DTP assignment 4

Ma Lok Sum, Zoe (a1819866)

2022-09-04

Executive Summary

In Melbourne, Australia, the water supply is overseen by the Melbourne Water Corporation (MWC). Previous predictions of evaporation rates at MWC's reservoirs are incorrect in light of current changes in Melbourne's climate. As a result, I write a report on evaporation to guarantee the security of the city's water supply. In order to help manage the Cardinia Reservoir in Melbourne's South East, I will examine the impacts of Melbourne's daily weather on evaporation in this report by creating a new linear model based on the previous financial year. This article also discusses the temporal and climatic variables, including lowest temperature and 9 a.m. humidity, that will have a major influence on the quantity of evaporation on January 13, 2020.

Methods

Get the data

```
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(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
```

```
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.8     v stringr 1.4.0
## v tidyr   1.2.0     v forcats 0.5.1
## v readr   2.1.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

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

```
library(readr)
melbourne <- read_csv("/Users/zoema/Downloads/DTP RStudio/melbourne.csv")
```

```
## New names:
## Rows: 300 Columns: 22
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (5): Date, Direction of maximum wind gust, 9am wind direction, 9am win... dbl
## (16): ...1, Minimum temperature (Deg C), Maximum Temperature (Deg C), R... time
## (1): Time of maximum wind gust
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

```
melbourne
```

```
## # A tibble: 300 x 22
##     ...1 Date    Minim~1 Maxim~2 Rainf~3 Evapo~4 Sunsh~5 Direc~6 Speed~7 Time ~8
##    <dbl> <chr>     <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <chr>     <dbl> <time>
##  1     1 2019-0~    18.4    22.2     0       7       7.5 SSW          39 15:23
##  2     2 2019-0~    15.9    29.5     0       6.6     9.3 SSW          26 14:53
##  3     3 2019-0~    14.6    22.1     1.4     6.4    13.3 SSW          33 11:12
##  4     4 2019-0~    17.1    23.1     0       9      11.1 SSW          39 16:20
##  5     5 2019-0~    16.7    24.1     0       7.2    10.7 SSW          43 15:36
##  6     6 2019-0~    16.1    20.5     0.6     7.4    12.5 SSE          37 13:02
##  7     7 2019-0~    13.5    21.4     0       8.2    11.2 SSW          31 14:21
##  8     8 2019-0~    17.7    24.7     0      10.2    11.2 SSW          44 15:06
##  9     9 2019-0~    18.7    32.3     0       9.6    13.2 SSW          31 15:40
## 10    10 2019-0~    19.4    30.4     0       8.6     7   SSW          31 13:48
## # ... with 290 more rows, 12 more variables: `9am Temperature (Deg C)` <dbl>,
## #   `9am relative humidity (%)` <dbl>, `9am cloud amount (oktas)` <dbl>,
## #   `9am wind direction` <chr>, `9am wind speed (km/h)` <chr>,
## #   `9am MSL pressure (hPa)` <dbl>, `3pm Temperature (Deg C)` <dbl>,
## #   `3pm relative humidity (%)` <dbl>, `3pm cloud amount (oktas)` <dbl>,
## #   `3pm wind direction` <chr>, `3pm wind speed (km/h)` <dbl>,
## #   `3pm MSL pressure (hPa)` <dbl>, and abbreviated variable names ...
## # i Use `print(n = ...)` to see more rows, and `colnames()` to see all variable names
```

Bivariate summaries

extract month in variable "Date" and create a new variable

```
month_melbourne <- str_match(melbourne$Date,"(\\d+)-(\\d+)-(\\d+)")
melbourne$Month <- month_melbourne[,3]
melbourne <- melbourne %>%
  relocate(Month, .after=Date)
```

```
melbourne$Month <- as.factor(melbourne$Month)
melbourne
```

```
## # A tibble: 300 x 23
##     ...1 Date       Month Minim~1 Maxim~2 Rainf~3 Evapo~4 Sunsh~5 Direc~6 Speed~7
##    <dbl> <chr>      <fct>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <chr>     <dbl>
##  1     1 2019-01-2 01       18.4    22.2     0       7       7.5 SSW          39
##  2     2 2019-01-3 01       15.9    29.5     0       6.6     9.3 SSW          26
##  3     3 2019-01-6 01       14.6    22.1     1.4     6.4    13.3 SSW          33
##  4     4 2019-01-7 01       17.1    23.1     0       9      11.1 SSW          39
##  5     5 2019-01-8 01       16.7    24.1     0       7.2    10.7 SSW          43
##  6     6 2019-01-9 01       16.1    20.5     0.6     7.4    12.5 SSE          37
##  7     7 2019-01-~ 01       13.5    21.4     0       8.2    11.2 SSW          31
##  8     8 2019-01-~ 01       17.7    24.7     0      10.2    11.2 SSW          44
##  9     9 2019-01-~ 01       18.7    32.3     0       9.6    13.2 SSW          31
## 10    10 2019-01-~ 01       19.4    30.4     0       8.6     7   SSW          31
## # ... with 290 more rows, 13 more variables:
## #   `Time of maximum wind gust` <time>, `9am Temperature (Deg C)` <dbl>,
## #   `9am relative humidity (%)` <dbl>, `9am cloud amount (oktas)` <dbl>,
## #   `9am wind direction` <chr>, `9am wind speed (km/h)` <chr>,
## #   `9am MSL pressure (hPa)` <dbl>, `3pm Temperature (Deg C)` <dbl>,
## #   `3pm relative humidity (%)` <dbl>, `3pm cloud amount (oktas)` <dbl>,
## #   `3pm wind direction` <chr>, `3pm wind speed (km/h)` <dbl>, ...
## # i Use `print(n = ...)` to see more rows, and `colnames()` to see all variable names
```

create a new variable to count the day of the week in "date"

```
melbourne$Wday <- wday(melbourne$Date, week_start=1)
melbourne <- melbourne %>%
  relocate(Wday, .after=Month)
melbourne$Wday <- as.factor(melbourne$Wday)
melbourne
```

```
## # A tibble: 300 x 24
##     ...1 Date       Month Wday  Minimu~1 Maxim~2 Rainf~3 Evapo~4 Sunsh~5 Direc~6
##    <dbl> <chr>      <fct> <fct>    <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <chr>
##  1     1 2019-01-2  01    3         18.4    22.2     0       7       7.5 SSW
##  2     2 2019-01-3  01    4         15.9    29.5     0       6.6     9.3 SSW
##  3     3 2019-01-6  01    7         14.6    22.1     1.4     6.4    13.3 SSW
##  4     4 2019-01-7  01    1         17.1    23.1     0       9      11.1 SSW
##  5     5 2019-01-8  01    2         16.7    24.1     0       7.2    10.7 SSW
##  6     6 2019-01-9  01    3         16.1    20.5     0.6     7.4    12.5 SSE
##  7     7 2019-01-10 01    4         13.5    21.4     0       8.2    11.2 SSW
##  8     8 2019-01-12 01    6         17.7    24.7     0      10.2    11.2 SSW
##  9     9 2019-01-14 01    1         18.7    32.3     0       9.6    13.2 SSW
## 10    10 2019-01-15 01    2         19.4    30.4     0       8.6     7   SSW
## # ... with 290 more rows, 14 more variables:
## #   `Speed of maximum wind gust (km/h)` <dbl>,
## #   `Time of maximum wind gust` <time>, `9am Temperature (Deg C)` <dbl>,
## #   `9am relative humidity (%)` <dbl>, `9am cloud amount (oktas)` <dbl>,
## #   `9am wind direction` <chr>, `9am wind speed (km/h)` <chr>,
## #   `9am MSL pressure (hPa)` <dbl>, `3pm Temperature (Deg C)` <dbl>,
## #   `3pm relative humidity (%)` <dbl>, `3pm cloud amount (oktas)` <dbl>, ...
## # i Use `print(n = ...)` to see more rows, and `colnames()` to see all variable names
```

