Melbourne Rain Prediction

MATH 2319 Machine Learning Applied Project Phase I

Rahul K Gupta (s3635232) & Terrie Christensen (s3664899)

7 April 2018

Contents

1 Introduction

The objective of this project was to build classifiers to predict whether an individual earns more than USD 50,000 or less in a year from the 1994 US Census Data. The data sets were sourced from the UCI Machine Learning Repository. This project has two phases. Phase I focuses on data preprocessing and exploration, as covered in this report. We shall present model building in Phase II. The rest of this report is organised as follow. Section 2 describes the data sets and their attributes. Section 3 covers data pre-processing. In Section 4, we explore each attribute and their inter-relationships. The last section ends with a summary.

2 Data Set

The UCI Machine Learning Repository provides five data sets, but only `adult.data`, `adult.test`, and `adult.names` were useful in this project. `adult.data` and `adult.test` are the training and test data sets respectively. `adult.names` contains the details of attributes or variables. The training data set has 32,561 training observations. Meanwhile, the test data set has 16,281 test observations. Both data sets consist of 14 descriptives features and one target feature. In this project, we combined both training and test data into one. In Phase II, we would build the classifiers from the combined the data set and evaluate their performance using cross-validation.

2.1 Target Feature

The response feature of rain is given as:

$$\text{Tomorrow's Rain} = \begin{cases} Yes & \text{if Rain will occur tomorrow} \\ No & \text{otherwise} \end{cases} \tag{1}$$

The target feature has two classes and hence it is a binary classification problem. To reiterate, The goal is to predict **whether it will rain in Melbourne tomorrow**.

2.2 Descriptive Features

The variable description is produced here from `adult.names` file:

- age: continuous.
- workclass: Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
- fnlwgt: continuous.
- education: Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th- 8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.
- education-num: continuous.
- marital-status: Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married- spouse-absent, Married-AF-spouse.
- occupation: Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-*inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv- house-serv, Protective-serv, Armed-Forces.
- relationship: Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
- race: White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.
- sex: Female, Male.
- capital-gain: continuous.

- capital-loss: continuous.
- hours-per-week: continuous.
- native-country: United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El Salvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands.

Most of the descriptive features are self-explanatory, except fnlwgt which stands for "Final Weight" defined by the US Census. The weight is an "estimate of the number of units in the target population that the responding unit represents". This feature aims to allocate similar weights to people with similar demographic characteristics. For more details, see US Census.

3 Data Pre-processing

3.1 Preliminaries (Optional)

In this project, we used the following `R` packages.

```
library(knitr)
library(mlr)
library(tidyverse)
library(GGally)
library(cowplot)
require(corrplot)
library(reshape2)
theme_set(theme_minimal())
theme_weather <- theme_minimal() +
  theme(plot.subtitle = element_text(color = '#333333', face = "italic"),
        plot.caption = element_text(color = '#666666', face = "italic", size = 9))
```

Text

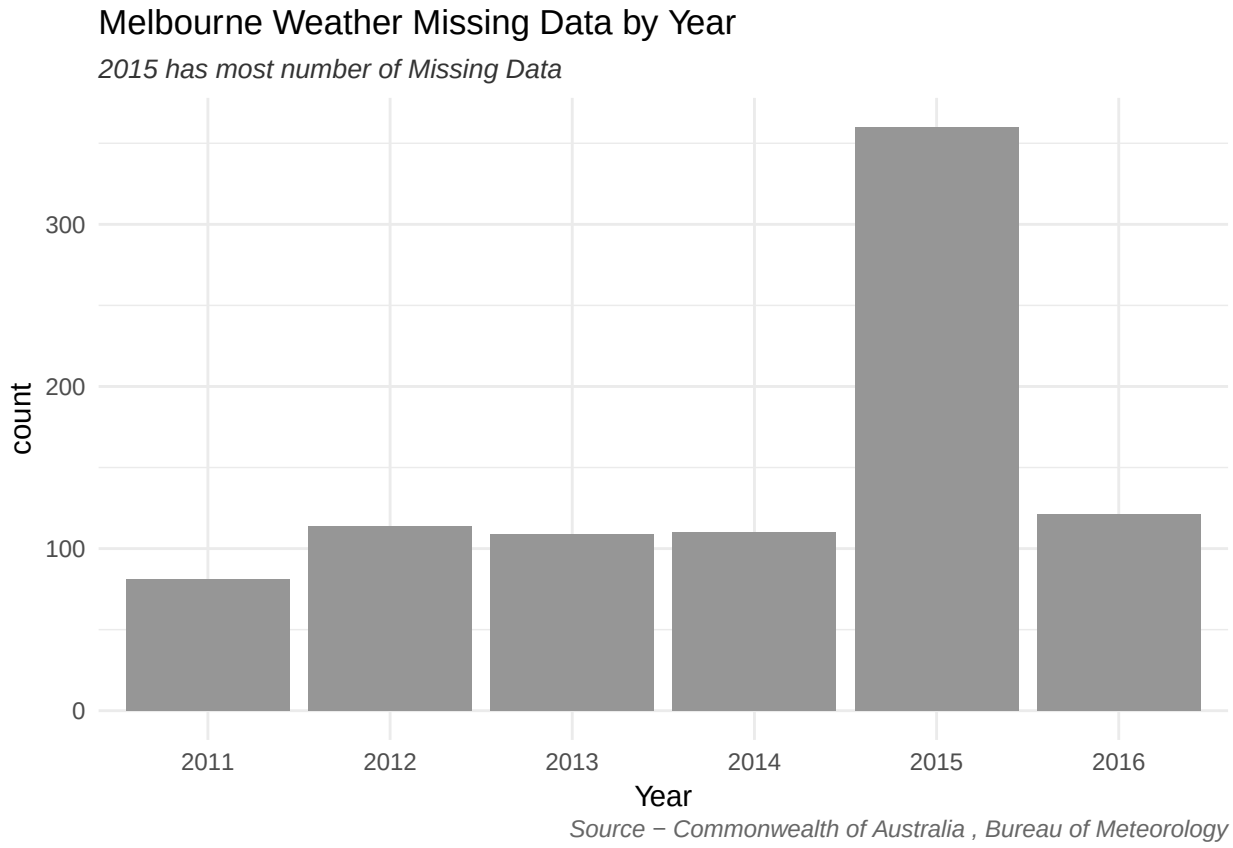
```
weather <- read.csv('weatherAUS 2.csv')
weather$Date = as.Date(weather$Date,'%Y-%m-%d')

# Filter data for Melbourne
Melbourne = weather[weather$Location %in% c('Melbourne','MelbourneAirport'),]

# Add Month and Year-Month
Melbourne$Month = strftime(Melbourne$Date,"%b")
Melbourne$MonthYear = strftime(Melbourne$Date,"%b-%Y")
```

