Statistical Computing with R
Masters in Data Science 503 (S7)
First Batch, SMS, TU, 2021

Shital Bhandary

Associate Professor

Statistics/Bio-statistics, Demography and Public Health Informatics

Patan Academy of Health Sciences, Lalitpur, Nepal

Faculty, Data Analysis and Decision Modeling, MBA, Pokhara University, Nepal

Faculty, FAIMER Fellowship in Health Professions Education, India/USA.

Review Preview (Unit 2, Session 2)

- Manipulating and Tyding data
- Data Transformation
- Data Wrangling
- **Data Mining**
- **Text Mining**

What to do after importing data in R: Tidy!

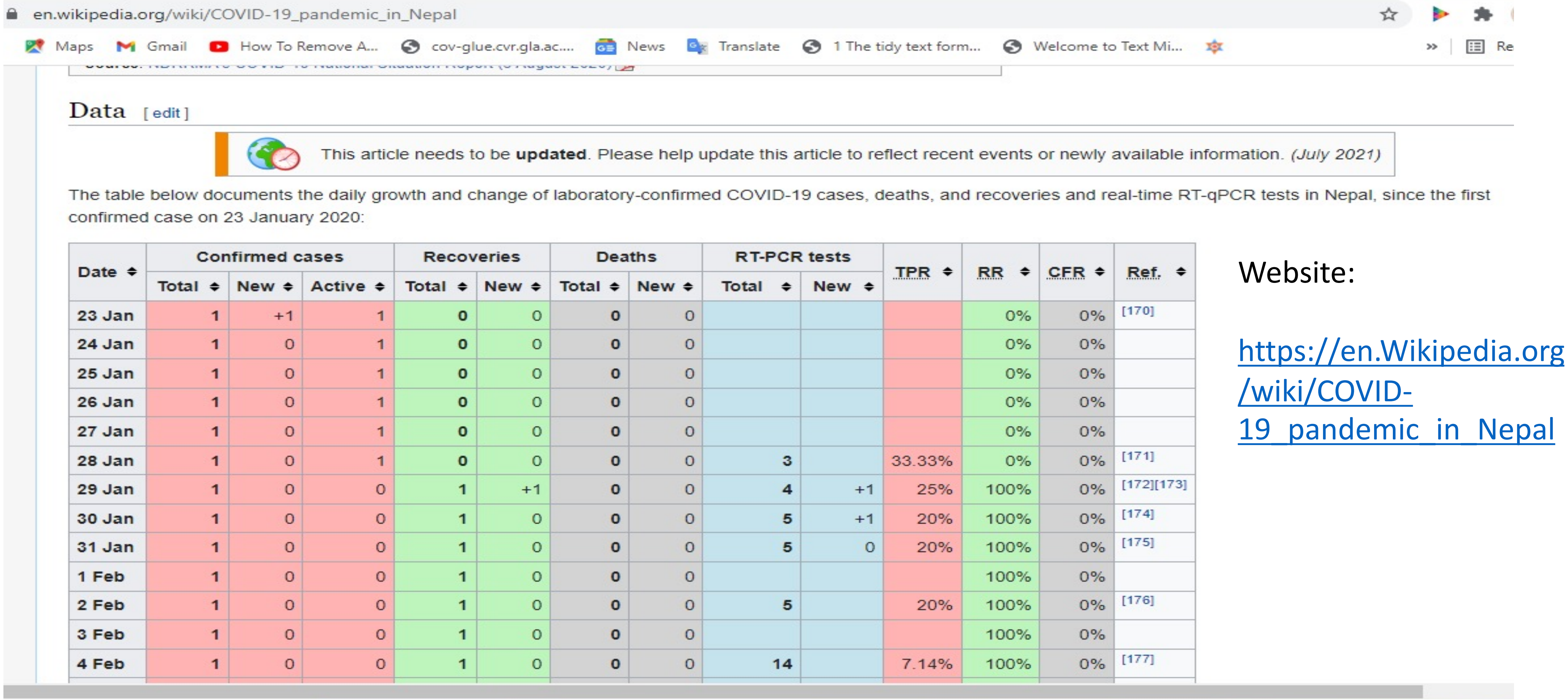
- Once you’ve imported your data, it is a good idea to **tidy** it. Tidying your data means storing it in a consistent form that matches the semantics of the dataset with the way it is stored.

- **In brief, when your data is tidy, each column is a variable, and each row is an observation. Column=Variable=Correct attributes too!**

- Tidy data is important because the consistent structure lets you focus your struggle on questions about the data, not fighting to get the data into the right form for different functions.

https://r4ds.had.co.nz/introduction.html

Exercise: Get this Wikipedia table in R Studio and create a working data.frame with “tidy”!



Data [edit]

This article needs to be **updated**. Please help update this article to reflect recent events or newly available information. *(July 2021)*

The table below documents the daily growth and change of laboratory-confirmed COVID-19 cases, deaths, and recoveries and real-time RT-qPCR tests in Nepal, since the first confirmed case on 23 January 2020:

Date	Confirmed cases			Recoveries		Deaths		RT-PCR tests		TPR	RR	CFR	Ref.
	Total	New	Active	Total	New	Total	New	Total	New				
23 Jan	1	+1	1	0	0	0	0				0%	0%	[170]
24 Jan	1	0	1	0	0	0	0				0%	0%	
25 Jan	1	0	1	0	0	0	0				0%	0%	
26 Jan	1	0	1	0	0	0	0				0%	0%	
27 Jan	1	0	1	0	0	0	0				0%	0%	
28 Jan	1	0	1	0	0	0	0	3		33.33%	0%	0%	[171]
29 Jan	1	0	0	1	+1	0	0	4	+1	25%	100%	0%	[172][173]
30 Jan	1	0	0	1	0	0	0	5	+1	20%	100%	0%	[174]
31 Jan	1	0	0	1	0	0	0	5	0	20%	100%	0%	[175]
1 Feb	1	0	0	1	0	0	0				100%	0%	
2 Feb	1	0	0	1	0	0	0	5		20%	100%	0%	[176]
3 Feb	1	0	0	1	0	0	0				100%	0%	
4 Feb	1	0	0	1	0	0	0	14		7.14%	100%	0%	[177]

Website:

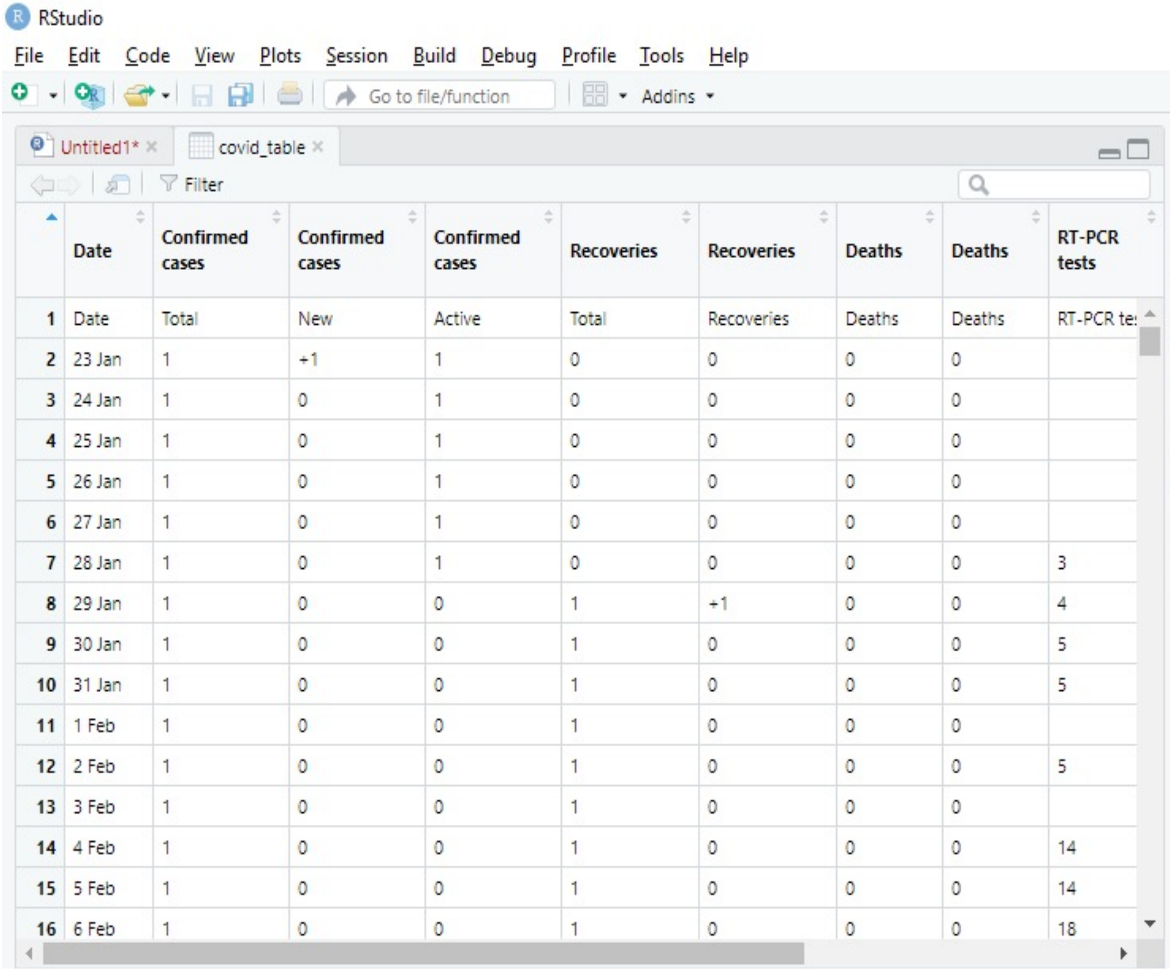
https://en.Wikipedia.org/wiki/COVID-19_pandemic_in_Nepal

Hint: Use the “rvest” library (in R Studio) (You can use the shared script too!)

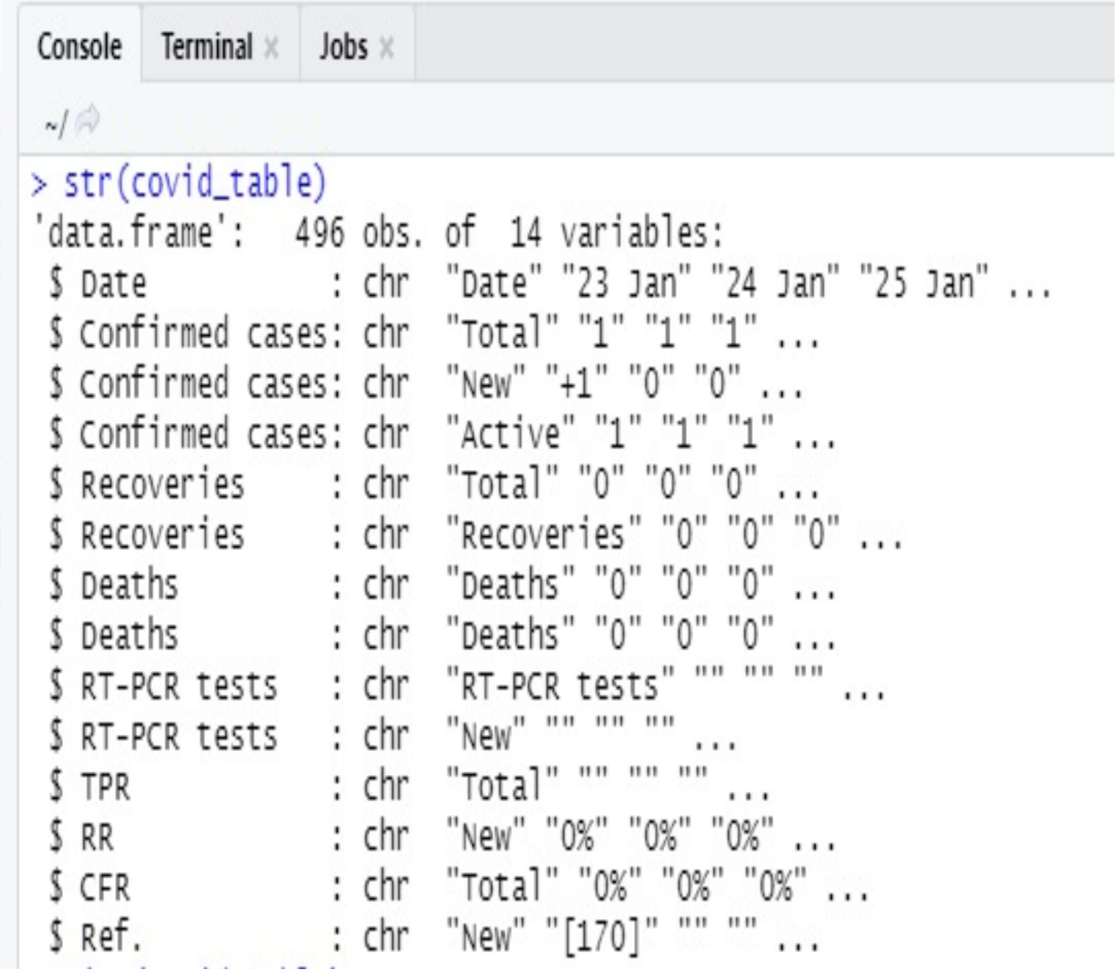
- library(rvest)
- library(dplyr)
- wiki_link <- "https://en.wikipedia.org/wiki/COVID-19_pandemic_in_Nepal"
- wiki_page <- read_html(wiki_link)
- tables <- wiki_page %>% html_table(fill = TRUE)   #Check the tables
- covid_table <- wiki_page %>%

  ```
        html_nodes("table") %>% .[16] %>%
        html_table() %>% .[[1]]
  ```

You will get this in R Studio:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Untitled1*
covid_table
Filter