tame the data for analysis: recode existing variables

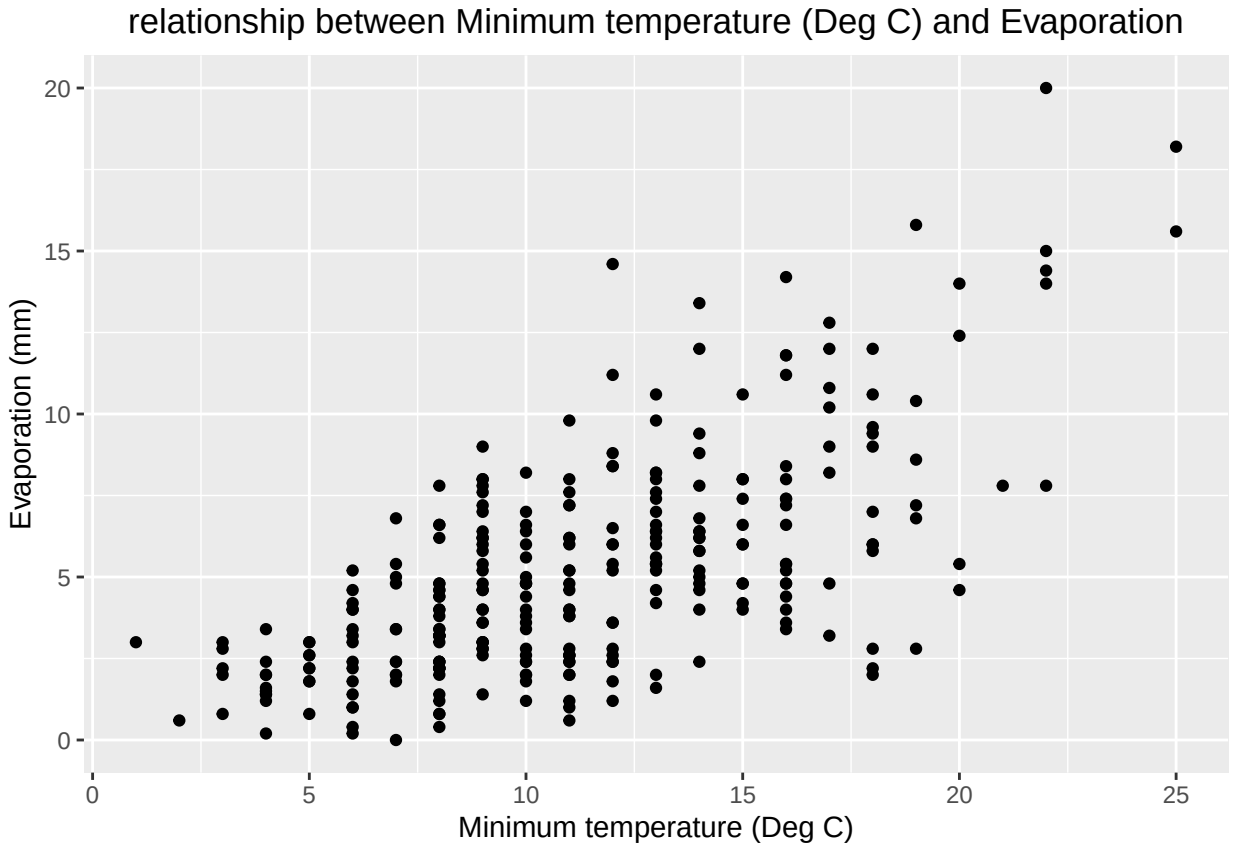
```
melbourne$`Minimum temperature (Deg C)` <- as.integer(melbourne$`Minimum temperature (Deg C)`)
melbourne$`Maximum Temperature (Deg C)`<- as.integer(melbourne$`Maximum Temperature (Deg C)`)
melbourne$`9am relative humidity (%)`<- as.integer(melbourne$`9am relative humidity (%)`)
melbourne
```

```
## # A tibble: 300 x 24
##     ...1 Date       Month Wday  Minimu~1 Maxim~2 Rainf~3 Evapo~4 Sunsh~5 Direc~6
##    <dbl> <chr>      <fct> <fct>    <int>   <int>   <dbl>   <dbl>   <dbl> <chr>
##  1     1 2019-01-2  01    3           18      22     0       7       7.5 SSW
##  2     2 2019-01-3  01    4           15      29     0       6.6     9.3 SSW
##  3     3 2019-01-6  01    7           14      22     1.4     6.4    13.3 SSW
##  4     4 2019-01-7  01    1           17      23     0       9      11.1 SSW
##  5     5 2019-01-8  01    2           16      24     0       7.2    10.7 SSW
##  6     6 2019-01-9  01    3           16      20     0.6     7.4    12.5 SSE
##  7     7 2019-01-10 01    4           13      21     0       8.2    11.2 SSW
##  8     8 2019-01-12 01    6           17      24     0      10.2    11.2 SSW
##  9     9 2019-01-14 01    1           18      32     0       9.6    13.2 SSW
## 10    10 2019-01-15 01    2           19      30     0       8.6     7   SSW
## # ... with 290 more rows, 14 more variables:
## #   `Speed of maximum wind gust (km/h)` <dbl>,
## #   `Time of maximum wind gust` <time>, `9am Temperature (Deg C)` <dbl>,
## #   `9am relative humidity (%)` <int>, `9am cloud amount (oktas)` <dbl>,
## #   `9am wind direction` <chr>, `9am wind speed (km/h)` <chr>,
## #   `9am MSL pressure (hPa)` <dbl>, `3pm Temperature (Deg C)` <dbl>,
## #   `3pm relative humidity (%)` <dbl>, `3pm cloud amount (oktas)` <dbl>, ...
## # i Use `print(n = ...)` to see more rows, and `colnames()` to see all variable names
```

plot graph and analysis it

```
library(ggplot2)
ggplot(melbourne, aes(x = `Minimum temperature (Deg C)`, y = `Evaporation (mm)`)) + geom_point() +
  ggtitle("relationship between Minimum temperature (Deg C) and Evaporation") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 8 rows containing missing values (geom_point).
```



