Assignment 2

ETC1010_5510

T11_Wed_skimr(Hanchen Wang, Hao Li, Jiaying Zhang, Mohammed Faizan, Karan Garg

Tuesday, May 18 2021

```
library(naniar)
library(broom)
library(ggmap)
library(knitr)
library(lubridate)
library(timeDate)
library(tsibble)
library(here)
library(readr)
library(tidyverse)
library(kableExtra)
library(ggResidpanel)
library(gridExtra)
```

```
tree_data0 <- read_csv("Data/Assignment_data.csv")
```

Part I

Question 1: Rename the variables *Date Planted* and *Year Planted* to *Dateplanted* and *Yearplanted* using the *rename()* function. Make sure *Dateplanted* is defined as a date variable. Then extract from the variable *Dateplanted* the year and store it in a new variable called *Year*. Display the first 6 rows of the data frame. (5pts)

```
tree_data <- as.tibble(tree_data0) %>% rename(Dateplanted=c("Date Planted"),
                                   Yearplanted=c("Year Planted")) %>%
  mutate(Dateplanted = dmy(Dateplanted)) %>%
  mutate(Year = year(Dateplanted))
head(tree_data) %>%
  kable(caption = "Tree Data: All Variables") %>%
  kable_styling(latex_options = c("scale_down", "hold_position"))
```

Table 1: Tree Data: All Variables

CoM ID	Common Name	Scientific Name	Genus	Family	Diameter Breast Height	Yearplanted	Dateplanted	Age Description	Useful Life Expectency	Useful Life Expectency Value	Precinct	Located in	UploadDate	CoordinateLocation	Latitude	Longitude	Easting	Northing	Year
1057605	White Poplar	Populus alba	Populus	Salicaceae	NA	1900	2000-01-01	NA	NA	NA	NA	Park	30/9/20	(-37.790893782047654, 144.92257141425543)	-37.79089	144.9226	317080.6	5815353	2000
1028440	London Plane	Platanus x acerifolia	Platanus	Platanaceae	62	1900	2000-01-02	Mature	6-10 years (>50% canopy)	10	NA	Park	30/9/20	(-37.84544751087332, 144.97904358884796)	-37.84545	144.9790	322184.5	5809408	2000
1058665	Small-leaved Linden	Tilia cordata	Tilia	Malvaceae	19	2000	2000-05-29	Semi-Mature	31-60 years	60	NA	Street	30/9/20	(-37.798902715216066, 144.96852132025538)	-37.79890	144.9685	321146.2	5814553	2000
1026352	Variegated Elm	Ulmus minor	Ulmus	Ulmaceae	26	1900	2000-01-02	Semi-Mature	21-30 years	30	NA	Street	30/9/20	(-37.81160317631319, 144.98355174888286)	-37.81160	144.9836	322500.1	5813172	2000
1038440	Canary Island Pine	Pinus canariensis	Pinus	Pinaceae	91	1900	2000-01-02	Mature	31-60 years	60	NA	Park	30/9/20	(-37.78240938533976, 144.96019784543748)	-37.78241	144.9602	320373.4	5816367	2000
1015128	London Plane	Platanus x acerifolia	Platanus	Platanaceae	99	1900	2000-01-02	Mature	11-20 years	20	NA	Street	30/9/20	(-37.79941634390208, 144.94984511347445)	-37.79942	144.9498	319503.0	5814460	2000

Question 2: Have you noticed any differences between the variables *Year* and *Yearplanted*? Why is that? Demonstrate your claims using R code. Fix the problem if there is one (Hint: Use *ifelse* inside a mutate function to fix the problem and store the data in *tree_data_clean*). After this question, please use the data in *tree_data_clean* to proceed. (3pts)

The corresponding values in the variables *Year* and *Yearplanted* are different. The newly created variable *Year* contains the year 2000 in all observations but one(1977). Correct value for the year of tree plantation is present in *Yearplanted*.

This difference is because the original variable, *Date Planted* in tree_data0 has the *Date Planted* as "2/1/00" where the dmy() interprets the year "00" as 2000 and hence for both 1900 and 2000. The year 1977 is mapped correctly owing to the fact that 2077 has not yet arrived. These claims can be seen below.

```
tree_data %>%
  select(`CoM ID`, Yearplanted, Dateplanted, Year) %>%
  filter(`CoM ID` %in% c("1028440", "1058665", "1060068")) %>%
  kable(caption = "Mismatching Years") %>%
  kable_styling(latex_options = c("hold_position"))
```

