STA207 Project: Covid-19 Analysis

Wun-Syuan Wu

March 2022

I. Abstract

The goal of this report is to examine the relationship between the total number of cases in different countries in 2022 and types and frequency of vaccinations. Poisson regression, negative binomial regression and multiple linear regression were the three types of models used to achieve this goal. We observed that the countries that used 5 different types of vaccines had a lesser cases as compared to 3 types. Additionally, the poly-vaccination (>2 doses) rates had a positive relation with total cases. We hope that the results of this study help governments make informed decisions about vaccination policies.

II. Introduction

2.1 Background

It has been almost two years since the Covid-19 pandemic began, and multiple strategies have been constructed to fight the SARS-Cov-2 virus. For example, different vaccines were invented by bio-tech companies and were injected in almost every country. There are at least 35 approved covid vaccines all around the world, and at least 197 countries used those approval vaccines[1]. This quantity implies that vaccination could be one of the most effective way to prevent the spread of this unpleasant virus. Moreover, some policies such as mask mandates and mobility restrictions were imposed to prevent people from getting infected [2].
In addition, a great amount of articles discussing about vaccination and restrictions. Some stated that the increase of vaccination rate can effectively reduce the severe cases[3] [4], some stated that vaccination might not be effective against some variant of covid virus[5] [6]. Still as people know, lots of countries still insist on giving mask mandate in order to prevent another outbreak of covid.

Therefore, in this project we would like to focus on the impact of vaccinations on the amount of cases happened in 2022 as to Feb.10 (from 2022-01-01 to 2022-02-10), which is exactly the time for omicron variant became severe all around the world. The question of interests therefore are whether the cumulative cases in the past 41 days of each country can be affected by (1) fully vaccinated rate (2) the time vaccine has been first implemented in a country (3) the number of vaccine types a country used and (4) restrictions(safety guidance) in each country.

2.2 Data

We obtained data from three resources. The first one is the required WHO covid dataset, it countains 182490 observations with 8 variables including countries, new cases, new death, cumulative cases and cumulative deaths. The second one is also from WHO, this data set is vaccine data. It has 228 observations with 14 variables. The 228 observations are coming from the amount of countries in the WHO, and variables such as first vaccine date of a country and the amount of vaccine type a country use are used in this report. In order to answer questions we're interested in, the data from WHO is not enough. We also obtained data about the restriction index from the website Our World in Data[7].

After cleaning and merging data from three sources, the final data set we obtained contains 164 observations with no NA value, and each one represent a country in WHO. The features including:

- `country` : Country names
- `ISO` : the iso number for each country
- `region` : the corresponding areas divided by WHO
- `oneplus_dose_per100` : the amount of people vaccinated at least on dose during the past few month
- `full_dose_per100` : the amount of people that were fully vaccinated during the past few month
- `first_vaccine_date` : the amount of days a country start to vaccinate from now
- `vaccines_types_use` : the amount of different types of vaccines that a country use, I divided the type by the method of the vaccine, and the total is five types.
- column 9-36 : whether a country use the type of vaccine, if yes(1), no(0)
- `total_new_cases` : total new cases detected within the past one month
- `total_new_deaths` : total new deaths reported within the past one month
- `stringency_index` : index of a country restriction level
- `population` : the population in a country

The variables will be mainly used in modeling are as following:

- Response variable: `Cumulative_cases` from WHO covid data.
- Explanatory variable 1: `PERSONS_FULLY_VACCINATED_PER100` from WHO vaccination data.
- Explanatory variable 2: `number_vaccine_type_use` from WHO's vaccination data.
- Explanatory variable 3: `FIRST_VACCINE_DATE` from WHO's vaccination data.
- Explanatory variable 4: `stringency_index` from Our World in Data's data.

To be more clear about the division of vaccine types, we introduce more information here. In WHO covid data, the type data is obtained by counting the total different product of several different brands. Therefore, there are 28 different names of vaccines, we can see from the figure below. For example, United states used "Janssen - Ad26.COV 2-S", "Moderna - Spikevax" and "Pfizer BioNTech - Comirnaty" vaccination, with different name of the vaccination, they are counting as a type. However, we did not divide the type by using vaccine names, since some of the vaccine with different name is with identical formula of vaccine, for instance, AstraZeneca - AZD1222 is the same product as Astrazeneca - Vaxzevria and SII - Covishield.
Instead, I used the different methods of exposure use in a vaccine to be a distinct count. "All vaccines work by exposing the human body to particles or molecules that trigger an immune response,the key difference is the method of exposure used." said by Rachel McArthur from Healthcare IT News[8]. Based on several articles[9] [10], I divided them into five different methods:

1. inactivated virus vaccine, e.g. Sinopharm, Sinovac vaccines.
2. viral vector vaccine, e.g.Oxford-AstraZeneca, Sputnik V
3. mRNA, e.g. Pfizer-BioNTech, Moderna
4. DNA vaccine, e.g.ZyCoV-D.
5. protein subunit vaccine, e.g.Novavax.

```
##  [1] "Beijing CNBG - BBIBP-CorV"   "Janssen - Ad26.COV 2-S"
##  [3] "Pfizer BioNTech - Comirnaty" "SII - Covishield"
##  [5] "AstraZeneca - Vaxzevria"     "Gamaleya - Gam-Covid-Vac"
##  [7] "Sinovac - CoronaVac"         "Moderna - Spikevax"
##  [9] "CanSino - Convidecia"        "Novavax - Covavax"
## [11] "Gamaleya - Sputnik-Light"    ""
## [13] "AstraZeneca - AZD1222"       "Gamaleya - Sputnik V"
## [15] "Bharat - Covaxin"            "Anhui ZL - Recombinant"
## [17] "IMB - Covidful"              "Shenzhen - LV-SMENP-DC"
## [19] "Wuhan CNBG - Inactivated"    "CIGB - CIGB-66"
## [21] "Finlay - Soberana Plus"      "Finlay - Soberana-02"
## [23] "Moderna - mRNA-1273"         "Zydus - ZyCov-D"
## [25] "Shifa - COVIran Barakat"     "RIBSP - QazVac"
## [27] "Julphar - Hayat-Vax"         "SRCVB - EpiVacCorona"
## [29] "Turkovac"
```

The final dataset being used in this project:

```
## # A tibble: 6 × 10
##   COUNTRY     ISO3  WHO_REGION PERSONS_VACCINA… PERSONS_FULLY_V… FIRST_VACCINE_D…
##   <fct>       <fct> <fct>                 <dbl>            <dbl>             <dbl>
## 1 Afghanistan AFG   EMRO                   11.9             10.2               352
## 2 Albania     ALB   EURO                   44.3             40.4               392
## 3 Algeria     DZA   AFRO                   16.5             13.2               375
## 4 Andorra     AND   EURO                   75.8             69.7               385
## 5 Angola      AGO   AFRO                   30.4             14.8               336
## 6 Argentina   ARG   AMRO                   88.5             77.9               407
## # … with 4 more variables: vaccine_type_used <fct>,
## #   total_new_cases_per_million <dbl>, total_new_deaths_per_million <dbl>,
## #   stringency_index <dbl>
```

