README

Samantha Scott

17/06/2022

Question 1

Code

```
gc()
```

```
##          used (Mb) gc trigger (Mb) limit (Mb) max used (Mb)
## Ncells 463347 24.8     989392 52.9         NA   668911 35.8
## Vcells 865289  6.7    8388608 64.0      16384  1838985 14.1
```

```
library(pacman)
p_load(tidyverse, lubridate)

list.files('Question1/Code', full.names = T, recursive = T) %>% as.list() %>% walk(~source(.))
```

Loading Data

```
 library(readr)
 Deaths_by_cause <- read_csv("Question1/Data/Covid/Deaths_by_cause.csv")
```

```
## Rows: 7273 Columns: 36
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): Entity, Code, Number of executions (Amnesty International)
## dbl (33): Year, Deaths - Meningitis - Sex: Both - Age: All Ages (Number), De...
##
## 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.
```

```
library(readr)
owid_covid_data <- read_csv("Question1/Data/Covid/owid-covid-data.csv")
```

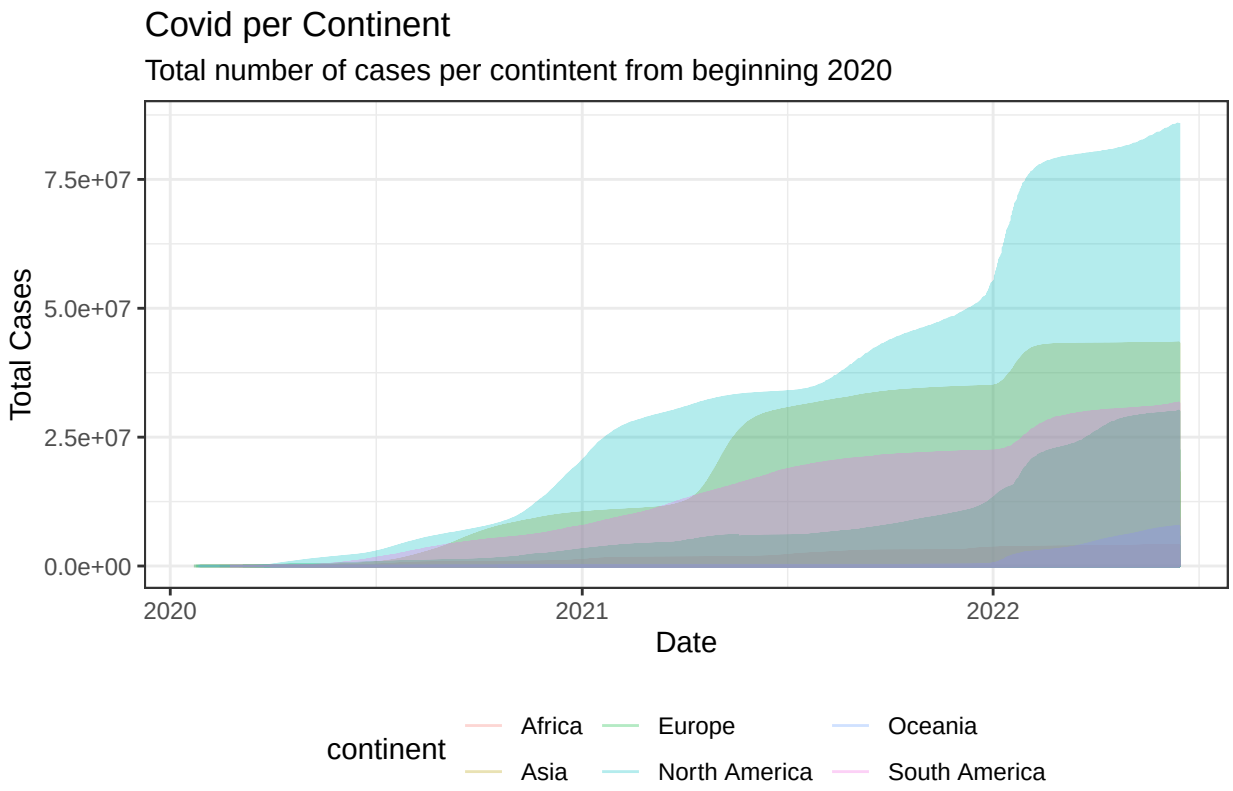
```
## Rows: 194260 Columns: 67
## -- Column specification --------------------------------------------------------
## Delimiter: ","
```

```
## chr   (4): iso_code, continent, location, tests_units
## dbl  (62): total_cases, new_cases, new_cases_smoothed, total_deaths, new_dea...
## date  (1): date
##
## 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.
```

Graph 1.1

```
g <- line_graph_continents(owid_covid_data)
```

```
## Adding missing grouping variables: `continent`
```



Note:OWID data used

As seen in the graph, the continents with the highest number of total cases is Europe and North America. African countries are seen to have not been as greatly by COVID than North American, South American and European countries. In comparison, the number of total countries in Africa seem almost insignificant in comparison to countries in other continents.

