Lab 5 Assessment Solution

Tom Palmer

26th October 2017

```
# ===============================================================================
# MATH550/SCC461 Statistical Programming Using R
# Lab 5 Assessment Solution
#
# ===============================================================================
library(ggplot2)
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(stringr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

```
setwd("H:\\all\\teaching\\Math550 Stats in Practice\\R\\MATH550\\R_course\\Lab5\\Coursework")

# ============================================================
# 1. Setting up the data
# ============================================================

# ===============================================================================
# load the data into R
# ===============================================================================
storm <- read.csv("Australia_severe_storms_1975-2015.csv",
                  stringsAsFactors=FALSE)

# ===============================================================================
# clean
# ===============================================================================

# parse timestamp, convert 'State' to categorical

storm <- mutate(storm,
                Database = factor(Database),
                Date.Time = dmy_hm(Date.Time),
                State    = factor(State)
)

sapply(storm, class)
```

```
## $Event.ID
## [1] "integer"
##
## $Database
## [1] "factor"
##
## $ID
## [1] "integer"
##
## $Date.Time
## [1] "POSIXct" "POSIXt"
##
## $Nearest.town
## [1] "character"
##
## $State
## [1] "factor"
##
## $Latitude
## [1] "numeric"
##
## $Longitude
## [1] "numeric"
##
## $Comments
## [1] "character"
##
## $X
## [1] "character"
##
## $X.1
## [1] "character"
##
## $X.2
## [1] "character"
##
## $X.3
## [1] "character"
##
## $X.4
## [1] "character"
```

```
# Combine the comments from columns ( Comments, X, X.1, X.2,
# X.3, X.4 ) into a single column containing comments. Call the
# column All.comments
storm$All.comments <- paste(storm$Comments, storm$X, storm$X.1,
                  storm$X.2, storm$X.3, storm$X.4, sep=" ")

# Select the following columns to keep for further analysis, Event.ID,
# Database, Date.Time, State, All.comments and make sure all
# variables of the appropriate type.
storm <- select(storm, Event.ID, Database, Date.Time, State, All.comments)

# run the following command
print(sapply(storm, class))
```

```
## $Event.ID
## [1] "integer"
##
## $Database
## [1] "factor"
##
## $Date.Time
## [1] "POSIXct" "POSIXt"
##
## $State
## [1] "factor"
##
## $All.comments
## [1] "character"
```

```
# ==========================================================
# 2. Extracts flash floods
# ==========================================================

# Create an indicator variable which states whether or not a storm
# event has resulted in a flash flood.
# Hint: Make sure you sort out all terms relating to flash floods.

# Expression for flash floods
# Alternative regular expressions for "flash floods":
# expr <- "\\b[fF]lash flood(s|ing)?\\b" # 840
# expr <- "\\b[fF]lash [Ff]lood(s|ing)?\\b" # 849
# expr <- "\\b[fF](a)?l(a|o)a?sh [fF]lood(s|ing)?\\b" # 852
# expr <- "\\b[fF](a)?l[a-z]*\\s[fF]l[a-z]*\\b" # 853
expr <- "\\b[fF]([a-z])?l[a-z]*\\s[fF]([a-z])?l[a-z]*\\b"
storm$is_flash <- str_detect(storm$All.comments, expr) # 854

