FIT5145: Foundation of Data Science

Assignment 2

Student ID: 27030768

A data analysis on Australia's waste generation, recovery and fate. We will start by loading and cleaning the data. Data cleaning done in this first step: 1. "Category" column is found to have inconsistent names.
- For consistency, every category ending with s is converted to singular form.
- "&" is converted to "and".
- "hw" and "toxic waste" are converted to "hazardous waste" as they mean the same. 2. There are a few values in "Category" that are missing,
and a few that are incorrectly encoded.
- Categories that have fewer than 10 entries are converted to match the Category
that most commonly contains its Type.

Type column is clean, so it does not need data cleaning. One entry for "space debris" was found. However, its type, "Defunct satellite",
did not match any other category, so it was left to be its own category.
It has a clear category, so it was not called "Unclassified".
The "broom" and "stringr" library were used for tidier model presentation and
string manipulation respectively. They can be installed with install.packages("broom")
and install.packages("stringr") respectively.

```
# Load all libraries used
# Change directory to file location
library(this.path)
setwd(this.path::here())

library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(stringr) # install.packages("stringr")
library(ggplot2)
library(broom) # install.packages("broom")
wastes <- read.csv("Wastes.csv")

# Data cleaning
# Clean the Category column
wastes <- wastes %>%
    mutate(
        Category = tolower(Category),
        # Remove trailing "s" only if the preceding letter is not "s" or "i"
        Category = str_replace(Category, "(?<![si])s$", ""),
        Category = str_replace(Category, "&", "and"),
        Category = str_trim(Category),
        Category = ifelse(Category == "hw", "hazardous waste", Category),
        Category = ifelse(Category == "toxic waste", "hazardous waste", Category),
    )

# Identify mappings between Type and the most frequent Category for that Type
type_category_map <- wastes %>%
    filter(!is.na(Category) & Category != "") %>%
    group_by(Type, Category) %>%
    summarise(n = n(), .groups = "drop") %>%
    arrange(Type, desc(n)) %>%
    group_by(Type) %>%
    slice_head(n = 1) %>%
    ungroup() %>%
    select(Type, Most_Common_Category = Category)

# Recode missing and low-frequency Categories based on Type
wastes <- wastes %>%
    left_join(type_category_map, by = "Type", suffix = c("", "_matched")) %>%
    mutate(
        Category = ifelse(is.na(Category) | Category == "" | (Category %in% (wastes %>% count
(Category) %>% filter(n < 10) %>% pull(Category))),
            Most_Common_Category,
            Category
        )
    ) %>%
    select(-Most_Common_Category)

# # Calculate sums after recoding
# print(wastes %>% group_by(Category) %>% summarise(Total_Tonnes = sum(Tonnes, na.rm = TRUE),
Count = n(), .groups = "drop") %>% arrange(Category))
```

1. How many unique “Category” values are there in the data file (6 marks)?

```
unique_categories <- unique(wastes$Category)
print(length(unique_categories))
```

```
## [1] 12
```

2. How many negative feedback comments are in the ‘Description’ column with an environmental impact score of 2 or 3 (4 marks)?

For Q2, I am assuming that every row with an environmental impact score of 2 or 3 is one negative feedback comment. Some comments were worded more harshly than others, but that is a subjective measure while “2/10” or “3/10” is objectively a low score. Some rows also had feedback in multiple sentences, while others only had a single sentence, but I considered it to be simply sentence structure and flow of speech.
As per the question, an environmental impact score of 1 is ignored.

```
negative_env_score <- length(grep(" 3/10.| 2/10.", wastes$Description))
print(negative_env_score)
```

```
## [1] 6312
```

3. For each Category value, calculate the fractions of waste tonnes of different waste sources, and then draw a chart to visualise the fraction numbers specific to the Category value.