Table 2: Mismatching Years

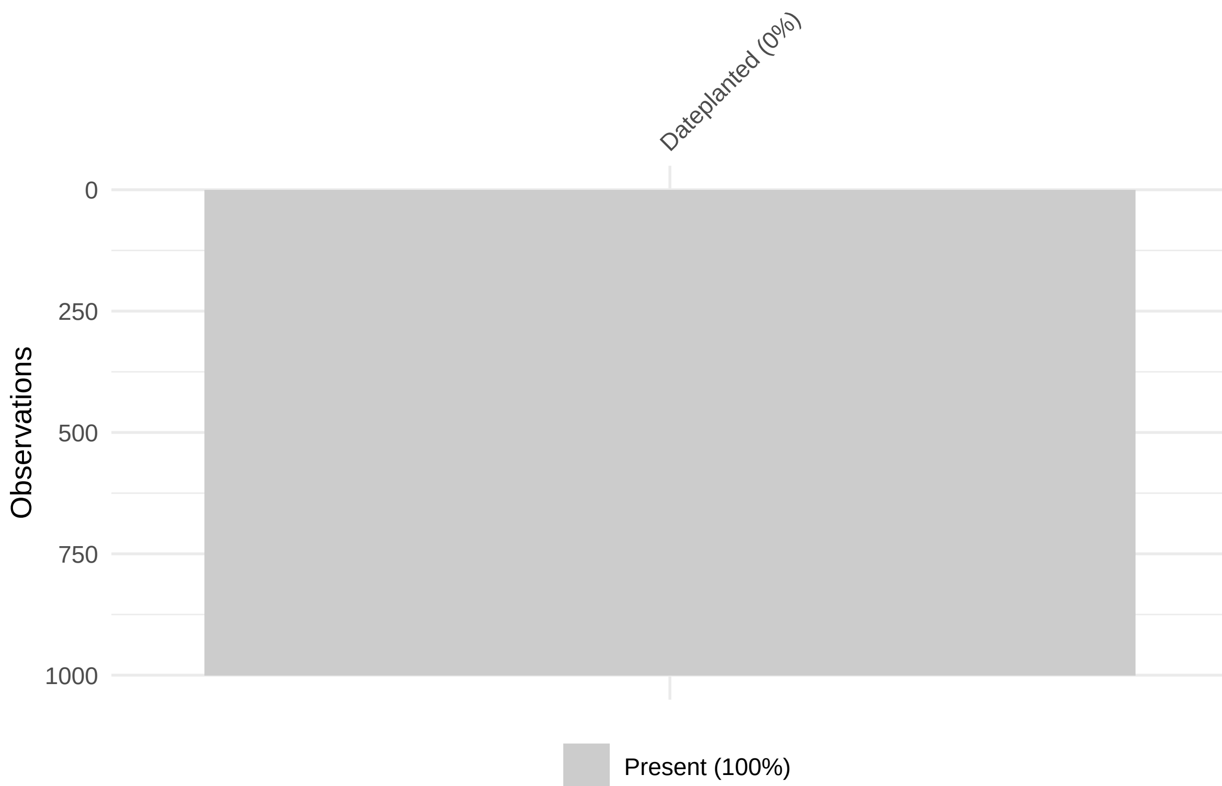
CoM ID	Yearplanted	Dateplanted	Year
1028440	1900	2000-01-02	2000
1058665	2000	2000-05-29	2000
1060068	1977	1977-07-07	1977

```
tree_data_clean <- tree_data %>%
  mutate(Dateplanted = str_replace(as.character(Dateplanted),
                                   "2000", as.character(Yearplanted))) %>%
  mutate(Year = Yearplanted) %>%
  mutate(Dateplanted = ymd(Dateplanted))
```

Question 3: Investigate graphically the missing values in the variable *Dateplanted* for the last 1000 rows of the data set. What do you observe? (max 30 words) (2pts)

We don't see any missing values in **"Dateplanted"**.

```
tree_data_singlevariable <- tree_data_clean %>%
  select(Dateplanted) %>%
  tail(1000)

vis_miss(tree_data_singlevariable)
```



Question 4: What is the proportion of missing values in each variable in the tree data set? Display the results in descending order of the proportion. (2pts)

The missingness in the variables of the tree data set is listed below in decsending order of proportion.

```
miss_var_summary(tree_data_clean) %>%
  mutate(pct_miss = round(pct_miss/100,3)) %>%
  rename(prop_miss = pct_miss) %>%
  kable(caption = "Proportion of missing values in each variable") %>%
  kable_styling(latex_options = "hold_position")
```

Table 3: Proportion of missing values in each variable

variable	n_miss	prop_miss
Precinct	6828	1.000
Diameter Breast Height	1454	0.213
Age Description	1454	0.213
Useful Life Expectency	1454	0.213
Useful Life Expectency Value	1454	0.213
Dateplanted	2	0.000
Common Name	1	0.000
Located in	1	0.000
CoM ID	0	0.000
Scientific Name	0	0.000
Genus	0	0.000
Family	0	0.000
Yearplanted	0	0.000
UploadDate	0	0.000
CoordinateLocation	0	0.000
Latitude	0	0.000
Longitude	0	0.000
Easting	0	0.000
Northing	0	0.000
Year	0	0.000

Question 5: How many observations have a missing value in the variable *Dateplanted*? Identify the rows and display the information in those rows. Remove all the rows in the data set of which the variable *Dateplanted* has a missing value recorded and store the data in *tree_data_clean1*. Display the first 4 rows of *tree_data_clean1*. Use R inline code to complete the sentense below. (6pts)