	Date	Confirmed cases	Confirmed cases	Confirmed cases	Recoveries	Recoveries	Deaths	Deaths	RT-PCR tests
1	Date	Total	New	Active	Total	Recoveries	Deaths	Deaths	RT-PCR te:
2	23 Jan	1	+1	1	0	0	0	0	
3	24 Jan	1	0	1	0	0	0	0	
4	25 Jan	1	0	1	0	0	0	0	
5	26 Jan	1	0	1	0	0	0	0	
6	27 Jan	1	0	1	0	0	0	0	
7	28 Jan	1	0	1	0	0	0	0	3
8	29 Jan	1	0	0	1	+1	0	0	4
9	30 Jan	1	0	0	1	0	0	0	5
10	31 Jan	1	0	0	1	0	0	0	5
11	1 Feb	1	0	0	1	0	0	0	
12	2 Feb	1	0	0	1	0	0	0	5
13	3 Feb	1	0	0	1	0	0	0	
14	4 Feb	1	0	0	1	0	0	0	14
15	5 Feb	1	0	0	1	0	0	0	14
16	6 Feb	1	0	0	1	0	0	0	18



```
> str(covid_table)
'data.frame':   496 obs. of  14 variables:
 $ Date           : chr  "Date" "23 Jan" "24 Jan" "25 Jan" ...
 $ Confirmed cases: chr  "Total" "1" "1" "1" ...
 $ Confirmed cases: chr  "New" "+1" "0" "0" ...
 $ Confirmed cases: chr  "Active" "1" "1" "1" ...
 $ Recoveries     : chr  "Total" "0" "0" "0" ...
 $ Recoveries     : chr  "Recoveries" "0" "0" "0" ...
 $ Deaths         : chr  "Deaths" "0" "0" "0" ...
 $ Deaths         : chr  "Deaths" "0" "0" "0" ...
 $ RT-PCR tests   : chr  "RT-PCR tests" "" "" "" ...
 $ RT-PCR tests   : chr  "New" "" "" "" ...
 $ TPR            : chr  "Total" "" "" "" ...
 $ RR             : chr  "New" "0%" "0%" "0%" ...
 $ CFR            : chr  "Total" "0%" "0%" "0%" ...
 $ Ref.           : chr  "New" "[170]" "" "" ...
```

Now, “tidy” the covid_table data scrapped from Wikipedia link in R Studio as follows:

- Row 0: Variable names (Spaces between words must have underscores!)
- Row 1: Variable names as data values (need to remove them)
- Row 1 names must be appended to Row 0 with “underscore” *a priori*

- Columns: Need to have proper attributes for each variable e.g.
  - Date = Date variable
  - Confirmed cases = Number (Integer)
  - Confirmed cases new = Number (Integer) and so on and so forth
  - “+” must be removed from Confirmed Cases New, Recoveries Recoveries, Deaths Deaths, RT PCR tests New variables
  - “%” must be removed from the TPR Total, RR New and CFR Total variables
  - Ref. variable can be dropped (deleted/removed) as it is not useful

We can do as follows: Part 1
(Do this with one of “tidyverse” packages!)

- #Changing column names: Column Underscore Row 1

names(covid_table) = paste(names(covid_table), covid_table[1,], sep = "_")

- #Removing first row

covid_table = covid_table[-1,]

- #Removing last column

covid_table <- covid_table[,-14]

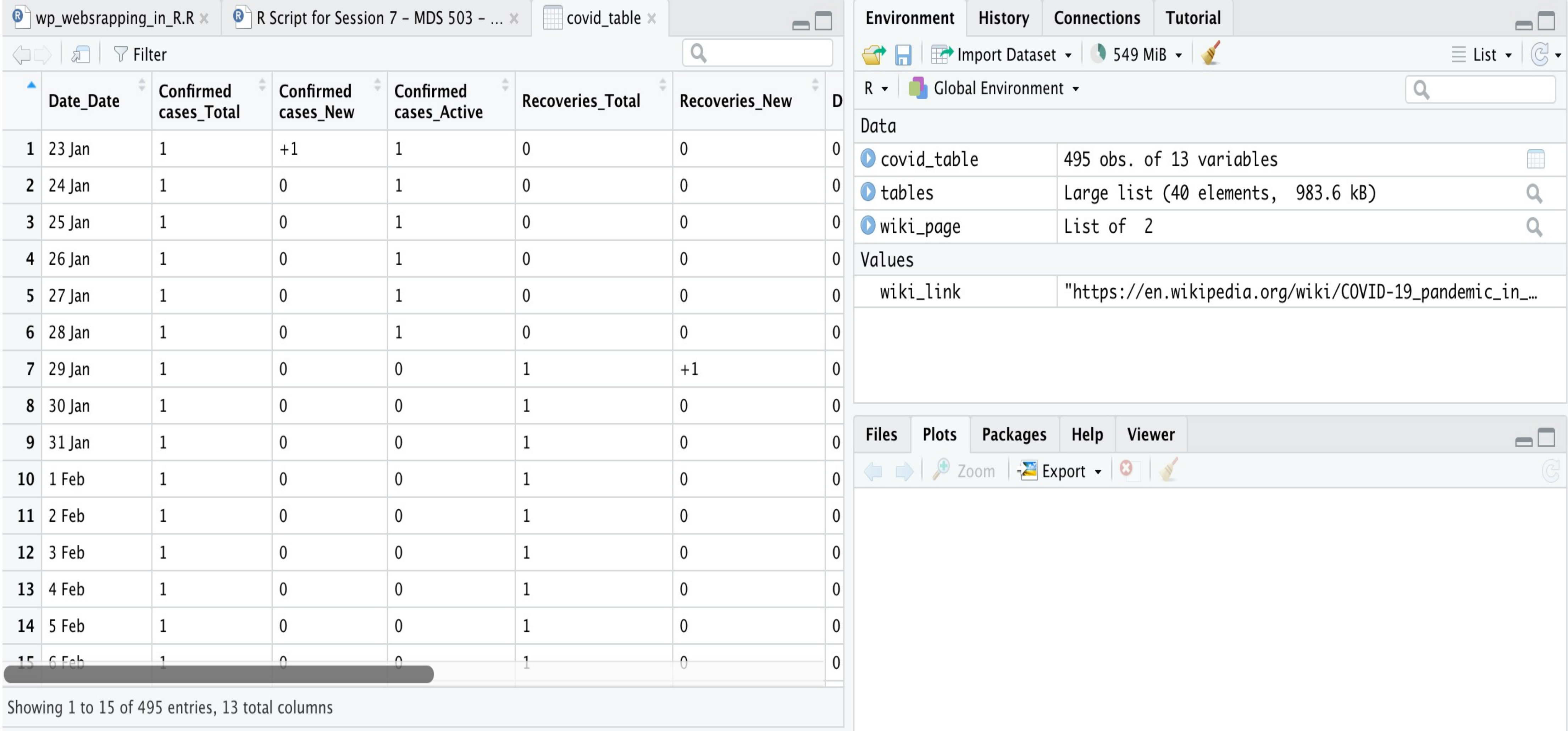
#Viewing the data

View(covid_table)

- #Checking structure of the data

str(covid_table)

This is what I got:



	Date_Date	Confirmed cases_Total	Confirmed cases_New	Confirmed cases_Active	Recoveries_Total	Recoveries_New	D
1	23 Jan	1	+1	1	0	0	0
2	24 Jan	1	0	1	0	0	0
3	25 Jan	1	0	1	0	0	0
4	26 Jan	1	0	1	0	0	0
5	27 Jan	1	0	1	0	0	0
6	28 Jan	1	0	1	0	0	0
7	29 Jan	1	0	0	1	+1	0
8	30 Jan	1	0	0	1	0	0
9	31 Jan	1	0	0	1	0	0
10	1 Feb	1	0	0	1	0	0
11	2 Feb	1	0	0	1	0	0
12	3 Feb	1	0	0	1	0	0
13	4 Feb	1	0	0	1	0	0
14	5 Feb	1	0	0	1	0	0
15	6 Feb	1	0	0	1	0	0

Showing 1 to 15 of 495 entries, 13 total columns

Data

covid_table	495 obs. of 13 variables
tables	Large list (40 elements, 983.6 kB)
wiki_page	List of 2

