2.R_Visualization

Kim Wong

04/01/2021

Material taken from

Graham J. Williams. 2017. The Essentials of Data Science: Knowledge Discovery Using R (1st ed.). Chapman & Hall/CRC.

Load required packages

```
library(dplyr)        # glimpse().
```

```
##
## 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(ggplot2)      # Visualise data.
library(magrittr)     # Data pipelines: %>% %<>% %T>% equals().
library(randomForest) # na.roughfix() for missing data.
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(rattle)       # normVarNames().
```

```
## Loading required package: tibble
```

```
## Loading required package: bitops
```

```
## Rattle: A free graphical interface for data science with R.
## Version 5.4.0 Copyright (c) 2006-2020 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.
```

```
##
## Attaching package: 'rattle'
```

```
## The following object is masked from 'package:randomForest':
##
##     importance
```

```
library(rattle.data)  # weatherAUS.
```

```
##
## Attaching package: 'rattle.data'
```

```
## The following objects are masked from 'package:rattle':
##
##     locationsAUS, weather, weatherAUS
```

```
library(scales)       # commas(), percent().
library(stringr)      # str_replace_all().
library(stringi)      # String concat operator %s+%.
```

Edgar Anderson's Iris Data

```
iris
```

```
##     Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
## 1            5.1         3.5          1.4         0.2     setosa
## 2            4.9         3.0          1.4         0.2     setosa
## 3            4.7         3.2          1.3         0.2     setosa
## 4            4.6         3.1          1.5         0.2     setosa
## 5            5.0         3.6          1.4         0.2     setosa
## 6            5.4         3.9          1.7         0.4     setosa
## 7            4.6         3.4          1.4         0.3     setosa
## 8            5.0         3.4          1.5         0.2     setosa
## 9            4.4         2.9          1.4         0.2     setosa
## 10           4.9         3.1          1.5         0.1     setosa
## 11           5.4         3.7          1.5         0.2     setosa
## 12           4.8         3.4          1.6         0.2     setosa
## 13           4.8         3.0          1.4         0.1     setosa
## 14           4.3         3.0          1.1         0.1     setosa
## 15           5.8         4.0          1.2         0.2     setosa
## 16           5.7         4.4          1.5         0.4     setosa
## 17           5.4         3.9          1.3         0.4     setosa
## 18           5.1         3.5          1.4         0.3     setosa
## 19           5.7         3.8          1.7         0.3     setosa
## 20           5.1         3.8          1.5         0.3     setosa
## 21           5.4         3.4          1.7         0.2     setosa
## 22           5.1         3.7          1.5         0.4     setosa
## 23           4.6         3.6          1.0         0.2     setosa
## 24           5.1         3.3          1.7         0.5     setosa
## 25           4.8         3.4          1.9         0.2     setosa
## 26           5.0         3.0          1.6         0.2     setosa
## 27           5.0         3.4          1.6         0.4     setosa
## 28           5.2         3.5          1.5         0.2     setosa
```

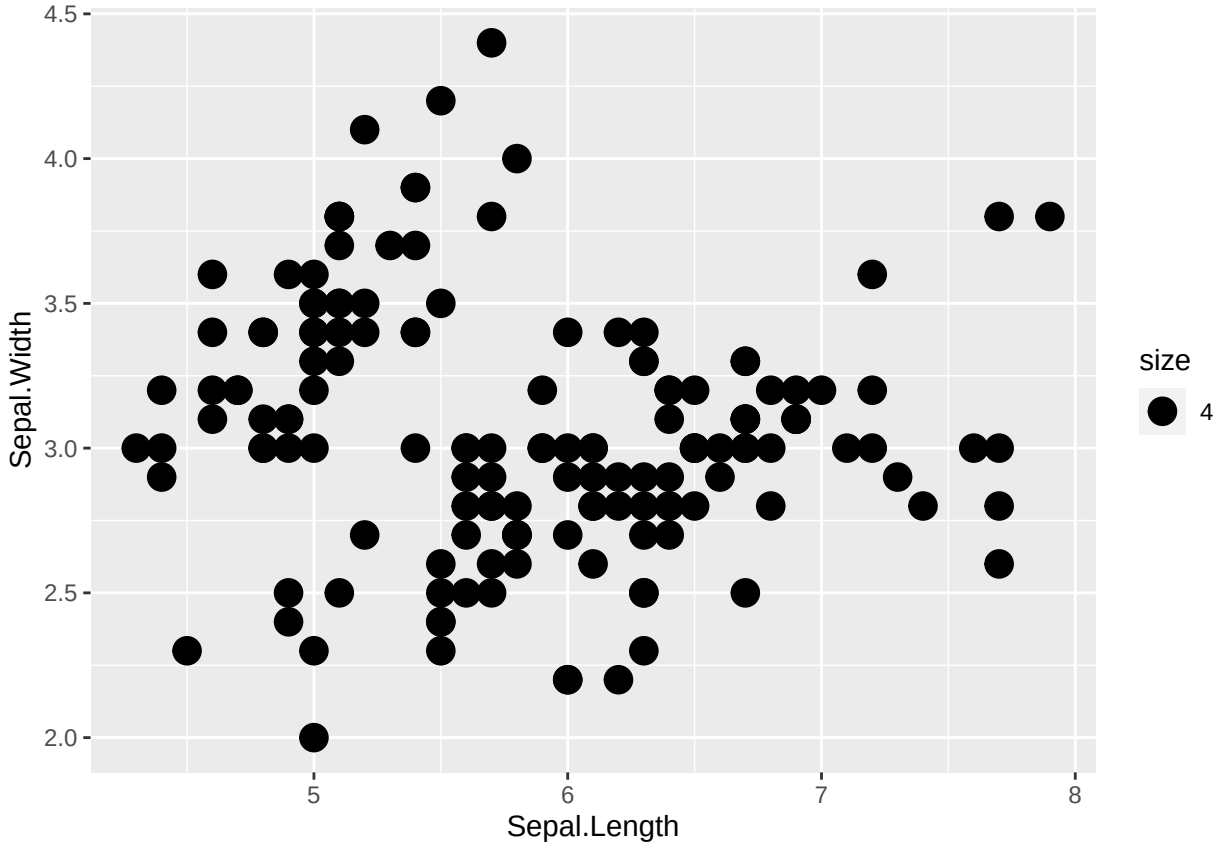
```
## 29          5.2         3.4          1.4         0.2     setosa
## 30          4.7         3.2          1.6         0.2     setosa
## 31          4.8         3.1          1.6         0.2     setosa
## 32          5.4         3.4          1.5         0.4     setosa
## 33          5.2         4.1          1.5         0.1     setosa
## 34          5.5         4.2          1.4         0.2     setosa
## 35          4.9         3.1          1.5         0.2     setosa
## 36          5.0         3.2          1.2         0.2     setosa
## 37          5.5         3.5          1.3         0.2     setosa
## 38          4.9         3.6          1.4         0.1     setosa
## 39          4.4         3.0          1.3         0.2     setosa
## 40          5.1         3.4          1.5         0.2     setosa
## 41          5.0         3.5          1.3         0.3     setosa
## 42          4.5         2.3          1.3         0.3     setosa
## 43          4.4         3.2          1.3         0.2     setosa
## 44          5.0         3.5          1.6         0.6     setosa
## 45          5.1         3.8          1.9         0.4     setosa
## 46          4.8         3.0          1.4         0.3     setosa
## 47          5.1         3.8          1.6         0.2     setosa
## 48          4.6         3.2          1.4         0.2     setosa
## 49          5.3         3.7          1.5         0.2     setosa
## 50          5.0         3.3          1.4         0.2     setosa
## 51          7.0         3.2          4.7         1.4 versicolor
## 52          6.4         3.2          4.5         1.5 versicolor
## 53          6.9         3.1          4.9         1.5 versicolor
## 54          5.5         2.3          4.0         1.3 versicolor
## 55          6.5         2.8          4.6         1.5 versicolor
## 56          5.7         2.8          4.5         1.3 versicolor
## 57          6.3         3.3          4.7         1.6 versicolor
## 58          4.9         2.4          3.3         1.0 versicolor
## 59          6.6         2.9          4.6         1.3 versicolor
## 60          5.2         2.7          3.9         1.4 versicolor
## 61          5.0         2.0          3.5         1.0 versicolor
## 62          5.9         3.0          4.2         1.5 versicolor
## 63          6.0         2.2          4.0         1.0 versicolor
## 64          6.1         2.9          4.7         1.4 versicolor
## 65          5.6         2.9          3.6         1.3 versicolor
## 66          6.7         3.1          4.4         1.4 versicolor
## 67          5.6         3.0          4.5         1.5 versicolor
## 68          5.8         2.7          4.1         1.0 versicolor
## 69          6.2         2.2          4.5         1.5 versicolor
## 70          5.6         2.5          3.9         1.1 versicolor
## 71          5.9         3.2          4.8         1.8 versicolor
## 72          6.1         2.8          4.0         1.3 versicolor
## 73          6.3         2.5          4.9         1.5 versicolor
## 74          6.1         2.8          4.7         1.2 versicolor
## 75          6.4         2.9          4.3         1.3 versicolor
## 76          6.6         3.0          4.4         1.4 versicolor
## 77          6.8         2.8          4.8         1.4 versicolor
## 78          6.7         3.0          5.0         1.7 versicolor
## 79          6.0         2.9          4.5         1.5 versicolor
## 80          5.7         2.6          3.5         1.0 versicolor
## 81          5.5         2.4          3.8         1.1 versicolor
## 82          5.5         2.4          3.7         1.0 versicolor
```

```
## 83          5.8         2.7          3.9         1.2 versicolor
## 84          6.0         2.7          5.1         1.6 versicolor
## 85          5.4         3.0          4.5         1.5 versicolor
## 86          6.0         3.4          4.5         1.6 versicolor
## 87          6.7         3.1          4.7         1.5 versicolor
## 88          6.3         2.3          4.4         1.3 versicolor
## 89          5.6         3.0          4.1         1.3 versicolor
## 90          5.5         2.5          4.0         1.3 versicolor
## 91          5.5         2.6          4.4         1.2 versicolor
## 92          6.1         3.0          4.6         1.4 versicolor
## 93          5.8         2.6          4.0         1.2 versicolor
## 94          5.0         2.3          3.3         1.0 versicolor
## 95          5.6         2.7          4.2         1.3 versicolor
## 96          5.7         3.0          4.2         1.2 versicolor
## 97          5.7         2.9          4.2         1.3 versicolor
## 98          6.2         2.9          4.3         1.3 versicolor
## 99          5.1         2.5          3.0         1.1 versicolor
## 100         5.7         2.8          4.1         1.3 versicolor
## 101         6.3         3.3          6.0         2.5  virginica
## 102         5.8         2.7          5.1         1.9  virginica
## 103         7.1         3.0          5.9         2.1  virginica
## 104         6.3         2.9          5.6         1.8  virginica
## 105         6.5         3.0          5.8         2.2  virginica
## 106         7.6         3.0          6.6         2.1  virginica
## 107         4.9         2.5          4.5         1.7  virginica
## 108         7.3         2.9          6.3         1.8  virginica
## 109         6.7         2.5          5.8         1.8  virginica
## 110         7.2         3.6          6.1         2.5  virginica
## 111         6.5         3.2          5.1         2.0  virginica
## 112         6.4         2.7          5.3         1.9  virginica
## 113         6.8         3.0          5.5         2.1  virginica
## 114         5.7         2.5          5.0         2.0  virginica
## 115         5.8         2.8          5.1         2.4  virginica
## 116         6.4         3.2          5.3         2.3  virginica
## 117         6.5         3.0          5.5         1.8  virginica
## 118         7.7         3.8          6.7         2.2  virginica
## 119         7.7         2.6          6.9         2.3  virginica
## 120         6.0         2.2          5.0         1.5  virginica
## 121         6.9         3.2          5.7         2.3  virginica
## 122         5.6         2.8          4.9         2.0  virginica
## 123         7.7         2.8          6.7         2.0  virginica
## 124         6.3         2.7          4.9         1.8  virginica
## 125         6.7         3.3          5.7         2.1  virginica
## 126         7.2         3.2          6.0         1.8  virginica
## 127         6.2         2.8          4.8         1.8  virginica
## 128         6.1         3.0          4.9         1.8  virginica
## 129         6.4         2.8          5.6         2.1  virginica
## 130         7.2         3.0          5.8         1.6  virginica
## 131         7.4         2.8          6.1         1.9  virginica
## 132         7.9         3.8          6.4         2.0  virginica
## 133         6.4         2.8          5.6         2.2  virginica
## 134         6.3         2.8          5.1         1.5  virginica
## 135         6.1         2.6          5.6         1.4  virginica
## 136         7.7         3.0          6.1         2.3  virginica
```

```
## 137          6.3         3.4          5.6         2.4  virginica
## 138          6.4         3.1          5.5         1.8  virginica
## 139          6.0         3.0          4.8         1.8  virginica
## 140          6.9         3.1          5.4         2.1  virginica
## 141          6.7         3.1          5.6         2.4  virginica
## 142          6.9         3.1          5.1         2.3  virginica
## 143          5.8         2.7          5.1         1.9  virginica
## 144          6.8         3.2          5.9         2.3  virginica
## 145          6.7         3.3          5.7         2.5  virginica
## 146          6.7         3.0          5.2         2.3  virginica
## 147          6.3         2.5          5.0         1.9  virginica
## 148          6.5         3.0          5.2         2.0  virginica
## 149          6.2         3.4          5.4         2.3  virginica
## 150          5.9         3.0          5.1         1.8  virginica
```