There are 2 observations with missing values in Dateplanted variable.

```
tree_data_clean %>%
  filter(is.na(Dateplanted)) %>%
  kable(caption = "Observations with missing Dateplanted") %>%
  kable_styling(latex_options = c("hold_position", "scale_down"))
```

Table 4: Observations with missing Dateplanted

CoM ID	Common Name	Scientific Name	Genus	Family	Diameter Breast Height	Yearplanted	Dateplanted	Age Description	Useful Life Expectency	Useful Life Expectency Value	Precinct	Located in	UploadDate	CoordinateLocation	Latitude	Longitude	Easting	Northing	Year
1024155	Cyprus Plane	Platanus orientalis	Platanus	Platanaceae	22	1900	NA	Semi-Mature	21-30 years	30	NA	Street	30/9/20	(-37.81605293763187, 144.95571785206653)	-37.81605	144.9557	320060.5	5812625	1900
1023092	London Plane	Platanus x acerifolia	Platanus	Platanaceae	29	1900	NA	Semi-Mature	6-10 years (>50% canopy)	10	NA	Street	30/9/20	(-37.80199573215045, 144.96671419201465)	-37.80200	144.9667	320994.6	5814206	1900

```
tree_data_clean1 <- tree_data_clean %>%
  filter(!is.na(Dateplanted))
  head(tree_data_clean1, 4) %>%
kable(caption = "tree_data_clean1") %>%
  kable_styling(latex_options = c("scale_down", "hold_position"))
```

\begin{table}[!h]

\caption{tree_data_clean1}

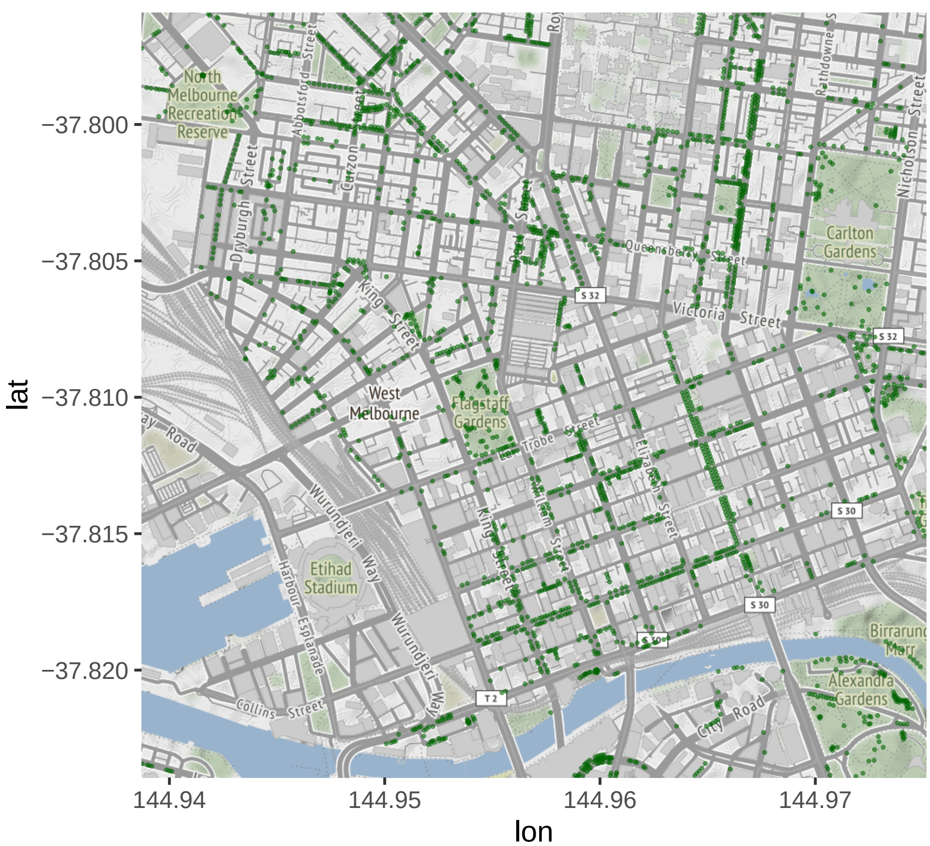
CoM ID	Common Name	Scientific Name	Genus	Family	Diameter Breast Height	Yearplanted	Dateplanted	Age Description	Useful Life Expectency	Useful Life Expectency Value	Precinct	Located in	UploadDate	CoordinateLocation	Latitude	Longitude	Easting	Northing	Year
1057605	White Poplar	Populus alba	Populus	Salicaceae	NA	1900	1900-01-01	NA	NA	NA	NA	Park	30/9/20	(-37.790893782047654, 144.92257141425543)	-37.79089	144.9226	317080.6	5815353	1900
1028440	London Plane	Platanus x acerifolia	Platanus	Platanaceae	62	1900	1900-01-02	Mature	6-10 years (>50% canopy)	10	NA	Park	30/9/20	(-37.84544751087332, 144.97904358884796)	-37.84545	144.9790	322184.5	5809408	1900
1058665	Small-leaved Linden	Tilia cordata	Tilia	Malvaceae	19	2000	2000-05-29	Semi-Mature	31-60 years	60	NA	Street	30/9/20	(-37.798902715216066, 144.96852132025538)	-37.79890	144.9685	321146.2	5814553	2000
1026352	Variegated Elm	Ulmus minor	Ulmus	Ulmaceae	26	1900	1900-01-02	Semi-Mature	21-30 years	30	NA	Street	30/9/20	(-37.81160317631319, 144.98355174888286)	-37.81160	144.9836	322500.1	5813172	1900

