Data Visualisation with R Programming

Woodwinn Teerapat

Necessary Libraries unpackaging

```
library(tidyr)
library(tibble)
library(dplyr)
library(lubridate)
library(ggplot2)
library(ggeasy)
library(RColorBrewer)
library(glue)
library(patchwork)
```

Data preparation of African Diamonds

```
data("diamonds")
glimpse(diamonds)
```

```
## Rows: 53,940
## Columns: 10
## $ carat   <dbl> 0.23, 0.21, 0.23, 0.29, 0.31, 0.24, 0.24, 0.26, 0.22, 0.23, 0.~
## $ cut     <ord> Ideal, Premium, Good, Premium, Good, Very Good, Very Good, Ver~
## $ color   <ord> E, E, E, I, J, J, I, H, E, H, J, J, F, J, E, E, I, J, J, J, I,~
## $ clarity <ord> SI2, SI1, VS1, VS2, SI2, VVS2, VVS1, SI1, VS2, VS1, SI1, VS1, ~
## $ depth   <dbl> 61.5, 59.8, 56.9, 62.4, 63.3, 62.8, 62.3, 61.9, 65.1, 59.4, 64~
## $ table   <dbl> 55, 61, 65, 58, 58, 57, 57, 55, 61, 61, 55, 56, 61, 54, 62, 58~
## $ price   <int> 326, 326, 327, 334, 335, 336, 336, 337, 337, 338, 339, 340, 34~
## $ x       <dbl> 3.95, 3.89, 4.05, 4.20, 4.34, 3.94, 3.95, 4.07, 3.87, 4.00, 4.~
## $ y       <dbl> 3.98, 3.84, 4.07, 4.23, 4.35, 3.96, 3.98, 4.11, 3.78, 4.05, 4.~
## $ z       <dbl> 2.43, 2.31, 2.31, 2.63, 2.75, 2.48, 2.47, 2.53, 2.49, 2.39, 2.~
```

```
sum(is.na(diamonds))
```

