# Reducing the number of high fatality accidents in UK

## 1. Background

The safety team classes major incidents as fatal accidents involving 3+ casualties. They are trying to learn more about the characteristics of these major incidents so they can brainstorm interventions that could lower the number of deaths.

Therefore, the purpose of this analysis was to discover any insights from a supplied accident dataset that would help the road safety team in the Department of Transport reduce the number of major incidents with answering a number of questions:

1. When do most major accidents occur?
2. Are there any patterns in the time of day and day of the week when major incidents occur?
3. What characteristics stand out in major incidents compared with other accidents?
4. Where should the planning team focus their brainstorming efforts to reduce major incidents?

## 2. The data

The reporting department have been collecting data on every accident that is reported. They've included this along with a lookup file for 2020's accidents.

*Published by the department for transport. https://data.gov.uk/dataset/road-accidents-safety-data Contains public sector information licensed under the Open Government Licence v3.0.*

The accidents table has 27 variables and 91199 observations. The lookup table has 5 variables, and 129 observations contains the metadata for the main data set, accidents.

```
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✓ ggplot2 3.3.5     ✓ purrr   0.3.4
## ✓ tibble  3.1.6     ✓ dplyr   1.0.8
```

```
## ✓ tidyr   1.2.0     ✓ stringr 1.4.0
## ✓ readr   2.1.2     ✓ forcats 0.5.1

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

## Read in the accidents data from the data folder.
accidents <- readr::read_csv("data/accident-data.csv", na = "-
1",show_col_types = FALSE)

## Warning: One or more parsing issues, see `problems()` for details

head(accidents)

## # A tibble: 6 × 27
##   accident_index accident_year accident_reference longitude latitude
##   <chr>                  <dbl> <chr>                  <dbl>    <dbl>
## 1 2020010219808           2020 10219808            -0.254       51.5
## 2 2020010220496           2020 10220496            -0.139       51.5
## 3 2020010228005           2020 10228005            -0.179       51.5
## 4 2020010228006           2020 10228006            -0.00168     51.5
## 5 2020010228011           2020 10228011            -0.138       51.5
## 6 2020010228012           2020 10228012            -0.0259      51.5
## # … with 22 more variables: accident_severity <dbl>, number_of_vehicles
<dbl>,
## #   number_of_casualties <dbl>, date <chr>, day_of_week <dbl>, time
<time>,
## #   first_road_class <dbl>, first_road_number <dbl>, road_type <dbl>,
## #   speed_limit <dbl>, junction_detail <dbl>, junction_control <dbl>,
## #   second_road_class <dbl>, second_road_number <dbl>,
## #   pedestrian_crossing_human_control <dbl>,
## #   pedestrian_crossing_physical_facilities <dbl>, light_conditions <dbl>,
…

lookup <- readr::read_csv('./data/road-safety-lookups.csv',
                          show_col_types = FALSE)
head(lookup)

## # A tibble: 6 × 5
##   table    `field name`       `code/format` label note
##   <chr>    <chr>              <chr>         <chr> <chr>
## 1 Accident accident_index     <NA>          <NA>  unique value for each
acciden…
## 2 Accident accident_year      <NA>          <NA>  <NA>
## 3 Accident accident_reference <NA>          <NA>  In year id used by the
police…
## 4 Accident longitude          <NA>          <NA>  Null if not known
## 5 Accident Latitude           <NA>          <NA>  Null if not known
## 6 Accident accident_severity  1             Fatal <NA>
```

## 2.1 Cleanning data.

Based on the lookup table, besides there were missing values in the accidents table, variables with a code '-1' were considered missing value or out of range.

```
#Find missing values in all columns.

sapply(accidents, is.na) %>%

  colSums() %>%

  tibble(variable= names(accidents), missing = .) %>%

  arrange(desc(missing)) %>%

  filter(missing > 0)

## # A tibble: 13 × 2
##    variable                                  missing
##    <chr>                                       <dbl>
##  1 junction_control                            38298
##  2 road_surface_conditions                       316
##  3 special_conditions_at_site                    218
##  4 carriageway_hazards                           208
##  5 pedestrian_crossing_human_control             143
##  6 pedestrian_crossing_physical_facilities       135
##  7 longitude                                      14
##  8 latitude                                       14
##  9 speed_limit                                    12
## 10 second_road_number                              7
## 11 junction_detail                                 2
## 12 light_conditions                                1
## 13 weather_conditions                              1
```

We find total 39369 missing values. However, the variable "junction_control" was given the sizable percentage (43.7%) of missing or unknown observations; therefore to avoid losing a significant amount of data, we keep all missing values in "junction_control" variable.

```
# remove missing values but keep NA in "junction_control" variable which
replace by "-1".

accidents <- accidents %>%

  mutate(junction_control = replace_na(junction_control, -1)) %>%

  na.omit()
```

Data after removing missing values reduce the size to 90706 observations.

