

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
École nationale Supérieure d'Informatique (ESI ex. INI)

2^{ème} année cycle supérieur SIT(2CS-SIT1)

Apprentissage par renforcement

COVID 19

Réalisé par :

ADLA Ilyes chiheb eddine

Année 2019/2020

Table des matières

• Etude statistique	3
• Test rapide	5
un aperçu des données	5
Les données de l'algerie	7
• Explorer les données	8
• Créer des visuels	10
Graphique linéaire par pays	11
Le Cas de l'algerie	13
Le Cas du monde entier	17
• Regression	19



Etude statistique

Pour utiliser un accès pratique aux données épidémiologiques sur l'épidémie de coronavirus, nous allons un package R, nCov2019 (<https://github.com/GuangchuangYu/nCov2019>) .

il comprend également des statistiques détaillées en temps réel, et des données historiques en Chine, jusqu'au niveau de la ville , Ce package nous permet aussi d'accéder aux dernières données et aux données historiques des cas de tous les pays, de tracer des données sur une carte et de créer divers graphiques

voici les commandes de l'installation de la source des données et son déploiement:

```
> library('remotes')
warning message:
le package 'remotes' a été compilé avec la version R 3.6.3
> detach("package:remotes", unload = TRUE)
> library(remotes)
warning message:
le package 'remotes' a été compilé avec la version R 3.6.3
> remotes::install_github("GuangchuangYu/nCov2019", dependencies = TRUE)
Downloading Github repo GuangchuangYu/nCov2019@master
These packages have more recent versions available.
It is recommended to update all of them.
which would you like to update?

1: All
2: CRAN packages only
3: None
4: ggplot2      (3.2.1   -> 3.3.0   ) [CRAN]
5: jsonlite     (1.6     -> 1.6.1   ) [CRAN]
6: tibble       (2.1.3   -> 3.0.0   ) [CRAN]
7: knitr        (1.26    -> 1.28    ) [CRAN]
8: rmarkdown    (1.18     -> 2.1     ) [CRAN]
9: rlang        (0.4.1   -> 0.4.5   ) [CRAN]
10: cli         (1.1.0   -> 2.0.2   ) [CRAN]
11: fansi       (0.4.0   -> 0.4.1   ) [CRAN]
12: lifecycle   (0.1.0   -> 0.2.0   ) [CRAN]
13: pillar      (1.4.2   -> 1.4.3   ) [CRAN]
14: vctrs       (0.2.0   -> 0.2.4   ) [CRAN]
15: glue        (1.3.1   -> 1.4.0   ) [CRAN]
16: digest      (0.6.22  -> 0.6.25  ) [CRAN]
17: Rcpp        (1.0.2   -> 1.0.4   ) [CRAN]
18: R6          (2.4.0   -> 2.4.1   ) [CRAN]
19: rstudioapi  (0.10    -> 0.11    ) [CRAN]
20: callr       (3.3.2   -> 3.4.3   ) [CRAN]
21: prettyunits (1.0.2   -> 1.1.1   ) [CRAN]
22: backports   (1.1.5   -> 1.1.6   ) [CRAN]
23: processx    (3.4.1   -> 3.4.2   ) [CRAN]
24: ps          (1.3.0   -> 1.3.2   ) [CRAN]
25: farver      (2.0.1   -> 2.0.3   ) [CRAN]
26: BH          (1.69.0-1 -> 1.72.0-3) [CRAN]
27: yaml       (2.2.0   -> 2.2.1   ) [CRAN]
28: mime        (0.7     -> 0.9     ) [CRAN]
29: tinytex    (0.17    -> 0.21    ) [CRAN]
30: xfun        (0.11    -> 0.12    ) [CRAN]
31: stringi     (1.4.3   -> 1.4.6   ) [CRAN]
32: tidyselect  (0.2.5   -> 1.0.0   ) [CRAN]
33: curl        (4.2     -> 4.3     ) [CRAN]

Enter one or more numbers, or an empty line to skip updates:
|
```

La suite de l'installation dans l'image suivante :

downloaded 515 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/fastmap_1.0.1.zip'

Content type 'application/zip' length 413224 bytes (403 KB)

downloaded 403 KB

package 'downloader' successfully unpacked and MD5 sums checked

package 'magick' successfully unpacked and MD5 sums checked

package 'shiny' successfully unpacked and MD5 sums checked

package 'prettydoc' successfully unpacked and MD5 sums checked

package 'httpuv' successfully unpacked and MD5 sums checked

package 'xtable' successfully unpacked and MD5 sums checked

package 'sourcetools' successfully unpacked and MD5 sums checked

package 'fastmap' successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\HP\AppData\Local\Temp\RtmpkzfziG\downloaded_packages

WARNING: Rtools is required to build R packages, but is not currently installed.

Please download and install Rtools custom from <http://cran.r-project.org/bin/windows/Rtools/>.

✓ checking for file 'C:\Users\HP\AppData\Local\Temp\RtmpkzfziG\renotes2814267b231f\Guangchuangyu-nCov2019-86c5db8\DESCRIPTION' (2.3s)

- preparing 'nCov2019': (1.4s)

✓ checking DESCRIPTION meta-information ...

- checking for LF line-endings in source and make files and shell scripts

- checking for empty or unneeded directories

- building 'nCov2019_0.3.4.tar.gz'

Installing package into 'C:/Users/HP/Documents/R/win-library/3.6'

(as 'lib' is unspecified)

* installing *source* package 'nCov2019' ...

** using staged installation

** R

** inst

** byte-compile and prepare package for lazy loading

** help

*** installing help indices

converting help for package 'nCov2019'

finding HTML links ... fini

dashboard htm1

get_nCov2019 htm1

load_nCov2019 htm1

trans_city htm1

trans_province htm1

** building package indices

** installing vignettes

** testing if installed package can be loaded from temporary location

** testing if installed package can be loaded from final location

* testing if installed package keeps a record of temporary installation path

✓ DONE (nCov2019)

>



Test rapide

Pour interroger les dernières données, vous pouvez les charger avec `get_nCov2019()` et obtenir par défaut le nombre total de cas confirmés en Chine.

Par défaut, le paramètre de langue est automatiquement défini sur chinois on doit le rendre en anglais

vous pouvez utiliser `summary(x)` pour obtenir des données chinoises récentes.

```
Warning message:
In readRDS(system.file("country_translate.rds", package = "nCov2019")) :
  strings not representable in native encoding will be translated to UTF-8
> library('nCov2019')
> x <- get_nCov2019(lang = 'en')
There were 37 warnings (use warnings() to see them)
> x
China (total confirmed cases): 83095
last update: 2020-04-07 20:36:49
> head(summary(x))
  confirm suspect dead heal nowConfirm nowSevere importedCase deadRate healRate  date noInfect
1      41       0    1   0          0          0            0      2.4      0.0 01.13         0
2      41       0    1   0          0          0            0      2.4      0.0 01.14         0
3      41       0    2   5          0          0            0      4.9     12.2 01.15         0
4      45       0    2   8          0          0            0      4.4     17.8 01.16         0
5      62       0    2  12          0          0            0      3.2     19.4 01.17         0
6     198       0    3  17          0          0            0      1.5      8.6 01.18         0
> |
```

un aperçu des données

La méthode pour accéder aux données historiques est fondamentalement la même que pour obtenir les dernières données, mais la fonction d'entrée est `load_nCov2019()`

```
→ > x <- get_nCov2019()
There were 37 warnings (use warnings() to see them)
→ > y <- load_nCov2019()
Warning messages:
1: In readRDS(system.file("country_translate.rds", package = "nCov2019")) :
  strings not representable in native encoding will be translated to UTF-8
2: In readRDS(system.file("oversea_province_translate.rds", package = "nCov2019")) :
  strings not representable in native encoding will be translated to UTF-8
3: In load(system.file("ncovEnv.rda", package = "nCov2019")) :
  strings not representable in native encoding will be translated to UTF-8
4: In load(system.file("ncovEnv.rda", package = "nCov2019")) :
  strings not representable in native encoding will be translated to UTF-8
5: In readRDS(system.file("oversea_province_translate.rds", package = "nCov2019")) :
  strings not representable in native encoding will be translated to UTF-8
6: In load(system.file("ncovEnv.rda", package = "nCov2019")) :
  strings not representable in native encoding will be translated to UTF-8
→ > y
nCov2019 historical data
last update: 2020-04-06
→ > x
China (total confirmed cases): 83095
last update: 2020-04-07 20:36:49
```

vous pouvez obtenir des données historiques globales en spécifiant le 'global' paramètre.

```
> x
china (total confirmed cases): 83095
last update: 2020-04-07 20:36:49
> x['global',]
```

	name	confirm	suspect	dead	deadRate	showRate	heal	healRate	showHeal
1	china	83095	89	3340	4.02	FALSE	77489	93.25	TRUE
2	United States	143071	0	2513	1.76	FALSE	4856	3.39	FALSE
3	Italy	97689	0	10779	11.03	FALSE	13030	13.34	FALSE
4	Spain	85195	0	7340	8.62	FALSE	14709	17.27	FALSE
5	Germany	63929	0	560	0.88	FALSE	9211	14.41	FALSE
6	Iran	44606	0	2898	6.5	FALSE	14656	32.86	FALSE
7	France	40751	0	2612	6.41	FALSE	7238	17.76	FALSE
8	United Kingdom	22472	0	1793	7.98	FALSE	179	0.8	FALSE
9	Switzerland	15526	0	312	2.01	FALSE	1823	11.74	FALSE
10	Belgium	11899	0	513	4.31	FALSE	1527	12.83	FALSE
11	Netherlands	11814	0	864	7.31	FALSE	253	2.14	FALSE
12	South Korea	9976	0	169	1.69	FALSE	5828	58.42	FALSE
13	Austria	9377	0	108	1.15	FALSE	636	6.78	FALSE
14	Turkey	9217	0	131	1.42	FALSE	105	1.14	FALSE
15	Portugal	6408	0	140	2.18	FALSE	43	0.67	FALSE
16	Canada	6320	0	65	1.03	FALSE	466	7.37	FALSE
17	Israel	4347	0	15	0.35	FALSE	139	3.2	FALSE
18	Norway	4284	0	26	0.61	FALSE	7	0.16	FALSE
19	Brazil	4256	0	136	3.2	FALSE	6	0.14	FALSE
20	Australia	4245	0	18	0.42	FALSE	244	5.75	FALSE
21	Sweden	3700	0	110	2.97	FALSE	16	0.43	FALSE
22	Czech Republic	2817	0	16	0.57	FALSE	11	0.39	FALSE
23	Denmark	2724	0	72	2.64	FALSE	73	2.68	FALSE
24	Malaysia	2626	0	37	1.41	FALSE	479	18.24	FALSE
25	Ireland	2615	0	46	1.76	FALSE	5	0.19	FALSE
26	Chile	2449	0	8	0.33	FALSE	156	6.37	FALSE
27	Luxembourg	1988	0	22	1.11	FALSE	40	2.01	FALSE
28	Romania	1952	0	46	2.36	FALSE	206	10.55	FALSE
29	Ecuador	1924	0	58	3.01	FALSE	3	0.16	FALSE
30	Poland	1905	0	26	1.36	FALSE	7	0.37	FALSE
31	Russia	1836	0	9	0.49	FALSE	66	3.59	FALSE
32	Japan	1724	0	52	3.02	FALSE	372	21.58	FALSE
33	Pakistan	1625	0	20	1.23	FALSE	29	1.78	FALSE
34	Philippines	1546	0	78	5.05	FALSE	42	2.72	FALSE
35	Thailand	1524	0	9	0.59	FALSE	127	8.33	FALSE
36	Saudi Arabia	1453	0	8	0.55	FALSE	115	7.91	FALSE
37	Indonesia	1414	0	122	8.63	FALSE	75	5.3	FALSE
38	South Africa	1280	0	2	0.16	FALSE	31	2.42	FALSE
39	Finland	1240	0	11	0.89	FALSE	10	0.81	FALSE
40	Greece	1156	0	38	3.29	FALSE	52	4.5	FALSE
41	India	1071	0	29	2.71	FALSE	99	9.24	FALSE
42	Iceland	1020	0	2	0.2	FALSE	135	13.24	FALSE
43	Mexico	993	0	20	2.01	FALSE	4	0.4	FALSE
44	Panama	989	0	24	2.43	FALSE	4	0.4	FALSE
45	Dominican Republic	901	0	42	4.66	FALSE	4	0.44	FALSE
46	Singapore	879	0	3	0.34	FALSE	228	25.94	FALSE
47	Peru	852	0	18	2.11	FALSE	16	1.88	FALSE
48	Argentina	820	0	80	9.76	FALSE	72	8.78	FALSE
49	Serbia	785	0	16	2.04	FALSE	15	1.91	FALSE
50	Croatia	713	0	6	0.84	FALSE	55	7.71	FALSE
51	Diamond Princess	712	0	11	1.54	FALSE	603	84.69	FALSE