N/A Checking

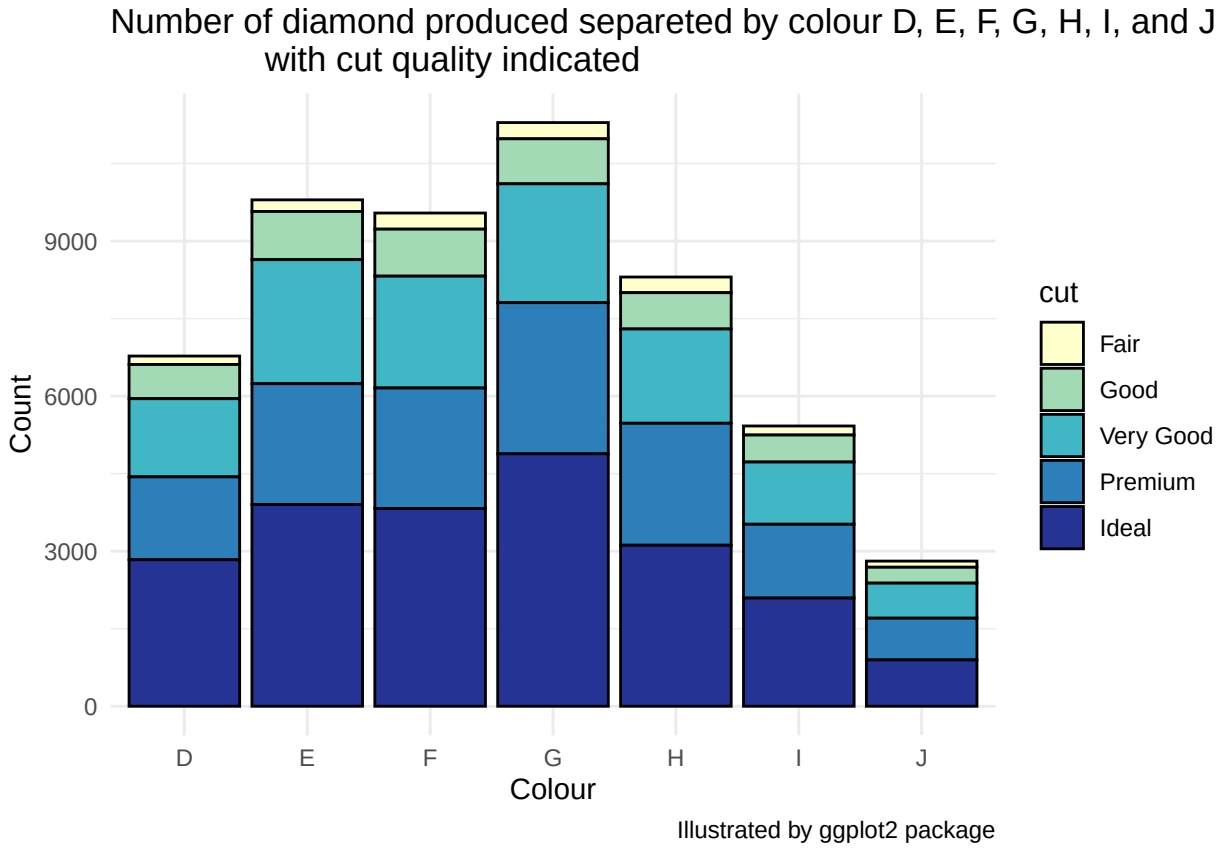
```
## [1] 0
```

None of N/A in this DataFrame.

Data Visualisation

Chart 1: Determine the most manufactured diamond colour

```
ggplot(diamonds, aes(color,
         fill = cut)) +
  geom_bar(color="black") +
  labs(title = "Number of diamond produced separeted by colour D, E, F, G, H, I, and J
                with cut quality indicated",
       x = "Colour",
       y = "Count",
       caption = "Illustrated by ggplot2 package") +
  theme_minimal() +
  scale_fill_brewer(palette="YlGnBu")
```