However, to accurately interpret this graph, it is imperative that one take in to consideration the population of the continents.

Graph 1.2

```
g <- brics_SI_deaths(owid_covid_data)
```

```
## `summarise()` has grouped output by 'location'. You can override using the
## `.groups` argument.
```

Question 2

Code

```
gc()
```

```
##            used  (Mb) gc trigger  (Mb) limit (Mb) max used  (Mb)
## Ncells  1317685  70.4    2167555 115.8         NA  2167555 115.8
## Vcells 15616928 119.2   31217699 238.2      16384 31110389 237.4
```

```
library(pacman)
p_load(tidyverse, lubridate)

list.files('Question2/Code', full.names = T, recursive = T) %>% as.list() %>% walk(~source(.))
```

Loading Data

```
library(readr)
london_weather <- read_csv("Question2/Data/London/london_weather.csv")
```

```
## Rows: 15341 Columns: 10
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (10): date, cloud_cover, sunshine, global_radiation, max_temp, mean_temp...
##
## 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.
```

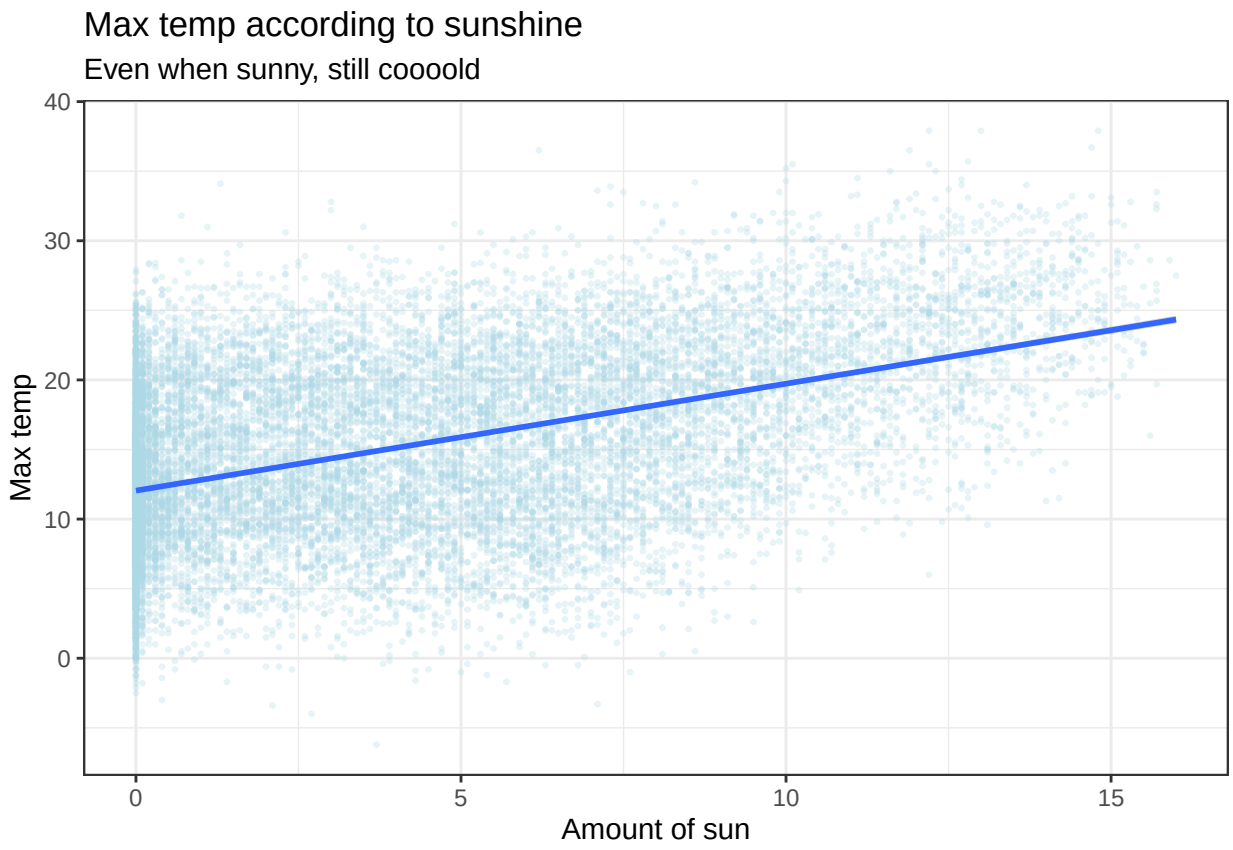
```
library(readr)
SUN_Hours <- read_csv("~/Desktop/20945043/Question2/Data/SUN Hours.csv")
```

```
## Rows: 20368 Columns: 17
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): TmStamp
## dbl (16): RecNum, BattV_Min, TrackerWM_Avg, Tracker2WM_Avg, ShadowWM_Avg, Su...
##
## 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.
```