A Simple scatterplot

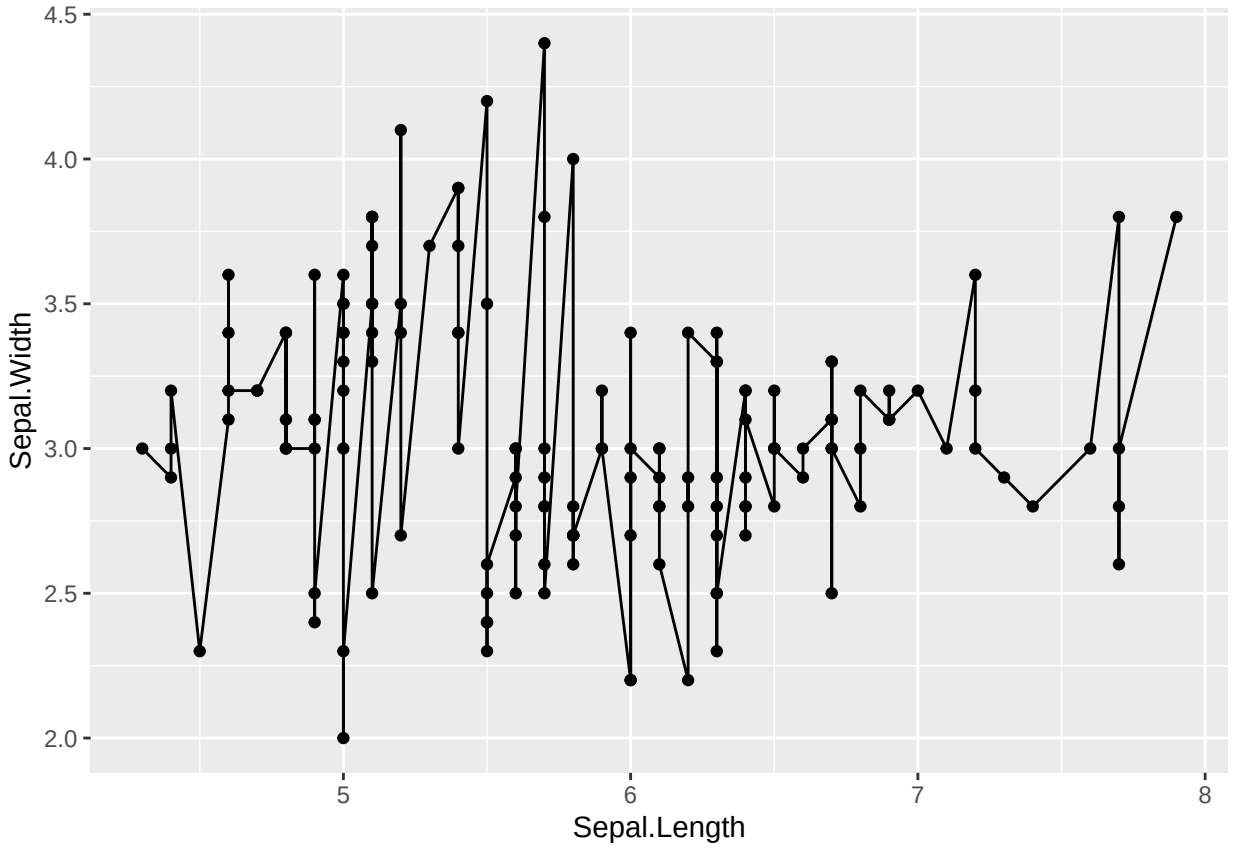
```
iris %>%
  ggplot(aes(x=Sepal.Length, y=Sepal.Width, size=4)) +
  geom_point()
```



A simple line plot with data points

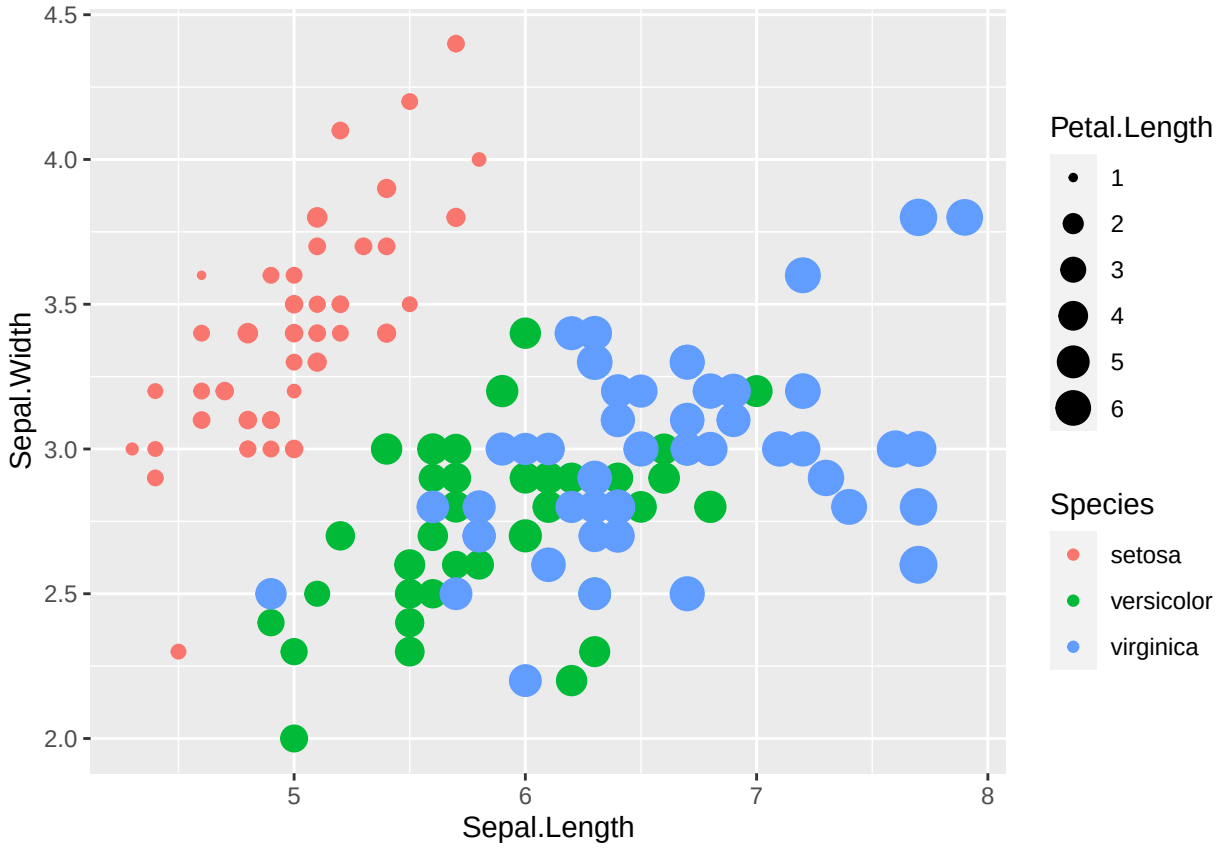
```
iris %>%
  ggplot(aes(x=Sepal.Length, y=Sepal.Width)) +
  geom_point() +
```

```
  geom_line()
```



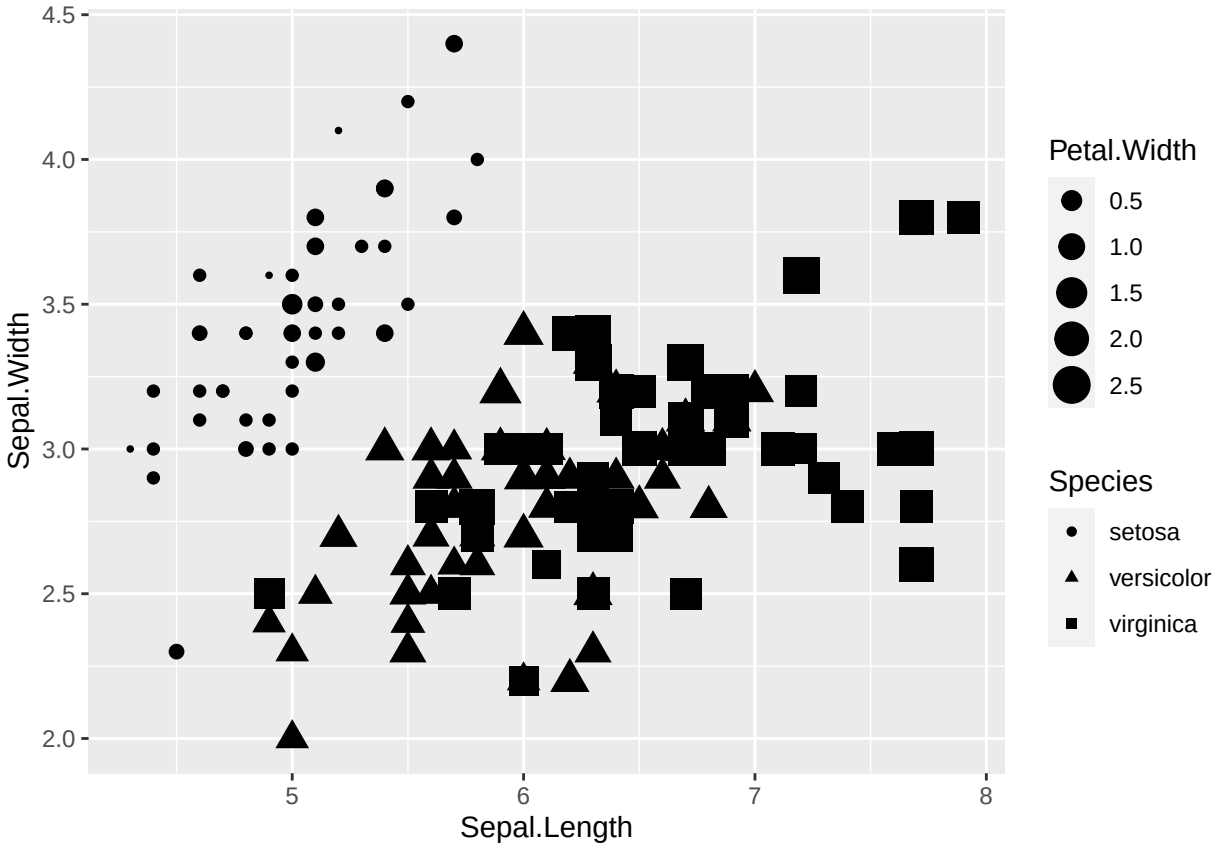
A Simple scatterplot but color coding of points based on species variable; also encode size

```
iris %>%
  ggplot(aes(x=Sepal.Length, y=Sepal.Width, color=Species)) +
  geom_point(aes(size=Petal.Length))
```