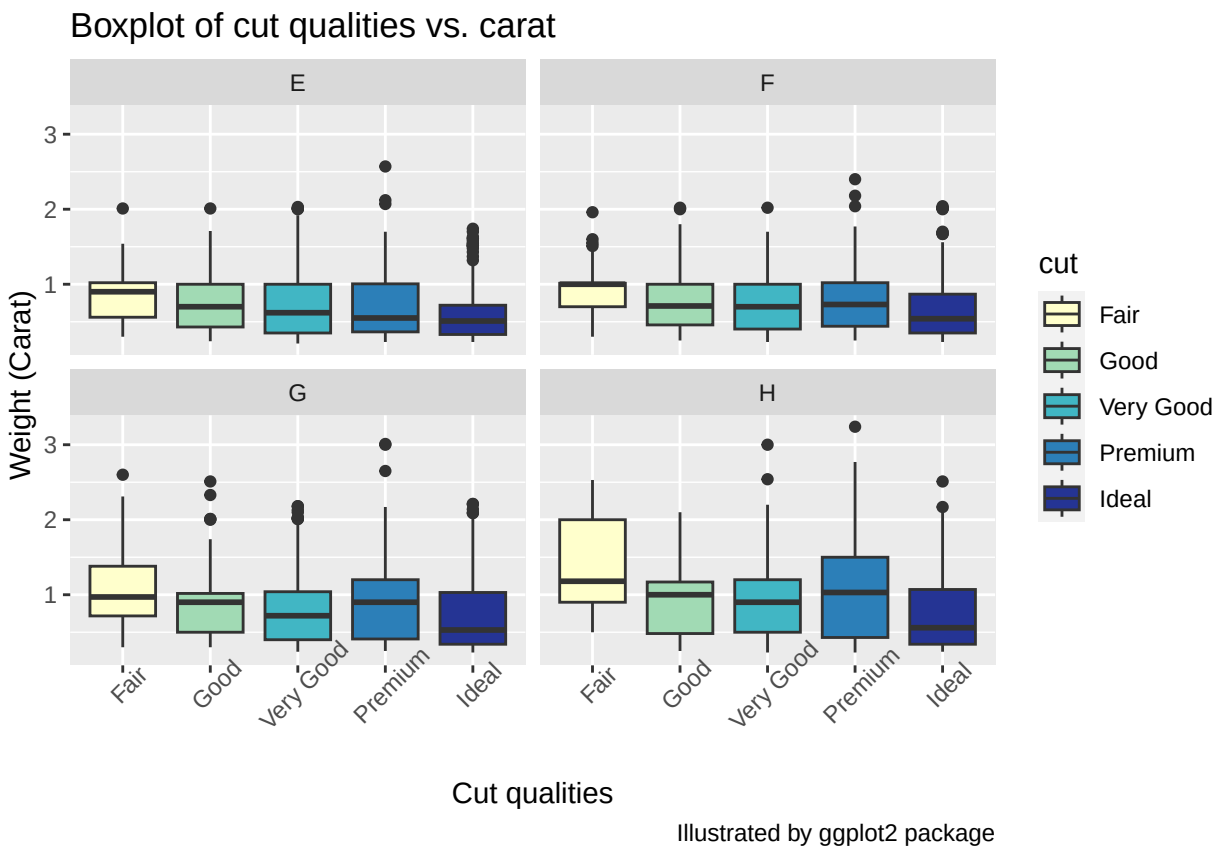


It is noticeably that colour G had the highest amount of production, followed by colour E, F, H, and so on.

Chart 2: The relationship between diamond cut qualities and weight (carat) among top 4 colours using boxplot from sample of 5,000 units.

```
top_color_d <- diamonds %>%
  filter(color == c("G", "E", "F", "H")) %>%
   sample_n(5000)

ggplot(top_color_d, aes(cut, carat,
```

```
      fill = cut)) +
  geom_boxplot() +
  facet_wrap(~ color, ncol = 2) +
  labs(title = "Boxplot of cut qualities vs. carat",
       caption = "Illustrated by ggplot2 package",
       x = "Cut qualities",
       y = "Weight (Carat)")+
  theme(axis.text.x = element_text(angle = 45)) +
  scale_fill_brewer(palette="YlGnBu")
```



The ideal cut quality in colour H have higher weight rather than other colours.

Chart 3: Relationship between Carat and Price (USD) by using scatter plotting together with smooth plotting to reveal the pattern

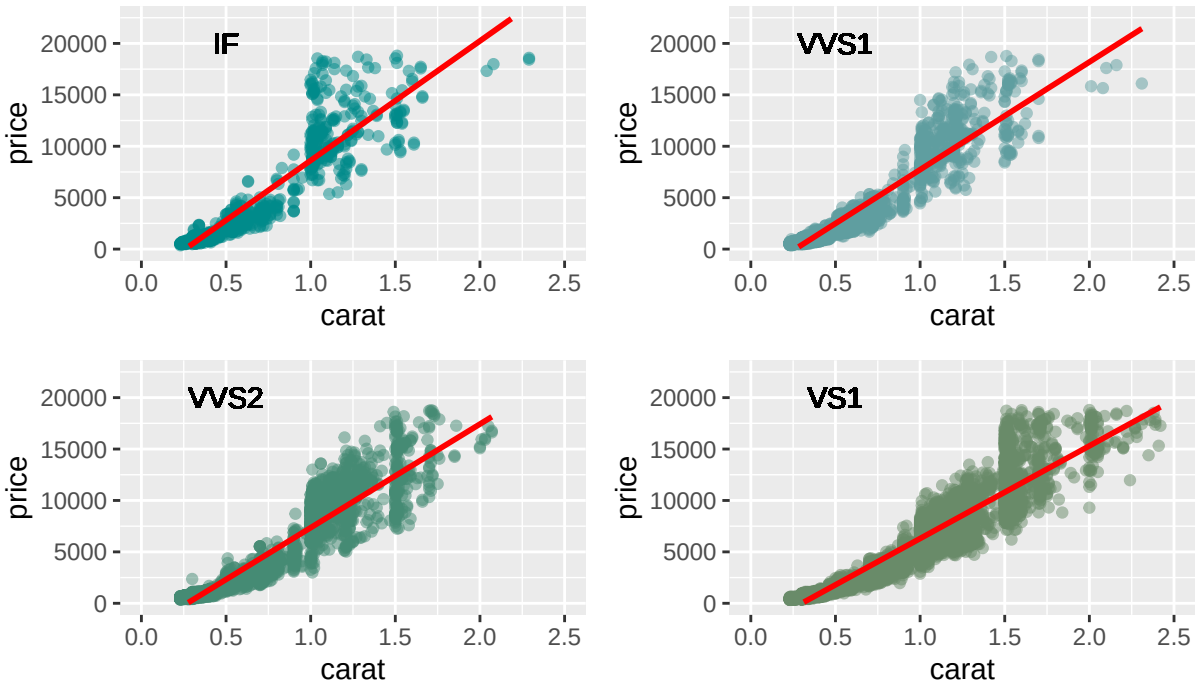
```
claritise <- c("IF", "VVS1", "VVS2", "VS1")
clarity_color <- c("darkcyan", "cadetblue", "aquamarine4", "darkseagreen4")

for (i in 1:4) {
  clarity_d <- diamonds %>% filter(clarity == claritise[i])
  temp_clarity <- claritise[i]

  #plot carat vs price from IF >> VS1
  clarity_plot <- ggplot(clarity_d, aes(carat, price)) +
    geom_point(color = clarity_color[i], alpha = 0.5) +
```

```
    xlim(0, 2.5) + ylim(0, 22500) +
    geom_smooth(method = "lm", color = "red", fill = "tomato") +
    geom_text(x = 0.5,
              y = 20000,
              label = glue("{temp_clarity}"))
    theme_minimal() +
    labs(x = "Weight (Carat)",
         y = "Price (USD)")
  assign(paste("plot",i, sep="_"), clarity_plot)
  i = i+1
}

plot_all <- (plot_1 + plot_2) / (plot_3 + plot_4)

plot_all + plot_annotation(title = "Scatter plot of weight (carat) vs. price (USD) for each clarity",
                           subtitle = "where IF is the clearest, followed by VVS1, VVS2, and VS1",
                           caption = "Illustrated by ggplot2 package")
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 6 rows containing missing values (`geom_smooth()`).
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 2 rows containing missing values (`geom_smooth()`).
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 2 rows containing missing values (`geom_smooth()`).
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 2 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 2 rows containing missing values (`geom_point()`).
## Warning: Removed 3 rows containing missing values (`geom_smooth()`).
```