III. Analysis

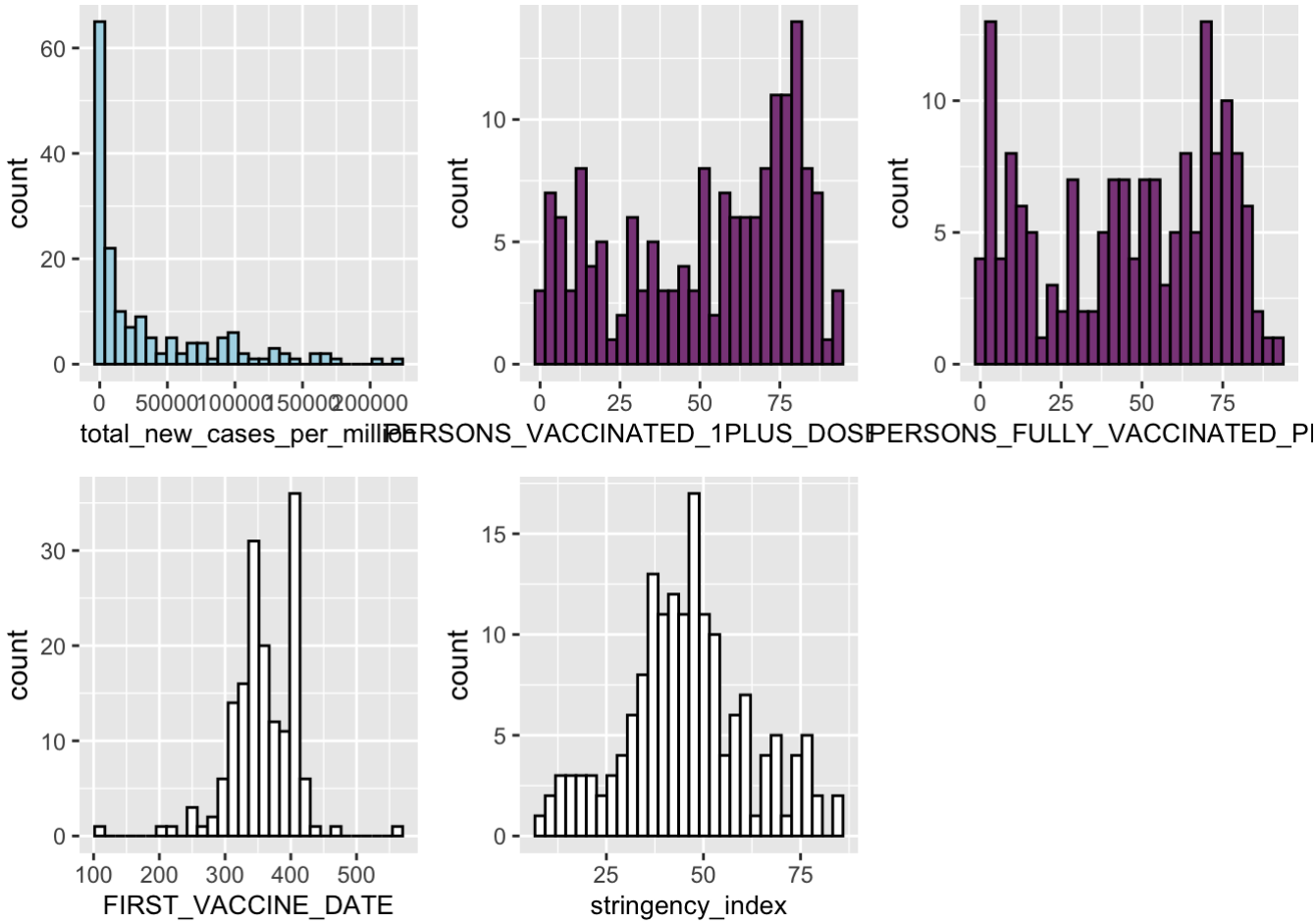
3.1 Descriptive Analysis

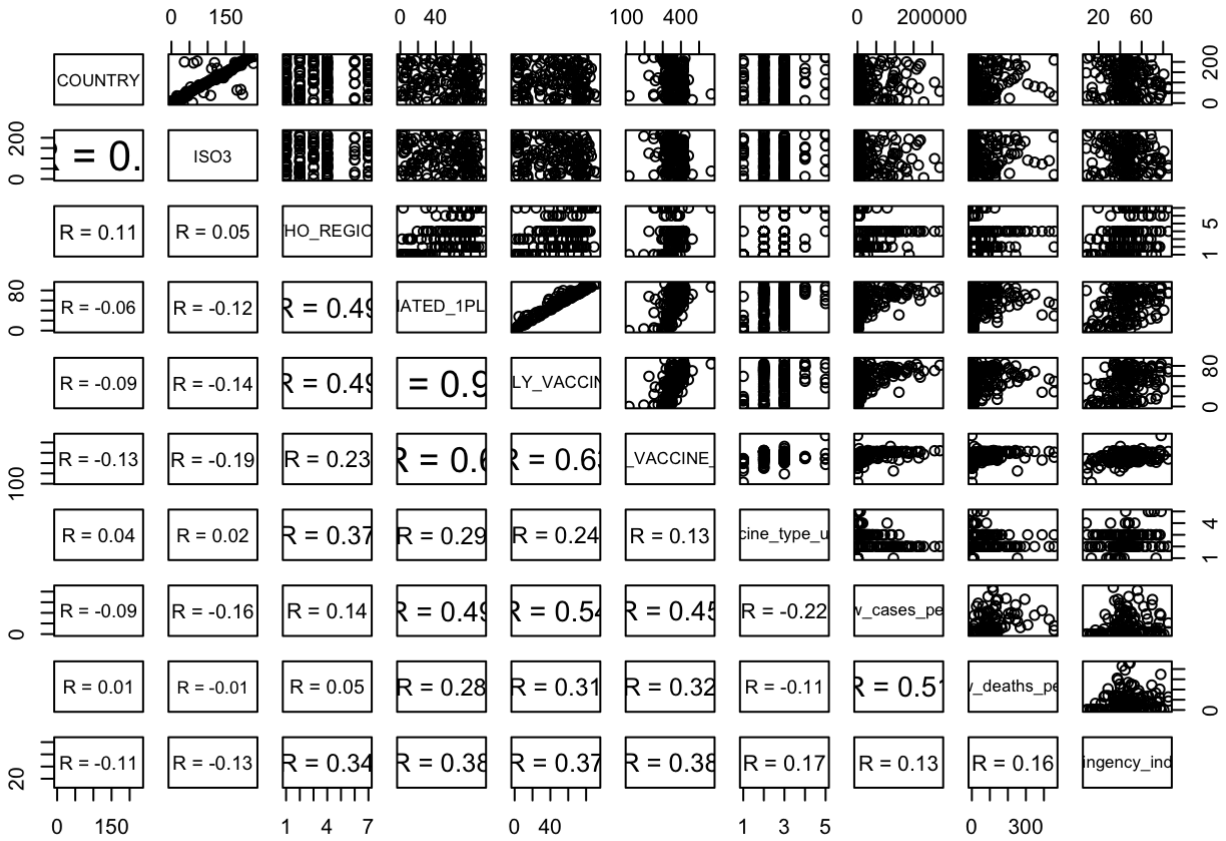
Before proceeding the modeling and predictive analysis, we examined our data by several figures.
Some of the countries has relatively high fully vaccinated rate, in which 92% of the population had already finished two doses. The average rate of it is 46%, meaning that almost half of the world are fully vaccinated, it might have a positive impact on the recent positive cases, which is exactly what we're going to examine later. Moreover, average amount of date from now that a country started vaccination is 359 days, almost a year. And most of the country implemented at least two types of vaccine.

```
##         COUNTRY          ISO3     WHO_REGION
##  Afghanistan:  1   ABW    :  1   AFRO :41
##  Albania    :  1   AFG    :  1   AMRO :30
##  Algeria    :  1   AGO    :  1   EMRO :19
##  Andorra    :  1   ALB    :  1   EURO :46
##  Angola     :  1   AND    :  1   OTHER: 0
##  Argentina  :  1   ARG    :  1   SEARO: 9
##  (Other)    :158   (Other):158   WPRO :19
##  PERSONS_VACCINATED_1PLUS_DOSE_PER100 PERSONS_FULLY_VACCINATED_PER100
##  Min.   : 0.067                       Min.   : 0.063
##  1st Qu.:28.768                       1st Qu.:20.715
##  Median :58.828                       Median :49.624
##  Mean   :51.874                       Mean   :45.901
##  3rd Qu.:75.788                       3rd Qu.:70.552
##  Max.   :93.181                       Max.   :92.207
##
##  FIRST_VACCINE_DATE vaccine_type_used total_new_cases_per_million
##  Min.   :114.0      1:12              Min.   :     0.0
##  1st Qu.:331.8      2:72              1st Qu.:   806.8
##  Median :356.5      3:68              Median :  9712.5
##  Mean   :359.1      4: 7              Mean   : 34721.2
##  3rd Qu.:406.0      5: 5              3rd Qu.: 52653.8
##  Max.   :567.0                        Max.   :220250.0
##
##  total_new_deaths_per_million stringency_index
##  Min.   :  0.00               Min.   : 8.33
##  1st Qu.:  4.00               1st Qu.:37.18
##  Median : 28.00               Median :45.30
##  Mean   : 72.55               Mean   :45.87
##  3rd Qu.:115.25               3rd Qu.:53.92
##  Max.   :457.00               Max.   :85.00
##
```

Based on the below histogram and boxplots, we could tell that substantial variation appears on **total_new_cases_per_million** and the distribution is right-skewed. This indicates that in the following analysis, transformation and the dispersed of the variance might need to be considered. In addition, **PERSONS_VACCINATED_1PLUS_DOSE_PER100** and **PERSONS_FULLY_VACCINATED_PER100** have really similar pattern in the shape of the distribution, and be highly correlated as well(correlation equals to 0.98). One of the reason of this situation might because of the overlapping calculation on person who got vaccinated. They might lead to multicollinearity in the following analysis, we decided to not consider **PERSONS_VACCINATED_1PLUS_DOSE_PER100** in the model. There is no obvious relationship between **total_new_cases_per_million** and **stringency_index** based on following graph. But there is a slightly positive linear relationship with between **total_new_cases_per_million** and **PERSONS_FULLY_VACCINATED_PER100**. We will discuss it in the later paragraph.

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

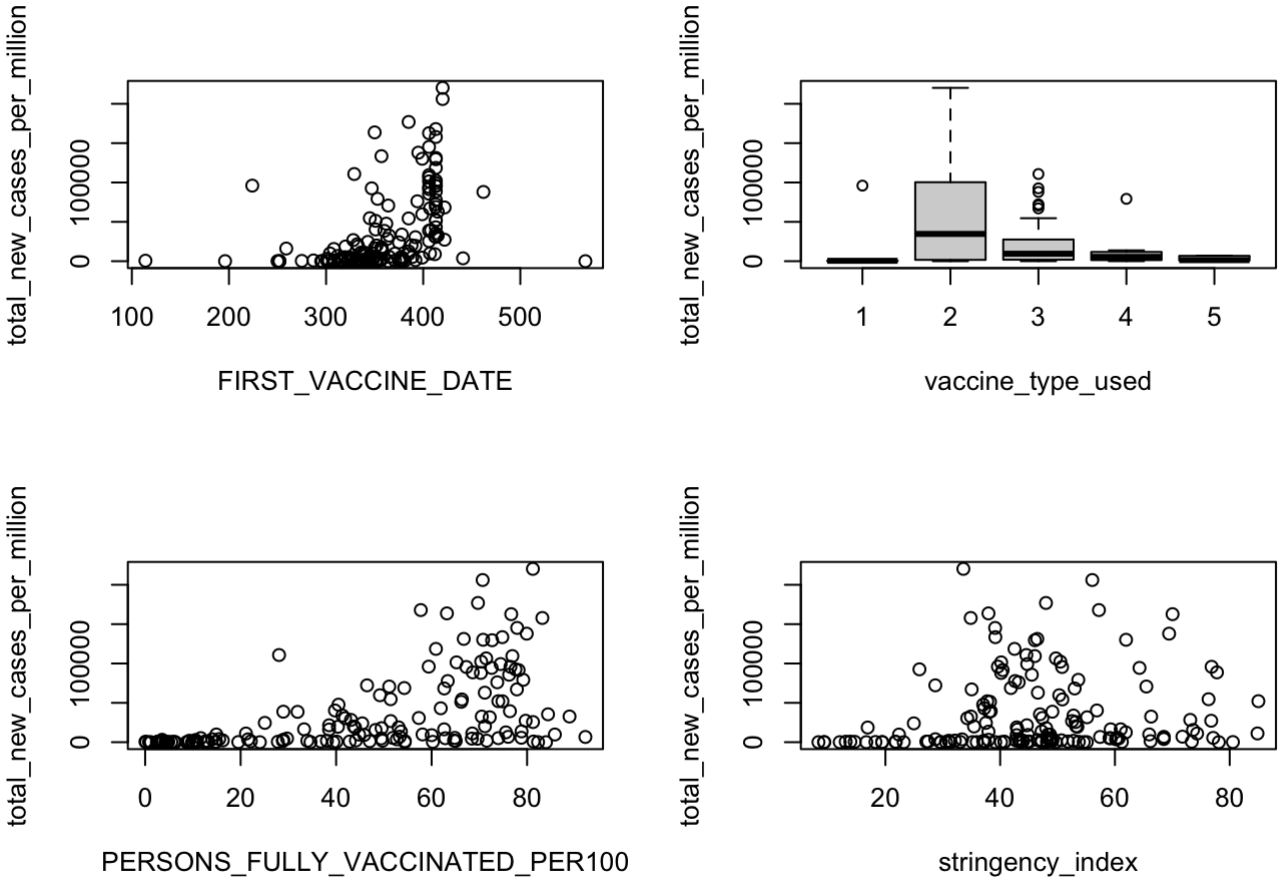

count
total_new_cases_per_mill
PERSONS_VACCINATED_1PLUS_DOSE
PERSONS_FULLY_VACCINATED_PE
FIRST_VACCINE_DATE
stringency_index


COUNTRY
ISO3
R = 0.11
R = 0.05
R = -0.06
R = -0.12
R = -0.09
R = -0.14
R = -0.13
R = -0.19
R = 0.04
R = 0.02
R = -0.09
R = -0.16
R = 0.01
R = -0.01
R = -0.11
R = -0.13
R = 0.23
R = 0.37
R = 0.14
R = 0.05
R = 0.34
R = 0.29
R = 0.24
R = 0.13
R = -0.22
R = -0.11
R = 0.17
R = 0.13
R = 0.16

Besides, the table below shows the average total new cases in the past 38 days by the number of vaccine type used in a country. It suggests that over-dispersion is present, because the conditional variances are far larger than the conditional mean. Theses difference indicates that with the count data as our response variable, instead of using poisson regression model, negative binomial regression model might be more appropriate.

