

## Covid data per US State

###Load the Data, give summary of data set, then use various data mining methods

```
dat1=read.csv("covid_project1.csv", header=TRUE, sep=',', dec='.', stringsAsFactors=FALSE)
dat1
```

##	State	Total.Covid.19.Deaths	Total.Covid.19.Cases
## 1	Alabama	10,686	518,000
## 2	Alaska	299	64,259
## 3	Arizona	17,062	848,000
## 4	Arkansas	5,661	332,000
## 5	California	60,245	3,690,000
## 6	Colorado	6,203	479,00
## 7	Connecticut	7,944	322,000
## 8	Delaware	1,573	97,784
## 9	Florida	33,967	2,110,000
## 10	Georgia	18,825	1,050,000
## 11	Hawaii	467	30,844
## 12	Idaho	1,995	183,000
## 13	Illinois	23,762	1,280,000
## 14	Indiana	13,131	699,000
## 15	Iowa	5,849	356,00
## 16	Kansas	4,938	307,000
## 17	Kentucky	6,328	436,000
## 18	Louisiana	10,216	449,000
## 19	Maine	750	53,434
## 20	Maryland	8,424	424,000
## 21	Massachusetts	17,376	655,000
## 22	Michigan	17,471	812,000
## 23	Minnesota	7,010	538,000
## 24	Mississippi	7,086	307,000
## 25	Missouri	9,000	591,000
## 26	Montana	1,521	106,000
## 27	Nebraska	2,328	214,000
## 28	Nevada	5,323	307,000
## 29	New Hampshire	1,253	87,978
## 30	New Jersey	24,826	947,000
## 31	New Mexico	3,978	193,000
## 32	New York	50,465	1,940,000
## 33	North Carolina	12,273	935,000
## 34	North Dakota	1,501	105,000
## 35	Ohio	18,827	1,040,000
## 36	Oklahoma	6,669	443,000
## 37	Oregon	2,451	169,000
## 38	Pennsylvania	25,421	1,070,000
## 39	Rhode Island	2,638	141,000
## 40	South Carolina	9,248	561,000
## 41	South Dakota	1,946	119,000

## 42	Tennessee	11,900	808,000	
## 43	Texas	49,273	2,820,000	
## 44	Utah	2,157	389,000	
## 45	Vermont	231	20,815	
## 46	Virginia	10,451	633,000	
## 47	Washington	5,373	377,00	
## 48	West Virginia	2,737	145,000	
## 49	Wisconsin	7,376	644,000	
## 50	Wyoming	701	56,873	
##	weekly.average.cases	vaccines.distributed	vaccinated	Total.Population
## 1	312	3,248,260	2,021,334	4,903,185
## 2	222	703,015	448,337	731,545
## 3	730	5,127,405	3,975,606	7,278,717
## 4	156	2,121,930	1,405,745	3,017,825
## 5	2,488	28,532,520	21,725,654	39,512,223
## 6	1,481	4,016,000	3,184,749	5,758,736
## 7	1,044	2,972,115	2,340,942	3,565,287
## 8	317	720,505	549,853	973,764
## 9	5,711	15,556,985	11,367,878	21,477,737
## 10	870	7,067,975	4,660,059	10,617,423
## 11	99	1,136,660	816,481	1,415,872
## 12	233	1,149,805	873,222	1,787,065
## 13	3,012	9,110,565	7,017,850	12,671,821
## 14	1,017	4,227,610	3,229,355	6,732,219
## 15	526	2,186,505	1,807,025	3,155,070
## 16	201	2,124,855	1,558,508	2,913,314
## 17	518	3,074,595	2,471,143	4,467,673
## 18	371	3,159,400	2,239,421	4,648,794
## 19	1,880	1,008,240	856,186	1,344,212
## 20	1,271	4,384,740	3,414,598	6,045,680
## 21	315	5,211,830	4,345,857	6,949,503
## 22	7,226	6,895,860	5,223,390	9,986,857
## 23	1,931	3,752,100	3,260,958	5,639,632
## 24	205	2,031,745	1,312,516	2,976,149
## 25	599	4,222,465	3,020,648	6,137,428
## 26	145	805,675	609,984	1,068,778
## 27	270	1,388,380	1,107,623	1,934,408
## 28	1,103	2,001,620	1,576,189	3,080,156
## 29	415	964,805	889,074	1,359,711
## 30	4,007	6,243,165	5,531,413	8,882,190
## 31	186	1,679,965	1,448,347	2,096,829
## 32	7,289	14,162,005	11,519,440	19,453,561
## 33	1,532	7,283,460	5,402,133	10,488,084
## 34	151	546,570	474,619	762,062
## 35	1,935	8,210,715	6,308,301	11,689,100
## 36	309	3,099,955	2,235,462	3,956,971
## 37	467	2,903,425	2,211,938	4,217,737
## 38	4,328	9,242,995	6,920,672	12,801,989
## 39	588	779,635	667,244	1,059,361
## 40	982	3,452,035	2,539,519	5,148,714
## 41	174	727,915	571,138	884,659
## 42	1,051	4,505,880	3,122,632	6,833,174
## 43	3,239	18,984,205	14,219,706	28,995,881
## 44	455	1,942,410	1,614,798	3,205,958

## 45	164	501,360	394,697	623,989
## 46	1,473	5,850,555	4,849,229	8,535,519
## 47	1,064	5,380,020	4,256,546	7,614,893
## 48	370	1,330,625	995,174	1,792,147
## 49	835	3,960,825	3,461,130	5,822,434
## 50	58	447,855	289,028	578,759
##	Covid.Cases.65.	Covid.Cases.under.65	state.provided.age.case.data.	recovered
## 1	67,923	335,415	Yes	325,733
## 2	8,870	53,291	Yes	0
## 3	113,068	733,977	Yes	0
## 4	48,900	282,804	Yes	324,582
## 5	379,037	3,203,925	Yes	0
## 6	0	0	No	0
## 7	66,158	254,619	Yes	0
## 8	0	0	No	0
## 9	302,496	1,768,260	Yes	0
## 10	165,730	689,647	Yes	0
## 11	5,004	24,212	Yes	0
## 12	0	0	No	0
## 13	253,118	1,019,617	Yes	1,251,293
## 14	0	0	No	0
## 15	50,637	203,428	Yes	0
## 16	35,185	195,708	Yes	0
## 17	94,638	337,369	Yes	50,372
## 18	0	0	No	0
## 19	11,829	41,605	Yes	0
## 20	0	0	No	0
## 21	0	0	No	0
## 22	0	0	No	0
## 23	100,587	439,078	Yes	0
## 24	0	0	No	296,569
## 25	19,278	100,731	Yes	0
## 26	0	0	No	103,540
## 27	0	0	No	0
## 28	0	0	No	0
## 29	17,222	70,681	Yes	83,501
## 30	0	0	No	0
## 31	33,854	159,373	Yes	174,849
## 32	0	0	No	0
## 33	133,780	788,599	Yes	0
## 34	0	0	No	101,993
## 35	0	0	No	981,253
## 36	63,028	376,958	Yes	423,402
## 37	0	0	No	0
## 38	261,247	807,395	Yes	962,076
## 39	0	0	No	0
## 40	0	0	No	0
## 41	25,895	93,810	Yes	115,346
## 42	0	0	No	797,011
## 43	0	0	No	0
## 44	31,311	358,415	Yes	389,760
## 45	0	0	No	17,266
## 46	119,125	497,199	Yes	0
## 47	0	0	No	0

## 48	0	0	No	0
## 49	0	0	No	569,194
## 50	0	0	No	47,126
##	State.provided.recovered.data.	total.number.of.tests	daily.positive.	
## 1	Yes	0	7.33%	
## 2	No	1,966,048	3.17%	
## 3	No	8,761,941	10.60%	
## 4	Yes	3,437,770	0	
## 5	No	56,078,860	1.70%	
## 6	No	2,788,638	4.97%	
## 7	No	8,002,182	3.37%	
## 8	No	673,728	5.40%	
## 9	No	0	6.94%	
## 10	No	7,975,847	4.50%	
## 11	No	0	1.70%	
## 12	No	0	5.07%	
## 13	Yes	21,102,407	0%	
## 14	No	9,187,249	4.08%	
## 15	No	4,580,733	0%	
## 16	No	1,335,069	2.70%	
## 17	Yes	5,173,076	2.99%	
## 18	No	6,671,092	0%	
## 19	No	2,164,556	2.68%	
## 20	No	0	5.40%	
## 21	No	0	2.30%	
## 22	No	12,514,712	0%	
## 23	No	8,567,524	0%	
## 24	Yes	2,633,209	0%	
## 25	No	6,286,100	4.70%	
## 26	Yes	0	0%	
## 27	No	1,017,623	0%	
## 28	No	1,427,948	4.90%	
## 29	Yes	712,674	0%	
## 30	No	12,578,312	0%	
## 31	Yes	3,079,551	0%	
## 32	No	47,244,227	4.10%	
## 33	No	11,641,898	4.60%	
## 34	Yes	1,841,920	5.45%	
## 35	Yes	0	0%	
## 36	Yes	0	0%	
## 37	No	4,300,000	4.10%	
## 38	Yes	12,148,589	0%	
## 39	No	3,680,071	2.40%	
## 40	No	6,990,408	4.50%	
## 41	Yes	452,497	8.00%	
## 42	Yes	9,051,099	0%	
## 43	No	0	0%	
## 44	Yes	2,442,968	0%	
## 45	Yes	365,190	1.90%	
## 46	No	8,700,856	6.60%	
## 47	No	6,115,030	3.70%	
## 48	No	2,541,058	4.16%	
## 49	Yes	3,353,928	3.80%	
## 50	Yes	207,049	2.70%	

##	State.provided.total.test.data.	PCR.	PCR..1	antigen.	antigen..1
## 1	No	0	0	0	0
## 2	Yes	0	0	0	0
## 3	Yes	0	0	0	0
## 4	Yes	260,450	2,638,124	84,643	454,553
## 5	Yes	0	0	0	0
## 6	Yes	0	0	0	0
## 7	Yes	0	0	0	0
## 8	Yes	0	0	0	0
## 9	No	0	0	0	0
## 10	Yes	834,801	0	211,288	0
## 11	No	0	0	0	0
## 12	No	0	0	0	0
## 13	Yes	0	0	0	0
## 14	Yes	0	0	0	0
## 15	Yes	320,212	2,646,531	65,452	1,530,743
## 16	Yes	0	0	0	0
## 17	Yes	0	0	0	0
## 18	Yes	0	0	0	0
## 19	Yes	42,621	1,812,581	14,503	274,853
## 20	No	0	0	0	0
## 21	No	0	0	0	0
## 22	Yes	0	0	0	0
## 23	Yes	507,098	33,179	0	0
## 24	Yes	0	0	0	0
## 25	Yes	0	0	0	0
## 26	No	0	0	0	0
## 27	Yes	0	0	0	0
## 28	Yes	0	0	0	0
## 29	Yes	0	0	0	0
## 30	Yes	0	0	0	0
## 31	Yes	0	0	0	0
## 32	Yes	0	0	0	0
## 33	Yes	800,770	0	128,636	0
## 34	Yes	0	0	0	0
## 35	No	0	0	0	0
## 36	No	0	0	0	0
## 37	Yes	0	0	0	0
## 38	Yes	0	0	0	0
## 39	Yes	0	0	0	0
## 40	Yes	0	0	0	0
## 41	Yes	0	0	0	0
## 42	Yes	784,564	6,561,563	154,237	1,550,735
## 43	No	0	0	0	0
## 44	Yes	0	0	0	0
## 45	Yes	0	0	0	0
## 46	Yes	0	0	0	0
## 47	Yes	0	0	0	0
## 48	Yes	0	0	0	0
## 49	Yes	0	0	0	0
## 50	Yes	0	0	0	0
##	Total..	Total...1	Daily.positive..given.	PCR.antigen.data.given.	
## 1	0	0	Yes	No	
## 2	0	0	Yes	No	

## 3	0	0	Yes	No
## 4	345,093	3,092,677	No	Yes
## 5	0	0	Yes	No
## 6	0	0	Yes	No
## 7	321,586	7,680,596	Yes	No
## 8	98,200	575,525	Yes	No
## 9	2,078,784	0	Yes	No
## 10	1,046,089	6,929,758	Yes	Yes
## 11	0	0	Yes	No
## 12	0	0	Yes	No
## 13	0	0	No	No
## 14	697,533	8,489,716	Yes	No
## 15	385,664	4,177,274	No	Yes
## 16	304,236	1,030,833	Yes	No
## 17	432,720	4,740,356	Yes	No
## 18	382,477	6,288,615	No	No
## 19	57,124	2,087,434	Yes	Yes
## 20	540,277	3,229,264	Yes	No
## 21	0	0	Yes	No
## 22	0	0	No	No
## 23	540,277	8,027,247	No	Yes
## 24	0	0	No	No
## 25	0	0	Yes	No
## 26	0	0	No	No
## 27	214,010	803,613	No	No
## 28	0	0	Yes	No
## 29	0	0	No	No
## 30	0	0	No	No
## 31	0	0	No	No
## 32	1,934,640	45,309,587	Yes	No
## 33	929,406	10,712,492	Yes	Yes
## 34	104,701	1,737,219	Yes	No
## 35	0	0	No	No
## 36	0	0	No	No
## 37	245,000	4,000,000	Yes	No
## 38	0	4,220,187	No	No
## 39	141,097	3,538,974	Yes	No
## 40	470,805	6,519,603	Yes	No
## 41	0	0	Yes	No
## 42	938,801	8,112,298	No	Yes
## 43	0	0	No	No
## 44	0	0	No	No
## 45	0	0	Yes	No
## 46	0	0	Yes	No
## 47	0	0	Yes	No
## 48	0	0	Yes	No
## 49	584,739	2,768,189	Yes	No
## 50	0	0	Yes	No
##	Total.....data.given.		X	X.1
## 1		No	NA	NA
## 2		No	NA	NA
## 3		No	NA	NA
## 4		Yes	NA	NA
## 5		No	NA	NA

```
## 6          No NA NA
## 7         Yes NA NA
## 8         Yes NA NA
## 9         Yes NA NA
## 10        Yes NA NA
## 11         No NA NA
## 12         No NA NA
## 13         No NA NA
## 14         Yes NA NA
## 15         Yes NA NA
## 16         Yes NA NA
## 17         Yes NA NA
## 18         Yes NA NA
## 19         Yes NA NA
## 20         Yes NA NA
## 21         No NA NA
## 22         No NA NA
## 23         Yes NA NA
## 24         No NA NA
## 25         No NA NA
## 26         No NA NA
## 27         Yes NA NA
## 28         No NA NA
## 29         No NA NA
## 30         No NA NA
## 31         No NA NA
## 32         Yes NA NA
## 33         Yes NA NA
## 34         Yes NA NA
## 35         No NA NA
## 36         No NA NA
## 37         Yes NA NA
## 38         Yes NA NA
## 39         Yes NA NA
## 40         Yes NA NA
## 41         No NA NA
## 42         Yes NA NA
## 43         No NA NA
## 44         No NA NA
## 45         No NA NA
## 46         No NA NA
## 47         No NA NA
## 48         No NA NA
## 49         Yes NA NA
## 50         No NA NA
```

```
summary(dat1)
```

```
##      State      Total.Covid.19.Deaths Total.Covid.19.Cases
## Length:50      Length:50              Length:50
## Class :character Class :character      Class :character
## Mode  :character Mode  :character      Mode  :character
## weekly.average.cases vaccines.distributed vaccinated
## Length:50      Length:50              Length:50
## Class :character Class :character      Class :character
```

```
## Mode :character      Mode :character      Mode :character
## Total.Population     Covid.Cases.65.      Covid.Cases.under.65
## Length:50            Length:50            Length:50
## Class :character     Class :character     Class :character
## Mode :character      Mode :character      Mode :character
## state.provided.age.case.data.  recovered
## Length:50            Length:50
## Class :character     Class :character
## Mode :character      Mode :character
## State.provided.recovered.data. total.number.of.tests  daily.positive.
## Length:50            Length:50            Length:50
## Class :character     Class :character     Class :character
## Mode :character      Mode :character     Mode :character
## State.provided.total.test.data.  PCR.          PCR..1
## Length:50            Length:50            Length:50
## Class :character     Class :character     Class :character
## Mode :character      Mode :character     Mode :character
## antigen.             antigen..1          Total..          Total...1
## Length:50            Length:50            Length:50            Length:50
## Class :character     Class :character     Class :character     Class :character
## Mode :character      Mode :character     Mode :character     Mode :character
## Daily.positive..given. PCR.antigen.data.given. Total.....data.given.
## Length:50            Length:50            Length:50
## Class :character     Class :character     Class :character
## Mode :character      Mode :character     Mode :character
## X                     X.1
## Mode:logical          Mode:logical
## NA's:50               NA's:50
##
```

```
# eliminate NA rows
```

```
dat1<-dat1[,-26]
dat1<-dat1[,-25]
```

```
dat1<-dat1[,-26]
dat1<-dat1[,-25]
dat1$daily.positive.<- as.numeric(sub("%","",dat1$daily.positive.))/100
```

```
#make every other column numerical
```

```
dat1$Total.Covid.19.Deaths <-as.numeric(gsub("","",dat1$Total.Covid.19.Deaths))
dat1$Total.Covid.19.Cases <- as.numeric(gsub("","",dat1$Total.Covid.19.Cases))
dat1$weekly.average.cases <- as.numeric(gsub("","",dat1$weekly.average.cases))
dat1$vaccines.distributed <- as.numeric(gsub("","",dat1$vaccines.distributed))
dat1$vaccinated <- as.numeric(gsub("","",dat1$vaccinated))
dat1$Total.Population <- as.numeric(gsub("","",dat1$Total.Population))
dat1$Covid.Cases.65. <- as.numeric(gsub("","",dat1$Covid.Cases.65.))
dat1$Covid.Cases.under.65 <- as.numeric(gsub("","",dat1$Covid.Cases.under.65))
dat1$recovered <- as.numeric(gsub("","",dat1$recovered))
dat1$total.number.of.tests <- as.numeric(gsub("","",dat1$total.number.of.tests))
dat1$daily.positive. <- as.numeric(gsub("","",dat1$daily.positive.))
dat1$PCR. <- as.numeric(gsub("","",dat1$PCR.))
dat1$PCR..1 <- as.numeric(gsub("","",dat1$PCR..1))
dat1$antigen. <- as.numeric(gsub("","",dat1$antigen.))
dat1$antigen..1 <- as.numeric(gsub("","",dat1$antigen..1))
dat1$Total.. <- as.numeric(gsub("","",dat1$Total..))
```



```

dat1$Total...1 <- as.numeric(gsub(",", "", dat1$Total...1))
#multiple Linear Regression Model for Cases and tests
multi.fit <- lm(dat1$Total.Covid.19.Cases~dat1$vaccinated+dat1$Total.Covid.19.Deaths+dat1$Covid.Cases.65.
summary(multi.fit)

##
## Call:
## lm(formula = dat1$Total.Covid.19.Cases ~ dat1$vaccinated + dat1$Total.Covid.19.Deaths +
##     dat1$Covid.Cases.65. + dat1$Covid.Cases.under.65, data = dat1,
##     na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -394613  -41286   14204   69090  431588
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.020e+03  3.062e+04   0.164   0.8705
## dat1$vaccinated      6.186e-02  2.585e-02   2.393   0.0209 *
## dat1$Total.Covid.19.Deaths  3.052e+01  6.683e+00   4.567  3.83e-05 ***
## dat1$Covid.Cases.65.    -8.454e-01  7.094e-01  -1.192   0.2396
## dat1$Covid.Cases.under.65  2.872e-01  1.348e-01   2.130   0.0387 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 146500 on 45 degrees of freedom
## Multiple R-squared:  0.962, Adjusted R-squared:  0.9586
## F-statistic: 284.5 on 4 and 45 DF, p-value: < 2.2e-16
#96.2% of the Variability can be explained in the data model for COVID 19 cases, the p value is incred
multi.fit2 <- lm(dat1$total.number.of.tests ~ dat1$Total.. + dat1$Total...1+dat1$PCR.+ dat1$PCR..1+ dat1$
summary(multi.fit2)

##
## Call:
## lm(formula = dat1$total.number.of.tests ~ dat1$Total.. + dat1$Total...1 +
##     dat1$PCR. + dat1$PCR..1 + dat1$antigen. + dat1$antigen..1,
##     data = dat1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5634048 -3815604 -2366700    4751 51823948
##
## Coefficients: (1 not defined because of singularities)
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      4.255e+06  1.482e+06   2.870 0.006279 **
## dat1$Total..    -3.650e+00  4.166e+00  -0.876 0.385695
## dat1$Total...1   1.038e+00  2.624e-01   3.954 0.000275 ***
## dat1$PCR.       -1.922e+00  1.580e+01  -0.122 0.903753
## dat1$PCR..1           NA           NA      NA      NA
## dat1$antigen.     9.273e+00  7.835e+01   0.118 0.906324
## dat1$antigen..1  -1.325e+00  4.841e+00  -0.274 0.785572
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
##
## Residual standard error: 9018000 on 44 degrees of freedom
## Multiple R-squared:  0.331, Adjusted R-squared:  0.2549
## F-statistic: 4.353 on 5 and 44 DF,  p-value: 0.002638

#33.11% variability can be explained in COVID 19 test model however the p value is greater than .0001s

#anova test for Case model
anova(multi.fit)

## Analysis of Variance Table
##
## Response: dat1$Total.Covid.19.Cases
##
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
dat1\$vaccinated	1	2.3985e+13	2.3985e+13	1116.8497	< 2.2e-16 ***
dat1\$Total.Covid.19.Deaths	1	3.0540e+11	3.0540e+11	14.2205	0.0004718 ***
dat1\$Covid.Cases.65.	1	4.9719e+10	4.9719e+10	2.3151	0.1351187
dat1\$Covid.Cases.under.65	1	9.7422e+10	9.7422e+10	4.5364	0.0386851 *
Residuals	45	9.6641e+11	2.1476e+10		

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#all predictors show good F values and contribute to the reduction of the total error of the case mode
#anova test for Test model
anova(multi.fit2)

## Analysis of Variance Table
##
## Response: dat1$total.number.of.tests
##
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
dat1\$Total..	1	4.6600e+14	4.6600e+14	5.7302	0.021005 *
dat1\$Total...1	1	1.2951e+15	1.2951e+15	15.9254	0.000246 ***
dat1\$PCR.	1	2.5701e+12	2.5701e+12	0.0316	0.859716
dat1\$Antigen.	1	4.0392e+11	4.0392e+11	0.0050	0.944134
dat1\$Antigen..1	1	6.0935e+12	6.0935e+12	0.0749	0.785572
Residuals	44	3.5782e+15	8.1324e+13		

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#the predictors of both positive and negative PCR and antigen show a very small F value and contribute

# remove PCR and antigen positive and negative in the Test Model
multi.fit2U <- update(multi.fit2, . ~ . - dat1$PCR.)
multi.fit2U <- update(multi.fit2U, . ~ . -dat1$PCR..1)
multi.fit2U <- update(multi.fit2U, . ~ . -dat1$Antigen.)
multi.fit2U <- update(multi.fit2U, . ~ . -dat1$Antigen..1)
multi.fit2U

##
## Call:
## lm(formula = dat1$total.number.of.tests ~ dat1$Total.. + dat1$Total...1,
##     data = dat1)
##
## Coefficients:
## (Intercept)      dat1$Total..      dat1$Total...1
##      4.188e+06      -3.835e+00       1.037e+00
```

```
#test for improvement of the model after the elimination of PCR +/- and antigen +/- coefficients
summary(multi.fit2U)
```

```
##
## Call:
## lm(formula = dat1$total.number.of.tests ~ dat1$Total.. + dat1$Total...1,
##     data = dat1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5466043 -3743740 -2367452   11687 51891141
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.188e+06  1.421e+06   2.946 0.004993 **
## dat1$Total..  -3.835e+00  3.753e+00  -1.022 0.312086
## dat1$Total...1  1.038e+00  2.519e-01   4.119 0.000153 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8736000 on 47 degrees of freedom
## Multiple R-squared:  0.3293, Adjusted R-squared:  0.3007
## F-statistic: 11.54 on 2 and 47 DF,  p-value: 8.39e-05
```

```
# Test Model line of fit improved by 30.07%
```

```
#Data Mining
```

```
# k means clustering
#set random seed
set.seed(123)
#change the Y/N columns to 1/2 aswell as change the numbers in characters "" to factors
#age case data
dat1$state.provided.age.case.data.<-factor(dat1$state.provided.age.case.data.)
dat1$state.provided.age.case.data.<-as.numeric(dat1$state.provided.age.case.data.)
dat1$state.provided.age.case.data.
```

```
## [1] 2 2 2 2 2 1 2 1 2 2 2 1 2 1 2 2 2 1 2 1 1 2 1 2 1 1 1 2 1 2 1 1 2 1 2
## [39] 1 1 2 1 1 2 1 2 1 1 1 1
```

```
#recovery data provided
dat1$State.provided.recovered.data.<-factor(dat1$State.provided.recovered.data.)
dat1$State.provided.recovered.data.<-as.numeric(dat1$State.provided.recovered.data.)
#test data provided by State
dat1$State.provided.total.test.data.<-factor(dat1$State.provided.total.test.data.)
dat1$State.provided.total.test.data.<-as.numeric(dat1$State.provided.total.test.data.)
#daily positive data given
dat1$Daily.positive..given.<-factor(dat1$Daily.positive..given.)
dat1$Daily.positive..given.<-as.numeric(dat1$Daily.positive..given.)
#antigen and pcr given
dat1$PCR.antigen.data.given.<-factor(dat1$PCR.antigen.data.given.)
dat1$PCR.antigen.data.given.<-as.numeric(dat1$PCR.antigen.data.given.)
# total positive and negative given
dat1$Total.....data.given.<-factor(dat1$Total.....data.given.)
dat1$Total.....data.given.<-as.numeric(dat1$Total.....data.given.)
```

```

#convert daily positive percentages to decimals
dat1$daily.positive.<- as.numeric(sub("%","",dat1$daily.positive.))/100

#make every other column numerical
dat1$Total.Covid.19.Deaths <-as.numeric(gsub("","",dat1$Total.Covid.19.Deaths))
dat1$Total.Covid.19.Cases <- as.numeric(gsub("","",dat1$Total.Covid.19.Cases))
dat1$weekly.average.cases <- as.numeric(gsub("","",dat1$weekly.average.cases))
dat1$vaccines.distributed <- as.numeric(gsub("","",dat1$vaccines.distributed))
dat1$vaccinated <- as.numeric(gsub("","",dat1$vaccinated))
dat1$Total.Population <- as.numeric(gsub("","",dat1$Total.Population))
dat1$Covid.Cases.65. <- as.numeric(gsub("","",dat1$Covid.Cases.65.))
dat1$Covid.Cases.under.65 <- as.numeric(gsub("","",dat1$Covid.Cases.under.65))
dat1$recovered <- as.numeric(gsub("","",dat1$recovered))
dat1$total.number.of.tests <- as.numeric(gsub("","",dat1$total.number.of.tests))
dat1$daily.positive. <- as.numeric(gsub("","",dat1$daily.positive.))
dat1$PCR. <- as.numeric(gsub("","",dat1$PCR.))
dat1$PCR..1 <- as.numeric(gsub("","",dat1$PCR..1))
dat1$antigen. <- as.numeric(gsub("","",dat1$antigen.))
dat1$antigen..1 <- as.numeric(gsub("","",dat1$antigen..1))
dat1$Total.. <- as.numeric(gsub("","",dat1$Total..))
dat1$Total...1 <- as.numeric(gsub("","",dat1$Total...1))

#convert dat1 to data frame
covid_df<-as.data.frame(dat1, stringsAsFactors = FALSE)
covid_df<- as.data.frame(sapply(covid_df, function(x) gsub("\","", x)))
#create data function to fix data types and round
to_numeric_and_round_func <- function(x){
  round(as.numeric(as.character(x)),2)
}

#Mutate the columns to proper data type
dat1 <- dat1%>% mutate_at(vars(-one_of("State")), to_numeric_and_round_func)
#get rid of state column
dat1$State <- NULL
#Set Random Seed
set.seed(1234)
#Kmeans function for cases,deaths,weekly cases, vaccines distributed, vaccines taken
covid_means<- kmeans(dat1[,1:5], centers = 5)
covid_means_table <- data.frame(covid_means$size, covid_means$centers)
covid_means_df <- data.frame(Cluster = covid_means$cluster, dat1)
print(covid_means_df)

