Being Master Chefs to Fight Against COVID-19

OCTOBER 29, 2020

STAT4011 Project I

Authored by:

King Yeung, CHAN 1155119394

1. Introduction

1.1. Background

"SOCIAL DISTANCING" maybe is the most common phrase that we have heard during the ages of quarantine. More and more people went over the kitchen and started their new pages on being Master Chefs in their own lives. To capture the current circumstances quickly, we are going to visualise the overall trends and investigate the correlation between the novel coronavirus and the wholesale prices of fresh food in Hong Kong. We anticipate the outcome can address "the social issue" that happens in Hong Kong.

At the late end of 2019, the novel coronavirus, also known as COVID-19, has become an epidemic and communicable diseases around the world. Remote learning and working have become a common approach to maintain social distancing. Meanwhile, the Hong Kong Government (HKG) banned on dine-in services a couple of times. Cook at home becomes a popular activity to kill time during the months of isolation. As we are staying at home, we may only rely on the information available on the internet. However, fake news, untrue information presented as news, has become a critical issue in Hong Kong. During the pandemic of the novel coronavirus, there were rumours of a shortage of necessities in Hong Kong. That baloney led to panic buying and chaos. We propose to verify whether the change in cooking behaviour does lead to any possible in shortage or price changes in necessities. We hope the result can address the fake news problem in Hong Kong. In the project, consequently, we aim to raise public awareness on tackle disinformation.

1.2. Data Source

Since the thesis statement involves multiple topics, we are going to grab data from a variety of sources.

1. *Novel Corona Virus 2019 Dataset* from Kaggle provides information for visualising the past situation related to the novel coronavirus. Kaggle is famous for many competitions about Machine Learning and Data Science. The available dataset is organised on a daily basis from international organisations. Thus, we assume the provided data are dependable.
2. *Wholesale Prices of Major Fresh Food* from DATA.GOV.HK offers daily wholesale prices on different food categories. DATA.GOV.HK is a data repository organised by the HKG. Data are published to value the community for both commercial and non-commercial purpose. Hence, we consider the given data are error-free.
3. *Watching the Pandemic* from YouTube Culture & Trends summarises the data on people watching behaviour during the outbreak of the novel coronavirus. To secure the claims of causality, those data give us an insight onto the linkage between the novel coronavirus and the cooking behaviour.

2. Abstract

The line plot gives us an insight into the past situation about the novel coronavirus. The number of confirmed cases, deaths and recovered cases increased irregularly. It implies that the outbreak was unstable during the past nine months. There was a tremendous number of viewers from YouTube watching cooking tutorials. More and more people were exploring their talent in cooking. By constructing the Wilcoxon Sign-Rank test, it shows that there was a change in the median of the wholesale price compared to last year. However, the SRL models inform us that there were only 22% of the adjustment in wholesale prices explained by the number of confirmed cases. We may interpret such a phenomenon that there was a little effect on the wholesale price by the change in the number of confirmed cases. People have no reason to scare about the shortage or price changes under the pandemic. Moreover, the ES model predicts the outbreak will out of control again in the future. Here we suggest the public not to believe news from unreliable websites initially and think twice on it.

3. Exploratory Data Analysis

Before any further studies on the project, we give a rundown on the dataset we intend to use later, which also provides the principle of our data implementation.

3.1. Data Background

The raw data from Kaggle and DATA.GOV.HK are treated as time series data since each of those data was an observation measured at a successive point in time. Due to the limited number of available data from the websites, we only focus on data observed from 22 January to 30 September for both 2019 and 2020.

3.2. Missing Data and Data Cleansing

There are some missing data from *Wholesale Prices of Major Fresh Food* published from DATA.GOV.HK. As we assumed that data behave continuously, we replace the missing data to the available data from the next day. Moreover, the units vary on different types of food category. For the consistency of the project, we rescale all the unit into Catty per Hong Kong Dollars (HKD). Some rescaling approaches approximate to the unit. For example, we roughly convert 10 Eggs as 1 Catty per HKD here. We, furthermore, concatenate all the relevant variables into an out of the box data frame. (see Appendixes 10.1. for details)

3.3. Data Specification

Table 1 illustrates the description of data in terms of their metadata.

