Time Series Analysis of Seasonal Flu in USA

Gupta, Moro, Kannan

1. Business Understanding

Introduction: Influenza, commonly known as flu, is an infectious respiratory illness caused by influenza viruses that infects the nose, throat, and occasionally the lungs. This infection which is caused by virus families Influenza A, B, C, and D is transmitted throughout human populations all year round, but the infection rate grows ten-fold during the cooler temperature periods. Hence the outbreak of influenza virus normally occurs during the fall and winter periods of the year making it a seasonal event.

The common reason behind this flu strike during winter includes lack of sunlight and people's lifestyle. During the winter months, the production of Vitamin D and Melatonin in the human body is affected adversely due to lack of sun light. This compromises the human immune system as both Vitamin D and Melatonin are crucial factors to maintain strong immunity. Also, drier and colder air makes the environment suitable for the virus to survive better.

These flu infections are highly contagious, and it causes mild to severe illness. The serious outcome of this infection can result in hospitalization or even death. This imposes high risk to certain age group especially the elderly, children and infants. Therefore, proper infection control measures need to be in place.

Infection Control Measures: The common precautionary steps recommended to prevent flu include maintaining good hygiene by cleaning hands regularly, sanitizing or disinfecting surfaces, staying home while sick and practicing active and healthy lifestyle that can supplement the generation of the required Vitamin D and Melatonin. Despite all these, the most important measure to prevent seasonal flu infection is to administer a vaccine.

Each annual flu season is normally associated with a major influenza virus subtype. The associated subtype changes each year due to development of immunological resistance to a previous year's strain (through exposure and vaccinations). Due to this ever-changing nature of these viruses, the vaccines developed cannot be 100% effective to prevent infections from all different strains but is usually 40 to 60% effective. This is still a notable proportion as it helps in significant reduction in health burden of illness, hospital visits and death.

Proper infection control measures help in building immunity among large population of people and even a small increase in "community immunity" will prevent further spread of infection thus protecting the "high risk" population and avoid hospitals getting overwhelmed with huge number of patients.

Preparedness: As discussed above, the impact of an influenza outbreak on individuals and society can be disastrous but can be controlled with proper precautionary planning in place. This means framing a comprehensive plan to mitigate the risk of a severe outbreak. This should include set of guidelines and recommendation to public on hygiene, respiratory etiquette, environmental measures, travel advice, vaccination etc.

Developing a statistical model for flu patterns using historical data can be helpful for this planning. Most of the action items from such precautionary plan like maintaining enough supply of vaccines, sanitizers, face masks, regulations on public gathering, availability of emergency / intensive care can be effectively implemented if an appropriate model is available for influenza occurrences. Hence our focus was to build a statistical model using historical influenza data, in order to be better prepared for any future influenza outbreak.

Assumption: Though this case study was performed in 2020, the dataset only included data till 2019 and did not include any data on influenza cases during covid19 pandemic. We assumed that the future influenza cases (in 2020) are not influenced by the novel corona virus and the underlying process that generated the flu cases till 2019 will remain the same in 2020 as well.

2. Data Evaluation / Engineering

The influenza data used for this case study have been downloaded from then World Health Organization (WHO) web site, specifically from the Influenza Laboratory Surveillance Information page [1]. The page reports influenza data collected through the FluID global platform and represents a comprehensive database where local health entities can upload and read activities for all WHO regions. The system can be used to report both quantitative and qualitative assessments of regional spread of influenza related illness [2]. As World Health Organization is among the most important specialized agencies in the world for the international public health, we can assume the data reported to be of a very high quality and ideal for our case study.

The dataset has been downloaded from the above website in a comma separated value (csv) format. It includes 22 variables related to weekly influenza spread measures in the United States from 2010 to 2019. We have a total of 521 rows (observations), and each row represents the flu infections in the United States for a specific week. Table 1 shows the list of columns available in the dataset and their description.