```

##	Cluster	Total.Covid.19.Deaths	Total.Covid.19.Cases	weekly.average.cases
## 1	1	10686	518000	312
## 2	5	299	64259	222
## 3	2	17062	848000	730
## 4	1	5661	332000	156
## 5	3	60245	3690000	2488
## 6	2	6203	47900	1481
## 7	1	7944	322000	1044
## 8	5	1573	97784	317
## 9	3	33967	2110000	5711
## 10	4	18825	1050000	870
## 11	5	467	30844	99
## 12	5	1995	183000	233

## 13	4	23762	1280000	3012
## 14	2	13131	699000	1017
## 15	1	5849	35600	526
## 16	1	4938	307000	201
## 17	1	6328	436000	518
## 18	1	10216	449000	371
## 19	5	750	53434	1880
## 20	2	8424	424000	1271
## 21	2	17376	655000	315
## 22	4	17471	812000	7226
## 23	2	7010	538000	1931
## 24	1	7086	307000	205
## 25	2	9000	591000	599
## 26	5	1521	106000	145
## 27	5	2328	214000	270
## 28	1	5323	307000	1103
## 29	5	1253	87978	415
## 30	4	24826	947000	4007
## 31	1	3978	193000	186
## 32	3	50465	1940000	7289
## 33	4	12273	935000	1532
## 34	5	1501	105000	151
## 35	4	18827	1040000	1935
## 36	1	6669	443000	309
## 37	1	2451	169000	467
## 38	4	25421	1070000	4328
## 39	5	2638	141000	588
## 40	1	9248	561000	982
## 41	5	1946	119000	174
## 42	2	11900	808000	1051
## 43	3	49273	2820000	3239
## 44	1	2157	389000	455
## 45	5	231	20815	164
## 46	2	10451	633000	1473
## 47	2	5373	37700	1064
## 48	5	2737	145000	370
## 49	2	7376	644000	835
## 50	5	701	56873	58

##	vaccines.distributed	vaccinated	Total.Population	Covid.Cases.65.
----	----------------------	------------	------------------	-----------------

## 1	3248260	2021334	4903185	67923
## 2	703015	448337	731545	8870
## 3	5127405	3975606	7278717	113068
## 4	2121930	1405745	3017825	48900
## 5	28532520	21725654	39512223	379037
## 6	4016000	3184749	5758736	0
## 7	2972115	2340942	3565287	66158
## 8	720505	549853	973764	0
## 9	15556985	11367878	21477737	302496
## 10	7067975	4660059	10617423	165730
## 11	1136660	816481	1415872	5004
## 12	1149805	873222	1787065	0
## 13	9110565	7017850	12671821	253118
## 14	4227610	3229355	6732219	0
## 15	2186505	1807025	3155070	50637

## 16	2124855	1558508	2913314	35185
## 17	3074595	2471143	4467673	94638
## 18	3159400	2239421	4648794	0
## 19	1008240	856186	1344212	11829
## 20	4384740	3414598	6045680	0
## 21	5211830	4345857	6949503	0
## 22	6895860	5223390	9986857	0
## 23	3752100	3260958	5639632	100587
## 24	2031745	1312516	2976149	0
## 25	4222465	3020648	6137428	19278
## 26	805675	609984	1068778	0
## 27	1388380	1107623	1934408	0
## 28	2001620	1576189	3080156	0
## 29	964805	889074	1359711	17222
## 30	6243165	5531413	8882190	0
## 31	1679965	1448347	2096829	33854
## 32	14162005	11519440	19453561	0
## 33	7283460	5402133	10488084	133780
## 34	546570	474619	762062	0
## 35	8210715	6308301	11689100	0
## 36	3099955	2235462	3956971	63028
## 37	2903425	2211938	4217737	0
## 38	9242995	6920672	12801989	261247
## 39	779635	667244	1059361	0
## 40	3452035	2539519	5148714	0
## 41	727915	571138	884659	25895
## 42	4505880	3122632	6833174	0
## 43	18984205	14219706	28995881	0
## 44	1942410	1614798	3205958	31311
## 45	501360	394697	623989	0
## 46	5850555	4849229	8535519	119125
## 47	5380020	4256546	7614893	0
## 48	1330625	995174	1792147	0
## 49	3960825	3461130	5822434	0
## 50	447855	289028	578759	0
##	Covid.Cases.under.65 state.provided.age.case.data. recovered			
## 1	335415		2	325733
## 2	53291		2	0
## 3	733977		2	0
## 4	282804		2	324582
## 5	3203925		2	0
## 6	0		1	0
## 7	254619		2	0
## 8	0		1	0
## 9	1768260		2	0
## 10	689647		2	0
## 11	24212		2	0
## 12	0		1	0
## 13	1019617		2	1251293
## 14	0		1	0
## 15	203428		2	0
## 16	195708		2	0
## 17	337369		2	50372
## 18	0		1	0

## 19	41605	2	0
## 20	0	1	0
## 21	0	1	0
## 22	0	1	0
## 23	439078	2	0
## 24	0	1	296569
## 25	100731	2	0
## 26	0	1	103540
## 27	0	1	0
## 28	0	1	0
## 29	70681	2	83501
## 30	0	1	0
## 31	159373	2	174849
## 32	0	1	0
## 33	788599	2	0
## 34	0	1	101993
## 35	0	1	981253
## 36	376958	2	423402
## 37	0	1	0
## 38	807395	2	962076
## 39	0	1	0
## 40	0	1	0
## 41	93810	2	115346
## 42	0	1	797011
## 43	0	1	0
## 44	358415	2	389760
## 45	0	1	17266
## 46	497199	2	0
## 47	0	1	0
## 48	0	1	0
## 49	0	1	569194
## 50	0	1	47126
##	State.provided.recovered.data. total.number.of.tests daily.positive.		
## 1	2	0	0
## 2	1	1966048	0
## 3	1	8761941	0
## 4	2	3437770	0
## 5	1	56078860	0
## 6	1	2788638	0
## 7	1	8002182	0
## 8	1	673728	0
## 9	1	0	0
## 10	1	7975847	0
## 11	1	0	0
## 12	1	0	0
## 13	2	21102407	0
## 14	1	9187249	0
## 15	1	4580733	0
## 16	1	1335069	0
## 17	2	5173076	0
## 18	1	6671092	0
## 19	1	2164556	0
## 20	1	0	0
## 21	1	0	0

## 22	1	12514712	0
## 23	1	8567524	0
## 24	2	2633209	0
## 25	1	6286100	0
## 26	2	0	0
## 27	1	1017623	0
## 28	1	1427948	0
## 29	2	712674	0
## 30	1	12578312	0
## 31	2	3079551	0
## 32	1	47244227	0
## 33	1	11641898	0
## 34	2	1841920	0
## 35	2	0	0
## 36	2	0	0
## 37	1	4300000	0
## 38	2	12148589	0
## 39	1	3680071	0
## 40	1	6990408	0
## 41	2	452497	0
## 42	2	9051099	0
## 43	1	0	0
## 44	2	2442968	0
## 45	2	365190	0
## 46	1	8700856	0
## 47	1	6115030	0
## 48	1	2541058	0
## 49	2	3353928	0
## 50	2	207049	0
## State.provided.total.test.data.	PCR.	PCR..1	antigen. antigen..1 Total..
## 1	1	0	0 0 0
## 2	2	0	0 0 0
## 3	2	0	0 0 0
## 4	2	260450	260450 84643 454553 345093
## 5	2	0	0 0 0
## 6	2	0	0 0 0
## 7	2	0	0 0 321586
## 8	2	0	0 0 98200
## 9	1	0	0 0 2078784
## 10	2	834801	834801 211288 0 1046089
## 11	1	0	0 0 0
## 12	1	0	0 0 0
## 13	2	0	0 0 0
## 14	2	0	0 0 697533
## 15	2	320212	320212 65452 1530743 385664
## 16	2	0	0 0 304236
## 17	2	0	0 0 432720
## 18	2	0	0 0 382477
## 19	2	42621	42621 14503 274853 57124
## 20	1	0	0 0 540277
## 21	1	0	0 0 0
## 22	2	0	0 0 0
## 23	2	507098	507098 0 0 540277
## 24	2	0	0 0 0

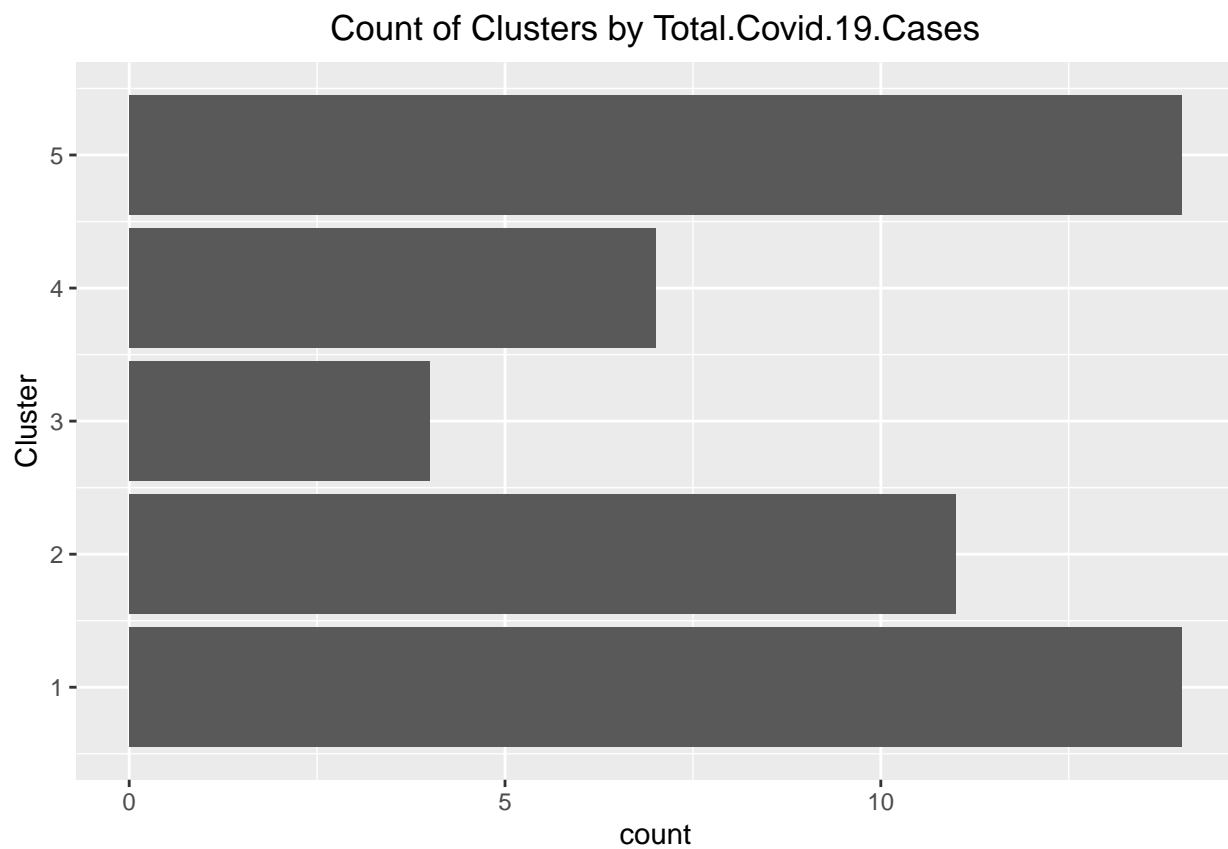


## 25	2	0	0	0	0	0
## 26	1	0	0	0	0	0
## 27	2	0	0	0	0	214010
## 28	2	0	0	0	0	0
## 29	2	0	0	0	0	0
## 30	2	0	0	0	0	0
## 31	2	0	0	0	0	0
## 32	2	0	0	0	0	1934640
## 33	2	800770	800770	128636	0	929406
## 34	2	0	0	0	0	104701
## 35	1	0	0	0	0	0
## 36	1	0	0	0	0	0
## 37	2	0	0	0	0	245000
## 38	2	0	0	0	0	0
## 39	2	0	0	0	0	141097
## 40	2	0	0	0	0	470805
## 41	2	0	0	0	0	0
## 42	2	784564	784564	154237	1550735	938801
## 43	1	0	0	0	0	0
## 44	2	0	0	0	0	0
## 45	2	0	0	0	0	0
## 46	2	0	0	0	0	0
## 47	2	0	0	0	0	0
## 48	2	0	0	0	0	0
## 49	2	0	0	0	0	584739
## 50	2	0	0	0	0	0
##	Total...1 Daily.positive..given. PCR.antigen.data.given.					
## 1	0	2		1		
## 2	0	2		1		
## 3	0	2		1		
## 4	3092677	1		2		
## 5	0	2		1		
## 6	0	2		1		
## 7	7680596	2		1		
## 8	575525	2		1		
## 9	0	2		1		
## 10	6929758	2		2		
## 11	0	2		1		
## 12	0	2		1		
## 13	0	1		1		
## 14	8489716	2		1		
## 15	4177274	1		2		
## 16	1030833	2		1		
## 17	4740356	2		1		
## 18	6288615	1		1		
## 19	2087434	2		2		
## 20	3229264	2		1		
## 21	0	2		1		
## 22	0	1		1		
## 23	8027247	1		2		
## 24	0	1		1		
## 25	0	2		1		
## 26	0	1		1		
## 27	803613	1		1		

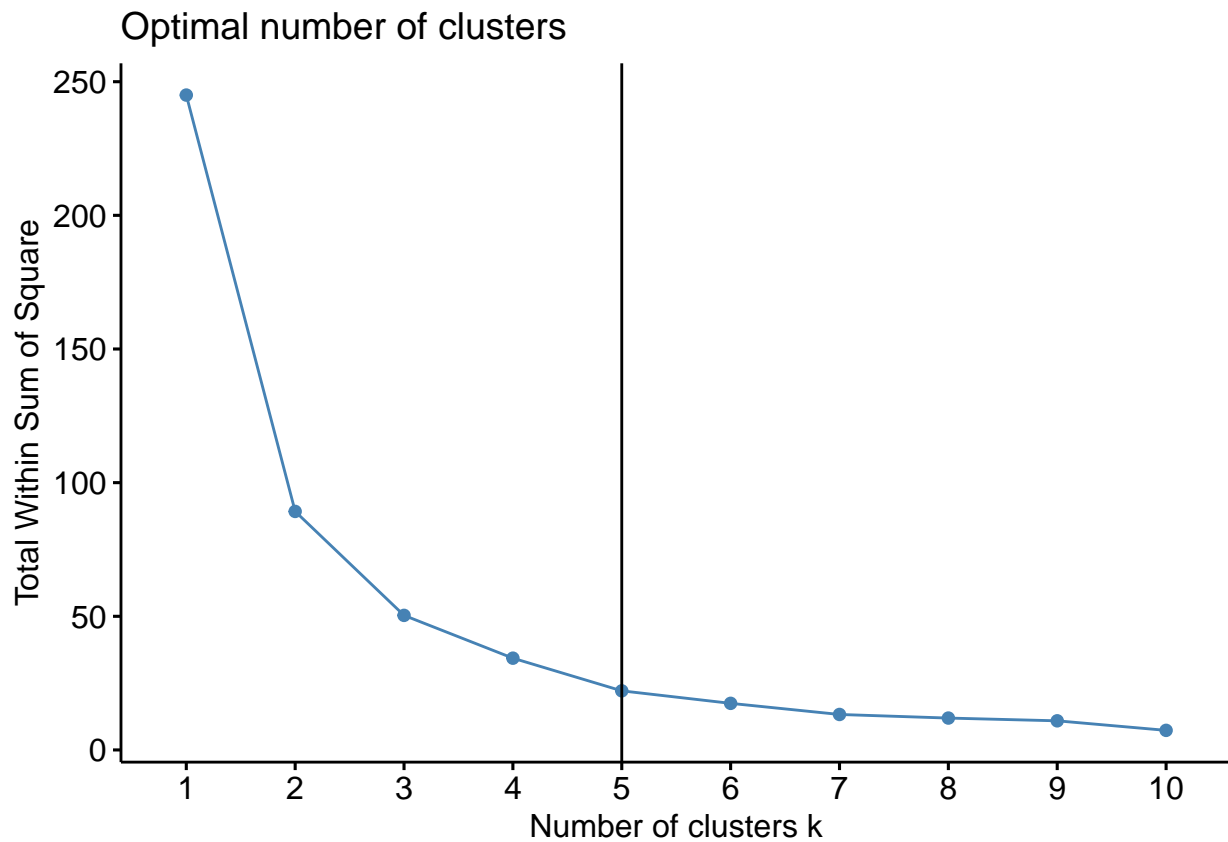
## 28	0	2	1
## 29	0	1	1
## 30	0	1	1
## 31	0	1	1
## 32	45309587	2	1
## 33	10712492	2	2
## 34	1737219	2	1
## 35	0	1	1
## 36	0	1	1
## 37	4000000	2	1
## 38	4220187	1	1
## 39	3538974	2	1
## 40	6519603	2	1
## 41	0	2	1
## 42	8112298	1	2
## 43	0	1	1
## 44	0	1	1
## 45	0	2	1
## 46	0	2	1
## 47	0	2	1
## 48	0	2	1
## 49	2768189	2	1
## 50	0	2	1
##	Total.....data.given.		
## 1		1	
## 2		1	
## 3		1	
## 4		2	
## 5		1	
## 6		1	
## 7		2	
## 8		2	
## 9		2	
## 10		2	
## 11		1	
## 12		1	
## 13		1	
## 14		2	
## 15		2	
## 16		2	
## 17		2	
## 18		2	
## 19		2	
## 20		2	
## 21		1	
## 22		1	
## 23		2	
## 24		1	
## 25		1	
## 26		1	
## 27		2	
## 28		1	
## 29		1	
## 30		1	

```
## 31      1
## 32      2
## 33      2
## 34      2
## 35      1
## 36      1
## 37      2
## 38      2
## 39      2
## 40      2
## 41      1
## 42      2
## 43      1
## 44      1
## 45      1
## 46      1
## 47      1
## 48      1
## 49      2
## 50      1
```

```
##ggplot of covid cases with the cluster analysis
ggplot(data = covid_means_df, aes(y = Cluster)) +
  geom_bar(aes(fill = Total.Covid.19.Cases)) +
  ggtitle("Count of Clusters by Total.Covid.19.Cases") +
  theme(plot.title = element_text(hjust = 0.5))
```