Variable	Type	Description
Date	Date	Date of a particular observation
Confirmed	Numeric	Number of confirmed patients in a particular day
Deaths	Numeric	Number of deaths in a particular day
Recovered	Numeric	Number of recovered patients in a particular day
Eggs	Numeric	Average wholesale price of Eggs from Mainland, Germany and USA
Freshwater fish	Numeric	Average wholesale price of grass carp, big head and mud carp
Livestock / Poultry	Numeric	Average wholesale price of pig, cattle and chicken
Marine fish	Numeric	Average wholesale price of golden tread, horse head, yellow croaker, big eyes, scads, breams, hairtail, mackerel, croakers and filefish
Vegetables	Numeric	Average wholesale price of flowering cabbage, white cabbage, Chinese lettuce, Chinese kale, European celery, potato, spinach, yard-long bean, broccoli, green cabbage, tomato and Chinese spinach

(*Table 1: metadata of variables*)

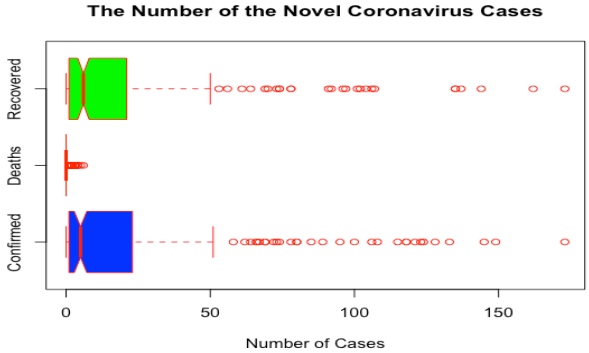
3.4. Summary of the Dataset

Table 2 summarises the statistics of variables of interest. The statistics in brackets are describing the data in 2019. From Table 2, we can spot out that the wholesale prices differ from 2019 and 2020. We will verify such a claim with a statistical hypothesis test in the later section.

Variable	Min.	Q1	Median	Mean	Q3	Max.
Confirmed	0	1	5	20.11	23	173
Deaths	0	0	0	0.42	0	6
Recovered	0	1	6	19.16	21	173
Eggs	7.25 (7.50)	7.60 (7.75)	7.75 (7.85)	7.70 (7.82)	7.80 (7.90)	8.10 (8.15)
Freshwater fish	16.53 (16.63)	16.77 (16.93)	16.87 (17.00)	16.84 (17.18)	16.9 (17.37)	18.27 (18.17)
Livestock / Poultry	29.19 (15.35)	36.2 (29.29)	38.17 (31.93)	37.62 (31.29)	39.4 (33.8)	45.16 (39.04)
Marine fish	31.40 (34.80)	32.40 (36.90)	33.60 (38.30)	33.61 (38.35)	34.20 (39.60)	37.50 (41.70)
Vegetables	5.38 (5.44)	6.24 (5.95)	6.57 (6.50)	6.97 (7.11)	7.53 (8.26)	12.95 (10.28)

(*Table 2: summary of variables*)

Many statistical analysis methods rely on the distribution of data. To understand how data behave in the dataset, we use boxplots to grab a quick inspect into the distributions of different variables. Figure 1 on the right-hand side, it shows that there are lots of outliers in the boxplot. Variables “Confirmed”, “Deaths” and “Recovered” are right-skewed intuitively. It seems reasonable since time-series data vary from time to time.



(*Figure 1: boxplots of confirmed cases, deaths and recovered cases*)

(*Figure 2: boxplots of wholesale price variables*)

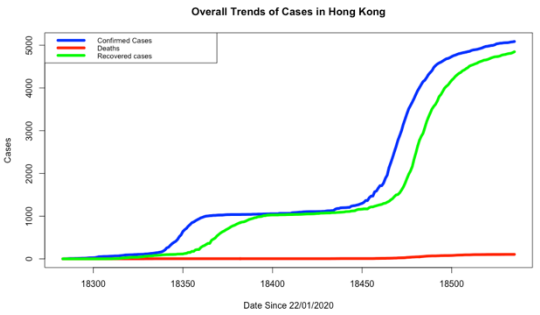
Figure 2 models the distributions of different types of food category. It gives us an idea that the variability among variables was not significant. They seem to be normally distributed although there are few outliers. However, Shapiro-Wilk normality tests do not conclude such a result. All the p-values are significantly less than $\alpha = 0.05$. In other words, they are not normally distributed. Hence, non-parametric tests would be more favourable on those data.

Furthermore, since we propose to forecast the number of confirmed cases in the later section, we set up a Kruskal Wallis test on whether there is seasonality in the "Confirmed" variable. It returns a p-value of 0.604, which does not provide strong evidence on there is a seasonality in the data. Knowing with such a characteristic, we intend to use exponential smoothing method to make inference on the future.