### 2.2 Convert to appropriate values.

```
## Convert date and time to date/time format

accidents$hour=format(as.POSIXct(accidents$time), format = "%H") # group into
hour

accidents$date <- as.Date(accidents$date, format = "%d/%m/%Y")

accidents$month <- format(accidents$date,format ="%m") # group the dates into
months

accidents$day_of_week = factor(accidents$day_of_week,

                              labels = c("Sun", "Mon", "Tue",

                                         "Wed", "Thur", "Fri", "Sat"))
## Covert values of accident severity

accidents$accident_severity=factor(accidents$accident_severity,labels=c("Fata
l","Serious","Slight"))
```

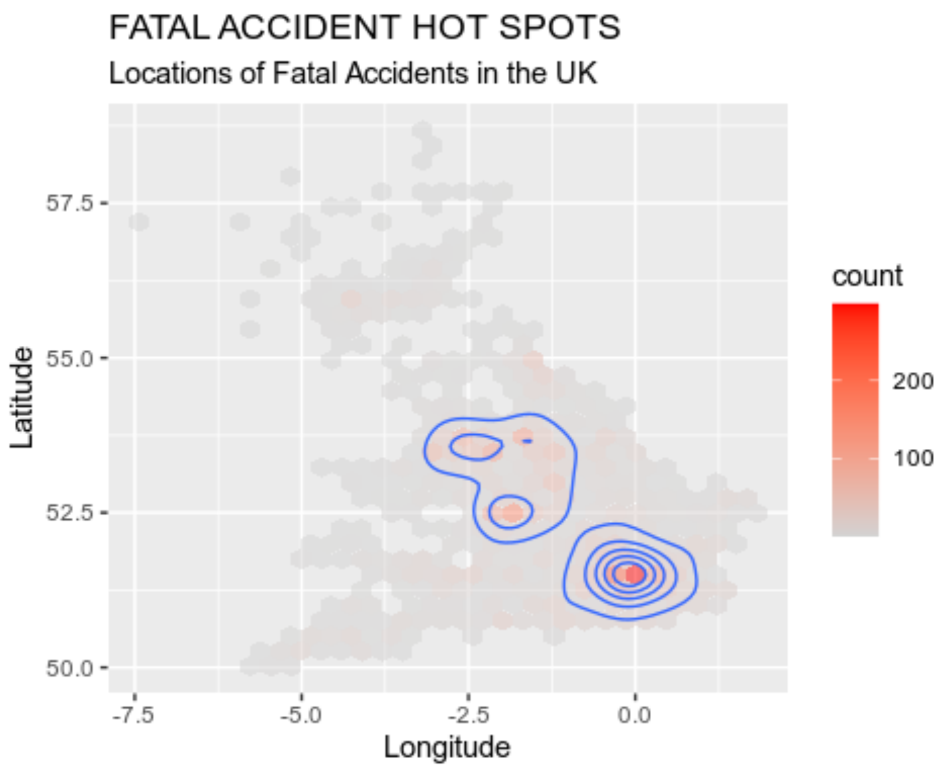
## 3. The Analyzis.

### 3.1. Where do most major accidents occur?

We map out the areas where major fatal accidents happen- the hotspots by plotting the map of major accidents. Remember that in this analysis, the major accidents involved 3+ casualties.

```
library(hexbin) # install library for geom_hex.

accidents %>%

  ## Filter out major accidents
        filter(number_of_casualties >= 3) %>%

  ## Plot locations of major accidents
        ggplot(aes(x = longitude, y = latitude)) +

  ## Add a hex geom
         geom_hex(alpha = 0.5) +
```

```
  ## Add a gem dendity 2D
       geom_density_2d() +

   ## Add labels and titles
       labs(x = "Longitude", y = "Latitude",

      title = "FATAL ACCIDENT HOT SPOTS",

      subtitle = "Locations of Fatal Accidents in the UK")+

    scale_fill_gradient(low = "lightgray", high = "red")
```