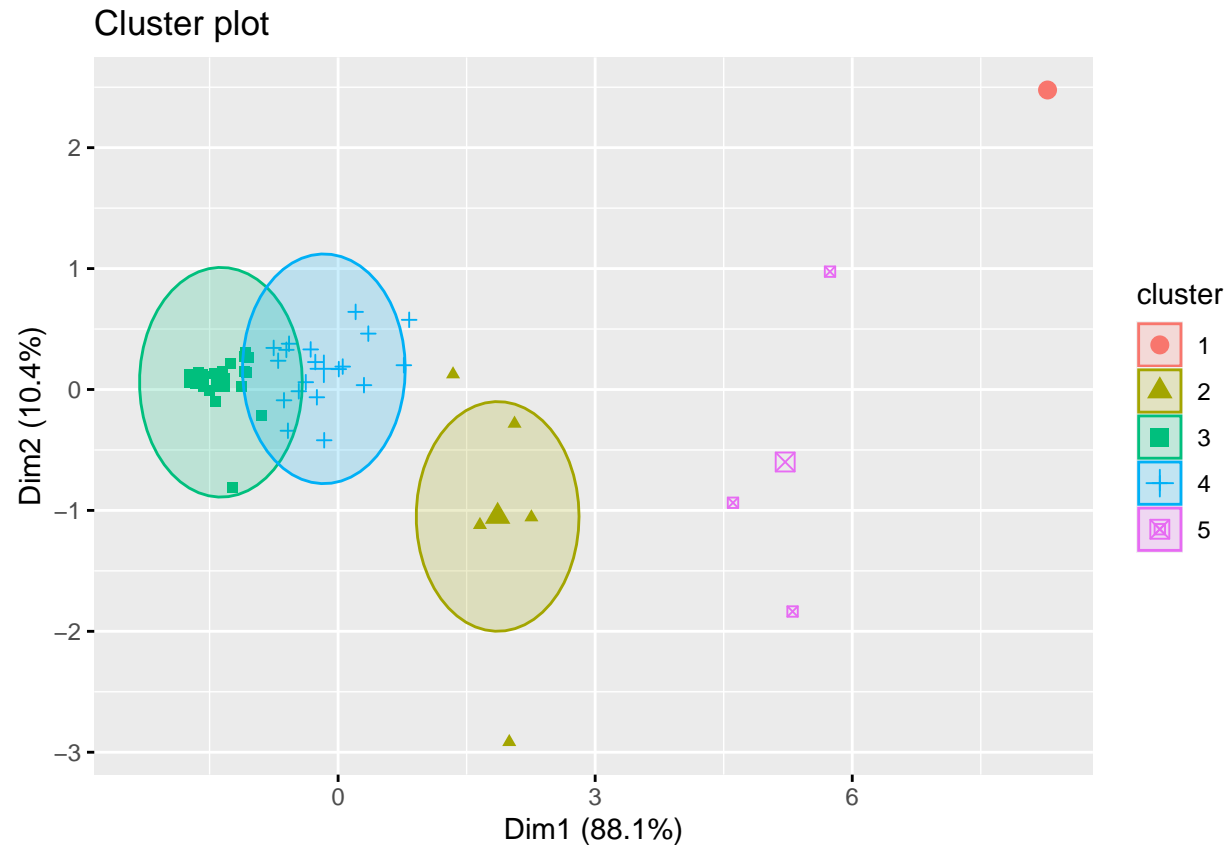


```
#Fancy Kmeans
fviz_nbclust(scale(dat1[,1:5]), kmeans, nstart=100, method = "wss") +
  geom_vline(xintercept = 5, linetype = 1)
```



```
covid_means_fancy <- kmeans(scale(dat1[,1:5]), 5, nstart = 100)
#plot the clusters
fviz_cluster(covid_means_fancy, data = scale(dat1[,1:5]), geom = c("point"), ellipse.type = "euclid")

## Too few points to calculate an ellipse
## Too few points to calculate an ellipse
```



*#All Eclipses are compact for 2,3,4 clusters so they are compact but there is very high variability esp*

*#Principal Component Analysis*

```
dat1<-dat1[,-26]
```

```
dat1<-dat1[,-25]
```

```
dat1$State <- NULL
```

*#make every other column numerical*

```
dat1$Total.Covid.19.Deaths <-as.numeric(gsub(",","",dat1$Total.Covid.19.Deaths))
```

```
dat1$Total.Covid.19.Cases <- as.numeric(gsub(",","",dat1$Total.Covid.19.Cases))
```

```
dat1$weekly.average.cases <- as.numeric(gsub(",","",dat1$weekly.average.cases))
```

```
dat1$vaccines.distributed <- as.numeric(gsub(",","",dat1$vaccines.distributed))
```

```
dat1$vaccinated <- as.numeric(gsub(",","",dat1$vaccinated))
```

```
dat1$Total.Population <- as.numeric(gsub(",","",dat1$Total.Population))
```

```
dat1$Covid.Cases.65. <- as.numeric(gsub(",","",dat1$Covid.Cases.65.))
```

```
dat1$Covid.Cases.under.65 <- as.numeric(gsub(",","",dat1$Covid.Cases.under.65))
```

```
dat1$recovered <- as.numeric(gsub(",","",dat1$recovered))
```

```
dat1$total.number.of.tests <- as.numeric(gsub(",","",dat1$total.number.of.tests))
```

```
dat1$daily.positive. <- as.numeric(gsub(",","",dat1$daily.positive.))
```

```
dat1$PCR. <- as.numeric(gsub(",","",dat1$PCR.))
```

```
dat1$PCR..1 <- as.numeric(gsub(",","",dat1$PCR..1))
```

```
dat1$antigen. <- as.numeric(gsub(",","",dat1$antigen.))
```

```
dat1$antigen..1 <- as.numeric(gsub(",","",dat1$antigen..1))
```

```
dat1$total.. <- as.numeric(gsub(",","",dat1$total..))
```

```
dat1$total...1 <- as.numeric(gsub(",","",dat1$total...1))
```

*#make a matrix version of dat1*

```
dat1_mat <- as.matrix(dat1)
```

```
princomp(dat1, cor = FALSE, scores = TRUE, covmat = NULL, subset = rep(TRUE, nrow(dat1_mat)))
```

```
## Call:
```

```
## princomp(x = dat1, cor = FALSE, scores = TRUE, covmat = NULL,  
## subset = rep(TRUE, nrow(dat1_mat)))
```

```
##
```

```
## Standard deviations:
```

```
##      Comp.1      Comp.2      Comp.3      Comp.4      Comp.5      Comp.6  
## 1.356148e+07 6.899722e+06 4.127907e+06 4.340853e+05 3.667248e+05 2.909974e+05  
##      Comp.7      Comp.8      Comp.9      Comp.10     Comp.11     Comp.12  
## 2.574594e+05 1.766506e+05 1.634020e+05 1.330548e+05 1.000309e+05 2.286686e+04  
##      Comp.13     Comp.14     Comp.15     Comp.16     Comp.17     Comp.18  
## 1.371811e+04 1.651520e+03 8.439615e+02 3.624105e-01 3.266519e-01 3.080034e-01  
##      Comp.19     Comp.20     Comp.21     Comp.22     Comp.23  
## 2.622966e-01 2.520436e-01 2.080417e-01 0.000000e+00 0.000000e+00  
##
```

```
## 23 variables and 50 observations.
```

```
#load and plot principal components
```

```
summary(pc.cr <- princomp(dat1_mat, cor = FALSE))
```

```
## Importance of components:
```

```
##              Comp.1      Comp.2      Comp.3      Comp.4  
## Standard deviation 1.356148e+07 6.899722e+06 4.127907e+06 4.340853e+05  
## Proportion of Variance 7.382544e-01 1.910975e-01 6.839929e-02 7.563833e-04  
## Cumulative Proportion 7.382544e-01 9.293519e-01 9.977512e-01 9.985076e-01  
##              Comp.5      Comp.6      Comp.7      Comp.8  
## Standard deviation 3.667248e+05 2.909974e+05 2.574594e+05 1.766506e+05  
## Proportion of Variance 5.398490e-04 3.399147e-04 2.660781e-04 1.252628e-04  
## Cumulative Proportion 9.990475e-01 9.993874e-01 9.996535e-01 9.997787e-01  
##              Comp.9      Comp.10     Comp.11     Comp.12  
## Standard deviation 1.634020e+05 1.330548e+05 1.000309e+05 2.286686e+04  
## Proportion of Variance 1.071782e-04 7.106451e-05 4.016618e-05 2.098963e-06  
## Cumulative Proportion 9.998859e-01 9.999570e-01 9.999971e-01 9.999992e-01  
##              Comp.13     Comp.14     Comp.15     Comp.16  
## Standard deviation 1.371811e+04 1.651520e+03 8.439615e+02 3.624105e-01  
## Proportion of Variance 7.554059e-07 1.094862e-08 2.859151e-09 5.272219e-16  
## Cumulative Proportion 1.000000e+00 1.000000e+00 1.000000e+00 1.000000e+00  
##              Comp.17     Comp.18     Comp.19     Comp.20  
## Standard deviation 3.266519e-01 3.080034e-01 2.622966e-01 2.520436e-01  
## Proportion of Variance 4.283140e-16 3.808053e-16 2.761703e-16 2.550017e-16  
## Cumulative Proportion 1.000000e+00 1.000000e+00 1.000000e+00 1.000000e+00  
##              Comp.21     Comp.22     Comp.23  
## Standard deviation 2.080417e-01 0 0  
## Proportion of Variance 1.737372e-16 0 0  
## Cumulative Proportion 1.000000e+00 1 1
```

```
loadings(pc.cr)
```

```
##
```

```
## Loadings:
```

```
##              Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6  
## Total.Covid.19.Deaths  
## Total.Covid.19.Cases  
## weekly.average.cases
```

## vaccines.distributed	0.335	0.328	0.222		-0.274	0.177
## vaccinated	0.258	0.237	0.158	0.244	-0.295	0.344
## Total.Population	0.468	0.479	0.359		0.359	-0.294
## Covid.Cases.65.						
## Covid.Cases.under.65				-0.509	-0.480	
## state.provided.age.case.data.						
## recovered					0.245	0.811
## State.provided.recovered.data.						
## total.number.of.tests	0.725	-0.277	-0.629			
## daily.positive.						
## State.provided.total.test.data.						
## PCR.				-0.353	0.212	-0.131
## PCR..1				-0.353	0.212	-0.131
## antigen.						
## antigen..1				-0.387	0.457	0.256
## Total..				-0.497	-0.331	
## Total...1	0.272	-0.725	0.628			
## Daily.positive..given.						
## PCR.antigen.data.given.						
## Total.....data.given.						
##	Comp.7	Comp.8	Comp.9	Comp.10	Comp.11	Comp.12
## Total.Covid.19.Deaths						
## Total.Covid.19.Cases	0.242	0.336		0.802	0.412	
## weekly.average.cases						
## vaccines.distributed			-0.224	-0.329	0.670	
## vaccinated	-0.442	-0.284	0.348	0.355	-0.249	
## Total.Population	0.229	0.149			-0.358	
## Covid.Cases.65.					-0.119	0.972
## Covid.Cases.under.65	0.225	-0.221	-0.465	0.215	-0.329	-0.184
## state.provided.age.case.data.						
## recovered	0.470		0.152	-0.123		
## State.provided.recovered.data.						
## total.number.of.tests						
## daily.positive.						
## State.provided.total.test.data.						
## PCR.		-0.444	0.290		0.150	
## PCR..1		-0.444	0.290		0.150	
## antigen.						
## antigen..1	-0.629	0.187	-0.364	0.105		
## Total..	-0.127	0.541	0.524	-0.211		
## Total...1						
## Daily.positive..given.						
## PCR.antigen.data.given.						
## Total.....data.given.						
##	Comp.13	Comp.14	Comp.15	Comp.16	Comp.17	Comp.18
## Total.Covid.19.Deaths		0.986	0.151			
## Total.Covid.19.Cases						
## weekly.average.cases		0.151	-0.988			
## vaccines.distributed						
## vaccinated						
## Total.Population						
## Covid.Cases.65.						
## Covid.Cases.under.65						
## state.provided.age.case.data.					0.639	0.395

```

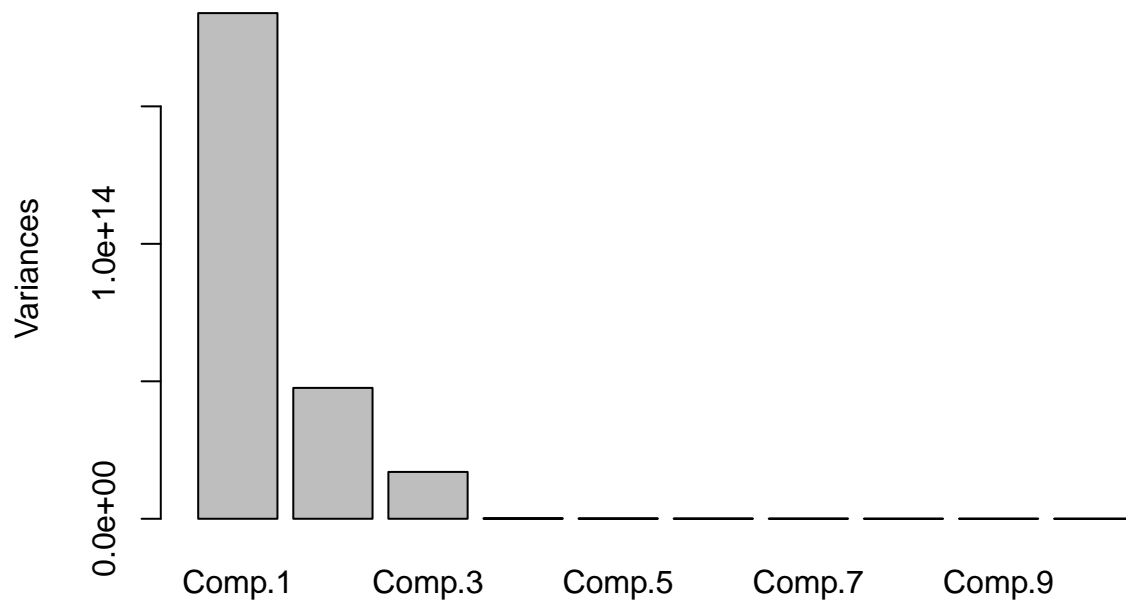
## recovered
## State.provided.recovered.data.          0.708
## total.number.of.tests
## daily.positive.
## State.provided.total.test.data.          0.125  -0.216   0.282
## PCR.
## PCR..1
## antigen.          0.985
## antigen..1
## Total..
## Total...1
## Daily.positive..given.          -0.844          0.473
## PCR.antigen.data.given.          0.156  -0.125   0.115
## Total.....data.given.          0.491  -0.145   0.725
##
## Comp.19 Comp.20 Comp.21 Comp.22 Comp.23
## Total.Covid.19.Deaths
## Total.Covid.19.Cases
## weekly.average.cases
## vaccines.distributed
## vaccinated
## Total.Population
## Covid.Cases.65.
## Covid.Cases.under.65
## state.provided.age.case.data.  -0.343   0.379   0.418
## recovered
## State.provided.recovered.data.   0.428  -0.177  -0.526
## total.number.of.tests
## daily.positive.          0.942   0.335
## State.provided.total.test.data.  0.658   0.652
## PCR.          0.237  -0.666
## PCR..1        -0.237   0.666
## antigen.
## antigen..1
## Total..
## Total...1
## Daily.positive..given.          -0.117  -0.199
## PCR.antigen.data.given.        -0.514   0.418  -0.713
## Total.....data.given.          -0.460
##
##
## Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9
## SS loadings      1.000  1.000  1.000  1.000  1.000  1.000  1.000  1.000  1.000
## Proportion Var   0.043  0.043  0.043  0.043  0.043  0.043  0.043  0.043  0.043
## Cumulative Var   0.043  0.087  0.130  0.174  0.217  0.261  0.304  0.348  0.391
##
## Comp.10 Comp.11 Comp.12 Comp.13 Comp.14 Comp.15 Comp.16 Comp.17
## SS loadings      1.000  1.000  1.000  1.000  1.000  1.000  1.000  1.000
## Proportion Var   0.043  0.043  0.043  0.043  0.043  0.043  0.043  0.043
## Cumulative Var   0.435  0.478  0.522  0.565  0.609  0.652  0.696  0.739
##
## Comp.18 Comp.19 Comp.20 Comp.21 Comp.22 Comp.23
## SS loadings      1.000  1.000  1.000  1.000  1.000  1.000
## Proportion Var   0.043  0.043  0.043  0.043  0.043  0.043
## Cumulative Var   0.783  0.826  0.870  0.913  0.957  1.000

```

```
plot(pc.cr)
```



## pc.cr



```
biplot(pc.cr)
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

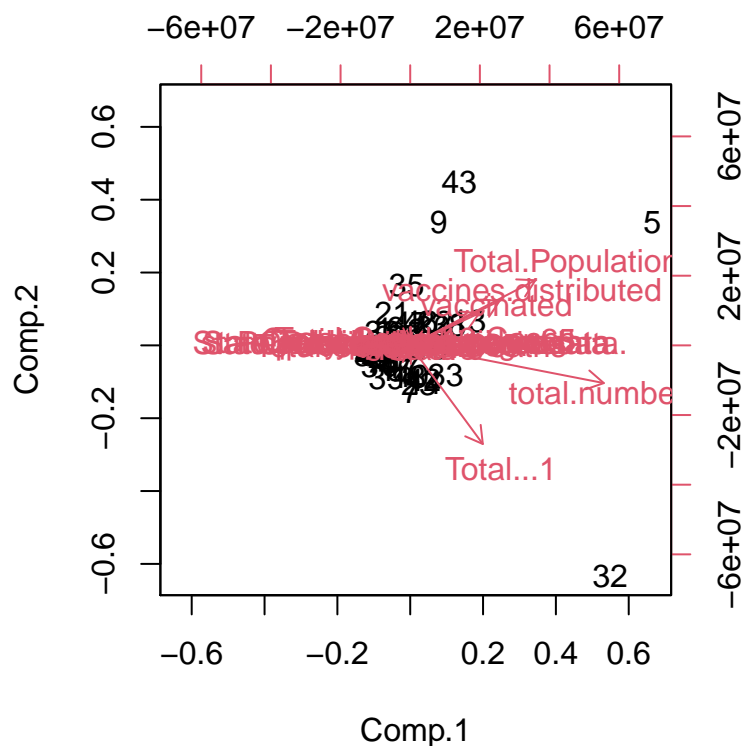
```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```



```
#second plot
pc.cr$scores
```

##	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5
## [1,]	-6891800.56	2224530.27	931235.7	-83111.853	123558.246
## [2,]	-8708141.68	-1558531.82	-2622386.3	61786.184	-119466.917
## [3,]	1728891.58	2038475.00	-2999140.5	-62547.256	-169541.178
## [4,]	-4984433.68	-2413552.16	-288801.5	-541461.572	220266.161
## [5,]	63728297.12	16451342.64	-13157065.3	-235863.422	-231591.834
## [6,]	-3943477.19	2354540.07	-166190.8	339634.719	16385.353
## [7,]	349630.93	-6241728.09	251021.5	89771.995	-405836.950
## [8,]	-9341284.14	-1474202.56	-1331994.4	71284.250	-108569.629
## [9,]	7544447.40	16491868.67	11237945.8	-1139786.314	-1480788.401
## [10,]	5456882.05	-394111.33	3677217.3	-1137466.275	256060.876
## [11,]	-9574231.94	-458327.72	-986237.6	122189.218	-134961.173
## [12,]	-9375093.75	-256966.19	-834718.0	127270.078	-5414.051
## [13,]	15345602.37	3262149.09	-7456712.9	-159325.284	245611.035
## [14,]	3576542.14	-5018144.73	1616019.6	47263.854	160524.488
## [15,]	-3687694.58	-3360213.36	-200586.3	-825219.628	604199.981
## [16,]	-7081727.75	-347306.76	-279822.1	-66116.009	-241260.241
## [17,]	-1999568.91	-2821699.88	558293.6	-67713.697	-252504.762
## [18,]	-449402.86	-4311077.48	644634.7	95537.738	53311.844
## [19,]	-7502712.22	-2641513.96	-1078645.9	-2996.083	13354.195
## [20,]	-4745453.26	1103427.11	3886350.4	136694.361	-312956.829
## [21,]	-4678786.69	4392080.24	2487100.8	464240.566	-274990.453
## [22,]	6609849.79	3143587.09	-3774803.1	389928.273	399958.399
## [23,]	2342958.01	-5175922.75	1201535.7	-383187.662	-77242.230
## [24,]	-6495330.30	-17384.22	-1795141.4	80673.682	189675.207
## [25,]	-1177859.88	1621939.09	-2192835.1	169268.532	194330.410
## [26,]	-9898089.87	-779257.17	-1213726.5	106822.458	-68396.772
## [27,]	-8205694.54	-920946.62	-811919.8	62931.023	-139842.119