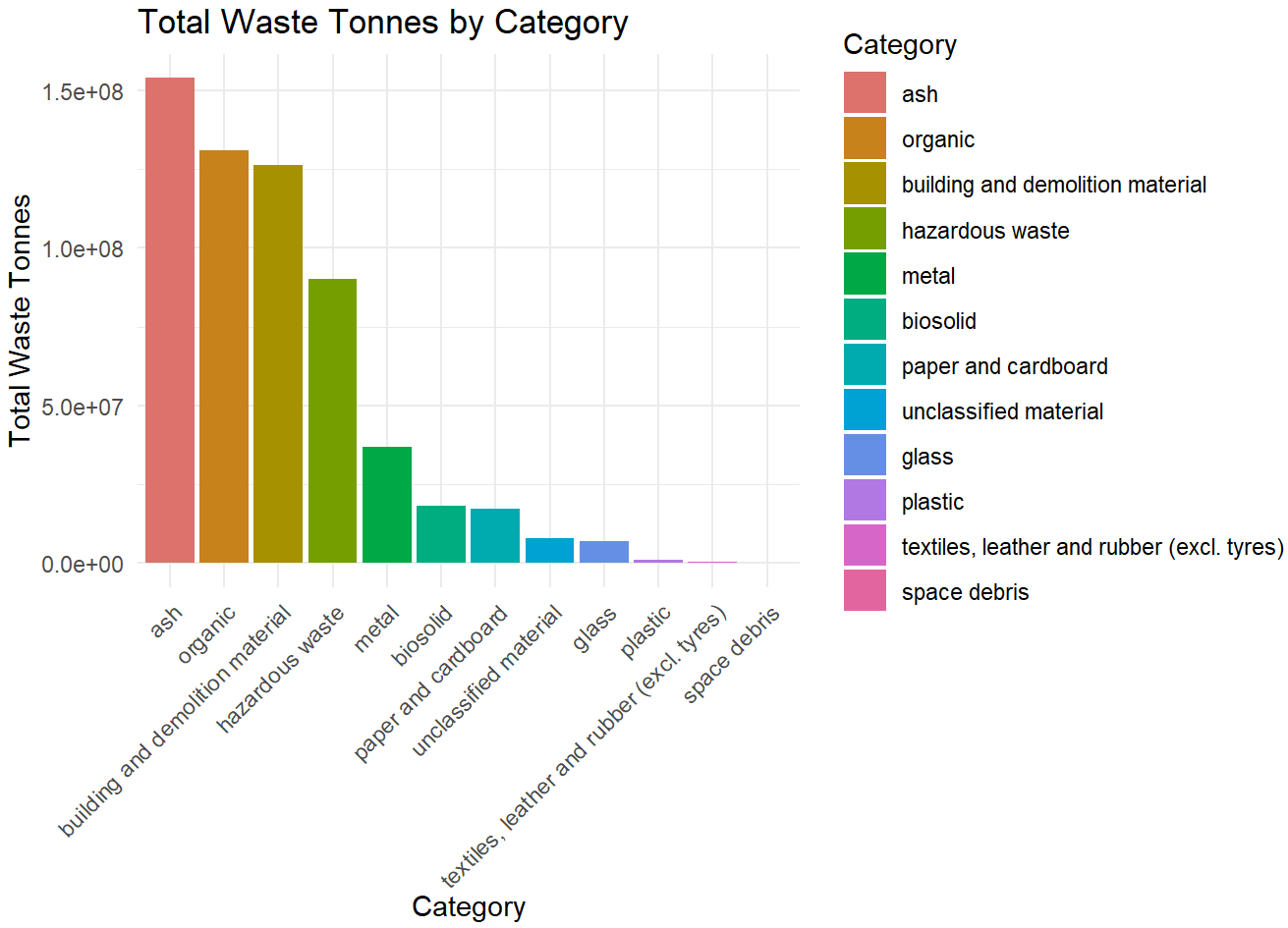
Waste tonnes are summed by each category, and then arranged and drawn in a bar chart. A bar chart was chosen as it has better visual clarity by virtue of not being too compact, unlike a pie chart.

```
# Calculate the sum of 'value' for each category
category_sums <- wastes %>%
    group_by(Category) %>%
    summarise(Total_Category_Wastes = sum(Tonnes, na.rm = TRUE)) %>%
    arrange(desc(Total_Category_Wastes)) # Descending order for plot

# Convert Category to a factor with levels in the desired order
category_sums$Category <- factor(category_sums$Category, levels = category_sums$Category)

# Solid colour bar chart
category_chart <- ggplot(category_sums, aes(x = Category, y = Total_Category_Wastes, fill = C
ategory)) +
    geom_bar(stat = "identity") +
    labs(title = "Total Waste Tonnes by Category", x = "Category", y = "Total Waste Tonnes",
fill = "Category") +
    theme_minimal() +
    theme(
        axis.text.x = element_text(angle = 45, hjust = 1),
        legend.position = "right"
    ) +
    scale_fill_hue(l = 60, c = 80) # Use a rainbow-like hue scale

print(category_chart)
```



4. Add the ‘Year’ and ‘State’ values from Year_State_ID.csv to Wastes.csv, compute the total waste tonnes for each year and state, and store the result in a dataset named ‘temp’. Then, use a single R function/ command to display the statistical information (i.e., Min, Max, and Mean) of the total waste tonnes for each state in ‘temp’.

