Math 535: Project Report

South Korea Covid-19 Cases Prediction

Team Member: Yeheng Liang,Thi Van Nguyen, Matthew Heym, Hanzhu Yan

Due Date = 5/25/2021

```
source("C:/Users/Bowen/Desktop/Spring 2021/final fianl report/535 R project data cleaning.r")
source("C:/Users/Bowen/Desktop/Spring 2021/final fianl report/535 plot ready.r")
source("C:/Users/Bowen/Desktop/Spring 2021/final fianl report/535 project model fitting and predction cum policy.r"
source("C:/Users/Bowen/Desktop/Spring 2021/final fianl report/535 matt.r")
source("C:/Users/Bowen/Desktop/Spring 2021/final fianl report/535 Thi.r")
```

Introduction

Coronavirus diseases 2019(COVID-19) has a substantial impact on every aspect of the world. It is an infectious disease caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). COVID-19 outbreak was first reported in December 2019 in Wuhan, China, and has resulted in an ongoing pandemic. At the time of writing this study, the total number of confirmed cases passed over 150 million in over 185 countries and territories worldwide, resulting in more than 3.1 million deaths. The World Health Organization has also announced that COVID-19 will very likely remain for multiple years until 85 of the world population has been vaccinated by effective COVID vaccine. It is precious to know what will happen in the coming weeks or months to prepare for possible waves.

Data Source

COVID-19 has infected more than 10,000 people in South Korea. KCDC (Korea Centers for Disease Control & Prevention) announces the information of COVID-19 quickly and transparently. They have made a structured dataset based on the report materials of KCDC and local governments.

The data set can be found on Kaggle:url{https://www.kaggle.com/kimjihoo/coronavirusdataset}

This big data set has many sub-datasets with different aspects about COVID-19. We decided to pick the weather, search trend, and policy to work on for our prediction. I pick the weather because SARS was first found in November 2002 in Guangdong, China was ended in June 2003 for no apparent reason. Guangdong is the southern part of China that experiences hot and wet weather around June. So, it is likely that the COVID-19 case has a negative association with temperature and humidity. Then, we pick search trend because it is very likely for a person to search for more information about COVID-19 when they or their friends and family is experiencing COVID-19 related symptoms. Lastly, policies are essential to stopping the spread of COVID-19, like a shelter in place, wearing a mask, and social distancing.

Goal

1. Using LDA, QDA,PCA, K-means for classification and clustering by provinces.

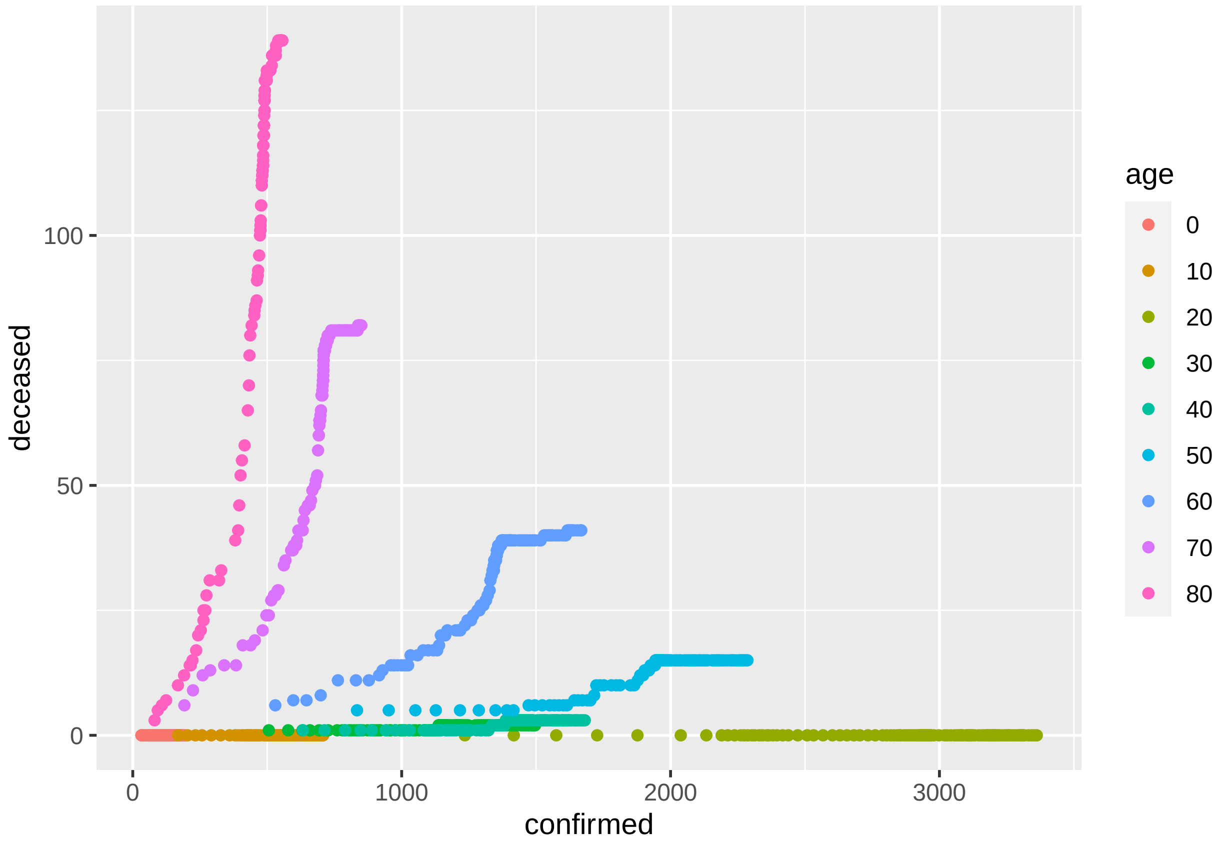
2. Find significant association between weather data, time age data, policy data and search trend data with new COVID cases.

3. Predict new COVID cases using weather data, policy data and search trend data.

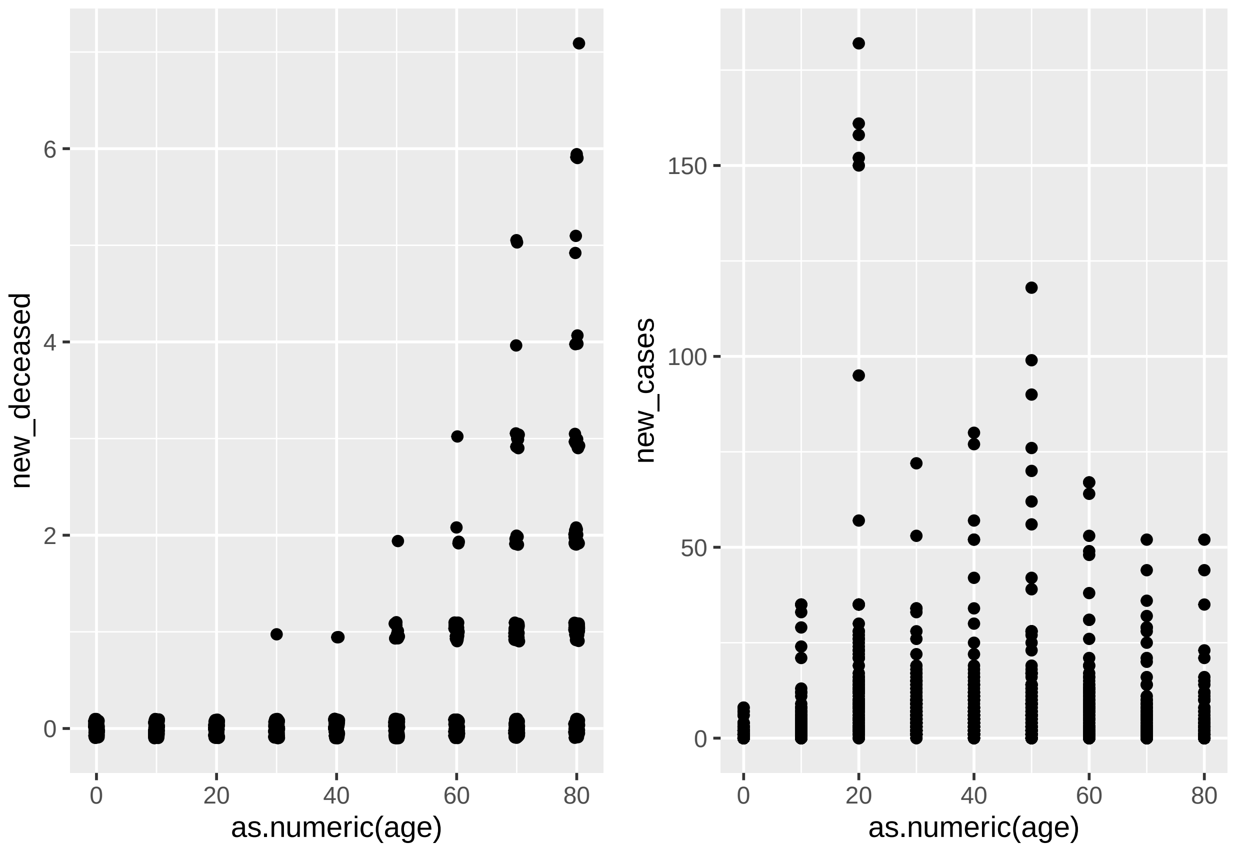
Data Visualization

We begin our data visualization by investigating whether age is a significant indicator for determining a new case of COVID. In doing so, we formulated a gg plot of the cumulative age data to get an initial understanding.

```
p111
```



```
grid.arrange(p112,p113,nrow=1)
```



Our ggplot consists of confirmed cases on the x-axis and deceased on the y axis. We observe that the younger age groups, mainly from 20-50, appear to be on the bottom right side, having significantly more confirmed cases and fewer deceased. However, the age groups 70-80 appear more towards the upper left and have fewer confirmed cases but more deceased.

We now enhance our analysis by constructing a new age data set consisting of new columns being new cases and new deceased each day. After doing so, we notice an exponential trend in the plot with age and new deceased. This meaning that as one grows older, the deceased rate rises exponentially, with people that are 80 having a much higher death rate than those that are below 60.

On the other hand, looking at the plot with age and new cases depicts a different result. We notice that one cannot conclude that as people get older, then more cases are likely to arise. The plot illustrates a non-linear trend with age and new cases in that the age range from 20-50 has significantly more cases than any other age group. This is probably as a result of this group socializing more and being less dormant compared to people who are older and not spreading the virus to the same degree. Next, we will look at the regression with age, deceased and new cases.

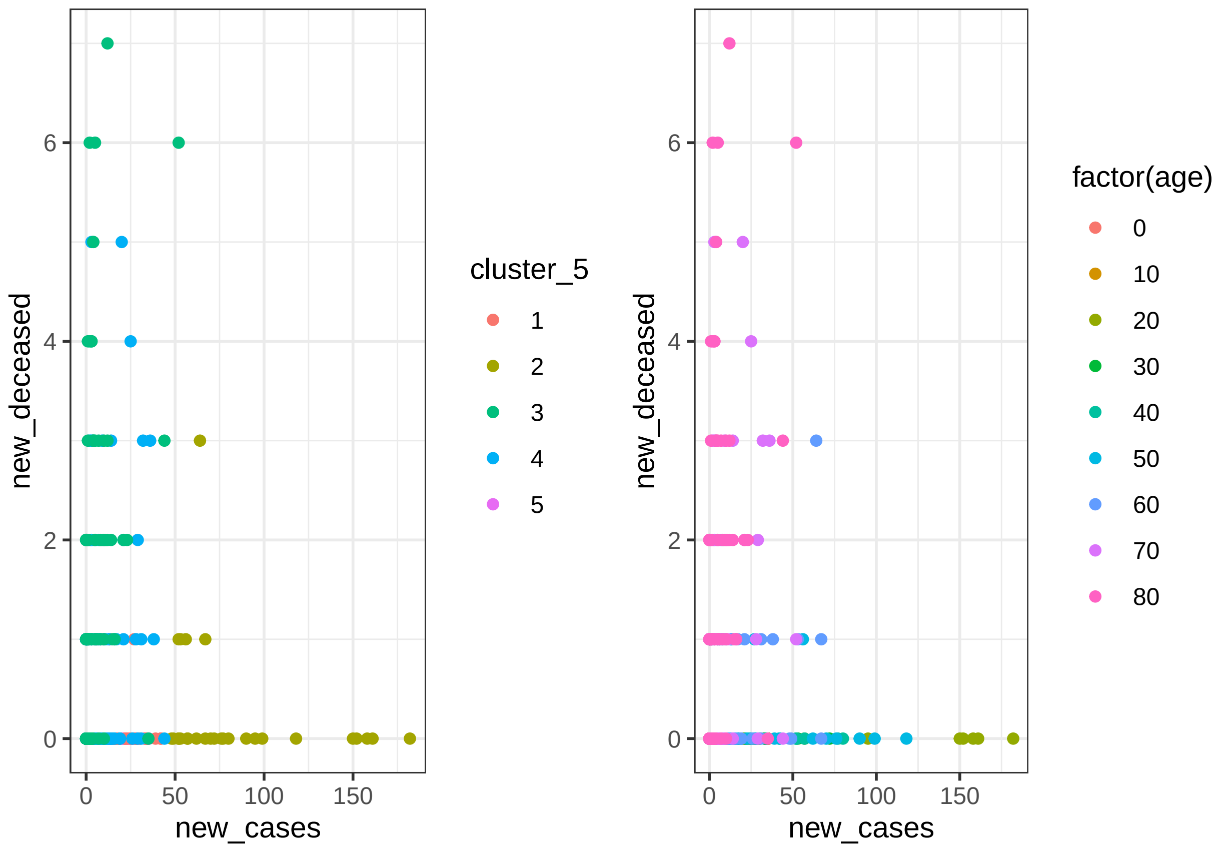
```
#indicating whether age is significant
reg2 <- lm(new_cases ~  age + new_deceased, data = Age)
summary(reg2)
```

```
##
## Call:
## lm(formula = new_cases ~ age + new_deceased, data = Age)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.725  -6.228  -2.492   0.772 164.275
```

```
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.3417     1.3834   0.970 0.332340
## age10          3.1500     1.9564   1.610 0.107668
## age20         16.3833     1.9564   8.374  < 2e-16 ***
## age30          6.8865     1.9564   3.520 0.000450 ***
## age40          7.3480     1.9564   3.756 0.000182 ***
## age50         10.5398     1.9573   5.385  8.9e-08 ***
## age60          7.3769     1.9671   3.750 0.000186 ***
## age70          2.4811     2.0064   1.237 0.216515
## age80         -0.3549     2.1124  -0.168 0.866608
## new_deceased   2.6220     0.7030   3.730 0.000202 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.15 on 1070 degrees of freedom
##   (9 observations deleted due to missingness)
## Multiple R-squared:  0.09705,    Adjusted R-squared:  0.08946
## F-statistic: 12.78 on 9 and 1070 DF,  p-value: < 2.2e-16
```