Les données de l'algerie

Dans le contenu du tableau précédant on peut remarque dans la ligne 63 les différentes informations sur notre pays

60	Iraq	547	0	42	7.68	FALSE	143	26.14	FALSE
61	Morocco	516	0	27	5.23	FALSE	13	2.52	FALSE
62	Bahrain	515	0	4	0.78	FALSE	279	54.17	FALSE
63	Algeria	511	0	31	6.07	FALSE	32	6.26	FALSE
64	Armenia	481	0	3	0.02	FALSE	30	0.24	FALSE
65	Ukraine	480	0	11	2.29	FALSE	6	1.25	FALSE
66

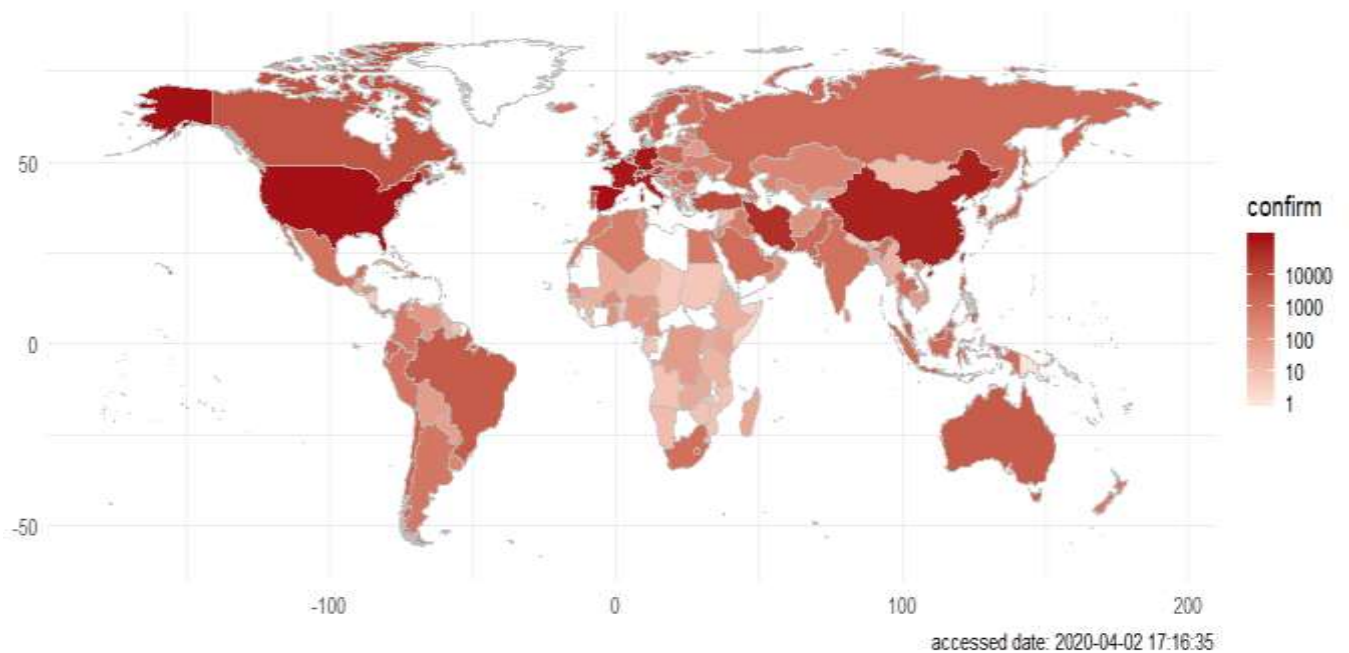
Visualisation de carte géographique

Pour obtenir un tracé de la carte du monde c'est vraiment simple. Juste en quelques lignes comme suit:

```
> x <- get_nCov2019()
There were 37 warnings (use warnings() to see them)
>
> x
China (total confirmed cases): 82773
last update: 2020-04-02 17:16:35
>
> plot(x)
warning: Ignoring unknown aesthetics: x, y
> |
```

COVID-19

confirmed cases: 950463





Explorer les données

Tout d'abord, explorons la structure globale des données actuelles.

DataExplorer est un package R qui peut rapidement créer une visualisation.

```
> library(DataExplorer)
Error in library(DataExplorer) :
  aucun package nommé 'DataExplorer' n'est trouvé
> install.packages("DataExplorer")
WARNING: Rtools is required to build R packages but is not currently installed. Please
  download and install the appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/HP/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
also installing the dependencies 'igraph', 'data.table', 'rmarkdown', 'networkD3'

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/igraph_1.2.5.zip'
Content type 'application/zip' length 9170943 bytes (8.7 MB)
downloaded 8.7 MB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/data.table_1.12.8.zip'
Content type 'application/zip' length 2276588 bytes (2.2 MB)
downloaded 2.2 MB

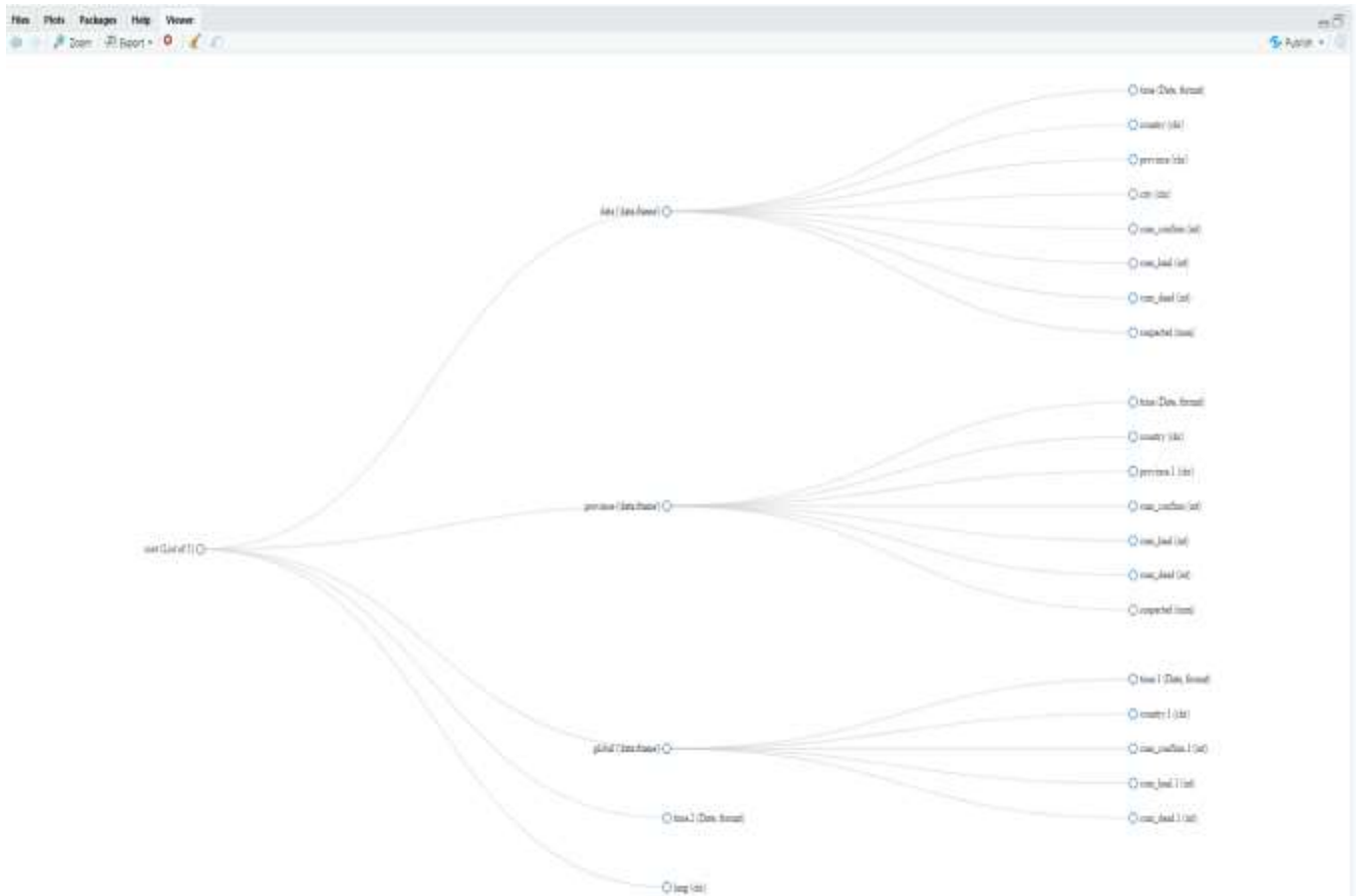
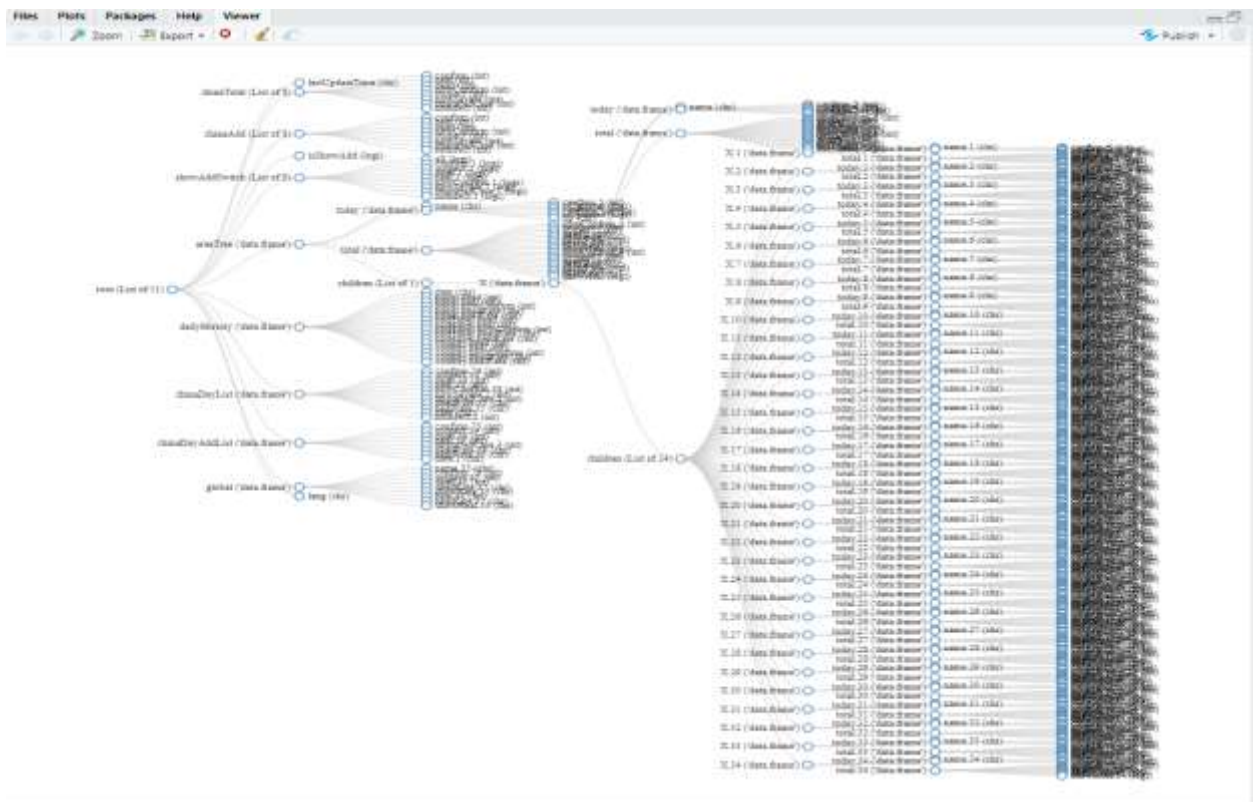
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/rmarkdown_2.1.zip'
Content type 'application/zip' length 3616882 bytes (3.4 MB)
downloaded 3.4 MB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/networkD3_0.4.zip'
Content type 'application/zip' length 196978 bytes (192 KB)
downloaded 192 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/DataExplorer_0.8.1.zip'
Content type 'application/zip' length 3208706 bytes (3.1 MB)
downloaded 3.1 MB

package 'igraph' successfully unpacked and MD5 sums checked
package 'data.table' successfully unpacked and MD5 sums checked
package 'rmarkdown' successfully unpacked and MD5 sums checked
package 'networkD3' successfully unpacked and MD5 sums checked
package 'DataExplorer' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\HP\AppData\Local\Temp\Rtmpkzfzig\downloaded_packages
> plot_str(x)
Error in plot_str(x) : could not find function "plot_str"
> library(DataExplorer)
warning message:
  le package 'DataExplorer' a été compilé avec la version R 3.6.3
> plot_str(x)
> plot_str(y)
> |
```