The contents of Year_State_ID.csv was merged into the dataframe for Wastes.csv by matching ID from Year_State_ID.csv to Year_State_ID from Wastes.csv. To avoid duplicate columns, the newly-merged ID column was removed. “Australia” was removed from the State columns as it is not a State.

```
year_state <- read.csv("Year_State_ID.csv")
# Merge the data frames based on ID -> Year_State_ID
wastes_merged <- merge(wastes, year_state[, c("ID", "Year", "State", "Economic_Growth")], by.
x = "Year_State_ID", by.y = "ID", all.x = FALSE)
wastes_merged$ID <- NULL

temp <- wastes_merged %>%
    filter(State != "Australia") %>%
    group_by(State)

# Compute and display the stats of waste tonnes for each year and state
temp %>% summarise(
        Min_Tonnes = min(Tonnes, na.rm = TRUE),
        Max_Tonnes = max(Tonnes, na.rm = TRUE),
        Mean_Tonnes = mean(Tonnes, na.rm = TRUE),
        Total_Tonnes = sum(Tonnes, na.rm = TRUE),
        SD_Tonnes = sd(Tonnes, na.rm = TRUE),
        Q1_Tonnes = quantile(Tonnes, 0.25, na.rm = TRUE),
        Median_Tonnes = median(Tonnes, na.rm = TRUE),
        Q3_Tonnes = quantile(Tonnes, 0.75, na.rm = TRUE),
        Count = n() # Number of observations for each state
    )
```

```
## # A tibble: 8 × 10
##   State Min_Tonnes Max_Tonnes Mean_Tonnes Total_Tonnes SD_Tonnes Q1_Tonnes
##   <chr>      <dbl>      <dbl>       <dbl>        <dbl>     <dbl>     <dbl>
## 1 ACT     0           209880.        852.     3924336.     7459.         0
## 2 NSW     0          4814588       46840.   209282905.   294289.         0
## 3 NT      0            62474         536.     2296427.     3039.         0
## 4 Qld    -2.02e- 8   4192637.      28069.   121680707.   215470.         0
## 5 SA      0          1308587        6937.    34756803.    45788.         0
## 6 Tas     0           249476.       1855.     7991224.    12271.         0
## 7 Vic     0          4478596.      32069.   152135435.   172091.         0
## 8 WA     -9.83e-14   1409354       11718.    57990754.    63938.         0
## # ℹ 3 more variables: Median_Tonnes <dbl>, Q3_Tonnes <dbl>, Count <int>
```

5. Draw a chart showing a yearly trend of total waste tonnes of food organics for each state. Convert the Year-Year formats of all Year values into Year formats.

The Year-Year format is converted into Year formats by observing the [x]-[y] format and taking the first number as separated by “-”. A line chart is chosen to easily show the changes in totals for each state as the years pass.