We model our regression with new cases as the response and the predictors being age and new deceased. We observe that the age groups between 20-60 are very significant with p values $< .05$. However, the age 70-80 group is very insignificant with p values of .21 and .87. From this regression, we can conclude that age provides a significant indication of new COVID cases. In this model, we can make an interpretation of age. For the age 10 bucket, there is expected to be 3.15 more new cases than if they were in the baseline bucket of the age 0 group. Additionally, we would expect there to be 16.3833 more new cases in the age 20 group than in the baseline age 0 group. Next, we'll explore K means clustering with age for data visualization. First, we added two more columns to the data set with a lag, consisting of deaths and cases yesterday, to obtain a better prediction for new COVID cases.

```
grid.arrange(p114,p115,nrow=1)
```

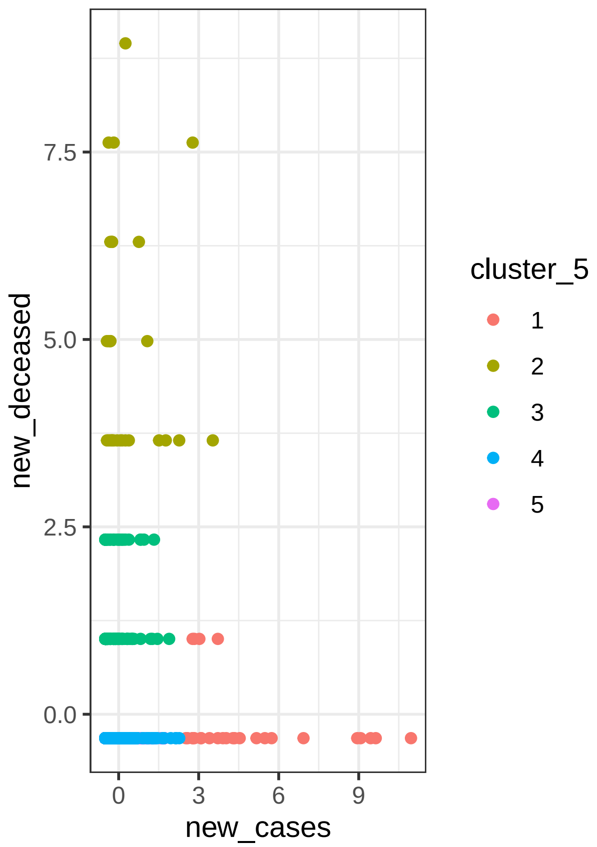


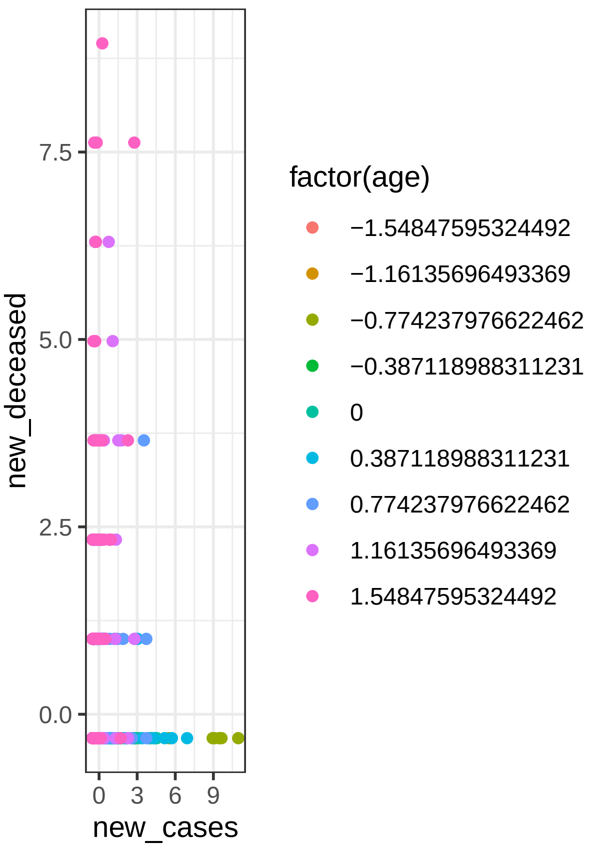
After applying K means clustering to the data, we notice a few interesting findings. We notice an almost complete overlap in profiling the age data to k means with the unscaled data. We divided the data into 5 clusters and see that people in the age 20-50 buckets are grouped, suggesting a similar profile. The second cluster is accounting for this age group on the bottom right with many new cases and few deceased. Moreover, we see the 3rd and 4th cluster is accounting for more deceased and less cases, preferably the age groups between 70 and 80. These groupings matching the age plot so similarly imply that age plays an important role for visualization when clustering the data.

Additionally, we did a multiple regression with new cases as the response, and age, new deceased, cases yesterday, and the clusters as predictors. In this regression, we would expect, for example cluster 2, to have 21 more new cases than the baseline cluster 1. This model illustrates that the categorical cluster variables are making accurate groupings relative to the age bucket groups. Also, the r^2 value of .86 is very good at representing most of the variation within the data. Lastly, the second cluster is significant in predicting new COVID cases due to its statistically significant p-value representing the age 20-50 group being clustered together. Therefore, this multiple regression supports the fact that age is influential when formulating the clusters.

To see another perspective, we look at the scaled data to account for other factors and less bias.

```r
grid.arrange(p116,p117,nrow=1)
```





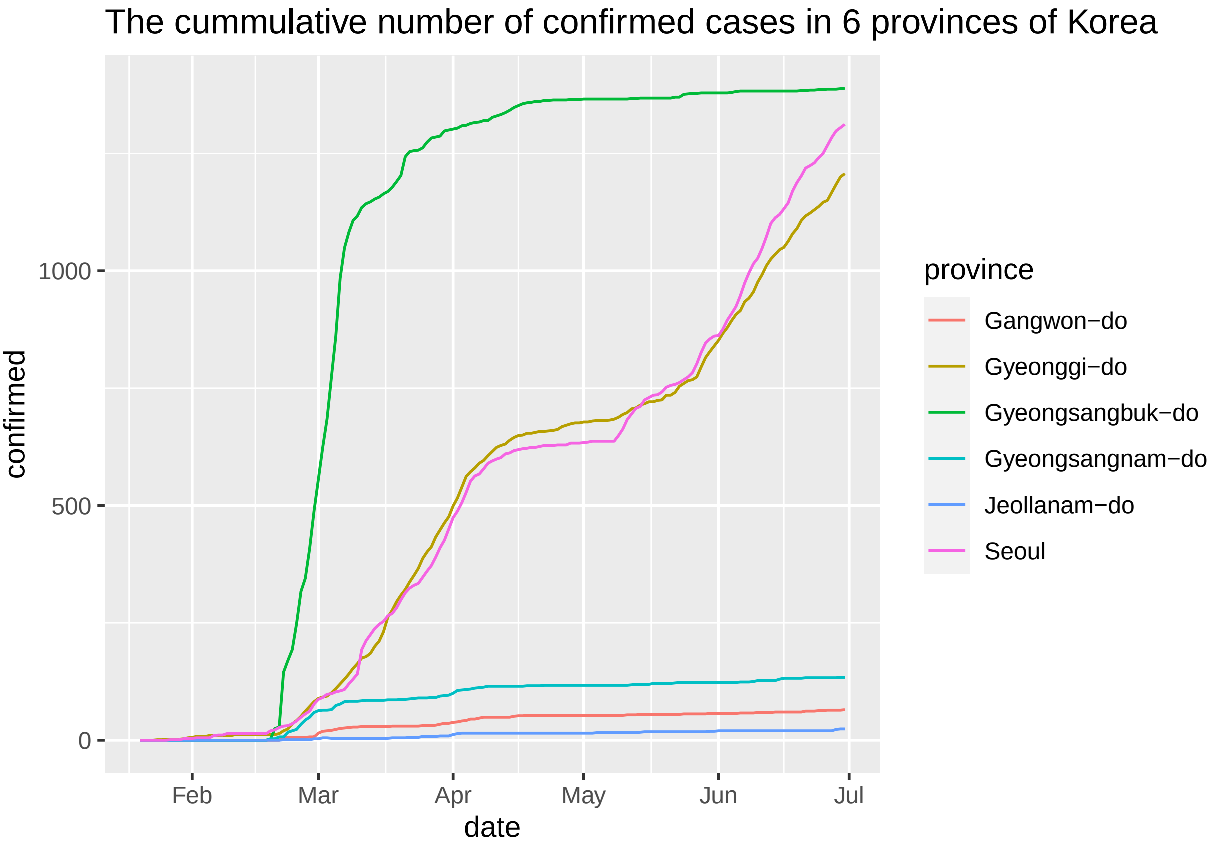
After scaling the data, we still arrive at similar findings. Although K means now splitting the data based on the risk of fatality, the significance of age is still prevalent. We see that the people in the age 20-50 age buckets are still low fatality (cluster 1) and people in the 70-80 age buckets are high and median fatality (cluster 2 and 3). These similarities in clustering with the unscaled data convey the significance of age grouping the data for visualization. The only difference is clusters 4 and 5, which are indiscernible because of such few cases and deaths. Overall, the K means clustering depicts age and its significance not with predicting new topics alone but also being influential for data visualization and exploration. Furthermore, policymakers and government officials might consider treating people in these age groups clustered together similarly instead of breaking them into smaller, arbitrary, non-data-driven groupings.

Classification and Clustering by Provinces

```
names(region_weather_data)
```

```
##  [1] "code.x"                  "province"
##  [3] "city"                    "latitude"
##  [5] "longitude"               "elementary_school_count"
##  [7] "kindergarten_count"      "university_count"
##  [9] "academy_ratio"           "elderly_population_ratio"
## [11] "elderly_alone_ratio"     "nursing_home_count"
## [13] "code.y"                  "date"
## [15] "avg_temp"                "min_temp"
## [17] "max_temp"                "precipitation"
## [19] "max_wind_speed"          "most_wind_direction"
## [21] "avg_relative_humidity"
```

```
p211
```



Since there are 17 different provinces with the same time points, I grouped the province's data and picked 6 provinces for clustering and classification. I pick 6 provinces in total 17 provinces because they have the earliest confirmed cases in Korea. Look at the plot; the plot shows the cumulative confirmed cases from January to June. We see that the top curve is Gyeongsangbuk-do which is a province in Eastern South Korea. It also has a coastal city well known for its ports and beaches among Koreans, so it attracts many tourists, which leads to rapidly increasing transmission rates of the Covid-19. The 2 middle curves of the confirmed cases are Seoul and Gyeonggi-do, which are also big cities with high population density. The other 3 provinces are "Gangwon-do," "Jeollanam-do," "Gyeongsangbuk-do," with the rate of infection, which is increasing slowly.

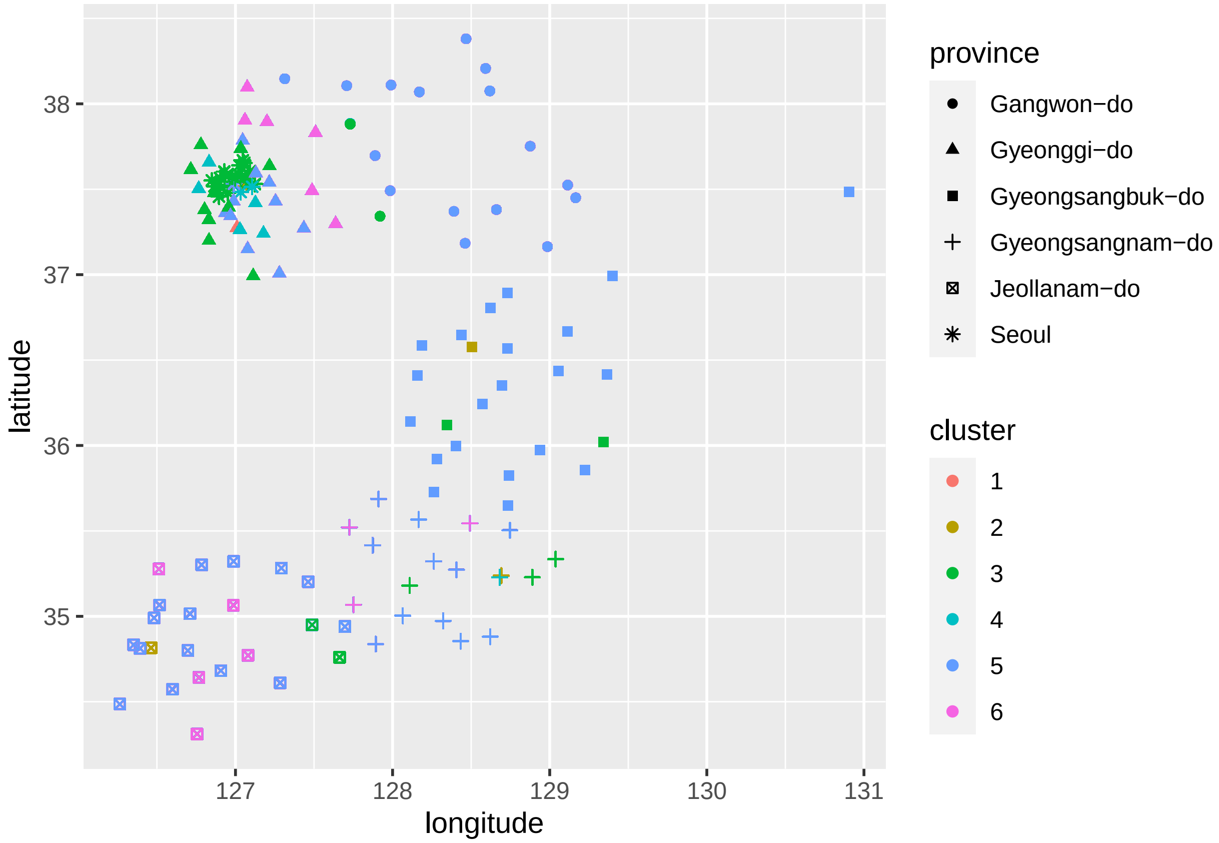
I use K-means method to cluster 6 regions by 14 predictors which are

elementary_school_count: the number of elementary schools
kindergarten_count: the number of kindergartens
university_count: the number of universities
academy_ratio: the ratio of academies
elderly_population_ratio: the ratio of the elderly population
elderly_alone_ratio: the ratio of elderly households living alone
nursing_home_count: the number of nursing homes

avg_temp: the average temperature
min_temp: the lowest temperature
max_temp: the highest temperature
precipitation: the daily precipitation
max_wind_speed: the maximum wind speed
most_wind_direction: the most frequent wind direction
avg_relative_humidity: the average relative humidity