\end{table}

The number of rows in the cleaned data set are 6826 and the number of columns are 20

Question 6: Create a map with the tree locations in the data set. (2pts)

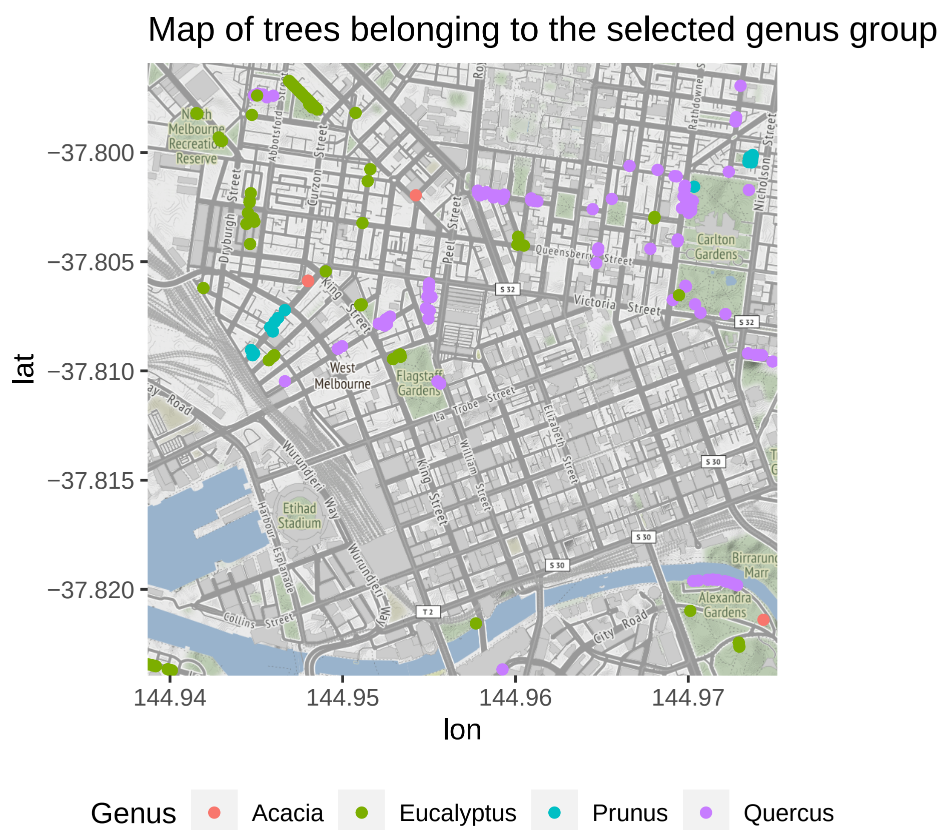
```r
# We have created the map below for you
melb_map <- read_rds(here::here("Data/melb-map.rds"))

# Here you just need to add the location for each tree into the map.
ggmap(melb_map) +
  geom_point(data = tree_data_clean1,
             aes(x = Longitude,
                 y = Latitude),
             colour = "#006400",
             alpha = 0.6,
             size = 0.2)
```



Question 7: Create another map and draw trees in the *Genus* groups of Eucalyptus, Macadamia, Prunus, Acacia, and Quercus. Use the “Dark2” color palette and display the legend at the bottom of the plot. (8pts)

```
selected_group <- tree_data_clean1 %>%
  filter(Genus %in% c("Eucalyptus","Macadamia","Prunus","Acacia","Quercus"))
```

```
ggmap(melb_map) +
  geom_point(data = selected_group,
             aes(x = Longitude,
                 y = Latitude,
                 color = Genus)) +
  scale_fill_brewer(palette="Dark2") +
  theme(legend.position = "bottom") +
  labs(title = "Map of trees belonging to the selected genus group")
```



Question 8: Filter the data *tree_data_clean1* so that only the variables *Year*, *Located in*, and *Common Name* are displayed. Arrange the data set by *Year* in descending order and display the first 4 lines. Call this new data set *tree_data_clean_filter*. Then answer the following question using inline R code: When (*Year*), where (*Located in*) and what tree (*Common Name*) was the first tree planted in Melbourne according to this data set? (8pts)

```
tree_data_clean_filter <- tree_data_clean1 %>%
 select(Year,`Located in`,`Common Name`) %>%
  arrange(-Year)


head(tree_data_clean_filter,4) %>%
  kable(caption = "Selected Variables of Tree Data") %>%
  kable_styling(latex_options = "hold_position")
```