J'ai utilisé la fonction `summary()` pour obtenir un aperçu statistique des données.

Globalement, la médiane des cas confirmés est de 231 alors que la moyenne est de 4578.1

```
> summary(x['global',])
```

name	confirm	suspect	dead	deadRate	showRate	heal	healRate	showHeal
Length:163	Min. : 1.0	Min. : 0.000	Min. : 0.0	Length:163	Length:163	Min. : 0.0	Length:163	Length:163
Class :character	1st Qu.: 26.5	1st Qu.: 0.000	1st Qu.: 0.0	Class :character	Class :character	1st Qu.: 0.0	Class :character	Class :character
Mode :character	Median : 231.0	Median : 0.000	Median : 2.0	Mode :character	Mode :character	Median : 7.0	Mode :character	Mode :character
	Mean : 4578.1	Mean : 0.546	Mean : 219.4			Mean : 964.1		
	3rd Qu.: 1045.5	3rd Qu.: 0.000	3rd Qu.: 19.0			3rd Qu.: 45.5		
	Max. : 143071.0	Max. : 89.000	Max. : 10779.0			Max. : 77489.0		

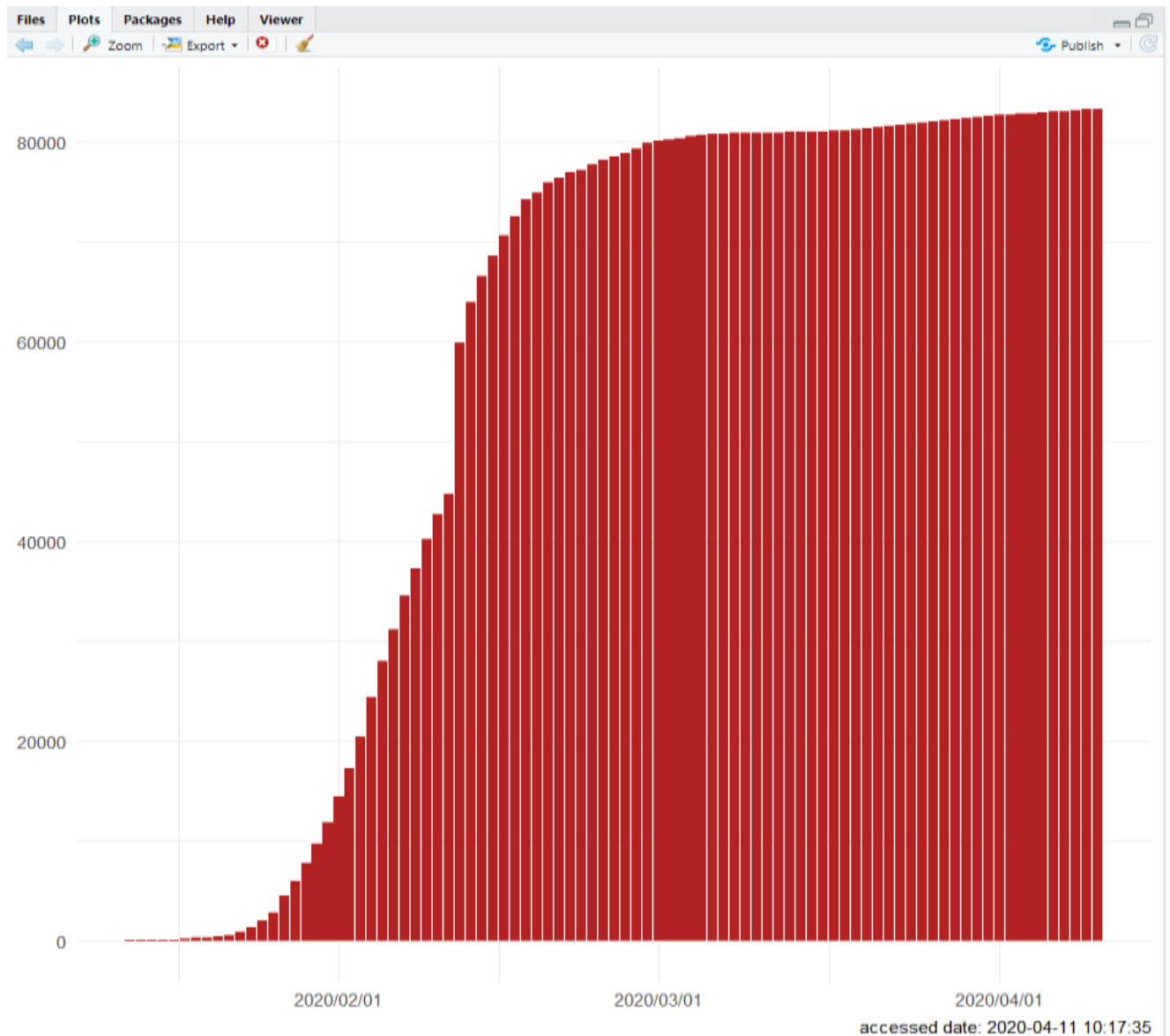


Créer des visuels

La visualisation est une autre méthode clé pour observer facilement la tendance de l'épidémie au niveau national et international.

Si on souhaite visualiser les données récapitulatives cumulatives, voici les commandes utilisées.

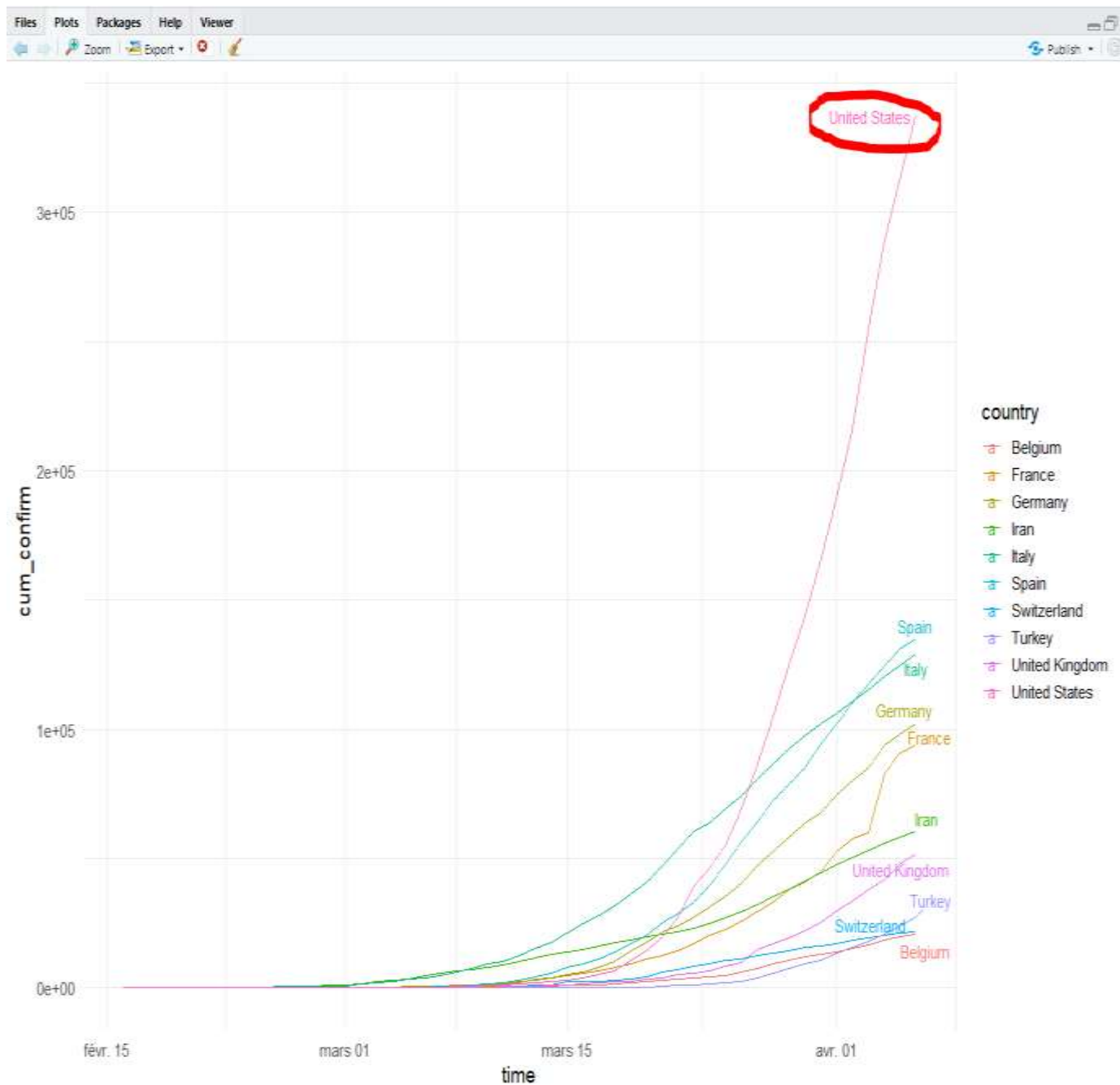
```
> d <- summary(x)
> ggplot(d, aes(as.Date(date, "%m.%d"), as.numeric(confirm))) + geom_col(fill = 'firebrick') + theme_minimal(base_size = 14) + xlab(NULL) + ylab(NULL) + scale_x_date(date_labels = "%Y/%m/%d") + labs(caption = paste("accessed date:", time(x)))
```



Graphique linéaire par pays

Voici un exemple ci dessous qui represente les 10 premiers pays avec les cas les plus confirmés en utilisant x ['global',] et sans prendre en consideration la chine

```
> d <- y ['global']
Warning message:
In readRDS(system.file("country_translate.rds", package = "nCov2019")) :
  strings not representable in native encoding will be translated to UTF-8
> d <- d[d$country != "china",]
> library(tidyverse)
> library(magrittr)
> n <- d %>% filter(time == time(y)) %>% top_n(10, cum_confirm) %>% arrange(desc(cum_confirm))
> require(ggplot2)
> require(ggrepel)
> ggplot(filter(d, country %in% n$country, d$time > "2020/02/15"), aes(time, cum_confirm, color=country)) + geom_line() + geom_text_repel(
  aes(label=country), function(d) d[d$time == time(y),]) + theme_minimal(base_size=14)
> |
```



Ce graphique montre les 10 premiers pays avec les cas les plus confirmés en dehors de la Chine. Les United states et Spain sont les pays les plus infectés et affichent une croissance exponentielle. Pendant ce temps, la Belgique a aplati sa courbe et ralenti la croissance avec une stratégie de confinement efficace.