Values

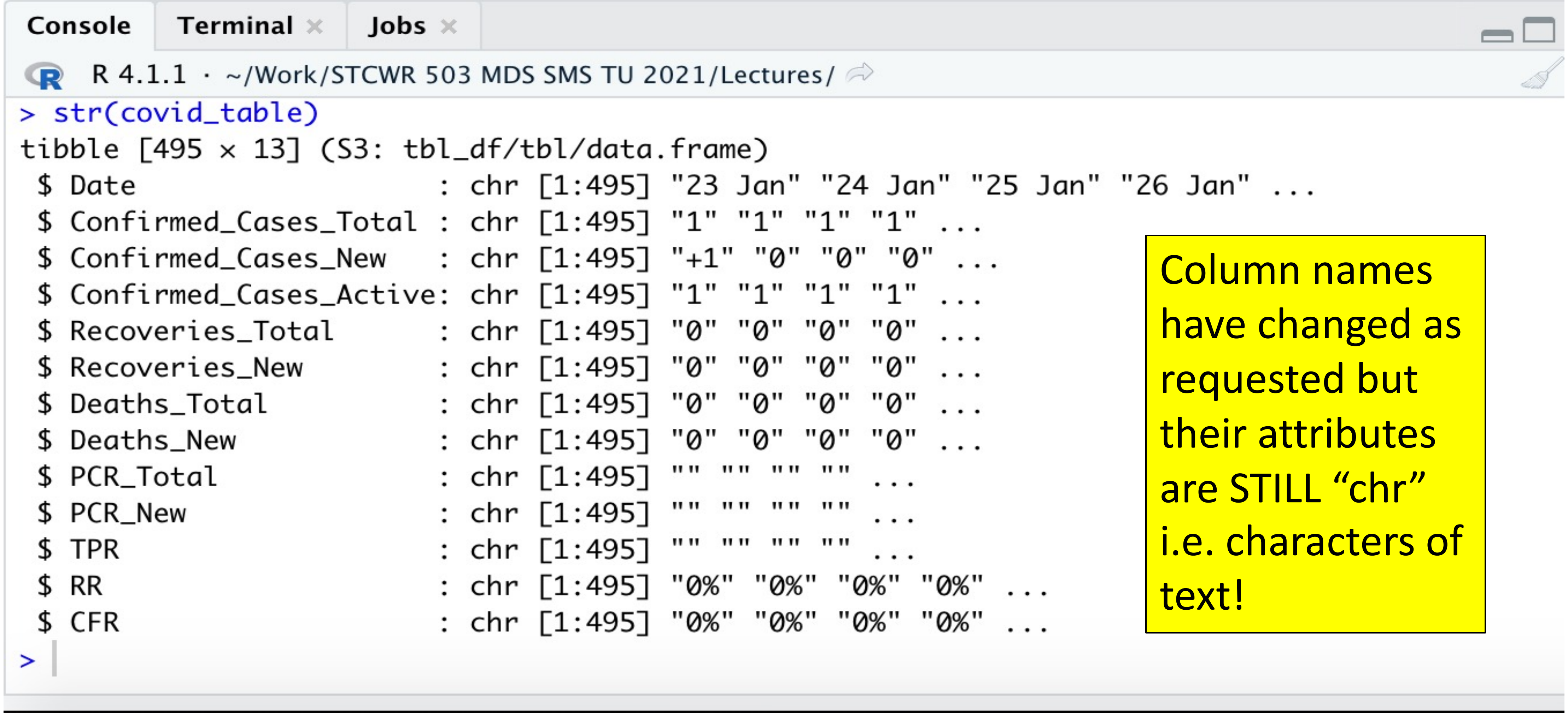
wiki_link	"https://en.wikipedia.org/wiki/COVID-19_pandemic_in_…

We can do as follows now: Part 2
(Do this with one of “tidyverse” packages!)

- **#Renaming the column names with underscore between spaces**

```
colnames(covid_table)                          #Checking column names to do correct coding below
names(covid_table)[names(covid_table) == "Date_Date"] = "Date"
names(covid_table)[names(covid_table) == "Confirmed cases_Total"] = "Confirmed_Cases_Total"
names(covid_table)[names(covid_table) == "Confirmed cases_New"] = "Confirmed_Cases_New"
names(covid_table)[names(covid_table) == "Confirmed cases_Active"] = "Confirmed_Cases_Active"
names(covid_table)[names(covid_table) == "RT-PCR tests_Total"] = "PCR_Total"
names(covid_table)[names(covid_table) == "RT-PCR tests_New"] = "PCR_New"
names(covid_table)[names(covid_table) == "TPR_TPR"] = "TPR"
names(covid_table)[names(covid_table) == "RR_RR"] = "RR"
names(covid_table)[names(covid_table) == "CFR_CFR"] = "CFR"
str(covid_table)
```

What has been changed?



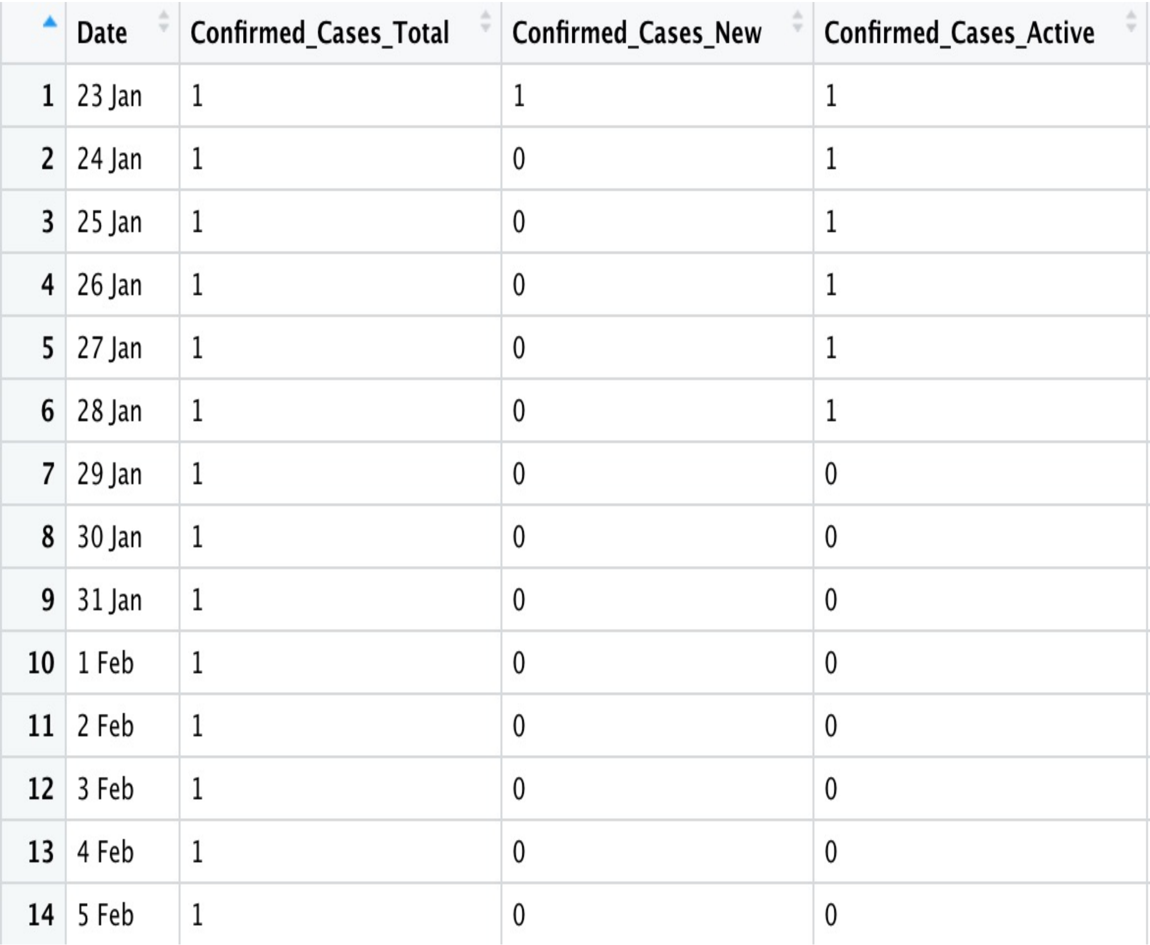
```
> str(covid_table)
tibble [495 × 13] (S3: tbl_df/tbl/data.frame)
 $ Date                  : chr [1:495] "23 Jan" "24 Jan" "25 Jan" "26 Jan" ...
 $ Confirmed_Cases_Total : chr [1:495] "1" "1" "1" "1" ...
 $ Confirmed_Cases_New   : chr [1:495] "+1" "0" "0" "0" ...
 $ Confirmed_Cases_Active: chr [1:495] "1" "1" "1" "1" ...
 $ Recoveries_Total      : chr [1:495] "0" "0" "0" "0" ...
 $ Recoveries_New        : chr [1:495] "0" "0" "0" "0" ...
 $ Deaths_Total          : chr [1:495] "0" "0" "0" "0" ...
 $ Deaths_New            : chr [1:495] "0" "0" "0" "0" ...
 $ PCR_Total             : chr [1:495] "" "" "" "" ...
 $ PCR_New               : chr [1:495] "" "" "" "" ...
 $ TPR                   : chr [1:495] "" "" "" "" ...
 $ RR                    : chr [1:495] "0%" "0%" "0%" "0%" ...
 $ CFR                   : chr [1:495] "0%" "0%" "0%" "0%" ...
> 
```