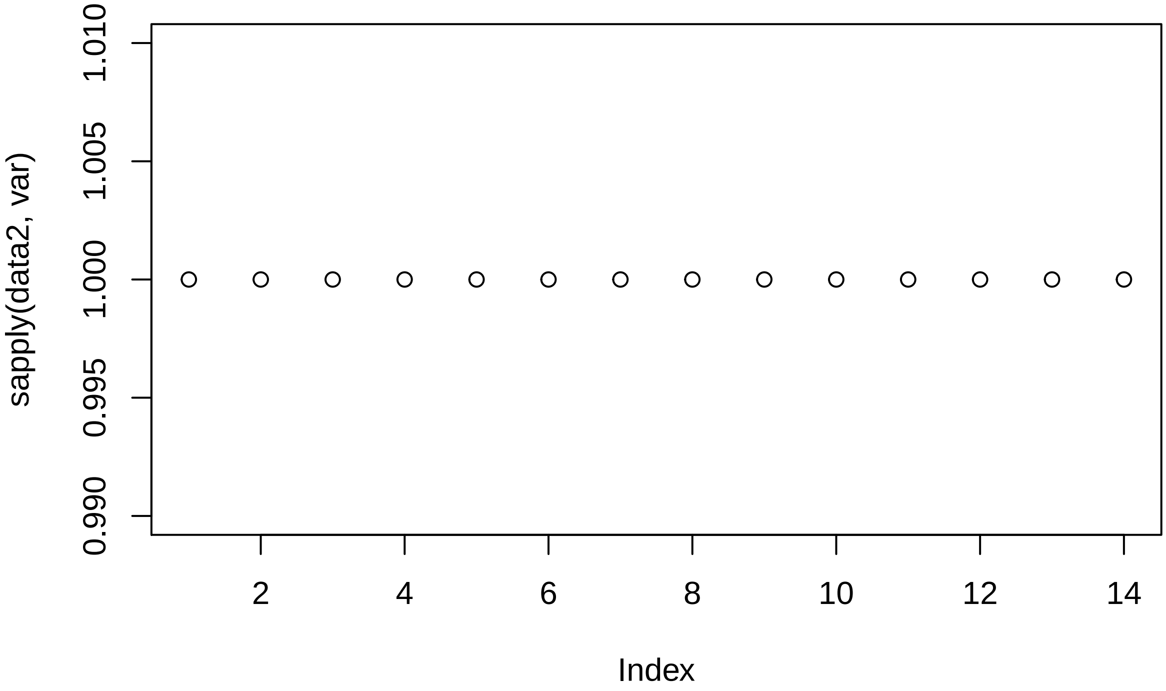
Then I split the data into training set and test set, saved the cluster number in the dataset as column.

```
p212
```

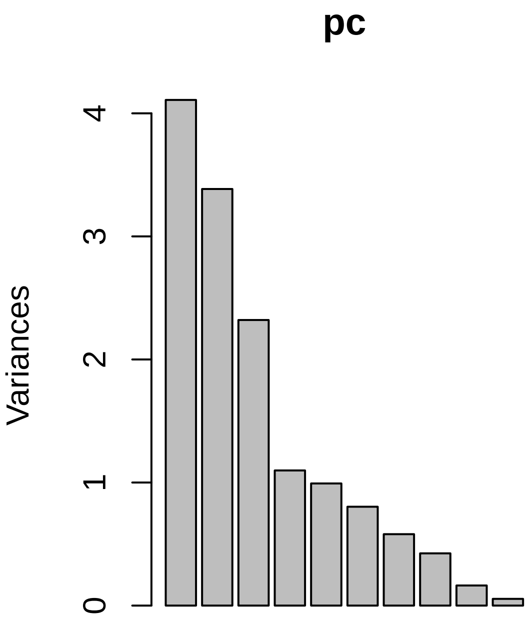


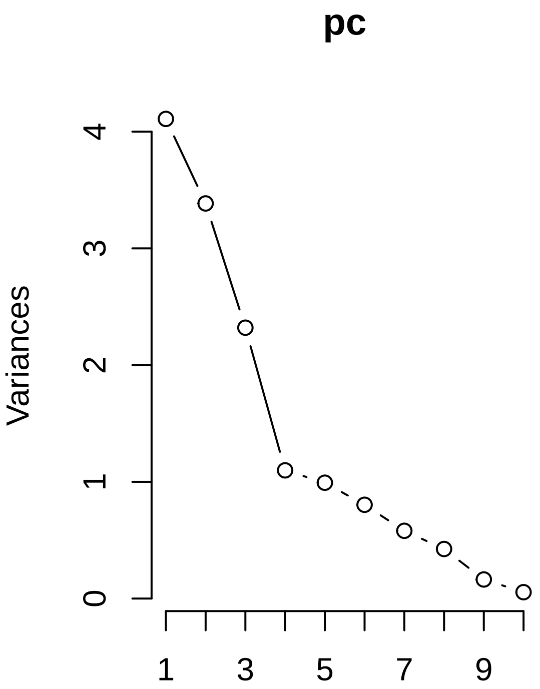
Look at the plot, we see that the clusters don't match with province. Then I scaled the data and Verified variance to uniform.

```
plot(sapply(data2, var))
```



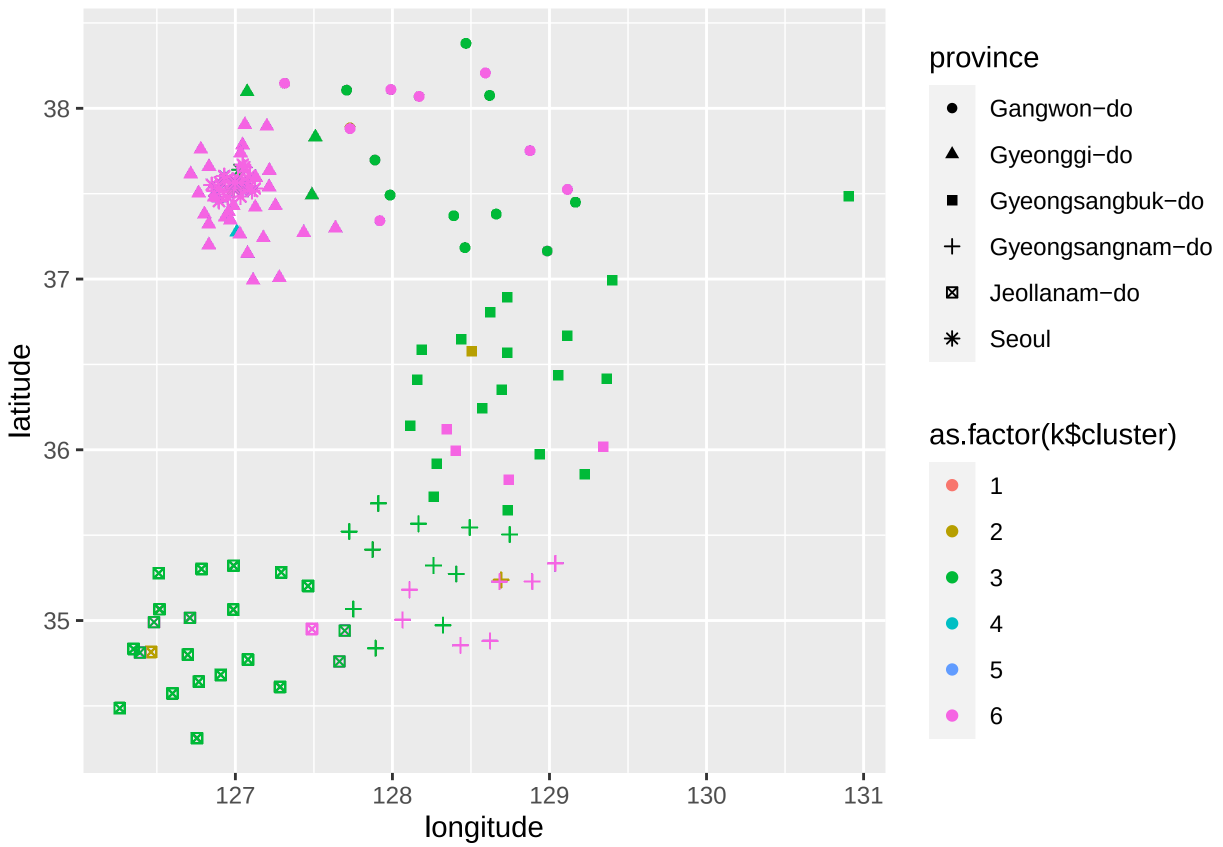
```
par(mfrow=c(1,2))
plot(pc)
plot(pc, type='l')
```





From the plots above, we can see that 4 components is 'elbow' and explains nearly 80% variance.

```
p216
```



After getting principal component vectors by prcomp instead of princomp and using 4 principal components, the result is similar with previous K-means plot with 14 predictors.

This part use LDA, QDA by using 7 geographical attributes which are:
elementary_school_count: the number of elementary schools
kindergarten_count: the number of kindergartens
university_count: the number of universities
academy_ratio: the ratio of academies
elderly_population_ratio: the ratio of the elderly population
elderly_alone_ratio: the ratio of elderly households living alone
nursing_home_count: the number of nursing homes

and 7 Climate features included:
avg_temp: the average temperature
min_temp: the lowest temperature
max_temp: the highest temperature
precipitation: the daily precipitation
max_wind_speed: the maximum wind speed
most_wind_direction: the most frequent wind direction
avg_relative_humidity: the average relative humidity

Then split randomly 2/3 for training dataset and 1/3 for test dataset.

```
## LDA
train_error_lda
```

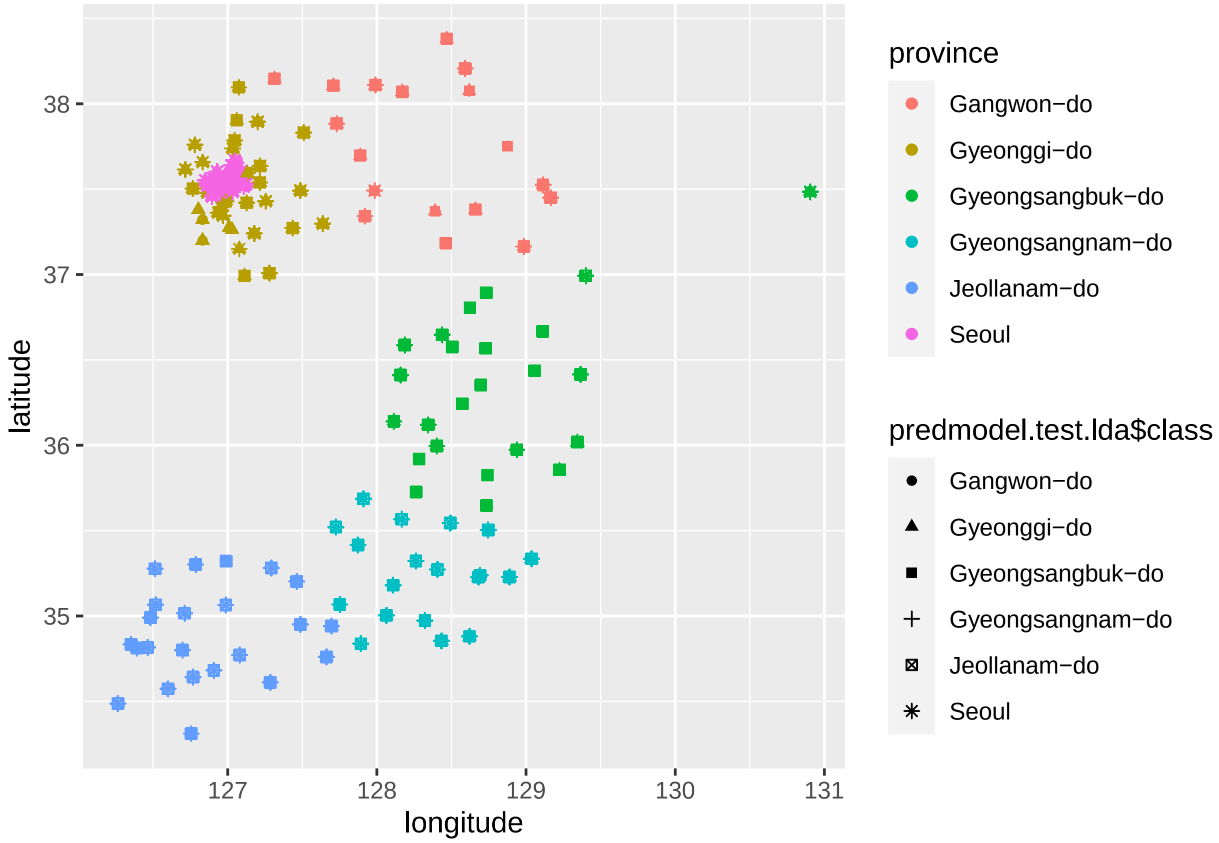
```
## [1] 0.2381898
```

```
test_error_lda
```

```
## [1] 0.2394283
```

For LDA, training error is 0.2381963, test_error is 0.2394283 which is similar to training_error.

```
## LDA
p217
```



In the plot above, many points are misclassification.