Le Cas de l'algerie


Nous pouvons également tracer le graphique linéaire pour voir la croissance des cas au fil du temps dans notre pays.

```
> d <- y ['global']
warning message:
In readRDS(system.file("country_translate.rds", package = "nCov2019")) :
  strings not representable in native encoding will be translated to UTF-8
> d <- d[d$country == "Algeria",]
> d
```

	time	country	cum_confirm	cum_heal	cum_dead
769	2020-02-26	Algeria	1	0	0
808	2020-02-27	Algeria	1	0	0
854	2020-02-28	Algeria	1	0	0
908	2020-02-29	Algeria	1	0	0
966	2020-03-01	Algeria	1	0	0
1028	2020-03-02	Algeria	3	0	0
1096	2020-03-03	Algeria	3	0	0
1170	2020-03-04	Algeria	8	0	0
1247	2020-03-05	Algeria	17	0	0
1328	2020-03-06	Algeria	17	0	0
1418	2020-03-07	Algeria	19	0	0
1512	2020-03-08	Algeria	20	0	0
1612	2020-03-09	Algeria	20	0	0
1719	2020-03-10	Algeria	20	0	0
1831	2020-03-11	Algeria	20	0	0
1988	2020-03-12	Algeria	25	8	1
2108	2020-03-13	Algeria	26	10	2
NA	<NA>	<NA>	NA	NA	NA
2242	2020-03-14	Algeria	27	10	2
NA.1	<NA>	<NA>	NA	NA	NA
2387	2020-03-15	Algeria	48	12	4
NA.2	<NA>	<NA>	NA	NA	NA
2540	2020-03-16	Algeria	54	12	4
NA.3	<NA>	<NA>	NA	NA	NA
2697	2020-03-17	Algeria	60	12	4
NA.4	<NA>	<NA>	NA	NA	NA
2857	2020-03-18	Algeria	72	12	5
NA.5	<NA>	<NA>	NA	NA	NA
3024	2020-03-19	Algeria	82	32	8
NA.6	<NA>	<NA>	NA	NA	NA
3194	2020-03-20	Algeria	90	32	10
NA.7	<NA>	<NA>	NA	NA	NA
3369	2020-03-21	Algeria	102	32	9
NA.8	<NA>	<NA>	NA	NA	NA
3549	2020-03-22	Algeria	139	32	15
NA.9	<NA>	<NA>	NA	NA	NA
3732	2020-03-23	Algeria	201	32	17
NA.10	<NA>	<NA>	NA	NA	NA
3918	2020-03-24	Algeria	230	32	17
NA.11	<NA>	<NA>	NA	NA	NA
4108	2020-03-25	Algeria	264	65	19
NA.12	<NA>	<NA>	NA	NA	NA
4300	2020-03-26	Algeria	302	65	21
NA.13	<NA>	<NA>	NA	NA	NA

La suite du tableau précédant est dans l'image suivante

1418	2020-03-07	Algeria	19	0	0
1512	2020-03-08	Algeria	20	0	0
1612	2020-03-09	Algeria	20	0	0
1719	2020-03-10	Algeria	20	0	0
1831	2020-03-11	Algeria	20	0	0
1988	2020-03-12	Algeria	25	8	1
2108	2020-03-13	Algeria	26	10	2
NA	<NA>	<NA>	NA	NA	NA
2242	2020-03-14	Algeria	27	10	2
NA.1	<NA>	<NA>	NA	NA	NA
2387	2020-03-15	Algeria	48	12	4
NA.2	<NA>	<NA>	NA	NA	NA
2540	2020-03-16	Algeria	54	12	4
NA.3	<NA>	<NA>	NA	NA	NA
2697	2020-03-17	Algeria	60	12	4
NA.4	<NA>	<NA>	NA	NA	NA
2857	2020-03-18	Algeria	72	12	5
NA.5	<NA>	<NA>	NA	NA	NA
3024	2020-03-19	Algeria	82	32	8
NA.6	<NA>	<NA>	NA	NA	NA
3194	2020-03-20	Algeria	90	32	10
NA.7	<NA>	<NA>	NA	NA	NA
3369	2020-03-21	Algeria	102	32	9
NA.8	<NA>	<NA>	NA	NA	NA
3549	2020-03-22	Algeria	139	32	15
NA.9	<NA>	<NA>	NA	NA	NA
3732	2020-03-23	Algeria	201	32	17
NA.10	<NA>	<NA>	NA	NA	NA
3918	2020-03-24	Algeria	230	32	17
NA.11	<NA>	<NA>	NA	NA	NA
4108	2020-03-25	Algeria	264	65	19
NA.12	<NA>	<NA>	NA	NA	NA
4300	2020-03-26	Algeria	302	65	21
NA.13	<NA>	<NA>	NA	NA	NA
4494	2020-03-27	Algeria	367	65	25
NA.14	<NA>	<NA>	NA	NA	NA
4691	2020-03-28	Algeria	409	77	26
NA.15	<NA>	<NA>	NA	NA	NA
4891	2020-03-29	Algeria	454	77	29
NA.16	<NA>	<NA>	NA	NA	NA
5092	2020-03-30	Algeria	511	77	31
NA.17	<NA>	<NA>	NA	NA	NA
5294	2020-03-31	Algeria	584	77	35
NA.18	<NA>	<NA>	NA	NA	NA
5498	2020-04-01	Algeria	716	77	44
NA.19	<NA>	<NA>	NA	NA	NA
5702	2020-04-02	Algeria	847	77	58
NA.20	<NA>	<NA>	NA	NA	NA
5906	2020-04-03	Algeria	986	61	86
NA.21	<NA>	<NA>	NA	NA	NA
6111	2020-04-04	Algeria	1171	62	105
NA.22	<NA>	<NA>	NA	NA	NA
6316	2020-04-05	Algeria	1251	90	130
NA.23	<NA>	<NA>	NA	NA	NA
6523	2020-04-06	Algeria	1320	90	152

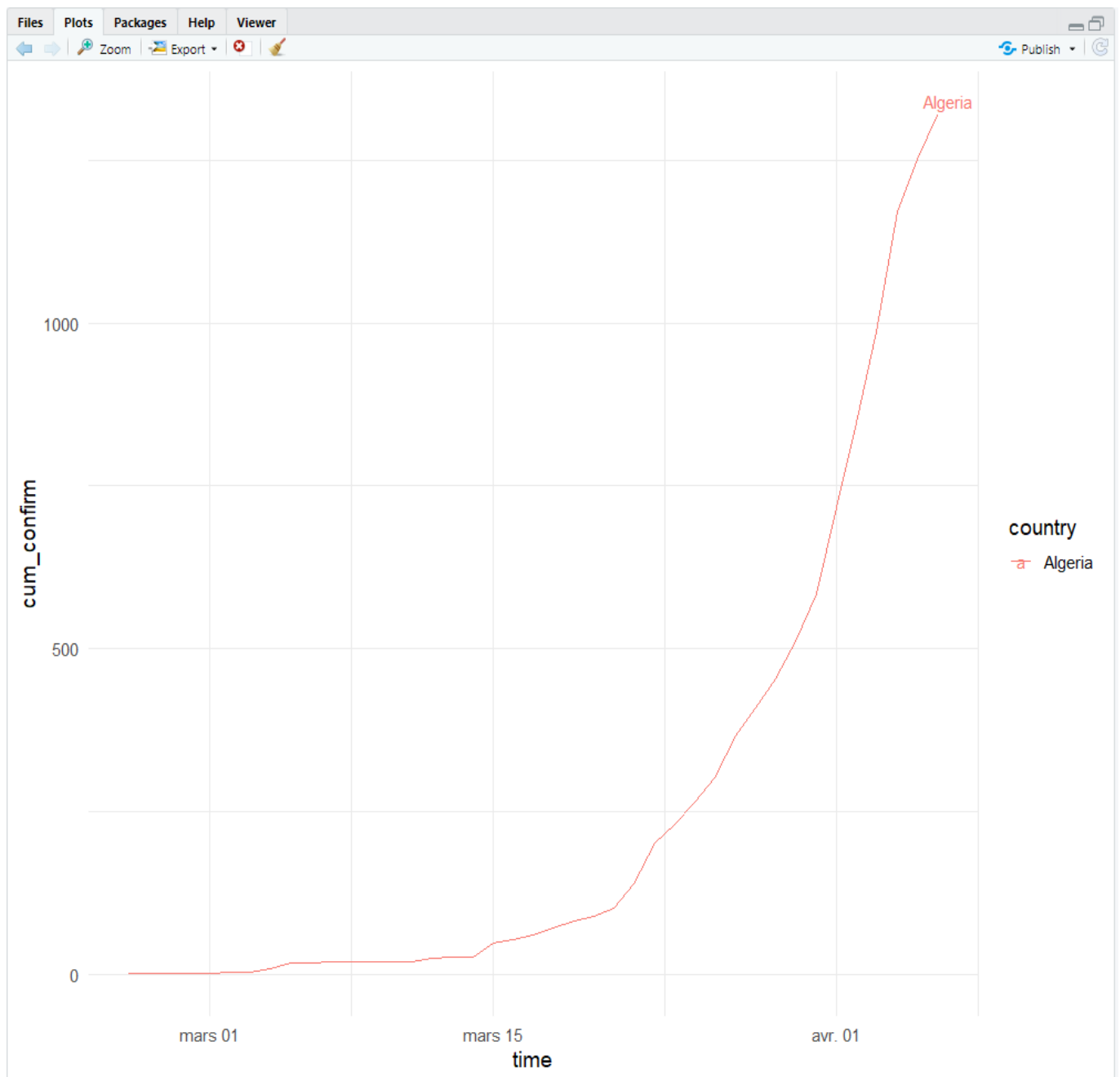


```

> ggplot(filter(d, d$time > "2020/02/15"), aes(time, cum_confirm, color=country)) + geom_line() + geom_text_repel(aes(label=country), function(d) d[d$time == time(y),]) + theme_minimal(base_size=14)
> |

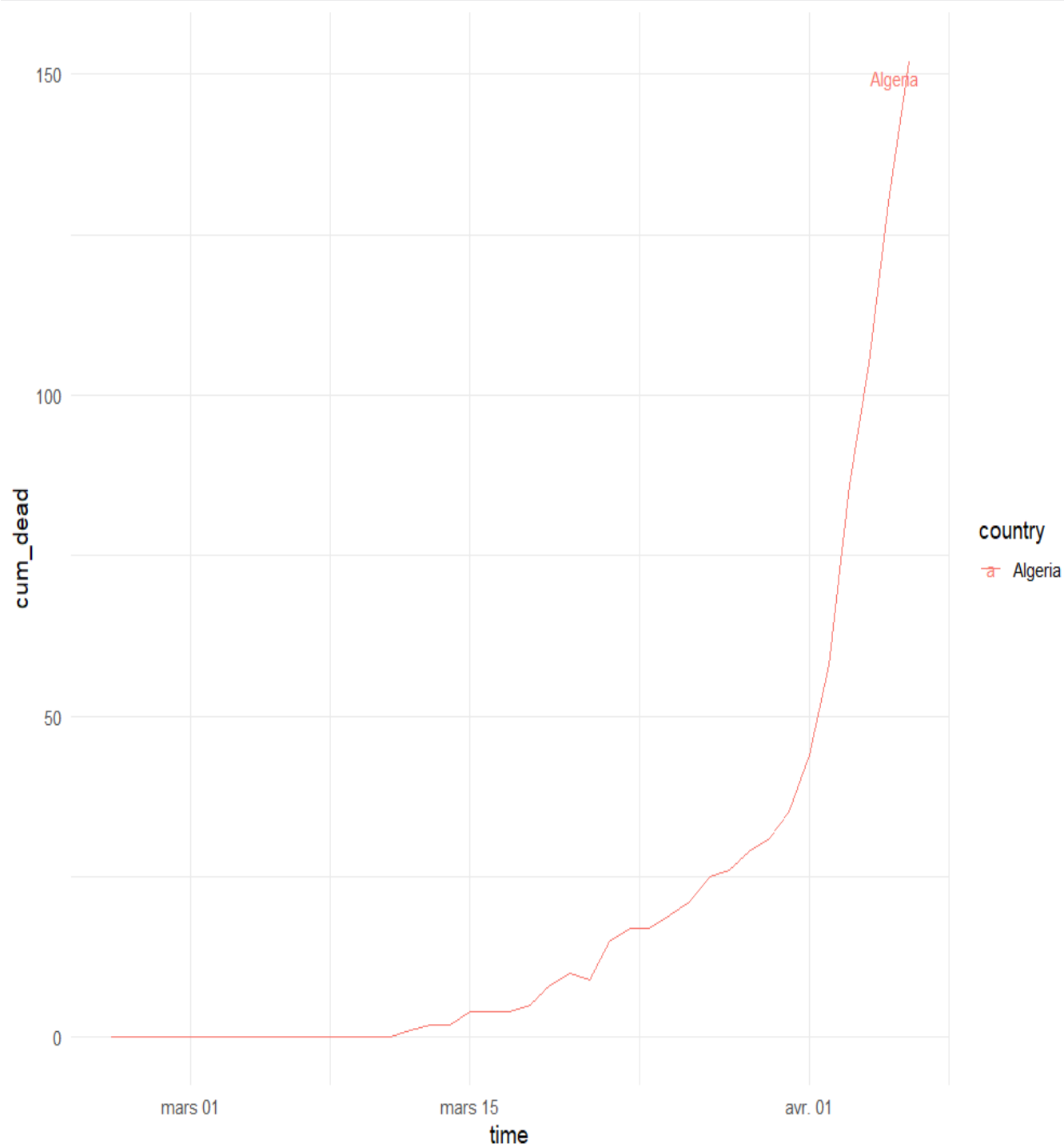
```