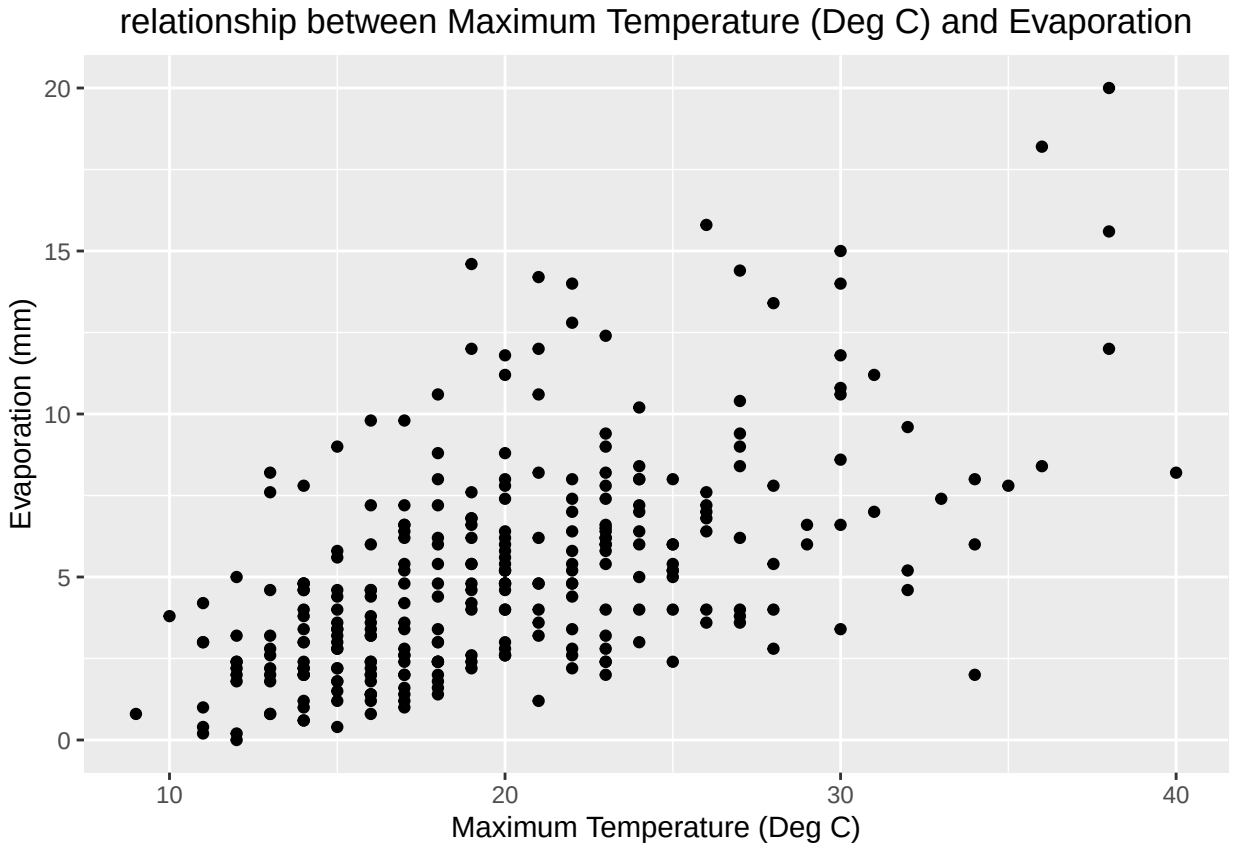

```
cor(melbourne$`Minimum temperature (Deg C)`, melbourne$`Evaporation (mm)`, use = "complete.obs")
```

```
## [1] 0.6621249
```

After reviewing the correlation and the plot, it is a strong positive linear relationship between Minimum temperature (Deg C) and Evaporation with potential outliers.

```
ggplot(melbourne, aes(x = `Maximum Temperature (Deg C)`, y = `Evaporation (mm)`)) + geom_point() +
  ggtitle("relationship between Maximum Temperature (Deg C) and Evaporation") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 8 rows containing missing values (geom_point).
```



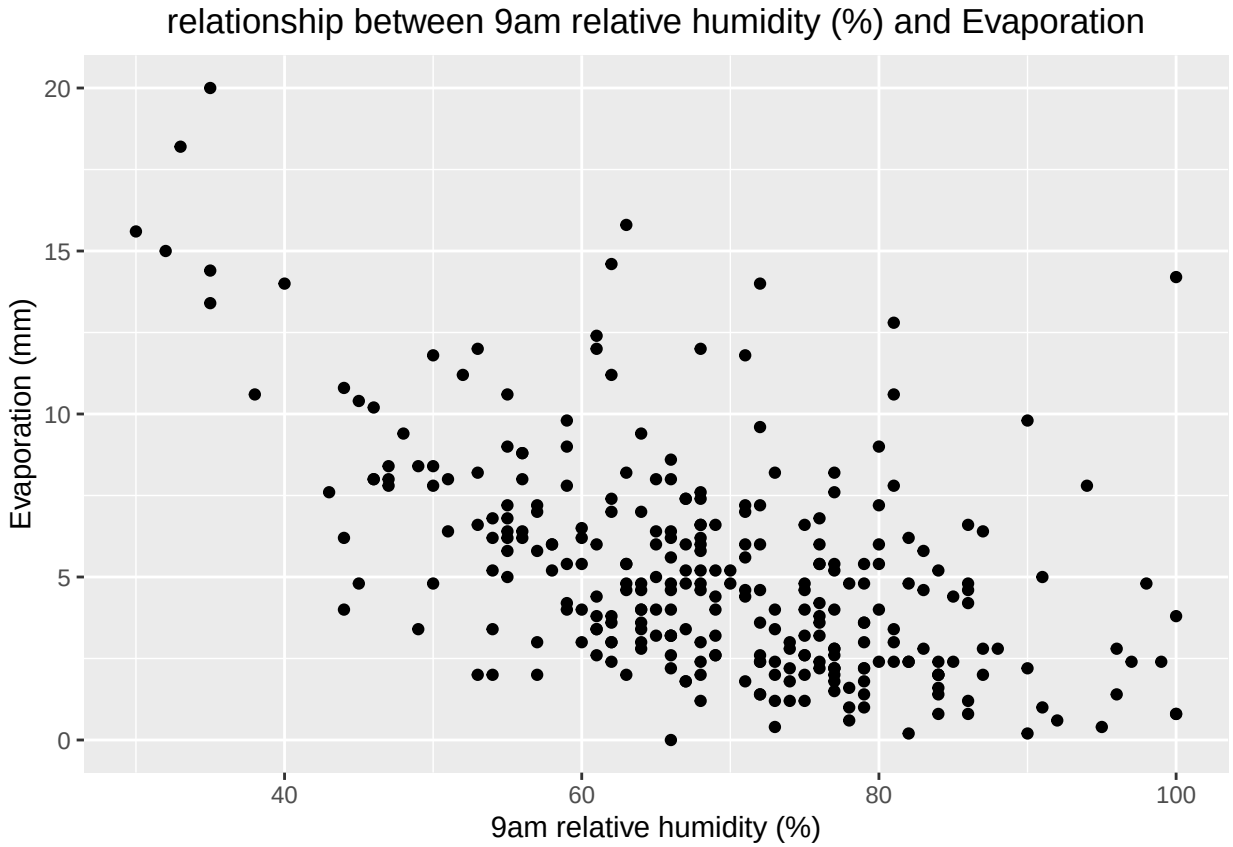

```
cor(melbourne$`Maximum Temperature (Deg C)`, melbourne$`Evaporation (mm)`, use = "complete.obs")
```

```
## [1] 0.5748685
```

After reviewing the correlation and the plot, it is a moderate positive linear relationship between Maximum Temperature (Deg C) and Evaporation with potential outliers.

```
ggplot(melbourne, aes(x = `9am relative humidity (%)`, y = `Evaporation (mm)`)) + geom_point() +
  ggtitle("relationship between 9am relative humidity (%) and Evaporation") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 8 rows containing missing values (geom_point).
```