Column names have changed as requested but their attributes are STILL “chr” i.e. characters of text!

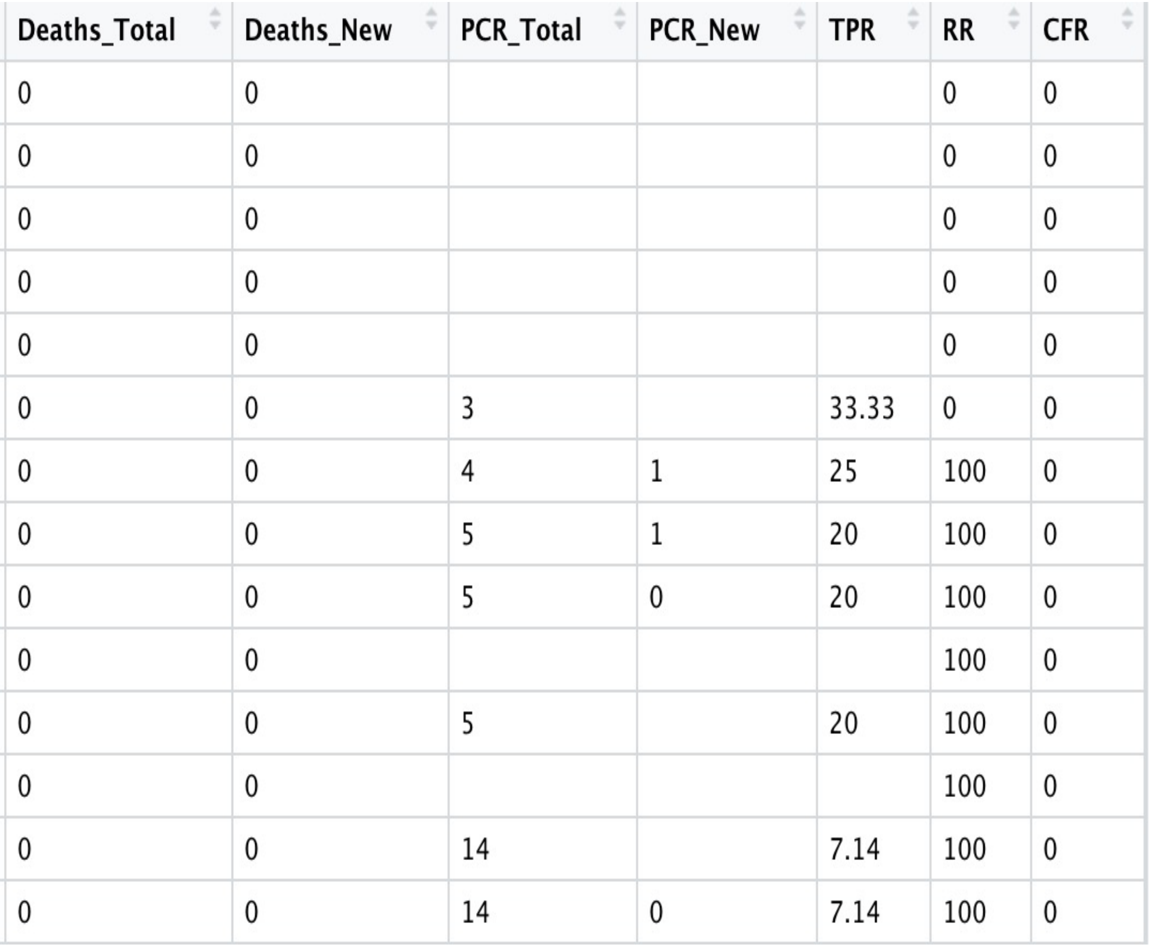
Removing “+” and “%” from certain variables: (Do this with one of “tidyverse” packages!)

- #Removing + from four variables
- covid_table$Confirmed_Cases_New = gsub('[+]', '', covid_table$Confirmed_Cases_New)
- covid_table$Recoveries_New = gsub('[+]', '', covid_table$Recoveries_New)
- covid_table$Deaths_New = gsub('[+]', '', covid_table$Deaths_New)
- covid_table$PCR_New = gsub('[+]', '', covid_table$PCR_New)
- #Removing % from three variables
- covid_table$TPR = gsub('[%]', '', covid_table$TPR)
- covid_table$RR = gsub('[%]', '', covid_table$RR)
- covid_table$CFR = gsub('[%]', '', covid_table$CFR)

This is what I got:



	Date	Confirmed_Cases_Total	Confirmed_Cases_New	Confirmed_Cases_Active
1	23 Jan	1	1	1
2	24 Jan	1	0	1
3	25 Jan	1	0	1
4	26 Jan	1	0	1
5	27 Jan	1	0	1
6	28 Jan	1	0	1
7	29 Jan	1	0	0
8	30 Jan	1	0	0
9	31 Jan	1	0	0
10	1 Feb	1	0	0
11	2 Feb	1	0	0
12	3 Feb	1	0	0
13	4 Feb	1	0	0
14	5 Feb	1	0	0