Graph 2.1

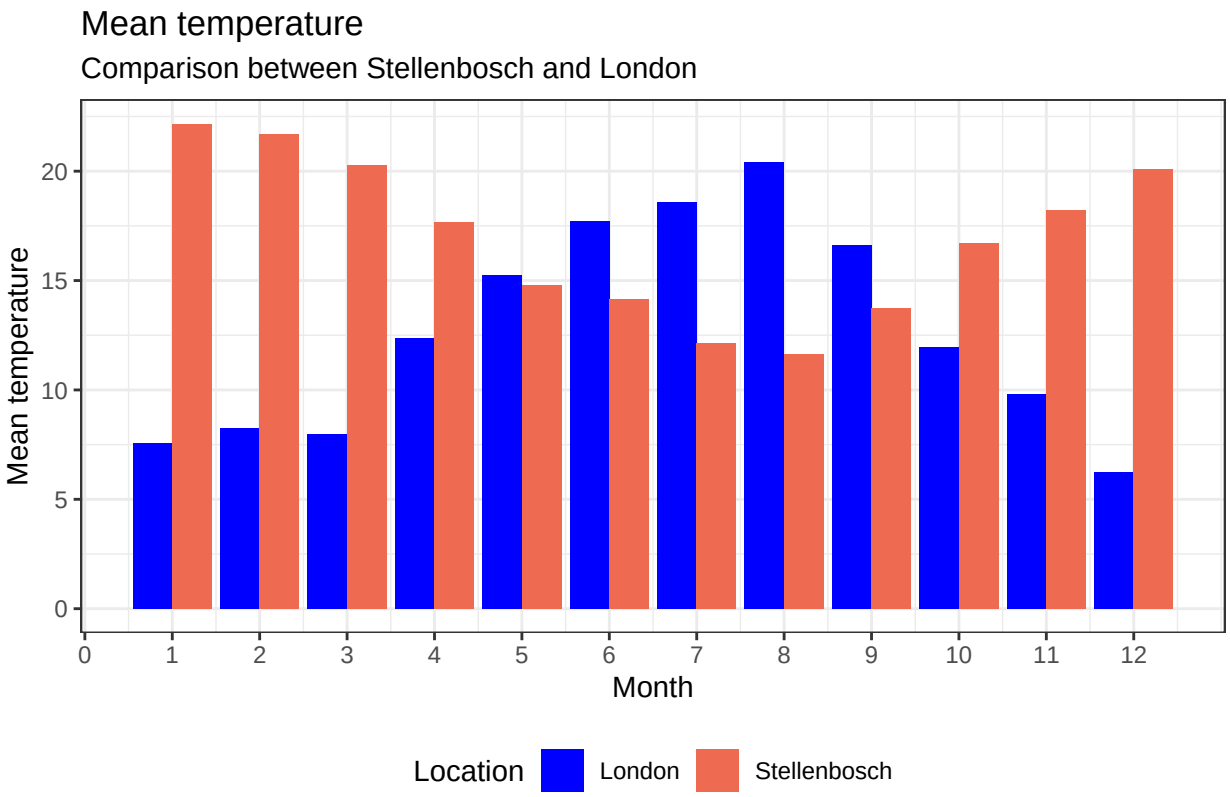
```
g <- london_code(london_weather)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Graph 2.2

```
g <- lond_c(london_weather, SUN_Hours)
```



Note:Sauran External Data used

Question 4

Code

```
gc()
```

```
##            used  (Mb) gc trigger  (Mb) limit (Mb) max used  (Mb)
## Ncells  2441825 130.5    3954015 211.2         NA  3954015 211.2
## Vcells 18209388 139.0   31217699 238.2      16384 31110389 237.4
```

```
library(pacman)
p_load(tidyverse, lubridate)

list.files('Question4/Code', full.names = T, recursive = T) %>% as.list() %>% walk(~source(.))
```

Loading Data

```
library(readr)
credits <- read_csv("~/Desktop/20945043/Question4/Data/netflix/credits.csv")
```

```
## Rows: 77213 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (4): id, name, character, role
## dbl (1): person_id
##
## 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.
```

```
library(readr)
titles <- read_csv("~/Desktop/20945043/Question4/Data/netflix/titles.csv")
```

```
## Rows: 5806 Columns: 15
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (8): id, title, type, description, age_certification, genres, production...
## dbl (7): release_year, runtime, seasons, imdb_score, imdb_votes, tmdb_popula...
##
## 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.
```

Graphs and Statistics

```
g <- netflix_code(titles)
```

```
## Adding missing grouping variables: `release_year`
## Selecting by ave_score
```

Figure 4.1

Table 1: IMDb Score: Movies vs Shows

	Movies	Shows
IMDb score	6.266980	7.017377

According to the data, shows did better than movies.

According to the accompanying data, the combination [‘scifi’, ‘family’, ‘fantasy’, ‘animation’, ‘action’] genre is most