## [28,]	-7262337.75	417884.55	-963706.1	177978.623	56086.621
## [29,]	-9118962.92	-717600.50	-1480932.9	115948.204	-113399.707
## [30,]	6005612.64	2461117.89	-4301991.7	541026.019	96834.598
## [31,]	-6667494.40	-645725.32	-2456816.6	100574.834	-171318.329
## [32,]	52654757.40	-30934898.10	9024597.0	720815.942	-229700.825
## [33,]	9339856.14	-3969207.48	3851814.2	-867958.833	-42637.665
## [34,]	-8355473.93	-2816507.93	-1462950.4	76327.188	-74114.172
## [35,]	-929152.07	8133091.59	5174740.3	443024.160	278112.252
## [36,]	-7331563.20	1771682.46	588472.3	-16828.002	-236793.849
## [37,]	-3098464.85	-2299700.80	462737.2	175873.481	-22338.707
## [38,]	10068886.38	2749726.31	886148.2	67890.567	56787.758
## [39,]	-6265016.30	-4368366.82	-1293939.6	110812.866	-96084.961
## [40,]	260614.38	-4157601.18	890653.0	82280.733	43797.668
## [41,]	-9689364.85	-1023734.14	-1590391.2	54972.924	-134590.664
## [42,]	3498945.87	-4605824.42	1580019.0	-1379976.147	1319913.540
## [43,]	12906871.14	21906785.09	15098601.5	976114.591	1020398.443
## [44,]	-6464054.22	204212.71	-1573948.5	-43878.811	53853.356
## [45,]	-10003255.45	-1248593.39	-1707495.0	104095.817	-96136.916
## [46,]	2725216.01	3084300.23	-2211462.4	189267.616	-68562.119
## [47,]	69363.59	3022204.03	-1119944.8	422787.458	72385.854
## [48,]	-7441103.45	-872786.75	-2372803.5	133728.761	-29420.265
## [49,]	-2663623.74	283756.84	1329625.4	75111.015	-74990.318
## [50,]	-10182574.02	-1267267.34	-1651652.9	79509.119	-61954.250
##	Comp. 6	Comp. 7	Comp. 8	Comp. 9	Comp. 10
## [1,]	-21323.699	4.434084e+05	74482.422	-318535.3523	-23769.554
## [2,]	-97443.303	-3.823449e+03	23512.804	-49934.9125	-25608.052
## [3,]	-142861.418	8.263102e+04	4961.315	-153115.9695	253349.433
## [4,]	121857.164	3.923395e+04	11488.890	-116870.9340	19088.668
## [5,]	-254089.849	-1.481785e+05	-44557.449	-198636.5394	42073.771
## [6,]	-66249.203	-2.506114e+05	-191029.188	34872.7279	-335735.801
## [7,]	94443.844	-7.106556e+04	-138376.995	-177817.3991	18550.323
## [8,]	-104085.731	-6.244933e+03	49150.560	-8904.0622	16589.046
## [9,]	178930.649	-2.385696e+05	404121.477	68604.9010	-83502.947
## [10,]	-711991.935	5.138342e+05	-204897.035	219421.7071	-141113.042
## [11,]	-56652.324	-5.658590e+04	-66941.358	-76142.9241	-66077.411
## [12,]	-146168.939	3.372562e+04	28052.217	-51500.3665	55002.578
## [13,]	801835.092	5.849277e+05	144247.181	194175.7221	-52957.350
## [14,]	-353689.897	1.970498e+05	393049.681	69570.9593	-14000.574
## [15,]	217393.109	-1.006532e+06	-15704.359	-393340.5819	31473.380
## [16,]	-89859.776	3.680392e+01	98303.671	-19911.4362	16308.965
## [17,]	-52003.171	6.338450e+04	60305.461	-62915.1654	79949.269
## [18,]	-232460.034	1.413290e+05	189067.144	-86410.2374	-79932.501
## [19,]	-3657.561	-2.035380e+05	-49149.328	-143917.9893	34444.790
## [20,]	64075.183	-2.365294e+05	-14705.953	89528.9613	-116424.176
## [21,]	246328.650	-4.208535e+05	-295685.834	100515.6474	169597.979
## [22,]	-299175.835	-1.075918e+05	200597.595	228604.3503	-205758.024
## [23,]	-205370.591	5.487773e+04	-440574.313	225828.8972	232864.166
## [24,]	-1808.018	2.464563e+05	148085.234	-22285.8074	-87788.182
## [25,]	-267891.440	6.892159e+04	155295.172	-16319.2978	-62332.092
## [26,]	-1818.905	3.382445e+04	-4155.846	-30530.5089	22573.260
## [27,]	-86675.634	-4.908135e+04	93241.153	66514.4452	32341.075
## [28,]	-163347.390	1.037125e+04	47026.744	-609.3588	64828.460
## [29,]	9309.115	-2.475515e+04	-53973.918	-223.5256	54099.636
## [30,]	13968.604	-4.295973e+05	37896.039	528405.2054	259981.758

## [31,]	140920.506	-4.102632e+04	-65760.628	42407.8481	55436.622
## [32,]	173828.938	-8.318326e+04	94490.089	-54394.5540	-2199.371
## [33,]	-384653.160	2.360307e+05	-571308.299	182818.7104	10894.564
## [34,]	-24199.426	6.349305e+04	49942.392	-4346.9215	45870.809
## [35,]	851246.133	1.928728e+05	-180868.487	123673.2831	-79846.396
## [36,]	385443.314	1.762286e+05	-148514.954	-187440.4022	67086.839
## [37,]	-134608.330	-3.516295e+04	-2014.342	-45527.8003	-187517.367
## [38,]	749936.389	4.549459e+05	-255224.403	-266554.6447	-138593.286
## [39,]	-101757.100	3.193961e+04	38498.789	-47080.9382	62339.260
## [40,]	-232807.159	1.292184e+05	245070.469	-16482.2411	-15130.852
## [41,]	29597.990	4.668784e+04	-20392.600	-52626.6764	63324.828
## [42,]	448779.449	-3.260968e+05	208796.843	134837.8078	17365.009
## [43,]	-418651.479	5.746009e+04	12941.945	-165709.5112	195267.821
## [44,]	109771.713	3.158827e+05	46346.648	-59181.2080	175476.381
## [45,]	-75600.420	-1.624773e+04	-10655.943	-31782.8725	-2399.038
## [46,]	-88045.061	-1.589991e+05	-125285.091	42727.4370	63914.669
## [47,]	-68443.836	-3.591292e+05	-218253.640	117200.6676	-513568.009
## [48,]	-124273.168	-2.642475e+04	40735.795	7584.3246	-25189.750
## [49,]	455029.890	3.603907e+04	193528.925	433900.9729	92926.666
## [50,]	-81031.944	4.501716e+04	24793.309	-52144.4372	6423.753
##	Comp. 11	Comp. 12	Comp. 13	Comp. 14	Comp. 15
## [1,]	-5977.9418	-11956.2851	-9012.774679	4275.826204	197.13002
## [2,]	120073.0920	4403.7874	-7452.841709	-1375.261805	5.55048
## [3,]	-42892.3373	6928.5561	7726.219598	3865.799611	124.35557
## [4,]	34944.4239	-12125.4270	30374.985618	394.591113	-606.87014
## [5,]	65066.6323	-44702.1241	1328.985265	-1134.277281	542.56220
## [6,]	-128456.4348	-16287.8172	4553.318685	2988.113485	84.96001
## [7,]	114115.3673	21376.7772	-1249.354000	-2172.487145	98.82128
## [8,]	37938.0560	-1530.4890	-1684.519820	-99.764158	40.50537
## [9,]	-15254.7252	11562.9539	-10546.122185	1130.687703	-816.78700
## [10,]	108276.0093	37908.9850	42211.655151	1065.862641	581.38288
## [11,]	67725.2342	-3079.6896	-2987.864790	-866.281667	70.98565
## [12,]	655.8952	-2806.5989	671.711173	-695.272940	-209.73740
## [13,]	-97166.1765	45557.1640	-6092.061262	-432.541885	781.65656
## [14,]	-190058.1887	-9268.3198	-995.166553	1285.908235	1045.27327
## [15,]	-160219.4491	23492.0722	2954.127166	1391.029415	238.59481
## [16,]	30547.8135	1816.8303	-3069.318554	-114.543337	595.63503
## [17,]	-154305.2317	24236.3839	6640.795514	-2671.515839	521.37676
## [18,]	23430.9336	-3603.9802	-9619.901042	1304.070650	1167.37047
## [19,]	-10478.5808	3264.0390	7825.550890	-1547.349636	-2011.91230
## [20,]	40385.6024	-9427.0697	7.197446	-1066.317297	412.09951
## [21,]	204011.4519	12048.4736	8937.919778	1560.633928	1559.46986
## [22,]	106779.4536	25904.6090	-17615.481021	624.636001	-3920.44983
## [23,]	-180309.1360	-3740.5966	-62772.839637	945.531031	196.71071
## [24,]	83068.2289	-11404.8654	-11940.427163	2572.840689	840.24585
## [25,]	107772.0472	16842.5220	-14632.400996	-2353.756835	661.43349
## [26,]	51398.7776	-7005.8327	-719.769576	-137.832747	-17.88287
## [27,]	39188.2014	640.9651	-1370.572711	-1567.080066	314.35341
## [28,]	-13719.1181	467.5003	1457.848463	233.609585	-592.56946
## [29,]	-45616.2799	-4253.8312	7110.525846	-237.680254	-318.88759
## [30,]	44001.5437	36098.2673	12708.541952	4098.361381	-189.39026
## [31,]	36962.1163	1833.7365	4068.295398	-259.240905	387.37691
## [32,]	44125.0735	-3399.9883	6968.804605	917.775803	-386.21163
## [33,]	10251.6732	-23673.2378	-3720.474492	-1121.387185	-814.43463

## [34,]	2160.6521	-8679.5054	1628.735269	-27.569545	53.88001
## [35,]	94006.8784	-51178.8560	3551.343696	316.954433	-315.46804
## [36,]	127063.0446	-21035.3966	220.152779	-73.512473	-452.61282
## [37,]	-77388.4690	-14164.1585	-346.790469	-1905.351740	207.30674
## [38,]	-61740.6660	71345.5118	-6015.688921	357.906126	-575.06779
## [39,]	15014.3525	-2778.4257	1338.407480	-372.566962	-291.20036
## [40,]	2951.3033	-1483.8090	-7345.056679	-1262.274888	395.35130
## [41,]	45752.7044	3440.7364	237.107481	-232.600270	16.14007
## [42,]	140609.7552	-19884.0354	-13269.320142	-1214.682944	298.95360
## [43,]	-28577.1297	11072.3224	-1877.213178	-1515.337231	183.90652
## [44,]	-231463.8864	-57228.0769	18616.322359	1098.837141	-1727.53323
## [45,]	33837.7109	-3680.3491	-494.390582	-224.253814	-50.39754
## [46,]	-237360.9259	35827.2148	15595.826161	-3131.117079	-120.13815
## [47,]	-145921.0012	-15507.2359	1820.831103	687.621325	947.14514
## [48,]	77267.0034	4328.5834	-5252.210043	-522.009803	240.86026
## [49,]	-134807.4845	-32469.0622	14409.919519	-2773.401864	563.96739
## [50,]	52332.1309	-4042.9282	-2882.568191	-9.326907	42.18990
##	Comp.16	Comp.17	Comp.18	Comp.19	Comp.20
## [1,]	-0.68734498	0.405647168	0.213282052	-0.321868797	-0.258736457
## [2,]	-0.34142564	0.147409369	0.070581996	-0.168686774	0.537165391
## [3,]	-0.49493342	-0.348736979	0.241676462	0.010664993	0.159325890
## [4,]	0.74643624	0.140445130	0.197722525	-0.165338431	0.224252547
## [5,]	0.14377979	-0.062215128	0.099750783	0.075906122	-0.185041090
## [6,]	-0.23550551	-0.132211244	0.114699149	0.163990889	0.106972254
## [7,]	0.13153213	-0.152844988	0.337322434	0.075239188	0.105290298
## [8,]	0.10040307	-0.635616463	0.451012514	0.148737097	-0.286918393
## [9,]	-0.09569505	0.061528388	-0.204775181	0.075187023	0.159115885
## [10,]	0.05255768	0.036116605	-0.068682941	-0.124699586	-0.017597574
## [11,]	-0.43307117	0.342415115	-0.166602432	-0.749747068	-0.067646345
## [12,]	-0.54945342	-0.353914933	-0.566297042	-0.479811837	-0.536746609
## [13,]	-0.23083481	-0.062784074	-0.227085015	-0.407119564	0.124220715
## [14,]	-0.06678152	-0.166676154	0.181239908	-0.242152077	-0.146305368
## [15,]	0.18760608	-0.010967954	-0.106691061	-0.118227589	0.004597364
## [16,]	0.11722082	0.001734503	0.687903916	-0.141188682	0.197015442
## [17,]	0.04755276	0.570974326	-0.007187524	0.262624413	-0.190949005
## [18,]	0.95755599	-0.270328799	-0.054866348	-0.033621213	0.027120617
## [19,]	-0.02564672	-0.330675438	0.742003012	-0.476089155	0.185016280
## [20,]	0.08644297	-0.116170011	0.125104369	-0.347001603	-0.526091449
## [21,]	-0.31120041	-0.253631707	0.058545338	-0.090055582	-0.182030900
## [22,]	0.13670326	0.194711408	-0.044017507	0.027119148	-0.168689904
## [23,]	0.33358504	0.084840451	0.061066563	-0.183716108	0.083657951
## [24,]	0.41633312	0.199442188	-0.253827592	0.308057471	0.257154422
## [25,]	-0.33845974	0.456788034	0.117044009	-0.274843393	0.576856115
## [26,]	0.36964724	0.356792483	-0.950587641	-0.125282697	-0.518070637
## [27,]	0.96267351	-0.431563733	-0.148259368	0.061109697	-0.073903181
## [28,]	-0.48707132	-0.513435422	-0.137851165	0.164577272	0.042132763
## [29,]	0.44243607	0.736183368	-0.412034535	0.196123773	0.361279871
## [30,]	0.26808277	0.084548253	0.139361494	-0.048189451	-0.063475746
## [31,]	0.48090529	0.715686991	-0.330948117	0.168309926	0.443009300
## [32,]	-0.24133563	0.254154098	-0.287224583	-0.024873811	0.163166221
## [33,]	-0.19111706	-0.142767544	0.054242862	0.128233169	-0.018278162
## [34,]	0.09693880	-0.058575872	0.405890945	0.489032487	-0.471304370
## [35,]	-0.01550386	-0.241132381	0.169604550	-0.163604521	0.074456627
## [36,]	0.23182465	0.287286889	-0.406831118	-0.224023483	-0.010179647

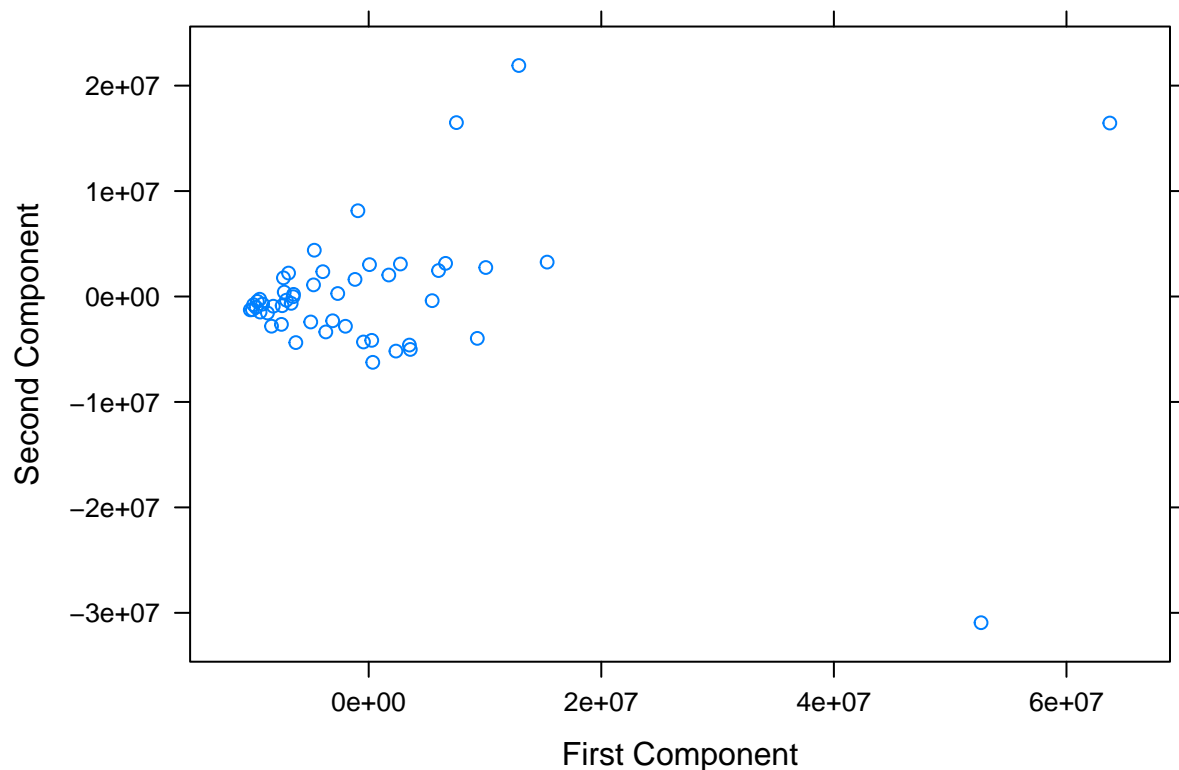
```

## [37,] 0.18677535 -0.373614575 0.175717408 0.076199668 -0.199654133
## [38,] 0.34667007 -0.412002624 0.311983807 0.423954116 -0.308525855
## [39,] 0.03740779 -0.695694511 0.330320635 0.127923920 -0.356321624
## [40,] 0.02845181 -0.347391664 0.197332639 -0.005116206 -0.117846996
## [41,] -0.39391856 0.580239795 0.085450809 0.283209294 0.322402781
## [42,] -0.39943797 0.028268208 -0.084153983 0.249549837 -0.103063599
## [43,] 0.17353946 0.138537820 -0.044979187 0.111056262 0.138624867
## [44,] 0.10504404 0.083695246 -0.236485556 0.004833870 0.239761993
## [45,] -0.29701696 0.158549319 -0.290248819 0.589607099 -0.044402529
## [46,] -0.45869079 0.118203819 -0.466054370 -0.114421367 0.054363921
## [47,] -0.05874873 0.233874382 -0.050736797 0.022454271 0.189513773
## [48,] -0.34126583 -0.380901118 -0.207158752 0.111827716 0.151952398
## [49,] -0.17205640 -0.041527805 0.434437817 0.101045158 -0.062346223
## [50,] -0.32159029 0.117307763 -0.249711364 0.573105115 -0.014299891
##      Comp.21      Comp.22      Comp.23
## [1,] -0.077475495 -8.954915e-04 -1.094361e-06
## [2,] 0.376531992 -4.914010e-04 8.951764e-07
## [3,] 0.230765945 -7.767000e-04 7.965144e-07
## [4,] -0.233791542 1.253506e-03 -4.941513e-08
## [5,] -0.149528103 2.071380e-04 -3.130692e-07
## [6,] 0.013085860 -3.969507e-04 4.371694e-07
## [7,] 0.240817869 1.032476e-04 5.656942e-07
## [8,] -0.019244952 7.140074e-05 2.657018e-07
## [9,] -0.016765376 -1.456528e-04 2.570224e-07
## [10,] 0.034923131 1.111176e-04 -1.463867e-07
## [11,] 0.384721560 -4.804095e-04 -6.981638e-07
## [12,] -0.018706017 -6.733836e-04 -8.980052e-07
## [13,] 0.104546062 -1.991838e-04 1.719103e-08
## [14,] 0.032149957 -3.006721e-05 -2.460609e-07
## [15,] 0.117624882 2.838167e-04 1.134519e-08
## [16,] 0.349656566 1.396876e-04 4.923589e-07
## [17,] -0.274937064 2.403832e-05 -9.065878e-07
## [18,] 0.177165016 1.366449e-03 3.763683e-07
## [19,] -0.282024055 2.594536e-04 1.550757e-07
## [20,] -0.016475890 2.037630e-04 -1.010266e-06
## [21,] -0.098310330 -4.082405e-04 -2.278822e-07
## [22,] 0.026110674 1.586909e-04 -3.072585e-07
## [23,] -0.095197441 5.970241e-04 -1.567707e-07
## [24,] -0.200088904 5.817087e-04 2.547729e-07
## [25,] 0.289133560 -4.189730e-04 4.952315e-07
## [26,] -0.290420910 6.131373e-04 -1.541443e-06
## [27,] 0.178133688 1.327784e-03 4.581221e-07
## [28,] -0.016850842 -7.581783e-04 6.912822e-07
## [29,] 0.119008350 5.674977e-04 8.715664e-08
## [30,] 0.056412503 3.844777e-04 -1.622632e-07
## [31,] 0.109493003 6.474211e-04 1.940041e-07
## [32,] 0.006731981 -3.302603e-04 5.315174e-09
## [33,] 0.065535071 -3.493816e-04 2.790615e-07
## [34,] -0.473020322 4.760570e-05 -6.998629e-07
## [35,] 0.140797697 8.258382e-06 3.622460e-07
## [36,] 0.202600888 3.563359e-04 -3.092957e-07
## [37,] 0.010602635 2.227681e-04 1.022175e-07
## [38,] -0.153220650 3.626390e-04 1.201784e-07
## [39,] -0.008840172 -2.345428e-05 2.100555e-07