Table 1 Fields Description

Column Name	Type	Description
Country	Text	Name of the Country the data is about (United States)
WHOREGION	Text	WHO region the country is part of. (Region of the Americas)
FLUREGION	Text	Transmission zone the data are related to (North America)
Year	Numeric	Year (2010 to 2019)
Week	Numeric	Week of the Year (1 to 52 or 53)
SDATE	Date	First day of the week
EDATE	Date	Last day of the week
SPEC_RECEIVED_NB	Numeric	Number of specimens collected
SPEC_PROCESSED_NB	Numeric	Number of specimens processes
AH1	Numeric	Number of influence type A, subtype H1, viruses detected
AH1N12009	Numeric	Number of influence type A, subtype H1N12009, viruses detected
AH1N3	Numeric	Number of influence type A, subtype H1N3, viruses detected
AH3	Numeric	Number of influence type A, subtype H3, viruses detected
AH5	Numeric	Number of influence type A, subtype H5, viruses detected
ANOTSUBTYPED	Numeric	Number of influence type A, without any subtype, viruses detected
INF_A	Numeric	Total Number of influence type A viruses detected
BYAMAGATA	Numeric	Number of influence type B, lineage Yamagata, viruses detected
BVICTORIA	Numeric	Number of influence type B, lineage Victoria, viruses detected
BNOTDETERMINED	Numeric	Number of influence type B, lineage not determined, viruses detected
INF_B	Numeric	Total Number of influence type B viruses detected
ALL_INF	Numeric	Total Number of influence viruses detected (type A and B combined)
TITLE	Text	Type of determined outbreak.

As the objective of this case study is to develop a statistical model to study the flu patterns, we will use the **ALL_INF** column as our target measure. This embodies the cases for all types of influenza infections in the United States. This historic data on influenza can be interpreted as a time series which is simply a series of data points ordered in time. In a timeseries, time is the independent variable and the goal is usually to make a forecast the target variable for the future. Hence, our statistical model will be a time series model that helps with studying the influenza occurrences in the past and forecast any future cases to help with planning.

2.1 Analyzing the target variable

The target variable (ALL_INF) has no nulls values and shows influenza cases ranging from two to 26,386 cases per week. Fig. 1 shows that most of the values are in the low range of 0 to 500 but that there are a few high peaks in specific periods (usually the colder months around the start and end of a year).

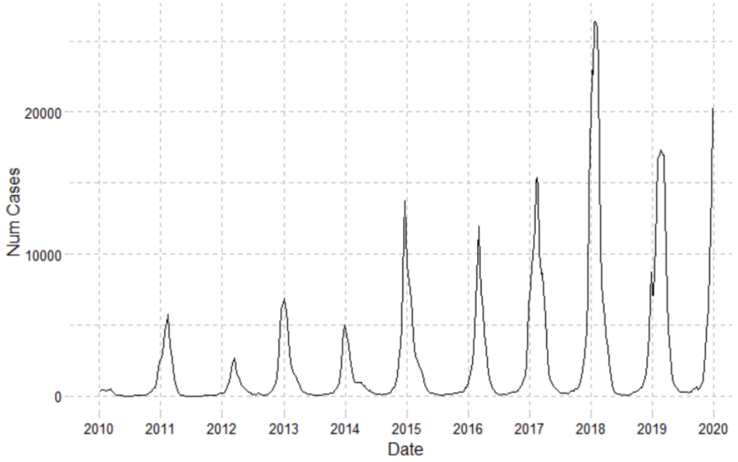


Fig. 1 Plot of the target variable throughout time

Fig. 1 shows that the peaks of the flu infections are increasing over time, while the valleys are relatively flat. This can sometimes be difficult for time series models to capture. To improve the performance of the model we prefer to have a more consistent pattern in the peaks and valleys. One way to achieve this is to take the log transformation of the target variable which results in squishing the abnormally high peak values and making it easier for the models to capture the underlying behavior. Hence, we proceeded to applying this transformation which results in the time series shown in Fig. 2. The resulting distribution shows a more balance representation of values with peaks reaching the logged value of 10 and equal number of valleys reaching the logged value of 2. Although the valleys are still not perfect (especially in the initial months), we believe this representation is more suited for a time-based model.

