Sprint -4

Date	24 November 2022
Team ID	PNT2022TMID29355
Project Name	Project – Signs with Smart Connectivity for Better Road Safety

- Project main
- Road safety
- Final_projects Codes
- Lable names (Excel)
- Thanks_Regard’s

1. Project main

```
---
always_allow_html: yes
output:
  html_document: default
  pdf_document: default
---

<style type="text/css"> body{
   /* Normal */ font-size:
   12px; margin-left: 20px;
  }
.column-left{ float:
  left;
  width: 40%;
  text-align: left;
}
.column-right{ float:
  right;
  width: 60%;
  text-align: right;
}
</style>
```

~~~
```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
packages <- c("plotly", "tidyverse",
"ggmap", "GGally", "gridExtra",
"scales", "viridis") newPackages <-
packages[!(packages %in%
installed.packages()[,"Package"])]
if(length(newPackages))
install.packages(newPackages)
library(tidyverse) library(plotly)
library(gridExtra) library(scales)
library(GGally) library(viridis)
library(ggmap)
load("passfail.RData")
```
~~~

~~~
```{r,echo=FALSE}
load("passfail.RData")
passfail <- passfail %>%
  mutate(totalFails = Fail1 + ifelse(is.na(Fail2), 0, Fail2), Totalpass = Pass1 + ifelse(is.na(Pass2),
0, Pass2))
```
~~~

~~~
```{r,echo=FALSE}
passfailGroup <- summarise(group_by(passfail, Centre), Pass1 = sum(Pass1), Fail1 = sum(Fail1),
Total1 = sum(Total1), Pass2 = sum(Pass2, na.rm = T), Fail2 = sum(Fail2, na.rm = T), Total2 =
sum(Total2, na.rm = T), Totalpass = sum(Totalpass), totalFails = sum(totalFails)) passfailGroup
<- mutate(passfailGroup, Pass1prop = Pass1/Total1, Pass2prop = Pass2/Total2, totalPassProp =
(Totalpass / (Total1 + Total2)), totalFailsProp = (totalFails / (Total1 + Total2)))
```
~~~

~~~
```{r,echo=FALSE}
passfailGroup$totalPassProp = round((passfailGroup$totalPassProp * 100), digits = 2)
passfailGroup$totalFailsProp = round((passfailGroup$totalFailsProp * 100), digits = 2)
passFailGroup1 <- passfailGroup[c(1, 8)] passFailGroup1$Test <- "Pass"
names(passFailGroup1) <- c("Centre", "Count", "Test") passFailGroup2 <-
passfailGroup[c(1, 9)] passFailGroup2$Test <- "Fail"
names(passFailGroup2) <- c("Centre", "Count", "Test")
passFailcount <- rbind(passFailGroup1, passFailGroup2)
```
~~~

### Analysis based on test centres

In this section we will analyse data from 2013 till 2018 about each test centre. As shown in the <a href = "https://github.com/NanawareAmol/R-project_Road-

safety/blob/master/Result/loc_spread_across_ireland.JPG">map</a>, the test centres are spread across the Ireland and the number of centres is more in highly populated areas such as dublin, cork etc.
The bar chart shows the total number of tests that each centre performed and the total pass and fail counts as well as percentages. So, based on the test counts, the top 3 test centre are, *Fonthill(770685)*, *Deansgrade(767484)*, and *Northpoint 2(729661)*. The botton 3 centres which performed less tests are, *Donegal Town(16315)*, *Cahirciveen(28806)* and *Clifden(38683)*.

```{r,echo=FALSE, fig.width=9,fig.height=3} t <- list(size = 8) p <- plot_ly(passfailGroup, x
= ~passfailGroup$Centre, y = ~passfailGroup$Totalpass, type =
'bar', name = 'Pass', text = paste("Total tests = ",
(passfailGroup$Totalpass+passfailGroup$totalFails), "<br>Passed =",
passfailGroup$totalPassProp,"%", "<br>Failed =", passfailGroup$totalFailsProp,"%"), opacity =
0.5, marker = list(color = '#3AC3E3', line = list(color = '#0D6EB0', width = 1))) %>%
add_trace(y = ~passfailGroup$totalFails, name = 'Fails', opacity = 0.5, marker = list(color =
'#0E84FF', line = list(color = '#0D6EB0', width = 1))) %>% layout(yaxis = list(title = 'Count'),
xaxis = list(title = 'Test Centres'), barmode = 'stack', font = t) p