3.2 Data Cleaning and Transformation

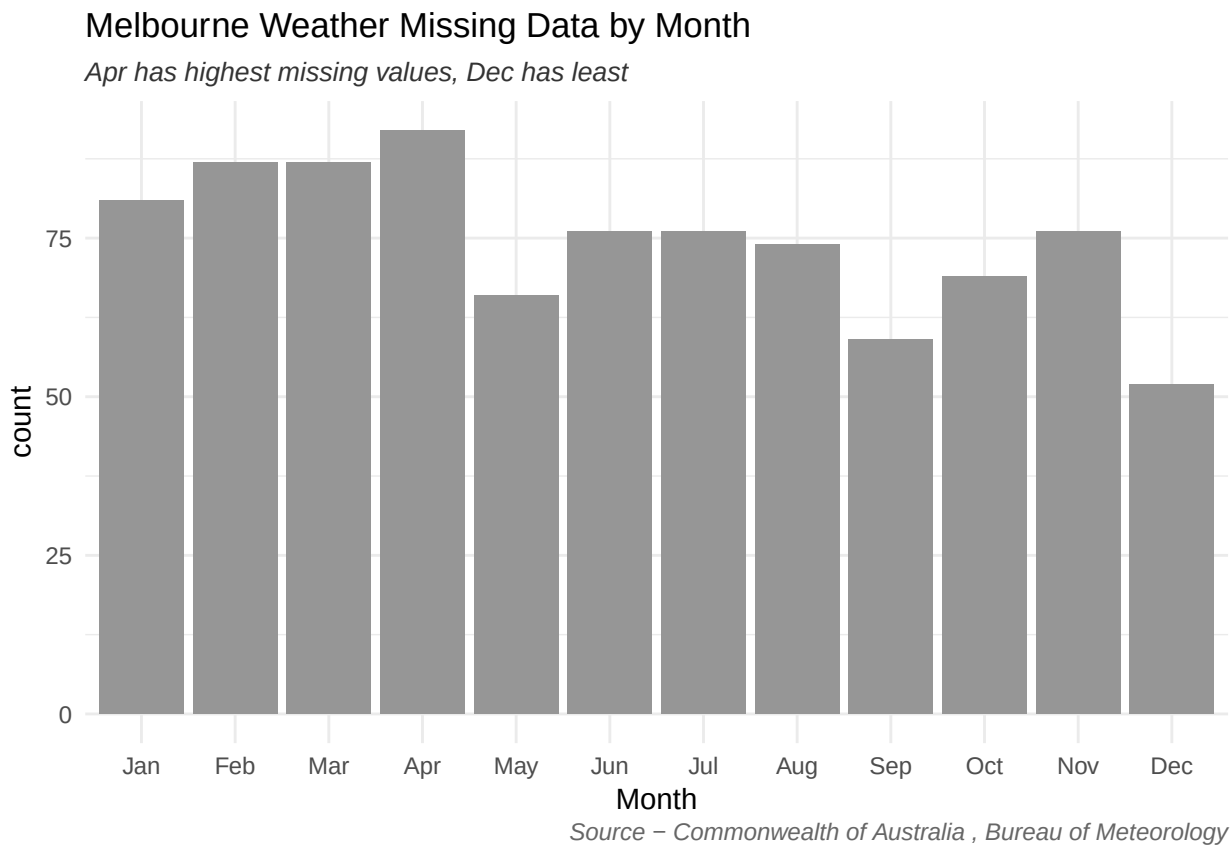
Evaluating missing values via visualizations

```
# Missing Year Data
missing_year = subset(Melbourne ,is.na(Melbourne$RainToday) | is.na(Melbourne$RainTomorrow))
missing_year$Year = strftime(missing_year$Date,"%Y")
ggplot(missing_year,aes(Year)) +
  geom_bar(fill='#969696') +
  ggtitle('Melbourne Weather Missing Data by Year') +
```

```
  labs(subtitle = '2015 has most number of Missing Data',
       caption="Source - Commonwealth of Australia , Bureau of Meteorology") +
  theme_weather
```




```
# Missing Month Data
missing_month = subset(Melbourne ,is.na(RainToday) | is.na(RainTomorrow))

missing_month$Month = factor(missing_month$Month,
                             levels = c('Jan','Feb','Mar','Apr','May','Jun',
                                        'Jul','Aug','Sep','Oct','Nov','Dec'),
                             ordered = TRUE)

ggplot(missing_month,aes(Month)) +
  geom_bar(fill='#969696') +
  ggtitle('Melbourne Weather Missing Data by Month') +
  labs(subtitle = 'Apr has highest missing values, Dec has least',
       caption="Source - Commonwealth of Australia , Bureau of Meteorology") +
  theme_weather
```




```
# Since the month data is distributed well, we can remove them from data set
```

```
# Missing data percentage
missing_percentage = count(missing_month)/count(Melbourne) * 100
missing_percentage
```

```
##          n
## 1 14.43083
```

```
# Remove data where Today and Tomorrow rain information is missing
# 14.4% data is missing
```

```
# Remove Missing Data
Melbourne = Melbourne %>%
  drop_na(RainToday) %>%
  drop_na(RainTomorrow)
```

With `str` and `summarizeColumns` (see Table 1), we noticed the following anomalies:

- All character columns contained excessive white space.
- The target feature, `income` had a cardinality of 4, which was supposed to be 2 since `income` must be binary.
- The `education_num` ranged from 1 to 16 which coincided with the cardinality of `education`. They might represent the same information.
- The max value of `capital_gain` was 99999, potentially a value to represent missing value.
- The max value of `hours_per_week` was 99. It could be a valid or missing value
- On surface, each feature had no missing value, especially the character features.

```
str(Melbourne)
```

```
## 'data.frame':    5307 obs. of  26 variables:
##  $ Date         : Date, format: "2009-01-01" "2009-01-02" ...
##  $ Location     : Factor w/ 49 levels "Adelaide","Albany",..: 20 20 20 20 20 20 20 20 20 20 ...
##  $ MinTemp      : num  11.2 7.8 6.3 8.1 9.7 13.5 15.8 9.7 10.2 10.7 ...
##  $ MaxTemp      : num  19.9 17.8 21.1 29.2 29 31.7 21.4 18.4 19.7 23.6 ...
##  $ Rainfall     : num  0 1.2 0 0 0 0 0.2 0.2 0 0 ...
##  $ Evaporation  : num  5.6 7.2 6.2 6.4 7.4 7.2 8.8 5.2 5.6 7.2 ...
##  $ Sunshine     : num  8.8 12.9 10.5 12.5 12.3 13.7 4.4 11.5 12.6 10.2 ...
##  $ WindGustDir  : Factor w/ 16 levels "E","ENE","ESE",..: 13 11 11 11 10 11 9 10 9 9 ...
##  $ WindGustSpeed: int  69 56 31 35 33 50 46 56 43 43 ...
##  $ WindDir9am   : Factor w/ 16 levels "E","ENE","ESE",..: 14 13 1 5 13 6 12 12 11 16 ...
##  $ WindDir3pm   : Factor w/ 16 levels "E","ENE","ESE",..: 13 11 9 11 11 9 11 11 9 9 ...
##  $ WindSpeed9am : int  33 31 13 2 9 11 17 28 19 7 ...
##  $ WindSpeed3pm : int  43 26 19 20 20 28 28 35 28 24 ...
##  $ Humidity9am  : int  55 50 51 67 51 50 98 51 51 76 ...
##  $ Humidity3pm  : int  37 43 35 23 31 34 67 42 42 46 ...
##  $ Pressure9am  : num  1005 1018 1021 1016 1012 ...
##  $ Pressure3pm  : num  1006 1019 1018 1013 1010 ...
##  $ Cloud9am     : int  7 6 1 5 6 0 8 2 2 7 ...
##  $ Cloud3pm     : int  7 7 7 4 2 1 7 5 7 1 ...
##  $ Temp9am      : num  15.9 12.5 13.4 16 19.4 21.3 16 14.5 14.2 14.5 ...
##  $ Temp3pm      : num  18.1 15.8 19.6 28.2 27.1 29.8 19.9 17.7 19.3 21.8 ...
##  $ RainToday    : Factor w/ 2 levels "No","Yes": 1 2 1 1 1 1 1 1 1 1 ...
##  $ RISK_MM      : num  1.2 0 0 0 0 0.2 0.2 0 0 0 ...
##  $ RainTomorrow : Factor w/ 2 levels "No","Yes": 2 1 1 1 1 1 1 1 1 1 ...
##  $ Month        : chr  "Jan" "Jan" "Jan" "Jan" ...
##  $ MonthYear    : chr  "Jan-2009" "Jan-2009" "Jan-2009" "Jan-2009" ...
```

```
summarizeColumns(Melbourne) %>% knitr::kable(caption =  'Feature Summary ')
```

Table 1: Feature Summary

name	type	na	mean	disp	median	mad	min	max	nlevs
Date	Date	0	NA	NA	NA	NA	1.0	2.0	3193
Location	factor	0	NA	0.4330130	NA	NA	0.0	3009.0	2
MinTemp	numeric	0	10.770379	4.4781066	10.4	4.59606	-1.0	30.5	0
MaxTemp	numeric	0	20.687130	6.4290651	19.4	6.22692	8.4	46.8	0
Rainfall	numeric	0	1.619031	4.8630541	0.0	0.00000	0.0	82.2	0
Evaporation	numeric	3	4.647134	3.3369524	4.0	2.96520	0.0	23.8	0
Sunshine	numeric	2	6.435721	3.9244251	6.6	4.89258	0.0	13.9	0
WindGustDir	factor	29	NA	NA	NA	NA	6.0	1639.0	16
WindGustSpeed	integer	29	46.214475	16.1970129	44.0	16.30860	11.0	122.0	0
WindDir9am	factor	76	NA	NA	NA	NA	25.0	1772.0	16
WindDir3pm	factor	23	NA	NA	NA	NA	37.0	1080.0	16
WindSpeed9am	integer	3	19.694570	11.7510252	17.0	11.86080	0.0	67.0	0
WindSpeed3pm	integer	0	22.509704	10.0576575	22.0	10.37820	0.0	76.0	0
Humidity9am	integer	10	68.850859	15.3960442	70.0	14.82600	11.0	100.0	0
Humidity3pm	integer	15	51.027778	17.0937630	50.0	14.82600	6.0	100.0	0
Pressure9am	numeric	0	1017.931581	7.7613507	1018.1	7.56126	988.9	1039.3	0
Pressure3pm	numeric	2	1016.080207	7.5895129	1016.4	7.56126	988.2	1036.0	0
Cloud9am	integer	273	5.255860	2.5129291	7.0	1.48260	0.0	8.0	0
Cloud3pm	integer	334	5.278906	2.3386018	6.0	1.48260	0.0	8.0	0
Temp9am	numeric	1	14.370222	4.8861684	13.8	4.59606	3.1	35.5	0
Temp3pm	numeric	3	19.152658	6.1762272	18.1	6.07866	6.2	46.1	0