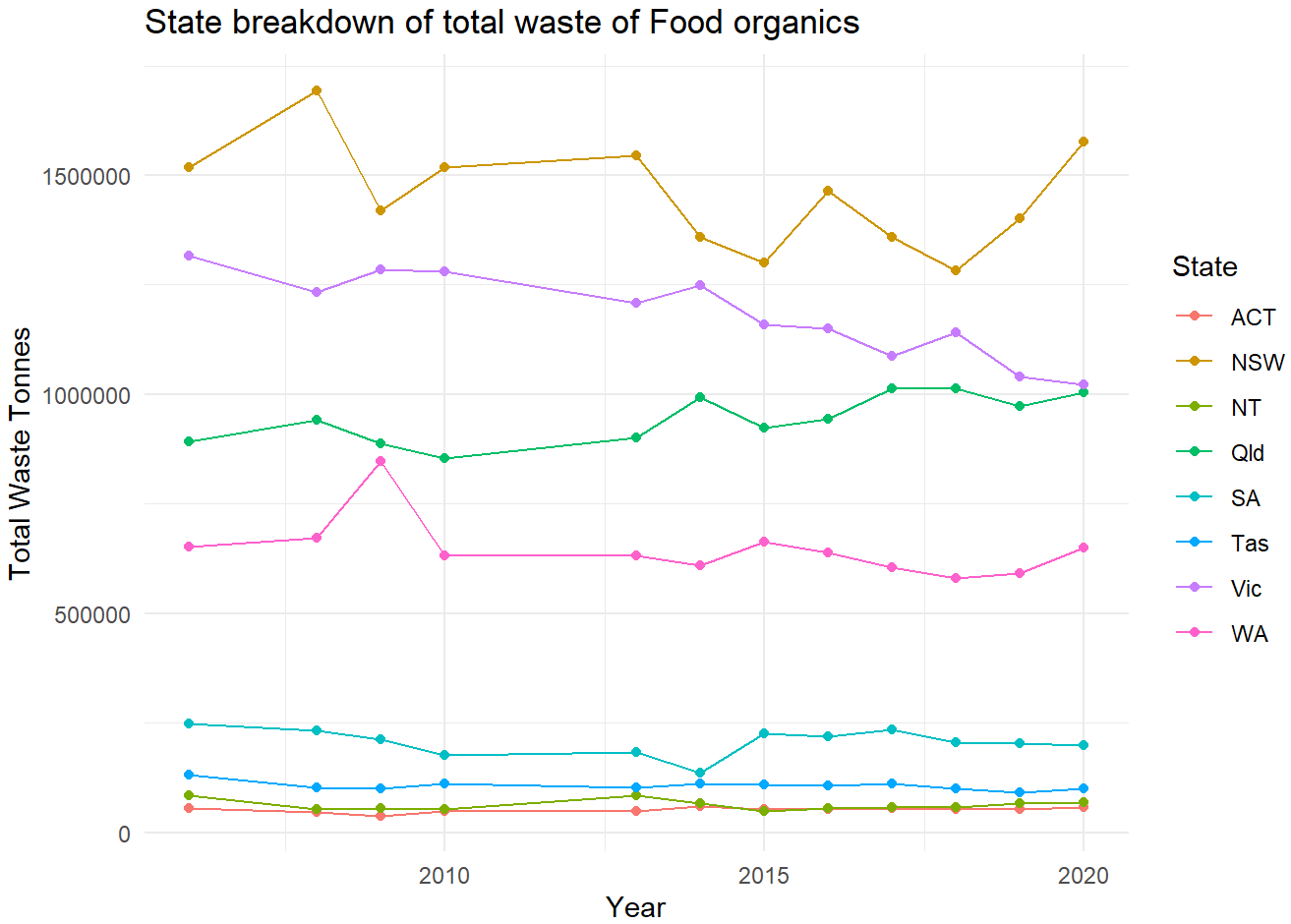
```
# Extract only the first year of the range
wastes_merged$Year <- as.numeric(sub("-.*", "", wastes_merged$Year))

filtered_type <- "Food organics"

# Filter data for Type = "Food organics"
waste_type_yearly <- wastes_merged %>%
    filter(Type == filtered_type) %>%
    group_by(Year, State) %>%
    summarise(Total_Waste_Tonnes = sum(Tonnes, na.rm = TRUE), .groups = "drop")

# Coloured line chart
yearly_trend_chart <- ggplot(waste_type_yearly, aes(x = Year, y = Total_Waste_Tonnes, color =
State)) +
    geom_line() +
    geom_point() +
    labs(
        title = paste("State breakdown of total waste of", filtered_type),
        x = "Year",
        y = "Total Waste Tonnes",
        color = "State"
    ) +
    theme_minimal() +
    theme(legend.position = "right")

# Display the chart
print(yearly_trend_chart)
```