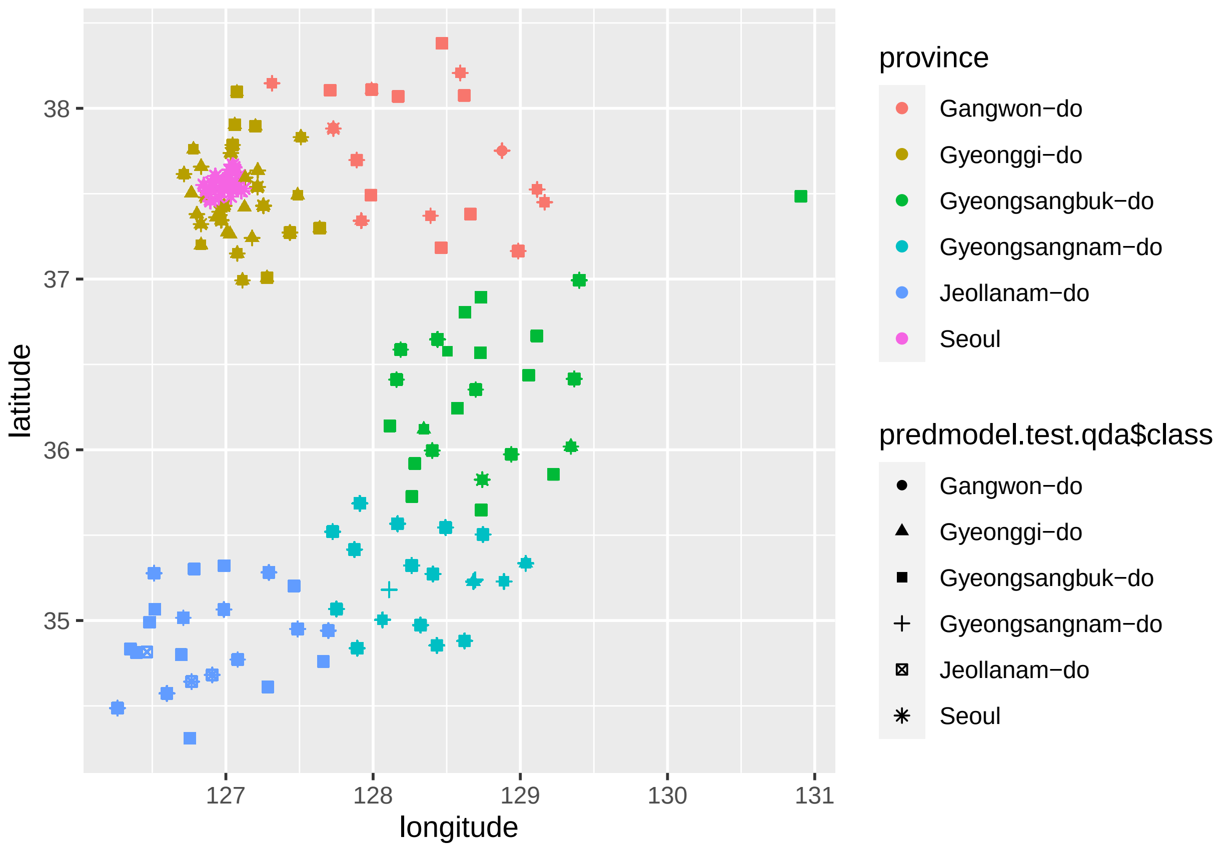
```
## QDA
train_error_qda
```

```
## [1] 0.1229627
```

```
test_error_qda
```

```
## [1] 0.1240274
```

```
## QDA
p218
```



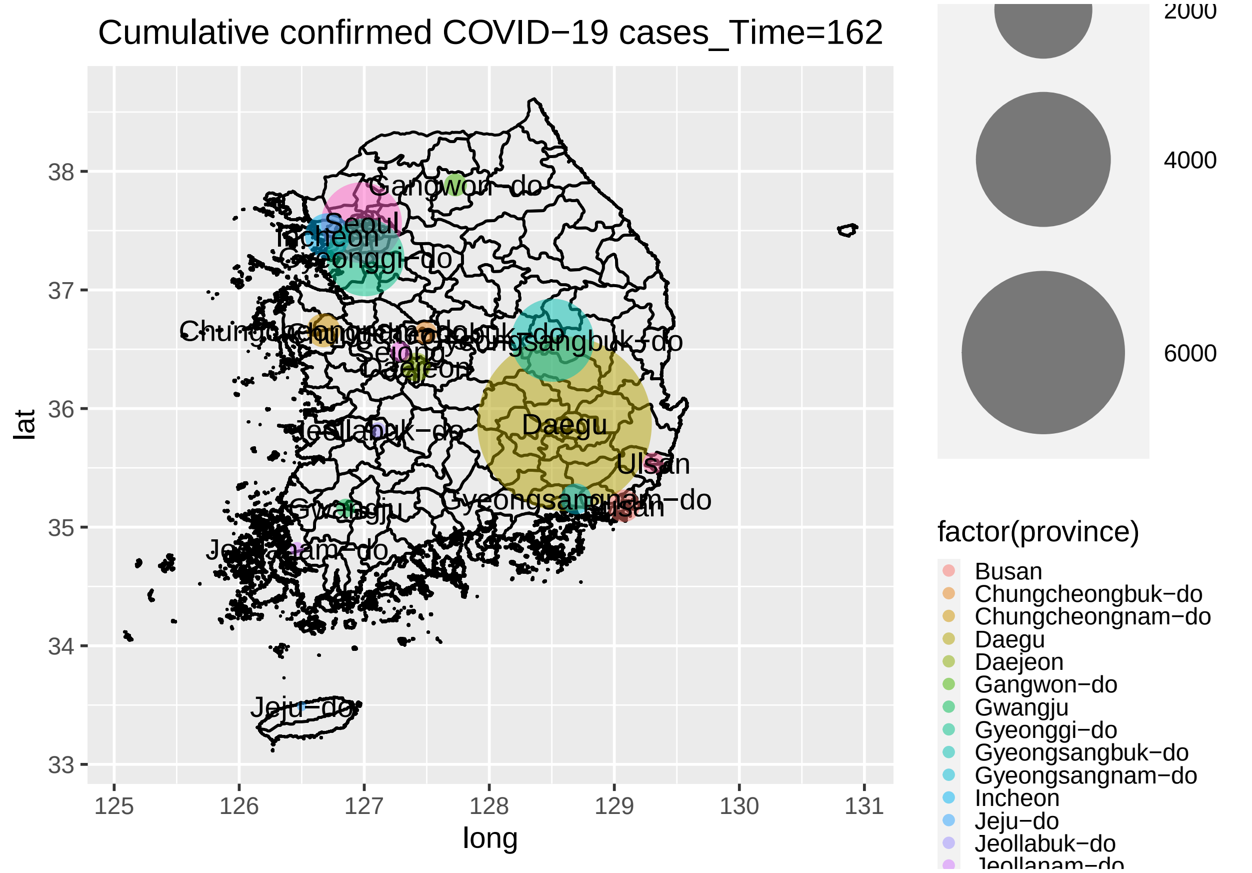
By using the same dataset with LDA, we get smaller training error and test error than LDA. QDA (Quadratic Discriminant Analysis) is used to find a non-linear boundary between classifiers. Therefore, in this case, QDA is better than LDA.

Data Cleaning

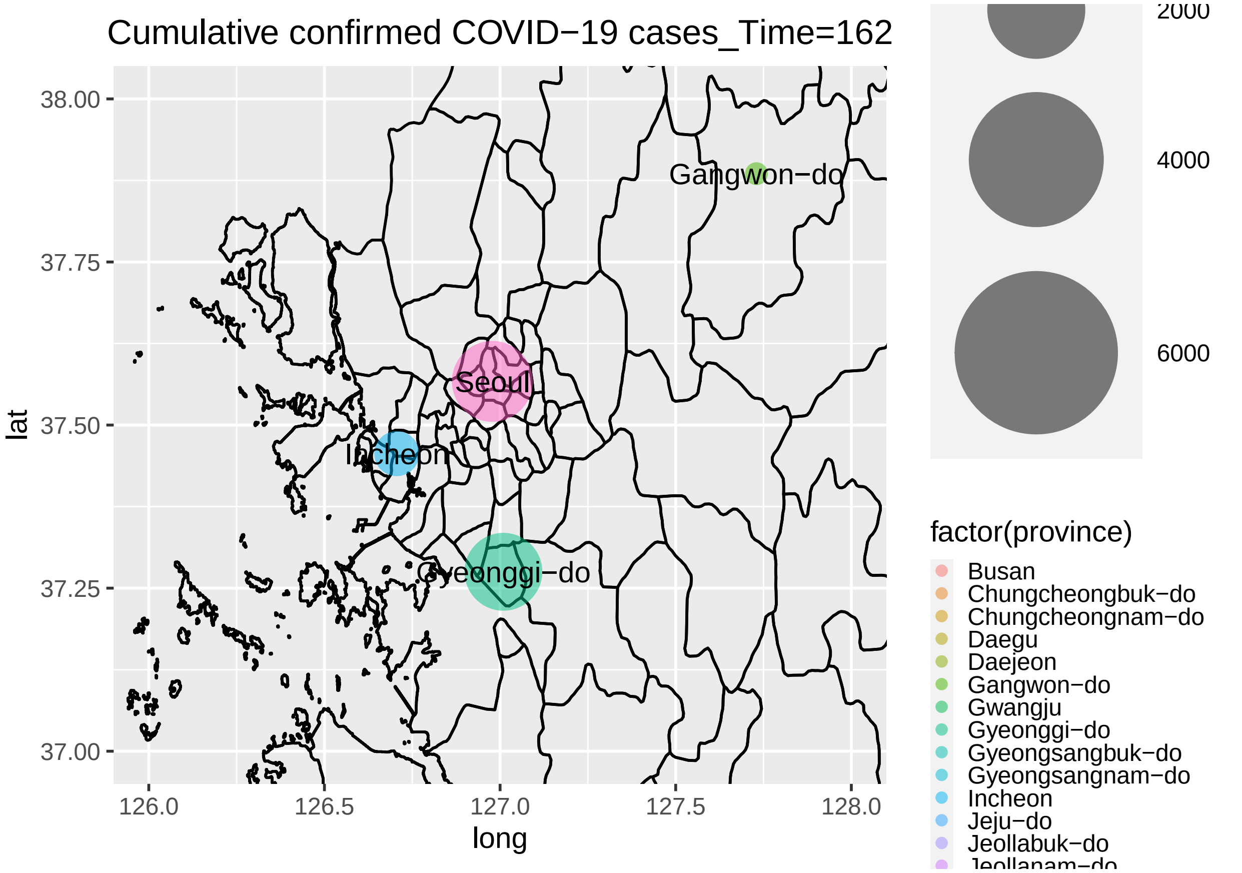
The data sets given by Kaggle are raw data set and need to do data cleaning before doing any model selection and prediction.

First, I group the raw data of Region, Search Trend, Time Province, and Weather by province and focus on Seoul, Incheon, and Gyeonggi Do. I decided to focus on Seoul, Incheon, and Gyeonggi Do in that these three provinces have similar characteristics of weather and cases trend.

```
P4
```

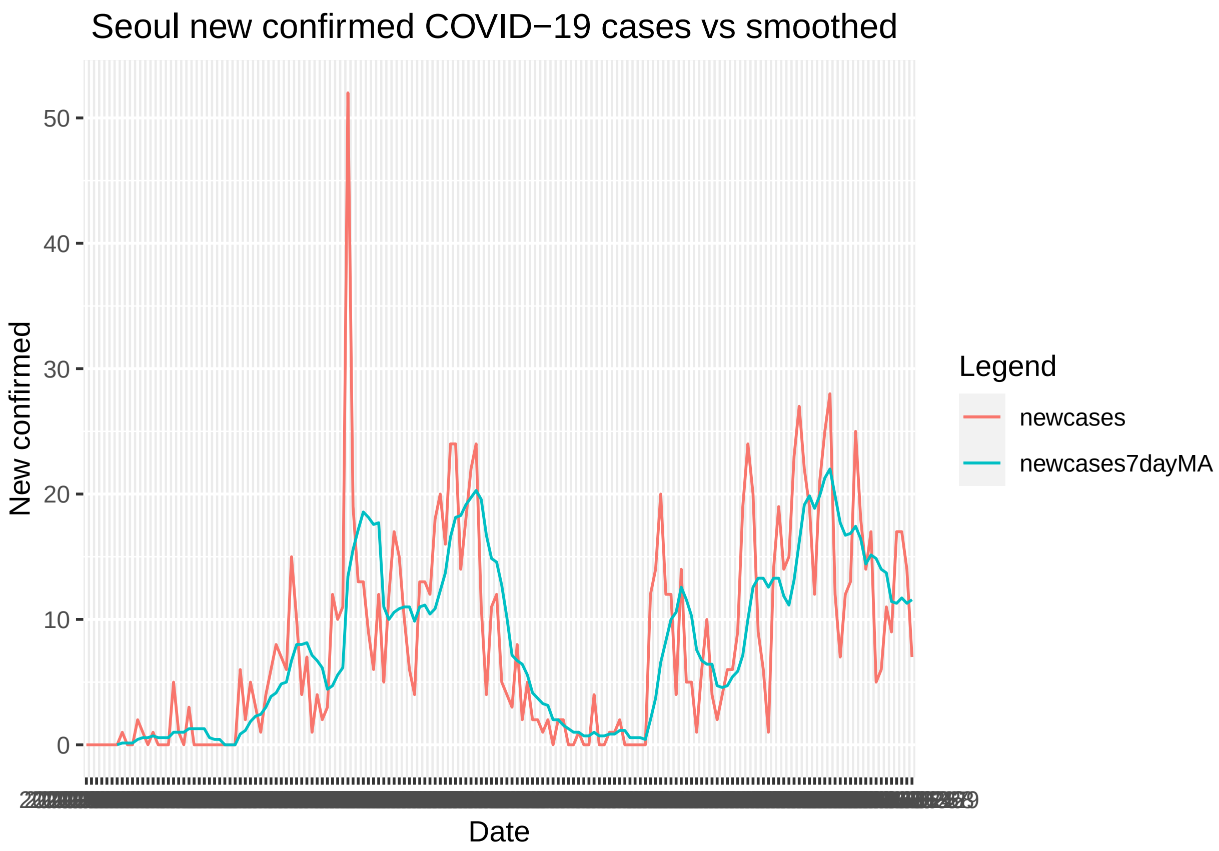


```
P45
```



Second, to handle the response variable cumulative confirmation of COVID cases with autoregressive Errors, I decided to apply the first differences procedure first. We will have new confirmed cases as the response. With the new confirmed cases, I smoothed it via MA7 to remove some noise peaks that can cause underestimation of new confirmed cases on the other days besides the peak date. In the end, our final response variable is Y_t 7 day moving average of new confirmed cases.

```
P18
```