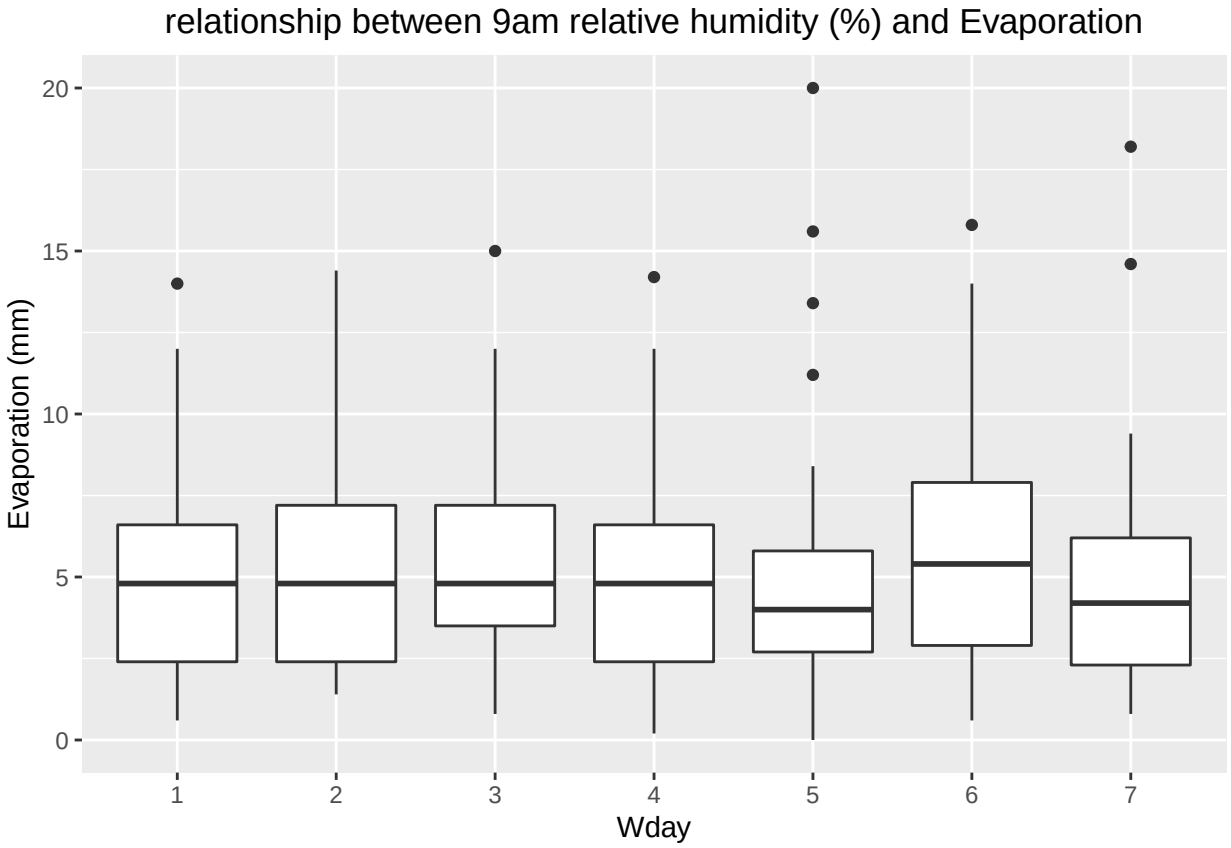

```
cor(melbourne$`9am relative humidity (%)`, melbourne$`Evaporation (mm)`, use = "complete.obs")
```

```
## [1] -0.5272612
```

After reviewing the correlation and the plot, It is a moderate negative linear relationship between 9am relative humidity (%) and Evaporation with potential outliers.

```
ggplot(melbourne, aes(x = Wday, y = `Evaporation (mm)`)) + geom_boxplot() +
  ggtitle("relationship between 9am relative humidity (%) and Evaporation") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 8 rows containing non-finite values (stat_boxplot).
```



Shape: Monday, Tuesday and Saturday have the normal distribution. Wednesday, Sunday and Friday have the positive skew. Thursday has the negative skew.

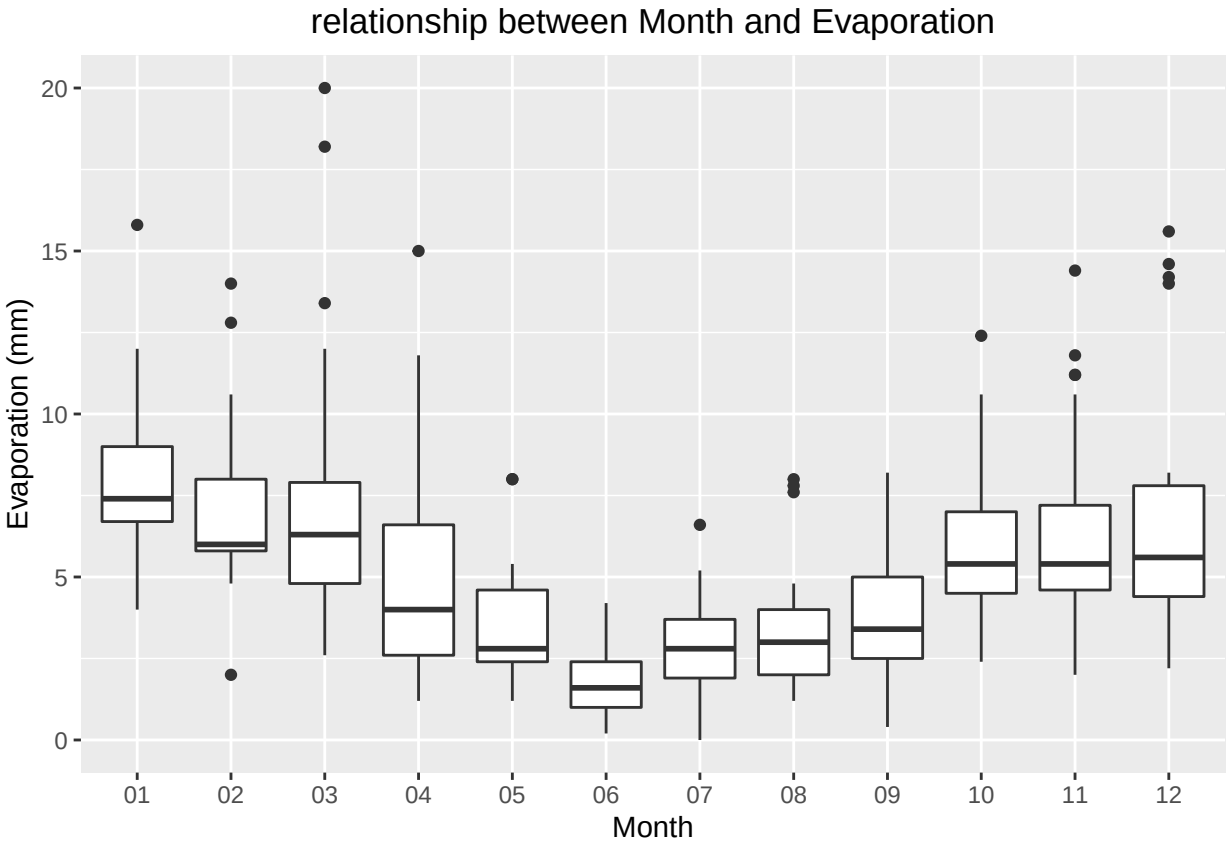
Location: Friday has the smallest evaporation and Saturday has the largest evaporation.

Spread: Friday has the smallest and simply Tuesday and Saturday have the largest.

Outliers: There are potential outliers except for Tuesday.

```
ggplot(melbourne, aes(x = Month, y = `Evaporation (mm)`)) + geom_boxplot() +
  ggtitle("relationship between Month and Evaporation") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 8 rows containing non-finite values (stat_boxplot).
```



Shape: All of the month are positive skew, except for June, July and August.

Location: June has the lowest evaporation. January has the largest evaporation.

Spread: June has the smallest spread. April has the largest spread.

Outliers: There are potential outliers except for June.