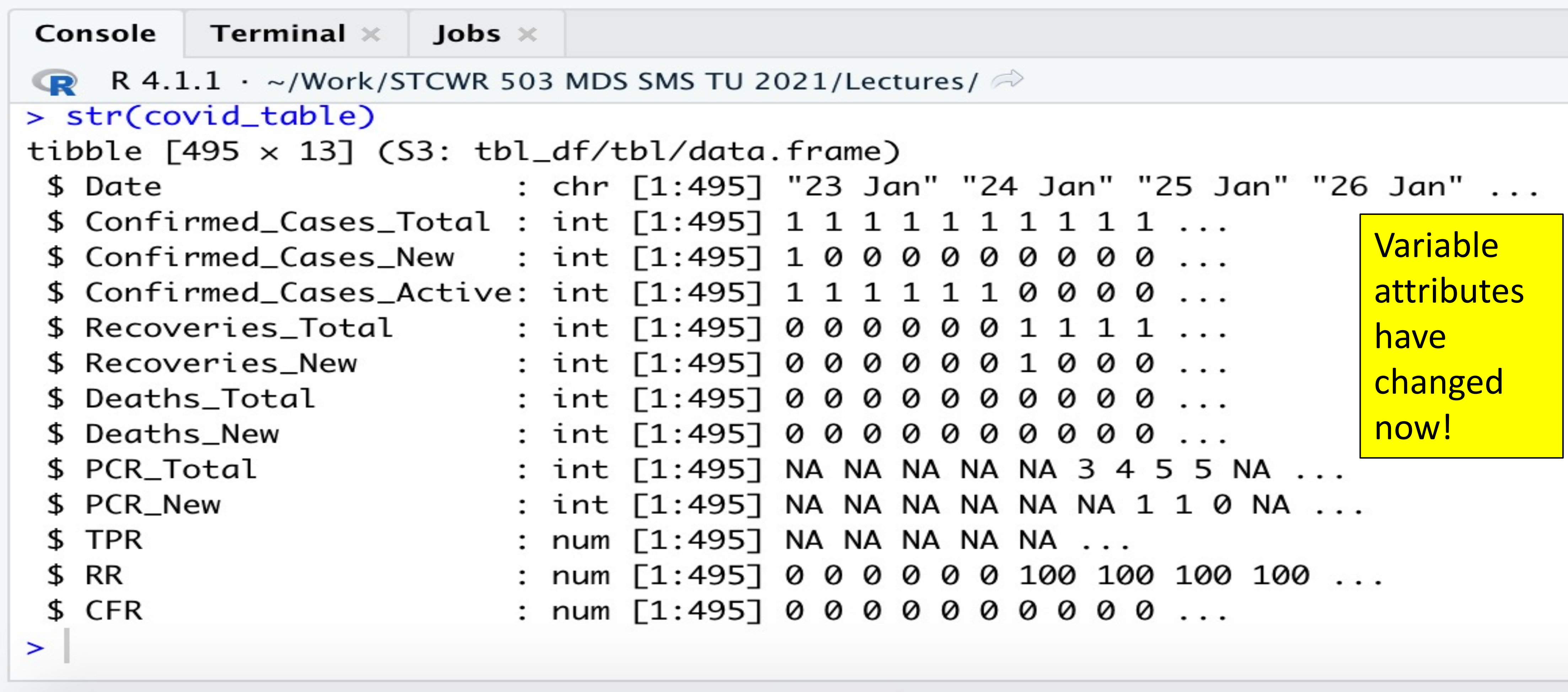


Deaths_Total	Deaths_New	PCR_Total	PCR_New	TPR	RR	CFR
0	0				0	0
0	0				0	0
0	0				0	0
0	0				0	0
0	0				0	0
0	0	3		33.33	0	0
0	0	4	1	25	100	0
0	0	5	1	20	100	0
0	0	5	0	20	100	0
0	0				100	0
0	0	5		20	100	0
0	0				100	0
0	0	14		7.14	100	0
0	0	14	0	7.14	100	0

Changing attributes of the variables:
(Do this with one of “tidyverse” packages!)

- #Converting chr variables as numbers and integers
- covid_table$Confirmed_Cases_Total = as.integer(covid_table$Confirmed_Cases_Total)
- covid_table$Confirmed_Cases_New = as.integer(covid_table$Confirmed_Cases_New)
- covid_table$Confirmed_Cases_Active = as.integer(covid_table$Confirmed_Cases_Active)
- covid_table$Recoveries_Total = as.integer(covid_table$Recoveries_Total)
- covid_table$Recoveries_New = as.integer(covid_table$Recoveries_New)
- covid_table$Deaths_Total = as.integer(covid_table$Deaths_Total)
- covid_table$Deaths_New = as.integer(covid_table$Deaths_New)
- covid_table$PCR_Total = as.integer(covid_table$PCR_Total)
- covid_table$PCR_New = as.integer(covid_table$PCR_New)
- covid_table$TPR = as.numeric(covid_table$TPR)
- covid_table$RR = as.numeric(covid_table$RR)
- covid_table$CFR = as.numeric(covid_table$CFR)
- covid_table$Ref = as.character(covid_table$Ref)
- str(covid_table)

What changes do you see?



```
R 4.1.1 · ~/Work/STCWR 503 MDS SMS TU 2021/Lectures/
> str(covid_table)
tibble [495 × 13] (S3: tbl_df/tbl/data.frame)
 $ Date                  : chr [1:495] "23 Jan" "24 Jan" "25 Jan" "26 Jan" ...
 $ Confirmed_Cases_Total : int [1:495] 1 1 1 1 1 1 1 1 1 1 ...
 $ Confirmed_Cases_New   : int [1:495] 1 0 0 0 0 0 0 0 0 0 ...
 $ Confirmed_Cases_Active: int [1:495] 1 1 1 1 1 1 0 0 0 0 ...
 $ Recoveries_Total      : int [1:495] 0 0 0 0 0 0 1 1 1 1 ...
 $ Recoveries_New        : int [1:495] 0 0 0 0 0 0 1 0 0 0 ...
 $ Deaths_Total          : int [1:495] 0 0 0 0 0 0 0 0 0 0 ...
 $ Deaths_New            : int [1:495] 0 0 0 0 0 0 0 0 0 0 ...
 $ PCR_Total             : int [1:495] NA NA NA NA NA 3 4 5 5 NA ...
 $ PCR_New               : int [1:495] NA NA NA NA NA NA 1 1 0 NA ...
 $ TPR                   : num [1:495] NA NA NA NA NA ...
 $ RR                    : num [1:495] 0 0 0 0 0 0 100 100 100 100 ...
 $ CFR                   : num [1:495] 0 0 0 0 0 0 0 0 0 0 ...
> 
```

Variable attributes have changed now!

Transforming data (after importing & tidying):

- Once tidying is done, a common first step is to transform it.
- Transformation includes narrowing in on observations of interest (like all people in one city, or all data from the last year), creating new variables that are functions of existing variables (like computing speed from distance and time), and calculating a set of summary statistics (like counts or means).
- Together, tidying and transforming are called **wrangling**, because getting your data in a form that's natural to work with often feels like a fight!