```
<hr style = "margin: 10px 0px 10px;">
<div style = "display: inline-block;float: left;width: 50%;">
#### <b>Total test passed for each test centre</b>
The following scatter plot show the total test pass count for each test centre from the year 2013 till year 2018. The questions that can be answered by this graph are, <br/>
1. which are the top 3 and last 3 centres based on total pass count?<br/>
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<b>(Deansgrade, Northpoint 2, Fonthill and Cahirciveen, Clifden, derrybeg resp.)</b><br/>
2. Which year has the highest and lowest total pass count?<br/>
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<b>2015 and 2014 respectively</b><br/>
But, in this graph we are not considering the total tests performed by the test centres which shows the actual performance of the tests. For this we will plot another graph.
</div>
<div style = "display: inline-block;width: 50%;padding-left: 15px;margin-bottom: 90px;">
#### <b>Test performance for each test centre</b>
The graph gives the overall idea of the test performance based on pass rate and the year.
As per the graph we can say that for year 2013, 2015, 2016, 2017 and 2018, the pass rate is higher that 55%. And the highest and lowest performance found in Kilkenny and Monaghan test centres respectively.
</div>

```{r,echo=FALSE,include=T, fig.width=9,fig.height=3}
#scatter plot for centre total pass per year passfail1 <- passfail passfail1$Centre <-
fct_reorder(passfail1$Centre, -passfail1$Totalpass) passfail1$TotalPass1 <-
passfail1$Totalpass p1 <- ggplotly(ggplot(data = passfail1, aes(x = Centre, y = Totalpass,
color = Year, size =
TotalPass1)) + geom_point(alpha = 0.5) + theme(axis.text.x = element_text(size=6, angle=-
  90, hjust = 0, vjust = 0.5), legend.position =
```

~~~~
"none", axis.ticks.x = element_blank(), panel.background = element_rect(fill = "white", colour =
"lightblue"), panel.grid.major.y = element_line()) + labs(x = "Test Centres", y = "Totol
  pass count"), tooltip = c("Centre","Year", "Totalpass"))
%>% layout(yaxis = list(gridcolor = toRGB("lightblue")), font = t)
```
<img src = "Result//3.jpg" style = "margin-left: 60px;margin-bottom: -18px;">
```{r,echo=FALSE, fig.width=10,fig.height=3}
passfail1$totPassPercentage <- round((passfail1$Totalpass / (passfail1$Totalpass +
passfail1$totalFails)) * 100, digits = 2) passfail1$totFailPercentage <-
round((passfail1$totalFails / (passfail1$Totalpass + passfail1$totalFails)) * 100,
digits = 2) passfail1$totPassPercentage1 <- round((passfail1$Totalpass /
(passfail1$Totalpass + passfail1$totalFails)) * 100, digits = 2) passfail1$Centre <-
fct_reorder(passfail1$Centre, -passfail1$totPassPercentage)
#scatter plot for centre pass percetage per year
p2 <- ggplotly(ggplot(data = passfail1, aes(x = Centre, y = totPassPercentage, color = Year, size
= totPassPercentage1)) + geom_point(alpha = 0.5) + theme(axis.text.x = element_text(size=6,
  angle=-90, hjust = 0, vjust = 0.5), legend.position =
"none", legend.background = element_blank(), axis.ticks.x = element_blank(), panel.background
= element_rect(fill = "white", colour = "lightblue"), panel.grid.minor = element_line(size = 0.5,
linetype = 'solid', colour = "lightblue")) + labs(x = "Test Centres", y = "Total Pass %"), tooltip =
c("Centre","Year", "totPassPercentage")) %>% layout(yaxis = list(gridcolor =
toRGB("lightblue")), font = t) #title = "Test centre pass% per year",
```
<div style = "width: 100%;">
<div style = "float: left;display: flex;">
```{r,echo=FALSE, fig.show="hold", fig.width=5, fig.height=3.5}
p1
```
</div>
<div style = "display: flex;">
```{r,echo=FALSE, fig.show="hold", fig.width=5, fig.height=3.5}
p2
```
</div></div>
<hr style = "margin: 10px 0px 10px;">
<div style = "float: left;">
```{r,echo=FALSE, fig.width=6,fig.height=2.5} p <- plot_ly(passfail, x = passfail$Year, y =
passfail$Totalpass, color = ~passfail$Year, type =
"box", text = paste("Centre = ", passfail$Centre)) %>% layout(title = "Yearly performance",
  yaxis = list(title = 'Total Pass Count'), xaxis = list(title =
'Year'), showlegend = FALSE, font = t, legend = list(x = 0.9, y = 0.98))
p
```
</div>
<div style = "float: right;width: 35%;margin-top: 25px;">
#### <b>Total pass count limits per year</b>
~~~~

The box plot shows the total pass count against each year. With this we can fetch the details on maximum and minimum pass counts per year, the meadian pass count and the oustanding pass count values which are shown as outliers (points) per year with the test centre name.
</div>

2. Road safety

~~~
--

title: "RoadSafety" author: "Amol |
Haojun | Japneet | Calum" date:
"11/11/2019"
output: html_document
---



```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)

~~~

```{r}
#Creating make n model data frame
#reading excel file
m_m <- readxl::read_excel('.\\DATA\\make_n_model\\mmAll.xlsx')
#demo tag
nrow(m_m)

# m_m <- full_join(m_m13, m_m14)
# paste(colnames(m_m13), " = ", colnames(m_m18))

```
```

3. Final_projects Codes

```
---
always_allow_html: yes author: "Amol | Haojun | Japneet | Calum"
                              output:
                                            html_document: default
                                             pdf_document: default
                                             pagetitle: Road Safety
```
```

~~~
---

<style type="text/css">
   body{  /*  Normal  */
   font-size: 12px;
  }
.column-left{ float:
  left;
  width: 40%;
  text-align: left;
}
.column-right{ float:
  right;
  width: 60%;
  text-align: right;
}
/* Clear floats after the columns */
.row:after { content:
  ""; display: table;
  clear: both;
}
.column-left1{ float:
  left;
  width: 80%;
  text-align: left;
}
.column-right1{ float:
  right;
  width: 20%;
  text-align:     right;
  padding-left: 15px;
  padding-top: 15px;
}
.column-left2{ float:
  left;
  width: 47.5%;
  text-align: left;
}
.column-right2{
  float:        right;
  width:    47.5%;
  text-align: right;
}
</style>

```{r setup, include=FALSE} knitr::opts_chunk$set(echo
= TRUE)
~~~

```
        ## Install and library necessary libraries

        packages <- c("plotly", "tidyverse", "ggmap", "GGally", "gridExtra", "scales", "viridis", "scatterplot3d",
"readxl") newPackages <- packages[!(packages %in%
        installed.packages()[,"Package"])] if(length(newPackages))
        install.packages(newPackages) library(tidyverse)
        library(plotly)
        library(gridExtra)
        library(scales)
        library(GGally)
        library(viridis) library(ggmap)
        library(scatterplot3d)
        library(readxl)

        # Load necessary data files

        load("passfail.RData")
        load("nct_geom.RData")