Scatter plot of weight (carat) vs. price (USD) for each clarity

where IF is the clearest, followed by VVS1, VVS2, and VS1



Illustrated by ggplot2 package

At the high clarity, price tends to be higher compared to the same weight of other level of clarity. Note: There are some outlines data removed due to limitation of axis.

Data preparation of Motor Trend Car Road Tests

```
data("mtcars")
glimpse(mtcars)
```

```
## Rows: 32
## Columns: 11
## $ mpg  <dbl> 21.0, 21.0, 22.8, 21.4, 18.7, 18.1, 14.3, 24.4, 22.8, 19.2, 17.8,~
## $ cyl  <dbl> 6, 6, 4, 6, 8, 6, 8, 4, 4, 6, 6, 8, 8, 8, 8, 8, 8, 4, 4, 4, 4, 8,~
## $ disp <dbl> 160.0, 160.0, 108.0, 258.0, 360.0, 225.0, 360.0, 146.7, 140.8, 16~
## $ hp   <dbl> 110, 110, 93, 110, 175, 105, 245, 62, 95, 123, 123, 180, 180, 180~
## $ drat <dbl> 3.90, 3.90, 3.85, 3.08, 3.15, 2.76, 3.21, 3.69, 3.92, 3.92, 3.92,~
## $ wt   <dbl> 2.620, 2.875, 2.320, 3.215, 3.440, 3.460, 3.570, 3.190, 3.150, 3.~
## $ qsec <dbl> 16.46, 17.02, 18.61, 19.44, 17.02, 20.22, 15.84, 20.00, 22.90, 18~
## $ vs   <dbl> 0, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0,~
## $ am   <dbl> 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,~
## $ gear <dbl> 4, 4, 4, 3, 3, 3, 3, 4, 4, 4, 4, 3, 3, 3, 3, 3, 3, 4, 4, 4, 3, 3,~
## $ carb <dbl> 4, 4, 1, 1, 2, 1, 4, 2, 2, 4, 4, 3, 3, 3, 4, 4, 4, 1, 2, 1, 1, 2,~
```