Model selection

1. Fit a model containing all the possible predictors

```
# fit a model
lm_melbourne <- lm(`Evaporation (mm)`~ Month+ Wday+`9am relative humidity (%)`
                   +`Minimum temperature (Deg C)`+`Maximum Temperature (Deg C)`+ Month:`9am relative hu
summary(lm_melbourne)
```

```
##
## Call:
## lm(formula = `Evaporation (mm)` ~ Month + Wday + `9am relative humidity (%)` +
##     `Minimum temperature (Deg C)` + `Maximum Temperature (Deg C)` +
##     Month:`9am relative humidity (%)`, data = melbourne)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7769 -1.0919 -0.0836  0.9587  9.8859
##
## Coefficients:
```

```
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         8.24248    3.22269   2.558   0.0111 *
## Month02                            -2.74451    4.35621  -0.630   0.5292
## Month03                             4.87489    3.63275   1.342   0.1808
## Month04                             2.10179    3.92136   0.536   0.5924
## Month05                            -3.72634    4.04473  -0.921   0.3578
## Month06                            -8.42168    4.90870  -1.716   0.0874 .
## Month07                            -3.38874    4.25006  -0.797   0.4260
## Month08                            -8.47262    4.31272  -1.965   0.0505 .
## Month09                            -2.89014    4.42856  -0.653   0.5146
## Month10                            -6.10370    3.93639  -1.551   0.1222
## Month11                             0.99812    3.72575   0.268   0.7890
## Month12                            -1.04577    3.85822  -0.271   0.7866
## Wday2                              -0.17700    0.48423  -0.366   0.7150
## Wday3                              -0.27072    0.48504  -0.558   0.5772
## Wday4                              -0.36068    0.48737  -0.740   0.4599
## Wday5                              -0.41668    0.49370  -0.844   0.3994
## Wday6                               0.53288    0.49819   1.070   0.2858
## Wday7                              -0.11817    0.48290  -0.245   0.8069
## `9am relative humidity (%)`        -0.10258    0.04793  -2.140   0.0333 *
## `Minimum temperature (Deg C)`       0.35361    0.04697   7.528 8.48e-13 ***
## `Maximum Temperature (Deg C)`       0.02419    0.03604   0.671   0.5027
## Month02:`9am relative humidity (%)`  0.03658    0.06576   0.556   0.5785
## Month03:`9am relative humidity (%)` -0.06368    0.05471  -1.164   0.2455
## Month04:`9am relative humidity (%)` -0.04016    0.05967  -0.673   0.5015
## Month05:`9am relative humidity (%)`  0.03505    0.05945   0.590   0.5560
## Month06:`9am relative humidity (%)`  0.09058    0.06713   1.349   0.1784
## Month07:`9am relative humidity (%)`  0.03768    0.06237   0.604   0.5463
## Month08:`9am relative humidity (%)`  0.11603    0.06402   1.812   0.0711 .
## Month09:`9am relative humidity (%)`  0.03542    0.06831   0.519   0.6045
## Month10:`9am relative humidity (%)`  0.09878    0.06002   1.646   0.1010
## Month11:`9am relative humidity (%)` -0.01318    0.05684  -0.232   0.8168
## Month12:`9am relative humidity (%)`  0.01570    0.05743   0.273   0.7848
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.118 on 260 degrees of freedom
##   (8 observations deleted due to missingness)
## Multiple R-squared:  0.6496, Adjusted R-squared:  0.6078
## F-statistic: 15.55 on 31 and 260 DF,  p-value: < 2.2e-16
```

After fitting a model including all the possible predictors in bivariable summaries, I try to determine the p-value using the linear model summary to determine the quantitative variable, which the p-value in Maximum Temperature variable is greater than 0.05 and highest.

```
# viewing the categorical variables
anova(lm_melbourne)
```

```
## Analysis of Variance Table
##
## Response: Evaporation (mm)
##                                  Df  Sum Sq Mean Sq  F value    Pr(>F)
## Month                            11 1145.08  104.10  23.2084 < 2.2e-16 ***
## Wday                              6   32.28    5.38   1.1994  0.306930
```

```
## `9am relative humidity (%)`         1  520.41  520.41 116.0236 < 2.2e-16 ***
## `Minimum temperature (Deg C)`       1  325.14  325.14  72.4891 1.363e-15 ***
## `Maximum Temperature (Deg C)`       1    3.04    3.04   0.6776  0.411174
## Month:`9am relative humidity (%)`  11  135.78   12.34   2.7519  0.002172 **
## Residuals                         260 1166.20    4.49
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Besides, I try to determine the p-value using an ANOVA for the categorical variables. I found that the p-value the week day variables is greater than 0.05 and highest. Therefore, I decide to remove the temperature and week day variable in order to get all the remaining predictors are significang enough.

```
# updated model: remove maximum temperature + wday
lm_melbourne_2 <- lm(`Evaporation (mm)`~ Month +`9am relative humidity (%)`
                     +`Minimum temperature (Deg C)`+ Month:`9am relative humidity (%)`, data = melbourne
summary(lm_melbourne_2)
```

```
##
## Call:
## lm(formula = `Evaporation (mm)` ~ Month + `9am relative humidity (%)` +
##     `Minimum temperature (Deg C)` + Month:`9am relative humidity (%)`,
##     data = melbourne)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.1628 -1.1240 -0.0840  0.9735  9.8099
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          8.729969   3.150256   2.771  0.00598 **
## Month02                             -3.125820   4.287519  -0.729  0.46661
## Month03                              4.573783   3.514136   1.302  0.19420
## Month04                              2.158819   3.791596   0.569  0.56958
## Month05                             -3.926593   3.954220  -0.993  0.32160
## Month06                             -9.201345   4.798432  -1.918  0.05623 .
## Month07                             -3.961410   4.173947  -0.949  0.34344
## Month08                             -9.524645   4.229461  -2.252  0.02514 *
## Month09                             -3.674975   4.295261  -0.856  0.39299
## Month10                             -5.750042   3.817172  -1.506  0.13316
## Month11                              1.066893   3.614761   0.295  0.76811
## Month12                             -0.625674   3.678827  -0.170  0.86508
## `9am relative humidity (%)`         -0.104636   0.046526  -2.249  0.02533 *
## `Minimum temperature (Deg C)`        0.364171   0.044091   8.260 6.85e-15 ***
## Month02:`9am relative humidity (%)`  0.041918   0.064584   0.649  0.51687
## Month03:`9am relative humidity (%)` -0.060534   0.052586  -1.151  0.25070
## Month04:`9am relative humidity (%)` -0.042060   0.057343  -0.733  0.46390
## Month05:`9am relative humidity (%)`  0.034871   0.057471   0.607  0.54453
## Month06:`9am relative humidity (%)`  0.098421   0.065039   1.513  0.13139
## Month07:`9am relative humidity (%)`  0.043274   0.060713   0.713  0.47662
## Month08:`9am relative humidity (%)`  0.128005   0.062235   2.057  0.04068 *
## Month09:`9am relative humidity (%)`  0.045716   0.065407   0.699  0.48519
## Month10:`9am relative humidity (%)`  0.091762   0.057723   1.590  0.11309
## Month11:`9am relative humidity (%)` -0.015520   0.054568  -0.284  0.77632
```

```
## Month12:`9am relative humidity (%)`  0.009228   0.054468   0.169  0.86560
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.112 on 267 degrees of freedom
##   (8 observations deleted due to missingness)
## Multiple R-squared:  0.6422, Adjusted R-squared:   0.61
## F-statistic: 19.97 on 24 and 267 DF,  p-value: < 2.2e-16
```

All the quantitative variables are significant enough in the updated model.