        # Private API key for google maps. Please do not share.

        register_google("AIzaSyDy7z18GxhakN5ACVLsdqQfIm5B9jRmXpA")
        ```

        ## NCT Statistics Report {.tabset}

        ### Pass/Fail overview - Calum

        ```{r,echo=FALSE} #

        Data preparation

        passfailtotals <-
summarise(group_by(passfail,Year),Pass1=sum(Pass1),Fail1=sum(Fail1),Total1=sum(Total1),Pass2=sum(Pass
2
),Fail2=sum(Fail2),Total2=sum(Total2)) passfailtotals <-
mutate(passfailtotals,Pass1prop=Pass1/Total1,Pass2prop=Pass2/Total2)[c(1,2,3,4,8,5,6,7,9
)]
        passfailtotals1 <- passfailtotals[c(1,2,3,4)] names(passfailtotals1)
        <- c("Year","Pass","Fail","Total") passfailtotals1$Test <- "First"
        passfailtotals2 <- passfailtotals[c(1,6,7,8)]
        names(passfailtotals2) <- c("Year","Pass","Fail","Total")
        passfailtotals2$Test <- "Retest" passfailtotals0 <-
        rbind(passfailtotals1,passfailtotals2) passfailtotals1 <-
        passfailtotals0[c(1,2,4,5)] names(passfailtotals1) <-
        c("Year","Count","Total","Test") passfailtotals1$Result <-
        "Pass" passfailtotals2 <- passfailtotals0[c(1,3,4,5)]
        names(passfailtotals2) <- c("Year","Count","Total","Test")
        passfailtotals2$Result <- "Fail"
```

~~~
passfailtotals0 <- rbind(passfailtotals1,passfailtotals2)
passfailtotals0$Result<-factor(passfailtotals0$Result,c("Pass","Fail"))
passfailtotals0$Test<-factor(passfailtotals0$Test,c("First","Retest")) ```
~~~

Let us begin with an overview of the data. The NCT is a test that all cars over 4 years of age must undergo to legally drive on roads in Ireland. We have NCT pass and fail data for almost 12 million cars tested from 2013 to 2018. This data was recorded from all 47 test centres scattered across Ireland. This includes both initial test and retest data. Please note retest data was not available for 2014, hence it was omitted from our report. Here's an overview of how this data is distributed.

~~~
```{r,echo=FALSE, warning=FALSE,fig.width=9, fig.height=2.5}

# Pass/Fail count barplot

p1 <- ggplot(passfailtotals0,aes(x=Year,y=Count, fill=Result))+
  geom_col(position="dodge")+ theme_bw()+
  theme(legend.position = "none",legend.title = element_blank())+
  scale_fill_manual(values = c("lightblue","slategray"))+
  facet_wrap(~Test)+
  scale_y_continuous(labels = comma)

# Pass/Fail rate barplot

p2 <- ggplot(passfailtotals0,aes(x=Year,y=Count, fill=Result))+
  geom_col(position="fill")+
  labs(y="Proportion")+
  geom_hline(yintercept = 0.5,col="red")+
  theme_bw()+
  theme(legend.key = element_rect(colour="black"), legend.position = c(0.912,0.85),legend.title = element_blank(), legend.background = element_rect(fill="transparent"), legend.text = element_text(size = 8))+
  scale_fill_manual(values = c("lightblue","slategray"))+
facet_wrap(~Test) # Arrange plots side by side

grid.arrange(p1, p2, ncol=2 )
```
~~~

As you can see the majority fail the first test, however the margins are quite close. As to be expected, the retest has a low fail rate.It is interesting to note that both total number of cars tested and pass proportion per year hasn't fluctuated much. One might expect that as the population increases, so too must the number of cars. One possible explanation for the lack of growth is that more people may be switching to public transport. We would also expect as technology advances cars should become more reliable, yet our data does not support this theory. Perhaps the NCT have included stricter requirements that would balance this increase.

<div class = "column-left">

<br><br><br>

#### **Which test centre should I go to?**

To the right we've ranked different centres by their first test pass proportions. Using an exponentially weighted mean we prioritized more recent results in our calculation. The top shows centres with relatively high pass rates and the bottom shows the centres with the lowest. Notice how consistent the scores are. This could be dues to higher quality vehicles in more affluent areas or it could indicate a bias in the testing centres. Our recomendations are if you live in Monaghan, take a weekend trip to Kilkenny for your car test, you may end up saving money.

<br><br><br><br><br>

#### **Is location a factor?**

To test the above theory we created the map to the right. The colour represents the same scale as above, with size representing the total volume of cars in 2018. There is a large cluster of low ranking centres in north-central and north-west Ireland. This may support our affluency theory. If we look at the Dublin area there are low ranking centres to the north and higher ranking centres to the south. This could be a reflection of the northside - southside distribution of wealth. It is intriguing that Kerry has some of the highest ranked centres, despite being a more rural county. Traffic volume seems less significant there are large centres and small centres at either end of the spectrum. </div>

~~~~
<div class = "column-right">
```{r,echo=FALSE, message=FALSE, include=F} #

data preparation for parallel coords and map

x    <-    data.frame(split(passfail$Pass1prop,passfail$Year))
names(x) <- c("2013","2014","2015","2016","2017","2018")
x <- cbind(x,nct_geom)
x$Total2018<- passfail$Total1[passfail$Year=="2018"]
z          <-          rev(diff(c(0,pexp(1:6,0.5))))          x          <-
arrange(x,desc(rowSums(mapply(`*`,select(x,starts_with("2")),z))))
x$Centre <-factor(x$Centre,levels=x$Centre) x$Rank <- 1:47
```

```{r,echo=FALSE, message=FALSE, include=T}

# Parallel coords plot

p <- ggparcoord(x, columns=1:6, groupColumn = "Centre")+
  geom_line(size=0.3)+
  theme_minimal()+
  scale_color_viridis(discrete = TRUE, direction = -1, option="C")+
  labs(x="",y="")
ggplotly(p, width = 550, height = 300, tooltip = c("Centre",".ID"))

# Ireland map with data points

Ire_map <- get_googlemap(center=c(-7.8,53.5), zoom=7,style =
~~~~

~~~
'feature:administrative|element:labels|visibility:off')
        p <- ggmap(Ire_map)+
          geom_point(data=x, aes(x=lat,y=lon, colour=Centre,
          size=Total2018))+ scale_radius(range=c(1,3))+ theme_bw()+
          scale_color_viridis(discrete = TRUE, direction = -1,
          option="C")+ theme(legend.position = "none")+ labs(x="", y="")
        ggplotly(p, width = 550, height = 300, tooltip=c("Centre","Total2018"))
        ```
        </div>
~~~