```

```
## [40,] -0.019227212  4.074283e-05  1.250246e-07
## [41,] -0.107314054 -6.217496e-04  8.293211e-08
## [42,]  0.002406820 -6.935724e-04 -2.422289e-08
## [43,] -0.039722276  2.383744e-04  1.289425e-07
## [44,]  0.401983795  5.350972e-05  6.593452e-07
## [45,] -0.513547746 -5.074738e-04 -2.027438e-07
## [46,]  0.142115047 -6.673856e-04 -1.737424e-08
## [47,] -0.003680710 -7.439477e-05  6.162083e-08
## [48,] -0.031278994 -5.089659e-04  6.606504e-07
## [49,] -0.171282162 -2.437887e-04 -1.189215e-07
## [50,] -0.502103335 -5.365348e-04 -1.175569e-07

pca.plot <- xyplot(pc.cr$scores[,2] ~ pc.cr$scores[,1])
pca.plot$xlabel <- "First Component"
pca.plot$ylabel <- "Second Component"
pca.plot
```



```
#Fuzzy C-means
x<-rbind(dat1$Total.Covid.19.Deaths,dat1$Total.Covid.19.Cases,dat1$weekly.average.cases)
x<-t(x)
result<-cmeans(x,5,50, verbose=TRUE, method = "cmeans")
```

```
## Iteration: 1, Error: 74381240707.8613281250
## Iteration: 2, Error: 68798435191.6853179932
## Iteration: 3, Error: 58164024770.0741119385
## Iteration: 4, Error: 38410507484.4967803955
## Iteration: 5, Error: 32870165447.9824638367
## Iteration: 6, Error: 30406240885.5644989014
## Iteration: 7, Error: 28320103654.3848609924
## Iteration: 8, Error: 26541673541.6770172119
```

```

## Iteration: 9, Error: 25371867224.2859306335
## Iteration: 10, Error: 24772623967.2916107178
## Iteration: 11, Error: 24489568248.3919181824
## Iteration: 12, Error: 24344638934.0754051208
## Iteration: 13, Error: 24255010316.8575286865
## Iteration: 14, Error: 24185577059.0031585693
## Iteration: 15, Error: 24120179321.1550102234
## Iteration: 16, Error: 24048556486.1487770081
## Iteration: 17, Error: 23959484512.1469650269
## Iteration: 18, Error: 23833881975.9684410095
## Iteration: 19, Error: 23631747757.4226570129
## Iteration: 20, Error: 23265329443.0152397156
## Iteration: 21, Error: 22566300793.1689453125
## Iteration: 22, Error: 21243770047.9325675964
## Iteration: 23, Error: 18483468688.1363868713
## Iteration: 24, Error: 15010111363.9576759338
## Iteration: 25, Error: 14092118562.6220760345
## Iteration: 26, Error: 13770166225.0370063782
## Iteration: 27, Error: 13614324080.7022991180
## Iteration: 28, Error: 13544794623.9480819702
## Iteration: 29, Error: 13515452921.6817340851
## Iteration: 30, Error: 13503184927.3750991821
## Iteration: 31, Error: 13497956377.8798103333
## Iteration: 32, Error: 13495658578.8578987122
## Iteration: 33, Error: 13494616399.9942951202
## Iteration: 34, Error: 13494130730.3455810547
## Iteration: 35, Error: 13493899606.0120468140
## Iteration: 36, Error: 13493787936.3614540100
## Iteration: 37, Error: 13493733413.6714687347
## Iteration: 38, Error: 13493706605.0307426453
## Iteration: 39, Error: 13493693362.1322402954
## Iteration: 40, Error: 13493686800.7165279388
## Iteration: 41, Error: 13493683543.4492912292
## Iteration: 42, Error: 13493681924.4450359344
## Iteration: 43, Error: 13493681119.0921688080
## Iteration: 44, Error: 13493680718.2781238556
## Iteration: 45 converged, Error: 13493680518.7342834473

```

```
print(result)
```

```
## Fuzzy c-means clustering with 5 clusters
```

```
##
```

```
## Cluster centers:
```

```
##      [,1]      [,2]      [,3]
```

```
## 1 58534.743 3554433.0 2604.8412
```

```
## 2  8541.306  501745.3  790.5348
```

```
## 3 19021.061  973974.8 2593.4547
```

```
## 4 42355.239 2096999.9 6146.5719
```

```
## 5  2496.471 110822.1  471.6960
```

```
##
```

```
## Memberships:
```

```
##      1          2          3          4          5
```

```
## [1,] 2.908455e-05 9.969566e-01 1.289615e-03 1.075358e-04 1.617162e-03
```

```
## [2,] 1.757594e-04 1.118536e-02 2.586637e-03 5.180602e-04 9.855342e-01
```

```
## [3,] 1.846048e-03 1.127419e-01 8.518673e-01 8.666197e-03 2.487857e-02
```



```

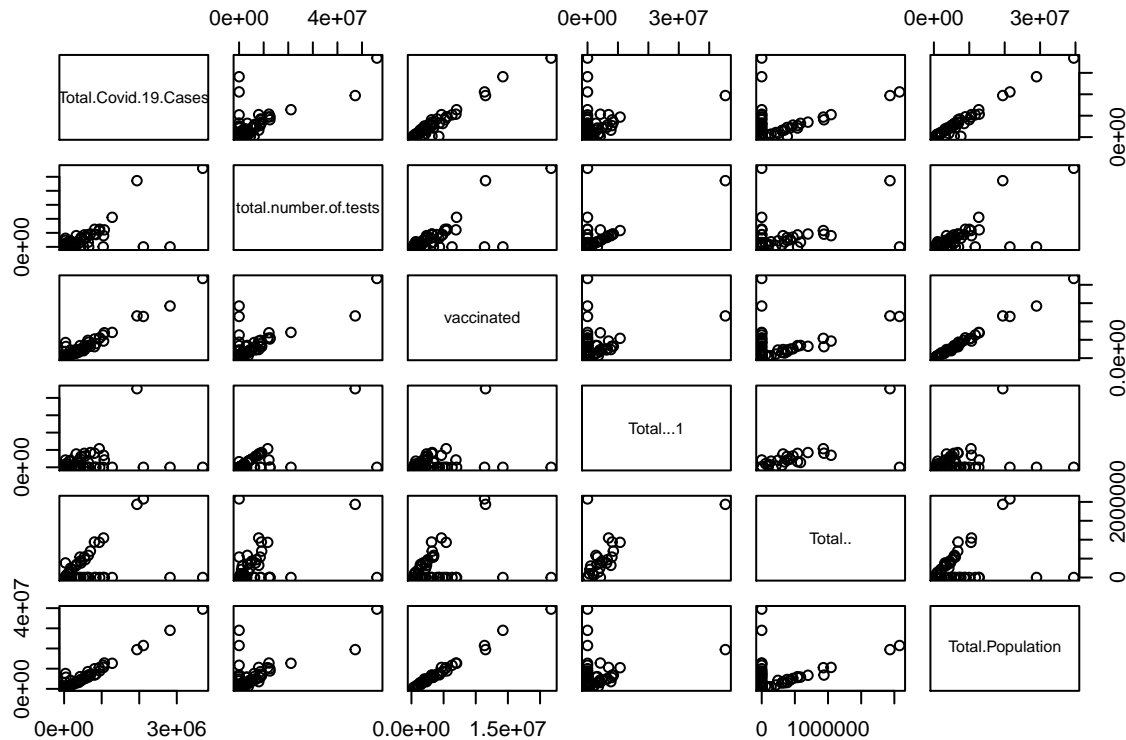
## [4,] 1.660624e-03 5.984520e-01 4.183365e-02 5.534445e-03 3.525192e-01
## [5,] 9.871902e-01 1.784671e-03 2.459296e-03 7.149732e-03 1.416115e-03
## [6,] 3.151831e-04 1.881860e-02 4.518971e-03 9.228941e-04 9.754244e-01
## [7,] 1.704833e-03 5.514757e-01 4.190422e-02 5.653063e-03 3.992622e-01
## [8,] 1.427767e-05 1.045386e-03 2.221844e-04 4.267587e-05 9.986755e-01
## [9,] 1.147320e-04 9.255175e-05 1.855037e-04 9.995473e-01 5.989539e-05
## [10,] 8.931842e-04 1.863596e-02 9.690107e-01 5.109114e-03 6.351001e-03
## [11,] 4.964990e-04 2.779832e-02 6.929342e-03 1.443772e-03 9.633321e-01
## [12,] 4.316968e-04 4.829010e-02 7.841512e-03 1.339211e-03 9.420975e-01
## [13,] 1.310479e-02 1.119090e-01 7.238669e-01 1.015319e-01 4.958737e-02
## [14,] 2.889868e-03 6.053990e-01 3.115534e-01 1.205375e-02 6.810393e-02
## [15,] 4.425913e-04 2.522554e-02 6.223824e-03 1.289537e-03 9.668185e-01
## [16,] 1.723882e-03 4.793135e-01 4.085906e-02 5.672909e-03 4.724307e-01
## [17,] 4.205329e-04 9.452908e-01 1.412611e-02 1.481993e-03 3.868052e-02
## [18,] 2.787571e-04 9.654784e-01 9.753685e-03 9.896709e-04 2.349952e-02
## [19,] 2.634070e-04 1.606336e-02 3.809537e-03 7.729840e-04 9.790907e-01
## [20,] 5.686870e-04 9.222046e-01 1.842231e-02 1.990762e-03 5.681367e-02
## [21,] 2.114769e-03 7.545785e-01 1.747553e-01 8.548875e-03 6.000265e-02
## [22,] 2.594908e-03 2.025395e-01 7.433658e-01 1.181743e-02 3.968239e-02
## [23,] 1.426987e-04 9.853790e-01 6.827797e-03 5.340912e-04 7.116441e-03
## [24,] 1.723992e-03 4.794712e-01 4.086592e-02 5.673427e-03 4.722654e-01
## [25,] 8.295594e-04 9.147110e-01 4.964891e-02 3.211371e-03 3.159914e-02
## [26,] 2.043391e-06 1.551474e-04 3.224926e-05 6.128933e-06 9.998044e-01
## [27,] 8.287288e-04 1.116659e-01 1.600771e-02 2.607595e-03 8.688900e-01
## [28,] 1.723874e-03 4.793456e-01 4.086000e-02 5.672928e-03 4.723976e-01
## [29,] 4.337670e-05 3.044381e-03 6.639033e-04 1.291195e-04 9.961192e-01
## [30,] 1.116302e-04 3.823494e-03 9.944062e-01 5.738290e-04 1.084840e-03
## [31,] 5.512003e-04 6.533947e-02 1.021019e-02 1.717744e-03 9.221814e-01
## [32,] 8.985983e-03 1.131282e-02 2.507108e-02 9.476349e-01 6.995256e-03
## [33,] 2.254056e-04 8.241282e-03 9.881111e-01 1.144998e-03 2.277251e-03
## [34,] 2.939097e-06 2.221605e-04 4.630564e-05 8.811768e-06 9.997198e-01
## [35,] 6.728044e-04 1.468051e-02 9.759138e-01 3.806320e-03 4.926566e-03
## [36,] 3.413406e-04 9.567818e-01 1.171760e-02 1.207678e-03 2.995157e-02
## [37,] 2.847055e-04 2.946957e-02 5.034914e-03 8.776874e-04 9.643331e-01
## [38,] 1.430618e-03 2.732561e-02 9.532784e-01 8.371349e-03 9.594058e-03
## [39,] 7.747571e-05 6.936606e-03 1.300860e-03 2.359091e-04 9.914491e-01
## [40,] 3.768012e-04 9.617401e-01 1.979120e-02 1.430804e-03 1.666106e-02
## [41,] 5.694961e-06 4.587984e-04 9.193660e-05 1.717648e-05 9.994264e-01
## [42,] 2.667673e-03 2.145744e-01 7.292480e-01 1.210712e-02 4.140277e-02
## [43,] 4.230076e-01 4.244877e-02 6.694682e-02 4.365142e-01 3.108257e-02
## [44,] 1.053509e-03 8.280397e-01 3.083199e-02 3.617608e-03 1.364572e-01
## [45,] 6.202152e-04 3.348153e-02 8.523089e-03 1.796328e-03 9.555788e-01
## [46,] 1.652653e-03 8.187627e-01 1.212740e-01 6.579585e-03 5.173112e-02
## [47,] 4.192341e-04 2.408205e-02 5.914724e-03 1.222511e-03 9.683615e-01
## [48,] 9.934889e-05 9.074269e-03 1.680314e-03 3.030411e-04 9.888430e-01
## [49,] 1.882459e-03 7.881573e-01 1.463055e-01 7.550688e-03 5.610406e-02
## [50,] 2.336684e-04 1.444250e-02 3.398111e-03 6.866731e-04 9.812390e-01
##
## Closest hard clustering:
## [1] 2 5 3 2 1 5 2 5 4 3 5 5 3 2 5 2 2 2 5 2 2 3 2 2 2 5 5 2 5 3 5 4 3 5 3 2 5 3
## [39] 5 2 5 3 4 2 5 2 5 5 2 5
##
## Available components:
## [1] "centers"      "size"         "cluster"      "membership"   "iter"

```

```
## [6] "withinerror" "call"
```

```
#Data Visualization
```

```
#Scatterplot finding the relationship between number of tests, number vaccinated, total covid cases, t  
pairs(~ Total.Covid.19.Cases + total.number.of.tests + vaccinated + Total...1 + Total.. + Total.Populat.
```



```
#make data look good
```

```
#Scatterplot matrix
```

```
scatterplotMatrix(~Total.Covid.19.Cases + vaccinated + total.number.of.tests + Total.. + Total...1 + To
```

```
## Warning in plot.window(...): "id.n" is not a graphical parameter
```

```
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
```

```
## Warning in title(...): "id.n" is not a graphical parameter
```

```
## Warning in plot.window(...): "id.n" is not a graphical parameter
```

```
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
```

```
## Warning in title(...): "id.n" is not a graphical parameter
```

```
## Warning in axis(side = side, at = at, labels = labels, ...): "id.n" is not a  
## graphical parameter
```

```
## Warning in plot.window(...): "id.n" is not a graphical parameter
```

```
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
```

```
## Warning in title(...): "id.n" is not a graphical parameter
```

```
## Warning in plot.window(...): "id.n" is not a graphical parameter
```

```
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
```

```
## Warning in title(...): "id.n" is not a graphical parameter
```

[illegible]

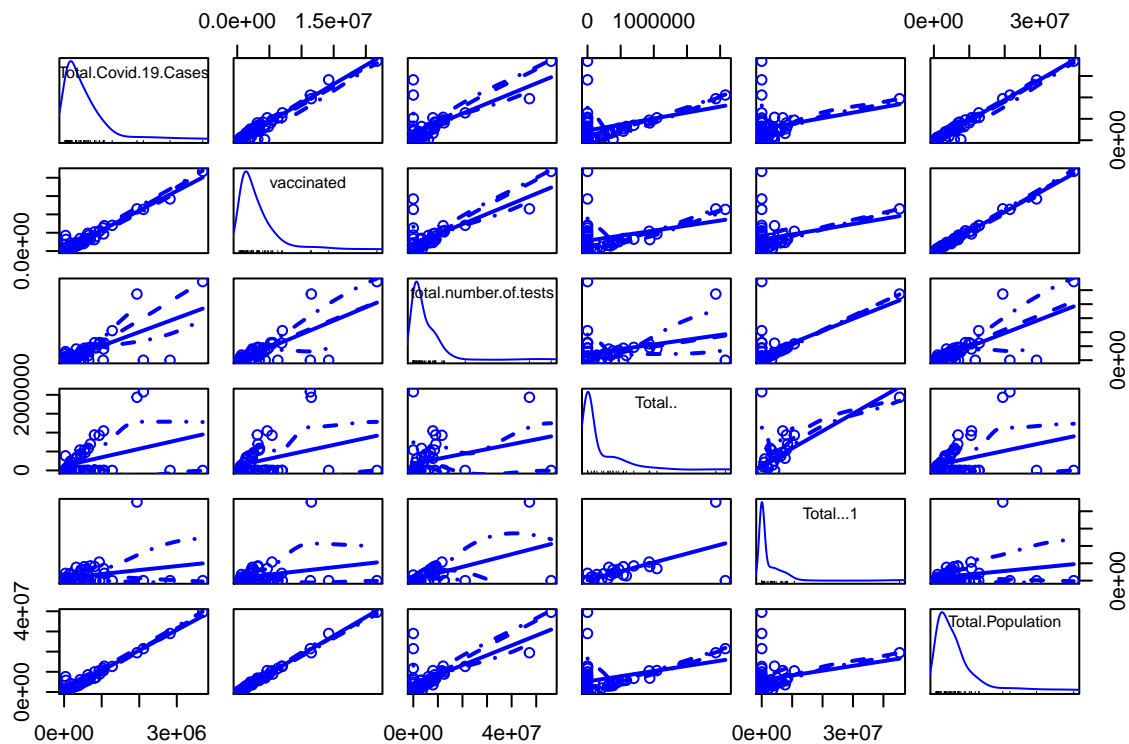
```

## Warning in title(...): "id.n" is not a graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "id.n" is not a
## graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "id.n" is not a
## graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in smoother(x[subs], y[subs], col = smoother.args$col[i], log.x =
## FALSE, : could not fit smooth
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter

```

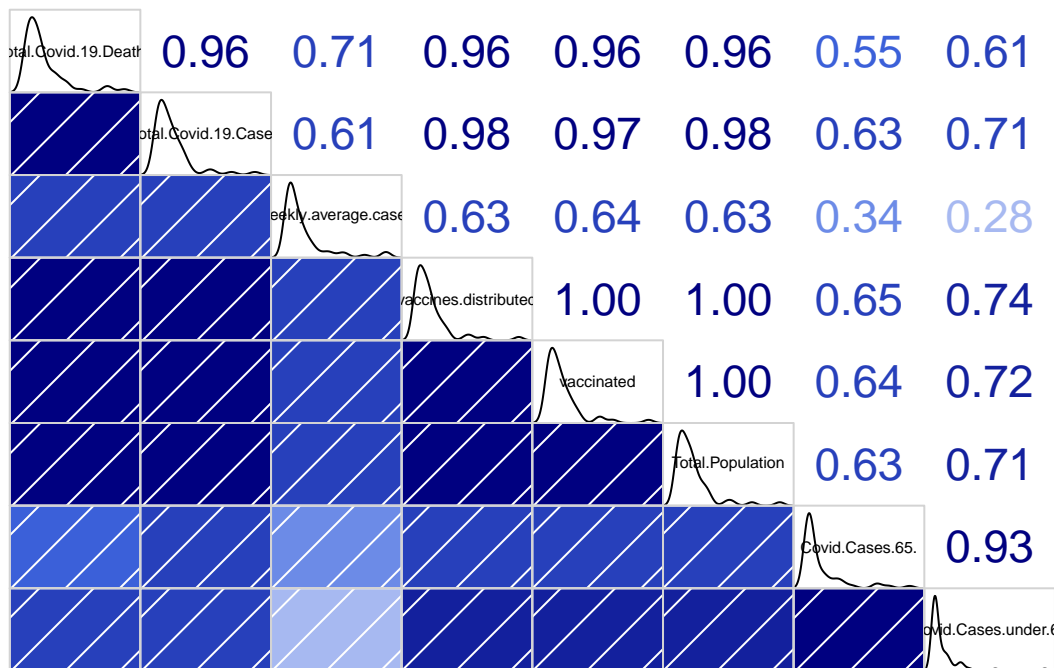


```
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "id.n" is not a
## graphical parameter
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in title(...): "id.n" is not a graphical parameter
```