https://r4ds.had.co.nz/introduction.html

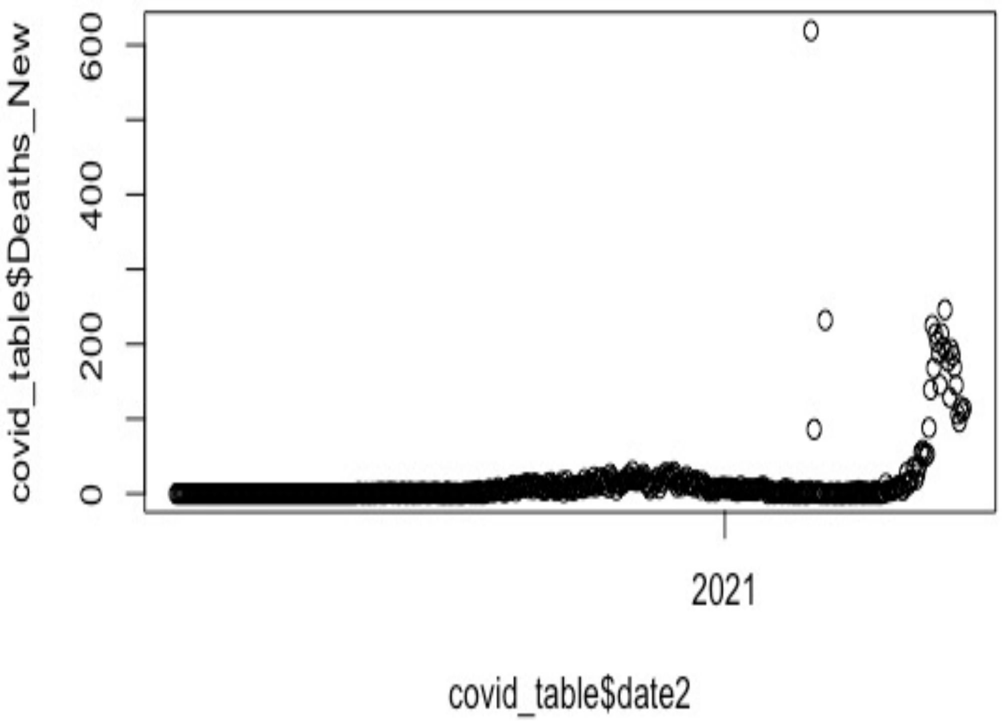
Creating new date variable and getting plot of <u>**daily new deaths (narrowing)**</u> by this date variable: (Do this with one of “tidyverse” packages!)

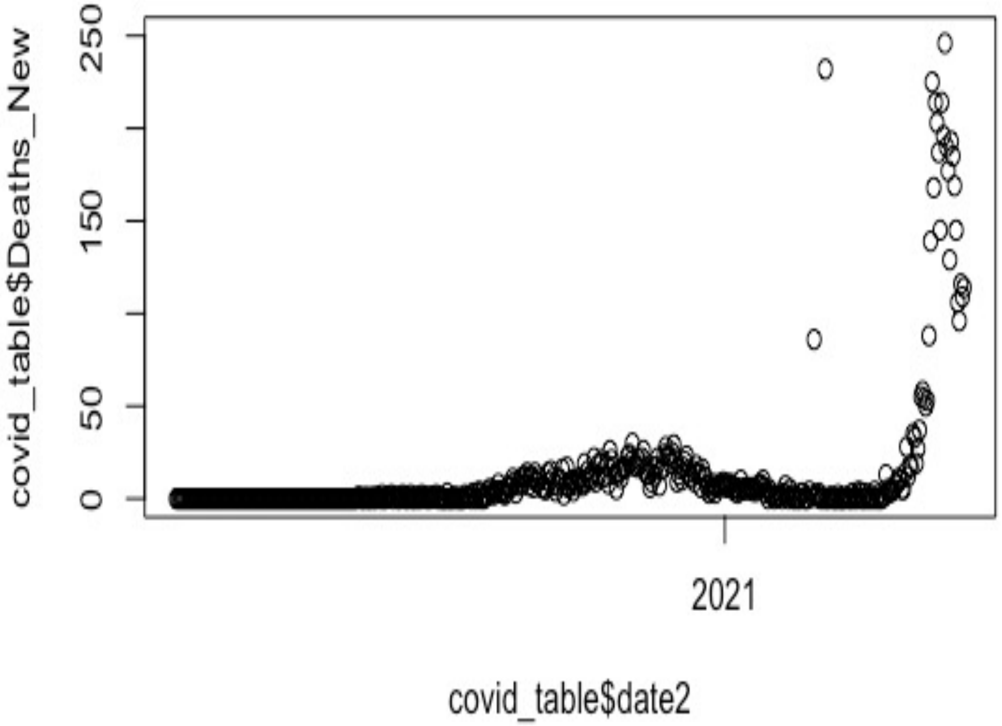
- #Changing date variable as date2
- date2 = seq(as.Date('2020-1-23'), by='days', length.out = 495)
- covid_table = cbind(covid_table, date2)

- #Plot
- plot(covid_table$date2, covid_table$Deaths_New)
- plot(covid_table$date2, covid_table$Deaths_New, ylim = range(0:250))

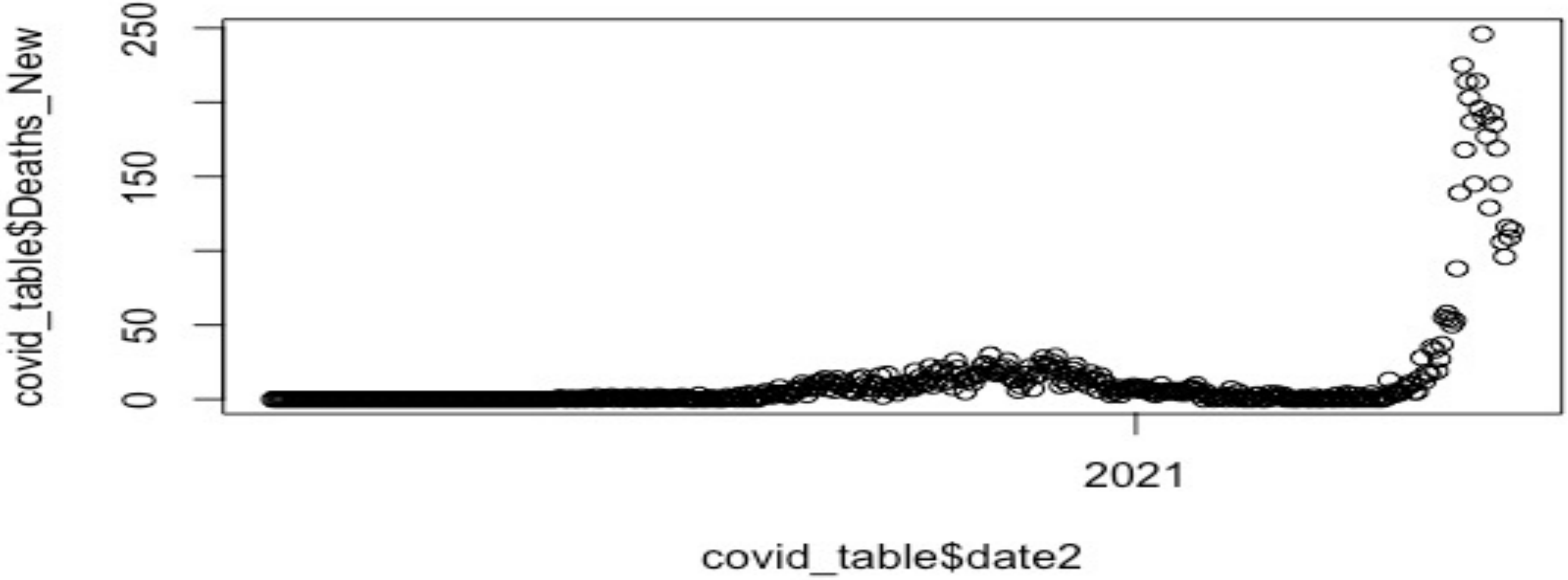
The two requested plots: Any problem here?

There are unusual “daily deaths” data!





After removing 3 unusual data : Is there any other alternative to this (self-learning)!



Code used to remove 3 unusual data (not recommended for real life problems/projects!): (Do this with one of “tidyverse” packages!)