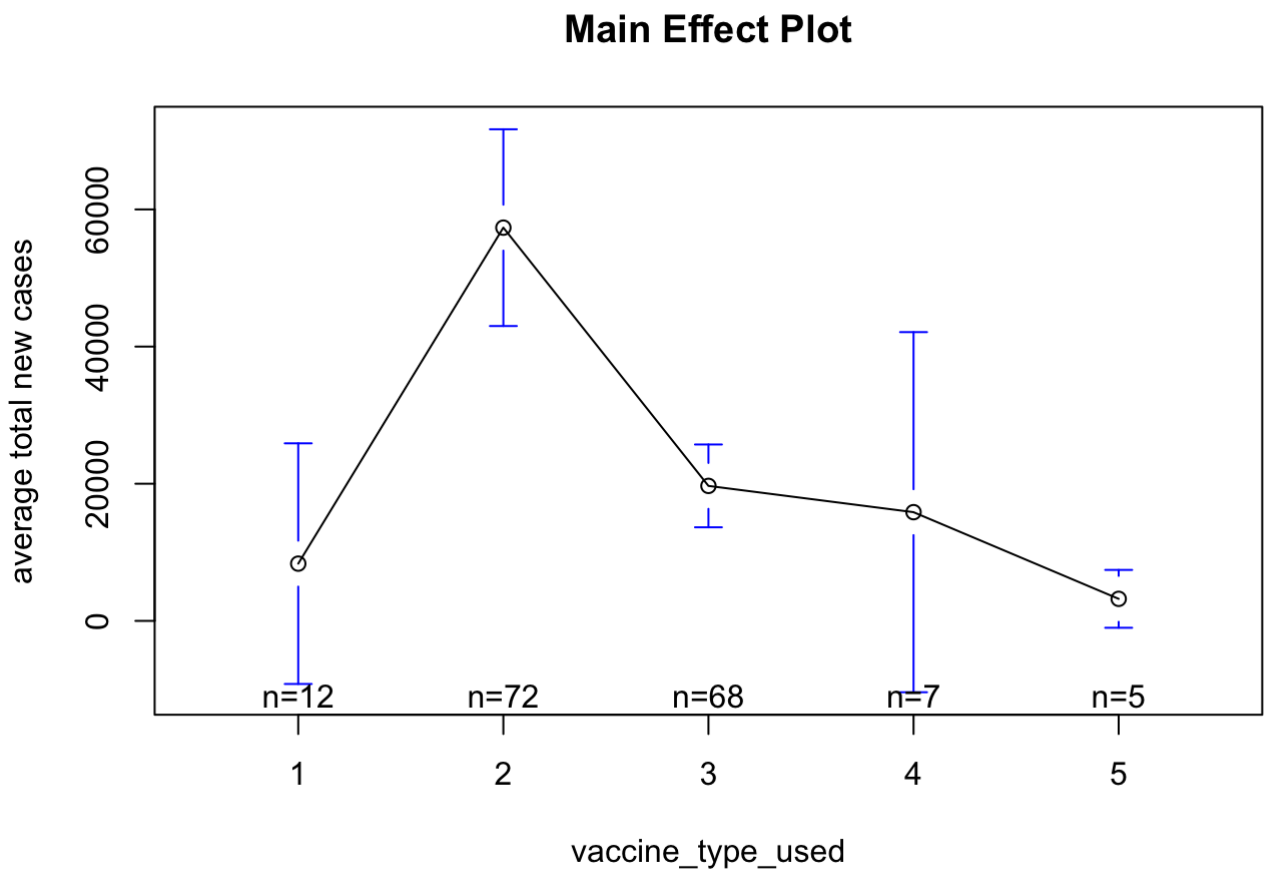
```
##                                1                                2
##  "M (SD) = 8346.83 (27613.23)" "M (SD) = 57340.75 (61039.26)"
##                                3                                4
## "M (SD) = 19684.49 (24945.15)" "M (SD) = 15853.29 (28389.81)"
##                                5
##   "M (SD) = 3214.20 (3397.17)"
```

Besides the graphs above, we would like to understand more relationship about our response variable and explanatory variables. The following three graphs are for this purpose.
To begin with, it shows only 2 type of vaccine being used by a country are more common and have larger variance. Even though we have less information of the other number of vaccine type used, it seems to have a negative relationship of the **vaccine_type_used** and **total_new_cases_per_million**. As for the relationship between total cases and the **first vaccination start date** and the relationship between **stringency index** are not obvious. Thus, we further created main effect plot and histogram. The main effect plot proves that there exist different effect by different number of vaccine has used.



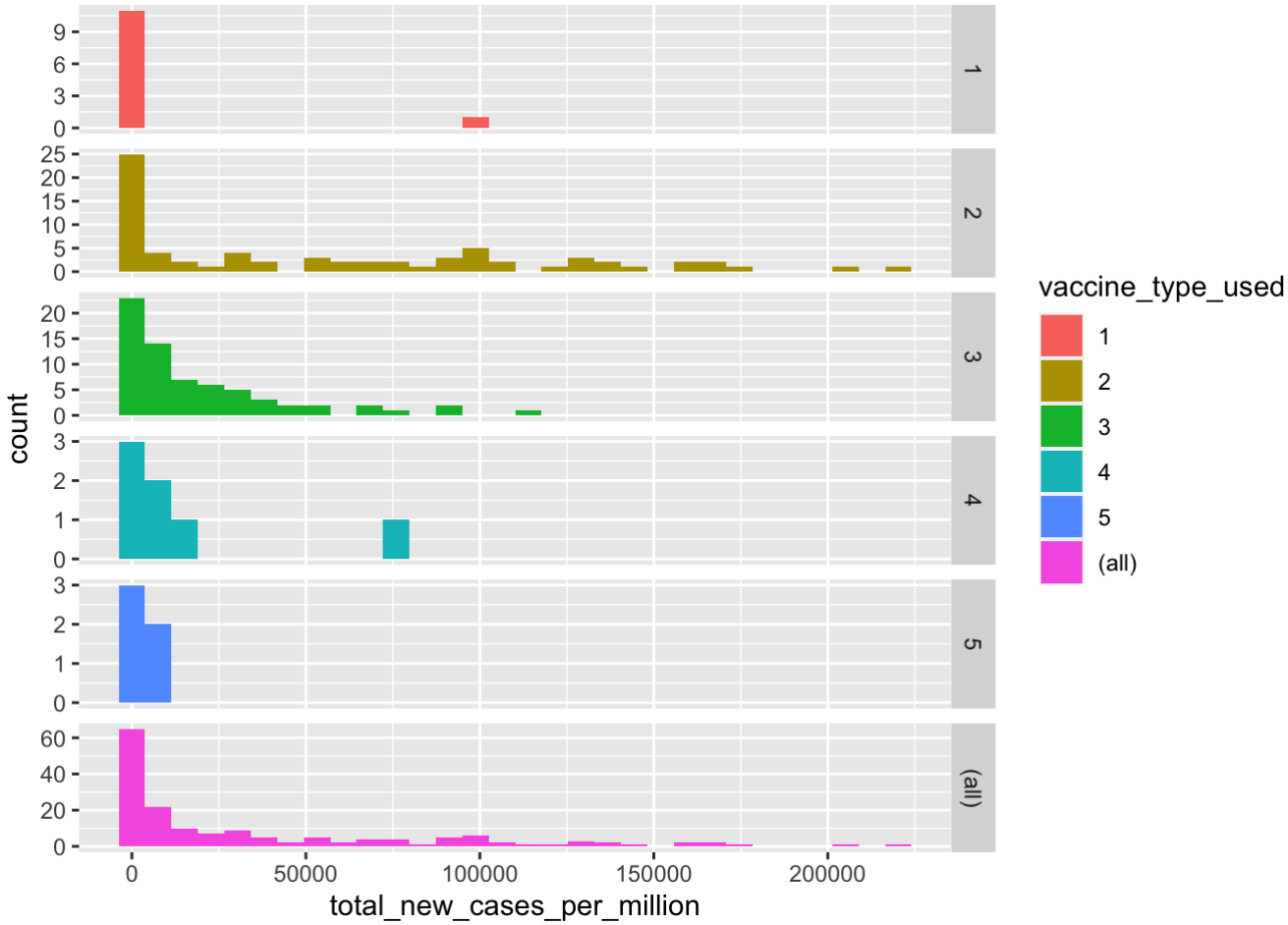

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```




```
#ggplot(covid_trim, aes(total_new_cases_per_million, fill = vaccine_type_used)) +
  #geom_histogram(position="dodge")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



3.2 Inferential Analysis

In this section, we proposed to use three approaches to conduct inferential analysis. Since our response variable is about count data, the first two approaches, we chose (1) Poisson regression, (2) Negative binomial regression. There are still several statistical models for count data, such as zero-inflated regression, quassi-poisson regression. The reason we did not choose them is because, there is only one zero value in our response value and according to the above analysis, we acquired that there is overdispersion in the data, and not underdispersion. The approach we chose (3) multiple linear regression, which is an more easily understanding and interpreting model and is suitable for both qualitative and quantitative independent variables.

3.2.1 Poisson Regression

Our first attempt for model fitting is Poisson Regression. It is an approach especially for count data with non-negative integers which could fix the problem that the fitted line in an ordinary least square model could yield negative values. Poisson regression model is a generalized linear model form of regression which is also known as log-linear model. That is, the response variable is usually distributed by the poisson distribution, and it use a logarithmic function as the link between the response variable and regressors.

The probability mass function of poisson distribution is

$$f(k,\lambda) = P(X = k) = \frac{\lambda^k \exp(-\lambda)}{k!}$$

,where $k = 0, 1, 2, \ldots$ is the total number of occurrences(events) and λ is the average rate of occurrence for the event being measured ($\lambda = k/n$), in addition the mean and variance of the random variable X are both equal to λ. ($E(X) = Var(X) = \lambda$)

The equation of poisson regression is

$$ln(Y) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_k X_k$$

, where Y is the response variable, X_i are regressors, β_i are regression coefficients and k is the number of regressors.
Often, it can also be written as

$$Y = \exp(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_k X_k)$$

after taking exponential.

Poisson regression assumes that (1) the response variables is poisson distributed, (2) the log of its expected value can be explained by a linear combination of covariates, (3) the variance is equal to the mean, and (4) observations must be independent to each other.

```
## Dispersion test of count data:
## 164 data points.
## Mean: 34721.25
## Variance: 2377350787
## Probability of being drawn from Poisson distribution: 0
```

Here, we can see that the variance are far larger than the mean, which violates the assumption. But in order to obtain more detail, we still fitted the model.
To capture the relationship between the number of total cases from 1/1/22 to 2/10/22 and explanatory variables, we first fitted an additive poisson regression model with 4 explanatory variables: **FIRST_VACCINE_DATE**, **vaccine_type_used**, **PERSONS_FULLY_VACCINATED_PER100**, and **stringency_index**.