4. Visualisation — Overall Trends of Cases in Hong Kong

We have already perceived the general ideas on the raw data. Momentarily, we are going to visualise the severeness about the novel coronavirus in Hong Kong. It would be efficient to inspect the behaviour of data and obtain a deeper understanding of the actual situation before we dive into the discussion.

Figure 3 demonstrates the situation of the novel coronavirus in the past nine months. The first thing we spot out is that the number of cases increased as time passed. We also notice that the increment of the death rate was not as fast as that of the confirmed cases. If we look at the chart carefully, we should comprehend that the novel coronavirus was under control from April to July. However, the circumstance was out of control again afterwards. Because of the unstable condition, staying at home is the merely choice we may subdue the outbreak.



(*Figure 3: line plots of the overall trends in Hong Kong*)

5. Derivative under the Novel Coronavirus

(*Figure 4: line plot of number of views being watched)*

Throughout the days in 2020, we spent most of our time at home to reduce the chances of being infected by others. People started to grab something to do rather than just sitting in front of a table for working or learning. According to YouTube Culture & Trends, cooking videos were the most popular videos during the quarantine. For example, tutorials made by Gordon Ramsay, a famous British chef, has been viewed over 1 million times within 2 months. At the same time, there was an enormous amount of recipe videos uploaded to fulfill the demand.

Master Chefs is the derivative under the novel coronavirus for sure. People started to equip themselves and took practices on their own. Here comes to our discussion. We want to know whether the change in our living behaviour would lead to any change in the prices of fresh food in Hong Kong. Given the fact that the wholesale price variables do not follow normal distributions, we are going to use the Wilcoxon Signed-Rank test to verify the statement. The test makes use of the magnitudes of the difference among data in 2019 and 2020. It compares the median of variables by labelling the sign of value after subtraction. More precisely, it is a non-parametric version of the paired t-test with less assumption on the data. The sum of positive signed rank denoted as R^+. We will reject H_0 if the test statistic falls into the critical region $z_{1-\alpha/2}$.

$$|T_0| = \frac{|R^+ - E(R^+)| - 0.5}{\sqrt{Var(R^+)}}$$

(*Equation 1: test statistic of Wilcoxon Signed-Rank test, with continuity correction factor*)

From the outputs (see Appendixes 10.5. for details), only the p-value of "Vegetables" is greater than the significant level at 5%. In other words, only the wholesale price of vegetables does not change over the pandemic of the novel coronavirus. However, it does not imply that the novel coronavirus was the primary reason of changing the wholesale prices. There may be other factors that were affecting the price directly, such as the Consumer Price Index and supply of that item. Such a problem motivates us into the next section.

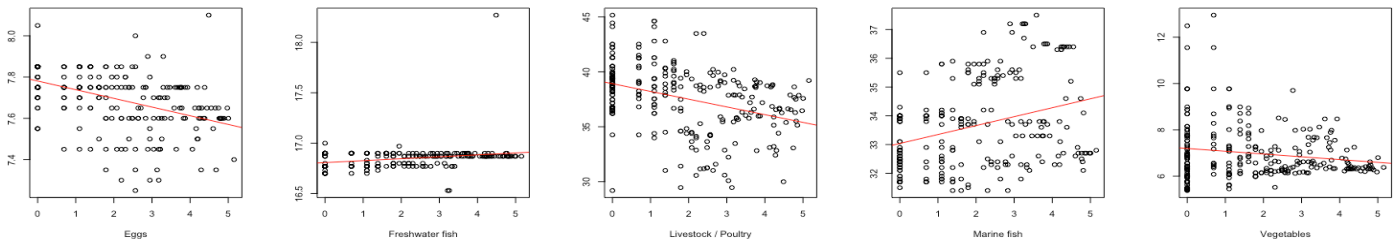
6. Association among the Confirmed Cases and the Wholesale Prices

As we notice that the novel coronavirus may not be the primary factor that affects the wholesale prices, we attempt to use simple linear regression (SRL) models to capture the association among the confirmed cases and the wholesale prices. SRL model is a robust statistical method to study the relationship between the response variable (RV) and the explanatory variables (EV). Since we are interested in whether there are strong relationships between the wholesale prices and the number of confirmed cases, we equate them as RV and EV, respectively. We assume there are independent random error terms $\boldsymbol{\varepsilon}$ with mean zero and fixed variance.

$$E(wholesale\ price_i) = \beta_0 + \beta_1 \times Number\ of\ Confirmed\ Cases_i$$

(*Equation 2: equation of SLR*)

To reduce the effect from outliers shown in Figure 1, we also apply power transformation $\Psi(x, 0)$ on EV to pool the extreme values into the data centre for obtaining a better association.