Table 5: Selected Variables of Tree Data

Year	Located in	Common Name
2000	Street	Small-leaved Linden
2000	Street	Spotted Gum
2000	Street	Drooping sheoak
2000	Park	Kanooka

The first tree was planted in 1900 at a Street and the tree name is London Plane

Question 9: How many trees were planted in parks and how many in streets? Tabulate the results (only for locations in parks and streets) using the function *kable()* from the *kableExtra* R package. (3pts)

```
tree_data_clean1 %>%
  filter(`Located in` %in% c("Park","Street")) %>%
  group_by(`Located in`) %>%
  summarise(Count = n()) %>%
  kable(caption = "Tree Count by Location") %>%
  kable_styling(latex_options = "hold_position")
```

Table 6: Tree Count by Location

Located in	Count
Park	2737
Street	4088

Question 10: How many trees are there in each of the Family groups in the data set *tree_data_clean1* (display the first 5 lines of the results in descending order)? (2pt)

```
tree_data_clean1 %>%
  group_by(Family) %>%
  summarise(`Number of trees` = n()) %>%
  arrange(-`Number of trees`) %>%
  head(5) %>%
  kable(caption = "Tree Count by Family") %>%
  kable_styling(latex_options = "hold_position")
```

Table 7: Tree Count by Family

Family	Number of trees
Myrtaceae	2102
Platanaceae	1512
Ulmaceae	1125
Fabaceae	327
Fagaceae	254

Question 11: Create a markdown table displaying the number of trees planted in each year (use variable *Yearplanted*) with common names Ironbark, Olive, Plum, Oak, and Elm (Hint: Use kable() from the gridExtra R package). What is the oldest most abundant tree in this group? (8pts)

Elm is the oldest most abundant tree in this group.

```
tree_data_clean1 %>%
  filter(`Common Name`
    %in% c("Ironbark", "Olive", "Plum", "Oak", "Elm")) %>%
  group_by(Yearplanted, `Common Name`) %>%
    summarise(`number of trees` = n()) %>%
    arrange(Yearplanted, desc(`number of trees`)) %>%
      knitr::kable(caption="Summary of trees in each year",booktabs = TRUE) %>%
  kable_styling(bootstrap_options = c("striped", "hover"), latex_options = "hold_position")
```

Table 8: Summary of trees in each year

Yearplanted	Common Name	number of trees
1900	Elm	179
1900	Ironbark	29
1900	Olive	17
1900	Oak	4
2000	Ironbark	23
2000	Elm	18
2000	Oak	9

Question 12: Select the trees with diameters (Diameter Breast Height) greater than 40 cm and smaller 100 cm and comment on where the trees are located (streets or parks). (max 25 words) (3pts)

We see that, for the diameters 41 to 56, there are more trees planted on the streets than in parks. Larger trees are prevalent more in parks and their number reduces with diameter.

```
large_trees_data <- tree_data_clean1 %>%
  filter(`Diameter Breast Height` %in% c(41:99)) %>%
  group_by(`Located in`, `Diameter Breast Height`) %>%
  summarise(`number of trees` = n()) %>%
  ungroup() %>%
  pivot_wider(names_from = `Located in`,
              values_from = `number of trees`)
```

Question 13: Plot the trees within the diameter range that you have selected in Question 12, which are located in parks and streets on a map using 2 different colours to differentiate their locations (streets or parks). (6pts)