```
##
## Call:
## glm(formula = total_new_cases_per_million ~ FIRST_VACCINE_DATE +
##     vaccine_type_used + PERSONS_FULLY_VACCINATED_PER100 + stringency_index,
##     family = "poisson", data = covid_trim)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -376.09  -124.09   -68.75    51.20   643.75
##
## Coefficients:
##                                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)                    6.931e+00  5.419e-03 1278.94   <2e-16 ***
## FIRST_VACCINE_DATE             6.079e-03  1.532e-05  396.71   <2e-16 ***
## vaccine_type_used2            -6.026e-02  3.540e-03  -17.02   <2e-16 ***
## vaccine_type_used3            -8.083e-01  3.510e-03 -230.30   <2e-16 ***
## vaccine_type_used4            -1.574e+00  4.577e-03 -343.93   <2e-16 ***
## vaccine_type_used5            -3.405e+00  8.955e-03 -380.21   <2e-16 ***
## PERSONS_FULLY_VACCINATED_PER100  3.276e-02  2.902e-05 1128.67   <2e-16 ***
## stringency_index              -5.300e-03  3.094e-05 -171.31   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 9632586  on 163  degrees of freedom
## Residual deviance: 4079729  on 156  degrees of freedom
## AIC: 4081471
##
## Number of Fisher Scoring iterations: 6
```

The results shows that every explanatory variables are statistically significant. However, when we examine the assumption of this model, the issue of overdispersion occurs which leads to poor hold of assumptions. The overdispersion is examined by `dispersiontest()` sysntax from **AER** packages with null hypothesis
H_0 : true dispersion is equal to 1 and alternative hypothesis
H_a : true dispersion is greater than 1
We could also intuitively examine it by comparing the conditional mean and variance. Take *vaccine_type_used* for example, the variance of only one type used is 27613, almost three times larger than the mean, and the other variance also presents the issue of over-dispersed variance.

3.2.2 Negative Binomial Regression

Overdispersion appears in our data might because of the underlying clustering in the sample. To put it another way, each observation represent to one country in our covid data, however, countries in one continent might have similar pattern or clustering in the data.

Due to the overdispersion in the poisson regression which will cause to underestimated standard error and inflated type I error, negative binomial regression seemed to be a better approach here.

Negative binomial regression is a generalization of poisson regression whose variance is assumed to be $Var(Y) = \mu(1 + \frac{\mu}{r})$, where k is the shape parameter.

The probability mass function of negative binomial is

$$f(k, r, p) = P(X = k) = \binom{k + r - 1}{r - 1}(1 - p)^k p^r$$

, where there are $k + r - 1$ samples, r successes, k failures and p is the probability of success.

It can also be written as

$$P(X = k) = \frac{\Gamma(r + k)}{k!\Gamma(r)}(\frac{r}{r + \mu})^r(\frac{\mu}{r + \mu})^k$$

,where $k = 0, 1, 2, \ldots$ Because its mean is $\mu = \frac{pr}{1-p}$, and then we can derive it to get $p = \frac{r}{\mu+r}$.

The equation for binomial regression is as same as poisson distribution.

$$ln(Y) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_k X_k$$

```
##
## Call:
## glm.nb(formula = total_new_cases_per_million ~ FIRST_VACCINE_DATE +
##     vaccine_type_used + PERSONS_FULLY_VACCINATED_PER100 + stringency_index,
##     data = covid_trim, init.theta = 0.6328498906, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.3095  -1.1649  -0.3695   0.1802   3.8002
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     5.720e+00  8.207e-01   6.969 3.19e-12 ***
## FIRST_VACCINE_DATE             -3.595e-05  2.639e-03  -0.014   0.9891
## vaccine_type_used2              8.179e-01  4.329e-01   1.889   0.0588 .
## vaccine_type_used3              1.011e+00  4.216e-01   2.399   0.0165 *
## vaccine_type_used4             -1.258e+00  6.484e-01  -1.940   0.0524 .
## vaccine_type_used5             -1.756e+00  7.113e-01  -2.469   0.0136 *
## PERSONS_FULLY_VACCINATED_PER100  6.074e-02  5.023e-03  12.094  < 2e-16 ***
## stringency_index                 1.068e-02  6.891e-03   1.550   0.1210
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.6328) family taken to be 1)
##
##     Null deviance: 366.84  on 163  degrees of freedom
## Residual deviance: 201.01  on 156  degrees of freedom
## AIC: 3486.2
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.6328
##           Std. Err.:  0.0594
##
##  2 x log-likelihood:  -3468.1790
```

First,we created a preliminary model, in which we put 4 explanatory variables we're interested in. In this model, the quantitative variable **FIRST_VACCINE_DATE** and **stringency_index** seem to not be significant, while **PERSONS_FULLY_VACCINATED_PER100** has a coefficient 0.0607, which is showed statistically significant base on the result.

The coefficient of this variable means that with one unit increase in **PERSONS_FULLY_VACCINATED_PER100**, there will be an 0.0607 increase for the expected log count of total new covid cases. This might seem to conflict to the common sense that the more people got fully vaccinated, the less positive cases would happen. However, in the previous EDA section, it indicated from the scatter plot of **PERSONS_FULLY_VACCINATED_PER100** and **total_new_cases_per_million** that they don't have a negative linear relationship but have a slightly positive relationship.

As to the **vaccine_type_used**, those indicator variables show the difference in log count of total new covid cases between that group and the reference group (only use one type). Here, **vaccine_type_used2** is 0.8179 more than the log count for **vaccine_type_use** = 1, **vaccine_type_used3** is 1.0113 more than that, and

vaccine_type_used4 and **vaccine_type_used5** is 1.2577 and 1.7561 less than the log count for reference group respectively.

In order to determine if the **vaccine_type_used** is overall statistically significant, and if we should leave out **FIRST_VACCINE_DATE** and **stringency_index**, we created other models without these targeting variables and compared them.

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: total_new_cases_per_million
##
Model
## 1                                          vaccine_type_used + PERSONS_FULLY_VACCINATE
D_PER100
## 2 FIRST_VACCINE_DATE + vaccine_type_used + PERSONS_FULLY_VACCINATED_PER100 + stringen
cy_index
##       theta Resid. df    2 x log-lik.   Test    df LR stat.   Pr(Chi)
## 1 0.6283444       158       -3469.805
## 2 0.6328499       156       -3468.179 1 vs 2     2 1.625727 0.4435861
```

We conducted deviance test

$$-2\ell(\hat{\beta}^{(0)}) - (-2\ell(\hat{\beta}))$$

, where $-2\ell(\hat{\beta}^{(0)})$ is the deviance of the reduced model and $-2\ell(\hat{\beta})$ is the deviance of the full model).
Our null hypotheses is
$H_0 : \beta_{FIRST.VACCINE.DATE} = \beta_{stringency.index} = 0$ and H_1 :not all of them are equal to 0.
Also, the test statistic has a chi-squared distribution with $k + 1 - r$ degree of freedom.

The result shows a failure to reject null hypothesis. It means that comparing to the full model(**model_nb0**), **FIRST_VACCINE_DATE** and **stringency_index** are not statistically significant, thus they are able to be removed.

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: total_new_cases_per_million
##                                                  Model     theta Resid. df
## 1                     PERSONS_FULLY_VACCINATED_PER100 0.5652558       162
## 2 vaccine_type_used + PERSONS_FULLY_VACCINATED_PER100 0.6283444       158
##      2 x log-lik.   Test    df LR stat.      Pr(Chi)
## 1       -3494.144
## 2       -3469.805 1 vs 2     4 24.33961 6.828109e-05
```

After removing **FIRST_VACCINE_DATE** and **stringency_index**, we compared this new model with the one without **vaccine_type_used**. We conducted the deviance test again, with null hypotheses
H_0 : all $\beta_{vaccine.type.used} = 0$ and H_1 : not all $\beta_{vaccine.type.used} = 0$.
And the 4 degrees of freedom chi-squared test indicates that **vaccine_type_used** is statistically significant.

Thus, our final negative binomial regression model equation is:

$$\widehat{ln(total.new.cases.per.million_i)} = \hat{\beta}_0 + \hat{\beta}_1 I(vaccine.type.used_i = 2) + \hat{\beta}_2 I(vaccine.type.used_i = 3) + \hat{\beta}_3 I(vaccine.type.used_i = 4) + \hat{\beta}_4 I(vaccine.type.used_i = 5) + \hat{\beta}_5 X_{PERSONS.FULLY.VACCINATED.PER100_i}$$

Without log scale, the equivalent model equation is:

$$\widehat{total.new.cases.per.million_i} = exp(\hat{\beta}_0 + \hat{\beta}_1 I(vaccine.type.used_i = 2) + \hat{\beta}_2 I(vaccine.type.used_i = 3) + \hat{\beta}_3 I(vaccine.type.used_i = 4) + \hat{\beta}_4 I(vaccine.type.used_i = 5) + \hat{\beta}_5 X_{PERSONS.FULLY.VACCINATED.PER100_i})$$

```
## MODEL INFO:
## Observations: 164
## Dependent Variable: total_new_cases_per_million
## Type: Generalized linear model
##   Family: Negative Binomial(0.6283)
##   Link function: log
##
## MODEL FIT:
## χ²() = , p =
## Pseudo-R² (Cragg-Uhler) = 0.53
## Pseudo-R² (McFadden) = 0.03
## AIC = 3483.80, BIC = 3505.50
##
## Standard errors: MLE
## ------------------------------------------------------------------------------
##                                    exp(Est.)     2.5%    97.5%   z val.       p
## ---------------------------------- ----------- -------- -------- -------- ------
## (Intercept)                           429.61   209.27   881.93    16.52   0.00
## vaccine_type_used2                      2.47     1.08     5.66     2.13   0.03
## vaccine_type_used3                      3.02     1.33     6.84     2.65   0.01
## vaccine_type_used4                      0.31     0.09     1.10    -1.81   0.07
## vaccine_type_used5                      0.23     0.06     0.90    -2.12   0.03
## PERSONS_FULLY_VACCINATED_PER100         1.06     1.06     1.07    15.40   0.00
## ------------------------------------------------------------------------------
```

From the result, it shows that with one exponential unit increase in **PERSONS_FULLY_VACCINATED_PER100**, there would be 1.06 increase in **total_cases_per_million**. And exp(**vaccine_type_used2**) is 2.47 more than the count for **vaccine_type_use** = 1, exp(**vaccine_type_used3**) is 3.02 more than it, and exp(**vaccine_type_used4**) and exp(**vaccine_type_used5**) is 0.31 and 0.23 less than the count for reference group respectively. In addition, the Pseudo-R^2 (Cragg-Uhler) = 0.53 and AIC = 3483.80.

3.2.3 Measure the goodness-of-fit

We compared the goodness-of-fit of two models by likelihood ratio test in order to understand whether negative binomial model we created here is better than the poisson regression model. The two models are both with the same variables from the same dataset.

The log-likelihood value of negative binomial model is -1734.902 and that of poisson regression model is -2136764, thus two times of their difference is 4270034, far larger than the critical value $\chi^2_{(1)} = 3.841459$ under significance level 0.05. Thus, it implies that negative binomial model has a better goodness-of-fit than the poisson model.
However, when we test the goodness-of-fit of negative binomial itself, it shows an evidence of lack-of-fit, which means we still have a lot to improve on the features of the model in the future.

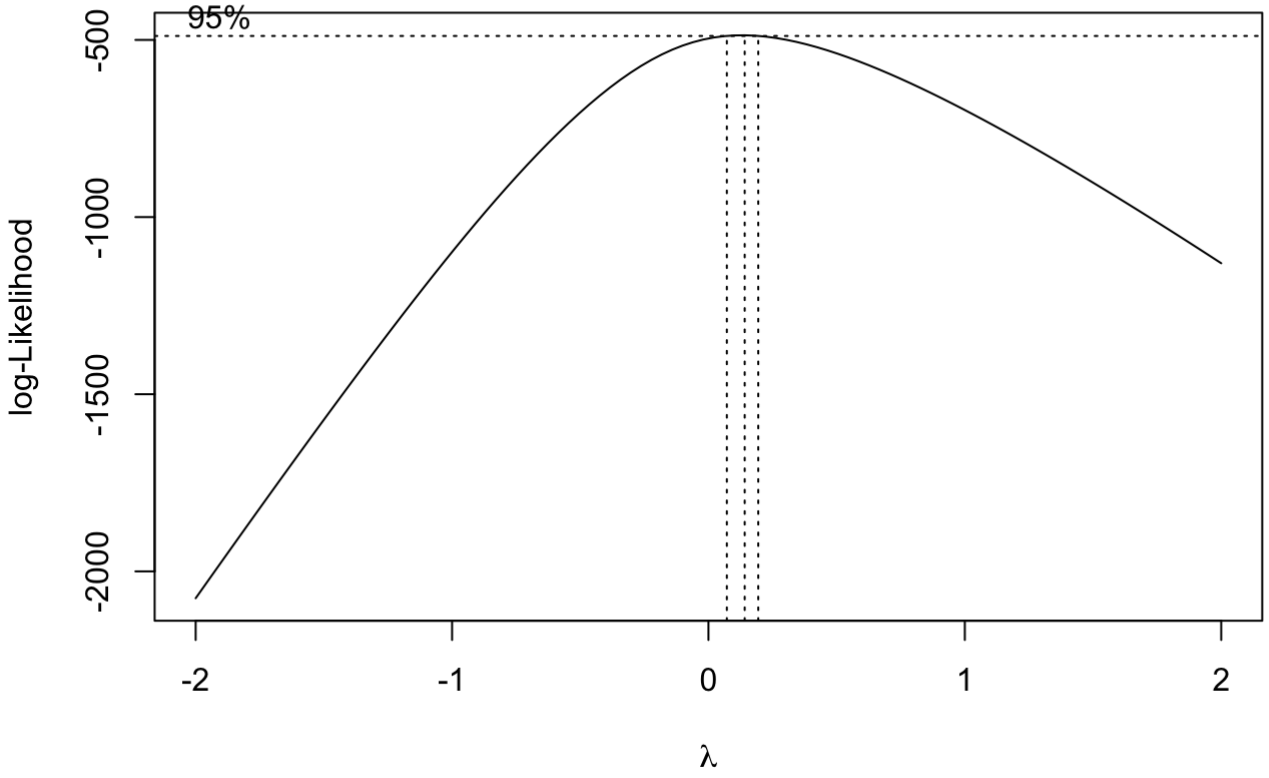
```
##             df         AIC
## model_pois1  6 4273540.235
## model_nb2    3    3500.144
```

3.3.4 Multiple Linear Regression

Our third attempt is to use multiple linear regression to explore the relationship between **total_new_cases_per_million** and the targeting 4 variables. The equation of multiple linear regression model is

$$Y = \beta_0 + \beta_1 X1 + \ldots + \beta_k Xk + \epsilon$$

, where Y is response variable, β is regression coefficients, X is independent variable and k is the number of independent variables, ϵ is the error term.
In the previous descriptive analysis section, the distribution of the response variable **total_new_cases_per_million** looks right-skewed, therefore, we performed box-cox transformation to see if any transformation on Y is necessary. Based on the result, performing log transformation on Y would be appropriate. Then the equation would turn to be $ln(Y) = \beta_0 + \beta_1 X1 + \ldots + \beta_k Xk + \epsilon$
Furthermore, based on the main plot of **vaccine_type_used**, one-way anova is also conducted to check whether the effect exist in different groups.



The one-way ANOVA model we used is:

$$Y_{ij} = \mu + \alpha_i + \epsilon_{ij}, i = 1, .., 5, \ j = 1, \ldots, n_i$$

,where Y_{ij} is observation of total new cases from different number of vaccine type a country has used, μ is overall average total new cases, α_i the effect on response variable by the number of vaccine type and ϵ_{ij} is i.i.d to $N(0, \sigma^2)$.
The assumption of normally distributed and constant variance of error term should be hold here.

And based on the ANOVA table, F test can be conducted to test whether there is any effect exist by different number of vaccine type a country has used.
It is conducted as following:
Null hypothesis: $H_0 : \alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5$
Alternative hypothesis: $H_1 : \textit{not all } \alpha_i \textit{ are the same}.$

- critical value: $F(1 - 0.05, 5 - 1, 164 - 5) = F(0.95, 4, 159) = 2.428522$
- test statistic: $F^* = MSTR/MSE = 5.722$
- rejection region: the set of $(2.428522, +\infty)$

Because the test statistics is in the rejection region, we can reject the null under significance level 0.05.