(*Figure 5: SRL lines of wholesale price in different types of food category against the number of confirmed cases*)

With the estimated coefficients (see Appendixes 10.6. for details), the models do not suggest there were strong associations among the wholesale prices and the number of confirmed cases. There were at most 22% of variation from the wholesale prices explained by the number of confirmed cases. We may conclude that there was only a little effect on the wholesale prices by the novel coronavirus. Here we emphasise that there is no need to be panic on the shortage or change in price under the outbreak. We also suggest the public think twice before following the instruction form the undependable websites.

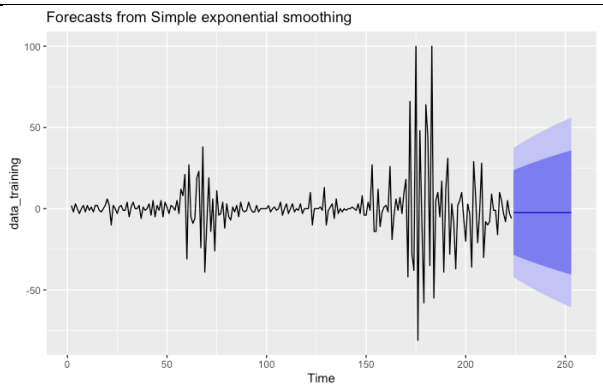
7. Forecasting about the Number of Confirmed Cases in the Next 30 Days

Back to the discussion on the circumstances, we intend to provide further information to the public on how to deal with fake news in the future. Here we initiate an exponential smoothing (ES) model to archive the goal. ES model is powerful for making short-term forecasts for time series data. It requires no assumption about the correlations between successive values. Besides, ES model favours in the given data since "Confirmed" has no trend and no seasonal pattern. For the ES model set up, there is an exponentially smoothed value E_i and observed value Y_i in a given period i. Moreover, there is also a smoothing coefficient $W \in (0,1)$, which controls the degree of weight on recent observations.

$$E_1 = Y_1 \text{ and } E_i = WY_i + (1 - W)E_{i-1}$$

(*Equation 3: equation of exponential smoothing model*)

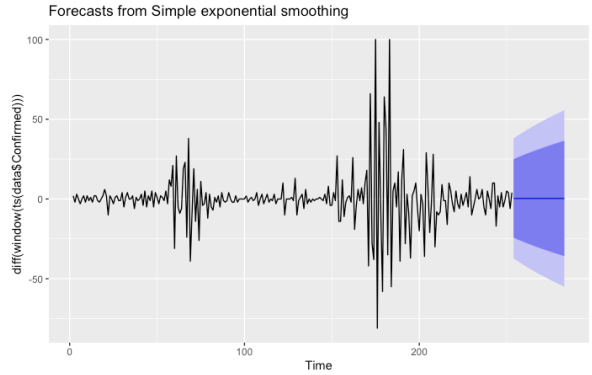
We initial W at 0.2, which will smooth out unwanted cyclical and irregular components in the forecasts. We treat the data from January to August as train data for the model and utilise the data in September as test data for validating the accuracy of the proposed model.



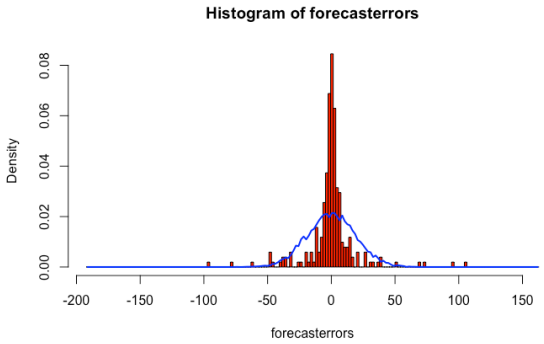
(*Figure 6: forecasts from SE model using training data*)

(*Figure 7: predicted value against the test data*)

Figure 6 displays the expected confirmed cases in September. The 80% prediction interval colour in dark blue whereas the 95% prediction interval colour in light blue. Figure 7 compares the forecast values and test data. Since the prediction intervals cover all the actual values in September, we consider the fitted model is good enough for our propose. Consequently, we model the ES model again using all the data as train data to predict the number of confirmed cases in the next 30 days.



(*Figure 8: forecasts from SE model using full dataset*)



(*Figure 9: histogram of forecast errors*)

Figure 8 warns us there is a potential outbreak in the next 30 days, where the number of confirm patients may increase over 50 per day. Figure 9 supports the accuracy of the fitted model. The forecast errors are approximately symmetric about zero, which do not violate the model assumption. We are going to use the ES model to conclude our discussion in the next section.