```
# viewing the categorical variables using an ANOVA
anova(lm_melbourne_2)
```

```
## Analysis of Variance Table
##
## Response: Evaporation (mm)
##                                   Df  Sum Sq Mean Sq  F value    Pr(>F)
## Month                             11 1145.08  104.10  23.3419 < 2.2e-16 ***
## `9am relative humidity (%)`        1  518.42  518.42 116.2449 < 2.2e-16 ***
## `Minimum temperature (Deg C)`      1  331.24  331.24  74.2732  6.08e-16 ***
## Month:`9am relative humidity (%)`  11  142.44   12.95   2.9035  0.001243 **
## Residuals                         267 1190.75    4.46
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

All the categorical variables are statistically significant enough in the updated model. Therefore, all the predictors in the updated model are significant.

results

In my final model, all the predictors are significant, which their p-value must 0.05 or lower. These term is differ compared to bivariate analyses because in the bivariate analyses include all the variables that we may think that it is useful for our prediction. However, some of the variables (Maximum temperature and week day) may not significant enough. It indicates there is insufficient evidence in our linear model to conclude that a non-zero correlation exists. Hence, removing those unsignificant variables can help us to use the most relevant variables to conduct the prediction. Otherwise, the linear model cannot predict the outcomes more accruately.

Model interpretation

```
# summary of the updated linear model
summary(lm_melbourne_2)
```

```
##
## Call:
## lm(formula = `Evaporation (mm)` ~ Month + `9am relative humidity (%)` +
##     `Minimum temperature (Deg C)` + Month:`9am relative humidity (%)`,
##     data = melbourne)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.1628 -1.1240 -0.0840  0.9735  9.8099
```

```
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          8.729969   3.150256   2.771  0.00598 **
## Month02                             -3.125820   4.287519  -0.729  0.46661
## Month03                              4.573783   3.514136   1.302  0.19420
## Month04                              2.158819   3.791596   0.569  0.56958
## Month05                             -3.926593   3.954220  -0.993  0.32160
## Month06                             -9.201345   4.798432  -1.918  0.05623 .
## Month07                             -3.961410   4.173947  -0.949  0.34344
## Month08                             -9.524645   4.229461  -2.252  0.02514 *
## Month09                             -3.674975   4.295261  -0.856  0.39299
## Month10                             -5.750042   3.817172  -1.506  0.13316
## Month11                              1.066893   3.614761   0.295  0.76811
## Month12                             -0.625674   3.678827  -0.170  0.86508
## `9am relative humidity (%)`         -0.104636   0.046526  -2.249  0.02533 *
## `Minimum temperature (Deg C)`        0.364171   0.044091   8.260 6.85e-15 ***
## Month02:`9am relative humidity (%)`  0.041918   0.064584   0.649  0.51687
## Month03:`9am relative humidity (%)` -0.060534   0.052586  -1.151  0.25070
## Month04:`9am relative humidity (%)` -0.042060   0.057343  -0.733  0.46390
## Month05:`9am relative humidity (%)`  0.034871   0.057471   0.607  0.54453
## Month06:`9am relative humidity (%)`  0.098421   0.065039   1.513  0.13139
## Month07:`9am relative humidity (%)`  0.043274   0.060713   0.713  0.47662
## Month08:`9am relative humidity (%)`  0.128005   0.062235   2.057  0.04068 *
## Month09:`9am relative humidity (%)`  0.045716   0.065407   0.699  0.48519
## Month10:`9am relative humidity (%)`  0.091762   0.057723   1.590  0.11309
## Month11:`9am relative humidity (%)` -0.015520   0.054568  -0.284  0.77632
## Month12:`9am relative humidity (%)`  0.009228   0.054468   0.169  0.86560
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.112 on 267 degrees of freedom
##   (8 observations deleted due to missingness)
## Multiple R-squared:  0.6422, Adjusted R-squared:   0.61
## F-statistic: 19.97 on 24 and 267 DF,  p-value: < 2.2e-16
```

After reviewing the linear model above, I have this model interpretation.

In the model selection part, we remove the variables of wday and maxmimium temperature in order to get the relationship of the linear model is significant enough. Therefore, we do not have the week day and maximum temperature coefficients in the linear model.

Since the variable of month is categorical predictors, I do not explain it. But, for the operations of month, it is calculated for each months, which the evaporation is affected by each month.

The intercept is 8.730.

Interpretation of intercept:The value of the intercept term of evaporation is 8.730mm when all the predictor variables are zero.

The coefficients of the 9am relative humidity (%) is -0.105.

Interpretation of coefficients of the 9am relative humidity (%): If the evaporation increase by 1mm, the 9am relative humidity (%) is expected to decrease by -0.105/ -10.5%, assuming all other predictor variables are held constant.

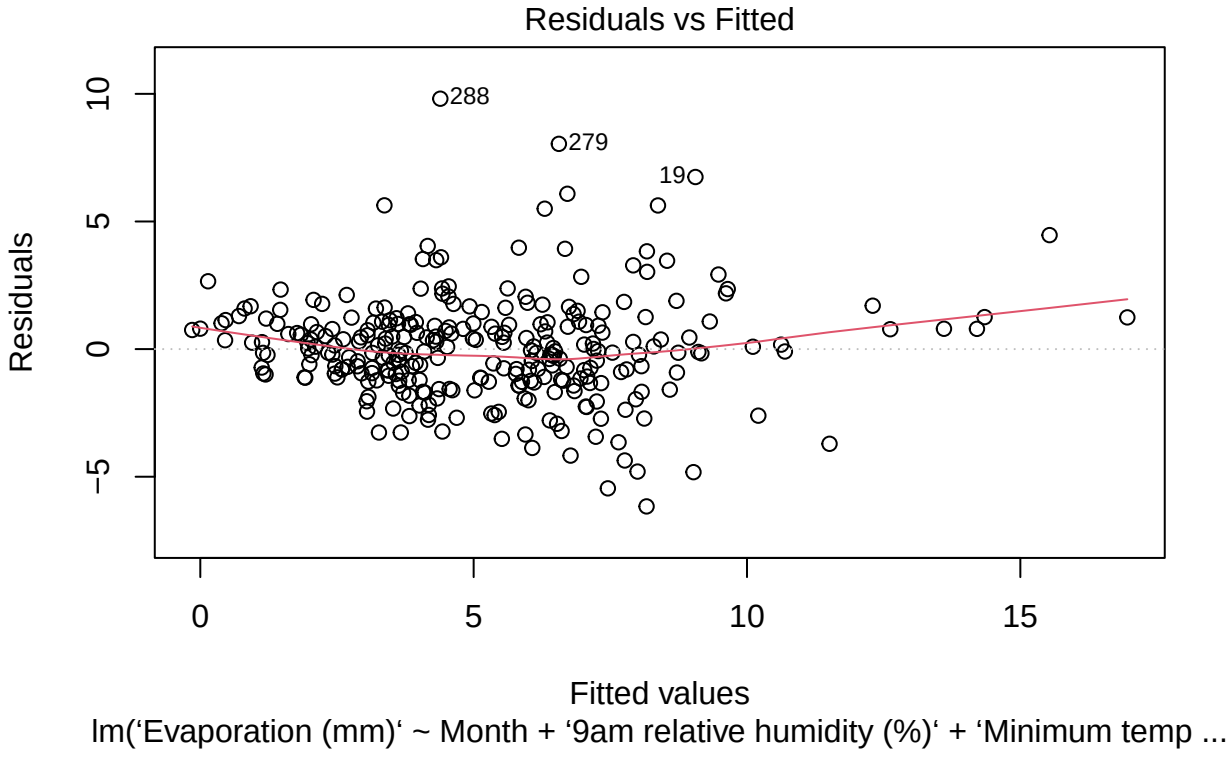
The coefficients of the Minimum temperature (Deg C) is 0.364.

Interpretation of coefficients of the Minimum temperature (Deg C): If the evaporation increase by 1mm, the Minimum temperature (Deg C) is expected to increase by 0.364 (Deg C), assuming all other predictor variables are held constant.