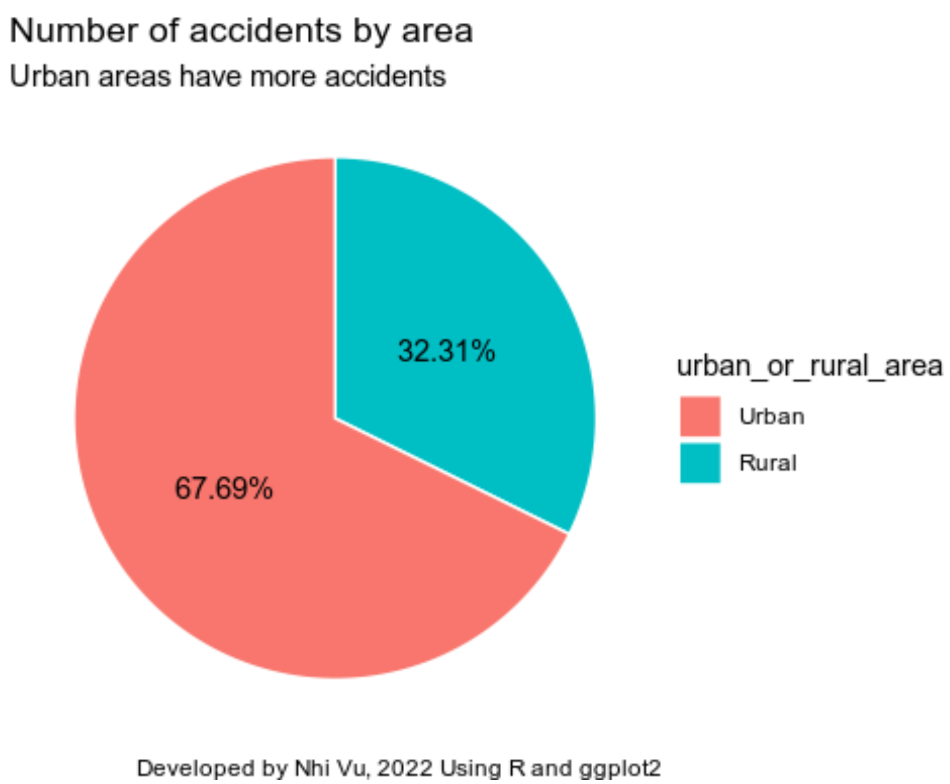



Major accidents happen all over the UK. However, the graph shows two primary regions with an unusually high concentration of accidents and fatalities. One of these regions is in the south-east (presumably around London) while the other region spans Central UK. The accident hot spot in the central UK has three key subareas where accidents concentrate. There are additional pockets of accident hotspots, one towards the North East and another to the North (Scotland).

Commonly, accidents often happen more frequently in Urban than in Rural which is not different from in UK. We find 67.69% accidents happened in UK Urban areas.