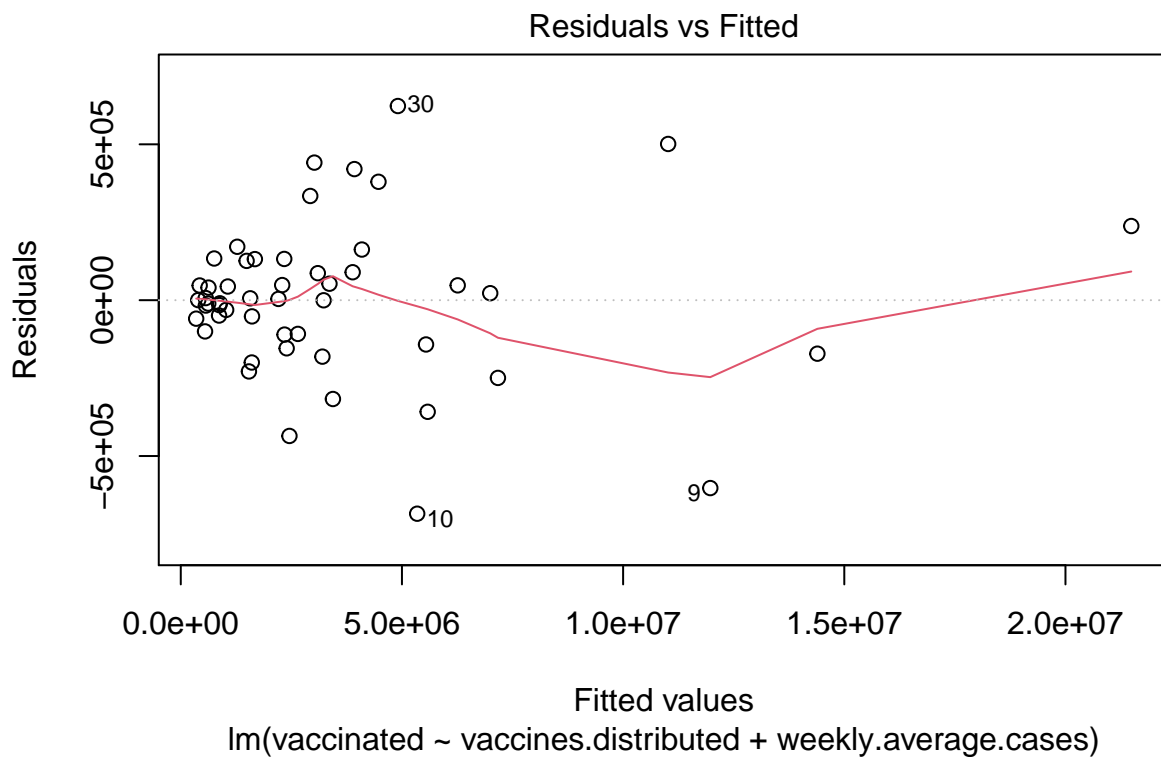


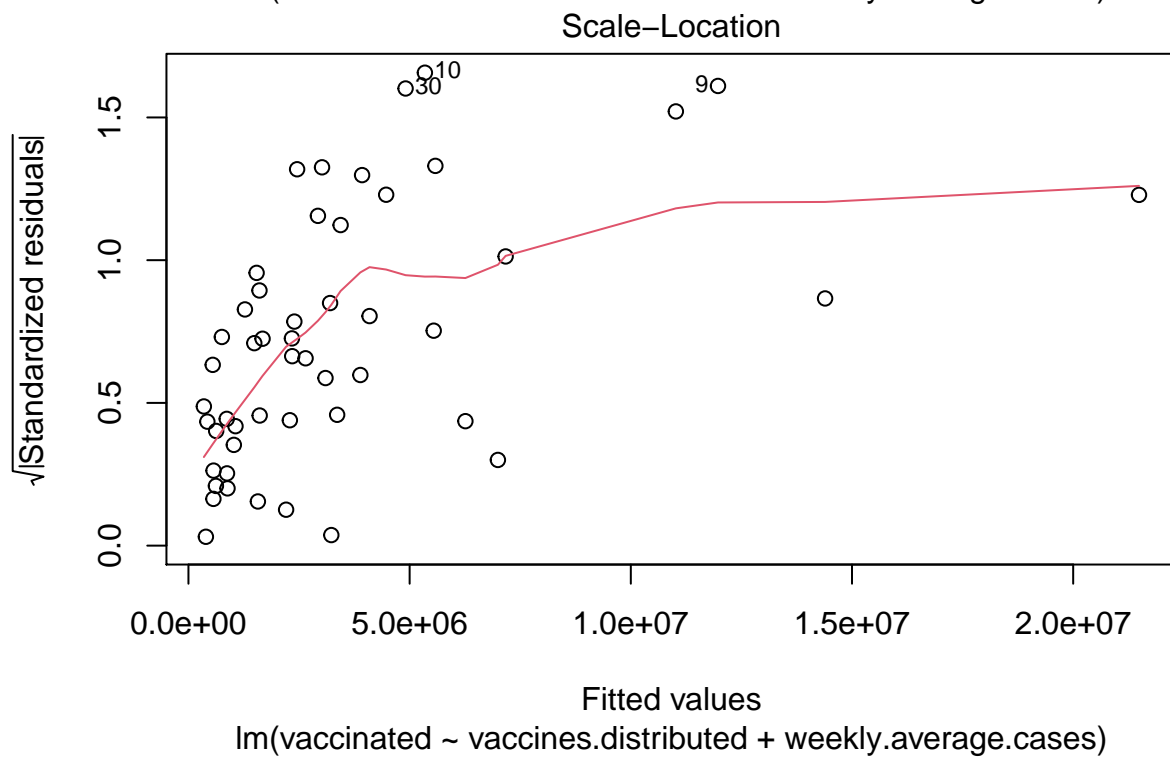
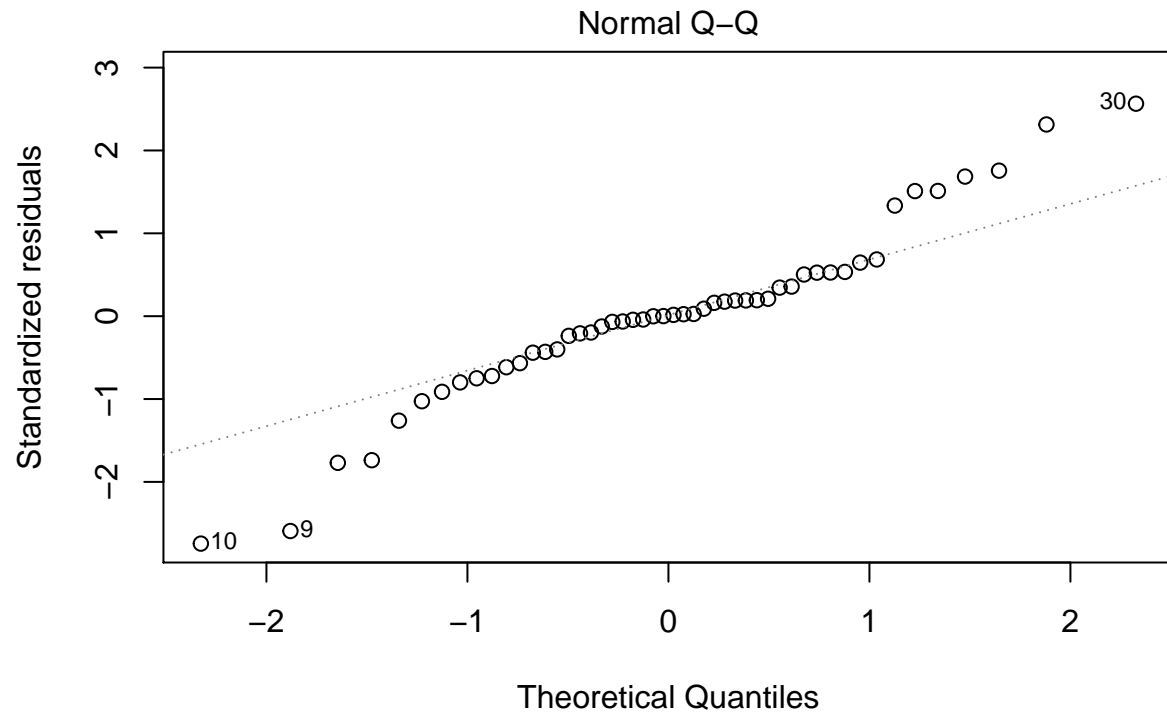
*#there is a strong linear relationship between cases and vaccinated, cases and population, vaccinated and population*

```
#correlation plots
corrgram(dat1[1:8], upper.panel=panel.cor, diag.panel=panel.density)
```

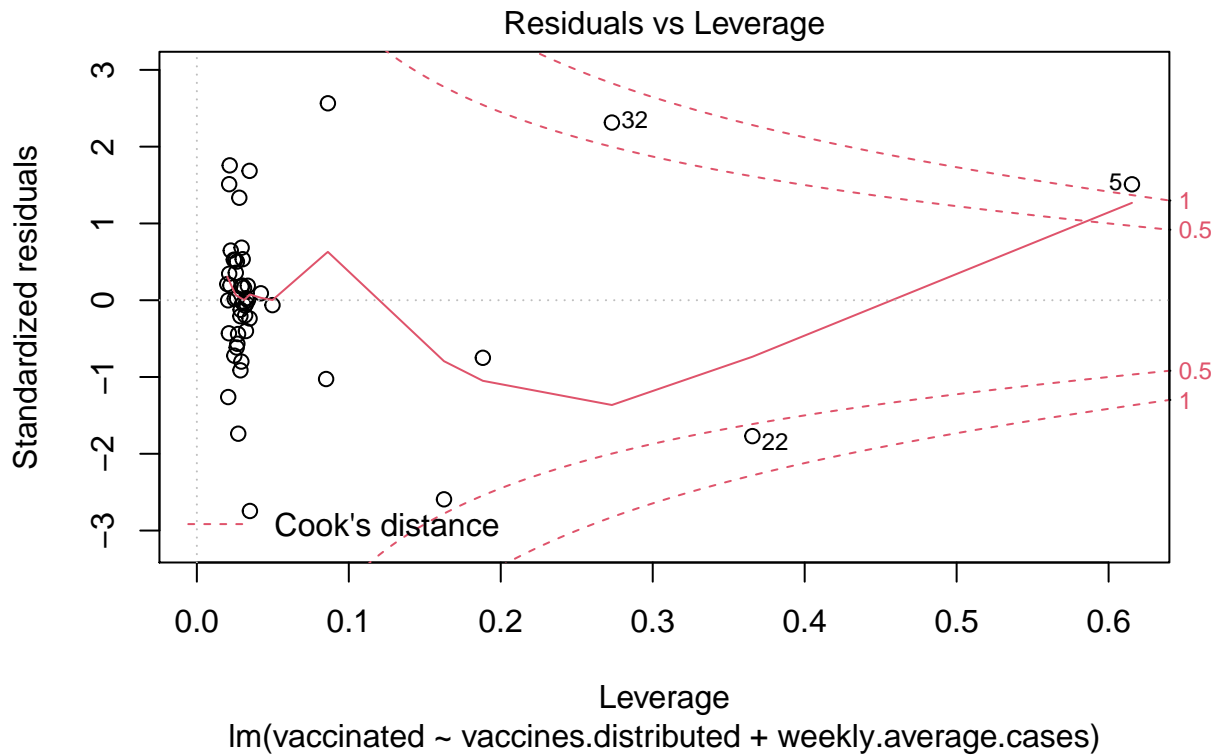


```
#added variable plot
dat1.mod <- lm(vaccinated~ vaccines.distributed + weekly.average.cases, data=dat1)
plot(dat1.mod)
```









```
avPlots(dat1.mod, id.n=2, ellipse=TRUE)
```

```
## Warning in plot.window(...): "id.n" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "id.n" is not a
## graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "id.n" is not a
## graphical parameter

## Warning in box(...): "id.n" is not a graphical parameter

## Warning in title(...): "id.n" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "id.n" is not a graphical
## parameter

## Warning in plot.window(...): "id.n" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "id.n" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "id.n" is not a
## graphical parameter

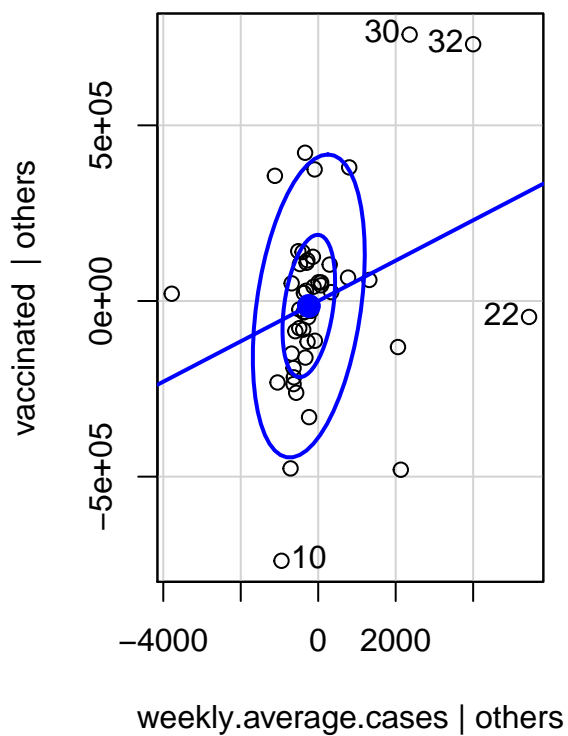
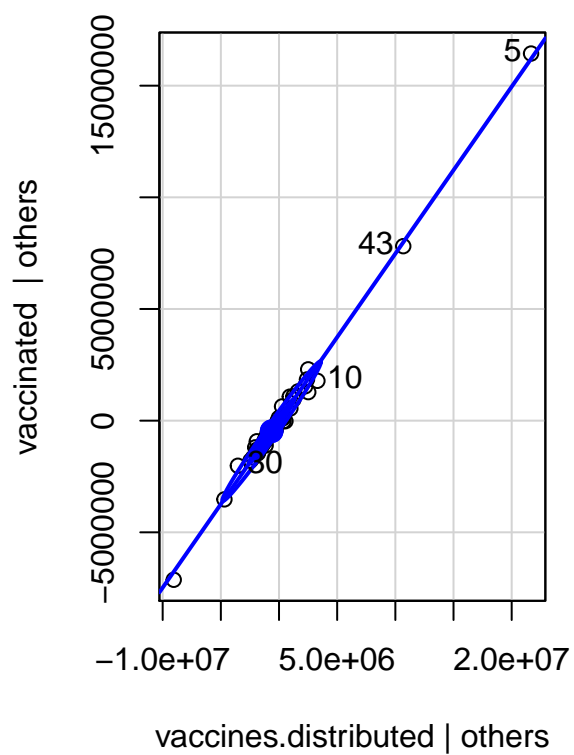
## Warning in axis(side = side, at = at, labels = labels, ...): "id.n" is not a
## graphical parameter

## Warning in box(...): "id.n" is not a graphical parameter

## Warning in title(...): "id.n" is not a graphical parameter

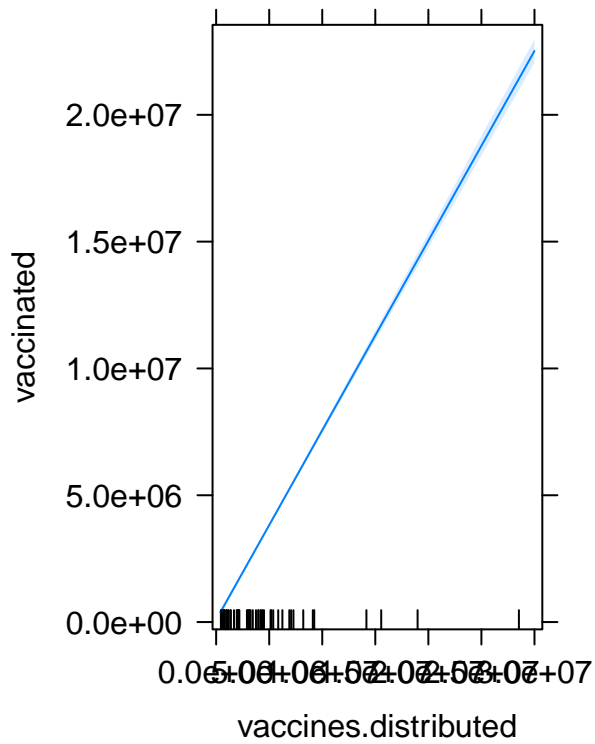
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "id.n" is not a graphical
## parameter
```

## Added-Variable Plots

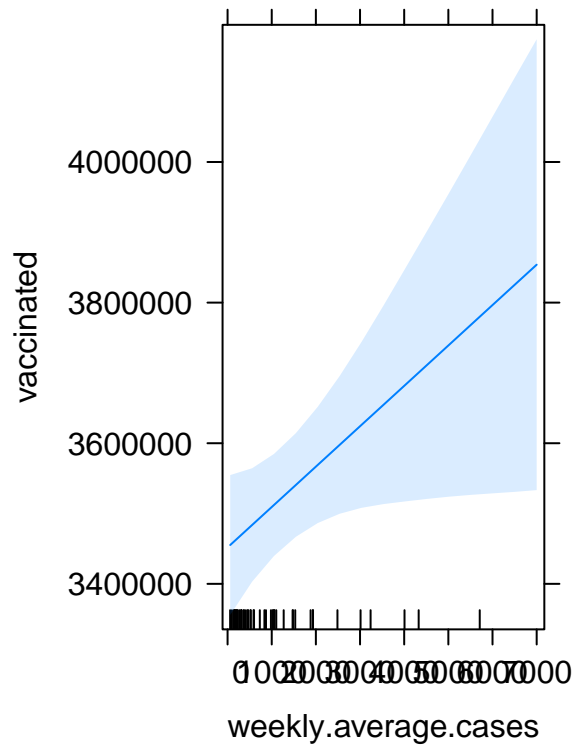


```
#effects plot
dat1.eff1 <- allEffects(dat1.mod)
plot(dat1.eff1)
```

**vaccines.distributed effect plot**



**weekly.average.cases effect plot**

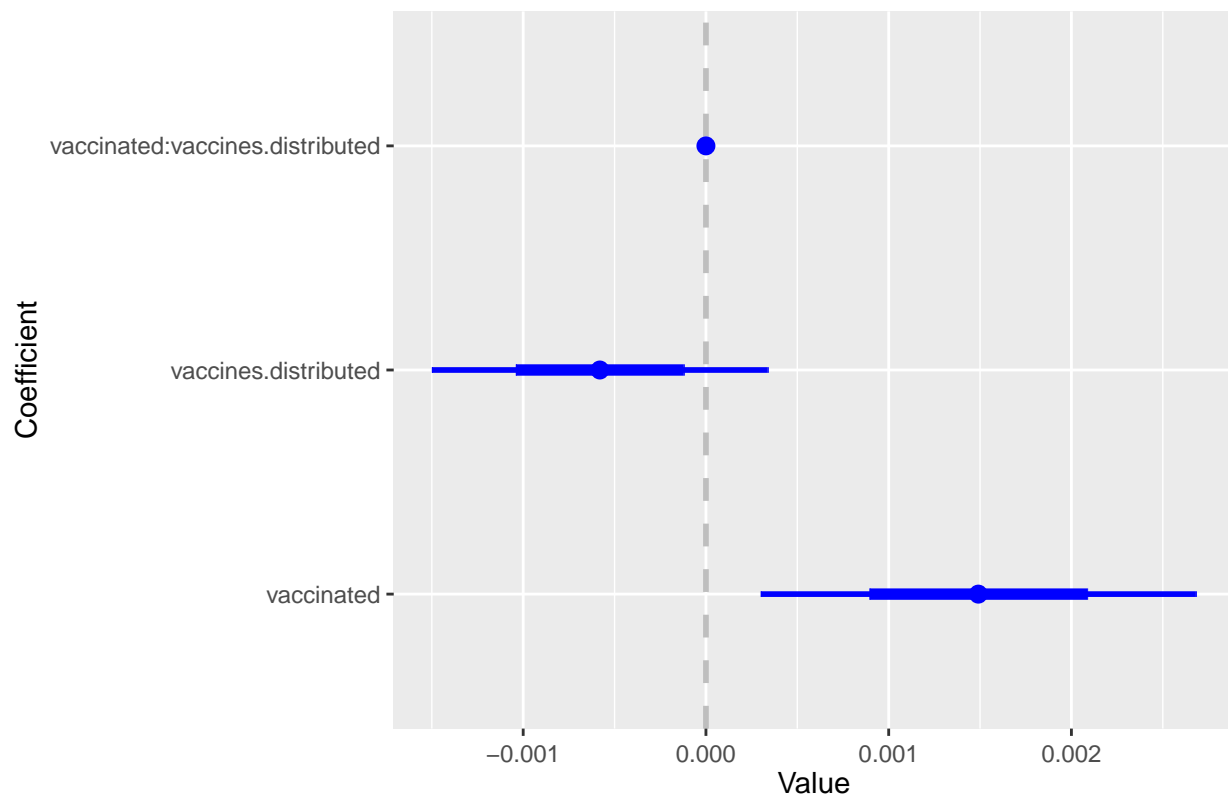


```
#coefficient plot
```

```
dat1.mod2 <- lm(weekly.average.cases~ vaccinated * vaccines.distributed, data=dat1)
```

```
coefplot(dat1.mod2, intercept=FALSE, lwdInner=2, lwdOuter=1, title="Coefficient Plot for weekly cases v
```

Coefficient Plot for weekly cases vs vaccinated \* vaccine

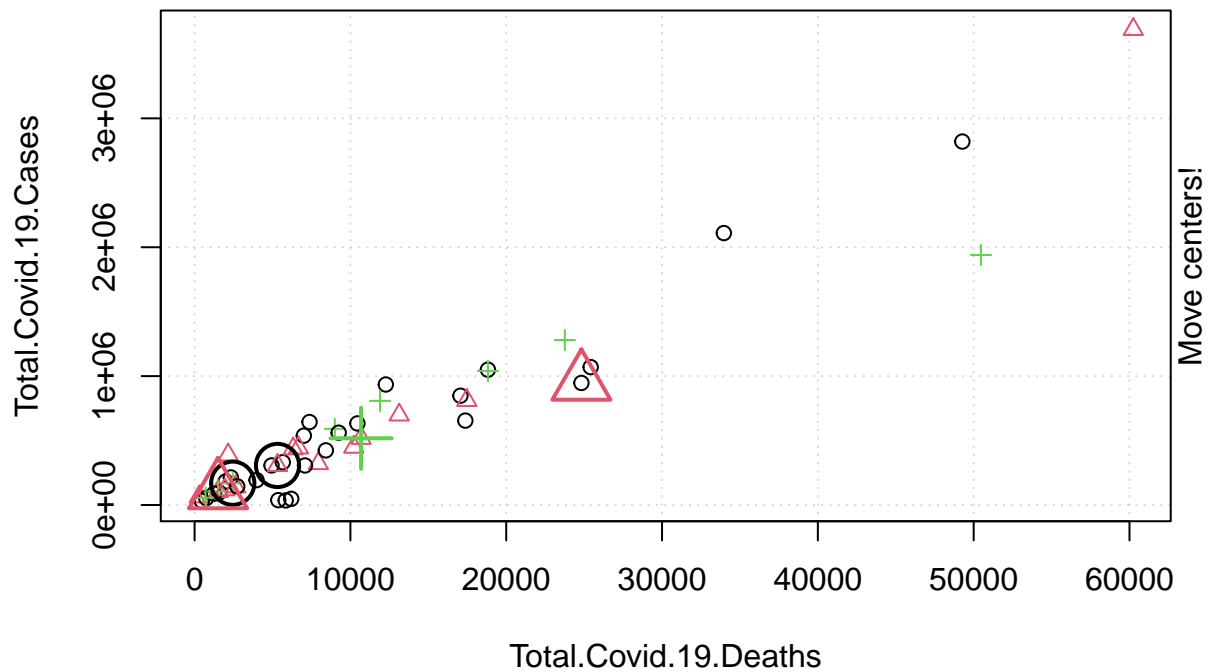


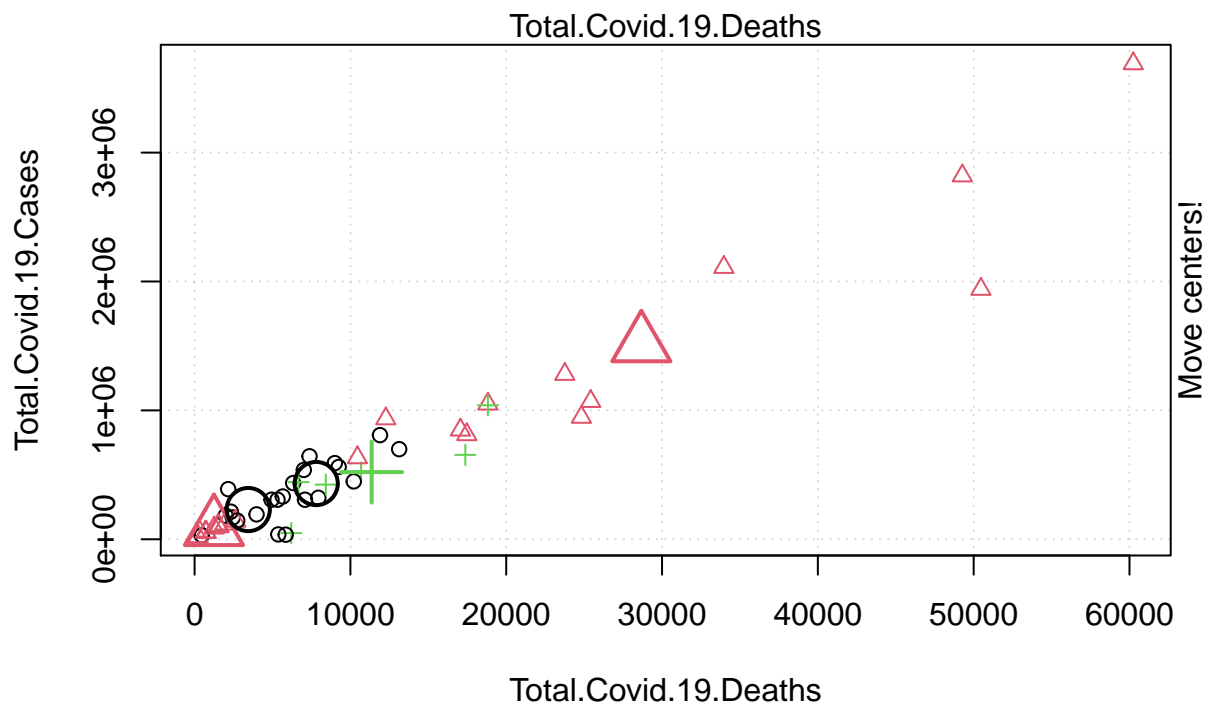
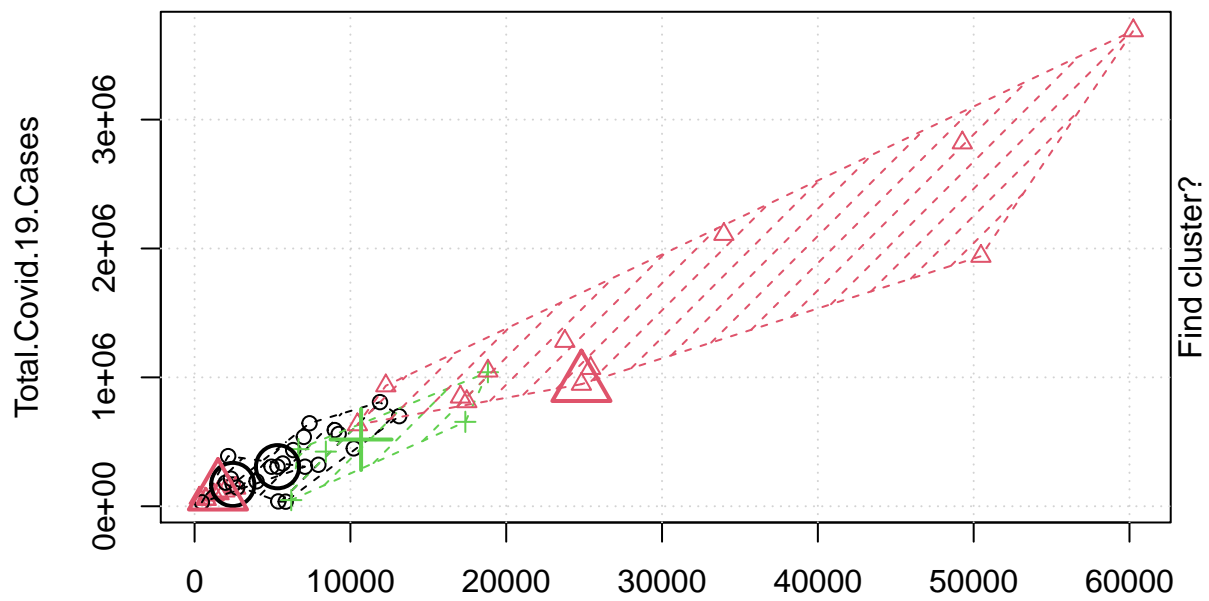
#Data Animations