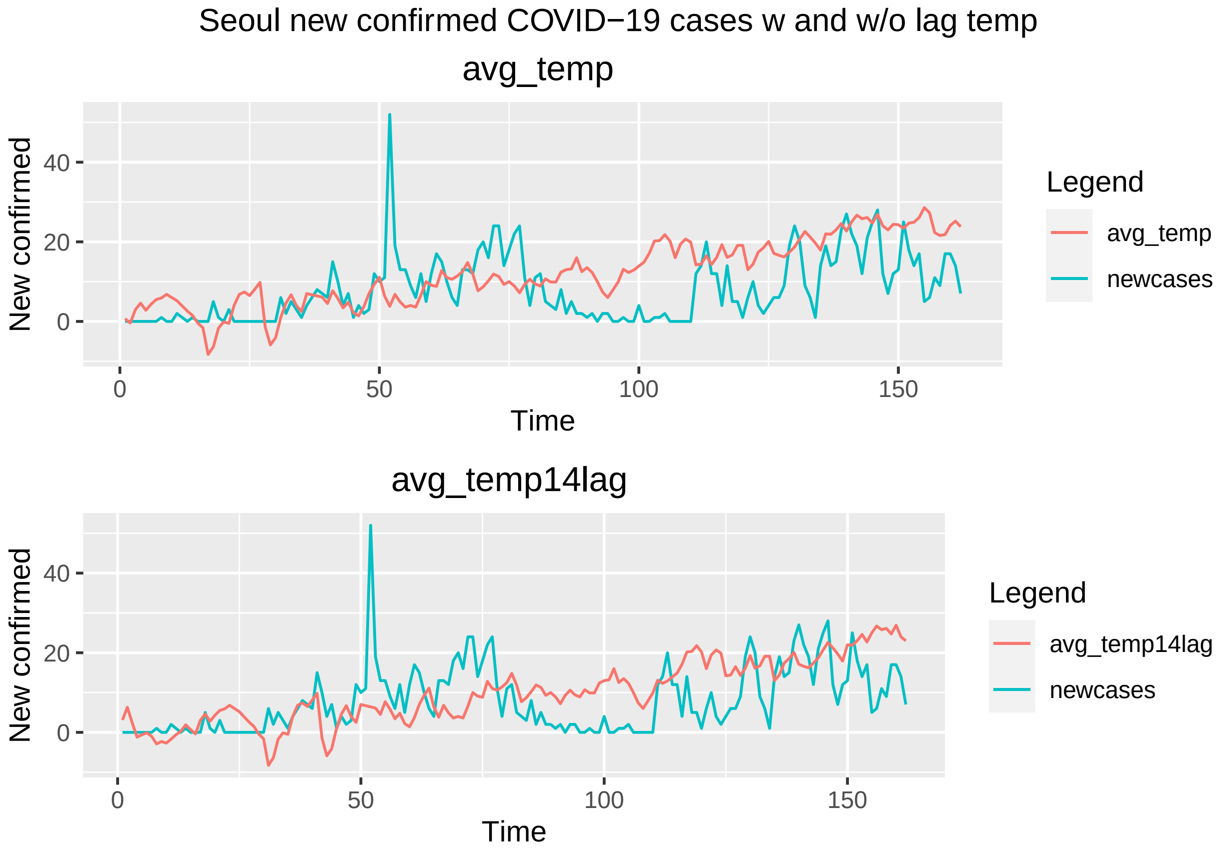
For the predictor part, we have four main predictor components:

1. Y_{t-7} : 7 Day Moving Average of New Confirmed case 7 day ago
2. Weather
3. Policy
4. Search Trend

To predict the 7 Day Moving Average of New Confirmed case today, it is very useful to use the 7 day moving average of new confirmed case 7 day ago because Covid-19 is an infectious disease.

To handle weather and policy, I decided to put a 14 days lags to accommodate for the incubation period of Covid-19. The incubation period of Covid-19 can be as short as five days and as long as twenty-one days. For example, the new case on 1/20/21 is corresponding to the weather on 1/06/21. Below is an example of Seoul's new confirmed cases with lagged average temperature. We can observe the opposite spike at around time 20,120, etc. But we can also observe the same direction spike at around time 35. So, that being said, the temperature might not be advantageous in predicting new confirmed cases.

```
grid.arrange(P24,P26,top = textGrob("Seoul new confirmed COVID-19 cases w and w/o lag temp",gp = gpar(fontsize = 12
```


Seoul new confirmed COVID−19 cases w and w/o lag temp

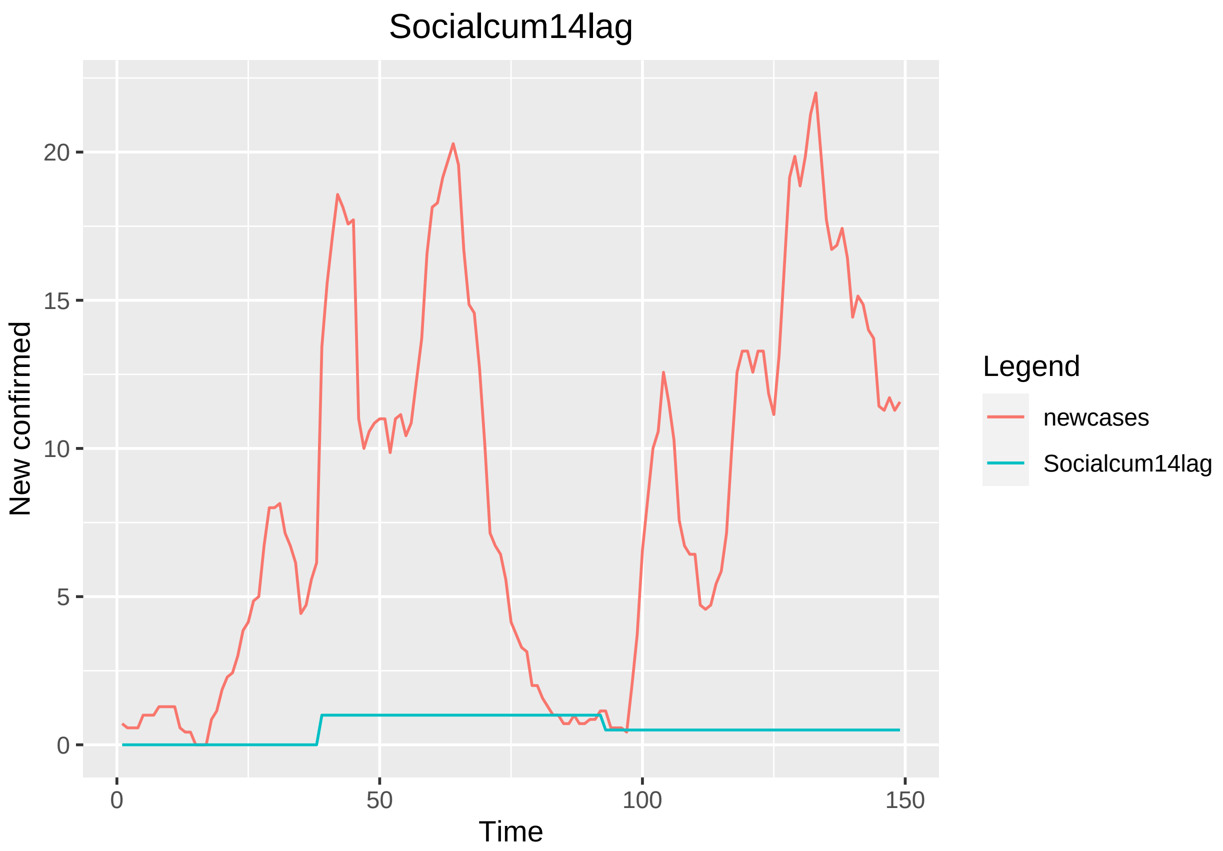
avg_temp

avg_temp14lag

For policy, I count it as cumulative number of new policies started with the corresponding date. When a policy started, I put a +1 to the count. When a policy is weaken or ended, I put -0.5 and -1 respectively.

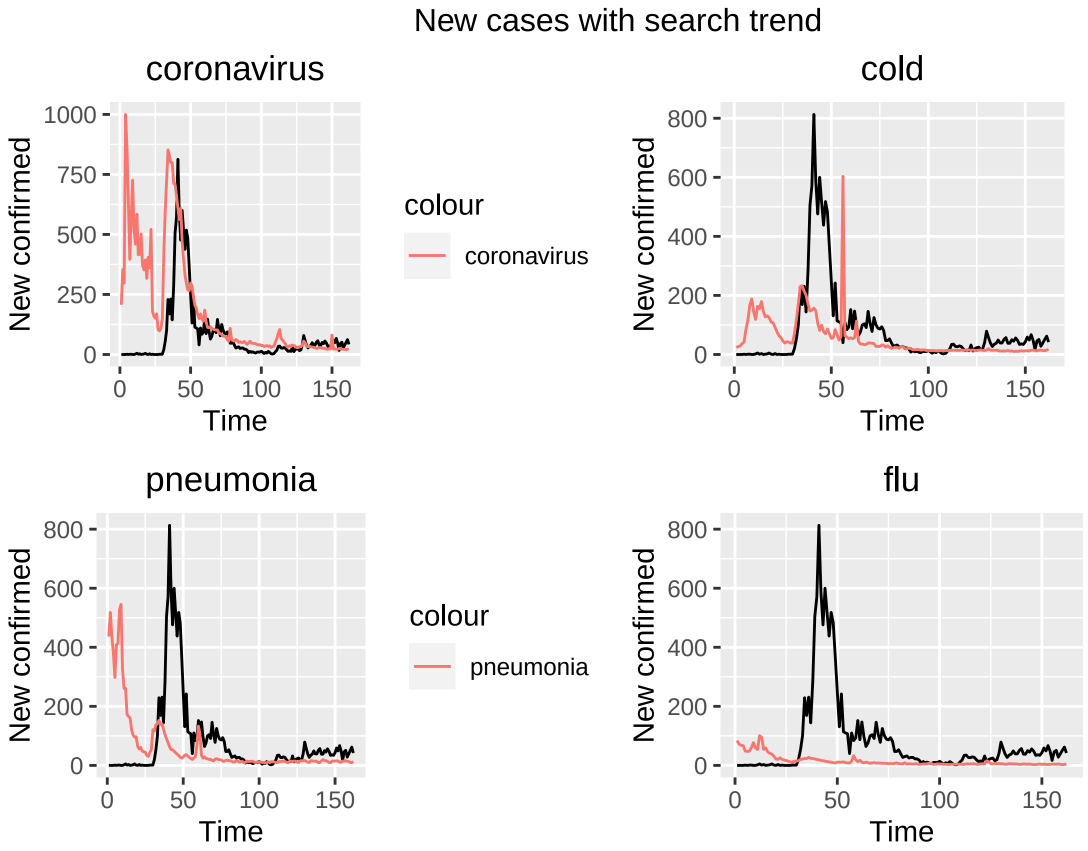
For example: Social Distancing Campaign Policy 1. 14 day lag of social policy with corresponding date. 2. Period of policy imposed, case start going down. 3. Once the policy have weaken, the cases climbed up again.

```
P27
```



Lastly, for search trends, there are two interpretations about it. The first interpretation is that when people feel sick, they will search coronavirus, cold, etc., for similar symptoms. For this first case, the search trend should lag for few dates to accommodate when a person needs to get tested and get their diagnostic result. The second interpretation is that News' content scale about Covid-19 cases has a positive association with the number of new cases per day. This means that the higher number of new confirmed implies more News channel will talk about Covid-19 suggests more search by people. Based on the trend plot between the word search of coronavirus and the new confirmed case. I decided to model search trend as the latter case.

```r
grid.arrange(P8,P9,P10,P11,nrow=2,top = textGrob("New cases with search trend",gp = gpar(fontsize = 12)))
```



Analysis Pipeline

1. Split the whole data set into training set and test set, then split the training set into training set and validation set. We don't want to use the test set until the last step.
2. Fit the model with different tuning parameters by using regression methods.
3. Find the best parameter with the smallest error by using validation data.
4. Refit the model on training set with smallest tuning parameters.
5. Fit the models on test set and compute the difference between the observed value and predicted value in order to get the final test error.

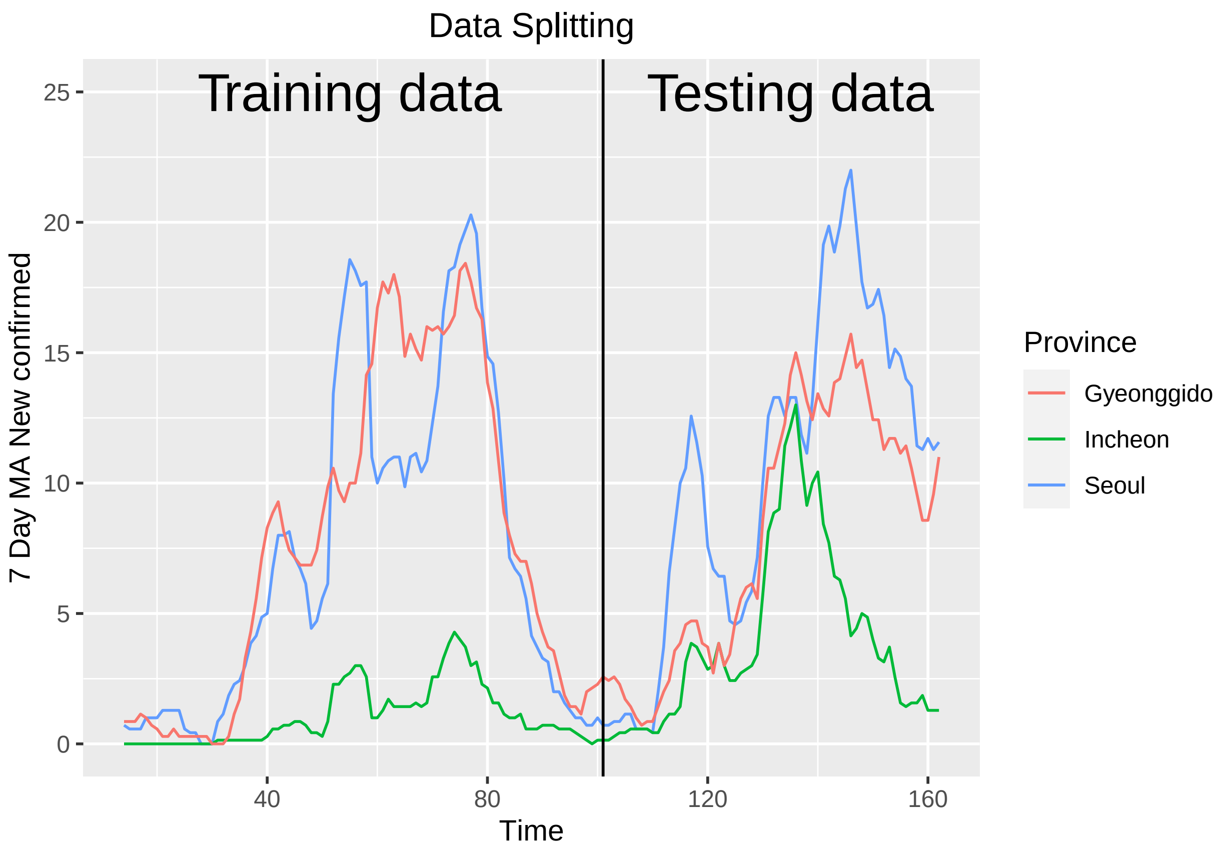
The criterion we used is mean absolute error(MAE), which is a measurement of errors between paired observations expressing the same phenomenon.