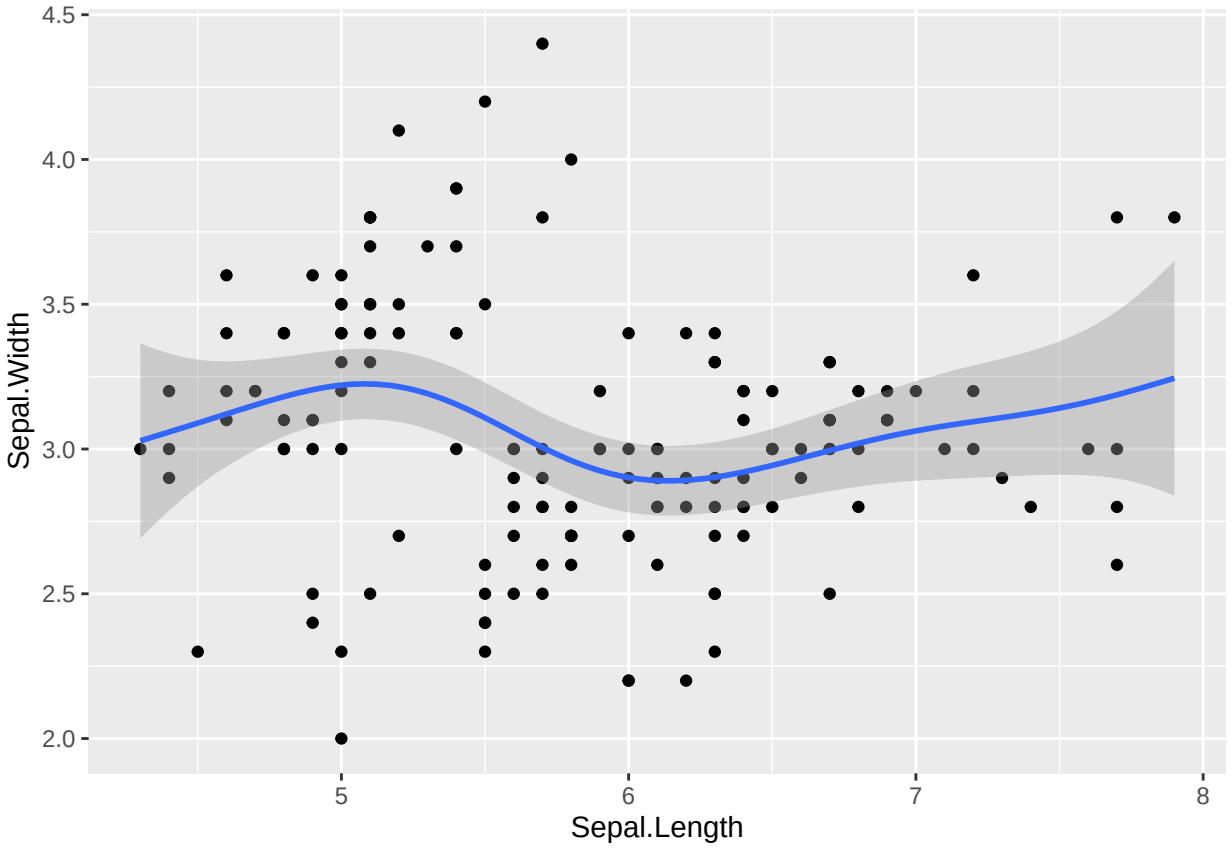
A Simple scatterplot but shape coding of points based on species variable

```
iris %>%
  ggplot(aes(x=Sepal.Length, y=Sepal.Width, shape=Species, size=Petal.Width)) +
  geom_point()
```



dimensions of data frame

```
iris %>%
  ggplot(aes(x=Sepal.Length, y=Sepal.Width)) +
  geom_point() +
  #stat_smooth(method="lm", formula=y~x+I(x^2))
  #stat_smooth(method="lm", formula=y~poly(x,2))
  #stat_smooth(method="loess", formula=y~x)
  stat_smooth(method="gam", formula=y~s(x), se=T)
```



Let's load the data which we output from the data cleaning notebook

```
fpath <- getwd() %>% print()
```

```
## [1] "/Users/kimwong/OneDrive - University of Pittsburgh/Documents/Kim F. Wong/CRC_Workshop/2021/Advar
```

generate timestamp

```
#dsdate  <- "_" %s+% format(Sys.Date(), "%Y%m%d") %T>% print()
dsdate <- "_20201104"
```

specify filename for dataset

```
dsname="cleaned_weatherAUS"
dsrdata <-
  file.path(fpath, dsname %s+% dsdate %s+% ".RData") %T>%
  file.path(fpath, dsname %s+% dsdate %s+% ".RData") %T>%
  print()
```

```
## [1] "/Users/kimwong/OneDrive - University of Pittsburgh/Documents/Kim F. Wong/CRC_Workshop/2021/Advar
```

Load R objects from file and list them

```
load(dsrdata) %>% print()
```

```
##  [1] "ds"        "dsname"    "dspath"    "dsdate"    "nobs"      "vars"
```

```
##  [7] "target"    "risk"      "id"        "ignore"    "omit"      "inputi"
## [13] "inputs"    "numi"      "numc"      "cati"      "catc"      "form"
## [19] "seed"      "train"     "validate"  "test"      "tr_target" "tr_risk"
## [25] "va_target" "va_risk"   "te_target" "te_risk"
```

Take a peak at the data

```
glimpse(ds)
```

```
## Rows: 173,890
## Columns: 27
## $ date            <date> 2008-12-01, 2008-12-02, 2008-12-03, 2008-12-04, 2008-~
## $ location        <fct> albury, albury, albury, albury, albury, albury, albury~
## $ min_temp        <dbl> 13.4, 7.4, 12.9, 9.2, 17.5, 14.6, 14.3, 7.7, 9.7, 13.1~
## $ max_temp        <dbl> 22.9, 25.1, 25.7, 28.0, 32.3, 29.7, 25.0, 26.7, 31.9, ~
## $ rainfall        <dbl> 0.6, 0.0, 0.0, 0.0, 1.0, 0.2, 0.0, 0.0, 0.0, 1.4, 0.0,~
## $ evaporation     <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~
## $ sunshine        <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~
## $ wind_gust_dir   <ord> w, wnw, wsw, ne, w, wnw, w, w, nnw, w, n, nne, w, sw, ~
## $ wind_gust_speed <dbl> 44, 44, 46, 24, 41, 56, 50, 35, 80, 28, 30, 31, 61, 44~
## $ wind_dir_9am    <ord> w, nnw, w, se, ene, w, sw, sse, se, s, sse, ne, nnw, w~
## $ wind_dir_3pm    <ord> wnw, wsw, wsw, e, nw, w, w, w, nw, sse, ese, ene, nnw,~
## $ wind_speed_9am  <dbl> 20, 4, 19, 11, 7, 19, 20, 6, 7, 15, 17, 15, 28, 24, NA~
## $ wind_speed_3pm  <dbl> 24, 22, 26, 9, 20, 24, 24, 17, 28, 11, 6, 13, 28, 20, ~
## $ humidity_9am    <dbl> 71, 44, 38, 45, 82, 55, 49, 48, 42, 58, 48, 89, 76, 65~
## $ humidity_3pm    <dbl> 22, 25, 30, 16, 33, 23, 19, 19, 9, 27, 22, 91, 93, 43,~
## $ pressure_9am    <dbl> 1007.7, 1010.6, 1007.6, 1017.6, 1010.8, 1009.2, 1009.6~
## $ pressure_3pm    <dbl> 1007.1, 1007.8, 1008.7, 1012.8, 1006.0, 1005.4, 1008.2~
## $ cloud_9am       <dbl> 8, NA, NA, NA, 7, NA, 1, NA, NA, NA, NA, 8, 8, NA, 0, ~
## $ cloud_3pm       <dbl> NA, NA, 2, NA, 8, NA, NA, NA, NA, NA, NA, 8, 8, 7, NA,~
## $ temp_9am        <dbl> 16.9, 17.2, 21.0, 18.1, 17.8, 20.6, 18.1, 16.3, 18.3, ~
## $ temp_3pm        <dbl> 21.8, 24.3, 23.2, 26.5, 29.7, 28.9, 24.6, 25.5, 30.2, ~
## $ rain_today      <fct> no, no, no, no, no, no, no, no, no, yes, no, yes, yes,~
## $ risk_mm         <dbl> 0.0, 0.0, 0.0, 1.0, 0.2, 0.0, 0.0, 0.0, 1.4, 0.0, 2.2,~
## $ rain_tomorrow   <fct> no, no, no, no, no, no, no, no, yes, no, yes, yes, yes~
## $ year            <fct> 2008, 2008, 2008, 2008, 2008, 2008, 2008, 2008, 2008, ~
## $ season          <fct> summer, summer, summer, summer, summer, summer, summer~
## $ cluster         <fct> area5, area5, area5, area5, area5, area5, area5, area5~
```

Count the number of missing values.

```
ds[vars] %>% is.na() %>% sum() %>% comcat()
```

```
## 403,758
```

Impute missing values.

```
ds[vars] %<>% na.roughfix()
```

Confirm that no missing values remain

```
ds[vars] %>% is.na() %>% sum() %>% comcat()
```

```
## 0
```

