Assignment 1

The following document includes Assignment 1 code and results

The following libraries are included for this assignment

- **readr** : to read dataset from a url
- **dplyr** : for dataset manapulation
- **rmarkdown** : for markdown

We will be using the Open Covid dataset for state of Ohio by The COVID Tracking project. See covidtracking.com for more details about the dataset.

Convert the table into dataframe so it is easy to work with.

Print descriptive statistics for quantitative variable

To print descriptive statistics we will be using the package dplyr: https://dplyr.tidyverse.org/reference/summarise_all.html

For the sake of readability we will be using the pipe notation `%>%` heavily used by dplyr. We pipe/chain function to select the columns death and hospitalized and calulate their mean, min and max.

Heavylifting is done by the `summarise_at` function it takes the following arguments:

- Columns to be summarized e.g. "death", "hospitalized"
- Functions to be used to summarize e.g. mean, min and max
- Should NA values be filtered out (TRUE)

```
## # A tibble: 1 x 6
##   death_min hospitalized_min death_max hospitalized_max death_mean
##       <dbl>            <dbl>     <dbl>            <dbl>      <dbl>
## 1         1               58     11070            45952      4283.
## # ... with 1 more variable: hospitalized_mean <dbl>
```

Print descriptive statistics for categorical variable

Here we only print all the data that have lower quality grade (B)

```
## # A tibble: 142 x 42
##    date       state dataQualityGrade death deathConfirmed deathIncrease
##    <date>     <chr> <chr>            <dbl>          <dbl>         <dbl>
##  1 2020-08-13 OH    B                 3755           3481            21
##  2 2020-08-12 OH    B                 3734           3460            26
##  3 2020-08-11 OH    B                 3708           3435            35
##  4 2020-08-10 OH    B                 3673           3405             4
##  5 2020-08-09 OH    B                 3669           3397             1
##  6 2020-08-08 OH    B                 3668           3396            16
##  7 2020-08-07 OH    B                 3652           3381            34
##  8 2020-08-06 OH    B                 3618           3348            22
##  9 2020-08-05 OH    B                 3596           3326            26
## 10 2020-08-04 OH    B                 3570           3301            31
## # ... with 132 more rows, and 36 more variables: deathProbable <dbl>,
```

```
## #   hospitalized <dbl>, hospitalizedCumulative <dbl>,
## #   hospitalizedCurrently <dbl>, hospitalizedIncrease <dbl>,
## #   inIcuCumulative <dbl>, inIcuCurrently <dbl>, negative <lgl>,
## #   negativeIncrease <dbl>, negativeTestsAntibody <lgl>,
## #   negativeTestsPeopleAntibody <lgl>, negativeTestsViral <lgl>,
## #   onVentilatorCumulative <lgl>, onVentilatorCurrently <dbl>, positive <dbl>,
## #   positiveCasesViral <dbl>, positiveIncrease <dbl>, positiveScore <dbl>,
## #   positiveTestsAntibody <lgl>, positiveTestsAntigen <dbl>,
## #   positiveTestsPeopleAntibody <lgl>, positiveTestsPeopleAntigen <lgl>,
## #   positiveTestsViral <dbl>, recovered <dbl>, totalTestEncountersViral <lgl>,
## #   totalTestEncountersViralIncrease <dbl>, totalTestResults <dbl>,
## #   totalTestResultsIncrease <dbl>, totalTestsAntibody <lgl>,
## #   totalTestsAntigen <dbl>, totalTestsPeopleAntibody <lgl>,
## #   totalTestsPeopleAntigen <lgl>, totalTestsPeopleViral <lgl>,
## #   totalTestsPeopleViralIncrease <dbl>, totalTestsViral <dbl>,
## #   totalTestsViralIncrease <dbl>
```

Transform variable

Get mean for columns death and hospatilized, grouped by their data quality. Here we use the `group_by` function that groups the reslting summary by the column `dataQualityGrade`

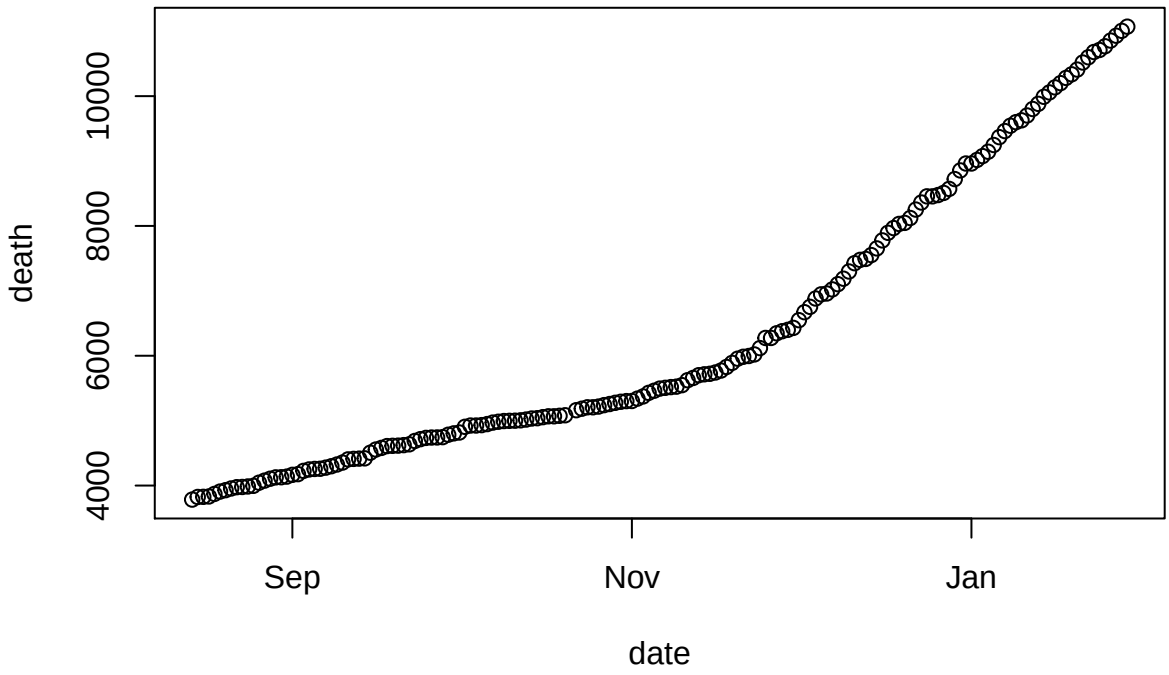
```
## # A tibble: 5 x 3
##   dataQualityGrade   Death Hospitalized
## * <chr>              <dbl>        <dbl>
## 1 #REF!             5149         17523
## 2 A+                6335         24502.
## 3 B                 1999.         6085.
## 4 D                    7           124.
## 5 <NA>                 2.33         70.5
```

Plot deaths vs days only for data with quality A+

Create an intermediate dataframe to plot the results. This dataframe is created by filtering the original data to only get results for A+ quality data. Then select only the columns `date` and `death`

This function plots the previously created dataframe.

```
plot(filtered_Data)
```



References

- https://covidtracking.com/
- https://uc-r.github.io/dplyr