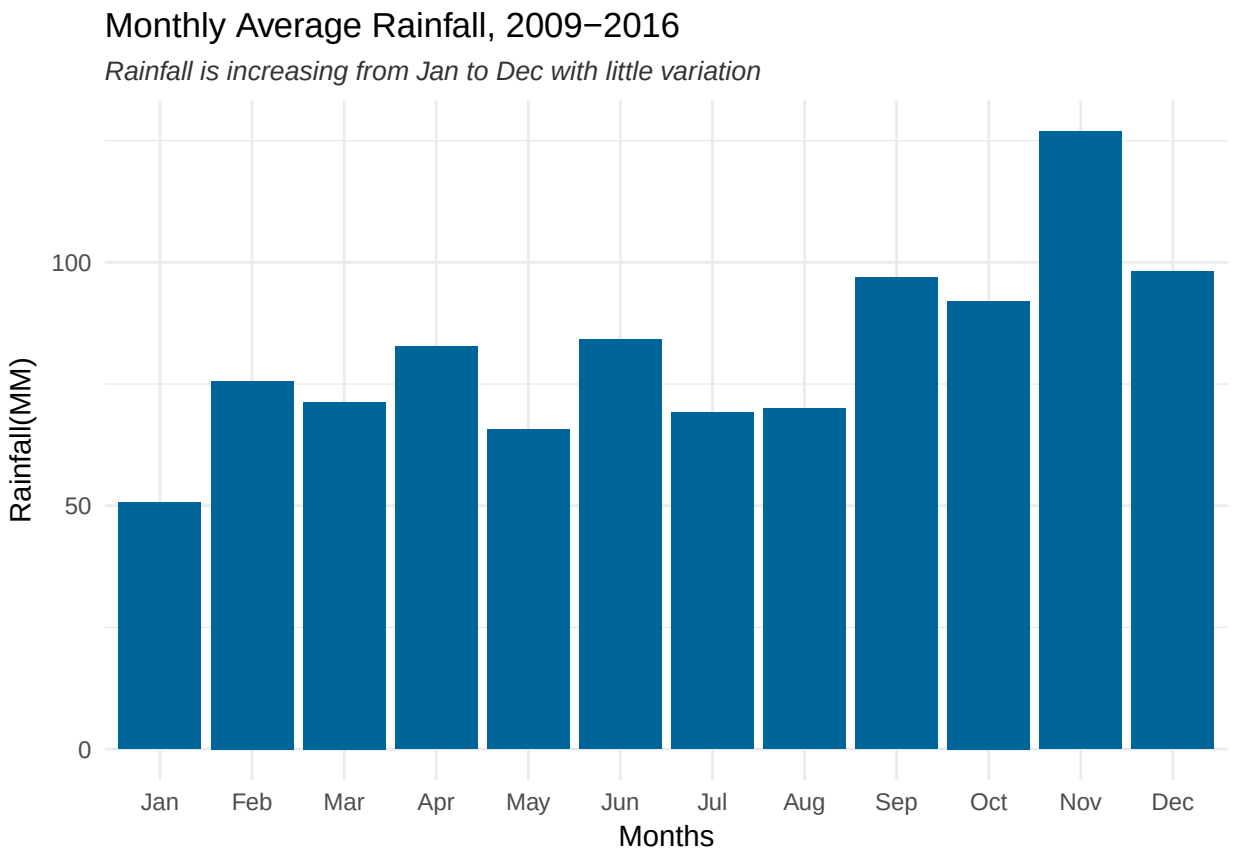
name	type	na	mean	disp	median	mad	min	max	nlevs
RainToday	factor	0	NA	0.2323347	NA	NA	1233.0	4074.0	2
RISK_MM	numeric	0	1.479932	4.4412158	0.0	0.00000	0.0	75.8	0
RainTomorrow	factor	0	NA	0.2234784	NA	NA	1186.0	4121.0	2
Month	character	0	NA	0.9072923	NA	NA	365.0	492.0	12
MonthYear	character	0	NA	0.9883173	NA	NA	28.0	62.0	105

```
# Change cardinal direction to degrees
direction = read.csv('direction.csv')
Melbourne$WindGustDirDegree = factor(Melbourne$WindGustDir,
                                     levels = direction$CardinalDirection,
                                     labels = direction$DegreeDirectionMean)

Melbourne$WindDir9amDegree = factor(Melbourne$WindDir9am,
                                     levels = direction$CardinalDirection,
                                     labels = direction$DegreeDirectionMean)

Melbourne$WindDir3pmDegree = factor(Melbourne$WindDir3pm,
                                     levels = direction$CardinalDirection,
                                     labels = direction$DegreeDirectionMean)
```

3.3 Univariate Visualisation

```
# --------------
# Visualizations
# --------------

# ------------------
# Monthly Average Rainfall from 2009-2016
total_yearmonth_rainfall = Melbourne %>%
  select(MonthYear,Rainfall) %>%
  group_by(MonthYear) %>%
  summarise(monthTotal = sum(Rainfall, na.rm = TRUE))

total_yearmonth_rainfall$month = substr(total_yearmonth_rainfall$MonthYear, 1, 3)

monthly_rainfall =  total_yearmonth_rainfall %>%
  group_by(month) %>%
  summarise(avgRainfall = mean(monthTotal))

monthly_rainfall$month = factor(monthly_rainfall$month,
                                levels = c('Jan','Feb','Mar','Apr',
                                           'May','Jun','Jul','Aug',
                                           'Sep','Oct','Nov','Dec'),
                                ordered = TRUE)

ggplot(monthly_rainfall,aes(month,avgRainfall)) +
  geom_bar(stat = 'identity',fill="#006699") +
  ggtitle('Monthly Average Rainfall, 2009-2016') +
  xlab('Months') +
  ylab('Rainfall(MM)') +
```

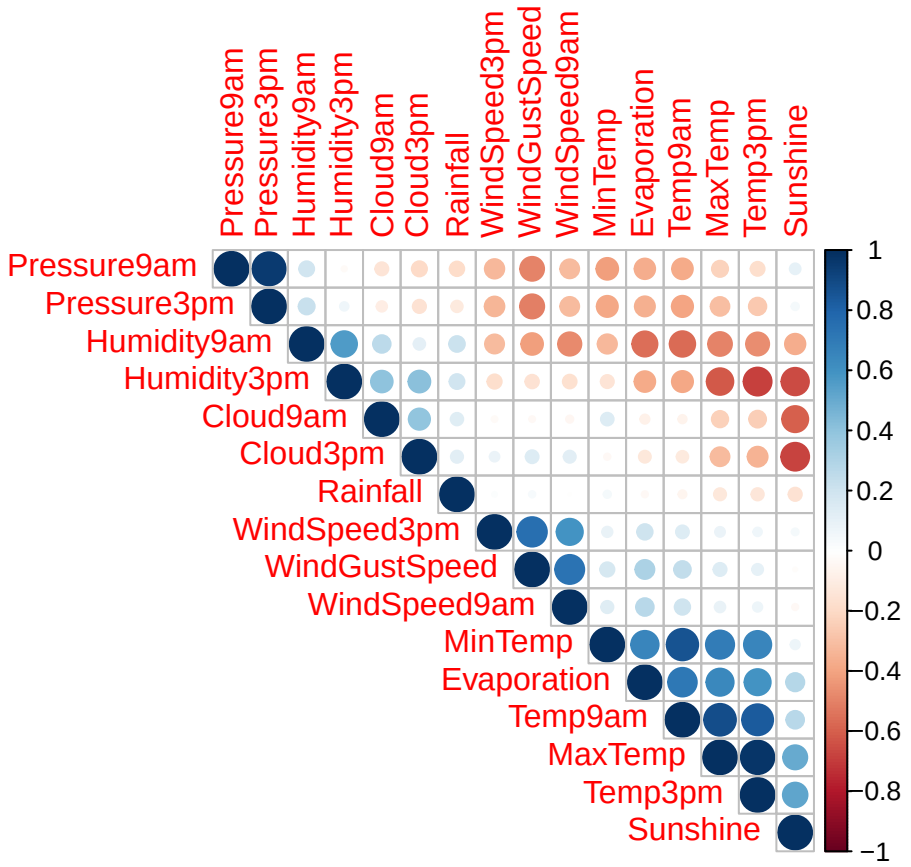
```
  labs(subtitle = 'Rainfall is increasing from Jan to Dec with little variation') +
  theme(plot.subtitle = element_text(color = '#333333',face = "italic"))
```




```
# Except Nov, the variation isn't too much between Consecutive years. The possibility of getting rain
# in Feb is almost same as Mar or Apr, hence we connot conclude the rain by month.
# Melbourne doesn't have a fix rainy season hence average rainfall is distributed
```