Take a peak at the data again

```
glimpse(ds)
```

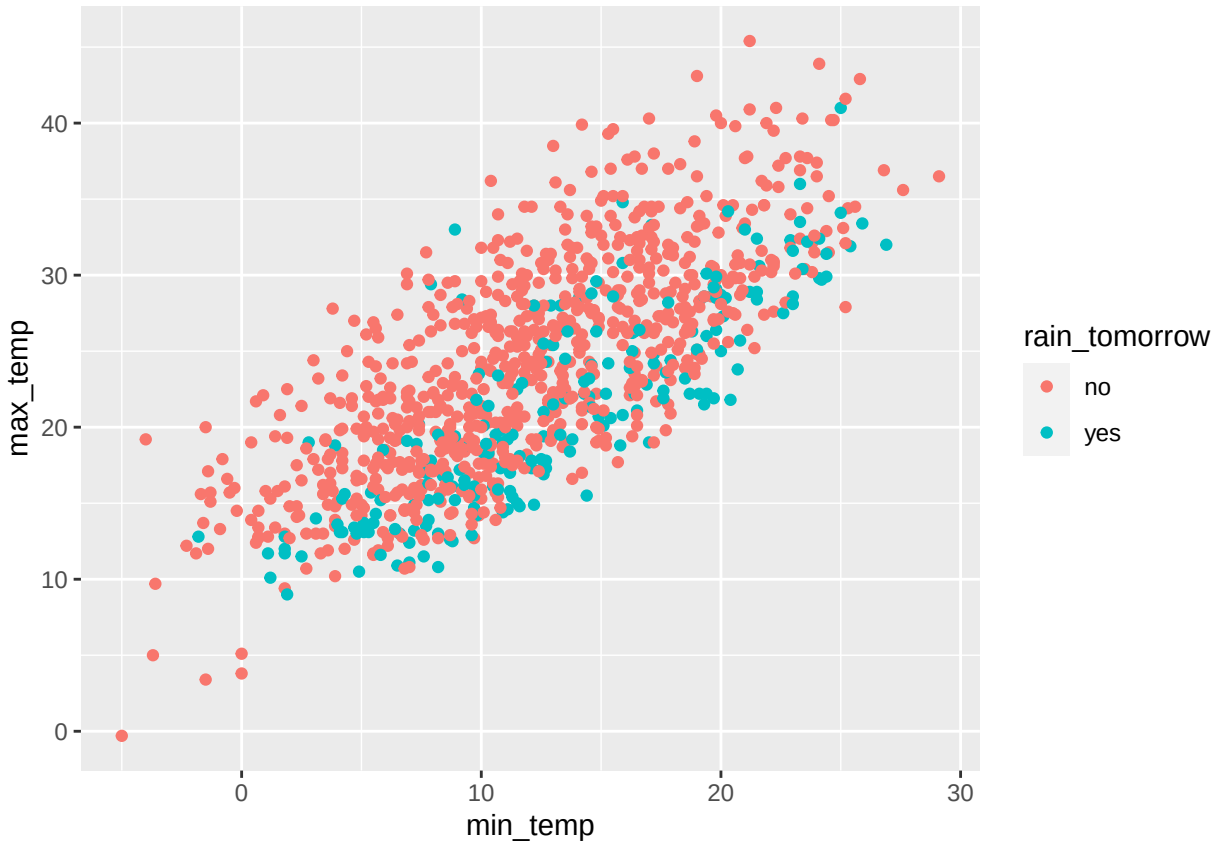
```
## Rows: 173,890
## Columns: 27
## $ date            <date> 2008-12-01, 2008-12-02, 2008-12-03, 2008-12-04, 2008-~
## $ location        <fct> albury, albury, albury, albury, albury, albury, albury~
## $ min_temp        <dbl> 13.4, 7.4, 12.9, 9.2, 17.5, 14.6, 14.3, 7.7, 9.7, 13.1~
## $ max_temp        <dbl> 22.9, 25.1, 25.7, 28.0, 32.3, 29.7, 25.0, 26.7, 31.9, ~
## $ rainfall        <dbl> 0.6, 0.0, 0.0, 0.0, 1.0, 0.2, 0.0, 0.0, 0.0, 1.4, 0.0,~
## $ evaporation     <dbl> 4.8, 4.8, 4.8, 4.8, 4.8, 4.8, 4.8, 4.8, 4.8, 4.8, 4.8,~
## $ sunshine        <dbl> 8.5, 8.5, 8.5, 8.5, 8.5, 8.5, 8.5, 8.5, 8.5, 8.5, 8.5,~
## $ wind_gust_dir   <ord> w, wnw, wsw, ne, w, wnw, w, w, nnw, w, n, nne, w, sw, ~
## $ wind_gust_speed <dbl> 44, 44, 46, 24, 41, 56, 50, 35, 80, 28, 30, 31, 61, 44~
## $ wind_dir_9am    <ord> w, nnw, w, se, ene, w, sw, sse, se, s, sse, ne, nnw, w~
## $ wind_dir_3pm    <ord> wnw, wsw, wsw, e, nw, w, w, w, nw, sse, ese, ene, nnw,~
## $ wind_speed_9am  <dbl> 20, 4, 19, 11, 7, 19, 20, 6, 7, 15, 17, 15, 28, 24, 13~
## $ wind_speed_3pm  <dbl> 24, 22, 26, 9, 20, 24, 24, 17, 28, 11, 6, 13, 28, 20, ~
## $ humidity_9am    <dbl> 71, 44, 38, 45, 82, 55, 49, 48, 42, 58, 48, 89, 76, 65~
## $ humidity_3pm    <dbl> 22, 25, 30, 16, 33, 23, 19, 19, 9, 27, 22, 91, 93, 43,~
## $ pressure_9am    <dbl> 1007.7, 1010.6, 1007.6, 1017.6, 1010.8, 1009.2, 1009.6~
## $ pressure_3pm    <dbl> 1007.1, 1007.8, 1008.7, 1012.8, 1006.0, 1005.4, 1008.2~
## $ cloud_9am       <dbl> 8, 5, 5, 5, 7, 5, 1, 5, 5, 5, 5, 8, 8, 5, 0, 8, 8, 5, ~
## $ cloud_3pm       <dbl> 5, 5, 2, 5, 8, 5, 5, 5, 5, 5, 5, 8, 8, 7, 5, 1, 1, 2, ~
## $ temp_9am        <dbl> 16.9, 17.2, 21.0, 18.1, 17.8, 20.6, 18.1, 16.3, 18.3, ~
## $ temp_3pm        <dbl> 21.8, 24.3, 23.2, 26.5, 29.7, 28.9, 24.6, 25.5, 30.2, ~
## $ rain_today      <fct> no, no, no, no, no, no, no, no, no, yes, no, yes, yes,~
## $ risk_mm         <dbl> 0.0, 0.0, 0.0, 1.0, 0.2, 0.0, 0.0, 0.0, 1.4, 0.0, 2.2,~
## $ rain_tomorrow   <fct> no, no, no, no, no, no, no, no, yes, no, yes, yes, yes~
## $ year            <fct> 2008, 2008, 2008, 2008, 2008, 2008, 2008, 2008, 2008, ~
## $ season          <fct> summer, summer, summer, summer, summer, summer, summer~
## $ cluster         <fct> area5, area5, area5, area5, area5, area5, area5, area5~
```

Randomly pull out a sample of 1000 observations from data set

```
ds %>% nrow() %>% sample(1000) -> sobs
```

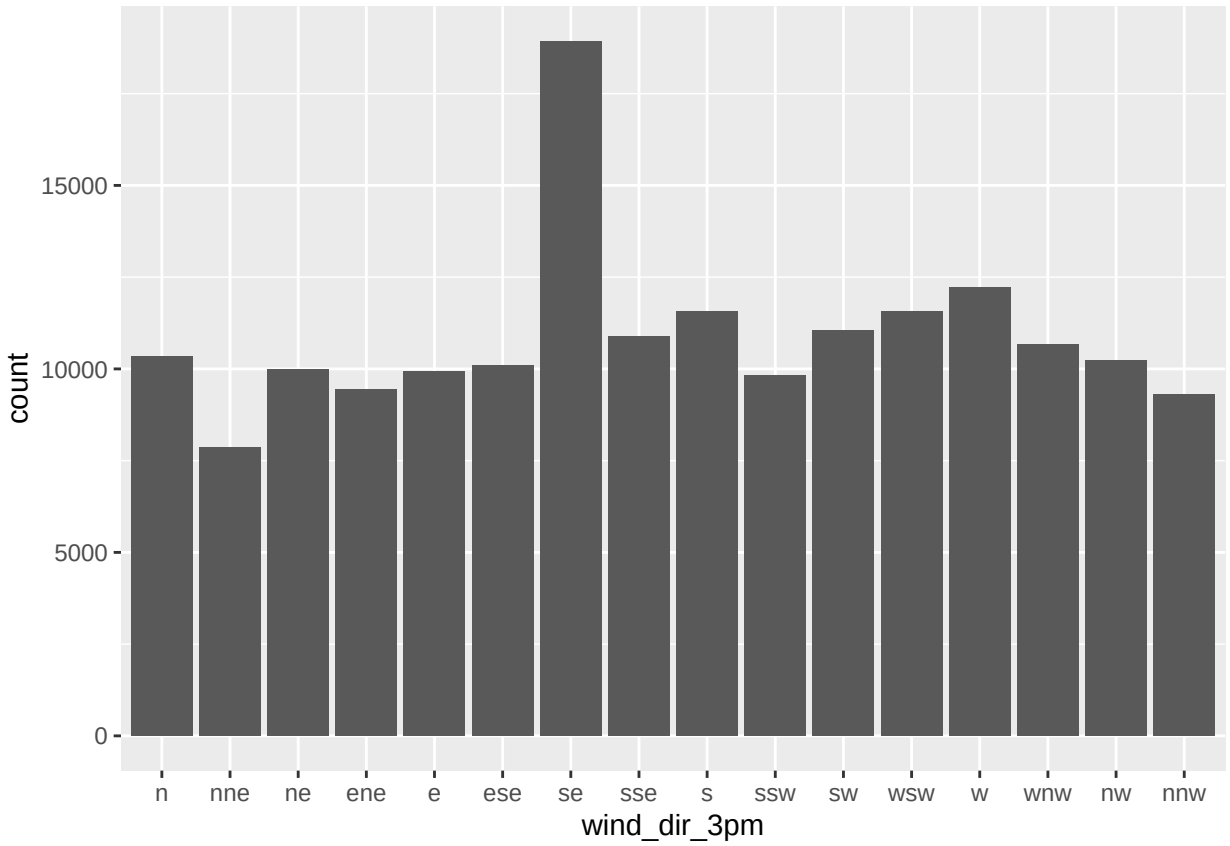
Do a scatterplot of the min and max temperatures and encode if it will rain tomorrow in the color

```
ds %>% extract(sobs,) %>%
  ggplot(aes(x=min_temp, y=max_temp, color=rain_tomorrow)) +
  geom_point()
```