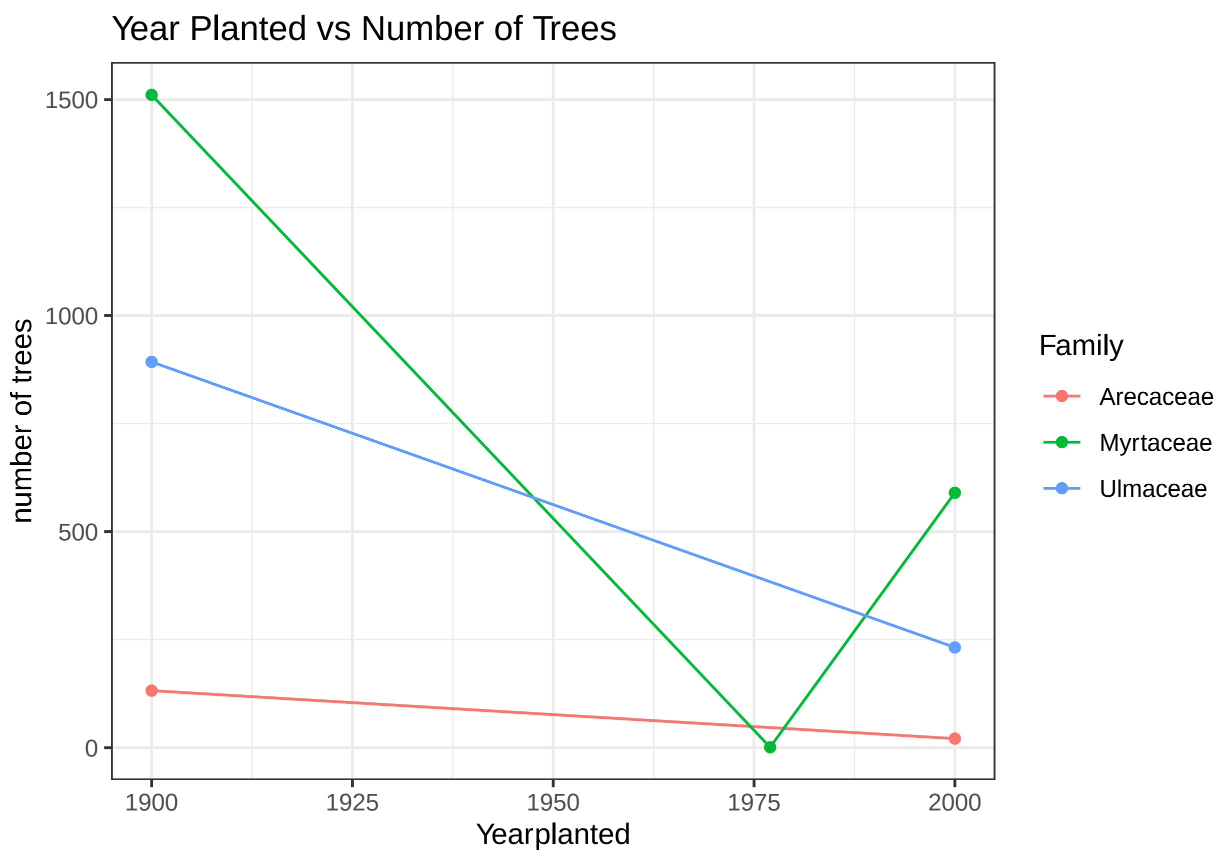
```
large_trees_data_parks <- tree_data_clean1 %>%
  filter(`Diameter Breast Height` > 40 &
           `Diameter Breast Height` < 100)
```

```
ggmap(melb_map) +
  geom_point(data = large_trees_data_parks ,
             aes(x = Longitude,
                 y = Latitude,
                 color = `Located in`)) +
  theme(legend.position = "bottom") +
  scale_color_brewer(palette = "Dark2") +
  labs(title = "Spatial Visualization of Large Trees")
```

Question 14: Create a time series plot (using geom_line) that displays the total number of trees planted per year in the data set *tree_data_clean1* that belong to the Families: Myrtaceae, Arecaceae, and Ulmaceae. What do you observe from the plot? (6pts)

We see that the number of trees that were planted decreases from 1900 to 2000. More trees belonging to Myrtaceae were planted with one tree uniquely planted in 1977.

```
Fig_data <- tree_data_clean1 %>%
  filter(`Family` %in% c("Myrtaceae", "Arecaceae", "Ulmaceae")) %>%
  group_by(`Yearplanted`, `Family`) %>%
  summarise(`number of trees` = n()) %>%
  arrange(desc(`number of trees`))

Fig_data %>%
  ggplot() +
  geom_line(mapping = aes(x = `Yearplanted`, y = `number of trees`, colour = `Family`)) +
  geom_point(mapping = aes(x = `Yearplanted`, y = `number of trees`, colour = `Family`))+
  theme(legend.position = "bottom") +
  theme_bw() +
  labs(title = "Year Planted vs Number of Trees")
```