Data Spliting

1. First 100 days of these three provinces as training. (Total 300 observations)
2. Split again randomly as training set and validation set. (Each 150 observations)
3. Last 50 days of these three provinces as testing set. (Total 150 observations)

Overall, we did a one-third, one-third, one-third split to the data into training set, validation set and testing set . But will not use the "future data" as training observations.

```
TrainandTestplot
```



Model Selection

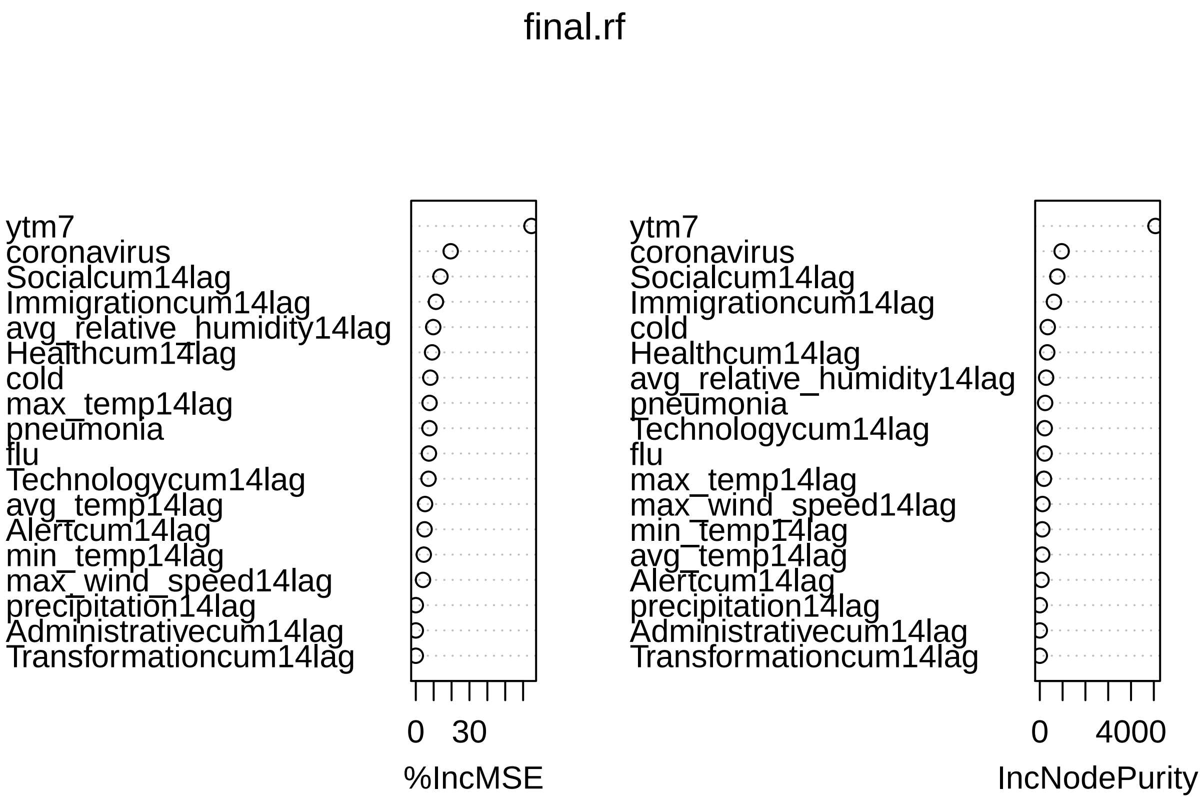
We use the training set to do the model selection.

1. Random Forest (Scaled)

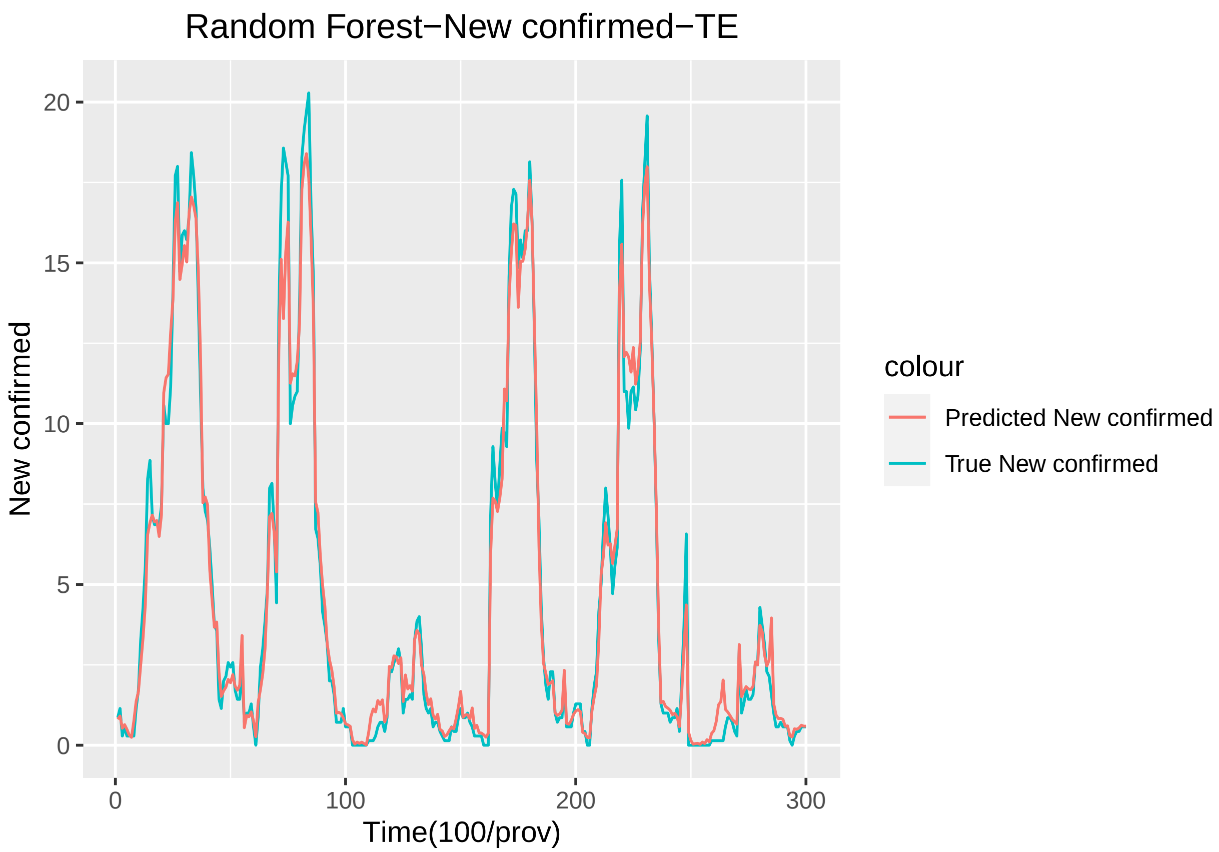
I use random forest because a single decision tree is not good at new sample prediction and is likely to overfit. Also, the data set is small, it won't take a long time to run, and we can get high accuracy.

When fitting a random forest, I started with mtry $= \frac{p}{3}$, which is around 8. For validation, I set the "mtry" plus and minus 2, and started at 50 trees to 300 trees. After using the validation set, I picked 300 as the optimal tree number and 5 as the optimal mtry. Since I was doing a regression tree, I set the nodesize = 5.

```
RFimplor=varImpPlot(final.rf)
```



```
TERFP
```



From the graphs above, we can see that the important measures of ytm7 and coronavirus are higher than other predictors, which are the top two most-picked nodes from tree building, and we got a high accuracy from random forest dealing with training data.

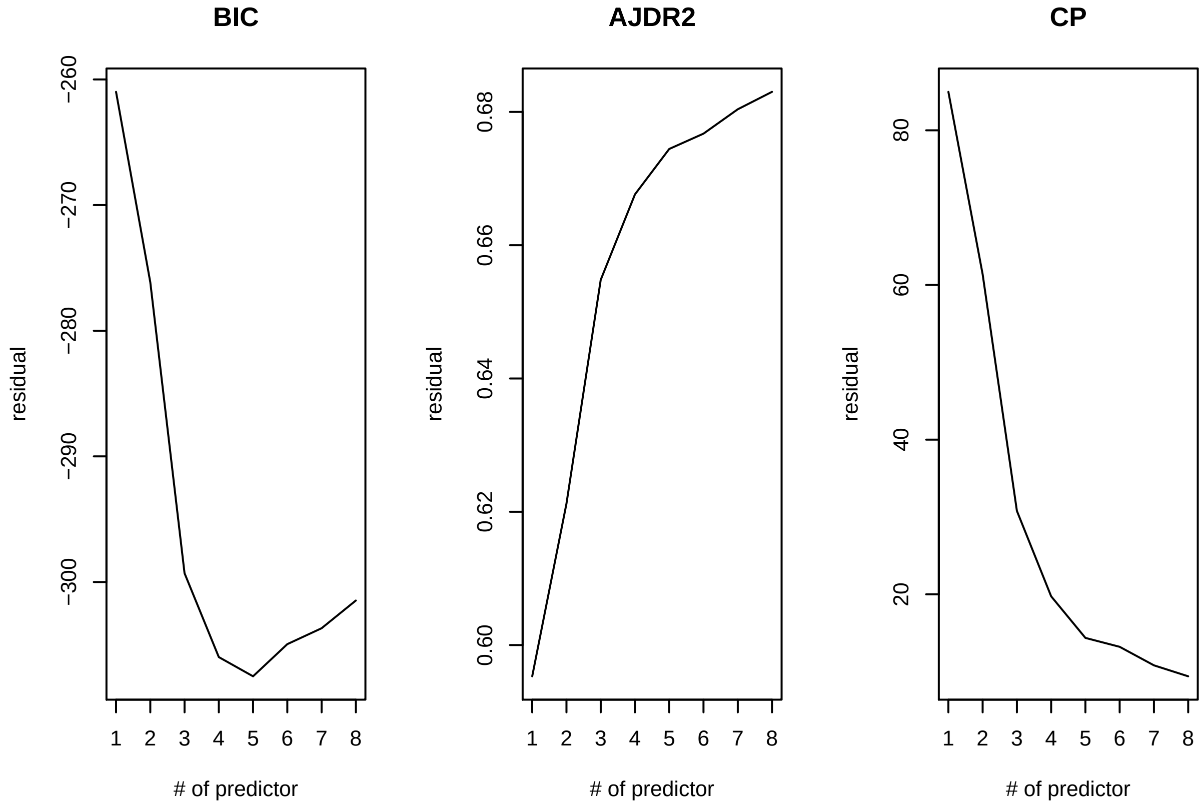
2. Linear Regression (OLS) and OLS with best subset (Scaled Invariant)

I fitted an OLS model by using all the parameters. We can optimize the OLS model by doing the best subset.

```
models = regsubsets(newcases7dayMA~., data = svmBigTrain)
summary(models)
```

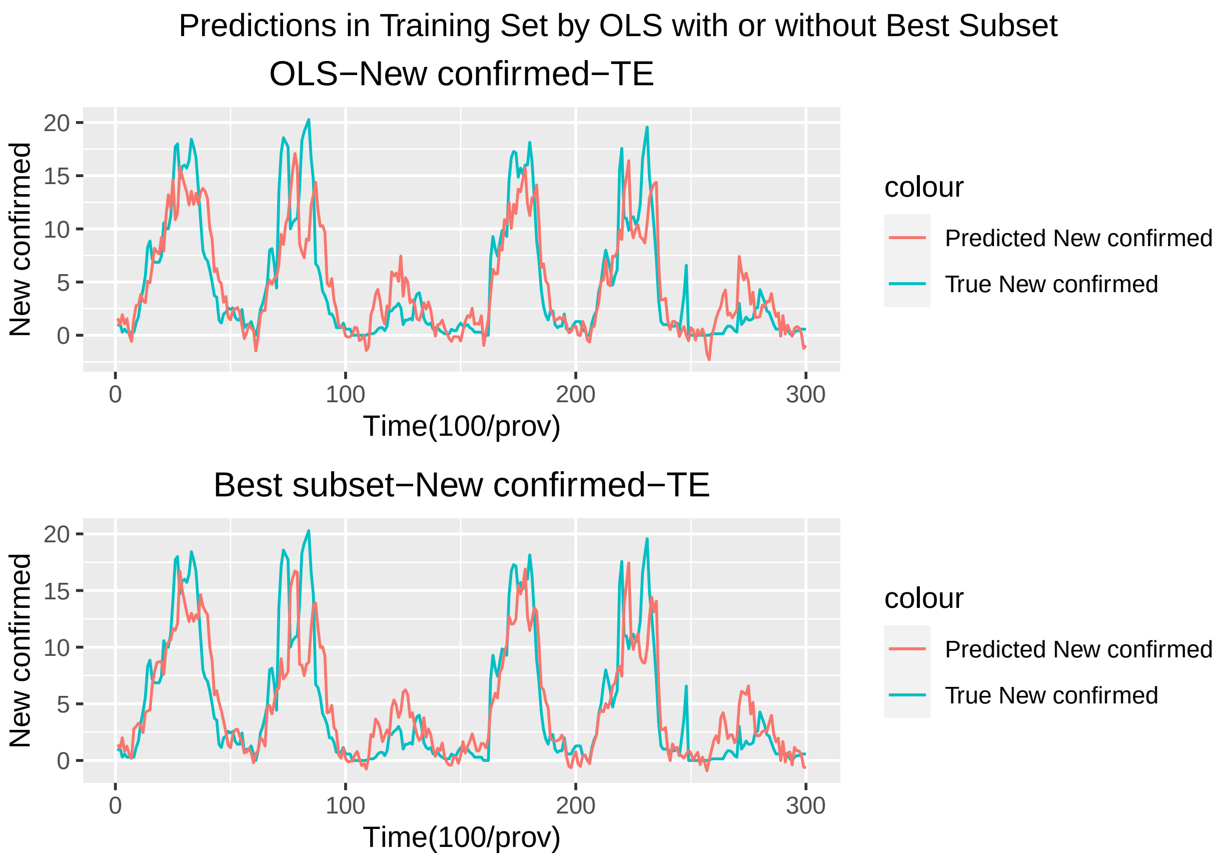
```
## Subset selection object
## Call: regsubsets.formula(newcases7dayMA ~ ., data = svmBigTrain)
## 15 Variables  (and intercept)
##                            Forced in Forced out
## avg_temp14lag                  FALSE      FALSE
## min_temp14lag                  FALSE      FALSE
## max_temp14lag                  FALSE      FALSE
## max_wind_speed14lag            FALSE      FALSE
## avg_relative_humidity14lag     FALSE      FALSE
## cold                           FALSE      FALSE
## flu                            FALSE      FALSE
## pneumonia                      FALSE      FALSE
## coronavirus                    FALSE      FALSE
## Alertcum14lag                  FALSE      FALSE
## Immigrationcum14lag            FALSE      FALSE
## Healthcum14lag                 FALSE      FALSE
## Socialcum14lag                 FALSE      FALSE
## Technologycum14lag             FALSE      FALSE
## ytm7                           FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: exhaustive
##          avg_temp14lag min_temp14lag max_temp14lag max_wind_speed14lag
## 1  ( 1 ) " "           " "           " "           " "
## 2  ( 1 ) " "           " "           " "           " "
## 3  ( 1 ) " "           " "           " "           " "
## 4  ( 1 ) " "           " "           " "           " "
## 5  ( 1 ) " "           " "           " "           " "
## 6  ( 1 ) " "           " "           " "           " "
## 7  ( 1 ) " "           " "           " "           " "
## 8  ( 1 ) " "           " "           " "           " "
##          avg_relative_humidity14lag cold flu pneumonia coronavirus
## 1  ( 1 ) " "                        " "  " " " "       " "
## 2  ( 1 ) "*"                        " "  " " " "       " "
## 3  ( 1 ) " "                        " "  " " " "       " "
## 4  ( 1 ) "*"                        " "  " " " "       " "
## 5  ( 1 ) "*"                        " "  " " " "       "*"
## 6  ( 1 ) "*"                        " "  " " "*"       "*"
## 7  ( 1 ) "*"                        " "  " " "*"       "*"
## 8  ( 1 ) "*"                        " "  "*" "*"       "*"
##          Alertcum14lag Immigrationcum14lag Healthcum14lag Socialcum14lag
## 1  ( 1 ) " "           " "                 " "            " "
## 2  ( 1 ) " "           " "                 " "            " "
## 3  ( 1 ) " "           "*"                 "*"            " "
## 4  ( 1 ) " "           "*"                 "*"            " "
## 5  ( 1 ) " "           "*"                 "*"            " "
## 6  ( 1 ) " "           "*"                 "*"            " "
## 7  ( 1 ) " "           "*"                 "*"            " "
## 8  ( 1 ) " "           "*"                 "*"            " "
##          Technologycum14lag ytm7
## 1  ( 1 ) " "                "*"
## 2  ( 1 ) " "                "*"
## 3  ( 1 ) " "                "*"
## 4  ( 1 ) " "                "*"
## 5  ( 1 ) " "                "*"
## 6  ( 1 ) " "                "*"
## 7  ( 1 ) "*"                "*"
## 8  ( 1 ) "*"                "*"
```

```r
par(mfrow=c(1,3))
plot(res.sum$bic,type="l",main="BIC", xlab="# of predictor", ylab="residual")
plot(res.sum$adjr2,type="l",main="AJDR2", xlab="# of predictor", ylab="residual")
plot(res.sum$cp,type="l",main="CP", xlab="# of predictor", ylab="residual")
```