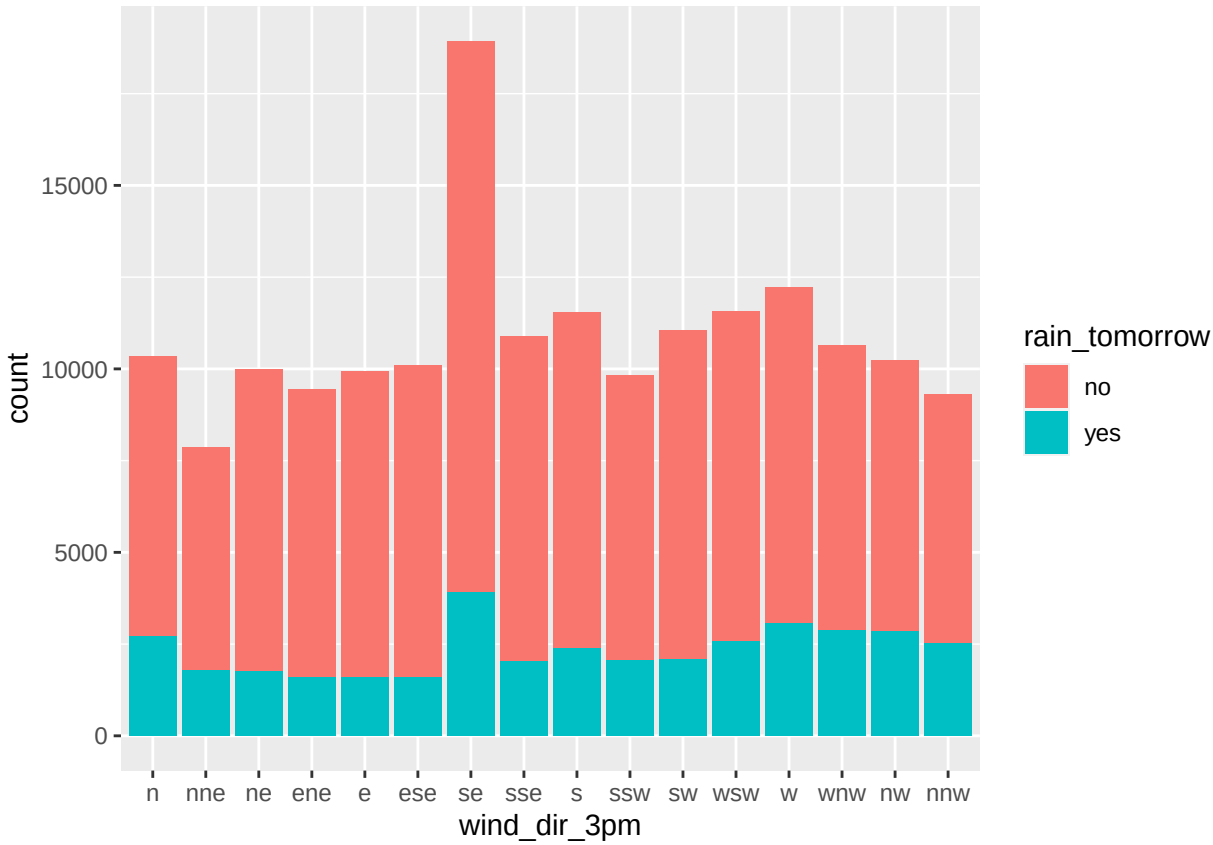
Standard bar chart.

```
ds %>% ggplot(aes(x=wind_dir_3pm)) + geom_bar()
```



Standard bar chart with binary outcome encoded in the color.

```
ds %>% ggplot(aes(x=wind_dir_3pm, fill=rain_tomorrow)) +
  geom_bar()
```



Define two shades of blue using ColorBrewer, http://colorbrewer2.org.

```
library(RColorBrewer)
blues2 <- brewer.pal(4,"Paired")[1:2] %T>% print()
```

```
## [1] "#A6CEE3" "#1F78B4"
```

Output the number of weather stations from data set

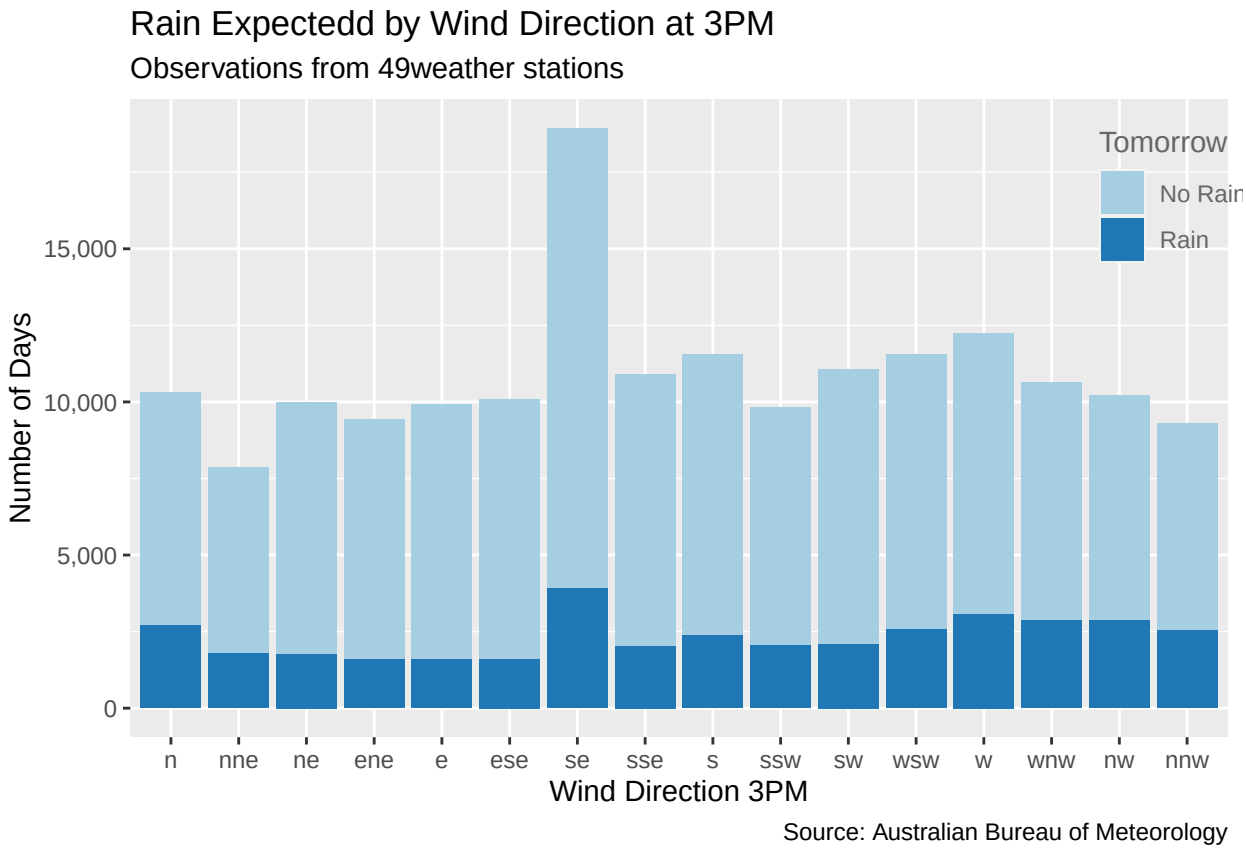
```
ds$location %>%
  unique() %>%
  length() %>%
  print() ->
  num_locations
```

```
## [1] 49
```

Generate a bar chart with more complex annotation

```
ds %>%
  ggplot(aes(x=wind_dir_3pm, fill=rain_tomorrow)) +
  geom_bar() +
  scale_fill_manual(values = blues2,
                    labels = c("No Rain", "Rain")) +
  scale_y_continuous(labels = comma) +
  theme(legend.position = c(0.95, 0.85),
```

```
        legend.title      = element_text(color="grey40"),
        legend.text       = element_text(color="grey40"),
        legend.background = element_rect(fill="transparent")) +
  labs(title    = "Rain Expectedd by Wind Direction at 3PM",
       subtitle = "Observations from " %s+%
                   num_locations %s+% "weather stations",
       caption  = "Source: Australian Bureau of Meteorology",
       x        = "Wind Direction 3PM",
       y        = "Number of Days",
       fill     = "Tomorrow")
```



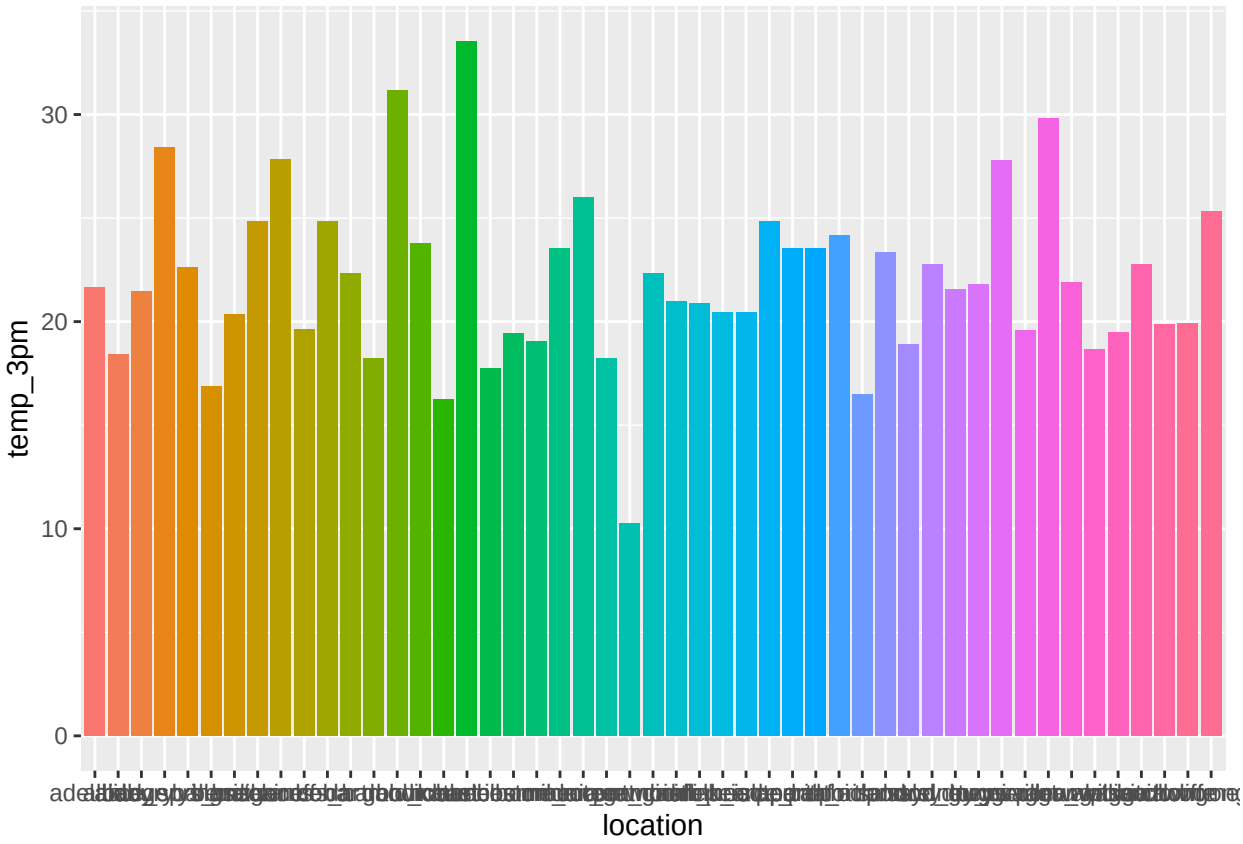
Bar chart of mean temperature at 3PM across all 49 weather stations

```
ds %>%
  ggplot(aes(x=location, y=temp_3pm, fill=location)) +
  geom_bar(stat="summary", fun.y="mean") +
  theme(legend.position="none")
```

```
## Warning: Ignoring unknown parameters: fun.y
```

```
## Warning: Removed 4157 rows containing non-finite values (stat_summary).
```

```
## No summary function supplied, defaulting to `mean_se()`
```