Part 2: Simulation Exercise

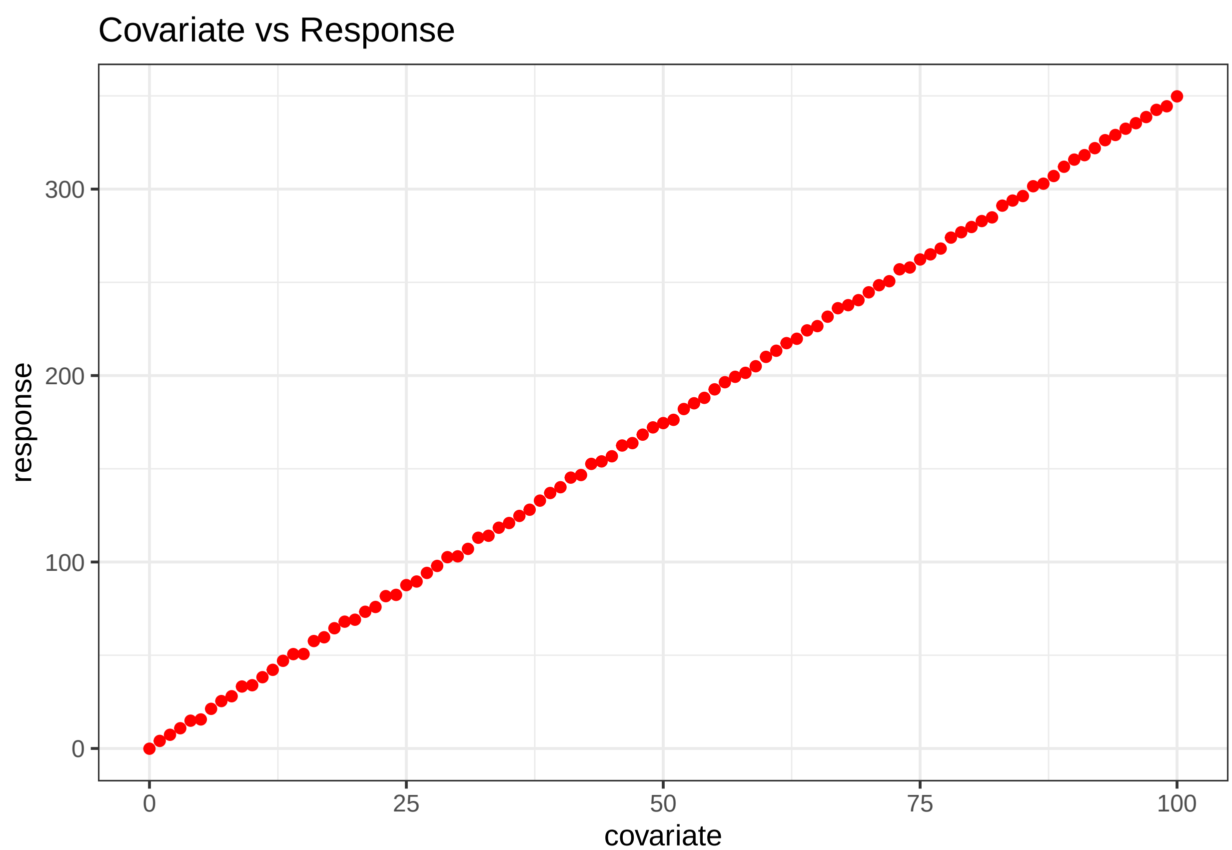
Question 15: Create a data frame called *simulation_data* that contains 2 variables with names *response* and *covariate*. Generate the variables according to the following model: $response = 3.5 \times covariate + epsilon$ **where *covariate* is a variable that takes values** $0, 1, 2, \ldots, 100$ **and** ϵ **is generated according to a Normal distribution (Hint: Use the function *rnorm()* to generate** $epsilon$**.) (3pts)**

```
set.seed(2021)

simulation_data <- tibble(covariate = 0:100) %>%
                        mutate(response = 3.5 * covariate + rnorm(101, 0, 1))
```

Question 16: Display graphically the relationship between the variables *response* and *covariate* (1pt) using a point plot. Which kind of relationship do you observe? (2pts)

We observe a linear relationship where the response variable increases with the covariate.

```
simulation_data %>%
  ggplot() +
  geom_point(mapping = aes(x = `covariate`,
                           y = `response`),
             colour = "red") +
  theme_bw() +
  labs(title = "Covariate vs Response")
```



Question 17: Fit a linear model between the variables *response* and *covariate* that you generate in Question 15 and display the model summary. (2pts)

```
simulation_data_lm <- lm(response~covariate, data=simulation_data)
summary(simulation_data_lm)
```

```
##
## Call:
## lm(formula = response ~ covariate, data = simulation_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.07431 -0.71466  0.05844  0.64196  2.25176
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.135896   0.199948    0.68    0.498
## covariate   3.493775   0.003455 1011.35   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.012 on 99 degrees of freedom
## Multiple R-squared:  0.9999, Adjusted R-squared:  0.9999
## F-statistic: 1.023e+06 on 1 and 99 DF,  p-value: < 2.2e-16
```

Question 18: What are the values for the intercept and the slope in the estimated model in Question 17 (Hint: Use the function *coef()*)? How do these values compare with the values in the simulation model? (max 50 words) (2pts)

```r
#coef(summary(simulation_data_lm))
slope_intercept <- tidy(summary(simulation_data_lm)) %>%
  select(term, estimate)
```

The generated model has a slope of 3.49 and an intercept of 0.14

The simulation data was generated from the equation, $response = 3.5 \times covariate + epsilon$ where epsilon is an error factor. The generated linear model is of the form $response = 3.4937754 \times covariate + 0.1358957$. The value 3.49 ~ 3.5 is the slope of the linear equation and the intercept of the model is 0.14. The fitted model differs from the simulation data in epsilon, which is centered around zero. The intercept of the model is close to zero.

```r
#coef(summary(simulation_data_lm))
slope_intercept %>%
  kable(caption = "Slope and Intercept")%>%
  kable_styling(latex_options = "hold_position")
```