Model diagnostics: check the assumptions for the linear model

linearity

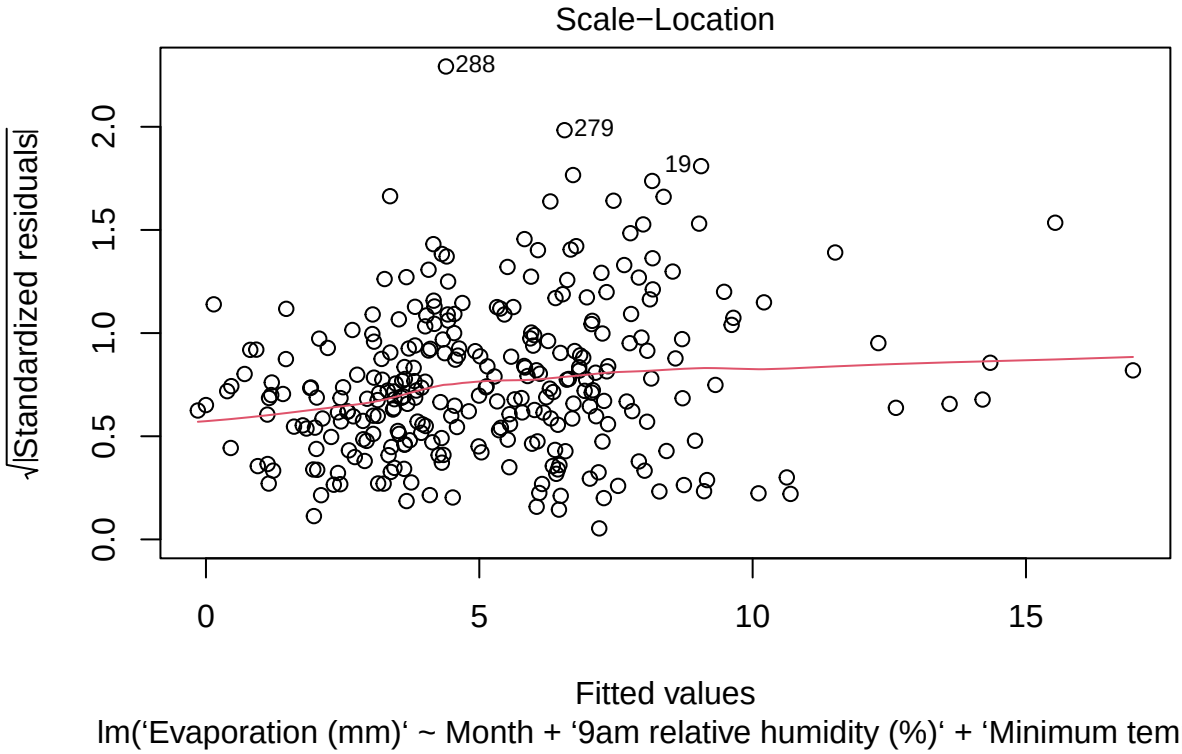
```
# linearity
plot(lm_melbourne_2, which = 1)
```



The linearity assumption is satisfied since it is almost no change in trend comparing with the residuals and fitted plot.

homoscedasticity

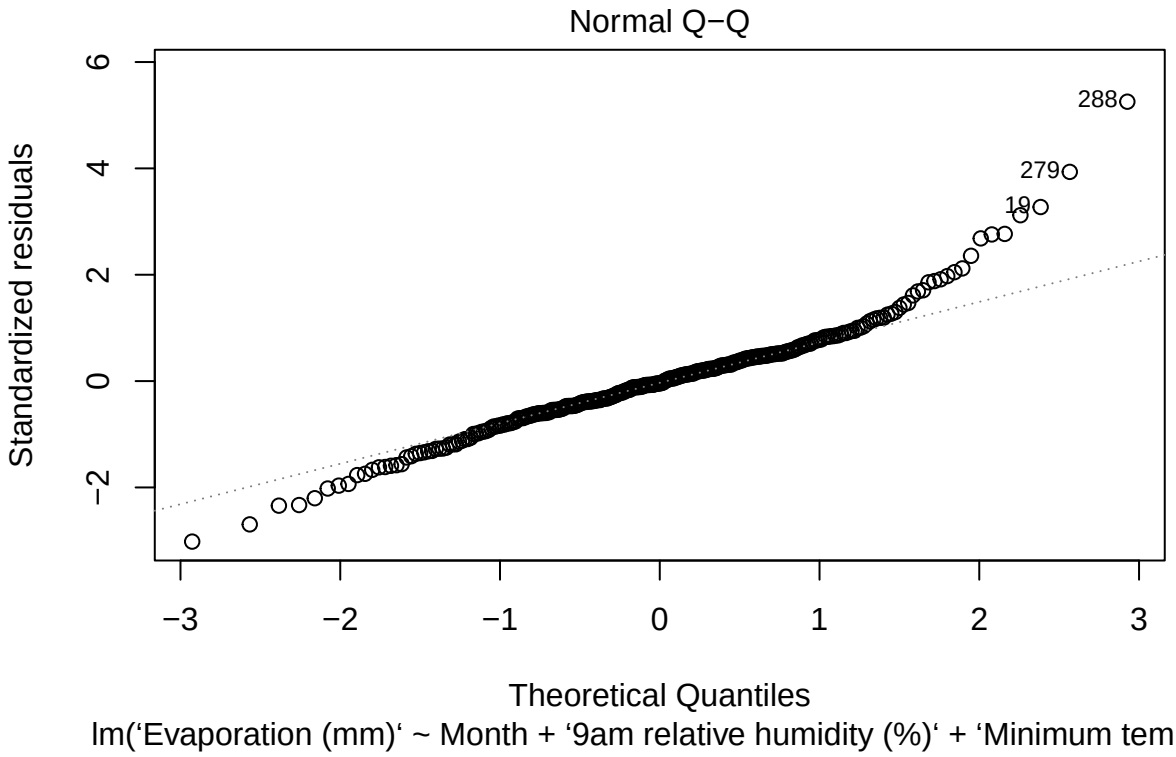
```
# homoscedasticity
plot(lm_melbourne_2, which = 3)
```



The homoscedasticity is satisfied since the red line is roughly horizontal across the plot.

normality

```
# normality
plot(lm_melbourne_2, which = 2)
```



The normality assumption is satisfied, which is mostly follow the trend line despite of three outliers.

Independence

The independence assumption is not satisfied because if one of the variables, such as the minimum temperature, date and humidity, are changed, it must affect the response variable.

Prediction

Since we remove the week day variable in the model selection for chasing a significant model, we only need month as a categorical predictor in our linear model to predict the value.

```
# first scenarios
new_data_1 <- tibble(date = "29-02-2020", `Minimum temperature (Deg C)` = 13.8,
                     `Maximum Temperature (Deg C)` =  23.2,`9am relative humidity (%)` =0.74)
month_melbourne_2 <- str_match(new_data_1$date,"(\\d+)-(\\d+)-(\\d+)")
new_data_1$Month <- month_melbourne_2[, 3]
prediction_1 <- round(predict(lm_melbourne_2, newdata = new_data_1, interval = "prediction"),3)
prediction_1
```

```
##      fit   lwr    upr
## 1 10.583 3.453 17.713
```

The interval is (3.453, 17.713), in which the lower bound of evaporation means 3.453, and the upper bound of evaporation means 17.713.

```
# second scenarios
new_data_2 <- tibble(date = "25-12-2020", `Minimum temperature (Deg C)` = 16.4,
                     `Maximum Temperature (Deg C)` =  31.9,`9am relative humidity (%)` =0.57)
```

```
month_melbourne_3 <- str_match(new_data_1$date,"(\\d+)-(\\d+)-(\\d+)")
new_data_2$Month <- month_melbourne_3[, 3]
prediction_2 <- round(predict(lm_melbourne_2, newdata = new_data_2, interval = "prediction"),3)
prediction_2
```

```
##      fit   lwr    upr
## 1 11.541 4.376 18.706
```