8. Conclusion

In reviewing the past situation about the novel coronavirus, Figure 3 visualises the overall trends on the number of confirmed cases, deaths and recovered patients. The number of cases increased intermittently, and the outbreak floated during the months of quarantine. With such a long vacation in our homes, there were a tremendous amount of new Master Chefs born while watching cooking videos on YouTube. From the Wilcoxon Sign-Rank test, we understand there was a change in the wholesale prices compared to a year before. Yet, the SRL models do not suggest there were high correlations among the number of confirmed cases and the wholesale prices. By retrieving the news in previous months, people were panic to buy because of the fake news over the network. Indeed, we may only rely on the information available on the internet when we were staying at home. However, that disinformation was a joke to the history we have seen. After the findings we have obtained, we suggest the public not to threaten under those circumstances. Think twice before taking action based on the information from websites. The ES model predicts the outbreak will out of control again. That means there may be another fake news in the future. Here we want to make a kindly remind to all of us to source back the truth. We should search for more information related to the news as a reference check. We should raise our awareness of fake news while fighting against the novel coronavirus.

9. Reference

DATA.GOV.HK (2020) Wholesale prices of major fresh food (average reference figures). Retrieved from https://data.gov.hk/en-data/dataset/hk-afcd-afcdlist-wholesale-prices

SRK (2020) Novel Corona Virus 2019 Dataset. Kaggle. Retrieved from https://www.kaggle.com/sudalairajkumar/novel-corona-virus-2019-dataset

Watching the Pandemic (2020) YouTube Culture & Trends. Retrieved from https://www.youtube.com/trends/articles/covid-impact/

YouTube During COVID-19 (2020) YouTube Culture & Trends. Retrieved from https://www.youtube.com/trends/articles/what-it-means-to-stayhome-on-youtube/

10. Appendixes

We implement all coding with the R programming language.