The plots above show that when the number of predictors is 4 or 5, we get the lowest BIC. In the graphs of adjusted R^2 and Mallow's C_p, the "elbow" happens when the number of predictors equals to 5. Therefore, I chose ytm7, coronavirus, avg_relative_humidity14lag, Immigrationcum14lag, and Healthcum14lag as the predictors in the OLS model after best subset. Recall that ytm7 and coronavirus were also chosen by random forest. Then I compared the OLS with or without best subset by graph below.

```
grid.arrange(TEOLSP,TEbestsubsetP,nrow=2,top = textGrob("Predictions in Training Set by OLS with or without Best Su
```

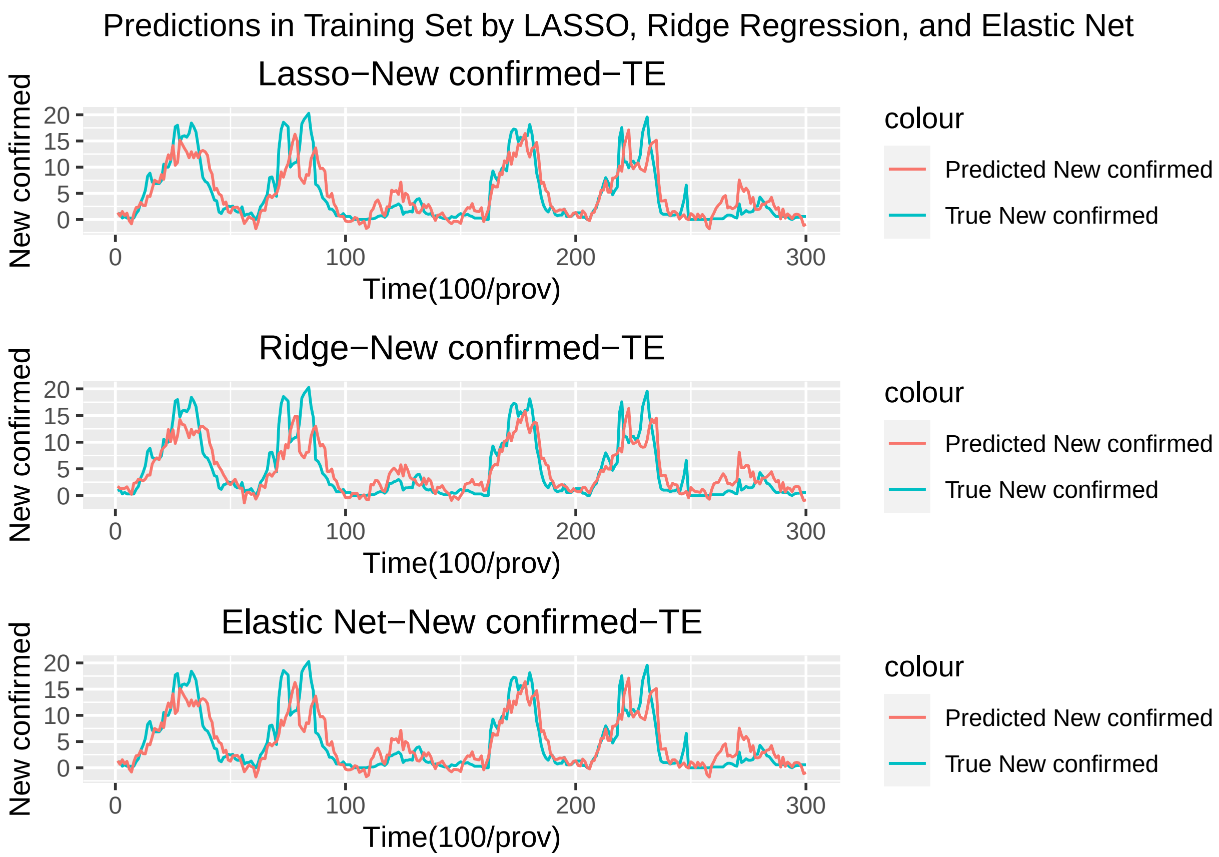


Compared these two graphs and based on the performance shown on the right-hand side of the graphs, the OLS after best subset may has a better prediction in training set. This is because the OLS with best subset selection improves out-of-sample accuracy of the regression model by eliminating unnecessary predictors, and yields a simple and easily interpretable model.

3. Elastic Net(lasso,ridge or mixed) (Scaled)

Then I tried LASSO, ridge regression, and elastic net by using scaled data.

```
grid.arrange(TELASP,TERIDP,TEENETP,nrow=3,top = textGrob("Predictions in Training Set by LASSO, Ridge Regression, a
```



It is hardly to claim which one is better by these graphs.

4. SVM with different kernel function (Scaled)

I also tried SVM with different kernel functions, which are linear, polynomial, and radial.

```
ValidationErrorBox=data.frame(
  svmlinearVE,svmpolyVE,svmradialVE
)
ValidationErrorBox
```

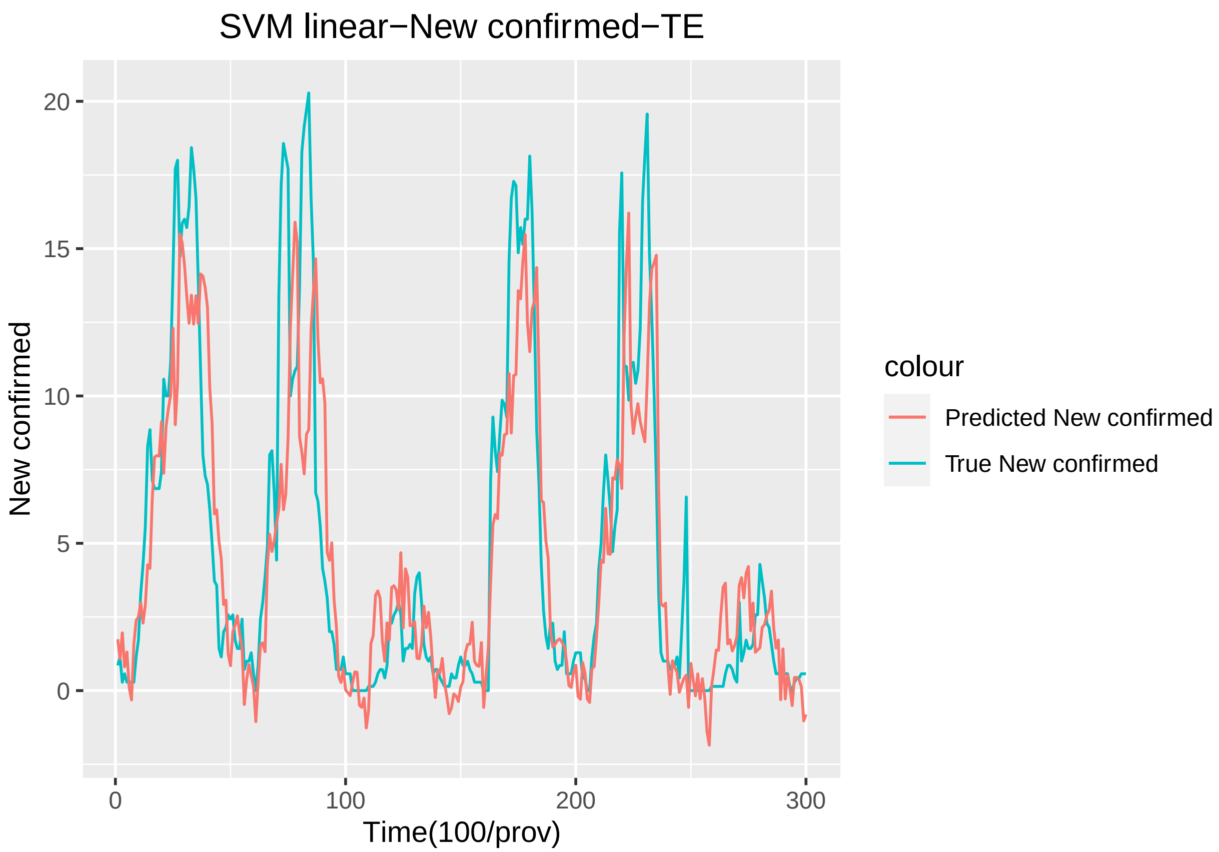
```
##   svmlinearVE svmpolyVE svmradialVE
## 1    1.935991  2.504752    1.949734
```

```
which.min(ValidationErrorBox)
```

```
## svmlinearVE
##           1
```

Then I compared their validation errors, and got the lowest one, which is SVM linear. Therefore, I chose SVM linear for future use.

```
TESVMLP
```



The prediction in training set by SVM Linear is more adequate than other algorithms. This may because SVM is a regularisation parameter, it can help us aviod overfitting.

5. Training Error Table

```
TrainingErrorBox=data.frame(
  TERF,TEOLS,TERID,TELAS,TEENET,TESVML,TEbestsubset
)
TrainingErrorBox
```

```
##        TERF    TEOLS   TERID    TELAS   TEENET   TESVML TEbestsubset
## 1 0.5177971 2.184255 2.26555 2.192835 2.192835 2.133451     2.181862
```

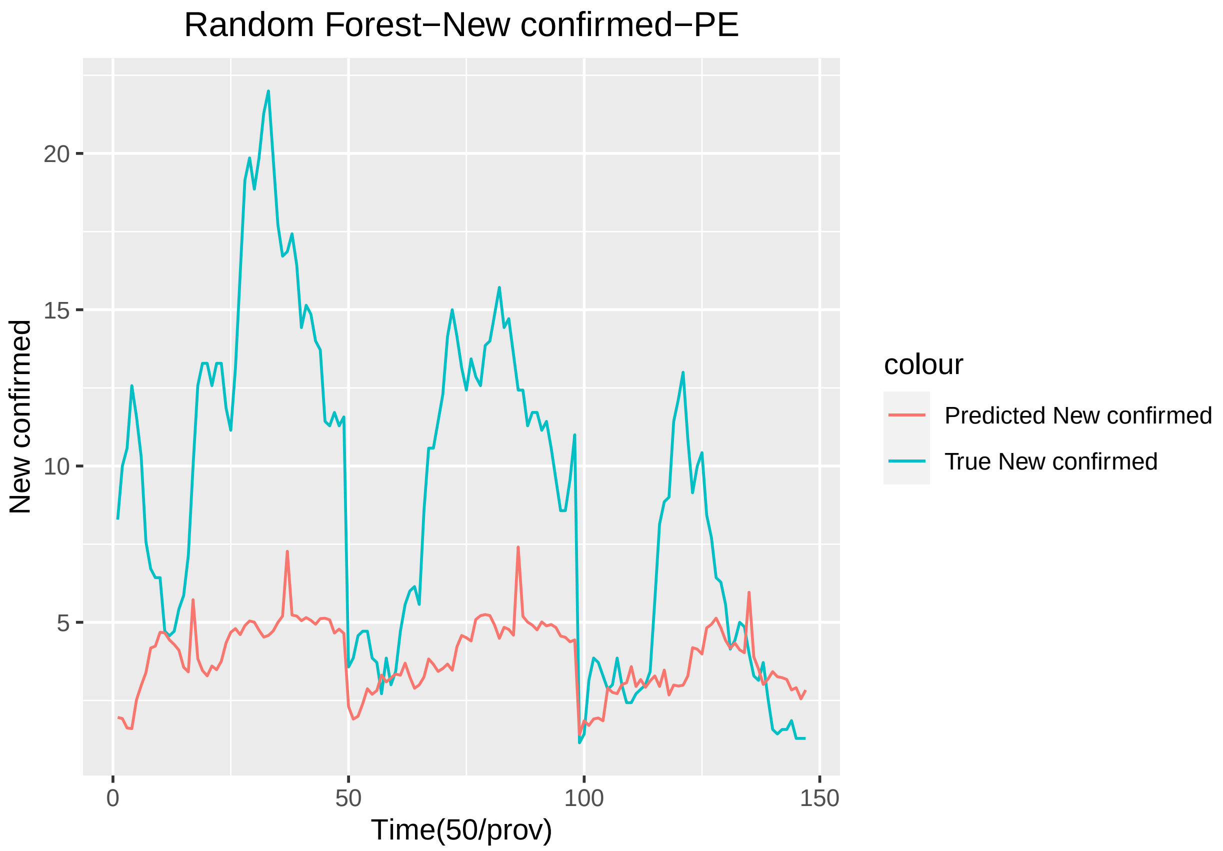
We can see that random forest method has the lowest training error, and other algorithms got similar values. However, it is normal that random forest has a high accuracy in training set. We still need to see how the prediction goes in testing set. The LASSO and elastic net got the same training error because the optimal $\alpha = 1$.