```
##                     Df Sum Sq Mean Sq F value  Pr(>F)
## vaccine_type_used   4  130.1   32.52   5.722 0.00025 ***
## Residuals         159  903.8    5.68
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The result of the first additive model shows that **FIRST_VACCINE_DATE** and **stringency_index** are not statistically significant, and with the F test, the results are proved. Furthermore, the estimate shows that with one unit increase by **PERSONS_FULLY_VACCINATED_PER100**, there would be 0.055064 increase on the logarithmic value of **total_new_cases_per_million**. Which is similar to the result of negative binomial regression model.

```
##
## Call:
## lm(formula = log(total_new_cases_per_million + 1) ~ FIRST_VACCINE_DATE +
##     vaccine_type_used + PERSONS_FULLY_VACCINATED_PER100 + stringency_index,
##     data = covid_trim)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.9925 -0.8901  0.3317  1.1405  3.9470
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    3.624890   1.153327   3.143  0.00200 **
## FIRST_VACCINE_DATE             0.004117   0.003709   1.110  0.26867
## vaccine_type_used2             0.604874   0.608261   0.994  0.32155
## vaccine_type_used3             0.356223   0.592436   0.601  0.54852
## vaccine_type_used4            -1.614714   0.911253  -1.772  0.07835 .
## vaccine_type_used5            -2.892044   0.999685  -2.893  0.00436 **
## PERSONS_FULLY_VACCINATED_PER100  0.055064   0.007060   7.800 8.29e-13 ***
## stringency_index               0.017691   0.009685   1.827  0.06968 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.767 on 156 degrees of freedom
## Multiple R-squared:  0.5289, Adjusted R-squared:  0.5077
## F-statistic: 25.02 on 7 and 156 DF,  p-value: < 2.2e-16
```

```
## Analysis of Variance Table
##
## Model 1: log(total_new_cases_per_million + 1) ~ vaccine_type_used + PERSONS_FULLY_VAC
CINATED_PER100
## Model 2: log(total_new_cases_per_million + 1) ~ FIRST_VACCINE_DATE + vaccine_type_use
d +
##     PERSONS_FULLY_VACCINATED_PER100 + stringency_index
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1    158 504.36
## 2    156 487.07  2    17.287 2.7683 0.06585 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

F test is again used to conduct the hypothesis testing to test whether **vaccine_type_used** should be leave in the model.
Null hypothesis: $H_0 : \beta_{vaccine.type.used} = 0$
Alternative hypothesis: $H_1 : \beta_{vaccine\ type\ used} \neq 0$
We rejected H0 under the significance $\alpha = 0.05$.
It means that **vaccine_type_used** can be keep in the model.
Thus, our final model is the one with two independent variables: **vaccine_type_used** and **PERSONS_FULLY_VACCINATED_PER100**.

The final fitted model is then:

$$log(Y) = 5.4009 + 0.7402 * I(vaccine\ type\ used = 2) +$$
$$0.4875 * I(vaccine\ type\ used = 3) - 1.8320 * I(vaccine\ type\ used = 4)$$
$$- 2.4248 * I(vaccine\ type\ used = 5) + 0.0637 * X_{PERSONS\ FULLY\ VACCINATED\ PER100}$$

```
## Analysis of Variance Table
##
## Model 1: log(total_new_cases_per_million + 1) ~ PERSONS_FULLY_VACCINATED_PER100
## Model 2: log(total_new_cases_per_million + 1) ~ vaccine_type_used + PERSONS_FULLY_VAC
CINATED_PER100
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    162 588.93
## 2    158 504.36  4    84.572 6.6235 5.927e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

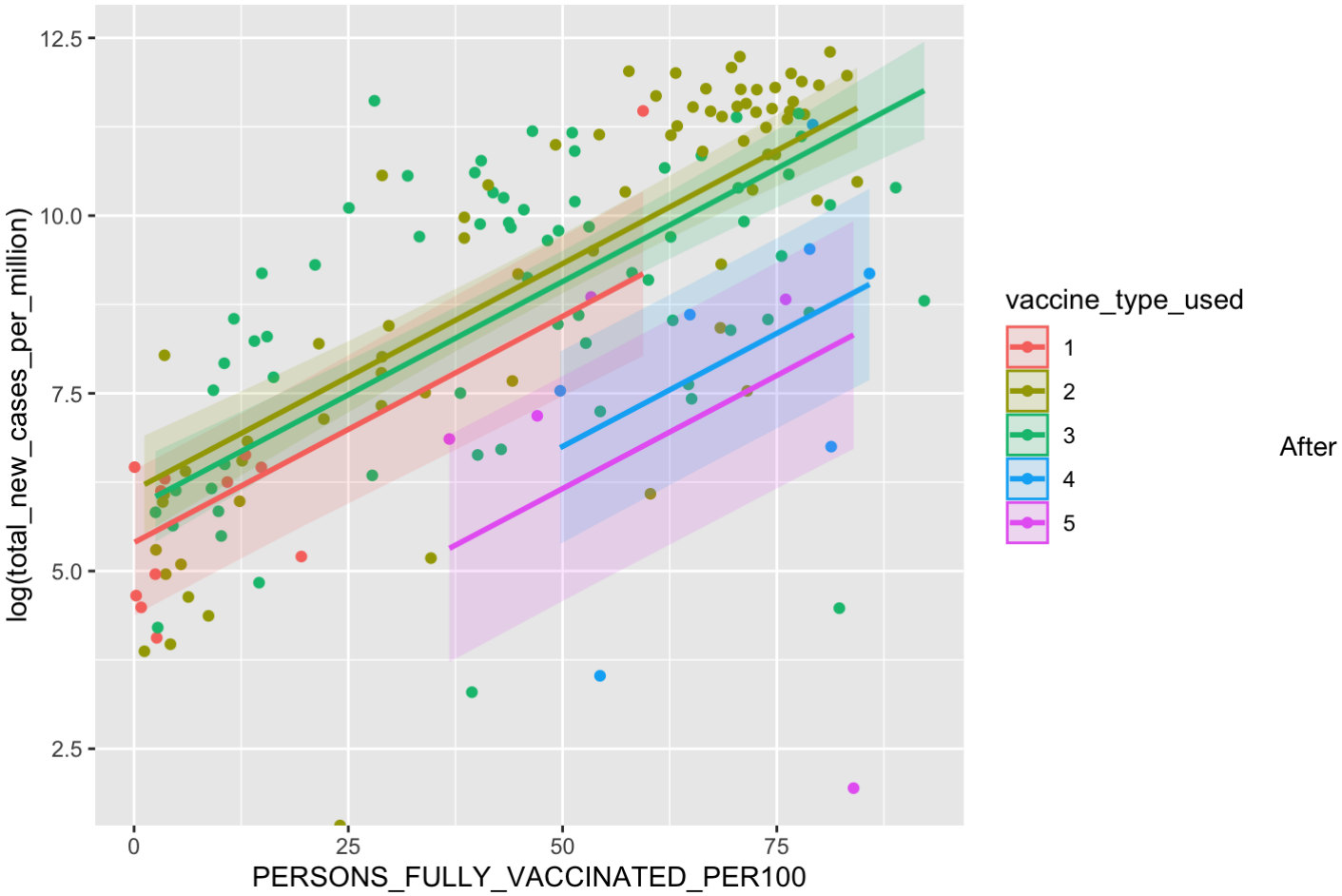
3.3.5 Model Selection

To select a better model in our three approaches, we chose to use $AIC = nlog(\frac{SSE}{n}) + 2k$ as our comparing method. The reason that I only chose AIC because none of the models I selected from are too complex, with a maximum of three variables. It is in the sense that choosing BIC which penalizes the model more for its complexity is not necessary in this case, on the other hand, more consideration should be given to the model performance. The result shows that the multiple linear regression has the smallest AIC compare to the other models.Thus, we chose to use the multiple regression model to perform the following predictive analysis.

```
##   Poisson Negative_Binomial      OLS
## 1 4273540          3483.805 663.6531
```

3.3.6 Predictive Analysis

```
ggplot(covid_trim_predict, aes(x = PERSONS_FULLY_VACCINATED_PER100, y = log(total_new_ca
ses_per_million), color = vaccine_type_used) ) +
     geom_point() +
     geom_ribbon(aes(ymin = lwr, ymax = upr, fill = vaccine_type_used, color = NULL), al
pha = .15) +
     geom_line(aes(y = fitted), size = 1)
```