10.1. R Code for Data Pre-processing

```
## read price data
readPrice = function(path){
  setwd(path)

  data = as.data.frame(matrix(NA, nr = 1, nc = 3))
  names(data) = c("Types", "Average", "Date")
  data = data[-1,]

  files = list.files(pattern = "*.csv")
  for(i in 1:length(files)) {
    # read data
    date = substr(files[i], 1, 8)
    fileName = paste0(date, "-Wholesale_Prices.csv")
    file.rename(files[i], fileName)
    data_input = read.csv(fileName, header = T, stringsAsFactors = F, fileEncoding =
"latin1")

    # keep only necessary rows
    flag = data_input[, "FRESH.FOOD.CATEGORY"] == "Livestock / Poultry" | data_input[,
"FRESH.FOOD.CATEGORY"] == "Marine fish" | data_input[, "FRESH.FOOD.CATEGORY"] == "Freshwater
fish" | data_input[, "FRESH.FOOD.CATEGORY"] == "Vegetables" | data_input[,
"FRESH.FOOD.CATEGORY"] == "Eggs"
    data_input = data_input[flag,]

    # data cleansing
    data_input$PRICE..THIS.MORNING. = as.character(data_input$PRICE..THIS.MORNING.)
    data_input$PRICE..THIS.MORNING. = gsub("^\\s+|\\s+$", "",
data_input$PRICE..THIS.MORNING.)
    data_input$PRICE..THIS.MORNING. = gsub("-", NA, data_input$PRICE..THIS.MORNING.)
    data_input$PRICE..THIS.MORNING. = as.numeric(data_input$PRICE..THIS.MORNING.)

    # modify data by unit
    data_input[which(data_input$FRESH.FOOD.CATEGORY == "Livestock / Poultry" &
data_input$UNIT == "($ / Picul)"), "PRICE..THIS.MORNING."] =
data_input[which(data_input$FRESH.FOOD.CATEGORY == "Livestock / Poultry" & data_input$UNIT
== "($ / Picul)"), "PRICE..THIS.MORNING."] / 100
    data_input[which(data_input$FRESH.FOOD.CATEGORY == "Eggs"), "PRICE..THIS.MORNING."] =
data_input[which(data_input$FRESH.FOOD.CATEGORY == "Eggs"), "PRICE..THIS.MORNING."] * 10
```

```
    # keep only necessary columns
    usefulVar = data_input[, c("FRESH.FOOD.CATEGORY", "PRICE..THIS.MORNING.")]

    # find menas by types
    usefulVar = aggregate(usefulVar$PRICE..THIS.MORNING., by =
list(usefulVar$FRESH.FOOD.CATEGORY), FUN = mean, na.rm = TRUE)
    names(usefulVar) = c("Types", "Average")
    usefulVar$Average = round(usefulVar$Average, 2)
    usefulVar$Date = as.numeric(date)

    # output data
    data = rbind(data, usefulVar)
    rm(date, fileName, data_input, usefulVar, flag)
  }
  return(data)
}

data_2019 = readPrice("/Users/jackchan/Downloads/price 2019")
data = readPrice("/Users/jackchan/Downloads/price 2020")


## read covid 19 data
setwd("/Users/jackchan/Downloads")
data_covid = read.csv("time_series_covid19_confirmed_global.csv", header = T)
data_covid_deaths = read.csv("time_series_covid19_deaths_global.csv", header = T)
data_covid_recovered = read.csv("time_series_covid19_recovered_global.csv", header = T)

# data cleansing
Date = gsub("X", "", names(data_covid))
Date = as.character(as.Date(Date[5:length(Date)], "%m.%d.%y"))

# subtract data
data_confirm_cumulative = as.numeric(data_covid[which(data_covid$Province.State == "Hong
Kong"), 5:length(names(data_covid))])
data_deaths_cumulative = as.numeric(data_covid_deaths[which(data_covid_deaths$Province.State
== "Hong Kong"), 5:length(names(data_covid_deaths))])
data_recovered_cumulative =
as.numeric(data_covid_recovered[which(data_covid_recovered$Province.State == "Hong Kong"),
5:length(names(data_covid_recovered))])
Confirmed = data_confirm_cumulative[1]
Deaths = data_deaths_cumulative[1]
Recovered = data_recovered_cumulative[1]
for(i in 2:length(data_confirm_cumulative)) {
  Confirmed[i] = data_confirm_cumulative[i] - data_confirm_cumulative[i - 1]
  Deaths[i] = data_deaths_cumulative[i] - data_deaths_cumulative[i - 1]
  Recovered[i] = data_recovered_cumulative[i] - data_recovered_cumulative[i - 1]
}

# negative value in Recovered
Recovered = abs(Recovered)

# create new data frame to store all data
data_new = as.data.frame(cbind(Date, Confirmed, Deaths, Recovered))
data_new$Date = as.Date(data_new$Date)
data_new$Confirmed =  as.numeric(as.character(data_new$Confirmed))
data_new$Deaths =  as.numeric(as.character(data_new$Deaths))
data_new$Recovered =  as.numeric(as.character(data_new$Recovered))

# create data frame by type
type = as.character(unique(data$Types))
for(i in 1:length(type)) {
  assign(type[i], data[which(data$Types == type[i]),])
}
```

```
# megre data into data_new
for(i in 1:length(type)) {
  tempDF = data[which(data$Types == type[i]),]
  tempDF$Date = as.character(tempDF$Date)
  tempDF$Date = as.Date(paste0(substr(tempDF$Date,1,4), "-", substr(tempDF$Date,5,6), "-",
substr(tempDF$Date,7,8)))
  tempDF = tempDF[, c("Average", "Date")]
  names(tempDF) = c(type[i], "Date")
  data_new = merge(data_new, tempDF, by = "Date", all.x = T)
  rm(tempDF)
}

# create data_2019 frame by type
type = as.character(unique(data_2019$Types))
for(i in 1:length(type)) {
  assign(type[i], data_2019[which(data_2019$Types == type[i]),])
}

# megre data_2019 into data_new
for(i in 1:length(type)) {
  tempDF = data_2019[which(data_2019$Types == type[i]),]
  tempDF$Date = as.character(tempDF$Date + 10000)
  tempDF$Date = as.Date(paste0(substr(tempDF$Date,1,4), "-", substr(tempDF$Date,5,6), "-",
substr(tempDF$Date,7,8)))
  tempDF = tempDF[, c("Average", "Date")]
  names(tempDF) = c(paste0(type[i], " 2019"), "Date")
  data_new = merge(data_new, tempDF, by = "Date", all.x = T)
  rm(tempDF)
}

# dealing with missing data
data_new[, c("Eggs", "Freshwater fish", "Livestock / Poultry", "Marine fish",
"Vegetables")][1,] = data_new[, c("Eggs", "Freshwater fish", "Livestock / Poultry", "Marine
fish", "Vegetables")][2,]

for(i in 2:nrow(data_new)) {
  if(is.na(data_new$Eggs[i]))
    data_new$Eggs[i] = data_new$Eggs[i - 1]
  if(is.na(data_new$`Freshwater fish`[i]))
    data_new$`Freshwater fish`[i] = data_new$`Freshwater fish`[i - 1]
  if(is.na(data_new$`Livestock / Poultry`[i]))
    data_new$`Livestock / Poultry`[i] = data_new$`Livestock / Poultry`[i - 1]
  if(is.na(data_new$`Marine fish`[i]))
    data_new$`Marine fish`[i] = data_new$`Marine fish`[i - 1]
  if(is.na(data_new$Vegetables[i]))
    data_new$Vegetables[i] = data_new$Vegetables[i - 1]

  if(is.na(data_new$`Eggs 2019`[i]))
    data_new$`Eggs 2019`[i] = data_new$`Eggs 2019`[i - 1]
  if(is.na(data_new$`Freshwater fish 2019`[i]))
    data_new$`Freshwater fish 2019`[i] = data_new$`Freshwater fish 2019`[i - 1]
  if(is.na(data_new$`Livestock / Poultry 2019`[i]))
    data_new$`Livestock / Poultry 2019`[i] = data_new$`Livestock / Poultry 2019`[i - 1]
  if(is.na(data_new$`Marine fish 2019`[i]))
    data_new$`Marine fish 2019`[i] = data_new$`Marine fish 2019`[i - 1]
  if(is.na(data_new$`Vegetables 2019`[i]))
    data_new$`Vegetables 2019`[i] = data_new$`Vegetables 2019`[i - 1]
}

data = data_new
rm(data_covid, data_covid_deaths, data_covid_recovered, data_new, data_2019,Eggs,
`Freshwater fish`, `Livestock / Poultry`, `Marine fish`, Vegetables)
```