### Analysis based on test centres - Amol

~~~
       ```{r,echo=FALSE}
       load("passfail.RData")
       passfail <- passfail %>%
mutate(totalFails = Fail1 + ifelse(is.na(Fail2), 0, Fail2), Totalpass = Pass1 + ifelse(is.na(Pass2), 0, Pass2))
        ```
~~~

~~~
        ```{r,echo=FALSE}
        passfailGroup <- summarise(group_by(passfail, Centre), Pass1 = sum(Pass1), Fail1 = sum(Fail1),
Total1 = sum(Total1), Pass2 = sum(Pass2, na.rm = T), Fail2 = sum(Fail2, na.rm = T), Total2 =
sum(Total2, na.rm = T), Totalpass = sum(Totalpass), totalFails = sum(totalFails)) passfailGroup <-
mutate(passfailGroup, Pass1prop = Pass1/Total1, Pass2prop = Pass2/Total2, totalPassProp = (Totalpass /
(Total1 + Total2)), totalFailsProp = (totalFails / (Total1 + Total2)))
        ```
~~~

~~~
        ```{r,echo=FALSE}
        passfailGroup$totalPassProp = round((passfailGroup$totalPassProp * 100), digits = 2)
        passfailGroup$totalFailsProp = round((passfailGroup$totalFailsProp * 100), digits = 2)
        passFailGroup1 <- passfailGroup[c(1, 8)] passFailGroup1$Test <- "Pass"
        names(passFailGroup1) <- c("Centre", "Count", "Test") passFailGroup2 <-
        passfailGroup[c(1, 9)] passFailGroup2$Test <- "Fail"
        names(passFailGroup2) <- c("Centre", "Count", "Test") passFailcount
        <- rbind(passFailGroup1, passFailGroup2)
        ```
~~~

In this section we will analyse data from 2013 till 2018 about each test centre. As shown in the <a href = "https://github.com/NanawareAmol/R-project_Road-safety/blob/master/Result/loc_spread_across_ireland.JPG">map</a>, the test centres are spread across the Ireland and the number of centres is more in highly populated areas such as dublin, cork etc.

The bar chart shows the total number of tests that each centre performed and the total pass and fail counts as well as percentages. So, based on the test counts, the top 3 test centre are, *Fonthill(770685)*, *Deansgrade(767484)*, and *Northpoint 2(729661)*. The botton 3 centres which performed less tests are, *Donegal Town(16315)*, *Cahirciveen(28806)* and *Clifden(38683)*.

~~~~
```{r,echo=FALSE, fig.width=9,fig.height=2.8}
t <- list(size = 8)
p <- plot_ly(passfailGroup, x = ~passfailGroup$Centre, y = ~passfailGroup$Totalpass, type = 'bar',
name = 'Pass', text = paste("Total tests = ", (passfailGroup$Totalpass+passfailGroup$totalFails), "<br>Passed
=", passfailGroup$totalPassProp,"%", "<br>Failed =", passfailGroup$totalFailsProp,"%"), opacity = 0.5,
marker
= list(color = '#3AC3E3', line = list(color = '#0D6EB0', width = 1))) %>% add_trace(y =
  ~passfailGroup$totalFails, name = 'Fails', opacity = 0.5, marker = list(color =
  '#0E84FF', line = list(color = '#0D6EB0', width = 1))) %>%
   layout(yaxis = list(title = 'Count'), xaxis = list(title = 'Test Centres'), barmode = 'stack', font = t,legend
= list(x = 0.93, y = 1))
p


```
~~~~

<hr style = "margin: 10px 0px 10px;">

<div style = "display: inline-block;float: left;width: 55%;"> ####
<b>Total test passed for each test centre</b>

The following scatter plot shows the total test pass count for each test centre from the year 2013 till year 2018. The questions that can be answered by this graph are, <br/>
1. which are the top 3 and last 3 centres based on total pass count?<br/>
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<b>(Deansgrade, Northpoint 2, Fonthill and Cahirciveen, Clifden, derrybeg resp.)</b><br/>
2. Which year has the highest and lowest total pass count?<br/>
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<b>2015 and 2013 respectively</b><br/>
But, in this graph we are not considering the total tests performed by the test centres which shows the actual performance of the tests. For this we will plot another graph.

</div>
<div style = "display: inline-block;width: 45%;padding-left: 15px;margin-bottom: 30px;">

#### <b>Test performance for each test centre</b>

The graph gives the overall idea of the test performance based on pass rate and the year.
As per the graph we can say that for year 2013, 2015, 2016, 2017 and 2018, the pass rate is higher that 55%. And the highest and lowest performance found in Kilkenny and Monaghan test centres respectively. The case with the 2014 being less in number is because of the incomplete data available from the NCT website and it can be processed in the same manner if we have the complete set. </div>

~~~~
```{r,echo=FALSE,include=F, fig.width=8.5,fig.height=3}
~~~~

~~~
        #scatter plot for centre total pass per year passfail1 <- passfail
        passfail1$Centre <- fct_reorder(passfail1$Centre, -passfail1$Totalpass)
        passfail1$TotalPass1 <- passfail1$Totalpass
        p1 <- ggplotly(ggplot(data = passfail1, aes(x = Centre, y = Totalpass, color = Year, size = TotalPass1))
+ geom_point(alpha = 0.5) + theme(axis.text.x = element_text(size=6, angle=-90, hjust = 0, vjust = 0.5),
         legend.position = "none",
axis.ticks.x = element_blank(), panel.background = element_rect(fill = "white", colour = "lightblue"),
panel.grid.major.y = element_line()) +
         labs(x = "Test Centres", y = "Totol pass count"), tooltip = c("Centre","Year", "Totalpass")) %>%
layout(yaxis = list(gridcolor = toRGB("lightblue")), font = t)