l'évolution du nombre de cas confirmés par jours



l'évolution du nombre des décès par jour

```
> ggplot(filter(d, d$time > "2020/02/15"), aes(time, cum_dead, color=country)) + geom_line() + geom_text_repel(
  aes(label=country), function(d) d[d$time == time(y),]) + theme_minimal(base_size=14)
> |
```



Le Cas du monde entier

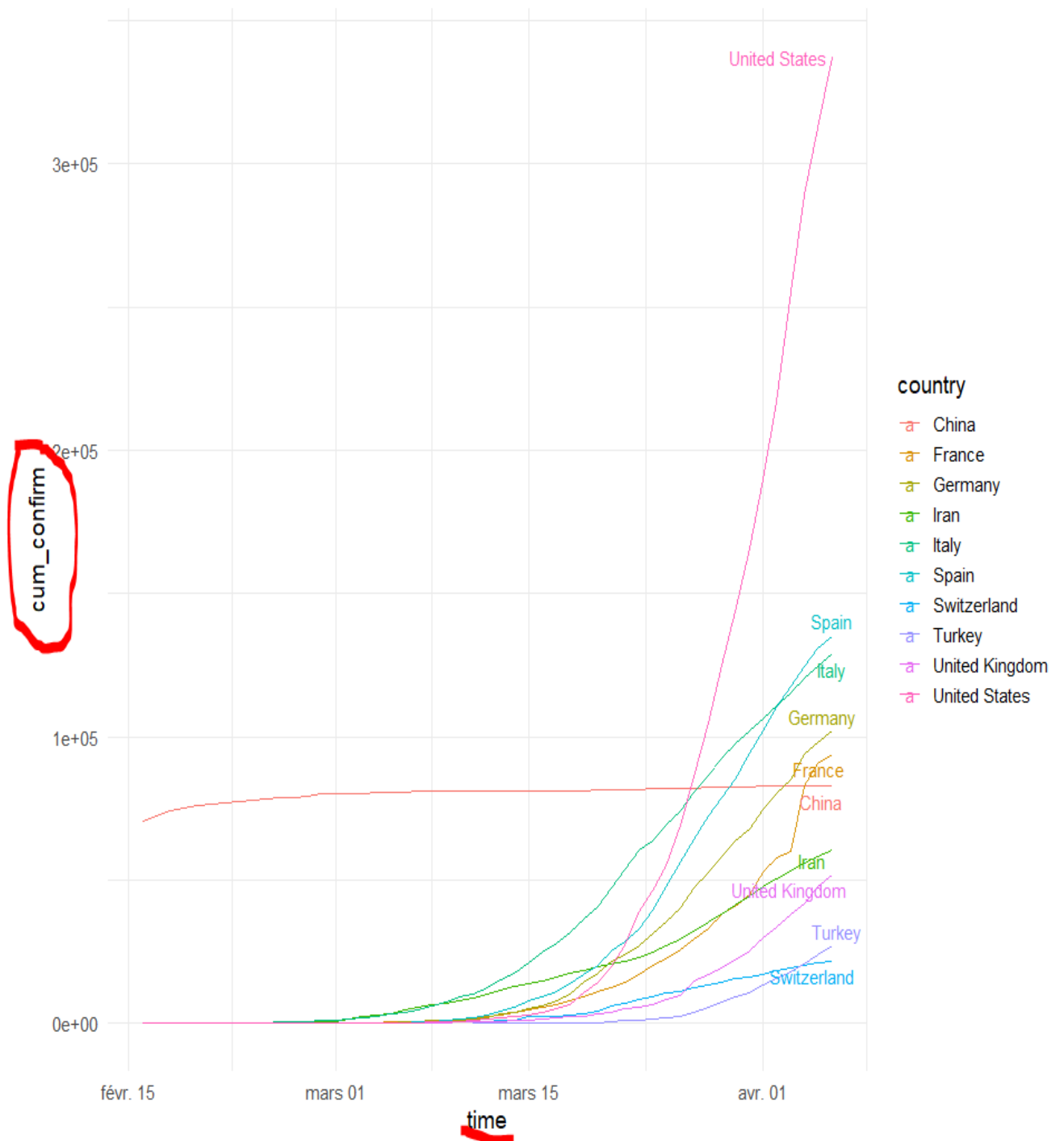
```
> d <- y['global']
warning message:
In readRDS(system.file("country_translate.rds", package = "nCov2019")) :
  strings not representable in native encoding will be translated to UTF-8
> d
```

	time	country	cum_confirm	cum_heal	cum_dead
1	2019-12-01	china	1	0	0
2	2019-12-02	china	1	0	0
3	2019-12-03	china	1	0	0
4	2019-12-04	china	1	0	0
5	2019-12-05	china	1	0	0
6	2019-12-06	china	1	0	0
7	2019-12-07	china	1	0	0
8	2019-12-08	china	1	0	0
9	2019-12-09	china	1	0	0
10	2019-12-10	china	1	0	0
11	2019-12-11	china	4	0	0
12	2019-12-12	china	4	0	0
13	2019-12-13	china	4	0	0

```
> n <- d %>% filter(time == time(y)) %>% top_n(10, cum_confirm) %>% arrange(desc(cum_confirm))
> n
```

	time	country	cum_confirm	cum_heal	cum_dead
1	2020-04-06	United States	337300	17582	9627
2	2020-04-06	Spain	135032	40437	13055
3	2020-04-06	Italy	128948	21815	15887
4	2020-04-06	Germany	101900	38154	1655
5	2020-04-06	France	93780	16354	8093
6	2020-04-06	China	83071	77450	3340
7	2020-04-06	Iran	60500	24236	3739
8	2020-04-06	United Kingdom	51608	229	5373
9	2020-04-06	Turkey	27069	1042	574
10	2020-04-06	Switzerland	21652	7298	724

```
>
> ggplot(filter(d, country %in% n$country, d$time > "2020/02/15"), aes(time, cum_confirm, color=country)) + geom_line()
+ geom_text_repel(aes(label=country), function(d) d[d$time == time(y),]) + theme_minimal(base_size=14)
>
```





Regression

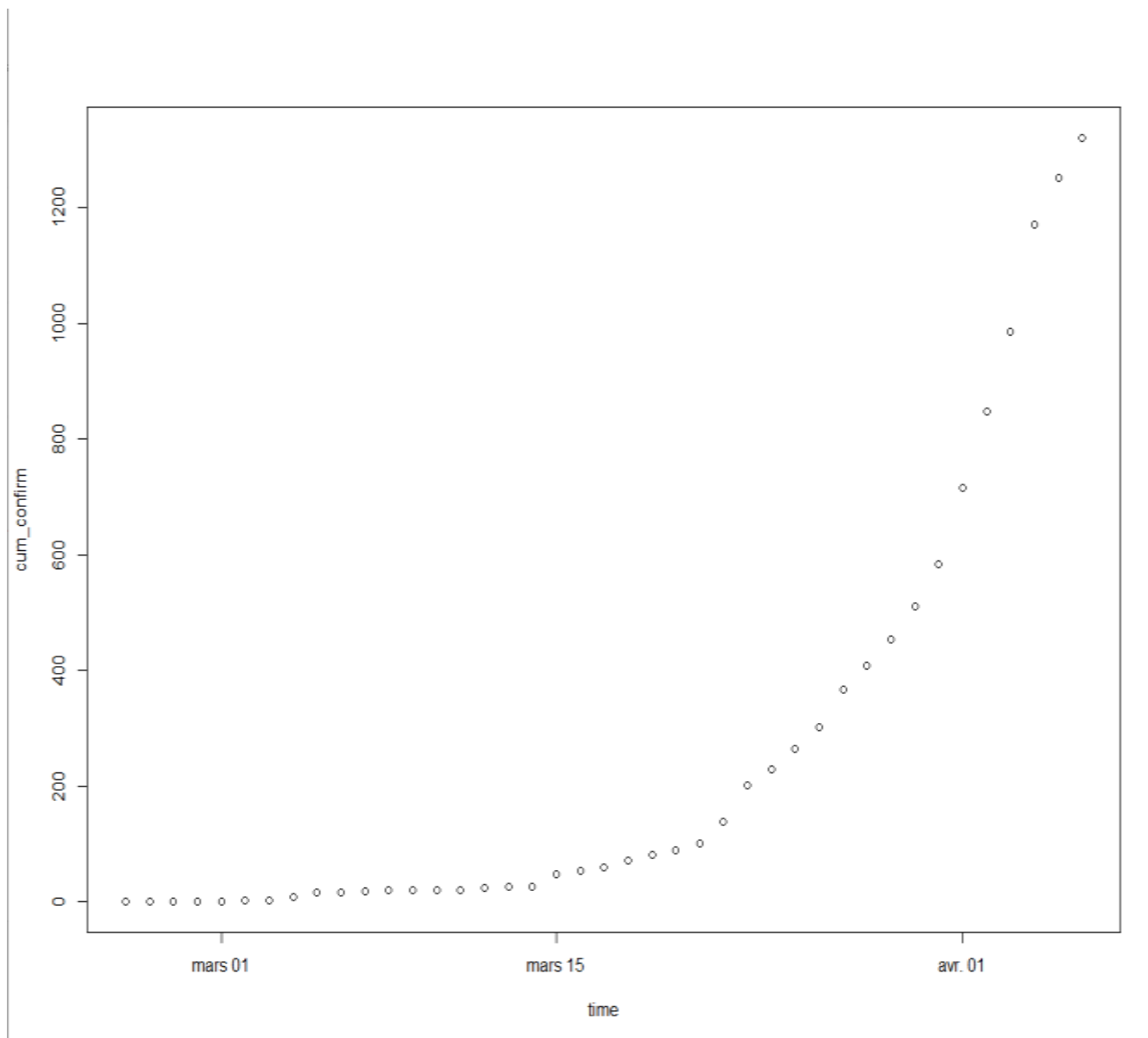
nous pouvons prédire la croissance future des cas avec les données existantes , Prenons l'exemple de l'algerie.

```
> aldata <- d %>% filter(d$country == "Algeria" ) %>% select(time,cum_confirm)
> aldata
```

	time	cum_confirm
1	2020-02-26	1
2	2020-02-27	1
3	2020-02-28	1
4	2020-02-29	1
5	2020-03-01	1
6	2020-03-02	3
7	2020-03-03	3
8	2020-03-04	8
9	2020-03-05	17
10	2020-03-06	17
11	2020-03-07	19
12	2020-03-08	20
13	2020-03-09	20
14	2020-03-10	20
15	2020-03-11	20
16	2020-03-12	25
17	2020-03-13	26
18	2020-03-14	27
19	2020-03-15	48
20	2020-03-16	54
21	2020-03-17	60
22	2020-03-18	72
23	2020-03-19	82
24	2020-03-20	90
25	2020-03-21	102
26	2020-03-22	139
27	2020-03-23	201
28	2020-03-24	230
29	2020-03-25	264
30	2020-03-26	302
31	2020-03-27	367
32	2020-03-28	409
33	2020-03-29	454
34	2020-03-30	511
35	2020-03-31	584
36	2020-04-01	716
37	2020-04-02	847
38	2020-04-03	986
39	2020-04-04	1171
40	2020-04-05	1251
41	2020-04-06	1320

```
> plot(aldata)
> |
```