6. Display the most recycled waste Type and the most disposed waste Type with the corresponding year. Also display the most increased waste Type over years.

```
# Most Recycled Waste Type and Year
most_recycled <- wastes_merged %>%
  filter(Fate == "Recycling") %>%
  group_by(Year, Type) %>%
  summarise(Total_Recycled = sum(Tonnes, na.rm = TRUE), .groups = "drop") %>%
  arrange(desc(Total_Recycled)) %>%
  slice_head(n = 1) %>%
  select(Year, Type, Total_Recycled) %>%
  ungroup()

recycled_output <- paste0(
  "--- Most Recycled Waste ---\n",
  "Type: ", most_recycled$Type, "\n",
  "Year: ", most_recycled$Year, "\n",
  "Total Recycled (Tonnes): ", most_recycled$Total_Recycled, "\n\n"
)

# Most Disposed Waste Type and Year
most_disposed <- wastes_merged %>%
  filter(Fate == "Disposal") %>%
  group_by(Year, Type) %>%
  summarise(Total_Disposed = sum(Tonnes, na.rm = TRUE), .groups = "drop") %>%
  arrange(desc(Total_Disposed)) %>%
  slice_head(n = 1) %>%
  select(Year, Type, Total_Disposed) %>%
  ungroup()

disposed_output <- paste0(
  "--- Most Disposed Waste ---\n",
  "Type: ", most_disposed$Type, "\n",
  "Year: ", most_disposed$Year, "\n",
  "Total Disposed (Tonnes): ", most_disposed$Total_Disposed, "\n\n"
)

# Most Increased Waste Type Over the Years
# Sum by year
yearly_waste_trends <- wastes_merged %>%
  group_by(Year, Type) %>%
  summarise(Total_Tonnes = sum(Tonnes, na.rm = TRUE), .groups = "drop") %>%
  arrange(Year, Type) %>%
  ungroup()

# Get total change over the years
most_increased <- yearly_waste_trends %>%
  group_by(Type) %>%
  summarise(
    Total_Increase = last(Total_Tonnes) - first(Total_Tonnes),
    .groups = "drop"
  ) %>%
  arrange(desc(Total_Increase)) %>%
  slice_head(n = 1) %>%
```

```
  select(Type, Total_Increase) %>%
  ungroup()

increased_output <- paste0(
  "--- Most Increased Waste (by Tonnes) ---\n",
  "Type: ", most_increased$Type, "\n",
  "Increase (Tonnes): ", most_increased$Total_Increase, "\n"
)

# Combine all output into a single cat() call
final_output <- paste0(recycled_output, disposed_output, increased_output)
cat(final_output)
```

```
## --- Most Recycled Waste ---
## Type: Bricks, concrete and pavers
## Year: 2020
## Total Recycled (Tonnes): 10478680.7224156
##
## --- Most Disposed Waste ---
## Type: Food organics
## Year: 2008
## Total Disposed (Tonnes): 4089186
##
## --- Most Increased Waste (by Tonnes) ---
## Type: Bricks, concrete and pavers
## Increase (Tonnes): 8345528.72241561
```

7. Investigate the factors influencing environmental impact scores. Analyze, discuss and reason the insights and conclusions.

Environmental score is extracted by extracting the first number from the format: “Environmental Impact Score: [x]/10.”
The variables analysed for impact are Economic_Growth, Tonnes, Stream, Fate, Core_Non.core, Category, Type. As this includes categorical data, a multiple regression analysis is chosen. The factors found are displayed as a table, arranged in descending order of magnitude of correlation, after filtering for $p < 0.05$ for statistically significant results.

Unsurprisingly, recycling was associated with higher environmental impact scores, and plastic with lower scores, but perhaps less obvious is the high impact of cardboard being good.