        ```
        <img src = "3.jpg" style = "margin-left: 60px;margin-bottom: -10px;">
        ```{r,echo=FALSE, fig.width=9,fig.height=3}
        passfail1$totPassPercentage <- round((passfail1$Totalpass / (passfail1$Totalpass +
passfail1$totalFails)) * 100, digits = 2) passfail1$totFailPercentage <- round((passfail1$totalFails /
(passfail1$Totalpass + passfail1$totalFails))
* 100, digits = 2) passfail1$totPassPercentage1 <- round((passfail1$Totalpass /
(passfail1$Totalpass + passfail1$totalFails)) * 100, digits = 2) passfail1$Centre <-
fct_reorder(passfail1$Centre, -passfail1$totPassPercentage)
        #scatter plot for centre pass percetage per year
        p2 <- ggplotly(ggplot(data = passfail1, aes(x = Centre, y = totPassPercentage, color = Year, size =
totPassPercentage1)) + geom_point(alpha = 0.5) + theme(axis.text.x = element_text(size=6, angle=-90, hjust
= 0, vjust = 0.5), legend.position = "none",
axis.ticks.x = element_blank(), panel.background = element_rect(fill = "white", colour = "lightblue"),
         panel.grid.minor = element_line(size = 0.5, linetype = 'solid', colour = "lightblue")) +
         labs(x = "Test Centres", y = "Total Pass %"), tooltip = c("Centre","Year", "totPassPercentage")) %>%
layout(yaxis = list(gridcolor = toRGB("lightblue")), font = t) #title = "Test centre pass% per year",
        ```
        <div style = "width: 100%;">
        <div style = "float: left;display: flex;">
        ```{r,echo=FALSE, fig.show="hold", fig.width=4.75, fig.height=3.3} p1
        ```
        </div>
        <div style = "display: flex;">
        ```{r,echo=FALSE, fig.show="hold", fig.width=4.75, fig.height=3.3} p2
        ```
        </div></div>
        <hr style = "margin: 10px 0px 10px;">
        <div style = "float: left;">
        ```{r,echo=FALSE, fig.width=6,fig.height=2.3}
        p <- plot_ly(passfail, x = passfail$Year, y = passfail$Totalpass, color = ~passfail$Year, type = "box",
text = paste("Centre = ", passfail$Centre)) %>% layout(title = "Yearly performance", yaxis = list(title =
'Total Pass Count'), xaxis = list(title = 'Year'),
font = t, legend = list(x = 0.92, y = 0.98, bgcolor = "transparent"), showlegend = FALSE)
        p
        ```
~~~

~~~
</div>
<div style = "float: right;width: 35%;margin-top: 25px;">
#### <b>Total pass count limits per year</b>
The box plot shows the total pass count against each year. With this we can fetch the details on
maximum and minimum pass counts per year, the median pass count and the outstanding pass count
values which are shown as outliers (points) per year with the test centre name. </div>

### Equipment Failure - Japneet
```{r, echo=FALSE}
####LOADING AND CLEANING THE DATASET####
df <- read_excel("mmAll.xlsx")

names(df)[9] <- "Vehicle and Safety Equipment"
names(df)[10] <- "Vehicle and Safety Equipment %"
names(df)[22] <- "Chassis and Body %"
names(df)[26] <- "Suspension Test %" names(df)[36]
<- "Incomplete Tests %"

df$reportYear <- as.factor(df$reportYear)
```



<div class = "column-left">

<br>
<br>

#### **Equipment Failure - An Overview**

<br>
<br>

<p style="text-align:justify">
The barplot, resulting from the exploratory data analysis, arranges the different vehicle item categories
in decreasing order of their failure percentage over a span of 6 years altogether. Overall, Lighting and Electrical
is the most failed item category with a failure percentage of 19.87 whereas Body and Chassis being the least
failed known category with a failure percentage of 4.67. The category Other being the least failed item category,
overall, includes the parts that are not covered in the major 12 categories and hence is the area of least interest
for this analysis.
</p>
<br>
<br>
<br>
<br>
<br>
<br>
~~~

#### **Analyzing Part Failures Per Report Year**

<br>
<br>

<p style="text-align:justify">
Diving further, we derive interesting insights on analyzing the item failure for each report year. Among the top 3 failure items overall, the Lighting and Electrical holds the topmost position throughout the entire span with a fail percentage hovering just around 20. However, the failure percentage for Steering and Suspension follows an increasing trend from 2014 to 2018 with an increase of 2.089%, which moves it up the list from third position in 2014 to a second position in 2015. A corresponding decrease in failure percentage of wheels and Tyres is observed which moves it down the list to become the third most failed item in 2018. </p>
<br>
<br>
<br>
<br>
<br>
<br>
<br>

#### **Is there Any Relationship between Top Vehicle Makes and Top 3 item failure categories?**

<br>
<br>

<p style="text-align:justify">
Certainly Yes. TOYOTA seem to have the lowest failure percentage among all the vehicle makes for all the three item categories. Collectively, all the top 5 makes have improved their 'Wheels and Tyres' over the 6 report years. However, an increase in failure percentage for 'Light and Electrical' and 'Steering and Suspension' is observed for almost all the makes with NISSAN and VOLKSWAGEN being an exception with a slight decrease of 0.348% for NISSAN and that of 2.154% for VOLKSWAGEN in failure percentage of 'Light and Electrical' parts.
</p>
</div>

<div class = "column-right">

<br>
<br>
<br>

~~~
        <p style="text-align:center">
        ```{r, echo=FALSE, fig.height=5, fig.width=12, warning=FALSE}