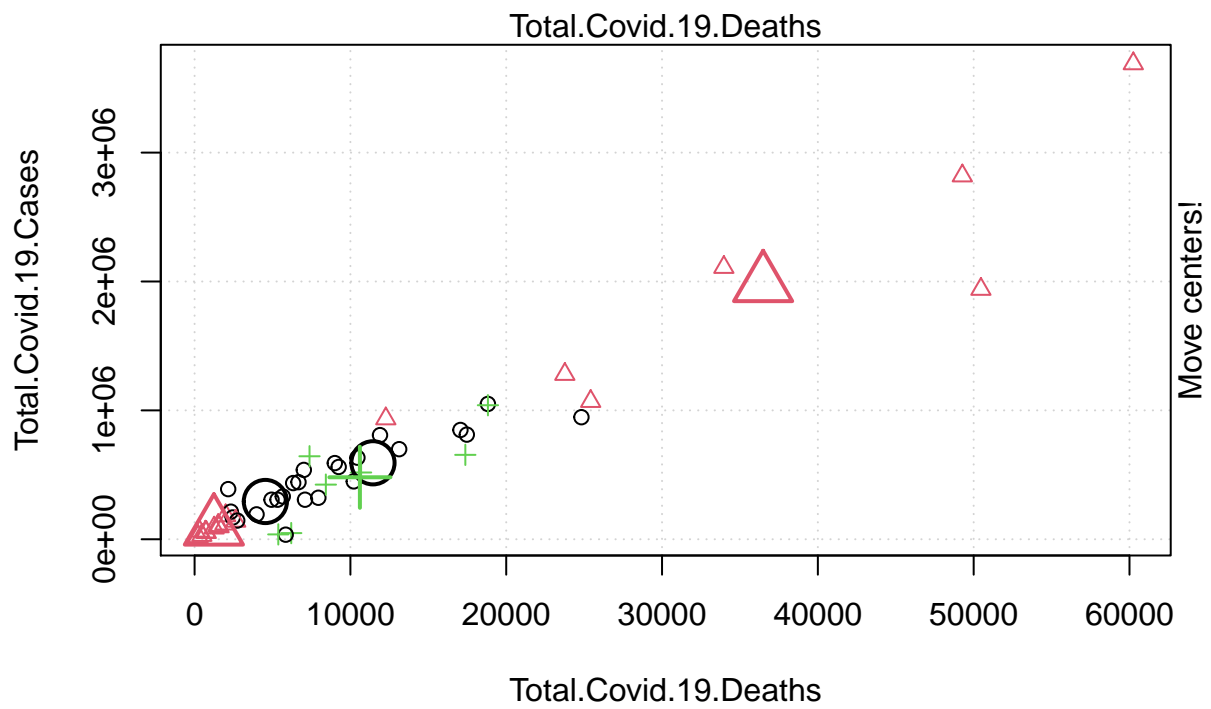
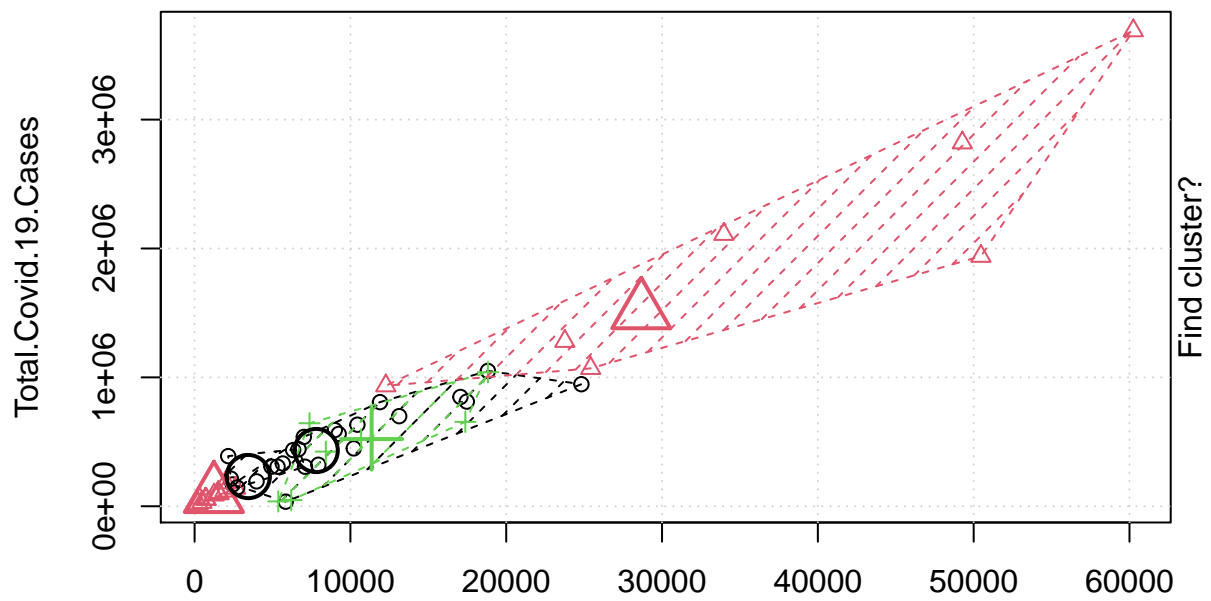
*#kmeans animation*

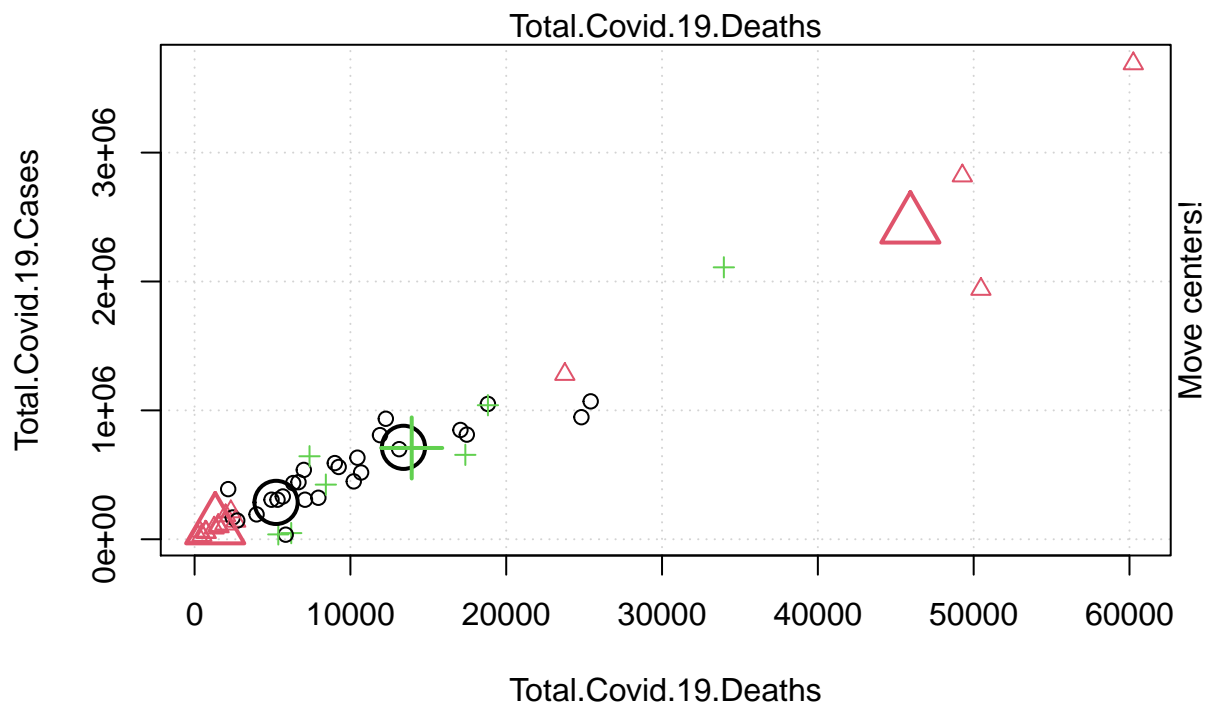
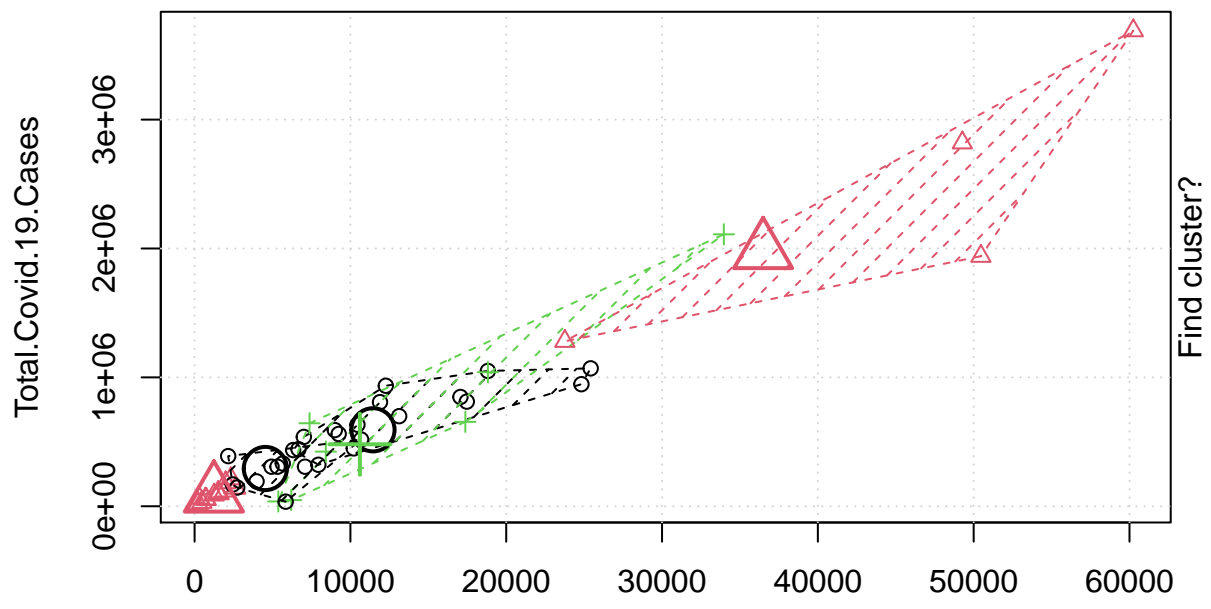
ani.options(interval = 1)

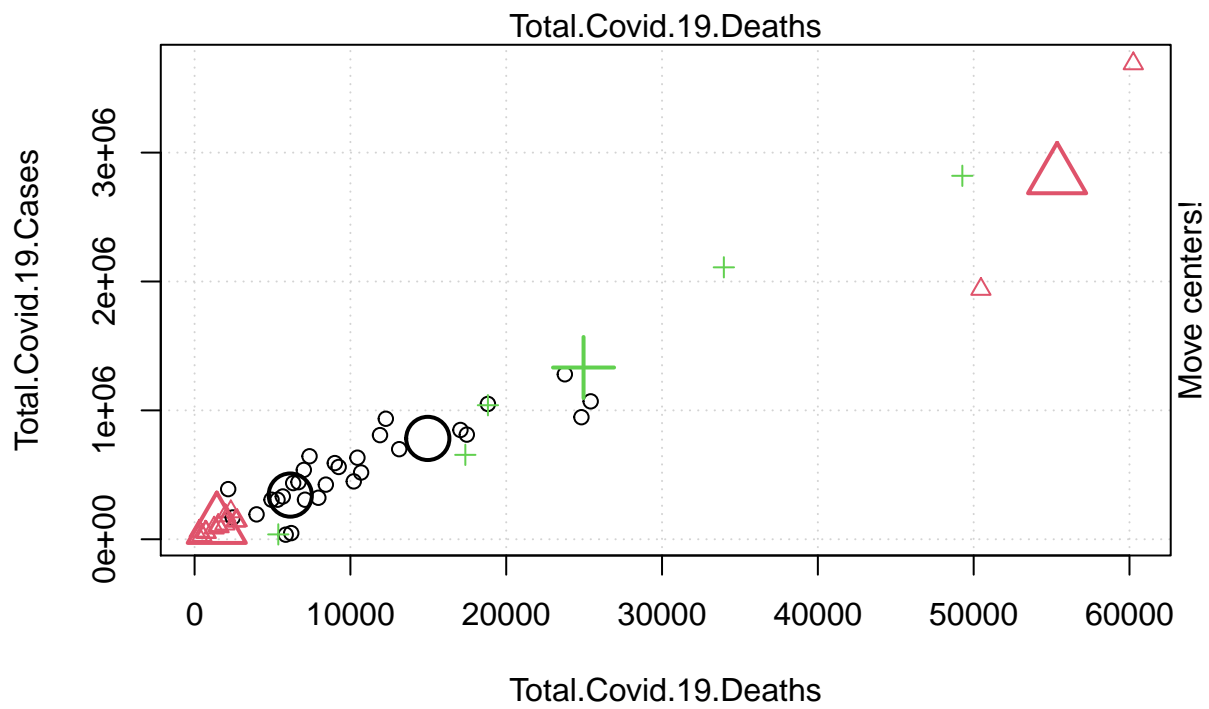
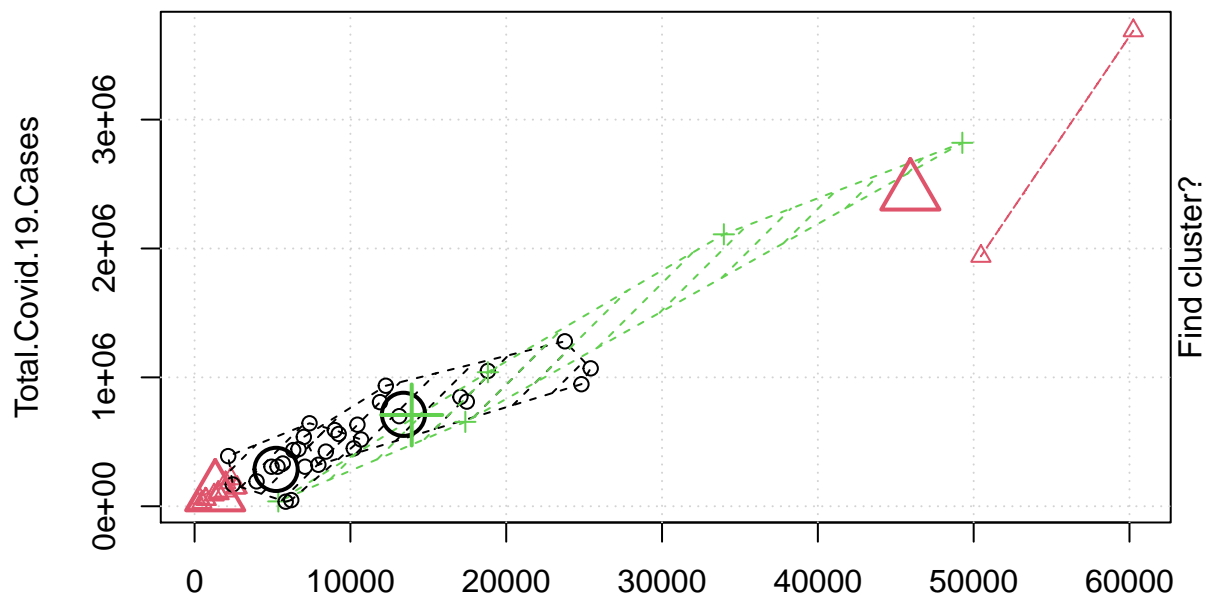
kmeans.ani(dat1, centers = 5)



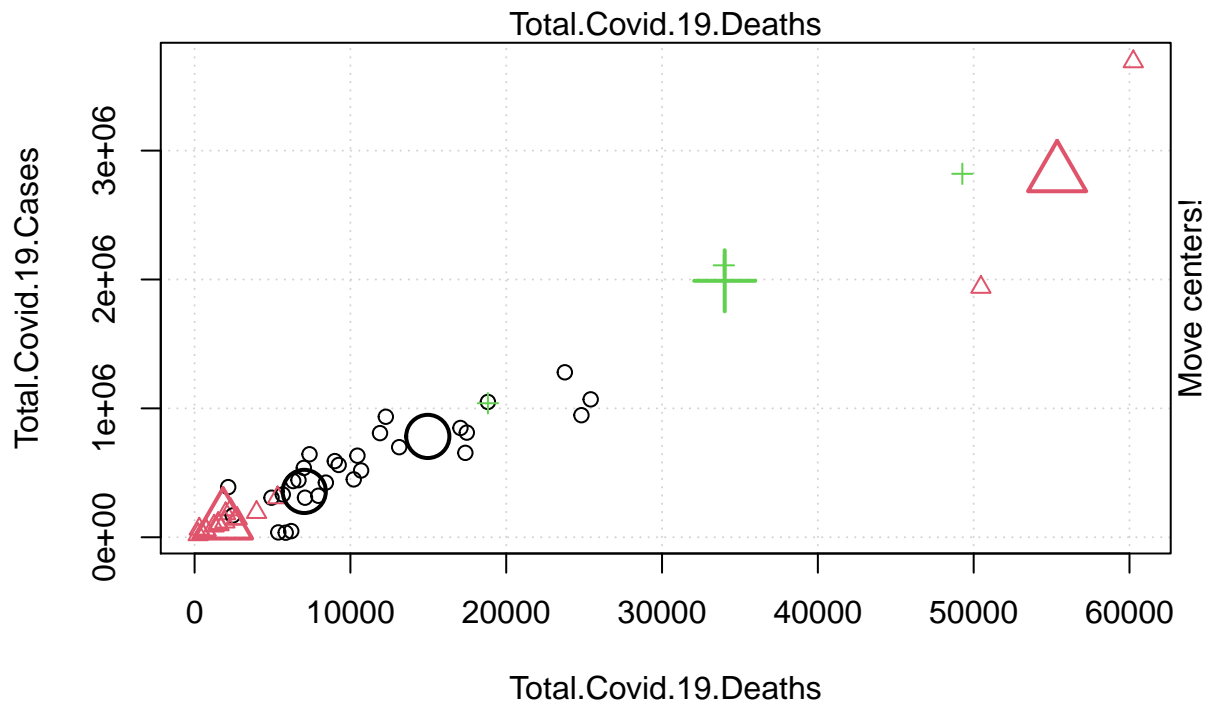
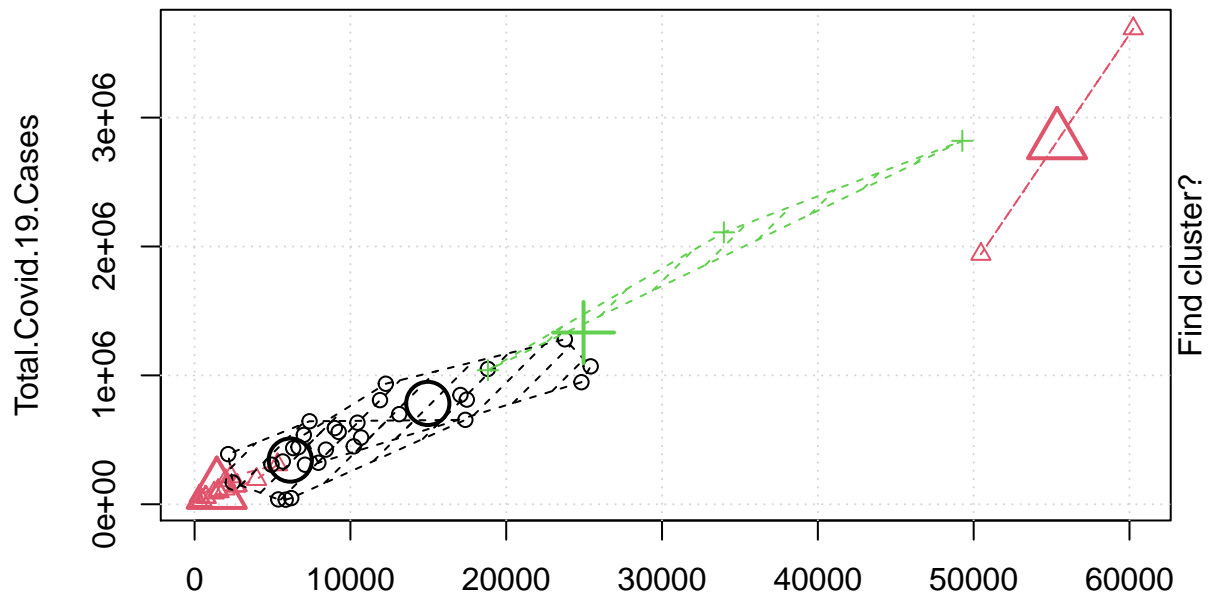