Prediction in testing set

We use the testing data to predict new confirmed cases.

1. Random Forest (Scaled)

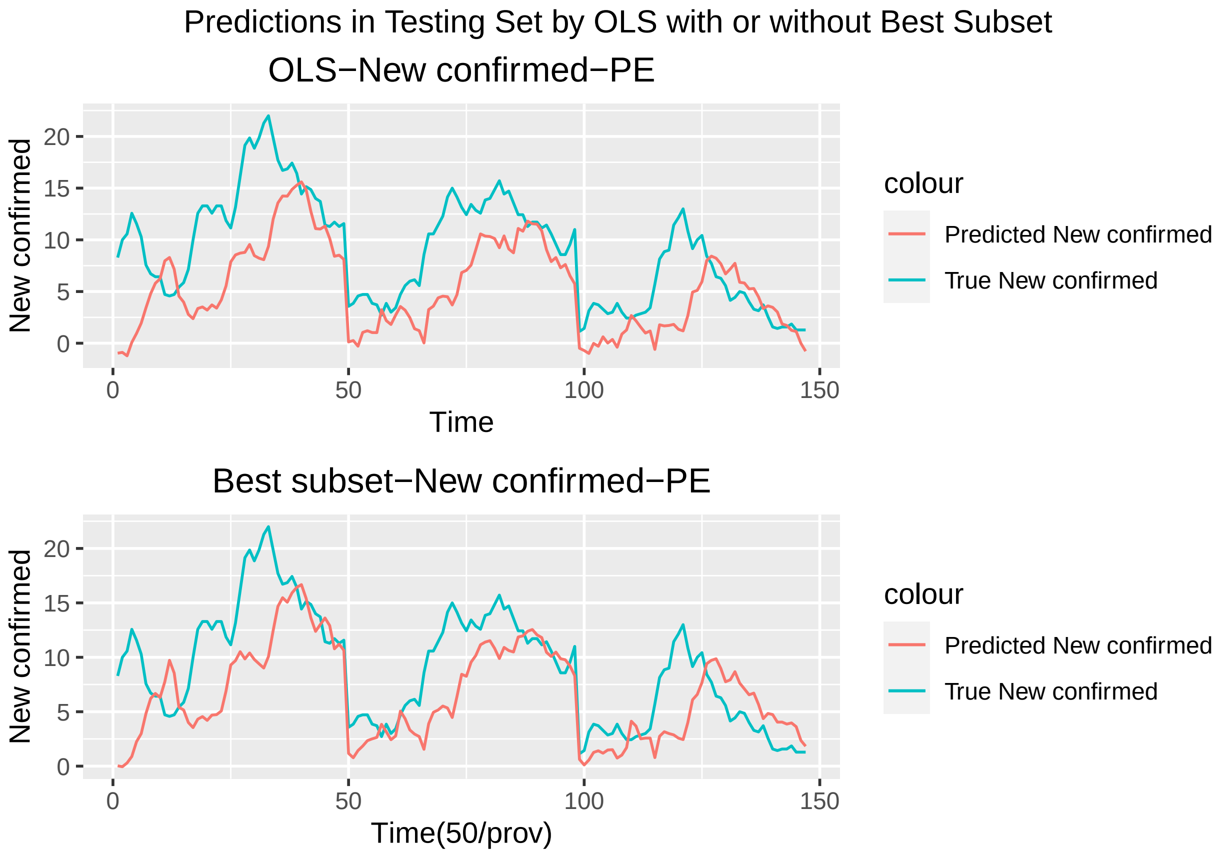
```
RFPEP
```



In prediction using training set, random forest did a good job in modeling the trend of new confirmed case, but it did a much worse job in predicting the actual number of new confirmed cases in testing data. This happens because the testing set is extrapolation. When we use decision trees or random forest to predict, they cannot deal with outside scope.

2. Linear Regression (OLS) and OLS with best subset (Scaled Invariant)

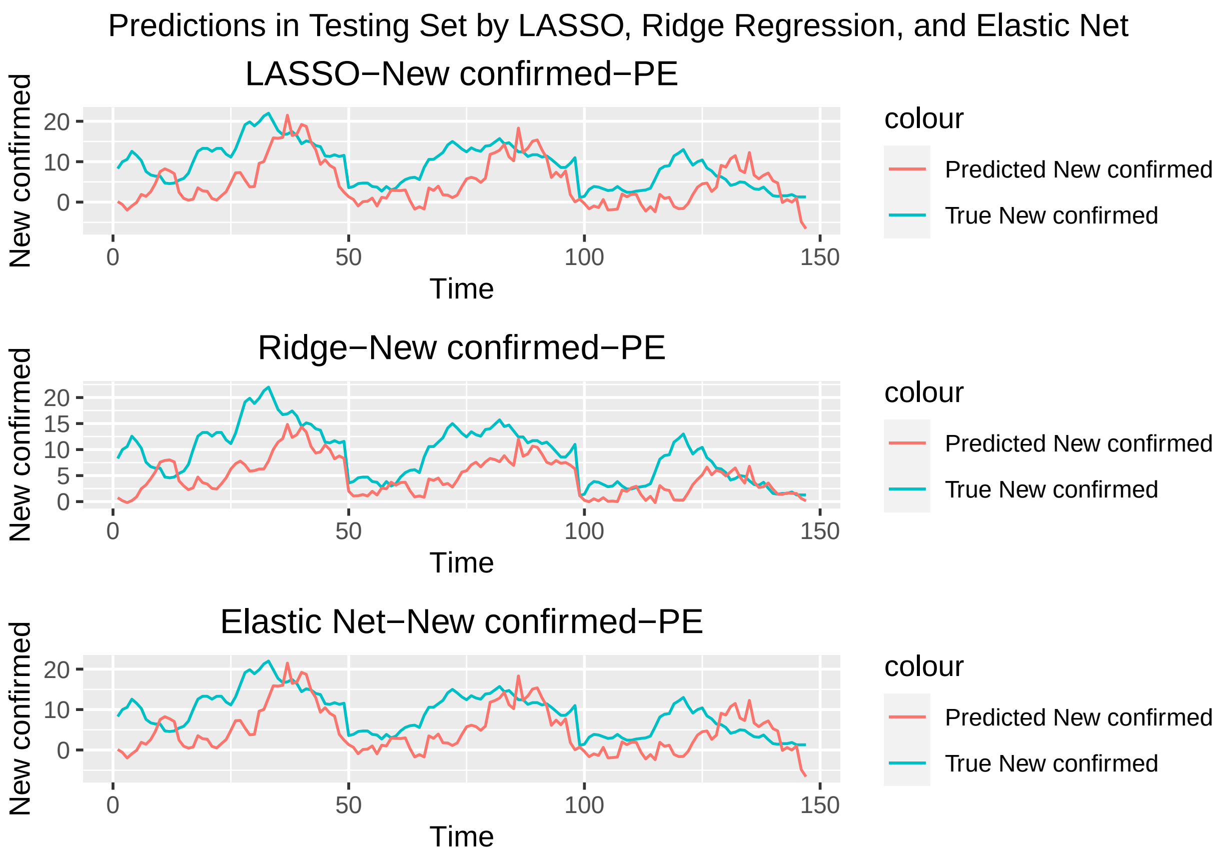
```
grid.arrange(OLSEP,bestsubsetPEP,nrow=2,top = textGrob("Predictions in Testing Set by OLS with or without Best Subs
```



The predicted new confirmed cases' line in the graph of OLS with best subset looks smoother and more tend to the true new confirmed cases' line.

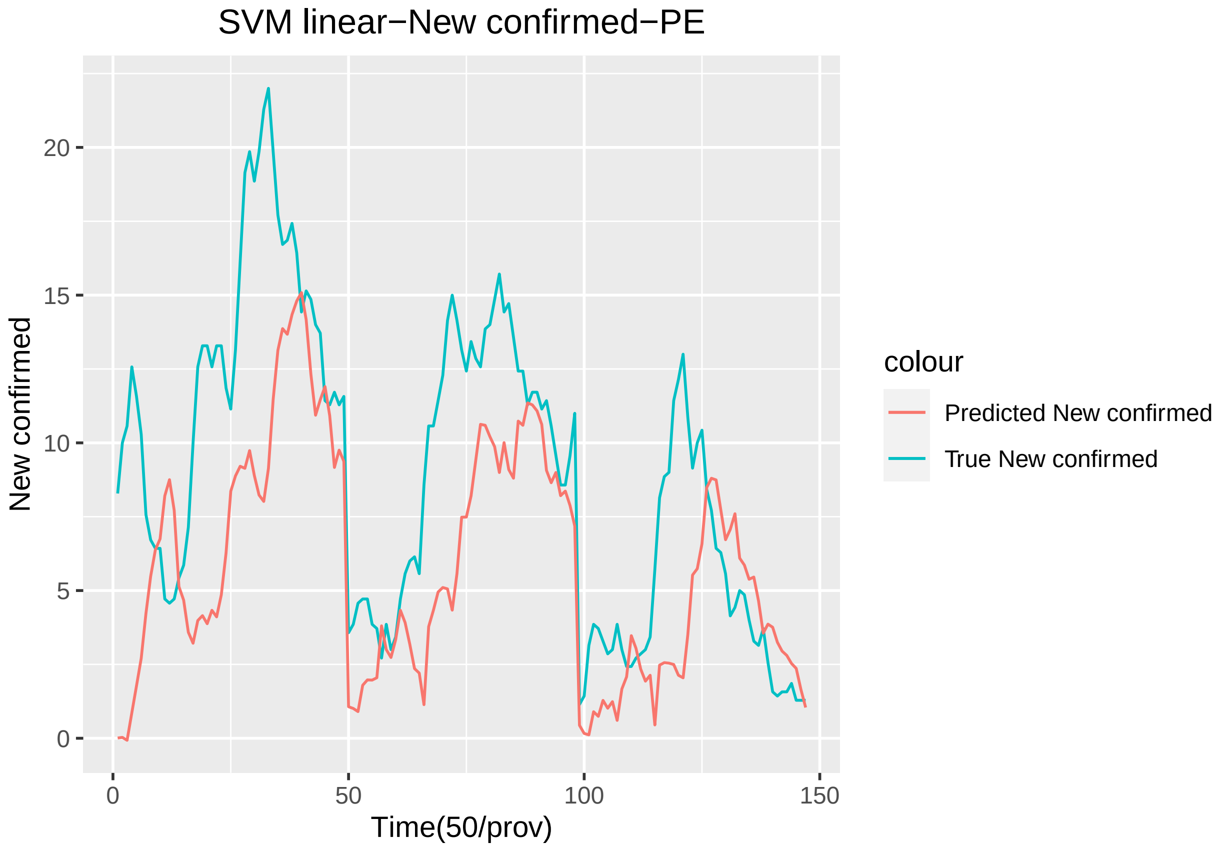
3. Elastic Net(lasso,ridge or mixed) (Scaled)

```
grid.arrange(lassoPEP,ridPEP,elasticPEP,nrow=3,top = textGrob("Predictions in Testing Set by LASSO, Ridge Regressio
```



4. SVM with different kernel function (Scaled)

```
svmlinearPEP
```



5. Prediction Error Table

```
PredictionErrorBox=data.frame(
  RFPE,OLSE,ridPE,lassoPE,elasticPE,svmlinearPE,bestsubsetPE
)

PredictionErrorBox
```

```
##       RFPE     OLSE    ridPE  lassoPE elasticPE svmlinearPE bestsubsetPE
## 1 5.420197 4.210133 4.554649 5.547614  5.547614     3.81019     3.574929
```

We can see that the ordinary least square after the best subset has the lowest prediction error. The random forest method has the highest prediction error because it usually does an excellent job at classification but not for regression problems. It does not give precise continuous nature prediction. In the case of regression, it doesn't predict beyond the range in the training data and may overfit particularly noisy datasets. We have very little control over our model when we use the random forest method. The prediction errors of LASSO, elastic net, and ridge regression are also very high. For LASSO and elastic net, this may happen because these two algorithms cannot make group selection. LASSO and elastic net tend to just pick one predictor out of the group when there are highly correlated predictors. For ridge regression, this may happen because it includes all the predictors in the model and cannot make prediction selection; it shrinks the coefficients towards zero and trades the variance for bias.

Conclusion

In conclusion, based on the performance of all the algorithms, we think that the SVM linear and OLS with the best subset did a great job, OLS and ridge regression have secondary performance, and random forest, LASSO, and elastic net are the worst. That is because SVM Linear with error tolerance can avoid overfitting, and OLS with the best subset removed multicollinearity and useless predictors. Ridge Regression can also handle multicollinearity, but it gets many noise predictors. Random Forest cannot deal with outside scope, Lasso cannot handle highly correlated data, and Elastic net comes out to choose $\alpha = 1$ from validation.

The overall performance of all the algorithms is not ideal. The reasons can be as follow. First, population data is not included in those Kaggle datasets. The rates of spread in provinces with large populations would be higher than those who have small populations. The population should be one of the factors in our prediction. Second, the Covid-19's incubation time is not stable. It can be as short as 5 days and as long as 21 days. This may cause different levels of postponement in the counts of new confirmed cases. Third, the past data do not lead to a good representation of future data because Covid-19 is an epidemic disease. The new cases can be contributed many unknown variables, such as the Covid-19 variant. Different types of Covid variant have different chances of infection, vulnerable populations, etc., which cannot be predicted.

In future work, to improve the accuracy of our model, we need to consider the rate of spread by operating the population data of each province. We only used three provinces' data in this project. We can also improve our prediction by containing the data of all the provinces in Korea. The more comprehensive data we use, the more accurate the model will be.