Assignment #2 - Exploratory Data Analysis

by François d'Entremont

Import required libraries

```
library(tidyverse)
library(skimr)
library(plotly)
library(knitr)
```

Introduction

I started by browsing kaggle.com looking through datasets but had difficulty finding one that had the requirements for this assignment. I finally managed to find one on the weather in Australia which was very large and had multiple variables that i could draw relationships between. This dataset seemed interesting to me and I was excited to try to make connections with the weather data. Is there a connection between the Temperature and Humidity, Temperature and Pressure, or maybe Temperature and Humidity? On days that it's raining at least 1mm, is the Pressure higher or lower than the mean?

```
weather_data <- read_csv("weatherAUS.csv")
weather_data <- as_tibble(weather_data)
class(weather_data)
```

The first step of tidying my data was to convert my dataset to a tibble so that I can fully use all of my libraries that have the tools necessary to manipulate the data. The next step was looking at the structure of my dataset and seeing which variables are applicable to my analysis. Using the str() function I could see all the variables and which data types they are.

```
str(weather_data)
summary(weather_data)
```

Skimr is a nice library package that shows a nice summary of your dataset

```
skim(weather_data)
```

Looking at the correlations between my variables using the cor() function:

```
cor(weather_data[sapply(weather_data, is.numeric)], use = "complete.obs")
```

I was able to remove some variables did not have strong relationships with other variables. I also looked at variables that have many null values, some up to 48% null, and decided they were not important enough to have in my dataset. Below is a table showing the percent of NA values in my dataset for each variable.

```
colSums(is.na(weather_data)/nrow(weather_data)*100)
```

I have decided only to keep the following variables: Date, Location, Temp3pm, Rainfall, Humidity3pm, WindSpeed3pm, Pressure3pm, RainToday, Season, Month, Temp3pm_quartile. Date is a good variable to have because I can split it into seasons. Location is a good one because it adds a spatial element to the data and I could group by Location to get further insight on the data. For the temperature I decided to keep only keep the temperature at 3pm because it's in the middle of the day and I feel like it's a good measure. There were too many columns with temperature values so to keep the database tidy, I stuck with one temperature variable. I also decided for consistency and to keep the database tidy, to keep the same 3pm values for humidity, windspeed and pressure. Rainfall is another important variable and RainToday is a categorical variable that says yes if it rained over 1mm and no otherwise. I have decided to create a new categorical variable named seasons and another categorical variable called Month.

```
weather_data <- weather_data %>% mutate(MonthNumber = as.integer(format(Date,"%m")))
weather_data$Season <- "Summer"
weather_data$Season[weather_data$MonthNumber >= 3 & weather_data$MonthNumber <= 5] <- "Autumn"
weather_data$Season[weather_data$MonthNumber >= 6 & weather_data$MonthNumber <= 8] <- "Winter"
weather_data$Season[weather_data$MonthNumber >= 9 & weather_data$MonthNumber <= 11] <- "Spring"
weather_data$Season <- as_factor(weather_data$Season)

weather_data <- weather_data %>% mutate(Month = format(Date,"%b"))
weather_data$Month <- as_factor(weather_data$Month)
weather_data$Month <- factor(weather_data$Month, levels =
                       c("January", "February", "March", "April", "May", "June",
                         "July", "August", "September", "October", "November", "December"))
levels(weather_data$Season)
levels(weather_data$Month)
```

Let's factor RainToday

```
weather_data$RainToday <- as_factor(weather_data$RainToday)
levels(weather_data$RainToday)
```

I have split the Temp3pm variable into 4 quartiles and named this categorical variable Temp3pm_quartile.

```
weather_data <- weather_data %>% mutate(Temp3pm_quartile =
                               ifelse(Temp3pm <= 16.60, "Very Low",
                                      ifelse(Temp3pm <= 21.1, "Low",
                                             ifelse(Temp3pm <= 26.40, "High", "Very High"))))
weather_data$Temp3pm_quartile <- factor(weather_data$Temp3pm_quartile, levels = c("Very Low", "Low", "High", "Very High"))
levels(weather_data$Temp3pm_quartile)
```

After selecting my variables, I decided to use the na.omit() function which removes all rows with NA values in them from the dataset. This reduced my dataset from 145460 to 125728 and made my dataset cleaner without sacrificing any significant quantity of data.

```
options(width = 60)
weather <- weather_data %>%
  select(Date, Location, Temp3pm, Rainfall, Humidity3pm, WindSpeed3pm,
         Pressure3pm, RainToday, Season, Month, Temp3pm_quartile)

weather <- na.omit(weather)
```

Now I am done editing my dataset and I can now do analysis on it. Let's do a quick summary of it.

```
options(width = 60)
str(weather)
```

```
summary(weather)
```