The interval is (4.376, 18.706), in which the lower bound of evaporation means 4.376, and the upper bound of evaporation means 18.706.

```
# third scenarios
new_data_3 <- tibble(date = "13-01-2020", `Minimum temperature (Deg C)` = 26.5,
                     `Maximum Temperature (Deg C)` =  44.3,`9am relative humidity (%)` =0.35)
month_melbourne_4 <- str_match(new_data_1$date,"(\\d+)-(\\d+)-(\\d+)")
new_data_3$Month <- month_melbourne_4[, 3]
prediction_3 <- round(predict(lm_melbourne_2, newdata = new_data_3, interval = "prediction"),3)
prediction_3
```

```
##      fit   lwr    upr
## 1 15.233 7.897 22.568
```

The intervals is (7.897, 22.568), in which the lower bound of evaporation means 7.897, and the upper bound of evaporation means 22.568.

```
# forth scenarios
new_data_4 <- tibble(date = "06-07-2020", `Minimum temperature (Deg C)` = 6.8,
                     `Maximum Temperature (Deg C)` =  10.6,`9am relative humidity (%)` =0.76)
month_melbourne_5 <- str_match(new_data_1$date,"(\\d+)-(\\d+)-(\\d+)")
new_data_4$Month <- month_melbourne_5[, 3]
prediction_4 <- round(predict(lm_melbourne_2, newdata = new_data_4, interval = "prediction"),3)
prediction_4
```

```
##     fit  lwr    upr
## 1 8.033 0.93 15.135
```

The intervals is (0.930, 15.135), in which the lower bound of evaporation means 0.930, and the upper bound of evaporation means 15.135.

Table for the intervals

```
library(expss)
```

```
## Loading required package: maditr
```

```
##
## To select rows from data: rows(mtcars, am==0)
```

```
##
## Attaching package: 'maditr'
```

```
## The following object is masked from 'package:purrr':
##
##     transpose
```

```
## The following object is masked from 'package:readr':
```

```
##
##     cols

## The following objects are masked from 'package:dplyr':
##
##     between, coalesce, first, last

##
## Use 'expss_output_viewer()' to display tables in the RStudio Viewer.
##  To return to the console output, use 'expss_output_default()'.

##
## Attaching package: 'expss'

## The following objects are masked from 'package:stringr':
##
##     fixed, regex

## The following objects are masked from 'package:purrr':
##
##     keep, modify, modify_if, when

## The following objects are masked from 'package:tidyr':
##
##     contains, nest

## The following object is masked from 'package:ggplot2':
##
##     vars

## The following objects are masked from 'package:dplyr':
##
##     compute, contains, na_if, recode, vars
```

```
prediction <- matrix(c(10.583, 3.453, 17.713, 11.540, 4.376, 18.706,
                       15.322, 7.897, 22.568, 8.0328, 0.930, 15.135),ncol=3,byrow=TRUE)
colnames(prediction) <- c("Predicted value","Lower bound","Upper bound")
rownames(prediction) <- c("First scenarios","Second scenarios"
                          ,"Third scenarios", "Forth scenarios")
prediction <- as.table(prediction)
prediction%>%knitr::kable(digits = 3, format.args = list(big.mark = ","),
                          caption ="Table for the intervals for making forecasts
                          on these particular days")
```

Table 1: Table for the intervals for making forecasts on these particular days

	Predicted value	Lower bound	Upper bound
First scenarios	10.583	3.453	17.713
Second scenarios	11.540	4.376	18.706
Third scenarios	15.322	7.897	22.568
Forth scenarios	8.033	0.930	15.135

```
new_data1 <- predict(lm_melbourne_2, newdata = new_data_1, interval = "prediction")
new_data1 <- tibble(
  `Minimum temperature (Deg C)` = c(13.8,16.4, 26.5, 6.8),
  `Maximum Temperature (Deg C)` = c(23.2, 31.9, 44.3, 10.6),
```

```
  `9am relative humidity (%)` = c(74,57,35,76),
  Month = c("02", "12","01", "07"),
  date = c( "29-02-2020", "25-12-2020" ,"13-01-2020" ,"06-07-2020"))

scenarios <- tibble(
  date = c( "29-02-2020", "25-12-2020" ,"13-01-2020" ,"06-07-2020" ),
  `Minimum temperature (Deg C)` = c(13.8, 16.4, 26.5, 6.8),
  `Maximum Temperature (Deg C)` = c(23.2, 31.9, 44.3, 10.6),
  `9am relative humidity (%)` = c(74,57,35,76),
  Predicted_value = predict(lm_melbourne_2, newdata = new_data1, interval = "prediction"))

scenarios %>% knitr::kable(digits = 1, format.args = list(big.mark = ","))
```

```
## Warning in `[<-.data.frame`(`*tmp*`, , isn, value = structure(list(`Minimum
## temperature (Deg C)` = structure(c("13.8", : provided 6 variables to replace 4
## variables
```

date	Minimum temperature (Deg C)	Maximum Temperature (Deg C)	9am relative humidity (%)	Predicted_value
29-02-2020	13.8	23.2	74	5.988514
25-12-2020	16.4	31.9	57	8.638394
13-01-2020	26.5	44.3	35	14.718227
06-07-2020	6.8	10.6	76	2.581360

since there is more than 9mm of evaporation at MWC's Cardinia Reservoir (new_data_3_prediction), we calculate the 95% confidence to ensure a continuous supply of water.

```
round(predict(lm_melbourne_2, newdata = new_data_3, interval = "confidence", level = 0.95),3)
```

```
##      fit  lwr    upr
## 1 15.233 9.19 21.276
```

January 13, 2020 can say with 95% confidence that this will occur since the fitted value of confidence level is 15.233, which the fitted value is more than the prediction.

```
round(predict(lm_melbourne_2, newdata = new_data_4, interval = "confidence", level = 0.95), 3)
```

```
##     fit   lwr    upr
## 1 8.033 2.275 13.791
```

July 6, 2020 can we say with 95% confidence that this will not occur since the fitted value of confidence level is 8.033, which the fitted value is more than the prediction.

Conclusion

The Melbourne Water Corporation (MWC) oversees the water supply in Melbourne, Australia. They must thus be required to assist in administrating their Cardinia Reservoir in the city's southeast. Previous predictions of evaporation rates at MWC's reservoirs are incorrect in light of current changes in Melbourne's climate. To secure the reliability of the city's water supply, the objective was to anticipate water evaporation.

The Melbourne Water Corporation was specifically interested in learning whether temporal and climatic elements had a substantial influence on the quantity of evaporation and would cause a significant or little shift in the amount of evaporation. A linear model was created to analyse this, especially for four scenarios: the date, the minimum temperature, the maximum temperature, and the humidity at nine in the morning. Following the application of the models, it was discovered that January 13, 2020, is the day with the majority of more than 9mm of evaporation, and July 6, 2020, is the date with the majority of less than 9mm of evaporation.

However, when building a linear model, there is one model assumption not satisfied– independence, which will lead to those predictors are not independent, so the prediction of evaporation is dependent, which easy to get affected by another variable. The linear model should build the model with more unexpected variables, such as sunshine, one of the variables to evaporate water quickly.

Finally, the Melbourne Water Corporation should also consider the possibility that the information from this period may not apply to future periods. Even though there was once an association, that does not guarantee that there will be one in the future. In addition, several other variables, such as environmental contamination, may impact the water supply even though they have no bearing on evaporation.