3.3.1 Correlation Matrix

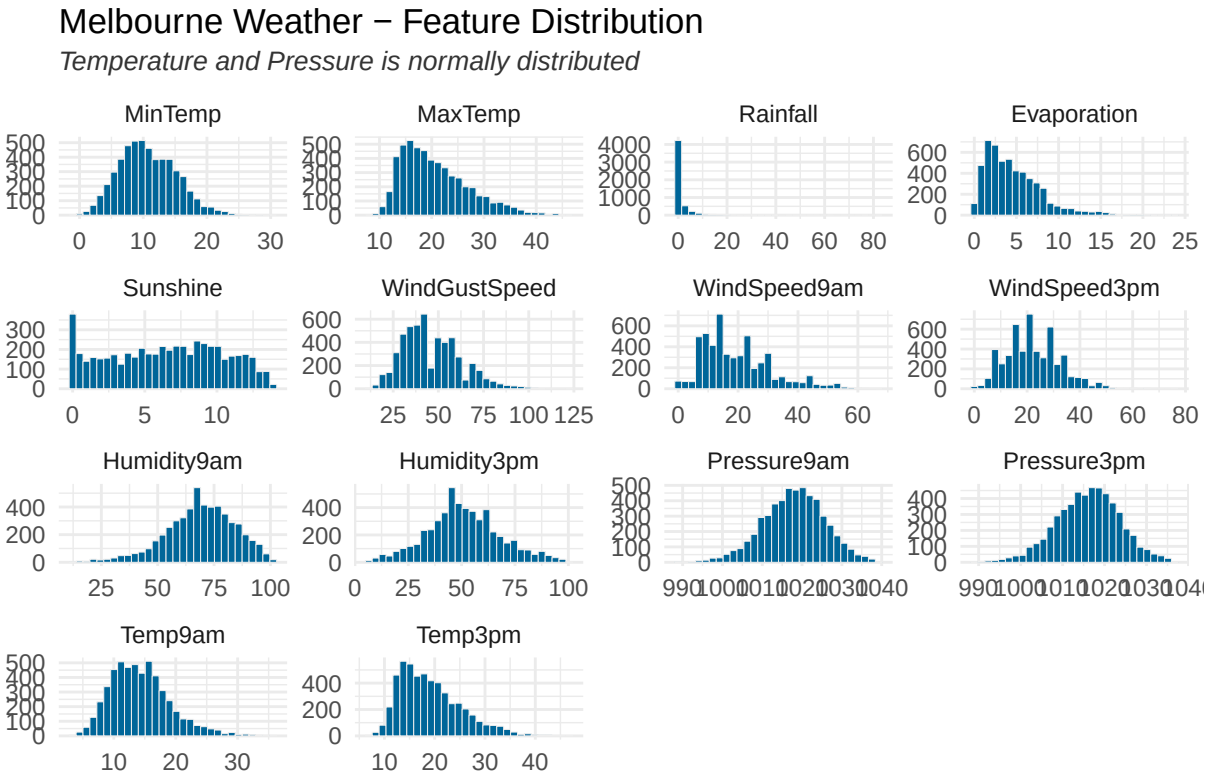
```
# ------------------
# Correlation Matrix
correlation_matrix = cor(Melbourne[,c(3:7,9,12:21)],use='na.or.complete')
corrplot(correlation_matrix, order = 'AOE', type = "upper")
```




```
# Very few factors has high correlation
```

3.3.2 continuous variables

```
# visualize continuous variables
continouous_cols = c('MinTemp','MaxTemp','Rainfall','Evaporation','Sunshine',
                     'WindGustSpeed','WindSpeed9am','WindSpeed3pm','Humidity9am',
                     'Humidity3pm','Pressure9am','Pressure3pm','Temp9am','Temp3pm')
continouous_variables = melt(Melbourne[,continouous_cols])

ggplot(continouous_variables) +
  stat_bin(aes(value), color='#EEEEEE', size=.1, fill="#006699") +
  facet_wrap(~variable,scales="free") +
  ggtitle('Melbourne Weather - Feature Distribution') +
  xlab('') +
  ylab('') +
  labs(subtitle = 'Temperature and Pressure is normally distributed',
       caption="Source - Commonwealth of Australia , Bureau of Meteorology") +
  theme_weather
```