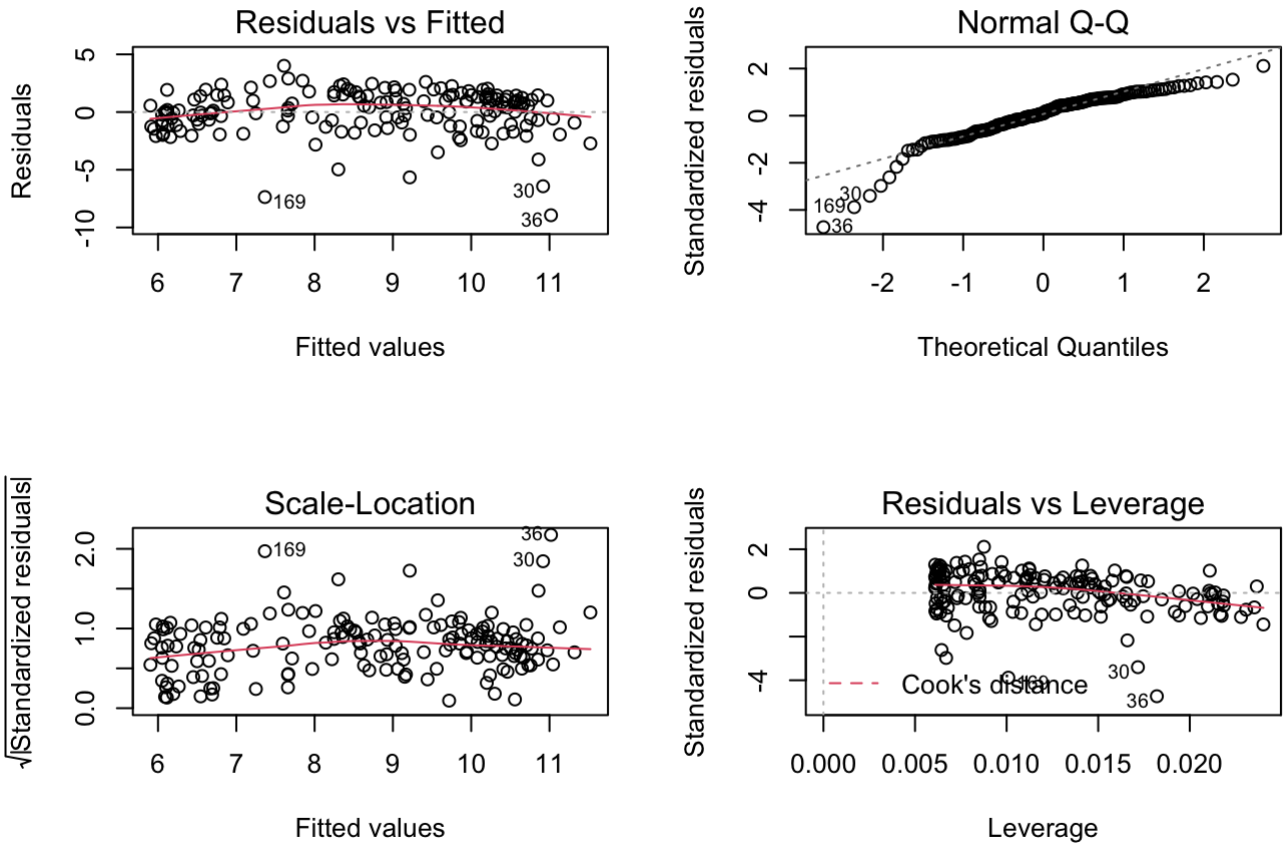


fitting the model, we can see that with different amount of vaccine type used, total cases performed differently. That is, two and three vaccine types used shows more total cases, and four and five vaccine types used show less cases. More the overall trend are the same, with positive relationship between fully vaccinated rate and total cases.

3.3.7 Sensitivity Analysis

3.3.7.1 Model Diagnostic

The residual and fitted plot shows that there is no obvious pattern for the residual. It indicates that the linearity and constant variance of residual are hold well for our model. However, in normal Q-Q plot, it is seen that more probability mass on the left tail and a little bit less probability mass on the right tail compares to a normal distribution. But it could because of the several countries (ex.China) has less total cases than most of the countries and are treated as outliers.



3.3.7.1 Other

In inferential analysis section, the multiple regression model with log transformation perform better than the poisson and negative binomial. I would like to see what will happen if we do not consider the log transformation on response variable. Based on the result, the R^2_{Adj} decreases from 0.4967 to 0.429, and the AIC of this model becomes 3922.026, which is similar to the one of negative binomial regression but less goodness-of-fit. However, the interpretation of this model could be easier than the one with logrithmic. For example, it shows that the **vaccine_type_use** = 4 has 54141 total new cases less than the reference **vaccine_type_use** = 1.

```
##
## Call:
## lm(formula = total_new_cases_per_million ~ vaccine_type_used +
##     PERSONS_FULLY_VACCINATED_PER100, data = covid_trim)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -77930 -24029  -2143  16694 130468
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    -2890.0    10712.0  -0.270  0.78767
## vaccine_type_used2              8820.3    12362.3   0.713  0.47660
## vaccine_type_used3            -23293.8    12189.2  -1.911  0.05781 .
## vaccine_type_used4            -54141.1    18871.7  -2.869  0.00468 **
## vaccine_type_used5            -55256.5    20422.1  -2.706  0.00756 **
## PERSONS_FULLY_VACCINATED_PER100   1032.5      117.4   8.796 2.31e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36840 on 158 degrees of freedom
## Multiple R-squared:  0.4466, Adjusted R-squared:  0.429
## F-statistic:  25.5 on 5 and 158 DF,  p-value: < 2.2e-16
```

```
##   Poisson Negative_Binomial  OLS_Log OLS_withoutLog
## 1 4273540          3483.805 663.6531       3922.026
```

VI. Conclusion

Again, our question of interest is whether the cumulative cases in the past 41 days of each country can be affected by (1) fully vaccinated rate (2) the time vaccine has been implemented (3) the number of vaccine types a country used and (4) restrictions(safety guidance) in each country.
From our three approaches for exploring the relationship, the result from all three models concluded that the start of the vaccine date and stringency index have no statistical evidence to be able to explain the total cases in the past 41 days. However, fully vaccinated rate and the number of vaccine types a country has delivered have impact on the total new cases. The more people in one country who got fully vaccinated, tend to have more new positive cases. While our findings, don't align with our intuition, that is we expected to see that higher poly-vaccination rate would convert to few cases we see the opposite effect. This phenomenon might happen because of the density and population which we didn't consider in our model. Additionally, it could also be due to effects such as new variants of covid that are evasive to the vaccination and this information was also not present in the model.
Besides, the more number of vaccine type a country used tend to have less cases in the past few weeks. This is quite an interesting result. Even though the power of explanation of our model might not be as good, the result leads to even more question of interests about this globally pandemic COVID-19.

In a causal analysis, the independent variables are taken as the cause of the response variable. Which is in our cases, the different amount of vaccine type used and fully vaccination rate could lead to different amount of total covid cases. However, because our data is totally observational and non-experimental, it may not be appropriate to performed causal inference on our data.

In the further research, if we have the data about how many proportion of each type of vaccine delivered to people, we might be able to acquire more hinder relationship between different types of vaccines and positive cases.

Acknowledgement

1.Approved Vaccines (https://covid19.trackvaccines.org/vaccines/approved/ (https://covid19.trackvaccines.org/vaccines/approved/))
2.Policy Responses to the Coronavirus Pandemic (https://ourworldindata.org/policy-responses-covid (https://ourworldindata.org/policy-responses-covid))
3.Six months of COVID vaccines: what 1.7 billion doses have taught scientists (https://www.nature.com/articles/d41586-021-01505-x (https://www.nature.com/articles/d41586-021-01505-x))
4.COVID-19 Vaccines vs Variants—Determining How Much Immunity Is Enough (https://jamanetwork.com/journals/jama/fullarticle/2777785 (https://jamanetwork.com/journals/jama/fullarticle/2777785))
5. Study shows dramatic decline in effectiveness of all three COVID-19 vaccines over time (https://www.latimes.com/science/story/2021-11-04/study-shows-dramatic-decline-in-effectiveness-of-covid-19-vaccines (https://www.latimes.com/science/story/2021-11-04/study-shows-dramatic-decline-in-effectiveness-of-covid-19-vaccines))
6. Omicron likely to weaken COVID vaccine protection (https://www.nature.com/articles/d41586-021-03672-3 (https://www.nature.com/articles/d41586-021-03672-3))
7. Our world in data (https://ourworldindata.org/coronavirus-testing#the-positive-rate-a-crucial-metric-for-understanding-the-pandemic (https://ourworldindata.org/coronavirus-testing#the-positive-rate-a-crucial-metric-for-understanding-the-pandemic))
8. The four types of COVID-19 vaccine – a snapshot (https://www.healthcareitnews.com/news/emea/four-types-covid-19-vaccine-snapshot (https://www.healthcareitnews.com/news/emea/four-types-covid-19-vaccine-snapshot))
9. Types of covid-19 vaccines (https://covid19.trackvaccines.org/vaccine-types/ (https://covid19.trackvaccines.org/vaccine-types/)) 10.The different types of COVID-19 vaccines (https://www.who.int/news-room/feature-stories/detail/the-race-for-a-covid-19-vaccine-explained (https://www.who.int/news-room/feature-stories/detail/the-race-for-a-covid-19-vaccine-explained))
11.Poisson Regression (https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Poisson_Regression.pdf (https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Poisson_Regression.pdf))
12.NEGATIVE BINOMIAL REGRESSION https://stats.oarc.ucla.edu/stata/dae/negative-binomial-regression/# (https://stats.oarc.ucla.edu/stata/dae/negative-binomial-regression/#):~:text=The%20form%20of%20the%20model,3)%20%2B%20b3math
13.Negative_binomial_distribution (https://en.wikipedia.org/wiki/Negative_binomial_distribution (https://en.wikipedia.org/wiki/Negative_binomial_distribution))
14.Poisson_regression (https://en.wikipedia.org/wiki/Poisson_regression (https://en.wikipedia.org/wiki/Poisson_regression))
15 Plot-fitted-lines (https://aosmith.rbind.io/2018/11/16/plot-fitted-lines/ (https://aosmith.rbind.io/2018/11/16/plot-fitted-lines/))

