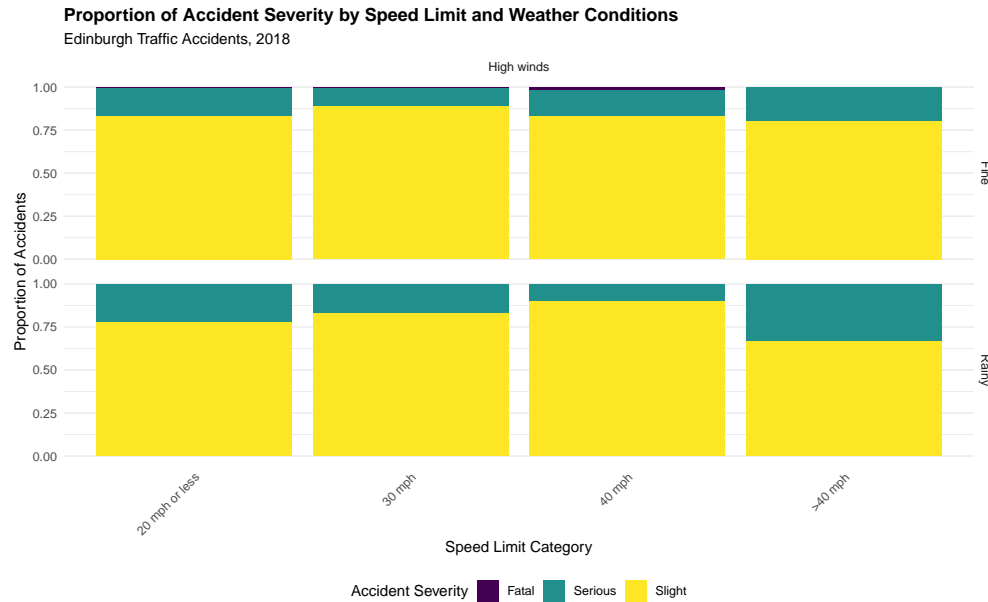
```
# Create visualization of weather conditions, speed limits, and accident severity
```

```
ggplot(accidents_filtered, aes(x = speed_category, fill = severity)) +
```

```
  geom_bar(position = "fill") +
  facet_grid(weather_type ~ wind_condition) +
  scale_fill_viridis_d() +
  theme_minimal() +
  labs(
```

```
    title = "Proportion of Accident Severity by Speed Limit and Weather Conditions",
    subtitle = "Edinburgh Traffic Accidents, 2018",
    x = "Speed Limit Category",
    y = "Proportion of Accidents",
    fill = "Accident Severity"
  ) +
```

```
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.title = element_text(face = "bold"),
    panel.grid.major.x = element_blank(),
    legend.position = "bottom"
  )
```



This visualization examines the relationship between speed limits, weather conditions, and accident severity in Edinburgh's 2018 traffic incidents, revealing several noteworthy patterns with significant road safety implications.

The analysis reveals that roads with higher speed limits consistently demonstrate greater proportions of serious and fatal accidents across all weather conditions. This progressive relationship is particularly evident in the 40 mph and >40 mph categories, where the proportion of severe outcomes increases substantially compared to lower-speed zones. This pattern aligns with established traffic safety principles that higher speeds generate greater kinetic energy during collisions, resulting in more severe injuries.

Weather conditions exhibit a complex influence on accident severity. Fine weather without high winds shows relatively consistent severity distributions across speed categories, likely reflecting "baseline" accident patterns. However, the introduction of high winds significantly alters this pattern, particularly in higher speed zones. In 40 mph zones during fine weather with high winds, the proportion of serious accidents increases noticeably compared to the same speed zones without high winds. This suggests that wind conditions may compromise vehicle stability or driver control, especially at higher speeds.

Rainy conditions demonstrate an interesting pattern: they generally show higher proportions of slight accidents in the 30 mph zones compared to fine weather. This counterintuitive finding may reflect adaptive behavior, where drivers exercise greater caution during visible adverse conditions, potentially reducing accident severity despite increased occurrence. However, when rain combines with high winds in 20 mph or less zones, we observe a notably higher proportion of serious accidents, indicating that this combination of adverse conditions may be particularly hazardous in city centers and residential areas with lower speed limits.

The data shows few fatal accidents across all categories, making pattern identification difficult for this severity level. However, their relative absence in the 20 mph or less zones across all weather conditions reinforces the safety benefits of lower urban speed limits.

These findings have significant implications for traffic management policy. They suggest that speed limit reductions could be particularly beneficial during adverse weather conditions, and that warning systems for high wind conditions might be especially important on higher-speed roadways. The data also supports the continued implementation of 20 mph zones in urban areas, which consistently show the lowest proportion of serious and fatal accidents across all weather conditions analyzed.

*Knit, commit, and push your changes to GitHub with an appropriate commit message. Make sure to commit and push all changed files so that your Git pane is cleared up afterwards and review the md document*



*on GitHub to make sure you're happy with the final state of your work.* Once you decide that you are done with the lab, choose the knit drop down and select **Knit to tufte\_handout** to generate a pdf. Download and submit that pdf to Canvas.