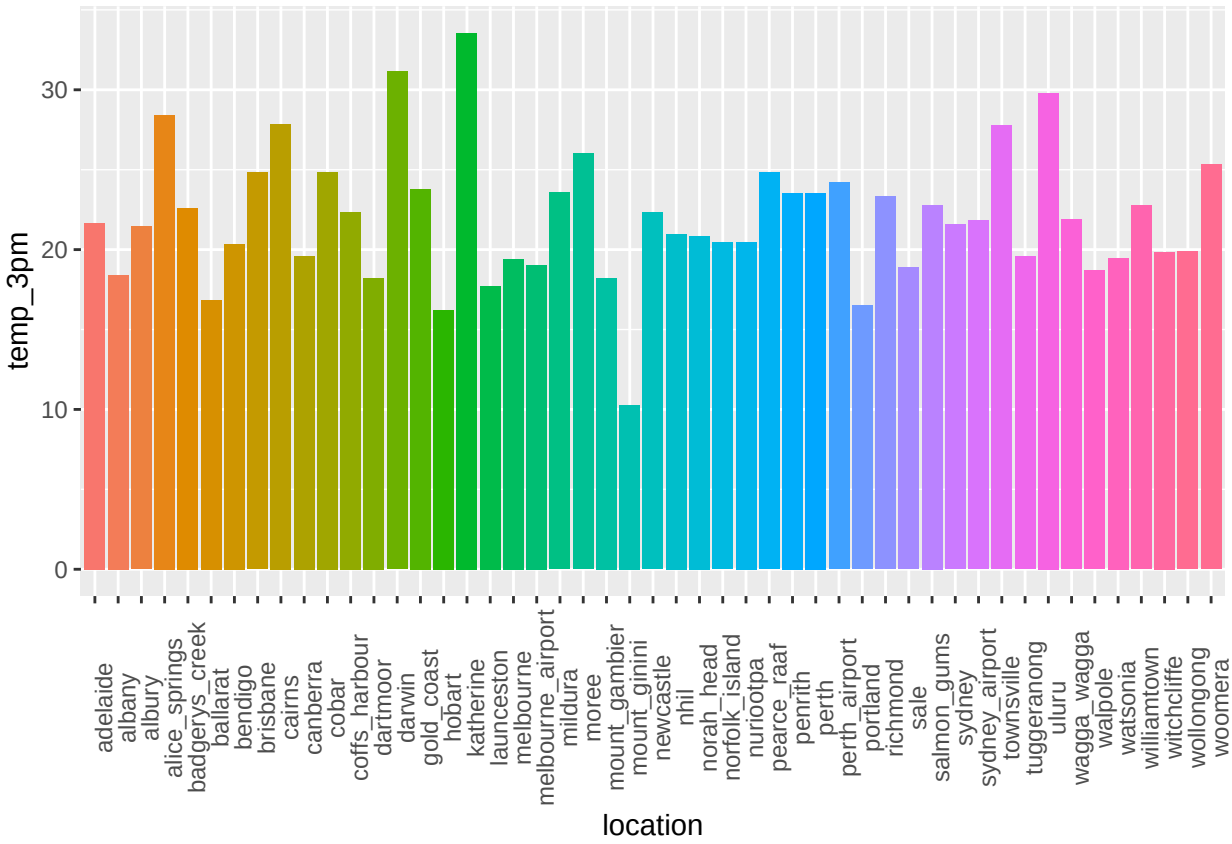
Since the above plot has all labeled weather stations cluttered together, let's rotate the labels by 90 degrees.

```
ds %>%
  ggplot(aes(x=location, y=temp_3pm, fill=location)) +
  geom_bar(stat="summary", fun.y="mean") +
  theme(legend.position="none") +
  theme(axis.text.x = element_text(angle=90))
```

```
## Warning: Ignoring unknown parameters: fun.y
```

```
## Warning: Removed 4157 rows containing non-finite values (stat_summary).
```

```
## No summary function supplied, defaulting to `mean_se()`
```



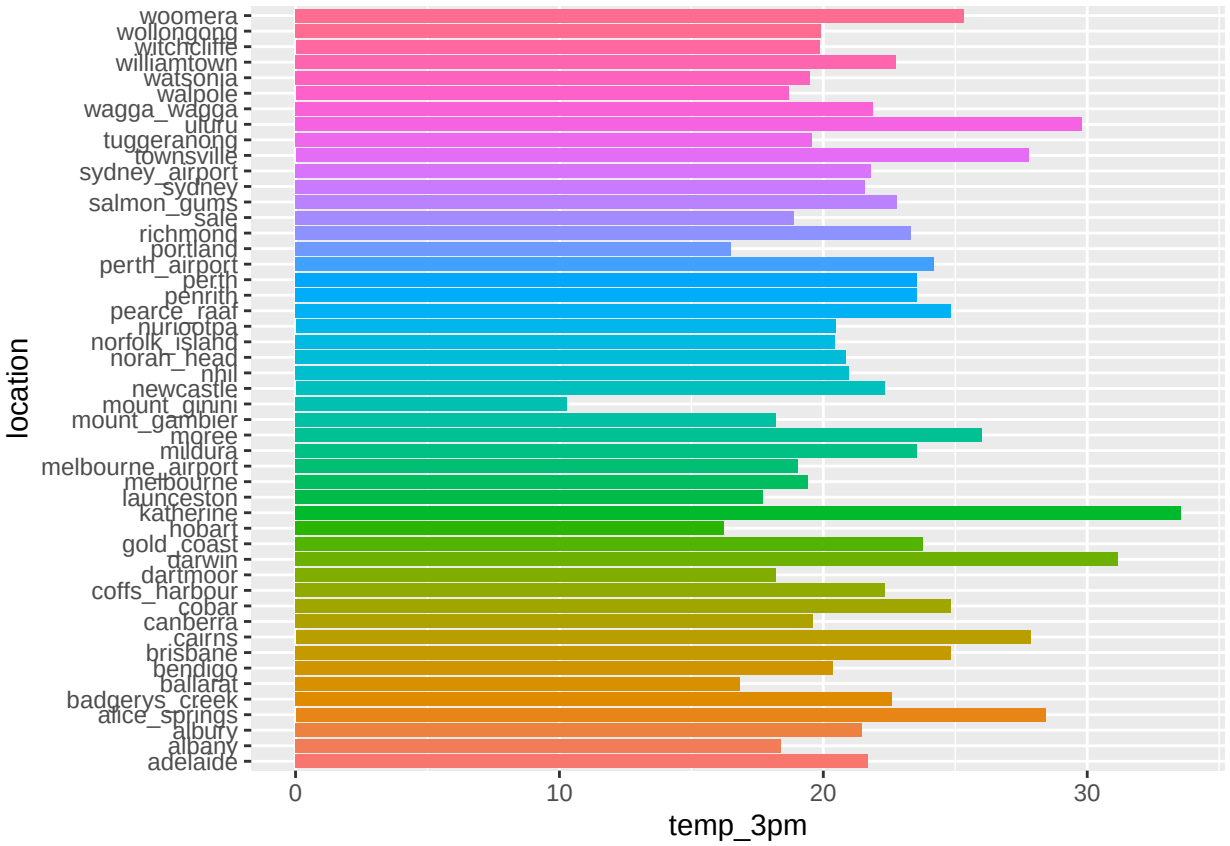
Instead of rotating the labels by 90 degrees, we can flip the x-y coordinates.

```
ds %>%
  ggplot(aes(x=location, y=temp_3pm, fill=location)) +
  geom_bar(stat="summary", fun.y="mean") +
  theme(legend.position="none") +
  #theme(axis.text.x = element_text(angle=90)) +
  coord_flip()
```

```
## Warning: Ignoring unknown parameters: fun.y
```

```
## Warning: Removed 4157 rows containing non-finite values (stat_summary).
```

```
## No summary function supplied, defaulting to `mean_se()`
```



We want to reorder the location in alphabetical order instead of the reverse

```
ds$location %>%
  levels() %>%
  rev() ->
  loc
```

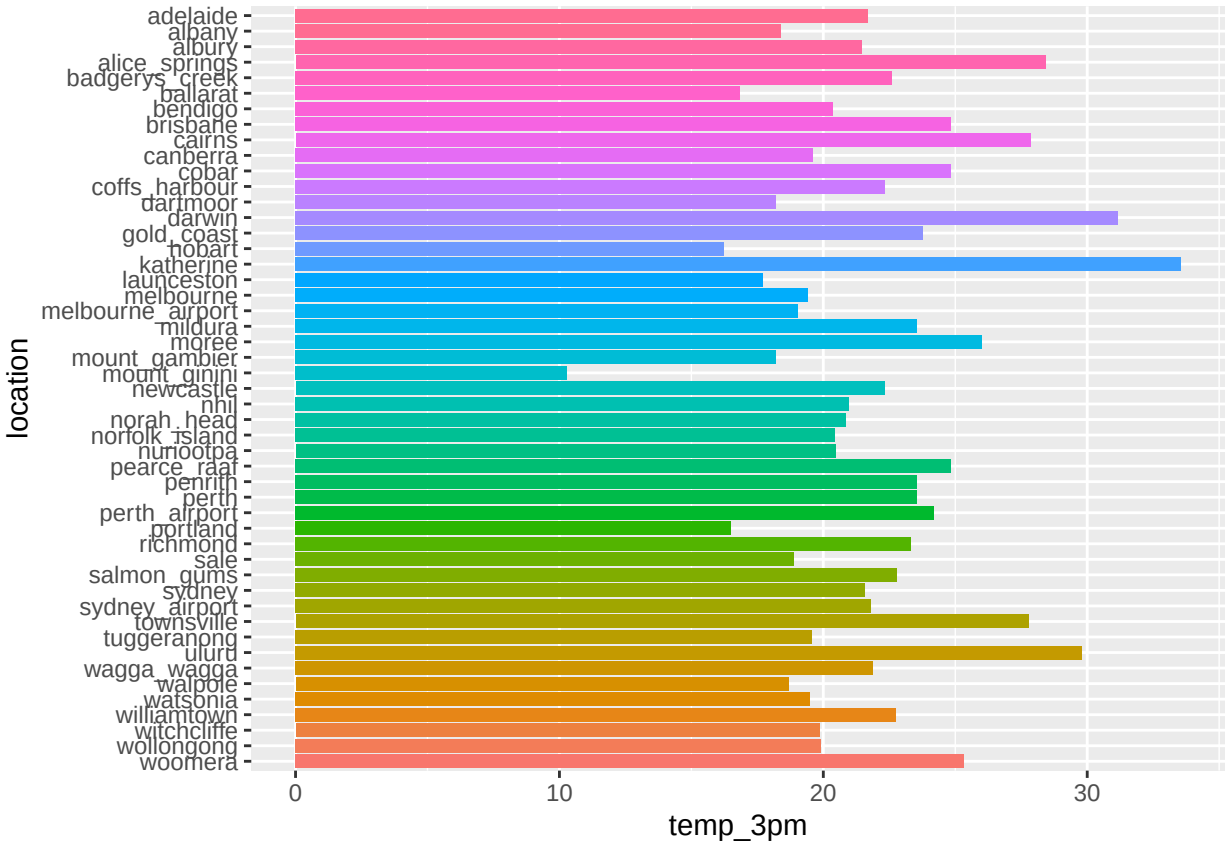
Use mutate within dplyr to temporarily place the location in alphabetical order

```
ds %>%
  mutate(location=factor(location,levels=loc)) %>%
  ggplot(aes(x=location, y=temp_3pm, fill=location)) +
  geom_bar(stat="summary", fun.y="mean") +
  theme(legend.position="none") +
  #theme(axis.text.x = element_text(angle=90)) +
  coord_flip()
```

```
## Warning: Ignoring unknown parameters: fun.y
```