En regardant le graphique, nous pouvons facilement constater que les cas croissent à un rythme exponentiel.



le package forecast de R fournit des méthodes et des outils pour afficher et analyser des prévisions de séries chronologiques univariées,


```

> library(forecast)
Error in library(forecast) : aucun package nommé 'forecast' n'est trouvé
> install.packages("forecast")
WARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/HP/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
also installing the dependencies 'xts', 'TTR', 'quadprog', 'quantmod', 'fracdiff', 'lmtest', 'timeDate', 'tseries', 'urca', 'zoo', 'RcppArmadillo'

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/xts_0.12-0.zip'
Content type 'application/zip' length 966349 bytes (943 KB)
downloaded 943 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/TTR_0.23-6.zip'
Content type 'application/zip' length 522006 bytes (509 KB)
downloaded 509 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/quadprog_1.5-8.zip'
Content type 'application/zip' length 54859 bytes (53 KB)
downloaded 53 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/quantmod_0.4.17.zip'
Content type 'application/zip' length 964925 bytes (942 KB)
downloaded 942 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/fracdiff_1.5-1.zip'
Content type 'application/zip' length 135055 bytes (131 KB)
downloaded 131 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/lmtest_0.9-37.zip'
Content type 'application/zip' length 363316 bytes (354 KB)
downloaded 354 KB

```

Nous pouvons utiliser forecast() pour faire des prédictions pour les 10 prochains jours , la période h est utilisée pour définir la période de prédiction

```

> library(forecast)
Registered S3 method overwritten by 'quantmod':
  method      from
  as.zoo.data.frame zoo
warning message:
le package 'forecast' a été compilé avec la version R 3.6.3
> fit <- tslm(case ~ trend)
> fc <- forecast(fit, h=10)
> plot(fc)

```

```
> summary(fit)

Call:
tslm(formula = case ~ trend)

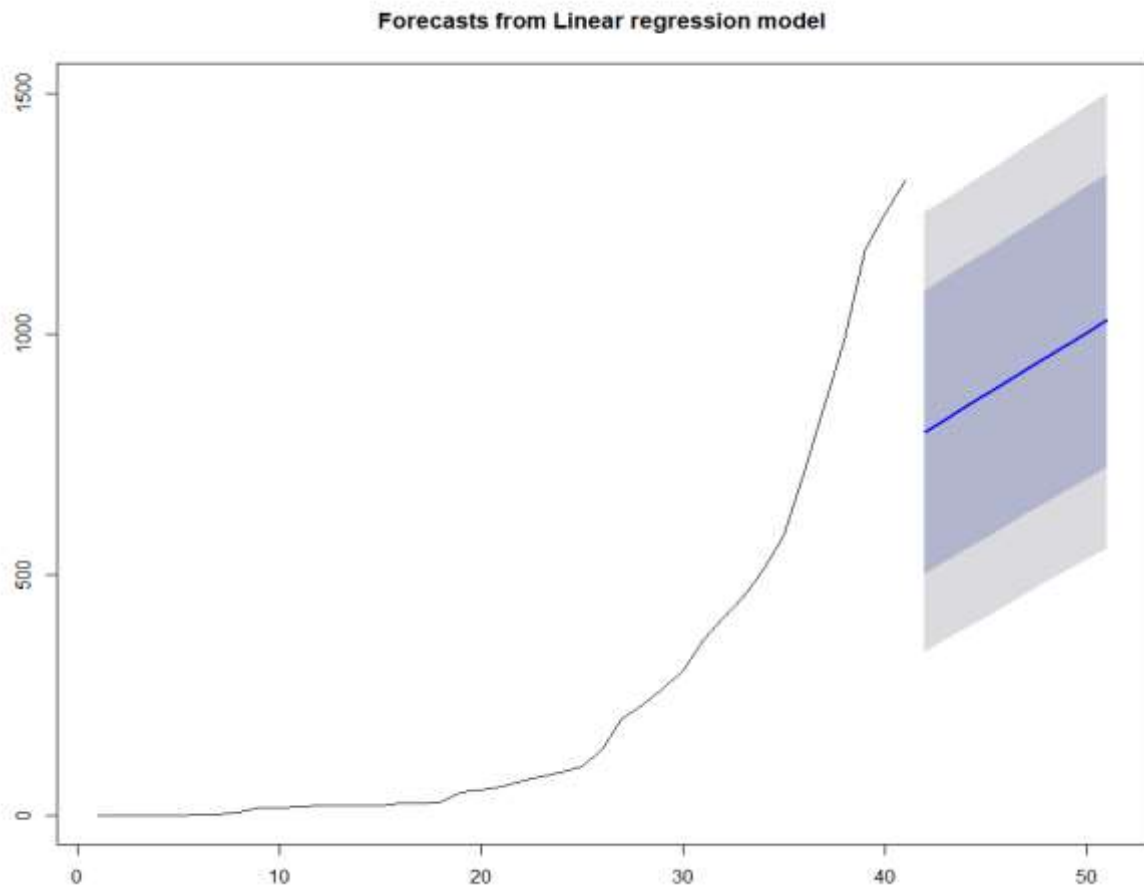
Residuals:
    Min       1Q   Median       3Q      Max
-257.04 -176.06  -55.28  134.09  548.22

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -285.87      68.44  -4.177 0.000161 ***
trend          25.80       2.84   9.084 3.61e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 215.1 on 39 degrees of freedom
Multiple R-squared:  0.6791,    Adjusted R-squared:  0.6709
F-statistic: 82.53 on 1 and 39 DF,  p-value: 3.609e-11

> |
```

Notre modèle log-linéaire est moyennement bon ajustement pour l'ensemble de données. Le carré R ajusté à 0,6791 indique que 67.9% de la variance peut être expliquée par notre modèle.



On peut remarque que la regression ou la suite des points sont decalés.

Dans ce cas, nous pouvons appliquer une régression log-linéaire pour modéliser et prédire la croissance car en regardant le graphique, nous pouvons facilement constater que les cas croissent à un rythme exponentiel.

```
> d <- y['global']
Warning message:
In readRDS(system.file("country_translate.rds", package = "nCov2019")) :
  strings not representable in native encoding will be translated to UTF-8
> aldata <- d %>% filter(d$country == "Algeria" & d$cum_confirm>100) %>% select(time,cum
_confirm)
> case <- ts(usdata[,2], start=1,frequency = 1)
> fit <- tslm(log(case) ~ trend)
> summary(fit)

call:
tslm(formula = log(case) ~ trend)

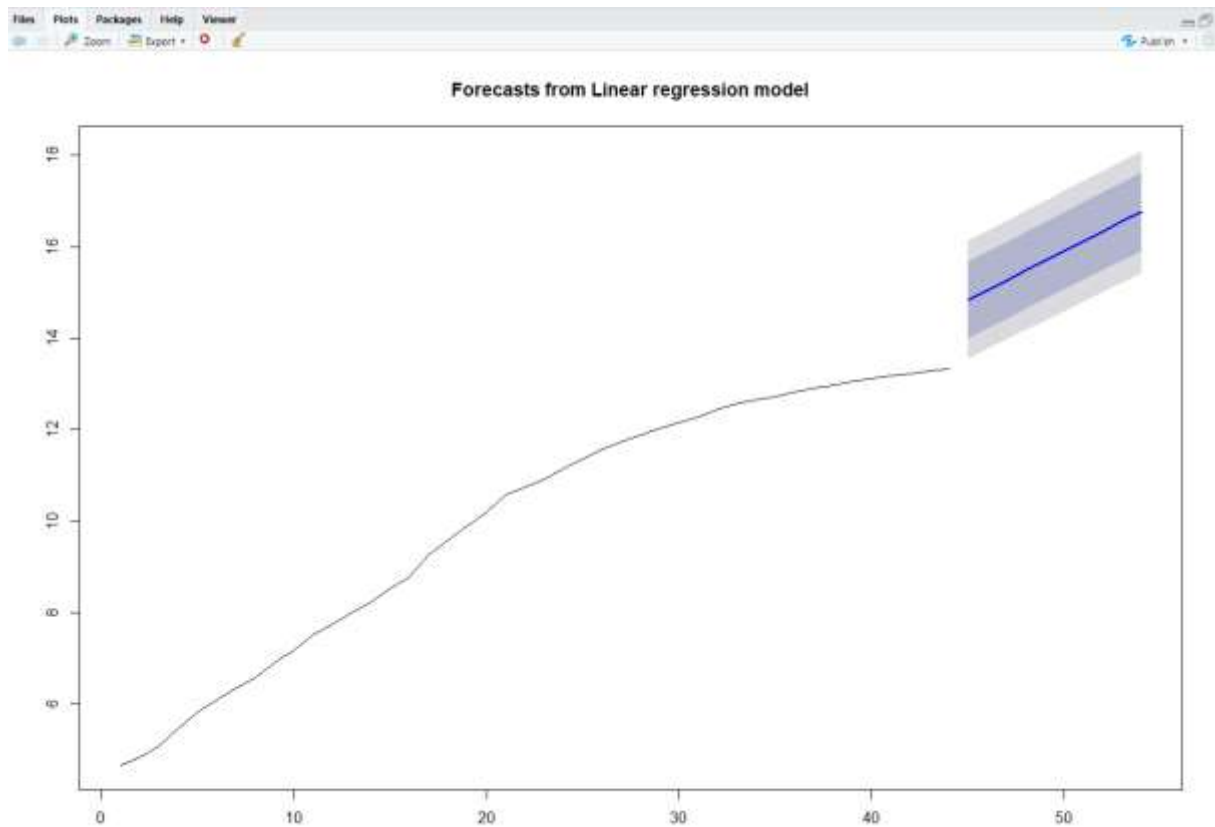
Residuals:
    Min       1Q   Median       3Q      Max
-1.29783 -0.42340  0.01246  0.53816  0.86435

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.223938   0.187626  27.84  <2e-16 ***
trend         0.213700   0.007262  29.43  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6117 on 42 degrees of freedom
Multiple R-squared:  0.9537,    Adjusted R-squared:  0.9526
F-statistic: 865.9 on 1 and 42 DF,  p-value: < 2.2e-16

> fc <- forecast(fit, h=10)
> plot(fc)
```

Voici le resultat graphe



Le resultat d la prediction

```
> fc
  Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
45    14.84046 14.00737 15.67355 13.54922 16.13170
46    15.05416 14.21860 15.88971 13.75910 16.34922
47    15.26786 14.42974 16.10598 13.96882 16.56690
48    15.48156 14.64078 16.32234 14.17840 16.78472
49    15.69526 14.85172 16.53880 14.38782 17.00270
50    15.90896 15.06256 16.75536 14.59709 17.22083
51    16.12266 15.27331 16.97201 14.80621 17.43911
52    16.33636 15.48396 17.18876 15.01519 17.65753
53    16.55006 15.69452 17.40560 15.22402 17.87610
54    16.76376 15.90499 17.62253 15.43271 18.09481
55    16.97746 16.11536 17.83956 15.64126 18.31367
56    17.19116 16.32565 18.05668 15.84967 18.53266
57    17.40486 16.53584 18.27388 16.05793 18.75179
58    17.61856 16.74595 18.49118 16.26606 18.97106
59    17.83226 16.95597 18.70856 16.47406 19.19047
60    18.04596 17.16590 18.92603 16.68192 19.41001
61    18.25967 17.37575 19.14358 16.88965 19.62968
62    18.47337 17.58552 19.36121 17.09725 19.84948
63    18.68707 17.79520 19.57893 17.30472 20.06941
64    18.90077 18.00480 19.79674 17.51207 20.28947
>
```



Etude sur l'âge et le sex

Le coronavirus (ou plus précisément le SRAS-CoV-2) à l'origine de la maladie COVID-19 est le sujet le plus important du dernier jour dans le monde.

nous allons le voir l'effet de l'âge du sexe sur la présence d'une maladie chronique - nous comparerons quatre groupes de pays: l'Asie de l'Est (où le virus a commencé à se propager), l'Europe où, fin mars, il y a eu la plus grande épidémie d'infection, Aux États-Unis où se trouvent maintenant le plus de décès et dans d'autres pays (Amérique du Sud, Afrique, Australie, Océanie et le reste de l'Asie).