Source − Commonwealth of Australia , Bureau of Meteorology

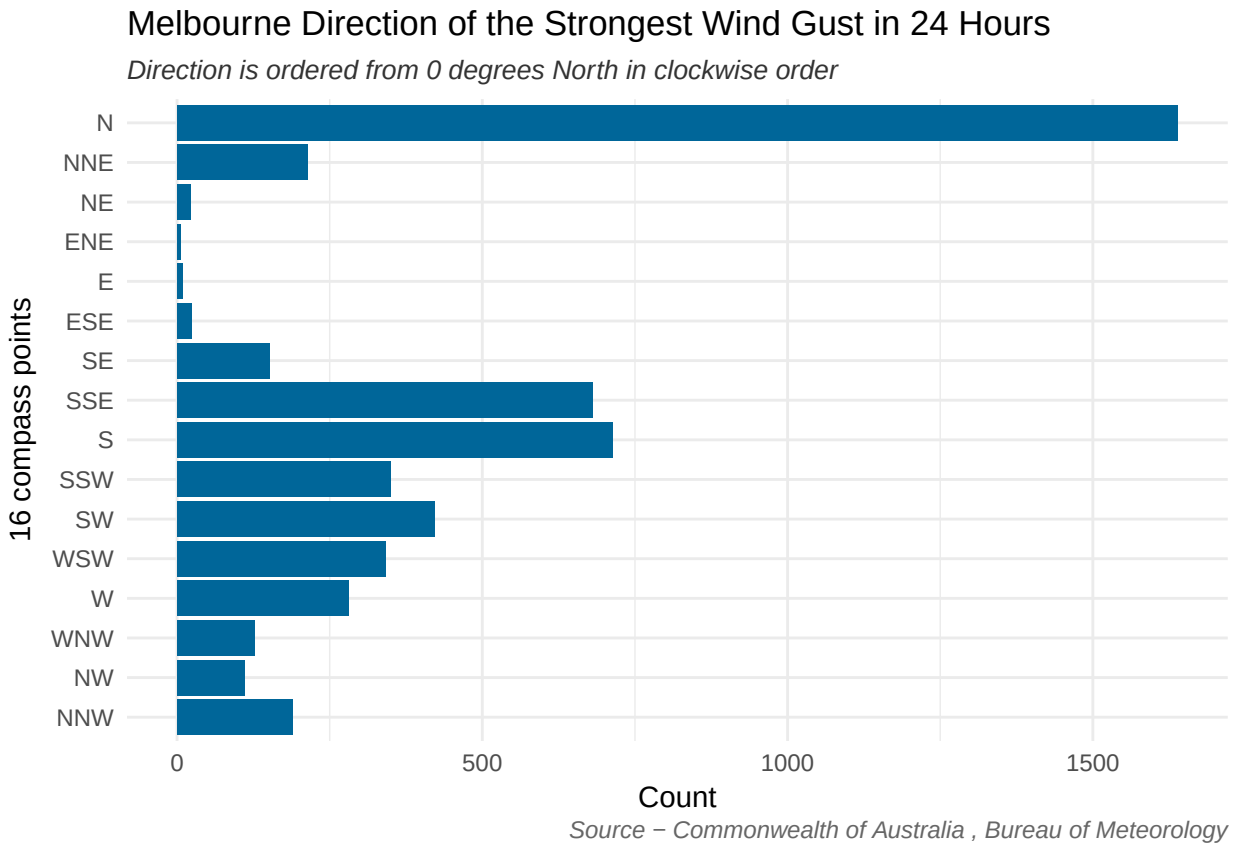
3.3.3 Categorical variables

```
# visualize categorical variables
head(Melbourne)
```

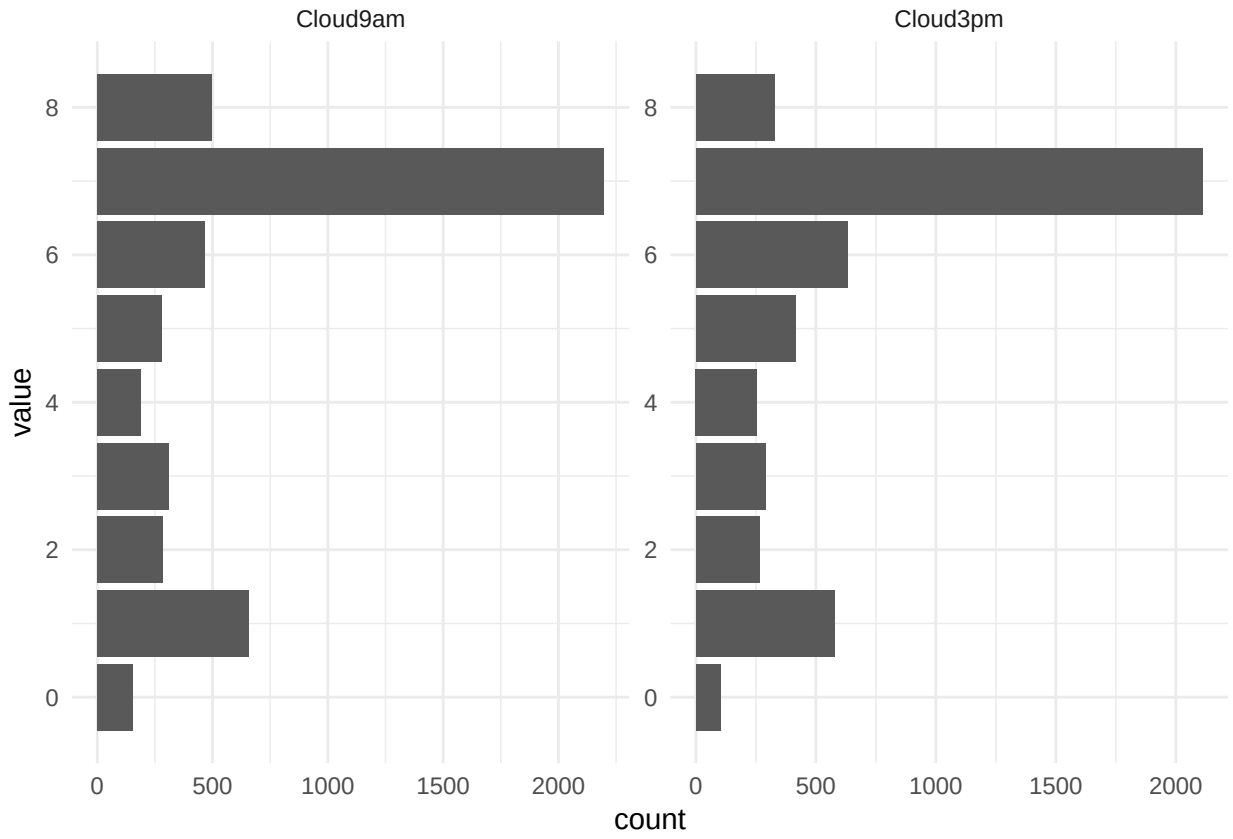
```
##             Date         Location MinTemp MaxTemp Rainfall Evaporation
## 64192 2009-01-01 MelbourneAirport    11.2    19.9      0.0         5.6
## 64193 2009-01-02 MelbourneAirport     7.8    17.8      1.2         7.2
## 64194 2009-01-03 MelbourneAirport     6.3    21.1      0.0         6.2
## 64195 2009-01-04 MelbourneAirport     8.1    29.2      0.0         6.4
## 64196 2009-01-05 MelbourneAirport     9.7    29.0      0.0         7.4
## 64197 2009-01-06 MelbourneAirport    13.5    31.7      0.0         7.2
##       Sunshine WindGustDir WindGustSpeed WindDir9am WindDir3pm
## 64192      8.8          SW            69          W         SW
## 64193     12.9         SSE            56         SW        SSE
## 64194     10.5         SSE            31          E          S
## 64195     12.5         SSE            35         NE        SSE
## 64196     12.3          SE            33         SW        SSE
## 64197     13.7         SSE            50        NNE          S
##       WindSpeed9am WindSpeed3pm Humidity9am Humidity3pm Pressure9am
## 64192           33           43          55          37      1005.1
## 64193           31           26          50          43      1018.0
## 64194           13           19          51          35      1020.8
## 64195            2           20          67          23      1016.2
## 64196            9           20          51          31      1011.9
## 64197           11           28          50          34      1010.7
```

```
##       Pressure3pm Cloud9am Cloud3pm Temp9am Temp3pm RainToday RISK_MM
## 64192      1006.4        7        7    15.9    18.1        No     1.2
## 64193      1019.3        6        7    12.5    15.8       Yes     0.0
## 64194      1017.6        1        7    13.4    19.6        No     0.0
## 64195      1012.8        5        4    16.0    28.2        No     0.0
## 64196      1010.3        6        2    19.4    27.1        No     0.0
## 64197      1007.7        0        1    21.3    29.8        No     0.2
##       RainTomorrow Month MonthYear WindGustDirDegree WindDir9amDegree
## 64192          Yes   Jan  Jan-2009               225              270
## 64193           No   Jan  Jan-2009             157.5              225
## 64194           No   Jan  Jan-2009             157.5               90
## 64195           No   Jan  Jan-2009             157.5               45
## 64196           No   Jan  Jan-2009               135              225
## 64197           No   Jan  Jan-2009             157.5             22.5
##       WindDir3pmDegree
## 64192              225
## 64193            157.5
## 64194              180
## 64195            157.5
## 64196            157.5
## 64197              180
```