By doing research into the country origin of each of the cars in the dataset, a vector of "origin" can be mutated into mtcars DataFrame by left_join function.

```
#Converting index into chr column named 'car'
remove(mtcars)
mtcars <- tibble::as_tibble(mtcars, rownames = 'car')
mtcars <- tibble::rowid_to_column(mtcars, "index")
head(mtcars, 5)
```

```
## # A tibble: 5 x 13
##   index car      mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
##   <int> <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     1 Mazda~  21       6   160   110  3.9   2.62  16.5     0     1     4     4
## 2     2 Mazda~  21       6   160   110  3.9   2.88  17.0     0     1     4     4
## 3     3 Datsu~  22.8     4   108    93  3.85  2.32  18.6     1     1     4     1
## 4     4 Horne~  21.4     6   258   110  3.08  3.22  19.4     1     0     3     1
## 5     5 Horne~  18.7     8   360   175  3.15  3.44  17.0     0     0     3     2
```

```
#Create new DataFrame of cars' origin
car <- mtcars$car
origin <- c("Japan", "Japan", "Japan", "United States", "United States", "United States",
            "United States", "Germany", "Germany", "Germany", "Germany", "Germany",
            "Germany", "Germany", "United States", "United States", "United States",
            "Italy", "Japan", "Japan", "Japan", "United States", "United States",
            "United States", "United States", "Italy", "Germany", "British", "United States",
            "Italy", "Italy", "Sweden")

cars_origin_df <- data.frame(car, origin)
glimpse(cars_origin_df)  #confirm the dataframe
```