        #########PLOT 1##########

cols <- c("Vehicle and Safety Equipment", "Lighting and Electrical", "Steering and Suspension",
 "Braking Equipment", "Wheels and Tyres", "Engine, Noise and Exhaust", "Chassis and Body", "Side Slip
 Test", "Suspension Test", "Light test", "Brake Test", "Emmissions", "OTHER")
        a <- df %>% dplyr::select(cols)
        b<-colSums(a)
        c <- data.frame(Part = names(b), Percent = unname(b)/sum(df$Total)*100)



p1  <-  ggplot(c)+  geom_col(mapping  =  aes(x  =  reorder(Part,  -Percent),  y  =  Percent,  fill  =  Percent),
 col="black")+ xlab("")+ theme_light()+ ylab("Failure Percentage(%)") +
         scale_fill_gradient(low = "lightblue", high = "brown")+
         coord_flip()+ scale_y_continuous(labels = function(x) paste0(x, "%"))+
         theme(legend.position = "none", panel.grid.major.x = element_blank(), panel.border
 = element_blank(), panel.grid.major.y = element_blank(), panel.grid.minor = element_blank(),
 axis.text=element_text(size=16), axis.title = element_text(size = 20))+ geom_text(aes( x =
 Part, y = Percent+0.9, label = round(Percent, 2)), size = 5)




        p1
        ~~~

        <br>
        <br>

        ```{r, echo=FALSE, warning=FALSE}

        #########PLOT 2##########

cols <- c("Total", "Vehicle and Safety Equipment", "Lighting and Electrical", "Steering and
 Suspension", "Braking Equipment", "Wheels and Tyres", "Engine, Noise and Exhaust", "Chassis and Body",
 "Side Slip Test", "Suspension Test", "Light test", "Brake Test", "Emmissions", "OTHER")
s <- df %>% dplyr :: select(c("reportYear", cols)) %>% group_by(reportYear) %>%
 summarise_if(is.numeric, sum, na.rm = TRUE) %>% mutate_at(vars(c(-1,-2)), funs(round((. / Total)*100,
 digits = 3)))

        m <- gather(s,-reportYear, key=Part, value= Failures) m
        <- m[7:84, ]

        p2 <- ggplotly(ggplot(data=m, mapping = aes(x = reportYear, y = Failures, colour = Part, group=1))+
         geom_point()+ theme_minimal()+
```

~~~
                geom_line()+xlab("Report Year")+ ylab("Failure Percentage") + scale_y_continuous(labels =
 function(x) paste0(x, "%")), height=400)

            p2
            ```

            ```{r, echo=FALSE, warning=FALSE, message=FALSE,fig.height=4, fig.width=5.5}
            #########PLOT 3##########



            e <- c("VehicleMake", "reportYear", "Total", "Lighting and Electrical", "Steering and Suspension",
 "Wheels and Tyres") s <- df %>%
            dplyr::select(e) %>%
            filter(VehicleMake %in% c("TOYOTA", "VOLKSWAGEN", "FORD", "NISSAN", "OPEL")) %>%
              group_by(VehicleMake, reportYear) %>% summarise_if(is.numeric, sum, na.rm = TRUE) %>%
              mutate_at(vars(c(-1,-2, -3)), funs(round((. / Total)*100, digits = 3)))

plot_ly(x=s$`Lighting and Electrical`, y=s$`Steering and Suspension`, z=s$`Wheels and Tyres`,
 type="scatter3d", mode="lines", color= as.factor(s$VehicleMake), marker = list(symbol = 'circle', sizemode =
 'diameter'), sizes = c(5, 150), text= s$reportYear, hovertemplate = paste('<i>Report Year</i>: %{text}',
                                 '<br><b>Lighting and Electrical</b>: %{x}%',
                                 '<br><b>Steering and Suspension</b>: %{y}%',
                                 '<br><b>Wheels and Tyres</b>: %{z}%')) %>%
                layout(scene = list(xaxis = list(title = 'Lighting and Electrical (%)'),
                                yaxis = list(title = 'Steering and Suspension (%)'),
                                zaxis = list(title = 'Wheels and Tyres (%)')))

            ```
            </p>
            </div>

            ### Make/Model analysis - Haojun



            ```{r echo=FALSE }
            ################## raw data ################## mmdata<-
            read_excel("mmAll1.xlsx",sheet=2,na="NA")
            ```


            ```{r ,echo=FALSE}
            ################## select top 15 market share make's name ##################
            # totol number of car in each make
            TotalMumMake        <-        mmdata        %>%
              group_by(VehicleMake)    %>%    summarise(
              MakeTotal=sum(Total, na.rm=T))
            # top 15 make names
            TotalMumMake<-arrange(TotalMumMake,desc(MakeTotal))
~~~

~~~
        Name15<-TotalMumMake$VehicleMake[1:15]
        ```


        ```{r , echo=FALSE}
        ################## prepare data for market share plot ##################
        # the number of car of top 15 make in each year
        TotaMumlMakeYear <- mmdata %>%
          group_by(VehicleMake,reportYear) %>%
          summarise( MakeTotal=sum(Total, na.rm=T))
        # the number of car in each year
        TotalMumYear<- mmdata %>%
        group_by(reportYear) %>%
          summarise( YearTotal=sum(Total, na.rm=T))
        # left join TotaMumlMakeYear and TotalMumYear
        MarketShare<-left_join(TotaMumlMakeYear,TotalMumYear,by="reportYear")
        # calculate market share of each brand in each year
        MarketShare$marketshare<-MarketShare$MakeTotal/MarketShare$YearTotal
        # select market share of top 15 make
        Top15<-filter(MarketShare,VehicleMake %in% c( "TOYOTA",'VOLKSWAGEN'
,'FORD','NISSAN','OPEL',
        'RENAULT','PEUGEOT','BMW','AUDI','MERCEDES BENZ',
        'HYUNDAI','SKODA','HONDA','MAZDA','CITROEN'))
        Top15<-arrange(Top15,desc(marketshare))
        #sort make for plot
        Top15$VehicleMake <- factor(Top15$VehicleMake, levels=c('TOYOTA', 'VOLKSWAGEN', 'FORD',
'NISSAN', 'OPEL', 'RENAULT','PEUGEOT','BMW','AUDI','MERCEDES BENZ','HYUNDAI','SKODA',
'HONDA','MAZDA','CITROEN'), ordered=TRUE)
        ```