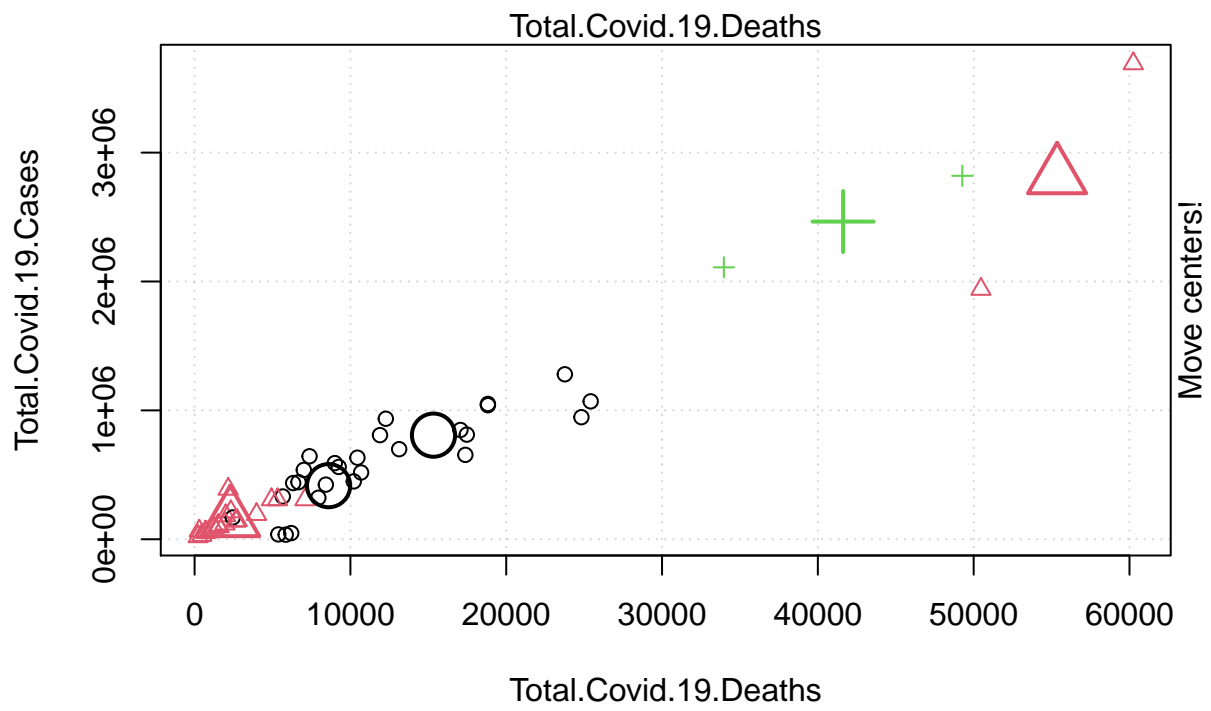
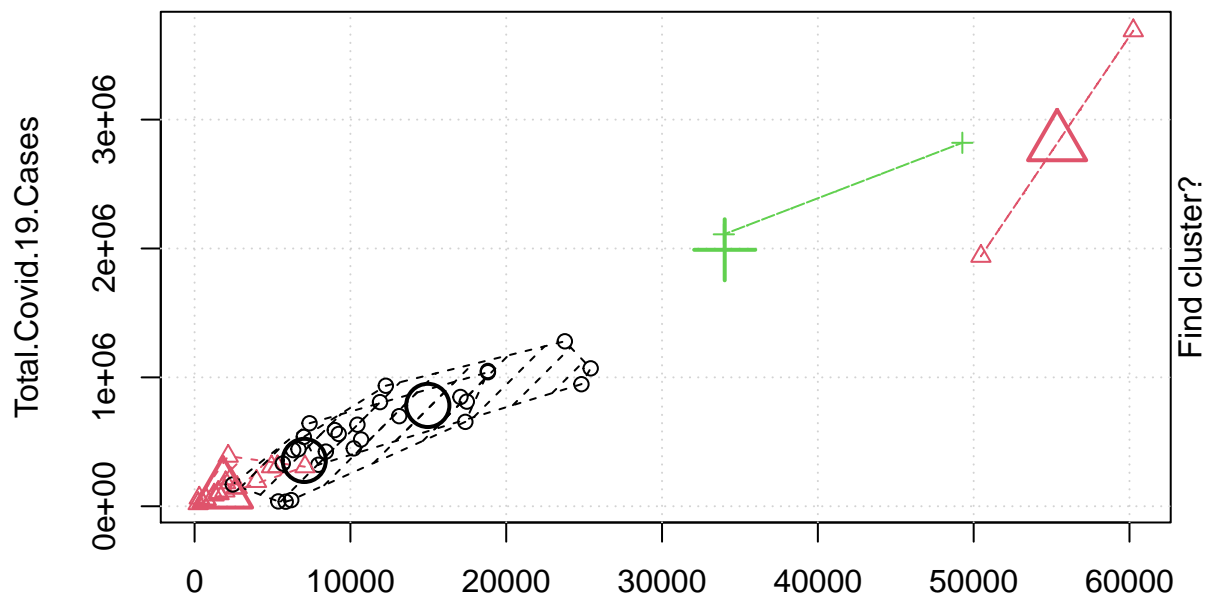


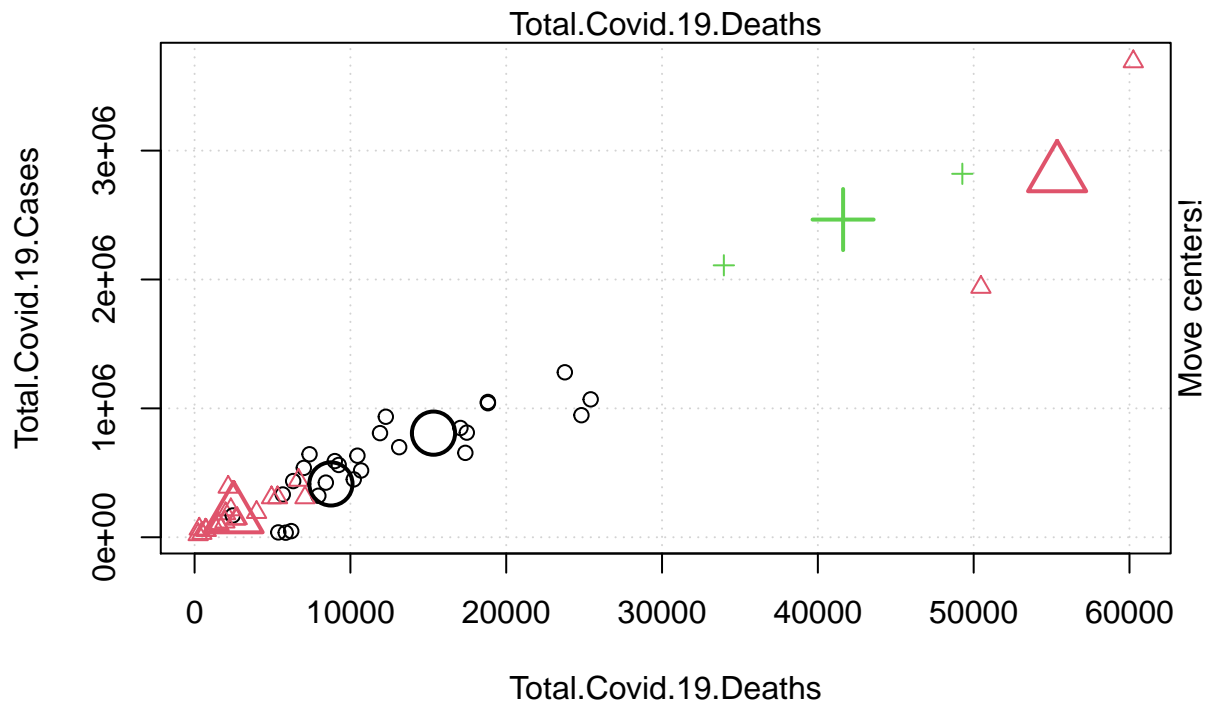
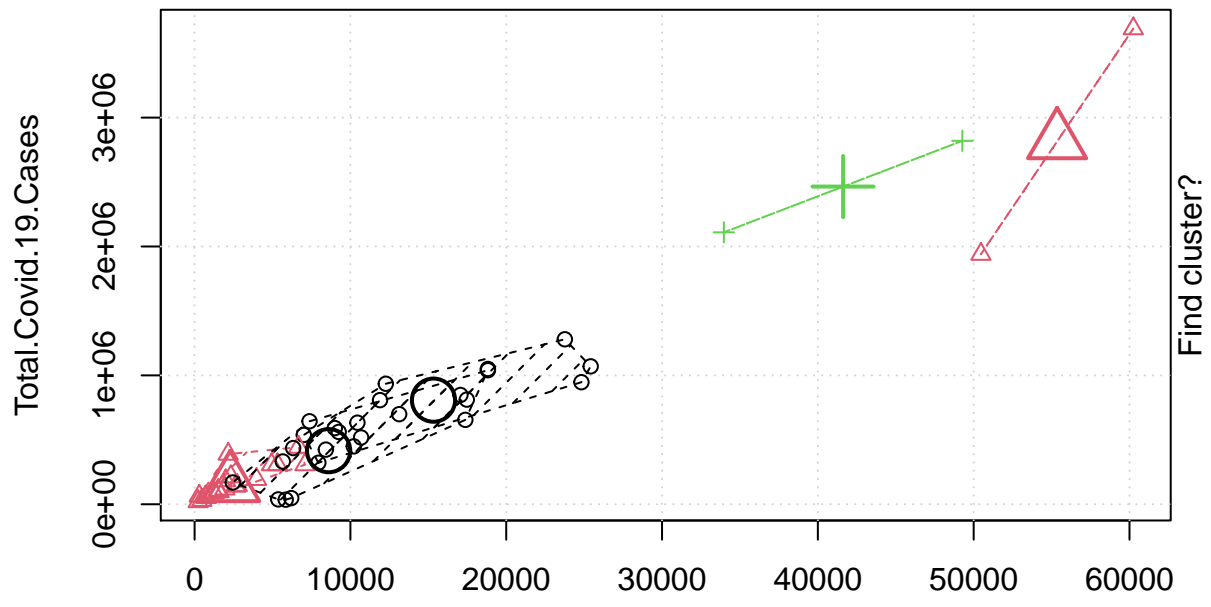


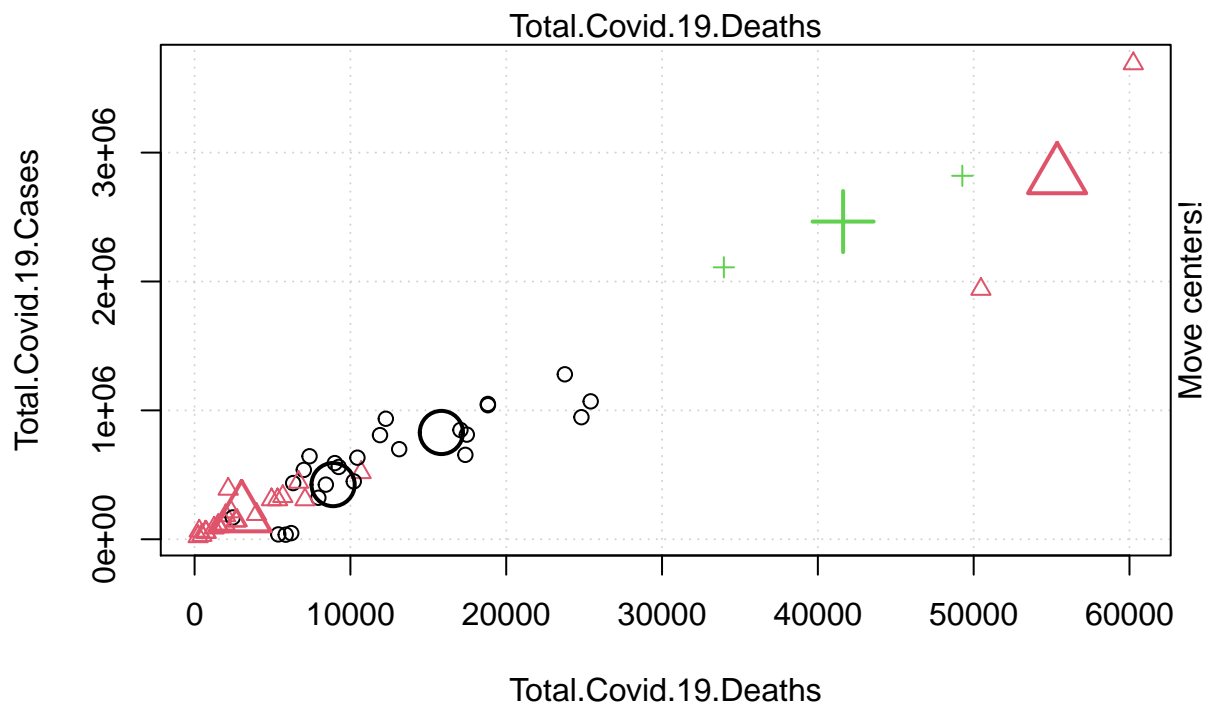
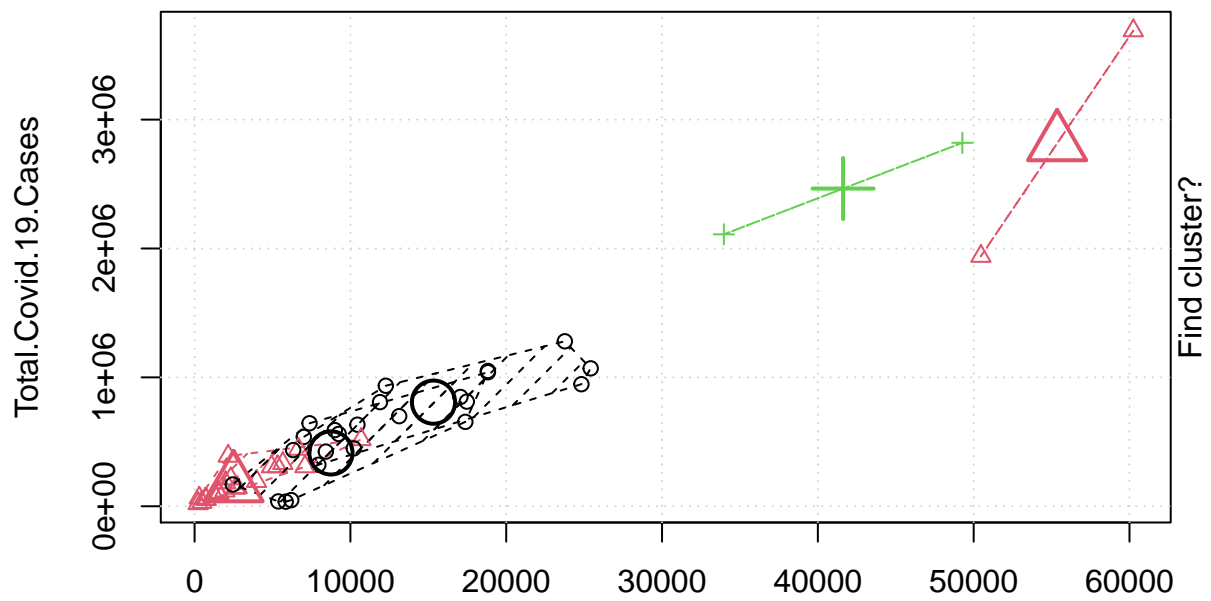


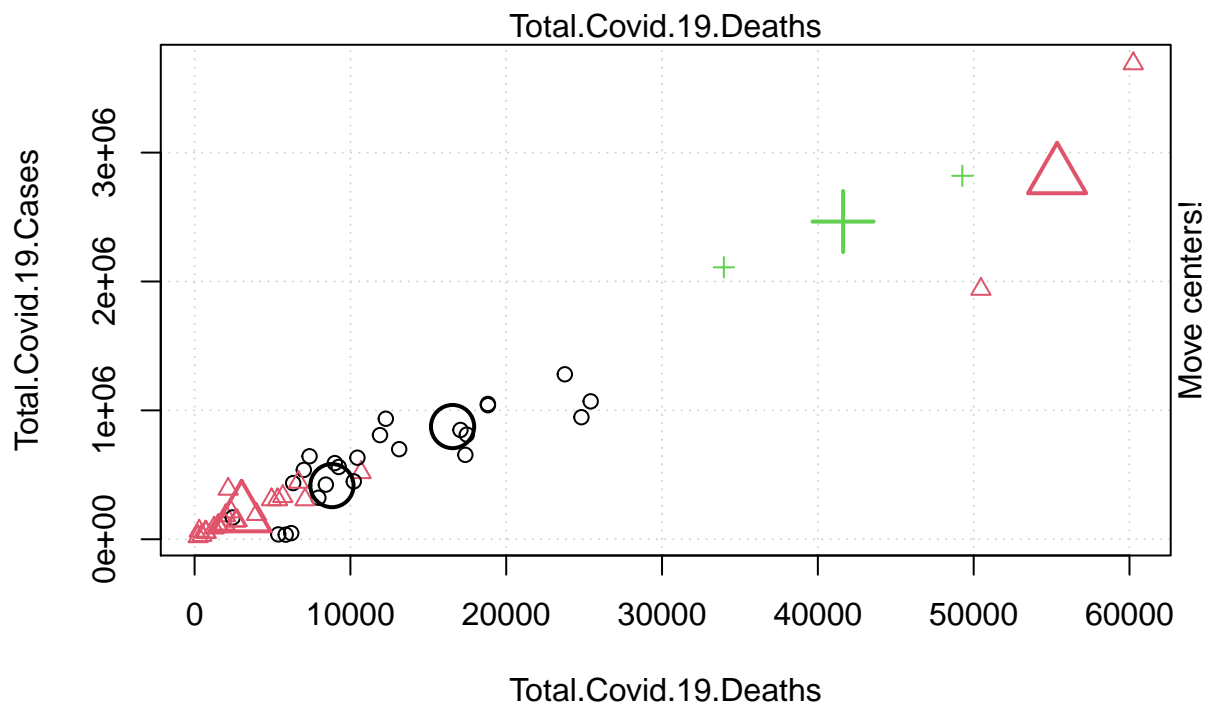
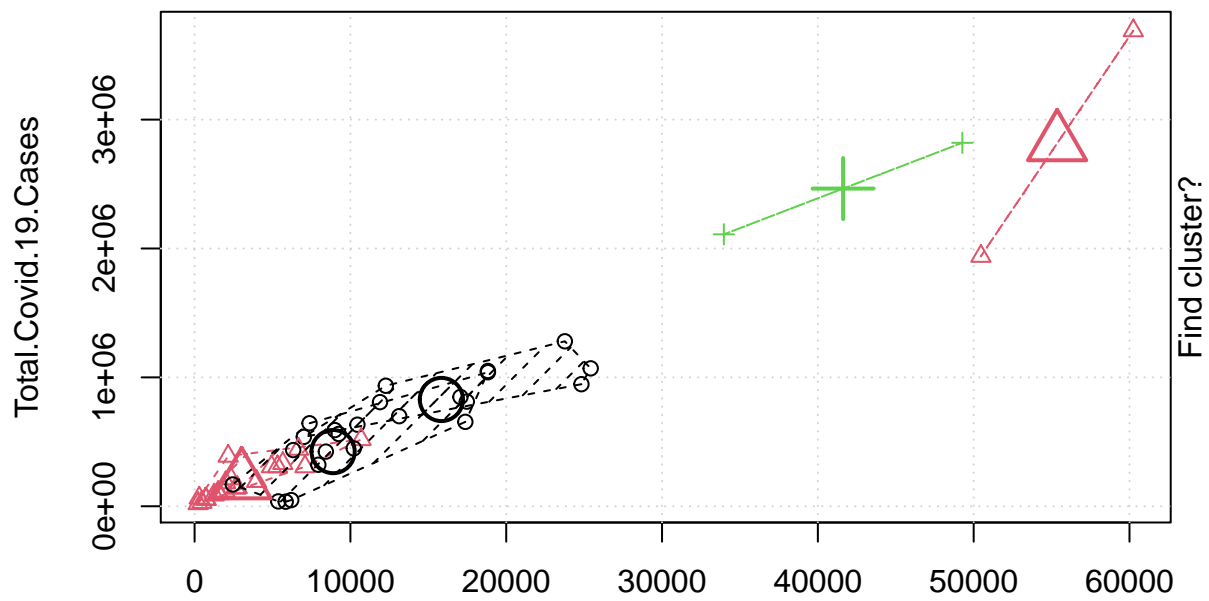


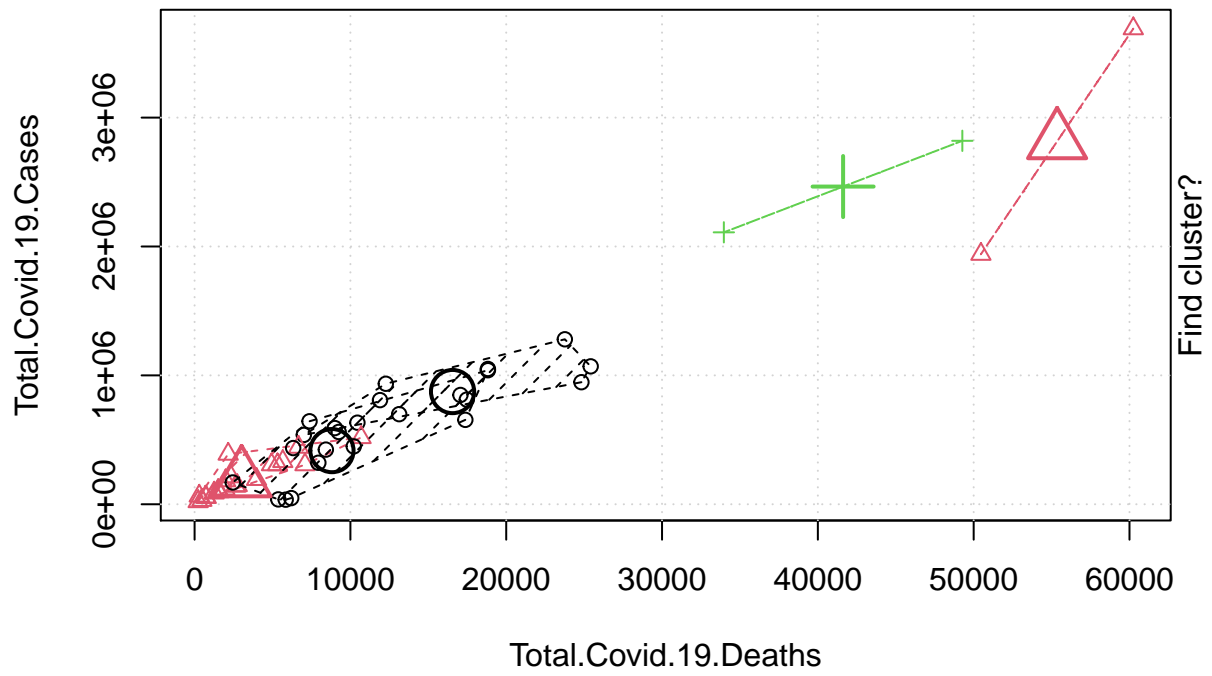












#end of project