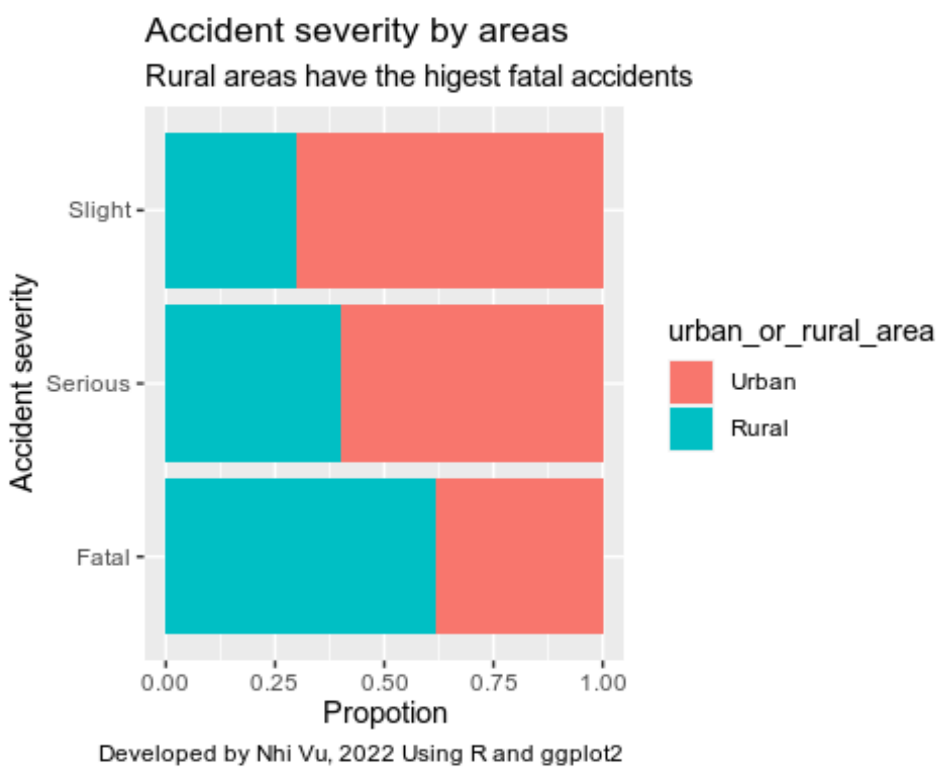
```
# Number of accidents in urban and rural_area, which code for Urban is 1, and
for Rural is 2.
```

```r
accidents$urban_or_rural_area=factor(accidents$urban_or_rural_area,labels=c("
Urban","Rural"))

df<- accidents %>% group_by(urban_or_rural_area) %>%
  summarise(count=n()) %>%
  arrange(desc(urban_or_rural_area))%>%
  mutate(percentage=round(count *100 /sum(count),2),
         lab.ypos = count/2 +c(0,cumsum(count)[-length(count)]))


#Visualization
ggplot(df, aes(x="", y=count, fill= urban_or_rural_area)) +
  geom_bar(stat="identity", width=1, color="white") +
  coord_polar("y", start=0) +
  theme_void()+ # remove background, grid, numeric labels
  geom_text(aes(y = lab.ypos,
            label = paste0(percentage, "%")))+
  labs( title = "Number of accidents by area",
        subtitle="Urban areas have more accidents",
        caption = "Developed by Nhi Vu, 2022 Using R and ggplot2")
```



However, splitting accidents by urban and rural locations shows a clear difference in the high fatal crash group relative to others. Rural areas take more than 50% of fatal accidents while the majority of serious and slight accidents happen in the urban areas.

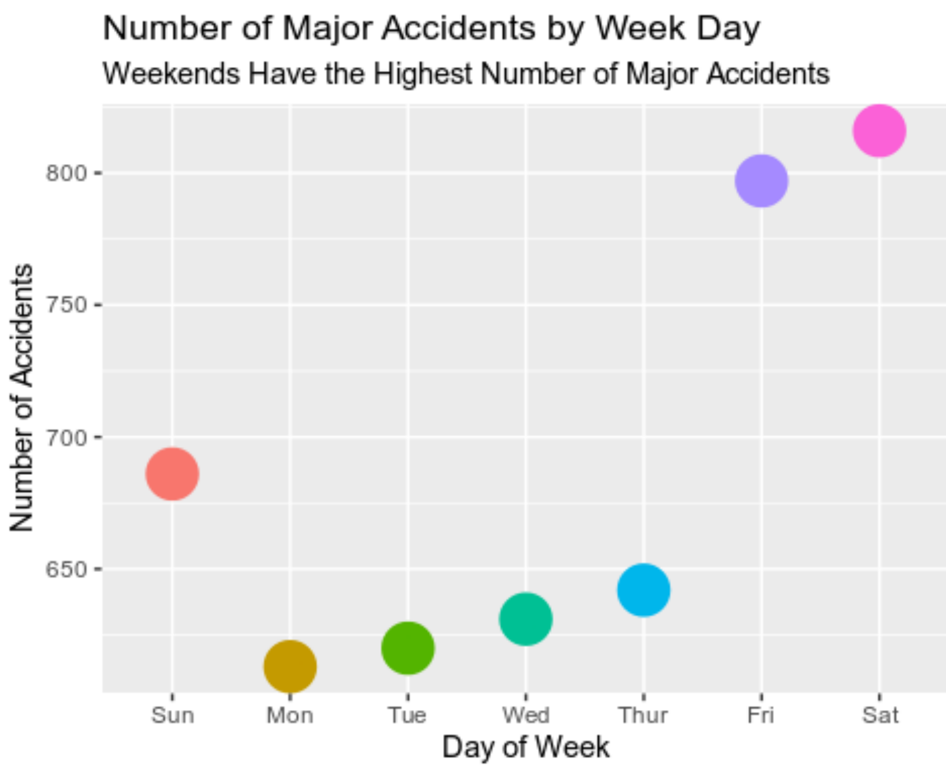
```
accidents %>% group_by(accident_severity,urban_or_rural_area)%>%
              summarise(count=n())%>%
              ggplot(aes(y=accident_severity,x=count,fill=
urban_or_rural_area))+
              geom_bar(position="fill", stat="identity") +
              labs(x = "Propotion", y = "Accident severity",
                   title = "Accident severity by areas",
                   subtitle="Rural areas have the higest fatal accidents",
                   caption = "Developed by Nhi Vu, 2022 Using R and ggplot2")

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




## 3.2. Are there any patterns in the time of day and day of the week when major incidents occur?