On va tout d'abord télécharger les packages nécessaire.

```
> library(tidyverse)
-- Attaching packages ----- tidyverse 1.3.0 --
v ggplot2 3.2.1      v purrr  0.3.3
v tibble  2.1.3      v dplyr  0.8.5
v tidyr   1.0.0      v stringr 1.4.0
v readr   1.3.1      v forcats 0.4.0
-- Conflicts ----- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()     masks stats::lag()
warning message:
le package 'dplyr' a été compilé avec la version R 3.6.3
> library(scales)

Attachement du package : 'scales'

The following object is masked from 'package:purrr':

  discard

The following object is masked from 'package:readr':

  col_factor

> library(RColorBrewer)
> library(ggthemes)
warning message:
le package 'ggthemes' a été compilé avec la version R 3.6.3
> library(gridExtra)

Attachement du package : 'gridExtra'

The following object is masked from 'package:dplyr':

  combine

> library(ggrepel)
> library(lubridate)

Attachement du package : 'lubridate'

The following object is masked from 'package:base':

  date

> library(scales)
> library(RColorBrewer)
> library(ggthemes)
> library(gridExtra)
> library(ggrepel)
> library(lubridate)
```

Il s'agira d'une analyse comparative, dont l'objectif principal est d'examiner les différences de propagation de la maladie dans les régions du monde indiquées. Le projet est régulièrement mis à jour (en moyenne tous les trois jours).

Une base de données supplémentaire a également été chargée dans laquelle nous avons des informations sur le sexe et l'âge des personnes infectées également réparties en groupes, mais ces informations ne s'appliquent qu'à un petit nombre de personnes.

```
> data2 <- read.csv(file.choose(),header = TRUE)
> data2
```

i..ID	age	sex	city	province	country	
1	1	male	Chaohu City,	Hefei City	Anhui	
2	2	47	male	Baohu District,	Hefei City	
3	3	49	male	High-Tech Zone,	Hefei City	
4	4	47	female	High-Tech Zone,	Hefei City	
5	5	50	female	Feidong County,	Hefei City	
6	6	N/A	N/A	Lu'an City	Anhui	
7	7	42	female	Fuyang City	Anhui	
8	8	female	Huabei City	Anhui	China	
9	9	59	female	Huainan City	Anhui	China
10	10	30	male	Hefei City	Anhui	China
11	11	N/A	N/A	Lu'an City	Anhui	China
12	12	39	male	Fuyang City	Anhui	China
13	13	N/A	N/A	Anqing City	Anhui	China
14	14	38	female	Chizhou City	Anhui	China
15	15	45	male	Bengbu City	Anhui	China
16	16	N/A	N/A	Changping District	Beijing	China
17	17	N/A	N/A	Changping District	Beijing	China
18	18	33	female	Changping District	Beijing	China
19	19	female	Daxing District	Beijing	China	
20	20	N/A	N/A	Daxing District	Beijing	China
21	21	37	male	Fengtai District	Beijing	China
22	22	39	male	Haidian District	Beijing	China

wuhan.0._not_wuhan.1.	latitude	longitude	geo_resolution
1	31.64696	117.7166	admin3
2	31.77863	117.3319	admin3
3	31.828313	117.224844	point
4	31.828313	117.224844	point
5	32.00123	117.5681	admin3
6	31.75941	116.3151	admin2
7	32.9188	115.7036	admin2
8	33.72722	116.7424	admin2
9	32.75738	116.734	admin2
10	31.79444	117.3428	admin2
11	31.75941	116.3151	admin2
12	32.9188	115.7036	admin2
13	30.61122	116.5956	admin2
14	30.28525	117.3658	admin2
15	33.10901	117.326	admin2
16	40.20789	116.2063	admin3
17	40.20789	116.2063	admin3
18	40.20789	116.2063	admin3
19	39.64238	116.4172	admin3
20	39.64238	116.4172	admin3
21	39.8306	116.2499	admin3
22	40.02186	116.2285	admin3

date_onset_symptoms	date_admission_hospital	date_confirmation
1	18.01.2020	20.01.2020
2	10.01.2020	21.01.2020
3	15.01.2020	20.01.2020
4	17.01.2020	23.01.2020

A partir de la base de données chargée, nous nous intéressons à 5 variables: la date, le nombre d'infection, de décès et de guérison dans une région donnée, et appartenant aux trois groupes mentionnés ci-dessus. Le nombre de lignes dans la base de données augmente avec le temps.

```
> data2$age <- as.numeric(data2$age)
> data2 <- data2[between(data2$age, 2, 100),]
> data2 <- data2[,c(1,2,3,6)]
> data2$Group <- 0
> data2$Group[data2$country=="Austria"]<-"Europe"
> data2$Group[data2$country=="Albania"]<-"Europe"
> data2$Group[data2$country=="Armenia"]<-"Europe"
> data2$Group[data2$country=="Belgium"]<-"Europe"
> data2$Group[data2$country=="Bulgaria"]<-"Europe"
> data2$Group[data2$country=="Croatia"]<-"Europe"
> data2$Group[data2$country=="Czech Republic"]<-"Europe"
> data2$Group[data2$country=="Denmark"]<-"Europe"
> data2$Group[data2$country=="Estonia"]<-"Europe"
> data2$Group[data2$country=="Finland"]<-"Europe"
> data2$Group[data2$country=="France"]<-"Europe"
> data2$Group[data2$country=="Germany"]<-"Europe"
> data2$Group[data2$country=="Greece"]<-"Europe"
> data2$Group[data2$country=="Hungary"]<-"Europe"
> data2$Group[data2$country=="Iceland"]<-"Europe"
> data2$Group[data2$country=="Ireland"]<-"Europe"
> data2$Group[data2$country=="Italy"]<-"Europe"
> data2$Group[data2$country=="Latvia"]<-"Europe"
> data2$Group[data2$country=="Lithuania"]<-"Europe"
> data2$Group[data2$country=="Netherlands"]<-"Europe"
> data2$Group[data2$country=="Poland"]<-"Europe"
> data2$Group[data2$country=="Portugal"]<-"Europe"
> data2$Group[data2$country=="Romania"]<-"Europe"
> data2$Group[data2$country=="Slovakia"]<-"Europe"
> data2$Group[data2$country=="Slovenia"]<-"Europe"
> data2$Group[data2$country=="Spain"]<-"Europe"
> data2$Group[data2$country=="Sweden"]<-"Europe"
> data2$Group[data2$country=="UK"]<-"Europe"
> data2$Group[data2$country=="Russia"]<-"Europe"
> data2$Group[data2$country=="Switzerland"]<-"Europe"
> data2$Group[data2$country=="Ukraine"]<-"Europe"
> data2$Group[data2$country=="Belarus"]<-"Europe"
> data2$Group[data2$country=="Norway"]<-"Europe"
> data2$Group[data2$country=="San Marino"]<-"Europe"
> data2$Group[data2$country=="Moldova"]<-"Europe"
> data2$Group[data2$country=="Malta"]<-"Europe"
> data2$Group[data2$country=="Liechtenstein"]<-"Europe"
> data2$Group[data2$country=="Luxembourg"]<-"Europe"
> data2$Group[data2$country=="China"]<-"East Asia"
> data2$Group[data2$country=="Mainland China"]<-"East Asia"
> data2$Group[data2$country=="Japan"]<-"East Asia"
> data2$Group[data2$country=="South Korea"]<-"East Asia"
> data2$Group[data2$country=="Mongolia"]<-"East Asia"
>
> data2$Group[data2$Group==0] <- "Rest of world"
```

```
> data2 <- data2[-4]
```

```
> data2
```

	i..ID	age	sex	Group
1	1	41	male	East Asia
2	2	64	male	East Asia
3	3	66	male	East Asia
4	4	64	female	East Asia
5	5	68	female	East Asia
7	7	59	female	East Asia
9	9	78	female	East Asia
10	10	41	male	East Asia
12	12	53	male	East Asia
14	14	51	female	East Asia
15	15	62	male	East Asia
18	18	45	female	East Asia
21	21	50	male	East Asia
22	22	53	male	East Asia
23	23	44	female	East Asia
24	24	62	male	East Asia
25	25	62	male	East Asia
26	26	20	female	East Asia
27	27	75	female	East Asia
28	28	59	male	East Asia
29	29	45	female	East Asia
38	38	61	male	East Asia
39	39	86	male	East Asia
40	40	27	male	East Asia
41	41	58	male	East Asia
56	56	61	female	East Asia
79	79	41	female	East Asia
80	80	92	male	East Asia
81	81	60	female	East Asia
83	83	43	male	East Asia
89	89	60	male	East Asia
90	90	33	male	East Asia
91	91	55	male	East Asia
NA	NA	NA	<NA>	Rest of world
NA.1	NA	NA	<NA>	Rest of world
NA.2	NA	NA	<NA>	Rest of world
NA.3	NA	NA	<NA>	Rest of world
NA.4	NA	NA	<NA>	Rest of world
NA.5	NA	NA	<NA>	Rest of world
NA.6	NA	NA	<NA>	Rest of world
99	99	87	male	East Asia
100	100	86	female	East Asia
101	101	48	male	East Asia
102	102	11	male	East Asia
103	103	84	female	East Asia
104	104	50	female	East Asia
120	120	99	female	East Asia
121	121	65	female	East Asia
122	122	87	female	East Asia
NA.7	NA	NA	<NA>	Rest of world
NA.8	NA	NA	<NA>	Rest of world
NA.9	NA	NA	<NA>	Rest of world
NA.10	NA	NA	<NA>	Rest of world