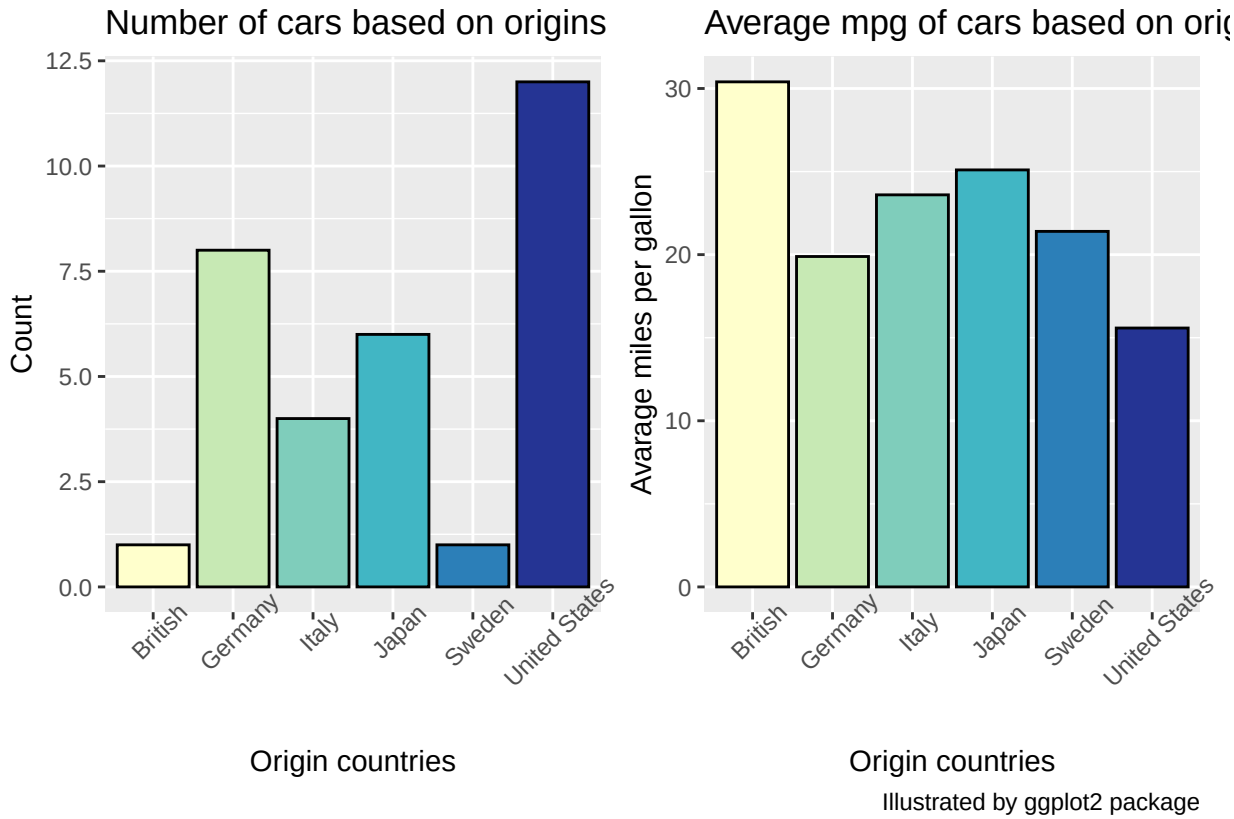
```
## Rows: 32
## Columns: 2
## $ car    <chr> "Mazda RX4", "Mazda RX4 Wag", "Datsun 710", "Hornet 4 Drive", "~
## $ origin <chr> "Japan", "Japan", "Japan", "United States", "United States", "U~
```

```
new_mtcars <- left_join(mtcars, cars_origin_df, by = 'car')
glimpse(new_mtcars)
```

```
## Rows: 32
## Columns: 14
## $ index  <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, ~
## $ car    <chr> "Mazda RX4", "Mazda RX4 Wag", "Datsun 710", "Hornet 4 Drive", "~
## $ mpg    <dbl> 21.0, 21.0, 22.8, 21.4, 18.7, 18.1, 14.3, 24.4, 22.8, 19.2, 17.~
## $ cyl    <dbl> 6, 6, 4, 6, 8, 6, 8, 4, 4, 6, 6, 8, 8, 8, 8, 8, 8, 4, 4, 4, 4, ~
## $ disp   <dbl> 160.0, 160.0, 108.0, 258.0, 360.0, 225.0, 360.0, 146.7, 140.8, ~
## $ hp     <dbl> 110, 110, 93, 110, 175, 105, 245, 62, 95, 123, 123, 180, 180, 1~
## $ drat   <dbl> 3.90, 3.90, 3.85, 3.08, 3.15, 2.76, 3.21, 3.69, 3.92, 3.92, 3.9~
## $ wt     <dbl> 2.620, 2.875, 2.320, 3.215, 3.440, 3.460, 3.570, 3.190, 3.150, ~
## $ qsec   <dbl> 16.46, 17.02, 18.61, 19.44, 17.02, 20.22, 15.84, 20.00, 22.90, ~
## $ vs     <dbl> 0, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, ~
## $ am     <dbl> 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, ~
## $ gear   <dbl> 4, 4, 4, 3, 3, 3, 3, 4, 4, 4, 4, 3, 3, 3, 3, 3, 3, 4, 4, 4, 3, ~
## $ carb   <dbl> 4, 4, 1, 1, 2, 1, 4, 2, 2, 4, 4, 3, 3, 3, 4, 4, 4, 1, 2, 1, 1, ~
## $ origin <chr> "Japan", "Japan", "Japan", "United States", "United States", "U~
```