        <div class = "column-left1" style = "width: 65%;">
        #### **Market Share of Car Makes**
        ```{r Top15,echo=FALSE,fig.height=5,fig.width=10}
        ################## plot of market share for top 15 make ##################
ggplot()+
          geom_point(data=Top15,mapping = aes(x=marketshare,y=
          VehicleMake,color=reportYear),size=2)+ geom_line(data=Top15,mapping = aes(x=marketshare
          ,y=VehicleMake,color=reportYear),size=2)+ labs(x="Market Share",y="Vehicle Make",
          color="Year")+ theme_bw()+
          theme(axis.text.x = element_text(face="bold", color="black", size=14,angle = 45,vjust =
              0.6), axis.text.y = element_text(face="bold", color="black", size=14), plot.title =
              element_text(hjust = 0.5), axis.title = element_text(size = 17), legend.position =
              c(0.9,0.6), legend.text = element_text(size=14), legend.title = element_blank()) +
          coord_flip()+ scale_x_continuous(labels =
          scales::percent,breaks=seq(0,1,0.05))
        ```

        ```{r ,echo=FALSE}
~~~

~~~~
####### data for plot of fail rate and distribution of each car age cut #######
# car age
mmdata_1<-cbind(mmdata,CarAge=mmdata$reportYear-mmdata$YearOfBirth)
# cut car age
mmdata_1$CarAge_cut<-cut(mmdata_1$CarAge,c(-999,4,6,8,10,12,14,16,18,999),labels = c(
  '[0,4]','(4,6]','(6,8]','(8,10]','(10,12]','(12,14]','(14,16]','(16,18]','(18,+)'))
# summarize number group by CarAge_cut
TotalMumAge <- mmdata_1 %>%
group_by(CarAge_cut) %>%
  summarise( MakeTotal=sum(Total, na.rm=T),
            FailTotal=sum(FAIL, na.rm=T) )
# calculate fail rate for each age cut
TotalMumAge$FailRate<-TotalMumAge$FailTotal/TotalMumAge$MakeTotal
# distribution of car age
TotalMumAge$CarAgePer<-TotalMumAge$MakeTotal/sum(TotalMumAge$MakeTotal)
```

<br>

#### **Fail Rate and Proportion of Car Ages**

```{r ,echo=FALSE,fig.height=4,fig.width=10}
################ plot of fail rate and distribution of each car age group ################ cols
<- c("Fail Rate" = "red", "Proportion" = "skyblue")
ggplot(data = TotalMumAge) +
  geom_point(mapping = aes(x = CarAge_cut, y = FailRate,colour="Fail Rate"),size=3)+
  labs(x="Car                                        Age",y="")+
  geom_bar(aes(CarAge_cut,weight=CarAgePer,fill="Proportion"),colour="black",width = 0.5)
  + theme_bw()+
  theme(plot.title = element_text(hjust = 0.5), axis.text.x =
      element_text( face="bold", color="black", size=14), axis.text.y =
      element_text( face="bold", color="black",size=14),
      legend.position = c(0.9,0.6), legend.text = element_text(size=14),
      axis.title = element_text(size = 17))+
  scale_colour_manual(name = "",values=cols)+
  scale_fill_manual(name="",values=cols)+
  scale_y_continuous(labels = scales::percent)

```
<br>
<br>
</div>



<div class = "column-right1" style = "text-align: left;width: 35%;">
<p>
<br>
~~~~

<br>
<br>
<br>
The plot shows the top 15 market share of car makes,which occupy more than 85% of the total market in Ireland. Toyota, Volkswagen and Ford rank top 3 market share in recent six years and have kept stable.The market share of Nissan, Opel and Renault have declined significantly from 2013 to 2018.But for Audi,Hyundai and Skoda, it has increased gradually.

<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>

The bar chart is the distribution of car ages, which shows the largest proportion of cars are between 8 and 14 years old. The point above the bar represents the fail rate of cars of this age in the first test. As the car age increase, fail rate rises linearly.
</p>
<br>
<br>
<br>
<br>
<br>
<br>
</div>

~~~
```{r ,echo=FALSE}
################ plot of fail rate for top 15 makes in different age cut #################
# summarize number for different makes in different age group
TotalMumAgeMake     <-     mmdata_1     %>%
  group_by(VehicleMake,CarAge_cut)      %>%
  summarise( MakeTotal=sum(Total, na.rm=T),
             FailTotal=sum(FAIL, na.rm=T))
# calculate fail rate
TotalMumAgeMake$FailRate<-TotalMumAgeMake$FailTotal/TotalMumAgeMake$MakeTotal
# selsect top 15 makes for plot
TotalMumAgeMake<-filter(TotalMumAgeMake,VehicleMake %in% c( "TOYOTA",'VOLKSWAGEN'
~~~

```
,'FORD','NISSAN','OPEL','RENAULT','PEUGEOT','BMW','AUDI','MERCEDES BENZ',
                                                        'HYUNDAI','SKODA','HONDA','MAZDA' ,'CITROEN'
          ))
          # calculate 1st Qu., median and 3rd Qu. of fail rate
          MakeFailRate<-
summary(subset(TotalMumAgeMake,TotalMumAgeMake$CarAge_cut=="(10,12]")$FailRate)
          # sort makes for plot lables
          TotalMumAgeMake$VehicleMake <- factor(TotalMumAgeMake$VehicleMake,
levels=c('HONDA','TOYOTA','MAZDA','NISSAN','MERCEDES BENZ','FORD',
'VOLKSWAGEN','OPEL','SKODA','BMW','AUDI','CITROEN','PEUGEOT','HYUNDAI','RENAULT' ))
          ```