```
# No. accidents by day
accidents %>%
  filter(number_of_casualties >= 3) %>%  # Only major accidents with
casualties >= 3.
```

```
  group_by(day_of_week) %>%
  summarise(accidents = n()) %>%
  ggplot(aes(x = day_of_week, y = accidents,col = factor(day_of_week))) +
  geom_point(size = 4,stroke = 4, show.legend = FALSE)+
  labs(x = "Day of Week", y = "Number of Accidents",
       title = "Number of Major Accidents by Week Day",
       subtitle = "Weekends Have the Highest Number of Major Accidents")
```



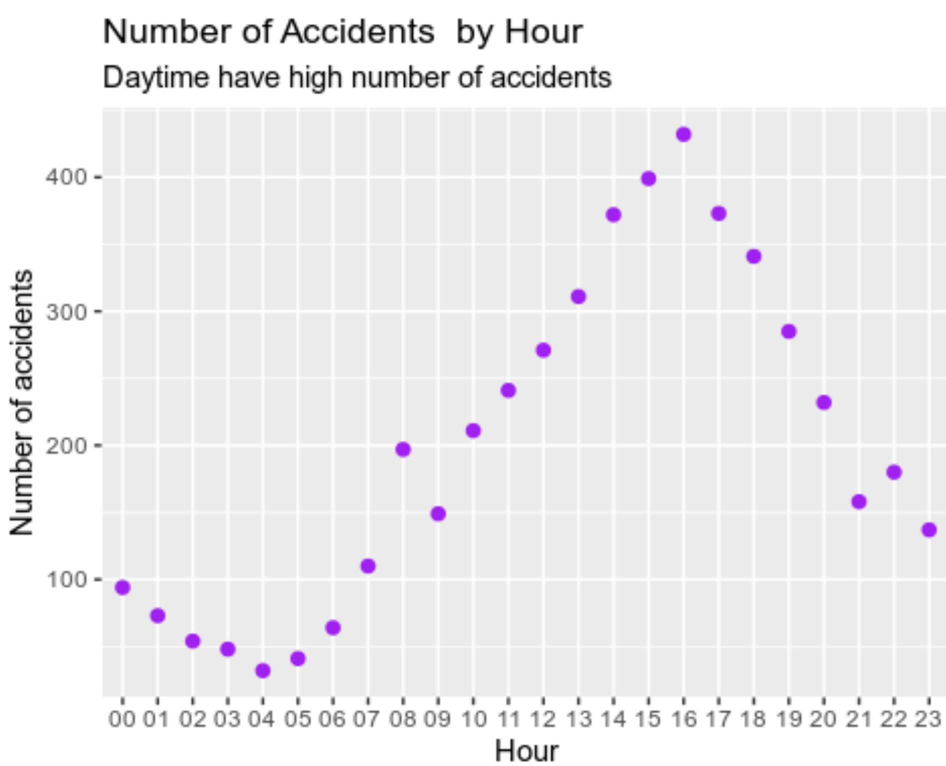

The graph of the number accident by day shows that weekend has highest accidents than other days on the week. Accidents rise gradually during the week and then spike on Friday, peaking on Saturday. The numbers then reduce on Sunday (to levels only lower than Friday and Saturday) and slump back to their minimum on Monday.