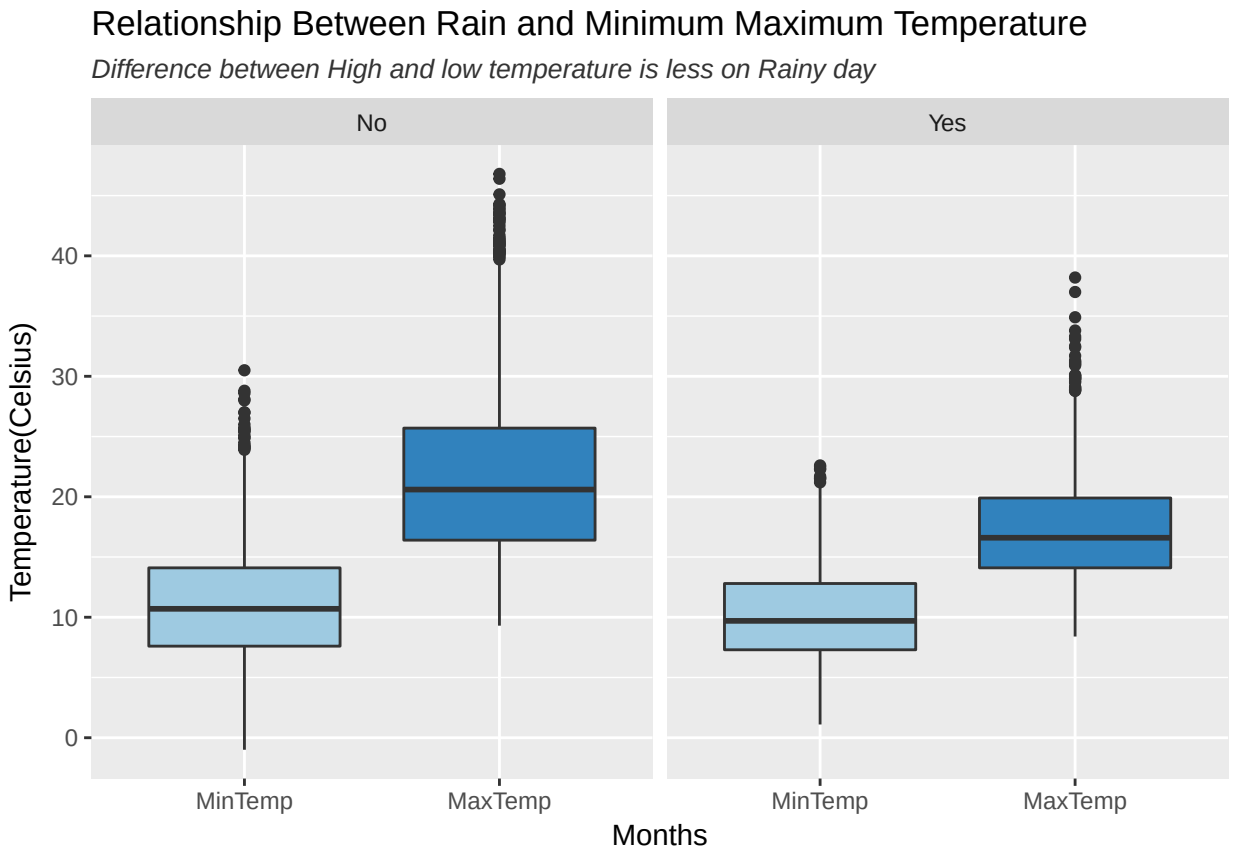
```
wind_direction = Melbourne
wind_direction$WindGustDir = factor(wind_direction$WindGustDir,
                      levels = rev(direction$CardinalDirection),
                      ordered = TRUE)

wind_direction %>%
  filter(!is.na(WindGustDir)) %>%
  ggplot(aes(WindGustDir)) +
  geom_bar(fill="#006699") +
  coord_flip() +
  ggtitle('Melbourne Direction of the Strongest Wind Gust in 24 Hours') +
  ylab('Count') +
  xlab('16 compass points') +
  labs(subtitle = 'Direction is ordered from 0 degrees North in clockwise order',
       caption="Source - Commonwealth of Australia , Bureau of Meteorology") +
  theme_weather
```




```
melt(Melbourne[,c('Cloud9am','Cloud3pm')]) %>%
ggplot(aes(value)) +
geom_bar() +
facet_wrap(~variable,scales="free") +
coord_flip()
```