10.2. R Code for EDA

```
## EDA
summary(data)

# boxplot for cases: skewed, suggest transfer data
boxplot(data[,2:4],
        main = "The Number of the Novel Coronavirus Cases",
        xlab = "Number of Cases",
        col = c("blue", "red", "green"),
        border = "red",
        horizontal = T,
        notch = T)

# boxplot for prices: skewed, suggest transfer data
boxplot(data[, 5:9],
        main = "Average Wholesale Prices",
        ylab = "Average Price",
        col = "orange",
        border = "red",
        horizontal = F,
        notch = T)

# normality test for wholesale prices
shapiro.test(data$Eggs)
shapiro.test(data$`Freshwater fish`)
shapiro.test(data$`Livestock / Poultry`)
shapiro.test(data$`Marine fish`)
shapiro.test(data$Vegetables)

# test of seasonality: not reject H_0, no seasonality
library("seastests")
kw(ts(data$Confirmed), freq = 3, diff = T, residuals = F, autoarima = T)
```

10.3. R Code for visualisation

```
## visualisation
# line plot for cases in Hong Kong
matplot(data$Date, cbind(cumsum(data$Confirm), cumsum(data$Deaths), cumsum(data$Recover)),
        main = "Overall Trends of Cases in Hong Kong",
        xlab = "Date Since 22/01/2020",
        ylab = "Cases",
        type = "l",
        col = c("blue", "red", "green"),
        lty = 1,
        lwd = 5)

legend("topleft",
       legend = c("Confirmed Cases", "Deaths", "Recovered cases"),
       col = c("blue", "re d", "green"),
       lty = 1,
       cex = 0.8,
       lwd = 5)

# time series for visualisation
data_training_forecasts = HoltWinters(diff(window(ts(data$Confirmed))), beta = F, gamma = F)
plot(data_training_forecasts)
data_training_forecasts$SSE

# time series for forecasting
library(tidyverse)
```

```
library(fpp2)
data_training = diff(window(ts(data$Confirmed), end = nrow(data) - 30))
data_testing = diff(window(ts(data$Confirmed), start = nrow(data) - 30))
autoplot(data_training)

# minimum alpha
alpha = seq(.01, .99, by = .01)
RMSE = NA
for(i in seq_along(alpha)) {
  fit = ses(data_training, alpha = alpha[i], h = 30)
  RMSE[i] = accuracy(fit, data_testing)[2,2]
}
alpha.fit = data_frame(alpha, RMSE)
alpha.min = filter(alpha.fit, RMSE == min(RMSE))
ggplot(alpha.fit, aes(alpha, RMSE)) + geom_line() + geom_point(data = alpha.min, aes(alpha,
RMSE), size = 2, color = "blue")

# forecast validation
data_training_ses = ses(data_training, alpha = 0.2, h = 30)
autoplot(data_training_ses)
accuracy(data_training_ses, data_testing)
p1 = autoplot(data_training_ses) + theme(legend.position = "bottom")
p2 = autoplot(data_testing) + autolayer(data_training_ses, alpha = .5) + ggtitle("Predicted
vs. actuals for the test data set")
gridExtra::grid.arrange(p1, p2, nrow = 1)

# model assumption validation
plotForecastErrors <- function(forecasterrors) {
  mybinsize = IQR(forecasterrors)/4
  mysd = sd(forecasterrors)
  mymin = min(forecasterrors) - mysd*5
  mymax = max(forecasterrors) + mysd*3
  mynorm = rnorm(10000, mean = 0, sd = mysd)
  mymin2 = min(mynorm)
  mymax2 = max(mynorm)
  if(mymin2 < mymin) {
    mymin <- mymin2
  }
  if(mymax2 > mymax) {
    mymax <- mymax2
  }
  mybins = seq(mymin, mymax, mybinsize)
  hist(forecasterrors, col = "red", freq = F, breaks = mybins)
  myhist = hist(mynorm, plot = F, breaks = mybins)
  points(myhist$mids, myhist$density, type = "l", col = "blue", lwd = 2)
}

plotForecastErrors(data_training_ses$residuals)

# whole data set as training dataset to forecast next 60 days: still possilbe to inrcrease
number of confirmed cases
data_ses = ses(diff(window(ts(data$Confirmed))), alpha = 0.2, h = 30)
autoplot(data_ses)
plotForecastErrors(data_ses$residuals)
```