```
# Major Accidents and Fatalities by Week Day
table1 <-accidents %>%
  filter(number_of_casualties >= 3)%>%
  group_by(day_of_week)%>%
  summarise(accidents=n(),fatalities=sum(number_of_casualties)) %>%
  arrange(desc(accidents))
data.frame(table1)

##   day_of_week accidents fatalities
## 1         Sat       816       2882
## 2         Fri       797       2768
## 3         Sun       686       2470
## 4        Thur       642       2262
## 5         Wed       631       2228
```

```
## 6          Tue        620        2183
## 7          Mon        613        2141
```

The table above also indicates major accidents mostly happens on Weekend, especially Saturday which has a highest number of causalities and major accidents.

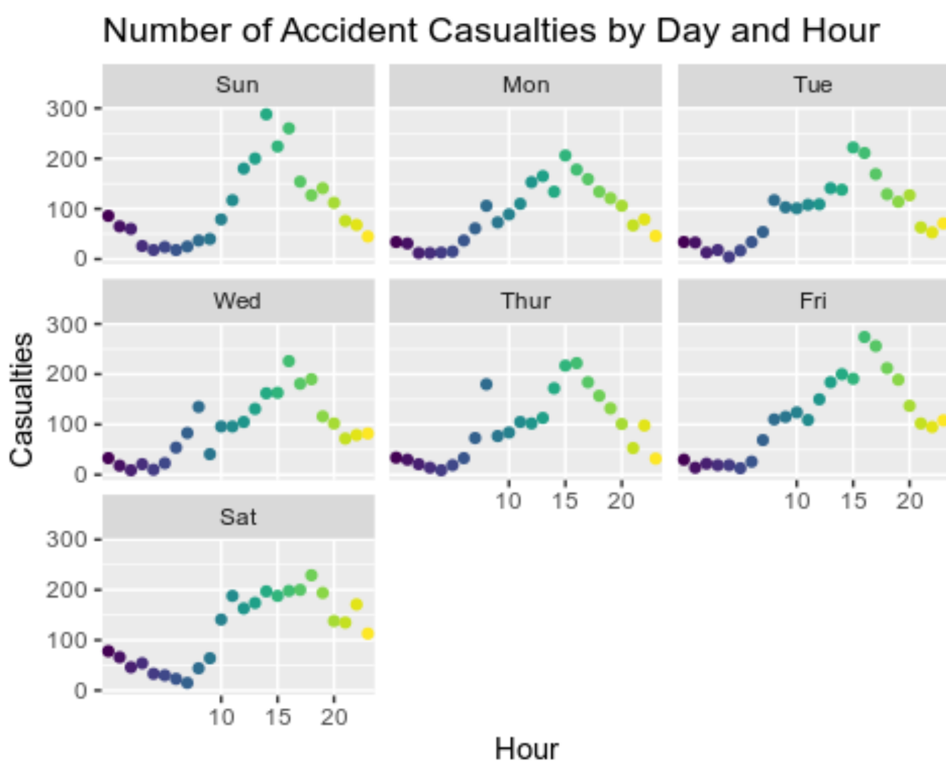
```
# Visualization number of major accident by hours
accidents %>%
  filter(number_of_casualties >= 3) %>%  # We focus on only major accidents with casualties >= 3.
  group_by(hour)%>%
  summarise(accidents_by_hour = n())%>%
  ggplot(aes(x=hour,y=accidents_by_hour))+
  geom_point(size=2, col="purple") +
  labs(x = "Hour", y = "Number of accidents",
       title = "Number of Accidents  by Hour",
       subtitle="Daytime have high number of accidents ")
```




Because of more vehicles on the street which increase the risk of accidents, daytime has a significantly high number of accidents. At peak hours, 8:00 am and from 2:00 pm to 6:00 pm, the number of accidents is very high. Accidents rise From 0:00 am to 6 am, accidents rise gradually, and then spike at 7 am, peaking at 8 am. Even though the number goes down after 8:00 am, it is still kept at a high level, then suddenly goes up significantly to reach the highest points at an interval 3:00 pm to 6:00 pm. After that, the number of accidents decrease gradually.

Next, we break down accident casualties by the time of day and day of the week.