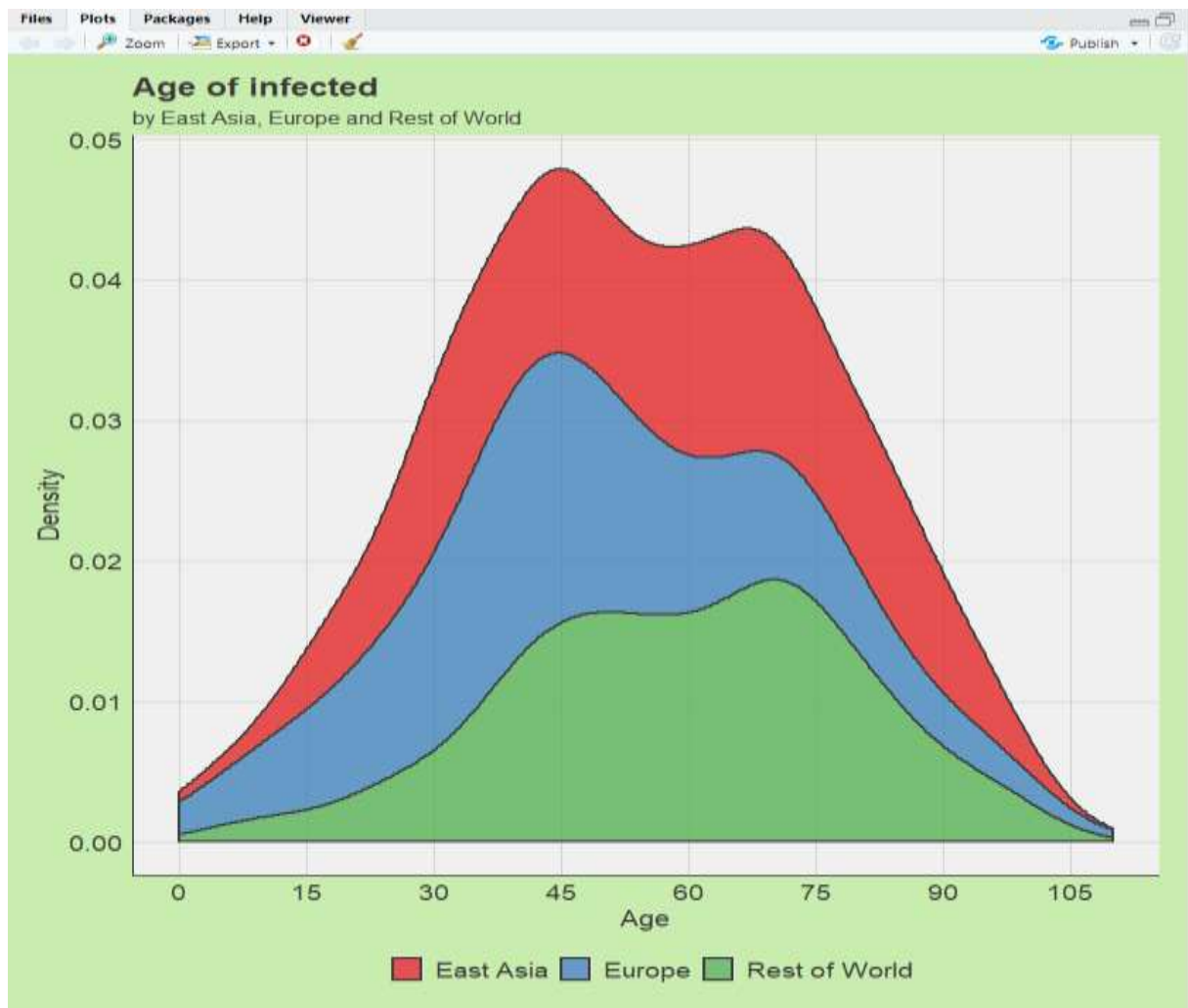
Nous passons maintenant à l'analyse de la deuxième base de données, dans laquelle nous avons des informations détaillées sur certaines personnes infectées réparties dans les mêmes groupes.

```
> head2 <- data2[sample(1:nrow(data2),5), ]
> head2 <- head2[order(head2$ID),]
Error in order(head2$ID) : argument 1 is not a vector
> head2
```

i..ID	age	sex	Group
2966	2994	34 female	Rest of world
5318	5388	32	East Asia
6622	6899	7 male	Rest of world
674	680	48 male	Rest of world
187	193	75 male	East Asia

```
> ggplot(data2, aes(age, fill = Group))+ geom_density(alpha = 0.75, position = "stack", size = 0.8, show.legend = T, col = "gray25")+ scale_x_continuous(limits = c(0,110), breaks = seq(0,110,15))+ scale_fill_brewer(palette = "Set1")+ labs(title = "Age of infected", subtitle = "by East Asia, Europe and Rest of world", y = "Density", x = "Age", fill = "")+ theme_fivethirtyeight()+ theme(legend.position="bottom", legend.direction="horizontal", axis.text = element_text(size = 14), axis.title = element_text(size = 15), legend.text = element_text(size = 14), axis.line = element_line(size = 0.4, colour = "grey10"), plot.background = element_rect(fill = "#C8EDAF"), legend.background = element_rect(fill = "#C8EDAF"))
Warning message:
Removed 30 rows containing non-finite values (stat_density).
> |
```

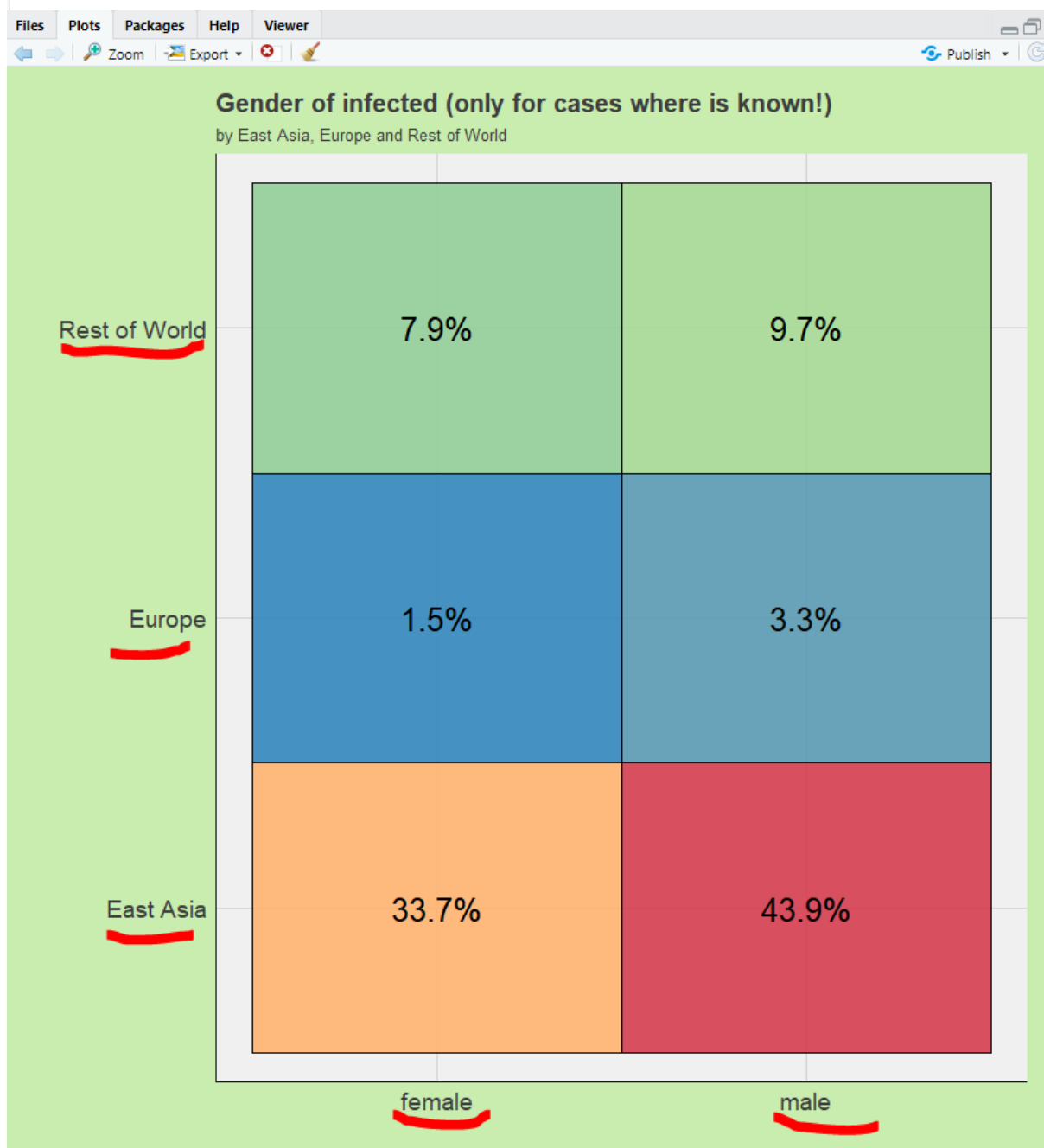
Le premier est l'âge des personnes infectées. nous créons donc 3 courbes de densité pour chaque groupe séparément Dans tous les cas, nous constatons une distribution bimodale avec des «crêtes» autour de 45 et 65 ans. Par conséquent, le plus souvent, le virus est détecté chez les personnes d'âge moyen, tandis que beaucoup moins souvent chez les jeunes et les personnes âgées. Il y a de légers changements dans la répartition entre les groupes, cependant, ils sont si petits que nous ne pouvons pas dire que l'âge des personnes infectées dépend de la région du monde.



```

> to_tile <- data2 %>% group_by(sex, Group) %>% summarise(count = n())
Warning message:
Factor `sex` contains implicit NA, consider using `forcats::fct_explicit_na`
> to_tile <- to_tile[c(4:6,8:10),]
> ggplot(to_tile, aes(sex, Group)) +geom_tile(aes(fill = count), colour = "black", alpha = 0.9)+geom_text(aes(label = paste0(round(count/sum(to_tile$count) * 100,1), "%")), size = 6.8)+scale_fill_distiller(palette = "spectral")+labs(title = "Gender of infected (only for cases where is known!)", subtitle = "by East Asia, Europe and Rest of world", y = "", x = "", fill = "")+theme_few()+theme(legend.position="none", axis.text = element_text(size = 14), axis.title = element_text(size = 15), legend.text = element_text(size = 14), axis.line = element_line(size = 0.4, colour = "grey10"), title = element_text(size = 10), plot.background = element_rect(fill = "#C8EDAF"), legend.background = element_rect(fill = "#C8EDAF"))
> |

```



Prediction de la variable de sortie.

Prédire selon le modèle que vous avez construit votre variable de sortie. voici le tableau qui resume la prediction faite par la fonction forecast()

```
> fc
  Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
45    14.84046 14.00737 15.67355 13.54922 16.13170
46    15.05416 14.21860 15.88971 13.75910 16.34922
47    15.26786 14.42974 16.10598 13.96882 16.56690
48    15.48156 14.64078 16.32234 14.17840 16.78472
49    15.69526 14.85172 16.53880 14.38782 17.00270
50    15.90896 15.06256 16.75536 14.59709 17.22083
51    16.12266 15.27331 16.97201 14.80621 17.43911
52    16.33636 15.48396 17.18876 15.01519 17.65753
53    16.55006 15.69452 17.40560 15.22402 17.87610
54    16.76376 15.90499 17.62253 15.43271 18.09481
55    16.97746 16.11536 17.83956 15.64126 18.31367
56    17.19116 16.32565 18.05668 15.84967 18.53266
57    17.40486 16.53584 18.27388 16.05793 18.75179
58    17.61856 16.74595 18.49118 16.26606 18.97106
59    17.83226 16.95597 18.70856 16.47406 19.19047
60    18.04596 17.16590 18.92603 16.68192 19.41001
61    18.25967 17.37575 19.14358 16.88965 19.62968
62    18.47337 17.58552 19.36121 17.09725 19.84948
63    18.68707 17.79520 19.57893 17.30472 20.06941
64    18.90077 18.00480 19.79674 17.51207 20.28947
> |
```




Régression logistique

La régression logistique est un dérivé de la régression linéaire où nous sommes intéressés à faire des prédictions binaires ou des prédictions de probabilité sur l'intervalle $[0, 1]$ avec une probabilité de seuil pour déterminer où nous séparons entre 0 et 1. La courbe ROC ou "caractéristique de fonctionnement du récepteur" La courbe est une méthode d'évaluation que nous pouvons utiliser pour évaluer l'efficacité d'un algorithme de classification binaire (Receiver Operating Characteristic) .

Dans cette partie on va traiter le cas du bon et mauvais client

Veuller voir le scripte pour comprendre les etapes