          ```{r,echo=FALSE,fig.height=3,fig.width=5 }
          ################ plot of fail ratefor top 15 makes in different age cut ##################
          p1<-ggplot(data = TotalMumAgeMake,mapping = aes(y = FailRate, x=VehicleMake)) +
            geom_boxplot(fill="lightgoldenrod")+ geom_hline(yintercept
            =c(0.5536,0.6139),colour="red",linetype="dotted")+ geom_hline(yintercept
            =c(0.5962),colour="red")+ geom_text(aes(x=-0.2,y=0.5536,label ="LQ",hjust=-
            0.2, vjust = 0.9), size = 2)+ geom_text(aes(x=-0.2,y=0.5962,label
            ="MED",hjust=-0.1, vjust = 0.9), size = 2)+ geom_text(aes(x=-
            0.2,y=0.6139,label ="UQ",hjust=-0.1, vjust = -0.2), size = 2)+ labs(x="Vehicle
            Make", y="Fail Rate")+ ggtitle("Fail Rate of Car Makes")+
            theme_bw()+
            theme(axis.text.x = element_text(angle = 45, face="bold", color="black", size=8,vjust =
                 0.6), axis.text.y = element_text(face="bold", color="black", size=8), plot.title =
                 element_text(hjust = 0.5,vjust = 0))+
            scale_y_continuous(labels = scales::percent)


          ```

          ```{r,echo=FALSE }
          ################ select quality of top 5 and bottom 5 model #################
          ## choose Top 15 brand
          mmdata_2<-filter(mmdata_1,VehicleMake %in% c( "TOYOTA",'VOLKSWAGEN'
,'FORD','NISSAN','OPEL', 'RENAULT','PEUGEOT','BMW','AUDI','MERCEDES
BENZ','HYUNDAI','SKODA','HONDA','MAZDA' ,'CITROEN'
          ))
          # compare the fail rate of different model in in same age cut((10,12])
          goodMakeModel <- subset(mmdata_2,CarAge_cut=="(10,12]") %>%
          group_by(VehicleMake,VehicleModel) %>%
            summarise( MakeModelTotal=sum(Total, na.rm=T),
                    FailTotal=sum(FAIL, na.rm=T)) #calculate
                   distribution       of         model
          goodMakeModel$MakeModelPercent<-
goodMakeModel$MakeModelTotal/sum(goodMakeModel$MakeModelTotal)
```

```
        # calculate fial rate of model
        goodMakeModel$MakeModelFialRate<-
goodMakeModel$FailTotal/goodMakeModel$MakeModelTotal
        # sort percentage of model
        goodMakeModel<-arrange(goodMakeModel,desc(MakeModelPercent))
        # choose popular model as the range of analysis (the number of these models shoold occupy at lease
80% of total)
        # choose top 50 model as range of analysis (the number of these models is at least 85% of total)
        #sum(goodMakeModel[1:50,]$MakeModelPercent)
        Top5Model<-arrange(goodMakeModel[1:50,],goodMakeModel[1:50,]$MakeModelFialRate)
        # choose quality top 5 and bottom 5 model for plot
        BestWorst5Model<-Top5Model[c(1:5,46:50),]
        BestWorst5Model$MakeModel<-
paste(BestWorst5Model$VehicleMake,BestWorst5Model$VehicleModel,sep = "-")
        # sort model for plot
        BestWorst5Model$MakeModel<- factor(BestWorst5Model$MakeModel, levels=c("TOYOTA-
YARIS",
        "TOYOTA-RAV 4","TOYOTA-COROLLA","HONDA-CIVIC","FORD-FIESTA", "RENAULT-
SCENIC",             "RENAULT-MEGANE","FORD-GALAXY","RENAULT-LAGUNA","HYUNDAI-
TRAJET"
        ))

        BestWorst5Model$rank<-ifelse(BestWorst5Model$MakeModelFialRate>0.6,"Bottom 5 Model","Top 5
Model")

        ```

        #### **Car Makes and Models Recommended**

        ```{r,echo=FALSE, ,fig.height=3,fig.width=9} p2<-ggplot()
        +
          geom_bar(BestWorst5Model,mapping=aes(x=MakeModel,weight=MakeModelFialRate,fill=rank),
                 width=0.5, colour="black")+
          scale_fill_manual(values = c("firebrick1","seagreen1"))+
          labs(x="Car  Model",y="Fail  Rate")+  ggtitle("Top  and
          Bottom 5 Model")+
          theme_bw()+
          theme(axis.text.x = element_text(face="bold", color="black", size=8,angle = 45, vjust = 0.6),
               axis.text.y = element_text(face="bold", color="black", size=8),
                plot.title = element_text(hjust = 0.5,vjust=0),
               legend.position = c(0.18,0.8),
               legend.title = element_blank(),
               legend.background = element_rect(fill="transparent"))+
        scale_y_continuous(labels = scales::percent) grid.arrange(p1,
        p2, ncol=2 )
        ```
```
```

<div class = "column-left2">

The box plot shows fail rates of different car ages of vehicle makes. The red lines are upper quartile, median and lower quartile fail rate in different car age groups of vehicle makes. Honda, Toyota and Mazda have a better quality, but Renault, Hyundai, Peugeot and Citroen are easy to fail in test. </div>

<div class = "column-right2" style = "text-align: left;">

The bar chart shows top 5 and bottom 5 models in fail rates. These models are selected from most popular 50 models of car between 10 and 12 years old, which occupy more than 85% market share in Ireland. TOYOTA-YARIS,TOYOTA-RAV 4,TOYOTA-COROLLA,HONDA-CIVIC and FORD- FIESTA are the most recommended models. Potential buyers of RENAULT-SCENIC,RENAULT- MEGANE,FORD-GALAXY,RENAULT-LAGUNA and HYUNDAI-TRAJET should be aware they have the least reliable rates. </div>

4. Label Name

The label are attached to github.

https://github.com/IBM-EPBL/IBM-Project-47820-1660802629/blob/main/Project%20Development%20Phase/Sprint%204/labels.csv

5. Regarding Thanks

# Thanks

- Check out [itsleeds.github.io/rrsrr/](https://itsleeds.github.io/ IBM-~~EPBL/IBM-Project-47820-1660802629~~ /)
- Get a GitHub account - Start asking questions
- Twitter `@robinlovelace`