```
accidents %>%
  filter(number_of_casualties >= 3)%>% # Focus on only major accident which causalities 3+
  group_by(day_of_week, hour)%>%
  summarise(casualities=sum(number_of_casualties)) %>%
  ggplot(aes(x=hour,y=casualities,col = factor(hour)))+
           geom_point(show.legend = FALSE)+
           facet_wrap(~ day_of_week)+
           scale_x_discrete(breaks = seq(0,23, 5))+
           scale_color_viridis_d() +
           labs(x = "Hour", y = "Casualties",
             title = "Number of Accident Casualties by Day and Hour")

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




From the graph, we find that working days ( Monday to Friday) have a similar trend that follows what we analyze from the graph of "Number of accidents by the hour", reaching first high level at around 8:00 am, then reducing before rising gradually to reach a peak from 3 pm to 6 pm.

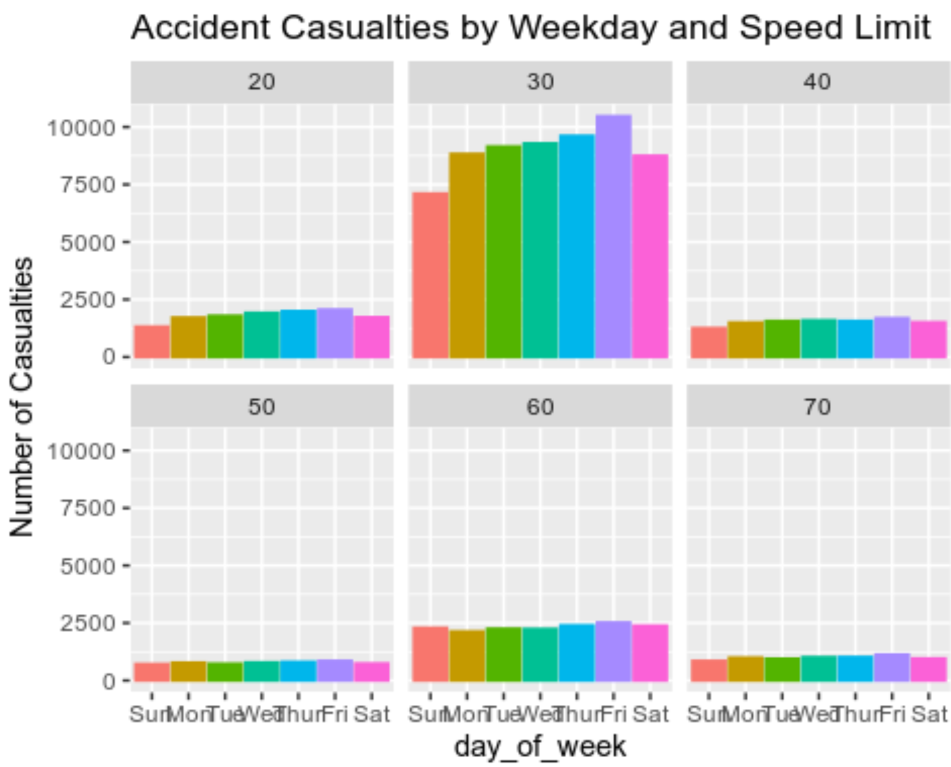
However, Saturdays and Sundays have a markedly different pattern. Accidents on Sundays have one peak period that begins around midday and peaks at 3:00 pm, followed by a rapid drop. Similarly, Saturday has one peak in the number of accident casualties. However, the peak accident period is prolonged, lasting from around 11.00 hours to 6:00 p, after which there is a gradual decline.

## 3.3. What characteristics stand out in major incidents compared with other accidents?

As we assume that major accidents have number of causalities more than 3, so minor accidents have number of causalities less than 3. Next, we will compare their characteristics, so we can brainstorm to find the solution that reduce number of major accidents.