10.4. R Code for Association

```
## assoication
# wilcoxon signed-rank test
wilcox.test(data$Eggs - data$`Eggs 2019`,
            y = NULL,
            alternative = "two.sided",
```

```
            mu = 0,
            paired = F,
            exact = NULL,
            correct = T,
            conf.int = F)

wilcox.test(data$`Freshwater fish` - data$`Freshwater fish 2019`,
            y = NULL,
            alternative = "two.sided",
            mu = 0,
            paired = F,
            exact = NULL,
            correct = T,
            conf.int = F)

wilcox.test(data$`Livestock / Poultry` - data$`Livestock / Poultry 2019`,
            y = NULL,
            alternative = "two.sided",
            mu = 0,
            paired = F,
            exact = NULL,
            correct = T,
            conf.int = F)

wilcox.test(data$`Marine fish` - data$`Marine fish 2019`,
            y = NULL,
            alternative = "two.sided",
            mu = 0,
            paired = F,
            exact = NULL,
            correct = T,
            conf.int = F)

wilcox.test(data$Vegetables - data$`Vegetables 2019`,
            y = NULL,
            alternative = "two.sided",
            mu = 0,
            paired = F,
            exact = NULL,
            correct = T,
            conf.int = F)

# power transform on confirmed cases
cases = ifelse(data$Confirmed == 0, 0, log(data$Confirmed))

# R^2 of prices and confirmed cases: the variation explained by confirmed cases is not
significant
par(mfrow = c(1, 5))
rsq = NA
for(i in 5:9) {
  plot(cases, data[, i],
       ylab = "",
       xlab = names(data)[i])
  reg = lm(data[, i]~cases)
  abline(reg, col = "red")
  rsq[i - 4] = summary(reg)$r.squared
}
rsq
```

10.5. Results of Wilcoxon Signed-Rank Test

```
      Wilcoxon signed rank test with continuity correction
```

```
data:  data$Eggs - data$`Eggs 2019`
V = 4241.5, p-value < 2.2e-16
alternative hypothesis: true location is not equal to 0

        Wilcoxon signed rank test with continuity correction
data:  data$`Freshwater fish` - data$`Freshwater fish 2019`
V = 765.5, p-value < 2.2e-16
alternative hypothesis: true location is not equal to 0

        Wilcoxon signed rank test with continuity correction
data:  data$`Livestock / Poultry` - data$`Livestock / Poultry 2019`
V = 31042, p-value < 2.2e-16
alternative hypothesis: true location is not equal to 0

        Wilcoxon signed rank test with continuity correction
data:  data$`Marine fish` - data$`Marine fish 2019`
V = 11, p-value < 2.2e-16
alternative hypothesis: true location is not equal to 0

        Wilcoxon signed rank test with continuity correction
data:  data$Vegetables - data$`Vegetables 2019`
V = 14384, p-value = 0.1796
alternative hypothesis: true location is not equal to 0
```

10.6. Coefficient of Determinations

```
[1] 0.2199812 0.0706120 0.1322168 0.1054066 0.0264461
```