# Print number of flash flood events
print(sum(storm$is_flash))
```

```
## [1] 854
```

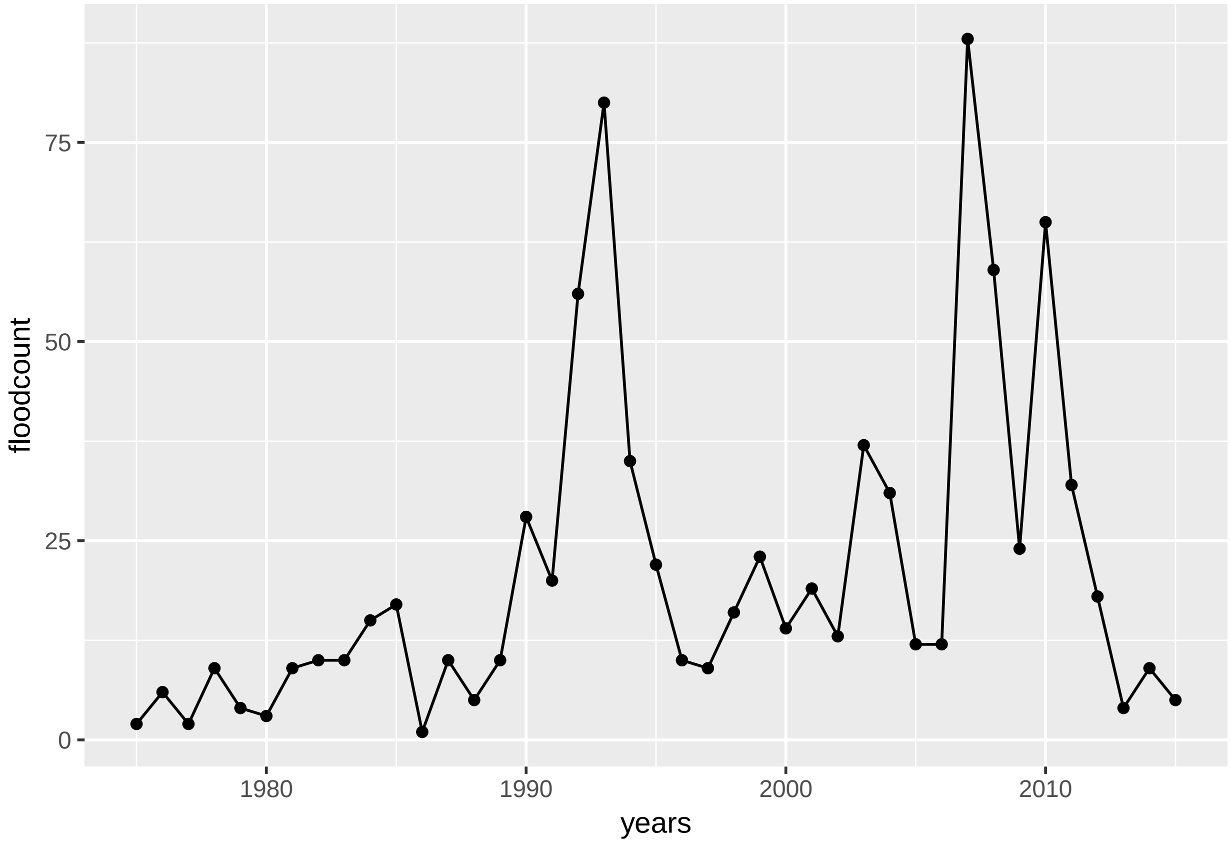
```
# Print a plot of the number of flash floods per year from 1975-2015.
# Hint: Create a vector to contain the number of flash floods per year.

# Year of event
storm$year <- year(storm$Date.Time)

# Sequence of years from 1975 to 2015
yearsdat <- data.frame(years=1975:2015, floodcount=NA)

for(i in 1:nrow(yearsdat)) {
  yearsdat$floodcount[i] = sum(storm$is_flash[storm$year == yearsdat$years[i]])
}

# Plot of flash floods
ggplot(yearsdat, aes(x=years, y=floodcount)) + geom_point() + geom_line()
```



```
# ==============================================================
# 3. Wind speed
# =============================================================

# For severe wind events often the wind speed is given. The wind
# speed is given in knots or km/h.

# Extract all wind speeds both those in knots and km/h.
# Hint: Knots can be abbreviated by kts or kt. Also note that wind
# speed can be a single, double or triple digit number.

# Convert kt/kts to knots
# expr_kts <- "\\b(kt(s))\\b"
# expr_kts <- "([0-9]*kt(s)?)|(\\b(kt(s)?))\\b|(knots(s)?)\\b"
expr_kts <- "((\\b)?([Kk][Tt](s)?))|(\\b)?([Kk]not(s)?)"
storm$All.comments <- str_replace(storm$All.comments, expr_kts, "knots")

# Extract out 1, 2, or 3 digit wind speeds
expr_speed <- "\\b([0-9]?[0-9]?[0-9](\\s)?km/h)|([0-9]?[0-9]?[0-9](\\s)?knot(s)?)\\b"
storm$wind_speed <- str_extract(storm$All.comments, expr_speed)
```

```r
# filter(storm, All.comments!="") %>% View()

# Extract out the speed as a number and
# indicator for whether or not the speed is in km/h
expr_digits_speed <- "(\\b)?([0-9]?[0-9]?[0-9])(\\b)?"
storm$speed <- as.numeric(str_extract(storm$wind_speed, expr_digits_speed))
expr_km <- "(km/h(r)?)"
storm$is_km <- str_detect(storm$wind_speed, expr_km)

# filter(storm, wind_speed != "") %>% View()

dim(storm)
```

```
## [1] 14457    10
```

```r
# Select only rows with windspeeds
windspeeds <- filter(storm, is_km==TRUE | is_km==FALSE)
dim(windspeeds)
```