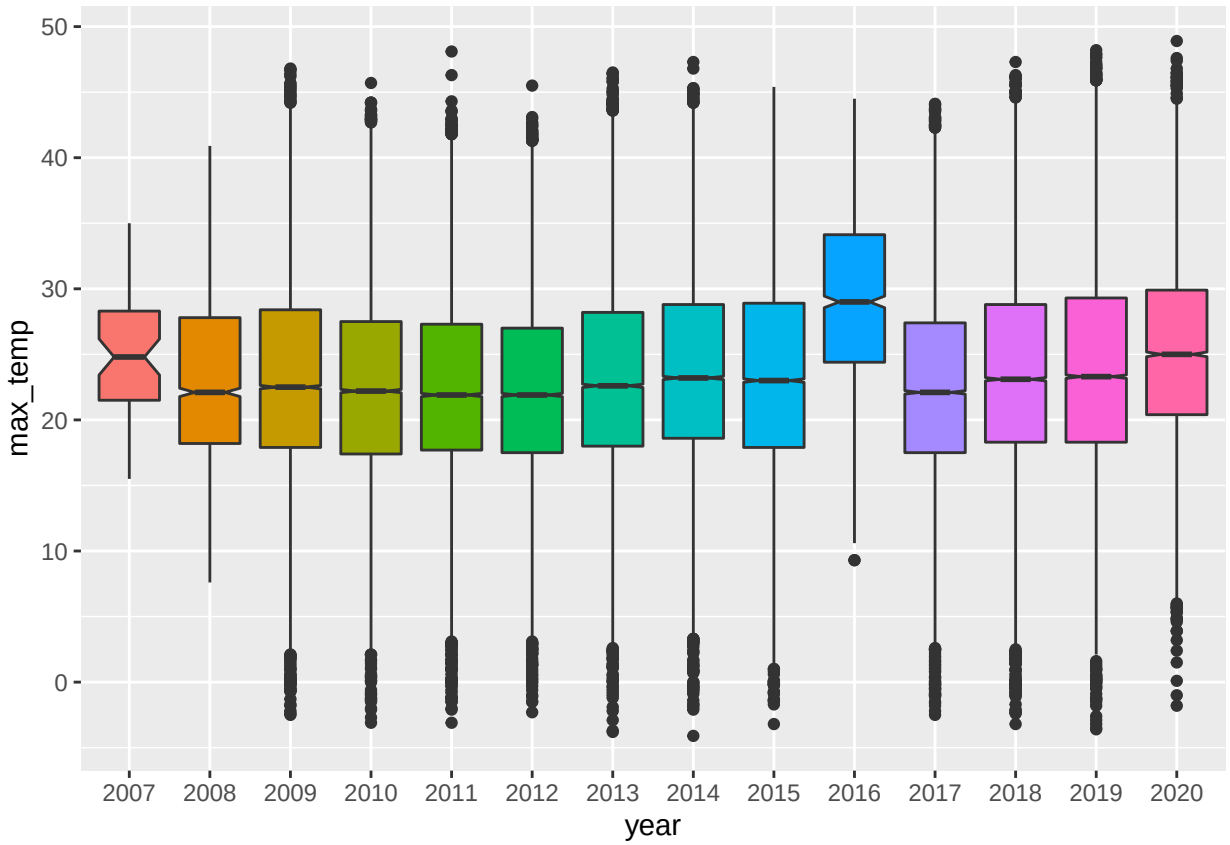
```
## Warning: Removed 4157 rows containing non-finite values (stat_summary).
```

```
## No summary function supplied, defaulting to `mean_se()`
```



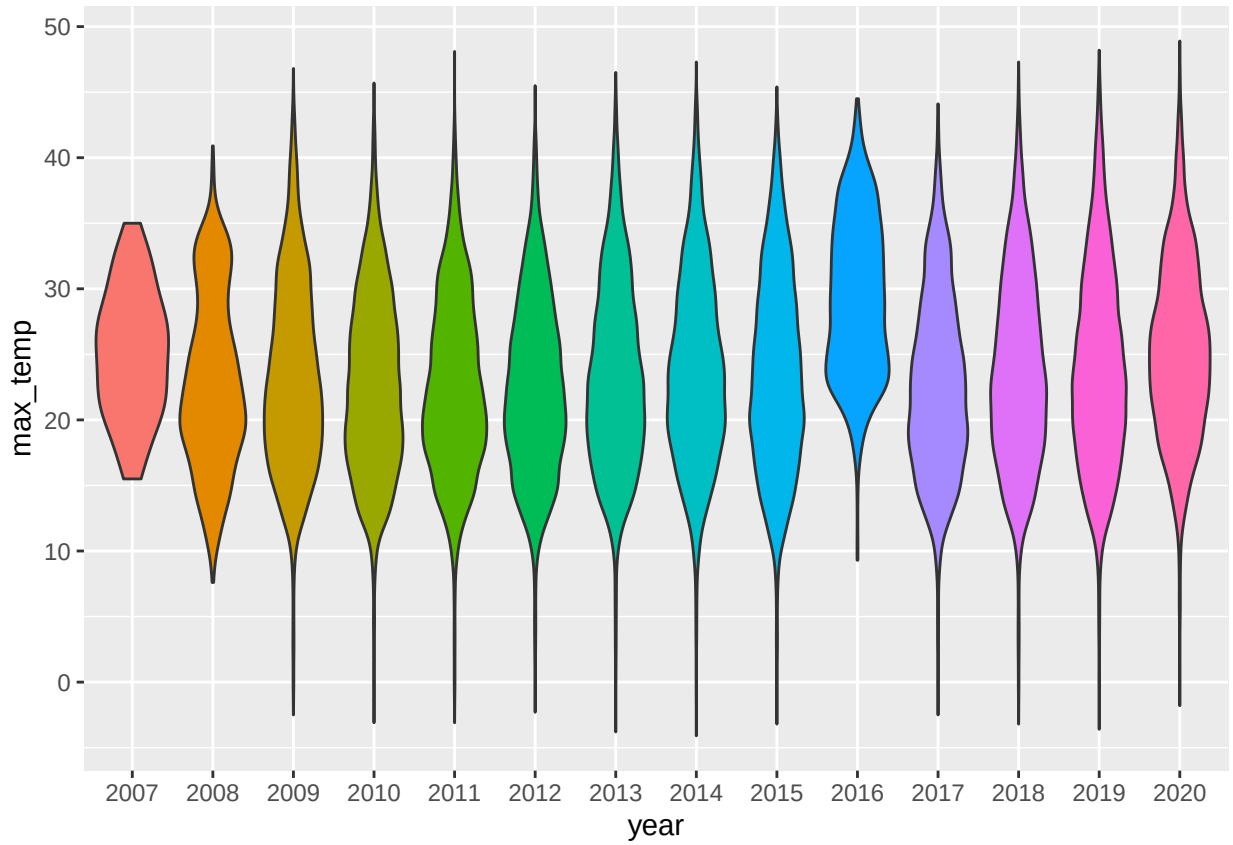
Output box plot of maximum temperature as a parameter of year

```
ds %>%
  ggplot(aes(year, max_temp, fill=year)) +
  geom_boxplot(notch = T) +
  theme(legend.position="none")
```



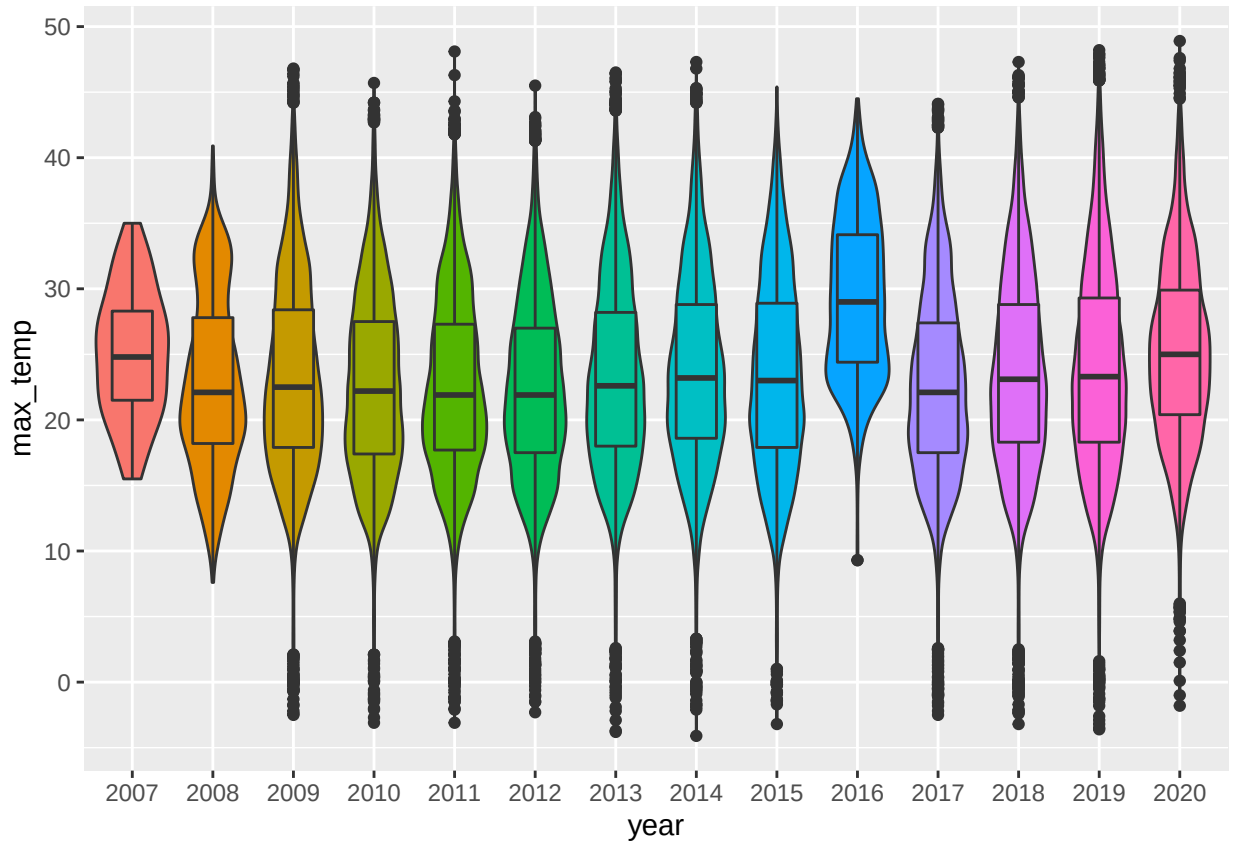
Output violin plot of maximum temperature as a parameter of year

```
ds %>%
  ggplot(aes(year, max_temp, fill=year)) +
  geom_violin() +
  theme(legend.position="none")
```