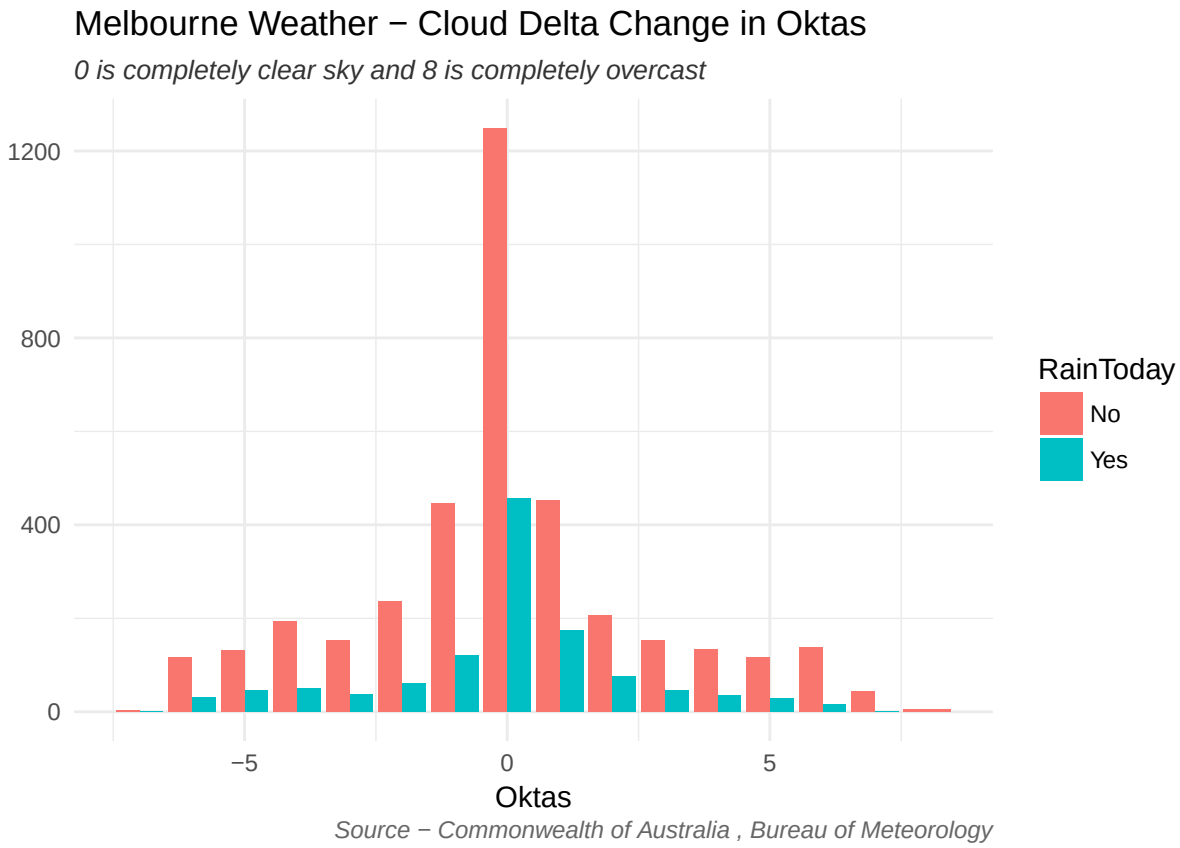

```
# ------------------------------------------------------------
# Relationship Between Rain and Minimum Maximum Temperature
# Convert wide to long
Temperature = Melbourne %>%
  select(MinTemp,MaxTemp,RainToday) %>%
  drop_na(RainToday) %>%
  melt(id.vars = 'RainToday')

ggplot(Temperature,aes(variable,value)) +
  geom_boxplot(fill=c('#9ecae1','#3182bd','#9ecae1','#3182bd')) +
  facet_grid(~RainToday) +
  theme_gray() +
  ggtitle('Relationship Between Rain and Minimum Maximum Temperature') +
  xlab('Months') +
  ylab('Temperature(Celsius)') +
  labs(subtitle = 'Difference between High and low temperature is less on Rainy day') +
  theme(plot.subtitle = element_text(color = '#333333',face = "italic"))
```



3.3.4 Monthly Average Rainfall from 2009-2016

```
# ------------------
# Monthly Average Rainfall from 2009-2016

# Not working

# Except Nov, the variation isn't too much between Consecutive years. The possibility of getting rain
# in Feb is almost same as Mar or Apr, hence we connot conclude the rain by month.
# Melbourne doesn't have a fix rainy season hence average rainfall is distributed
```

```
##################
# Delta change
##################

delta_cloud = Melbourne
delta_cloud$deltaCloud = delta_cloud$Cloud9am - delta_cloud$Cloud3pm
ggplot(delta_cloud, aes(deltaCloud, fill=RainToday)) +
  geom_bar(position = 'dodge') +
  ggtitle('Melbourne Weather - Cloud Delta Change in Oktas') +
  xlab('Oktas') +
  ylab('') +
  labs(subtitle = '0 is completely clear sky and 8 is completely overcast',
       caption="Source - Commonwealth of Australia , Bureau of Meteorology") +
  theme_weather
```



3.3.5 Comparision

```
# ------------------------------------------------
# Temperature frop from 9AM to 3 PM for the same day
temperature_drop= Melbourne[Melbourne$Temp9am < Melbourne$Temp3pm, ]
prop.table(table(temperature_drop$RainToday)) * 100
```

```
##
##       No      Yes
## 77.14115 22.85885
```

```
# Even if temperature drops by 3 PM, chances of having rain is just 22%
temperature_drop= Melbourne[Melbourne$Temp9am > Melbourne$Temp3pm, ]
prop.table(table(temperature_drop$RainToday)) * 100
```

```
##
##       No      Yes
## 71.37681 28.62319
```

```
# ------------------------------------------------


# ------------------------------------------------
# Relationship between today and tomorrow's rain
prop.table(table(Melbourne$RainToday, Melbourne$RainTomorrow,dnn=c('Rain Today','Rain Tomorrow'))) * 10
```

```
##           Rain Tomorrow
```

```
## Rain Today        No       Yes
##        No  62.954588 13.811946
##        Yes 14.697569  8.535896
```

```
# 9% times if it rains today then it rained tomorrow
# 14% times it rains today and didn't rain tomorrow
# If it didn't rain today then chances are it won't rain tomorrow

# TODO we can also do chi square test of independence
# ------------------------------------------------
```

4 Summary