Session information

```
sessionInfo()
```

```
## R version 4.1.1 (2021-08-10)
## Platform: aarch64-apple-darwin20 (64-bit)
## Running under: macOS Monterey 12.2.1
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.1-arm64/Resources/lib/libRblas.0.d
ylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.1-arm64/Resources/lib/libRlapack.d
ylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] jtools_2.1.4     AER_1.2-9        survival_3.2-11  car_3.0-11
##  [5] carData_3.0-4    gplots_3.1.1     ggpubr_0.4.0     tibble_3.1.4
##  [9] sandwich_3.0-1   ggplot2_3.3.5    lmtest_0.9-39    zoo_1.8-9
## [13] pscl_1.5.5       stringi_1.7.6    stringr_1.4.0    lubridate_1.7.10
## [17] tidyr_1.1.4      dplyr_1.0.7      MASS_7.3-54
##
## loaded via a namespace (and not attached):
##  [1] sass_0.4.0         jsonlite_1.7.2     splines_4.1.1      gtools_3.9.2
##  [5] bslib_0.3.1        Formula_1.2-4      assertthat_0.2.1   highr_0.9
##  [9] pander_0.6.4       cellranger_1.1.0   yaml_2.2.1         pillar_1.6.2
## [13] backports_1.2.1    lattice_0.20-44    glue_1.4.2         digest_0.6.28
## [17] ggsignif_0.6.3     colorspace_2.0-2   cowplot_1.1.1      htmltools_0.5.2
## [21] Matrix_1.3-4       pkgconfig_2.0.3    broom_0.7.9        haven_2.4.3
## [25] purrr_0.3.4        scales_1.1.1       openxlsx_4.2.4     rio_0.5.27
## [29] generics_0.1.0     farver_2.1.0       ellipsis_0.3.2     withr_2.4.3
## [33] cli_3.1.1          magrittr_2.0.1     crayon_1.4.1       readxl_1.3.1
## [37] evaluate_0.14      fansi_0.5.0        rstatix_0.7.0      forcats_0.5.1
## [41] foreign_0.8-81     tools_4.1.1        data.table_1.14.2  hms_1.1.1
## [45] lifecycle_1.0.0    munsell_0.5.0      zip_2.2.0          compiler_4.1.1
## [49] jquerylib_0.1.4    caTools_1.18.2     rlang_0.4.11       grid_4.1.1
## [53] rstudioapi_0.13    bitops_1.0-7       labeling_0.4.2     rmarkdown_2.11
## [57] gtable_0.3.0       abind_1.4-5        DBI_1.1.1          curl_4.3.2
## [61] R6_2.5.1           knitr_1.34         fastmap_1.1.0      utf8_1.2.2
## [65] KernSmooth_2.23-20 Rcpp_1.0.7         vctrs_0.3.8        tidyselect_1.1.1
## [69] xfun_0.26
```