Chart 4: Composition of origins with descendent sorting

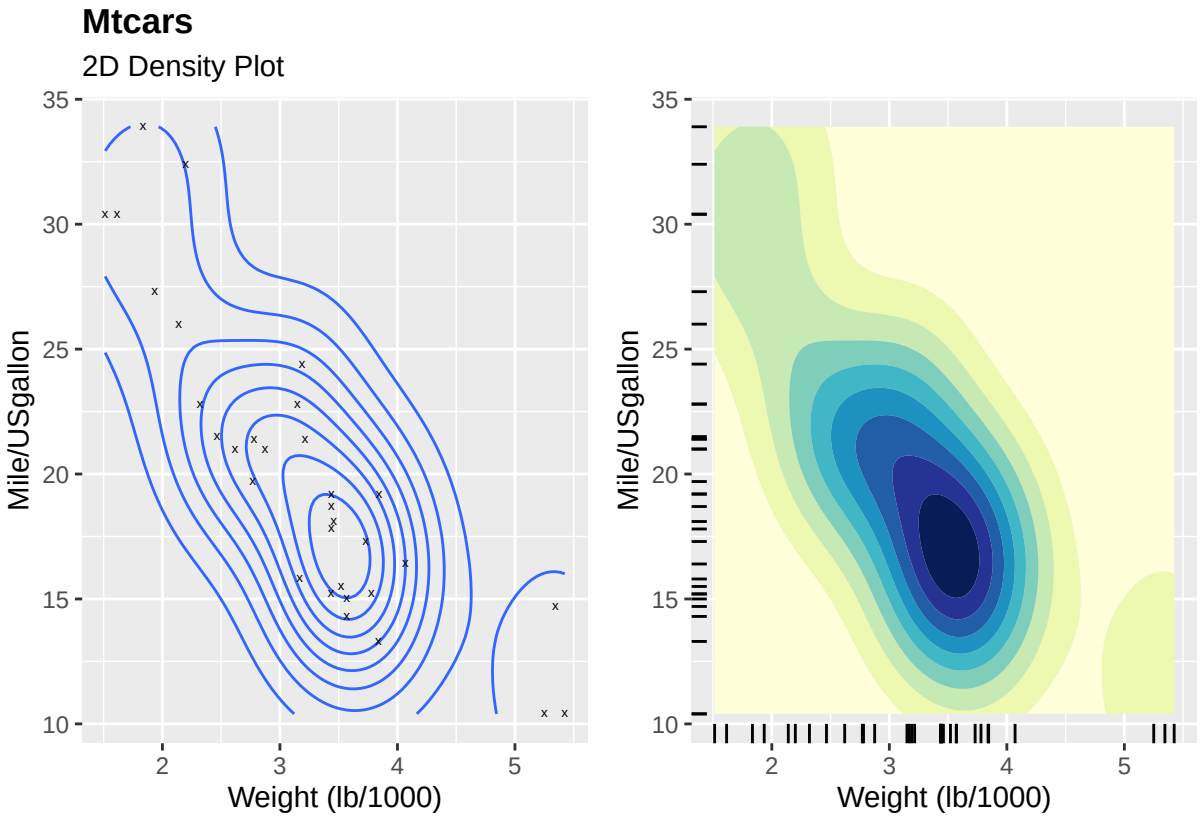
```
origin_plot1 <- ggplot(new_mtcars, aes(origin, fill = origin)) +
  geom_bar(color = "black") +
  labs(title = "Number of cars based on origins",
       x = "Origin countries",
       y = "Count") +
  theme(legend.position = 'none') +
  scale_fill_brewer(palette="YlGnBu") +
  theme(axis.text.x = element_text(angle = 45))

origin_plot2 <- new_mtcars %>%
  group_by(origin) %>%
  summarise(mean_mpg = mean(mpg)) %>%
  ggplot(aes(x = origin, y = mean_mpg, fill = origin)) +
  geom_bar(stat = 'identity', color = 'black') +
  labs(title = "Average mpg of cars based on origins",
       x = "Origin countries",
       y = "Avarage miles per gallon",
       caption = "Illustrated by ggplot2 package") +
  theme(legend.position = 'none') +
  scale_fill_brewer(palette="YlGnBu") +
  theme(axis.text.x = element_text(angle = 45))

origin_plot1 + origin_plot2
```



Most of cars are from the US, and have the lowest average mpg.

Chart 5: The relationship between Mile/USgallon (mpg) and Weight (lb/1000) using 2D density plot

```
mtplot1 <- ggplot(data = mtcars, aes(x = wt, y = mpg)) +
  geom_density_2d() +
  geom_point(shape = "x") +
  labs(title = "Mtcars",
       subtitle = "2D Density Plot",
       x = "Weight (lb/1000)",
       y = "Mile/USgallon") +
  theme(plot.title = element_text(face = "bold"))
  # theme(legend.position = 'none')

mtplot2 <- ggplot(data = mtcars, aes(x = wt, y = mpg)) +
  geom_density_2d_filled() +
  geom_rug() +
  labs(x = "Weight (lb/1000)",
       y = "Mile/USgallon") +
  theme(legend.position = 'none') +
  scale_fill_brewer(palette="YlGnBu")

mtplot1 + mtplot2
```



The charts illustrated that most of cars in this data have the weight between 3000 to 4000 lbs and consume the fuel around 12.5 to 20 mile per US gallon.