Overall, the model only had a R-squared score of 0.238931018144097 and an adjusted R-squared of 0.237725940313149, showing environmental impact is a complex phenomena with many more factors than these. It could also simply be a non-linear correlation, as regression analysis analyzes linear relationships.

```
# Extract the Environmental Impact Score
wastes_impact <- wastes_merged %>%
    mutate(
        Impact_Score_Text = str_extract(Description, "Environmental Impact Score: [0-9]+/10\
\."),
        Environmental_Impact_Score = as.numeric(str_extract(Impact_Score_Text, "[0-9]+"))
    ) %>%
    filter(!is.na(Environmental_Impact_Score)) # Keep only rows with scores

# Convert categorical variables to factors for regression
wastes_impact <- wastes_impact %>%
    mutate(
        Stream = as.factor(Stream),
        Fate = as.factor(Fate),
        Core_Non_core = as.factor(`Core_Non.core`), # Be mindful of the hyphen
        Category = as.factor(Category),
        Type = as.factor(Type)
    )

# Multiple Regression Analysis
# Include all relevant predictors
model <- lm(Environmental_Impact_Score ~ Economic_Growth + Tonnes + Stream +
    Fate + `Core_Non.core` + Category + Type, data = wastes_impact)

# The model is difficult to parse. Use tidy() from broom library for easier viewing.
tidy_model <- tidy(model)
# Filter for statistically significant terms (p-value < 0.05), then arrange for highest estim
ate first.
tidy_model <- tidy_model %>%
    filter(p.value < 0.05) %>%
    arrange(desc(abs(estimate)))

print(tidy_model)
```

```
## # A tibble: 36 × 5
##    term                                  estimate std.error statistic  p.value
##    <chr>                                    <dbl>     <dbl>     <dbl>    <dbl>
##  1 (Intercept)                               4.75     0.112      42.4  0
##  2 FateRecycling                             1.36     0.0164     83.1  0
##  3 TypeCertified compostable plastics       -1.06     0.217      -4.91 9.17e- 7
##  4 TypeCardboard                             1.04     0.129       8.06 7.78e-16
##  5 FateLong-term storage                    -0.999    0.233      -4.29 1.77e- 5
##  6 TypeNewsprint & magazines                 0.905    0.138       6.58 4.77e-11
##  7 TypeAsbestos (N220)                      -0.763    0.0474    -16.1  4.19e-58
##  8 TypeGlass from food and beverage conta…   0.740    0.246       3.01 2.66e- 3
##  9 TypeCeramics, tiles and pottery          -0.700    0.307      -2.28 2.26e- 2
## 10 Core_Non.coreNon-core waste               0.692    0.201       3.45 5.70e- 4
## # ℹ 26 more rows
```

```
# # Extract the R-squared and Adjusted R-squared values
# env_impact_r_squared <- summary(model)$r.squared
# env_impact_adjusted_r_squared <- summary(model)$adj.r.squared
#
# # Paste them into a single string variable
# env_impact_r_squared_values <- paste("\nR-squared:", env_impact_r_squared,
#                                      "\nAdjusted R-squared:", env_impact_adjusted_r_square
d)
#
# # Print the combined string
# cat("\nMultiple Regression Analysis:", env_impact_r_squared_values, "\n")
```

8. Filtering the data records of non-zero Category values of Hazardous wastes, and Type values of Tyres (T140), add a new column “Tonnes_range” categorizing it into the following bins: ○ [0, 10000) ○ [10000, 20000) ○ [20000, 40000) ○ [40000, 80000]

Then, for each state, display a chart to show the number of cases of different score_range values. What is observed?

There is a clear correlation of more populated states having cases of higher score_range values.
The lower populated states have more distributed cases with lower score_range on each case.