- #Repalce Deaths_New of 24 Feb as 1 in the data

```
covid_table[covid_table$date2=="2021-02-24", "Deaths_New"] = 1
```

- #https://www.france24.com/en/live-news/20210224-nepal-revises-coronavirus-death-toll

```
covid_table[covid_table$date2=="2021-02-26", "Deaths_New"] = 4
```

- #https://thehimalayantimes.com/covid-19/nepal-covid-19-update-112-new-cases-54-recoveries-and-four-fatalities-recorded-on-friday

```
covid_table[covid_table$date2=="2021-03-05", "Deaths_New"] = 0
```

- #https://thehimalayantimes.com/covid-19/nepal-covid-19-update-47-new-cases-62-recoveries-and-no-deaths-recorded-on-saturday

```
plot(covid_table$date2, covid_table$Deaths_New)
```

Transformation: Summary statistics!

summary(covid_table$Deaths_New) **#Interpretation?**

- Min. 1st Qu. Median Mean 3rd Qu. Max.
- 0.00 0.00 2.00 14.92 11.00 619.00

summary(covid_table$Deaths_Total) **#Interpretation?**

- Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
- 0.0 0.0 18.0 142.9 149.0 984.0 210

summary(covid_table$Deaths_CFR) **#Interpretation?**

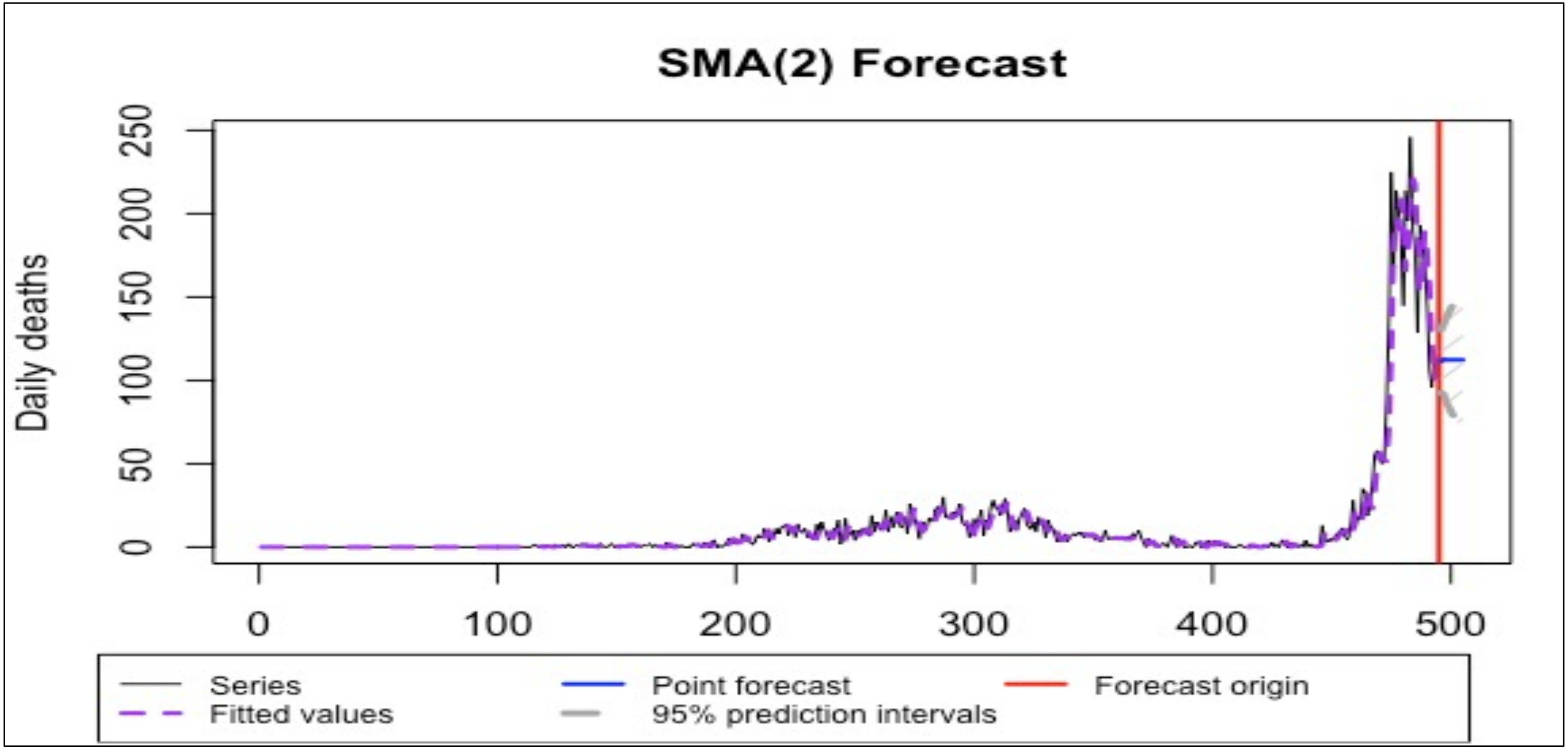
- Length Class Mode
- 0 NULL NULL **Why 0?** #What is the PROBLEM!

Once we have finished “data wrangling” i.e. tidying and transformation, we can then do:

- Data Visualization
  - We have already started it for deaths_new variable with date variable!
- Data Modelling
  - Since there are no errors in the deaths_new data and visualization is good, we will do/fit the modelling/a simple time series model now
  - We shall check which “moving average” level if most appropriate for the 495 days of COVID-19 data of Nepal after removing 3 unusual cases
  - We will use “smooth” package for this as it can “automatically” detects the best level of moving average for this data
  - More here: https://forecasting.svetunkov.ru/en/tag/sma/

https://r4ds.had.co.nz/introduction.html

Simple Moving Average Fit & Forecast for daily deaths data obtained after remove 3 outliers:



Code used for Auto SMA Fit and Forecast:

- #Simple Moving Average fit for new deaths data!

```
library(smooth)
sma = sma(covid_table$Deaths_New, h=14, silent=FALSE)

summary(sma)
forecast(sma)

plot(forecast(sma), main = "SMA(2) Forecast", ylab="Daily deaths")
```

Data Visualization: “ggplot2” package?

- **Visualisation** is a fundamentally human activity.
- A good visualisation will show you things that you did not expect, or raise new questions about the data.
- A good visualisation might also hint that you’re asking the wrong question, or you need to collect different data.
- Visualisations can surprise you, but don’t scale particularly well because they require a human to interpret them.

https://r4ds.had.co.nz/introduction.html

Models: “modelr” package?

- **Models** are complementary tools to visualisation.
- Once you have made your questions sufficiently precise, you can use a model to answer them.
- Models are a fundamentally mathematical or computational tool, so they generally scale well.
- Even when they don’t, it’s usually cheaper to buy more computers than it is to buy more brains!
- But every model makes assumptions, and by its very nature a model cannot question its own assumptions.
- That means a model cannot fundamentally surprise you.

https://r4ds.had.co.nz/introduction.html

Let us visit this website for Project Work:

- https://documenter.getpostman.com/view/9992373/SzS7PkXr

- And, get this data in R using R Studio to Data Wrangling:

- https://data.askbhunte.com/api/v1/covid/timeline

- https://data.askbhunte.com/api/v1/covid

- https://data.askbhunte.com/api/v1/covid/summary

Hint: Importing “timeline” JSON file in R!

- library(jsonlite)
- url <- 'https://data.askbhunte.com/api/v1/covid/timeline’
- covidtbl <- fromJSON(txt=url, flatten=TRUE)
- colnames(covidtbl)
- summary(covidtbl)

Question/Queries?

Thank you!

@shitalbhandary