*Speed limit*

```
accidents %>%

        ## Create a column of major and minor accidents
        mutate(major = case_when(

                number_of_casualties >= 3 ~ "Major",

                TRUE ~ "Minor"
        )) %>%

        ## Group by day of week and speed limit
        group_by(day_of_week, speed_limit) %>%

        ## Summarise total casualties
        summarise(total = sum(number_of_casualties)) %>%

        ## Plot day of week versus casualties
        ggplot(mapping = aes(x = day_of_week, y = total,

                             col = day_of_week, fill = day_of_week)) +

        geom_col(show.legend = FALSE) +

        facet_wrap(~ speed_limit) +

        labs(y = "Number of Casualties",

             title = "Accident Casualties by Weekday and Speed Limit")

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



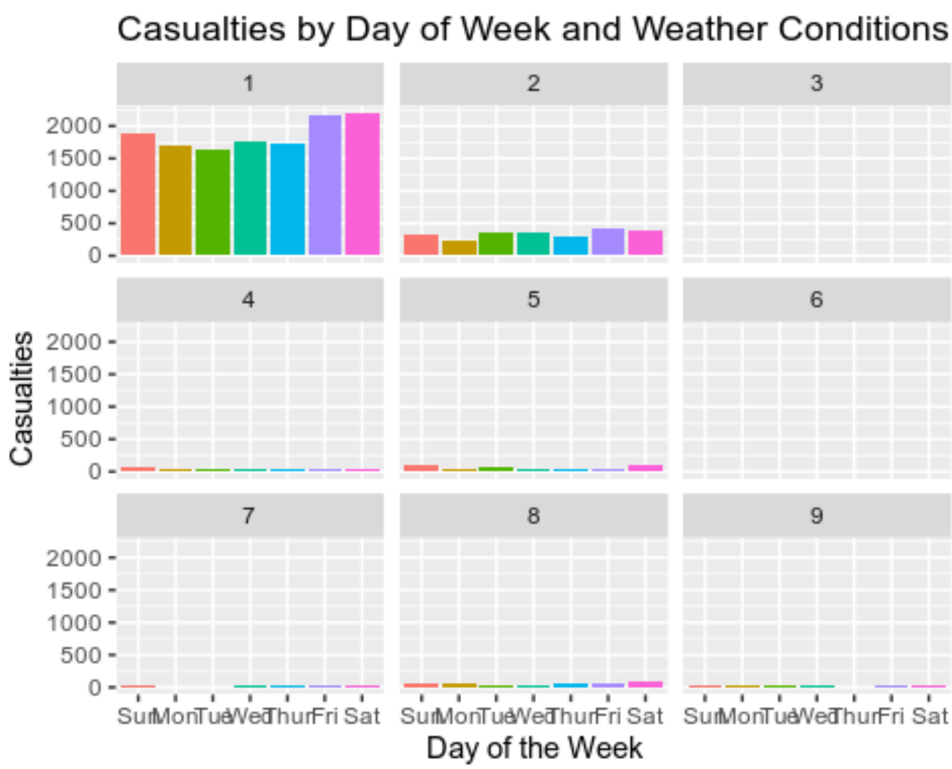

From the graph, We find the road stretches with a speed limit of 30 mph have the highest incident of accidents and accident casualties across all days of the week.

*Major Accident Casualties and Weather Conditions*

Most accidents on any day of the week happen when the weather is fine with no high winds. The second riskiest weather pattern is raining with no high winds. It would follow that people are most likely to speed when the weather is clear, hence the high number of casualties.

```
accidents %>%

        ## Get the major accidents
        filter(number_of_casualties >= 3) %>%

        ## Group by weather conditions and day of the week
        group_by(weather_conditions, day_of_week) %>%

        ## Summarise total casualties
        summarise(casualties = sum(number_of_casualties)) %>%

        ## Create new variable for proportion of casualties
        mutate(perc_casualties = casualties / sum(casualties) * 100) %>%

        ## Plot casualties versus day of week
        ggplot(mapping = aes(x = day_of_week, y = casualties,
```

```
                     fill = day_of_week)) +
      geom_col(show.legend = FALSE) +
      facet_wrap(~ weather_conditions) +
      labs(x = "Day of the Week", y = "Casualties",
           title = "Casualties by Day of Week and Weather Conditions")
## `summarise()` has grouped output by 'weather_conditions'. You can override
## using the `.groups` argument.
```




## Conclusion

In this analysis, I examined the road accidents data from the UK. The study of the data provided some valuable insights, the major ones being;

Accident casualties peak during weekends, Starting from Friday and falling on Sunday. Accidents casualties vary by time of day.

Major accidents and accident casualties mainly happen when the weather is fine. Major accidents and accident casualties mainly occur in road stretches with speed limits of 30 mph and 60 mph.

The recommendations to reduce significant accidents casualties should hence draw from these insights. None of the proposals would work very well in isolation. What is needed is a package of interventions that would help lower the tide of casualties from significant accidents.