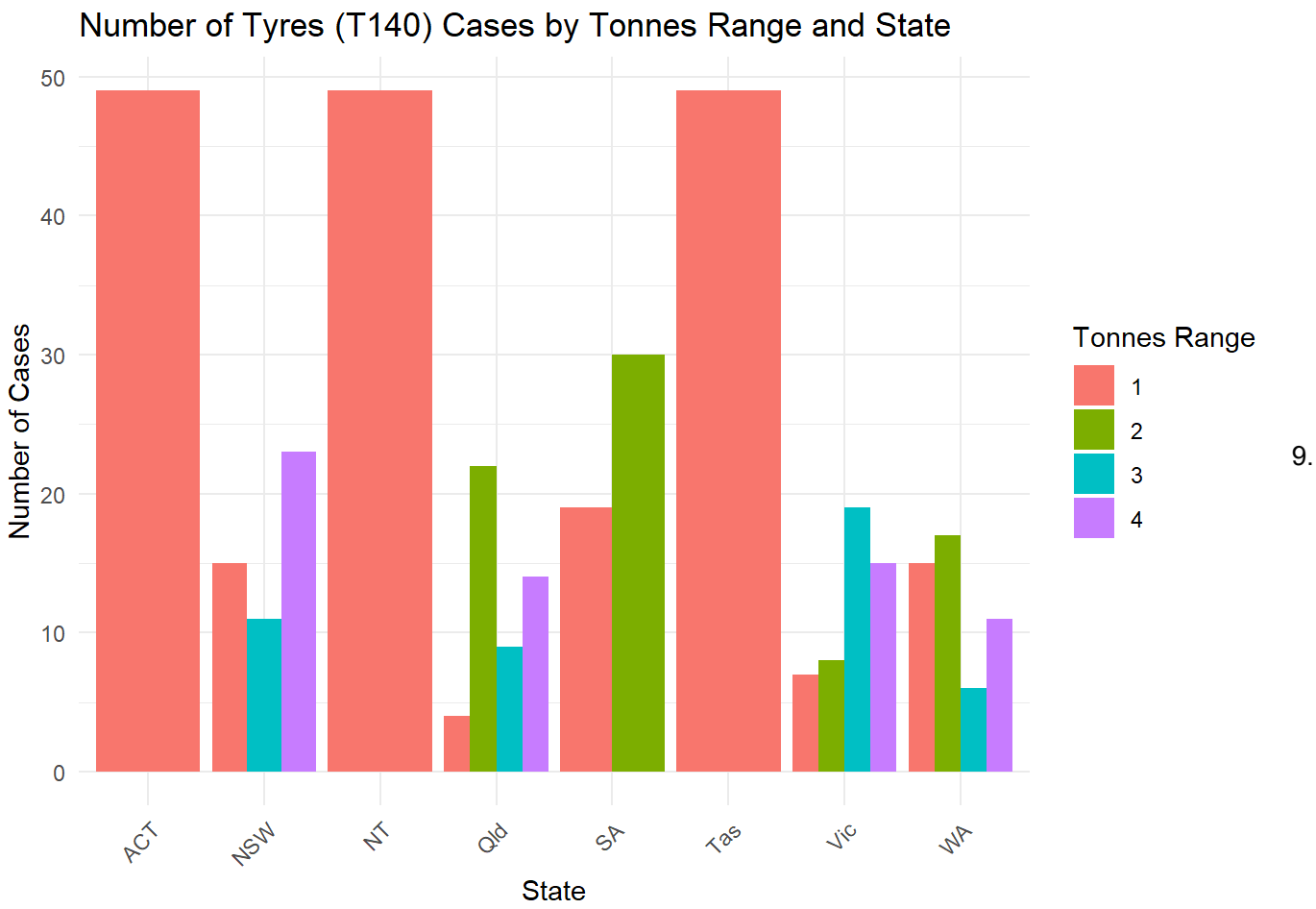
```
# Filter the data for Hazardous waste, Tyres (T140), Tonnes > 0
filtered_tyres <- wastes_merged %>%
    filter(Category == "hazardous waste", Type == "Tyres (T140)", Tonnes > 0)

# Add the Tonnes_range column
filtered_tyres <- filtered_tyres %>%
    mutate(
        Tonnes_range = case_when(
            Tonnes >= 0 & Tonnes < 10000 ~ "1",
            Tonnes >= 10000 & Tonnes < 20000 ~ "2",
            Tonnes >= 20000 & Tonnes < 40000 ~ "3",
            Tonnes >= 40000 & Tonnes <= 80000 ~ "4",
            TRUE ~ NA_character_ # Handle any values outside the specified ranges
        ),
        Tonnes_range = as.factor(Tonnes_range) # Convert to categorical factor
    )

# Count the number of cases for each Tonnes_range within each State
state_tonnes_range_counts <- filtered_tyres %>%
    group_by(State, Tonnes_range) %>%
    summarise(n = n(), .groups = "drop")

# Create bar chart
tonnes_range_by_state_chart <- ggplot(
    state_tonnes_range_counts,
    aes(x = State, y = n, fill = Tonnes_range)
) +
    geom_bar(stat = "identity", position = "dodge") +
    labs(
        title = "Number of Tyres (T140) Cases by Tonnes Range and State",
        x = "State",
        y = "Number of Cases",
        fill = "Tonnes Range"
    ) +
    theme_minimal() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))

# Display the bar chart
print(tonnes_range_by_state_chart)
```



9.

Investigate the factors influencing the yearly trend of total C&D waste tonnes. Analyze, discuss and reason the insights and conclusions.

A multiple regression analysis is run with all the variables found in Wastes.csv and Year_State_ID.csv. An additional data column of total waste tons per year is added to measure its impact. “Stream” and `Core-Non.core` columns are omitted, as after filtering to only items within the C&D stream, they all fall within the “C&D” stream and the “Core waste” group.