```
## [1] 1222   10
```

```r
# Convert km/h windspeeds to knots (1 knot = 1.852 km/h) and round to the
# nearest integer
windspeeds$speed[windspeeds$is_km==TRUE] <-
     round(windspeeds$speed[windspeeds$is_km==TRUE]/1.852, 0)

# check values of State
table(windspeeds$State)
```

```
##
##     NSW  NT QLD  SA TAS VIC  WA
##   1 175 106 272 332  13 130 193
```

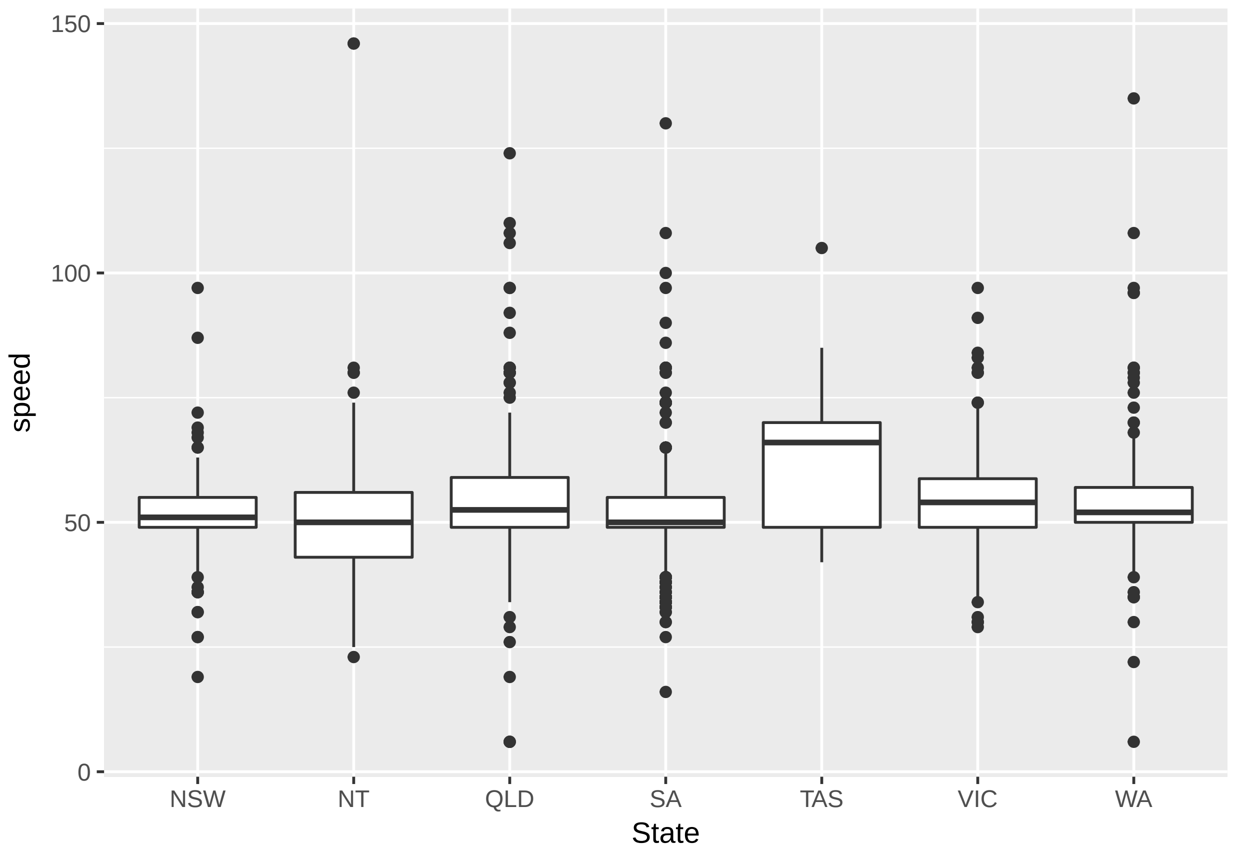
```r
levels(windspeeds$State) # 1 empty one
```

```
## [1] ""    "NSW" "NT"  "QLD" "SA"  "TAS" "VIC" "WA"
```

```r
windspeeds <- filter(windspeeds, State != "")
dim(windspeeds)
```

```
## [1] 1221   10
```

```r
# Print a boxplot of windspeeds by state
p1 <- ggplot(windspeeds, aes(x=State, y=speed)) + geom_boxplot()
p1
```


150
100
50
0
speed
NSW
NT
QLD
SA
TAS
VIC
WA
State