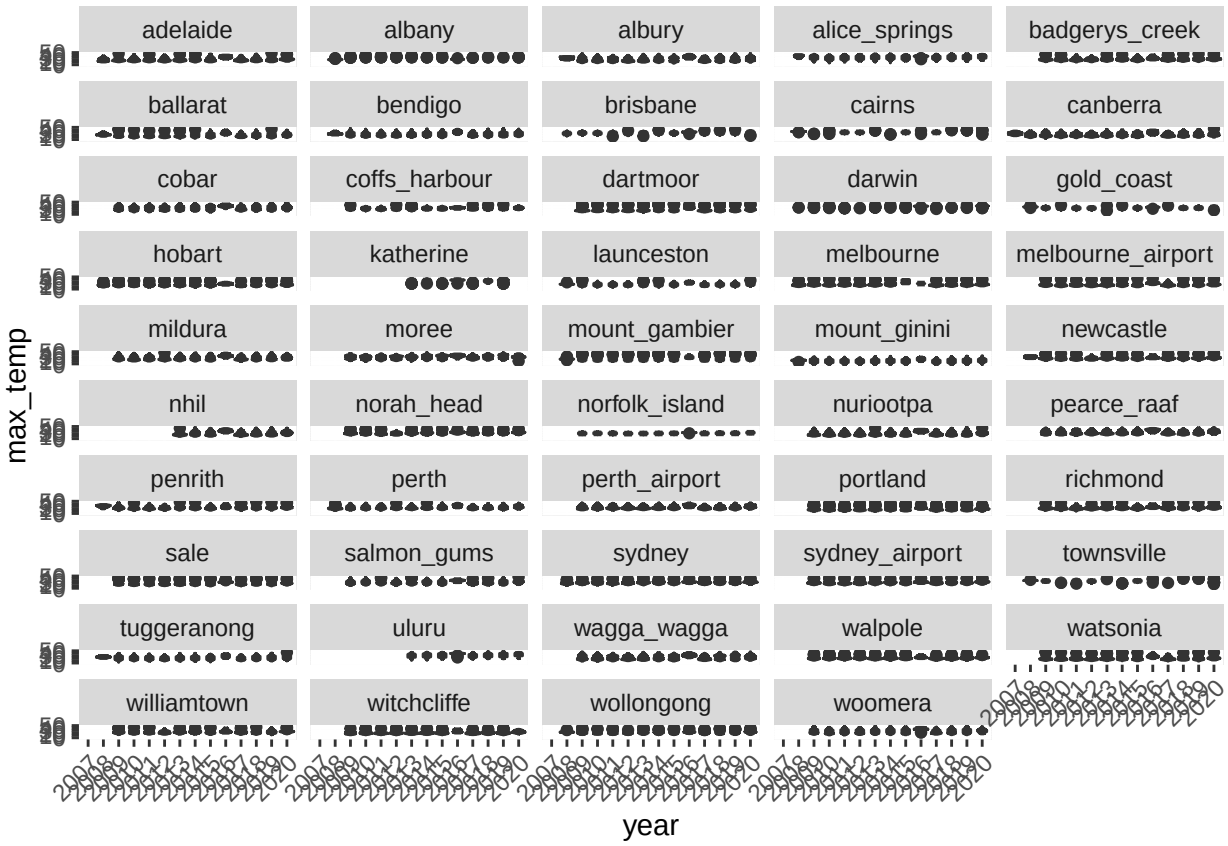
Overlay box plot and violin plot of maximum temperature as a parameter of year

```
ds %>%
  ggplot(aes(year, max_temp, fill=year)) +
  geom_violin() +
  geom_boxplot(width=0.5, position = position_dodge(width=0)) +
  theme(legend.position="none")
```



Use facet_wrap to split the plots across locations

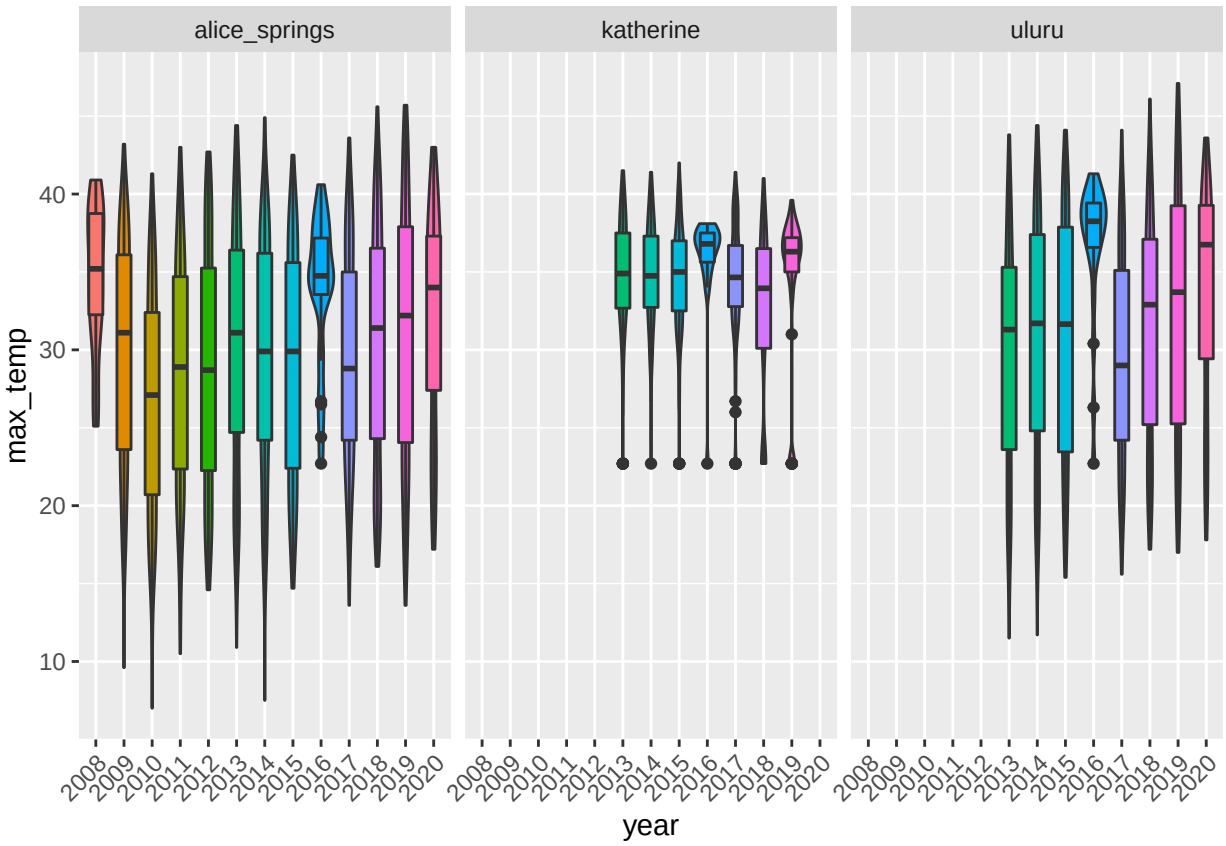
```
ds %>%
  ggplot(aes(x=year, y=max_temp, fill=year)) +
  geom_violin() +
  geom_boxplot(width=0.5, position=position_dodge(width=0)) +
  theme(legend.position="none") +
  theme(axis.text.x=element_text(angle=45, hjust=1)) +
  facet_wrap(~location, ncol=5)
```



Create a helper funtion so as to visualize clusters of location

```
myplot <- function(ds, n)
{
ds %>%
  filter(cluster==n) %>%
  ggplot(aes(x=year, y=max_temp, fill=year)) +
  geom_violin() +
  geom_boxplot(width=0.5, position=position_dodge(width=0)) +
  theme(legend.position="none") +
  theme(axis.text.x=element_text(angle=45, hjust=1)) +
  facet_wrap(~location, ncol=5)
}
```

```
myplot(ds,"area3")
```

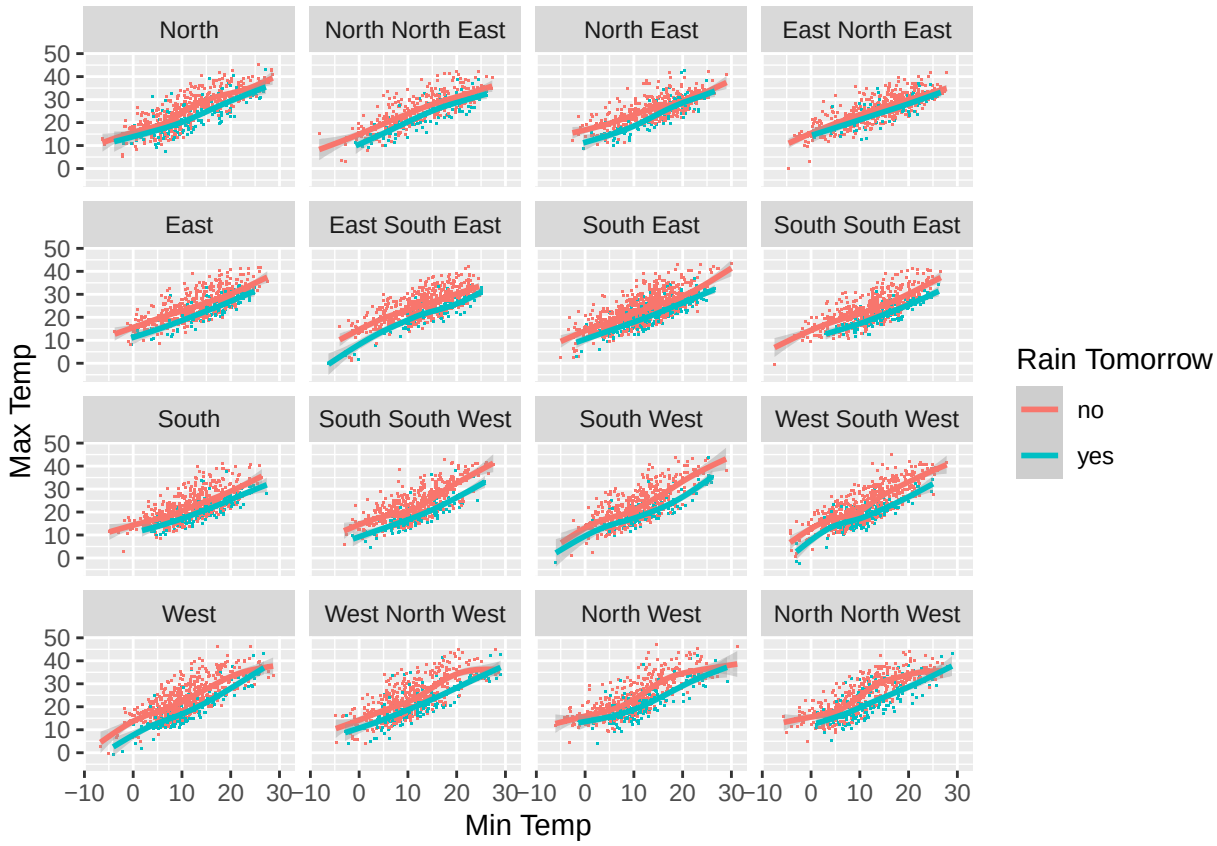


convert variables from character to factor class

```
lblr <- function(x)
{
  x %>%
    str_replace_all("n", "North ") %>%
    str_replace_all("s", "South ") %>%
    str_replace_all("e", "East ") %>%
    str_replace_all("w", "West ") %>%
    str_replace(" $", "")
}
```

convert wind direction to factor

```
ds %>%
  sample_n(10000) %>%
  ggplot(aes(x=min_temp, y=max_temp, colour=rain_tomorrow)) +
  geom_point(shape=".") +
  geom_smooth(method="gam", formula=y~s(x, bs="cs")) +
  facet_wrap(~wind_dir_3pm, labeller=labeller(wind_dir_3pm=lblr)) +
  labs(x      = "Min Temp",
       y      = "Max Temp",
       colour = "Rain Tomorrow")
```


North
North North East
North East
East North East
East
East South East
South East
South South East
South
South South West
South West
West South West
West
West North West
North West
North North West
Max Temp
Min Temp
Rain Tomorrow
no
yes