The R-squared score of this model is better than the previous, at R-squared: 0.333232778990791 and Adjusted R-squared: 0.329339140525223. No external data sources were used. Although this is a better R-squared score, it is still quite low, indicating many more factors are at play, or that it is a non-linear relationship.

```
# Calculate Total Waste Tonnes per Year across all streams
total_yearly_waste <- wastes_merged %>%
    group_by(Year) %>%
    summarise(Total_All_Waste_Tonnes = sum(Tonnes, na.rm = TRUE), .groups = "drop")

# Aggregate C&D Waste by Year
yearly_cd_data <- wastes_merged %>%
    filter(Stream == "C&D") %>%
    group_by(Year, Economic_Growth, Fate, Category, Type) %>%
    summarise(Total_C_D_Tonnes = sum(Tonnes, na.rm = TRUE), .groups = "drop")

# Merge Total Yearly Waste with C&D Yearly Data
yearly_cd_trends <- yearly_cd_data %>%
    left_join(total_yearly_waste, by = "Year")

# Convert categorical variables to factors for regression
yearly_cd_trends <- yearly_cd_trends %>%
    mutate(
        Fate = as.factor(Fate),
        Category = as.factor(Category),
        Type = as.factor(Type)
    )

# Multiple Regression Analysis
model_cd_yearly_all <- lm(
    Total_C_D_Tonnes ~ Economic_Growth + Total_All_Waste_Tonnes +
        Fate + Category + Type,
    data = yearly_cd_trends
)

# The model is difficult to parse. Use tidy() from broom library for easier viewing.
tidy_model_cd_yearly_all <- tidy(model_cd_yearly_all)

# Filter for statistically significant terms (p-value < 0.05)
significant_terms_cd_yearly_all <- tidy_model_cd_yearly_all %>%
    filter(p.value < 0.05) %>%
    arrange(desc(abs(estimate)))

print(significant_terms_cd_yearly_all)
```

```
## # A tibble: 20 × 5
##    term                                  estimate std.error statistic  p.value
##    <chr>                                    <dbl>     <dbl>     <dbl>    <dbl>
##  1 TypeBricks, concrete and pavers          6.75e+5   4.41e+4     15.3  3.44e-52
##  2 TypeRubble                               4.62e+5   4.52e+4     10.2  2.18e-24
##  3 Categoryglass                           -3.30e+5   6.21e+4     -5.31 1.11e- 7
##  4 Categoryplastic                         -3.27e+5   4.62e+4     -7.09 1.45e-12
##  5 Categorypaper and cardboard             -3.26e+5   4.38e+4     -7.45 9.90e-14
##  6 TypeCeramics, tiles and pottery         -3.26e+5   6.21e+4     -5.25 1.51e- 7
##  7 Categorytextiles, leather and rubber (… -3.26e+5   4.41e+4     -7.40 1.52e-13
##  8 Categoryhazardous waste                 -3.22e+5   4.10e+4     -7.85 4.66e-15
##  9 (Intercept)                              3.21e+5   4.07e+4      7.90 3.16e-15
## 10 TypePlasterboard & cement sheeting      -3.18e+5   4.41e+4     -7.22 5.54e-13
## 11 Categorymetal                           -3.18e+5   4.41e+4     -7.21 5.91e-13
## 12 Categoryorganic                         -2.95e+5   4.13e+4     -7.14 1.01e-12
## 13 FateWaste reuse                         -2.30e+5   7.82e+4     -2.94 3.30e- 3
## 14 Categoryunclassified material           -2.22e+5   4.62e+4     -4.81 1.54e- 6
## 15 TypeAsphalt                             -2.09e+5   4.42e+4     -4.73 2.24e- 6
## 16 TypeIron and steel                       9.06e+4   2.45e+4      3.70 2.19e- 4
## 17 TypeContaminated soils (N120)            5.97e+4   1.12e+4      5.32 1.07e- 7
## 18 TypeFood organics                       -2.82e+4   1.33e+4     -2.12 3.42e- 2
## 19 TypeOther organics                      -2.78e+4   1.34e+4     -2.07 3.83e- 2
## 20 Total_All_Waste_Tonnes                   1.85e-4   9.27e-5      1.99 4.62e- 2
```

```
# # Extract the R-squared and Adjusted R-squared values
# cd_r_squared <- summary(model_cd_yearly_all)$r.squared
# cd_adjusted_r_squared <- summary(model_cd_yearly_all)$adj.r.squared
#
# # Paste them into a single string variable
# cd_r_squared_values <- paste("\nR-squared:", cd_r_squared,
#                                     "\nAdjusted R-squared:", cd_adjusted_r_squared)
#
# # Print the combined string
# cat("\nMultiple Regression Analysis:", cd_r_squared_values, "\n")
```