Code Appendix

```
## ----include=FALSE-----------------------------------------------------------
library(MASS) #use to conduct negative binomial analysis
library(dplyr)
library(tidyr)
library(lubridate)
library(stringr)
library(stringi)
library(pscl)
library(lmtest)
library(ggplot2)
library(sandwich)
library(tibble)


## ----include=FALSE-----------------------------------------------------------
daily <- read.csv("WHO-COVID-19-global-data.csv")
vaccine <- read.csv("vaccination-data.csv")
test <- read.csv("owid-covid-data.csv")


## ----include=FALSE-----------------------------------------------------------
str(daily)


## ----include=FALSE-----------------------------------------------------------
#change Date_reported column to date data
daily <- daily %>% mutate(Date_reported = ymd(Date_reported))
#change qualitative data into factor
daily <- daily %>% mutate_if(is.character, as.factor)
str(daily)


## ----include=FALSE-----------------------------------------------------------
#examine numbers of missing values in each column
sapply(daily, function(x) sum(is.na(x)))


## ----include=FALSE-----------------------------------------------------------
#filter only same country name with vaccine
same_country <- unique(vaccine$COUNTRY)[(unique(vaccine$COUNTRY) %in% unique(daily$Count
ry))]
daily_need <- daily %>%
  filter(Country %in% same_country) %>%
  select(c(Date_reported,Country, WHO_region,New_cases,New_deaths)) %>%
  mutate(year = year(Date_reported),
         month = month(Date_reported),
         mday = mday(Date_reported)) %>%
  filter(year == '2022')


## ----include=FALSE-----------------------------------------------------------
feb_day_count <- daily_need %>%
```

```
  group_by(Country,month) %>%
  summarize(count_mday = n())


## ----include=FALSE-----------------------------------------------------------
daily_need <- daily_need %>%
  filter(Date_reported >= '2022-01-01' & Date_reported <= '2022-02-09')


## ----include=FALSE-----------------------------------------------------------
daily_final <- daily_need %>%
  group_by(Country) %>%
  summarise(total_new_cases = sum(New_cases),
            total_new_deaths = sum(New_deaths))


## ----include=FALSE-----------------------------------------------------------
daily_final


## ----include=FALSE-----------------------------------------------------------
str(vaccine)


## ----echo=FALSE--------------------------------------------------------------
#concatenate every row value of VACCINES_USED together
vac_use_vec <- sapply(vaccine$VACCINES_USED, paste)
#split it by "," and make it become a list
vac_use_list <- str_split(vac_use_vec,",")
#turn it back to a character vector
vac_use_vec <- unlist(vac_use_list)
#trim each character
vac_use_vec <- sapply(vac_use_vec,str_trim)
#find unique vaccines
vaccine_type <- stri_unique(vac_use_vec)
vaccine_type

## ----include=FALSE-----------------------------------------------------------
#make a table contains binary values of each vaccine type for each country
vaccine_type_binary <- data.frame(matrix(ncol = length(vaccine_type), nrow = nrow(vaccin
e)))
for (i in 1:length(vaccine_type)) {
  vaccine_type_binary[,i] <- as.integer(grepl(vaccine_type[i],vaccine$VACCINES_USED))
}
#change the columns name as same as vaccine_type
vaccine_type_name <- vaccine_type
vaccine_type_name <- str_replace_all(vaccine_type_name," - ","_")
vaccine_type_name <- str_replace_all(vaccine_type_name," ","_")
vaccine_type_name <- str_replace_all(vaccine_type_name,"-","_")
colnames(vaccine_type_binary) <- vaccine_type_name
vaccine_type_name
```

```
## ----include=FALSE-----------------------------------------------------------
vaccine_type_binary <- vaccine_type_binary %>%
  #remove one unnecessary row
  select(-c(12)) %>%
  #calculate the amount of certain type of vaccine a country has used
  mutate(inactivated_virus =
           Beijing_CNBG_BBIBP_CorV +
           Sinovac_CoronaVac +
           Bharat_Covaxin +
           IMB_Covidful +
           Wuhan_CNBG_Inactivated +
           Shifa_COVIran_Barakat +
           RIBSP_QazVac +
           Julphar_Hayat_Vax +
           Turkovac,
         viral_vector =
           Janssen_Ad26.COV_2_S +
           SII_Covishield +
           AstraZeneca_Vaxzevria +
           Gamaleya_Gam_Covid_Vac +
           CanSino_Convidecia +
           Gamaleya_Sputnik_Light +
           Gamaleya_Sputnik_V +
           AstraZeneca_AZD1222 +
           Shenzhen_LV_SMENP_DC,
         mRNA =
           Pfizer_BioNTech_Comirnaty +
           Moderna_Spikevax +
           Moderna_mRNA_1273,
         subunit =
           Novavax_Covavax +
           Anhui_ZL_Recombinant +
           CIGB_CIGB_66 +
           Finlay_Soberana_Plus +
           Finlay_Soberana_02 +
           SRCVB_EpiVacCorona,
         DNA = Zydus_ZyCov_D) %>%
  #if the country has use as least one of that type, code it 1, else 0
  mutate(inactivated_virus = ifelse(inactivated_virus != 0, 1, 0),
         viral_vector = ifelse(viral_vector != 0, 1, 0),
         mRNA = ifelse(mRNA != 0, 1, 0),
         subunit = ifelse(subunit != 0, 1, 0),
         DNA = ifelse(subunit != 0, 1, 0)) %>%
  mutate(vaccine_type_used = inactivated_virus + viral_vector + mRNA + subunit + DNA)



## ----include=FALSE-----------------------------------------------------------
#join back with vaccine data
vaccine <- cbind(vaccine, vaccine_type_binary)
```

```
## ----include=FALSE-----------------------------------------------------------
#change DATE_UPDATED column to date data
#change qualitative data into factors
#change binary columns and NUMBER_VACCINES_TYPES_USED into factor
vaccine <- vaccine %>%
  mutate(
    DATE_UPDATED = ymd(DATE_UPDATED),
    FIRST_VACCINE_DATE = ymd(FIRST_VACCINE_DATE)) %>%
  mutate_if(is.character, as.factor) %>%
  mutate_at(tail(colnames(vaccine), n = 29), factor)


## ----include=FALSE-----------------------------------------------------------
vaccine_final <- vaccine %>% select(! c('DATA_SOURCE', 'TOTAL_VACCINATIONS','PERSONS_VAC
CINATED_1PLUS_DOSE', 'TOTAL_VACCINATIONS_PER100', 'PERSONS_VACCINATED_1PLUS_DOSE','PERSO
NS_FULLY_VACCINATED'))


## ----include=FALSE-----------------------------------------------------------
#convert one iso_code to a correct one
test <- test %>% mutate(iso_code = replace(iso_code, iso_code == "OWID_KOS", "XKX"))
#find the same iso between vaccine_final and test
same_iso <- unique(vaccine_final$ISO3)[(unique(vaccine_final$ISO3) %in% unique(test$iso_
code))]


## ----include=FALSE-----------------------------------------------------------
#select target columns and rows
test_need <- test %>%
  select(c("iso_code","location", "date", "stringency_index", "population")) %>%
  filter(iso_code %in% same_iso)


## ----include=FALSE-----------------------------------------------------------
str(test_need)
#change data type
test_need <- test_need %>%
  mutate(date = ymd(date)) %>%
  mutate_if(is.character, as.factor)


## ----include=FALSE-----------------------------------------------------------
test_need <- test_need %>%
  mutate(year = year(date),
         month = month(date)) %>%
  filter(year == '2022')


## ----include=FALSE-----------------------------------------------------------
feb_day_count_test <- test_need %>%
  group_by(iso_code,month) %>%
```

```
  summarize(count_mday = n())


## ----include=FALSE-----------------------------------------------------------
test_need <- test_need %>%
  filter(date >= "2022-01-01" & date <= "2022-02-09")


## ----include=FALSE-----------------------------------------------------------
test_need <- na.omit(test_need)
test_final <-  test_need %>%
  group_by(iso_code) %>%
  summarise(stringency_index = mean(stringency_index),
            population = max(population))


## ----include=FALSE-----------------------------------------------------------
vaccine_and_daily <- inner_join(vaccine_final,daily_final, by = c("COUNTRY" = "Country"
))
covid <- inner_join(vaccine_and_daily, test_final, by = c("ISO3" = "iso_code"))


## ----include=FALSE-----------------------------------------------------------
#covid <- na.omit(covid)
sapply(covid, function(x) sum(is.na(x)))
covid <- na.omit(covid)


## ----include=FALSE-----------------------------------------------------------
covid <- covid %>%
  mutate(total_new_cases_per_million = round((total_new_cases/population)*1000000),
         total_new_deaths_per_million = round((total_new_deaths/population)*1000000),
         FIRST_VACCINE_DATE = as.Date('2022-02-09') - FIRST_VACCINE_DATE,
         FIRST_VACCINE_DATE = as.numeric(FIRST_VACCINE_DATE)
  )

#%>% mutate(FIRST_VACCINE_DATE = as.factor(FIRST_VACCINE_DATE))
#case_when(
           #FIRST_VACCINE_DATE <= as.Date('2020-09-30') ~ 0,
           #as.Date('2020-06-30') < FIRST_VACCINE_DATE & FIRST_VACCINE_DATE < as.Date('2
021-01-01') ~ 1,
           #as.Date('2020-12-31') < FIRST_VACCINE_DATE & FIRST_VACCINE_DATE < as.Date('2
021-04-01') ~ 2,
           #as.Date('2021-03-31') < FIRST_VACCINE_DATE & FIRST_VACCINE_DATE < as.Date('2
021-07-01') ~ 3,
           #as.Date('2021-06-30') < FIRST_VACCINE_DATE & FIRST_VACCINE_DATE < as.Date('2
021-10-01') ~ 4,
           #as.Date('2021-09-30') < FIRST_VACCINE_DATE & FIRST_VACCINE_DATE < as.Date('2
022-01-01') ~ 5)


## ----include=FALSE-----------------------------------------------------------
```

```
covid_trim <- covid %>%
  select(c("COUNTRY", "ISO3", "WHO_REGION", "PERSONS_VACCINATED_1PLUS_DOSE_PER100", "PER
SONS_FULLY_VACCINATED_PER100", "FIRST_VACCINE_DATE","vaccine_type_used", "total_new_case
s_per_million", "total_new_deaths_per_million", "stringency_index" ))  %>% mutate(vaccin
e_type_used = droplevels(vaccine_type_used))


## ----echo = FALSE----------------------------------------------------------
tibble(head(covid_trim,6))


## ----echo = FALSE----------------------------------------------------------
summary(covid_trim)


## ----echo = FALSE----------------------------------------------------------
#overview of a dependent variable
par(mfrow = c(2,3))
p1 <- ggplot(covid_trim, aes(x=total_new_cases_per_million)) + geom_histogram(color="bla
ck", fill="lightblue") + theme(axis.title.x = element_text(size = 10))
p2 <- ggplot(covid_trim, aes(x=PERSONS_VACCINATED_1PLUS_DOSE_PER100)) + geom_histogram(c
olor="black", fill="orchid4") + theme(axis.title.x = element_text(size = 10))
p3 <- ggplot(covid_trim, aes(x=PERSONS_FULLY_VACCINATED_PER100)) + geom_histogram(color=
"black", fill="orchid4") + theme(axis.title.x = element_text(size = 10))
p4 <- ggplot(covid_trim, aes(x=FIRST_VACCINE_DATE)) + geom_histogram(color="black", fill
="white") + theme(axis.title.x = element_text(size = 10))
p5 <- ggplot(covid_trim, aes(x=stringency_index)) + geom_histogram(color="black", fill=
"white") + theme(axis.title.x = element_text(size = 10))
library(ggpubr)
ggarrange(p1,p2,p3,p4,p5,
          ncol = 3, nrow = 2)


## ----include = FALSE-------------------------------------------------------
par(mfrow = c(2,3))
boxplot(covid_trim$total_new_cases_per_million, main="total_new_cases_per_million", type
="l")
boxplot(covid_trim$PERSONS_VACCINATED_1PLUS_DOSE_PER100, main="PERSONS_VACCINATED_1PLUS_
DOSE_PER100", type="l")
boxplot(covid_trim$PERSONS_FULLY_VACCINATED_PER100, main="PERSONS_FULLY_VACCINATED_PER10
0", type="l")
boxplot(covid_trim$FIRST_VACCINE_DATE, main="FIRST_VACCINE_DATE", type="l")
boxplot(covid_trim$stringency_index, main="stringency_index", type="l")

## ----echo = FALSE----------------------------------------------------------
panel.cor <- function(x, y) {
    par(usr = c(0, 1, 0, 1))
    r <- round(cor(x, y, use = "complete.obs"), 2)
    txt <- paste0("R = ", r)
    cex.cor <- 0.8/strwidth(txt)
    text(0.5, 0.5, txt, cex = cex.cor * (abs(r) + 1))
}
```

```
pairs(covid_trim, lower.panel = panel.cor)


## ----echo = FALSE----------------------------------------------------------
#To see if vaccine_type_used is a good explanatory varaible
#display the summary statistics by vaccine_type_used
with(covid_trim, tapply(total_new_cases_per_million, vaccine_type_used, function(x) {
  sprintf("M (SD) = %1.2f (%1.2f)", mean(x), sd(x))
}))


## ----echo = FALSE----------------------------------------------------------
#pairwise bivariate displays of dependent variables against each of the regressor
formula <- total_new_cases_per_million ~ FIRST_VACCINE_DATE + vaccine_type_used + PERSON
S_FULLY_VACCINATED_PER100 + stringency_index
par(mfrow = c(2,2))
plot(formula, data = covid_trim)

## ----echo = FALSE----------------------------------------------------------
library(gplots)
#show comparison with average wage by occupations
plotmeans(total_new_cases_per_million~vaccine_type_used,data=covid_trim, ylab = 'average
total new cases', main = 'Main Effect Plot')


## --------------------------------------------------------------------------
#ggplot(covid_trim, aes(total_new_cases_per_million, fill = vaccine_type_used)) +
  #geom_histogram(position="dodge")


## ----echo = FALSE----------------------------------------------------------
ggplot(covid_trim, aes(total_new_cases_per_million, fill = vaccine_type_used)) +
  geom_histogram() +
  facet_grid(vaccine_type_used ~ ., margins = TRUE, scales = "free")


## ----include = FALSE-------------------------------------------------------
dispersion_test <- function(x)
{
  res <- 1-2 * abs((1 - pchisq((sum((x - mean(x))^2)/mean(x)), length(x) - 1))-0.5)

  cat("Dispersion test of count data:\n",
      length(x), " data points.\n",
      "Mean: ",mean(x),"\n",
      "Variance: ",var(x),"\n",
      "Probability of being drawn from Poisson distribution: ",
      round(res, 3),"\n", sep = "")

  invisible(res)
}
```

```
## ----echo = FALSE----------------------------------------------------------
dispersion_test(covid_trim$total_new_cases_per_million)


## ----echo = FALSE----------------------------------------------------------
model_pois <- glm(total_new_cases_per_million ~ FIRST_VACCINE_DATE + vaccine_type_used +
PERSONS_FULLY_VACCINATED_PER100 + stringency_index, data = covid_trim, family = "poisso
n")
summary(model_pois)


## ----include = FALSE-------------------------------------------------------
library(AER)
dispersiontest(model_pois)


## ----include = FALSE-------------------------------------------------------
with(covid_trim, tapply(total_new_cases_per_million, vaccine_type_used, function(x) {
    sprintf("M (SD) = %1.2f (%1.2f)", mean(x), sd(x))
}))


## ----include = FALSE-------------------------------------------------------
#goodness-of-fit
1 - pchisq(summary(model_pois)$deviance, summary(model_pois)$df.residual)


## ----echo = FALSE----------------------------------------------------------
#preliminary model of negative binomial regression
model_nb0 <- glm.nb(total_new_cases_per_million ~ FIRST_VACCINE_DATE + vaccine_type_used
+ PERSONS_FULLY_VACCINATED_PER100 + stringency_index, data = covid_trim)
summary(model_nb0)

## ----include = FALSE-------------------------------------------------------
#goodness-of-fit
1 - pchisq(summary(model_nb0)$deviance, summary(model_nb0)$df.residual)


## ----include = FALSE-------------------------------------------------------
round(coef(model_nb0),4)


## ----echo = FALSE----------------------------------------------------------
#test whether to remove FIRST_VACCINE_DATE and stringency_index
model_nb1 <- update(model_nb0, . ~ . - FIRST_VACCINE_DATE - stringency_index)
anova(model_nb0,model_nb1)


## ----echo = FALSE----------------------------------------------------------
#test whether to leave vaccine_type_used
model_nb2 <- update(model_nb1, . ~ . - vaccine_type_used)
```

```
anova(model_nb1,model_nb2)


## ----echo = FALSE----------------------------------------------------------
library(jtools)
summ(model_nb1, exp = T)


## ----include = FALSE-------------------------------------------------------
model_pois1 <- glm(total_new_cases_per_million ~ vaccine_type_used + PERSONS_FULLY_VACCI
NATED_PER100, data = covid_trim, family = "poisson")
summary(model_pois1)


## ----include=FALSE---------------------------------------------------------
pchisq(2*(logLik(model_nb1)-logLik(model_pois1)), df = 1, lower.tail = FALSE)
qchisq(0.05,1,lower.tail = FALSE)


## ----include = FALSE-------------------------------------------------------
#goodness-of-fit
1 - pchisq(summary(model_nb1)$deviance, summary(model_nb1)$df.residual)


## ----echo = FALSE----------------------------------------------------------
AIC(model_pois1,model_nb2)


## ----include = FALSE-------------------------------------------------------
covid_trim_remove0 <- covid_trim %>%
  filter(total_new_cases_per_million != 0)

model_lm_pre <- lm(total_new_cases_per_million ~ FIRST_VACCINE_DATE + vaccine_type_used
 + PERSONS_FULLY_VACCINATED_PER100 + stringency_index, data = covid_trim_remove0)


## ----include = FALSE-------------------------------------------------------
plot(model_lm_pre)

## ----echo = FALSE----------------------------------------------------------
boxcox(model_lm_pre)


## ----echo = FALSE----------------------------------------------------------
fit_aov <- aov(log(total_new_cases_per_million+1) ~ vaccine_type_used, data = covid_tri
m)
summary(fit_aov)


## ----echo = FALSE----------------------------------------------------------
model_lm0 <- lm(log(total_new_cases_per_million+1) ~ FIRST_VACCINE_DATE + vaccine_type_u
sed + PERSONS_FULLY_VACCINATED_PER100 + stringency_index, data = covid_trim)
```

```
summary(model_lm0)


## ----echo = FALSE----------------------------------------------------------
model_lm1 <- update(model_lm0, .~. - FIRST_VACCINE_DATE - stringency_index)
#lm(total_new_cases_per_million ~ vaccine_type_used + PERSONS_FULLY_VACCINATED_PER100, d
ata = covid_trim)
#summary(model_lm1)
anova(model_lm1,model_lm0)

## ----include = FALSE-------------------------------------------------------
summary(model_lm1)


## ----echo = FALSE----------------------------------------------------------
model_lm2 <- update(model_lm1, .~. - vaccine_type_used)
anova(model_lm2,model_lm1)


## ----echo = FALSE----------------------------------------------------------
model_select <- data.frame(Poisson = AIC(model_pois1), Negative_Binomial = AIC(model_nb
1), OLS = AIC(model_lm1))
model_select


## ----echo = FALSE----------------------------------------------------------
covid_trim_predict <- covid_trim %>%
  select(c("total_new_cases_per_million","vaccine_type_used", "PERSONS_FULLY_VACCINATED_
PER100")) %>%
  mutate(fitted = fitted.values(model_lm1))

confit <- predict(model_lm1, interval = "confidence")
covid_trim_predict <- cbind(covid_trim_predict,confit)


## --------------------------------------------------------------------------
ggplot(covid_trim_predict, aes(x = PERSONS_FULLY_VACCINATED_PER100, y = log(total_new_ca
ses_per_million), color = vaccine_type_used) ) +
     geom_point() +
     geom_ribbon(aes(ymin = lwr, ymax = upr, fill = vaccine_type_used, color = NULL), al
pha = .15) +
     geom_line(aes(y = fitted), size = 1)


## ----echo = FALSE----------------------------------------------------------
par(mfrow=c(2,2))
plot(model_lm2)


## ----echo = FALSE----------------------------------------------------------
model_lm_without_log <- lm(total_new_cases_per_million ~ vaccine_type_used + PERSONS_FUL
LY_VACCINATED_PER100, data = covid_trim)
```

```
summary(model_lm_without_log)

## ----echo = FALSE----------------------------------------------------------
model_compare <- data.frame(Poisson = AIC(model_pois1), Negative_Binomial = AIC(model_nb
1), OLS_Log = AIC(model_lm1), OLS_withoutLog = AIC(model_lm_without_log))
model_compare


## --------------------------------------------------------------------------
sessionInfo()


## ----code = readLines(knitr::purl("/Users/alliewu/Desktop/STA207/Covid Project/Wun-syu
an_Wu_Project_Report.Rmd", documentation = 1)), echo = T, eval = F----
## NA
```