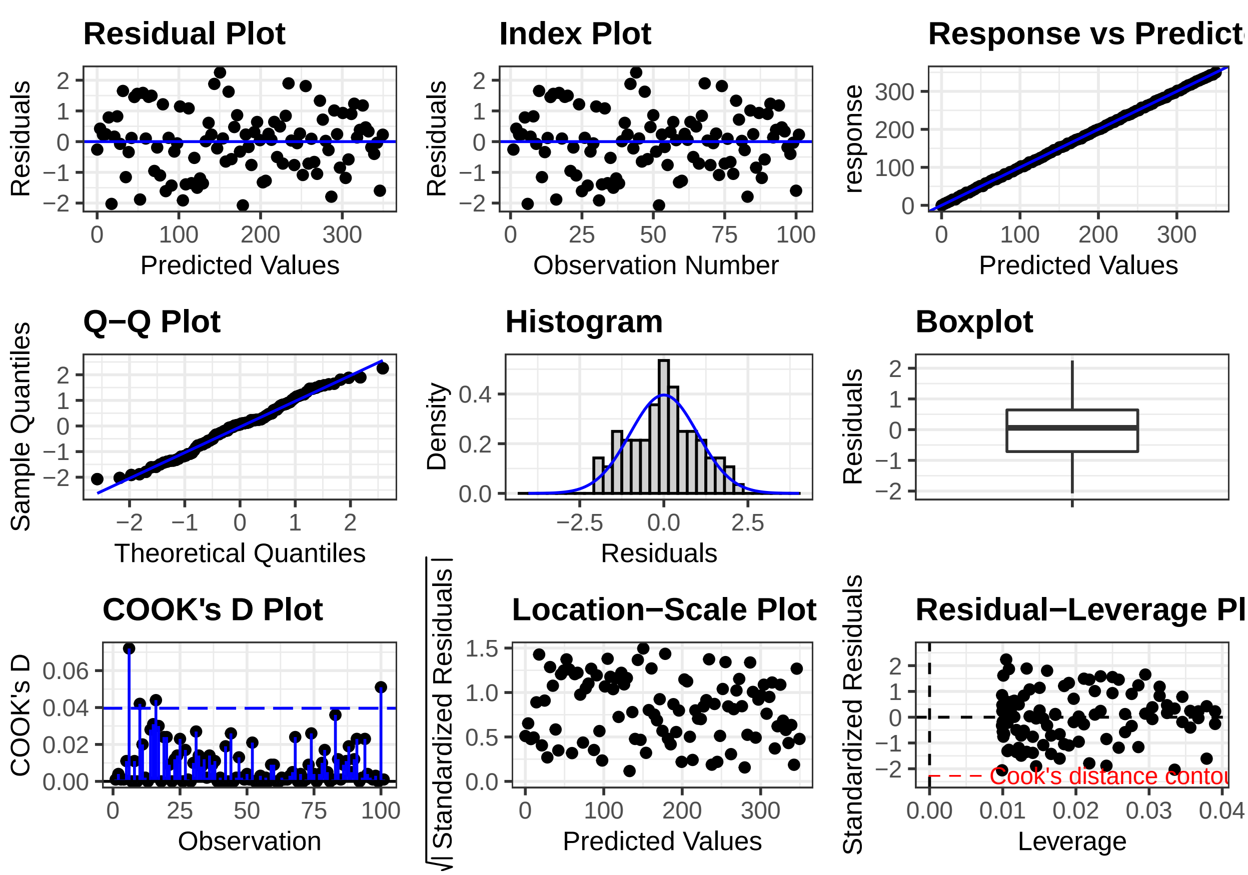
Table 9: Slope and Intercept

term	estimate
(Intercept)	0.1358957
covariate	3.4937754

Question 19: Create a figure to display the diagnostic plots of the linear model that you fit in Question 17. Comment on the diagnostic plots (max 50 words). Is this a good/bad model and why? (max 30 words) (4pts)

- The Residual plot is a scatter plot of predicted values vs residuals. Residual is the difference between actual values and the predicted values. For a good model, the residuals ~ 0. The residual plot for a model having randomly dispersed points suggests that the model is good.
- the Response vs Predicted plot is a scatter plot. A good model will have points aligned such that predicted values ~ response.
- The plots in the second row show the distribution of the residuals. A good model has a normal distribution of residuals centered around 0.

```
resid_panel(simulation_data_lm, plots = "all")
```



The plots below show the goodness of fit of the model representing the simulation data. The residual plot has points scattered indefinitely, the response vs predicted plot is a straight line(slope = 1, response ~ predicted), showing that it is a well fitted model. The residuals lie within (-1,1) with a median of 0 suggesting goodness of the model.

Question 20: Report R2, Radjusted, AIC, and BIC. Is this a good/bad model? Please explain your answer. (max 30 words) (2pts)

The model generated for the simulation data is a good model.

```
glance(simulation_data_lm) %>%
  select(r.squared, adj.r.squared, AIC, BIC) %>%
  kable(caption = "Measures of Goodness of Fit")%>%
  kable_styling(latex_options = "hold_position")
```

Table 10: Measures of Goodness of Fit

r.squared	adj.r.squared	AIC	BIC
0.9999032	0.9999022	293.0547	300.9001

The generated model has an R2 and Radjusted of 0.9999, and hence is a good model. The model with lowest AIC and BIC is a good model. For this model, the AIC and BIC are comparable and have low values.