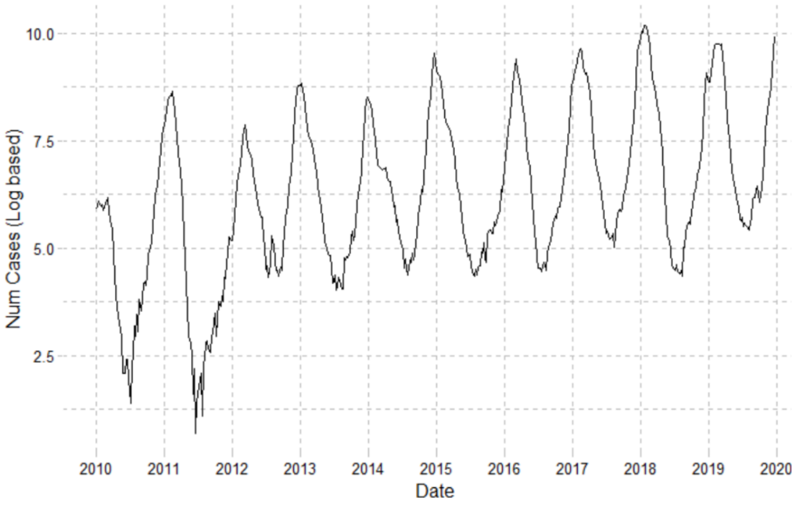


Fig. 2 Pot of the log-transformed target variable throughout time

2.2 Checking Stationarity and Seasonality

There are few aspects that needs to be addressed while dealing with time series modelling. This includes both Stationarity and Seasonality.

Stationarity: This is an important characteristic of time series that determines if its statistical properties change over time. There are 3 conditions to check the stationarity of a time series. For a timeseries data to be stationary, it must satisfy the following conditions:

a) **Constant Mean:** The mean of the time series should be constant over time.
b) **Constant Variance:** The variance of the time series should be constant over time.
c) **Constant Autocorrelation:** Constant Autocorrelation means constant degree of similarity between a given time series and a lagged version of itself over successive time intervals.

Seasonality: Seasonality refers to presence of any variation in the time series that occurs at specific intervals. In this case, we already mentioned that influenza infections are more prevalent during winter making it a seasonal event. Depending on the time series, seasonality can be weekly, monthly, quarterly, or yearly. In some cases, the time series can have multiple seasonal periods as well. Seasonal data is considered non-stationary since the mean is not constant over time (peaks at certain periods of time only and valleys at other periods of time).

Verifying Stationarity and Seasonality: While a stationary data can use a simple model, a non-stationary data may require a more complex analysis to identify and control the factors that determine the non-stationary behavior throughout time. From Fig. 2, we can clearly see a cyclic behavior that repeats every year. This is an early indication that the data is non-stationary as it is impacted by a seasonal behavior (we see peaks during the colder months and valleys during the summer months).

As we have only one observation per week, we cannot calculate the mean and variance for each week. But, based on above observations, we can assume the influenza behavior is cyclic and each week can be comparable across the different years to get a pseudo estimate of the mean and variance. Fig. 3 shows the comparison of the number of cases of influenza (log) per week across years. Each year is a shade of blue with most recent years being lighter. We notice that the most recent years have an increased number of cases compared to the earlier years in general. This points to an upward trend in the data. Furthermore, we see the winter months have a higher number of cases compared to summer months. This points to seasonality as discussed earlier. Hence the mean of the data does not seem to be constant over time and condition 1 for stationarity does not seem to be met. From Fig. 3 we can also see the range of the cases in each week across years is almost constant, implying a quasi-constant variance and that condition 2 for stationarity is satisfied. However, condition 1 is violated, we cannot assume stationarity and hence this data needs to be modeled appropriately.

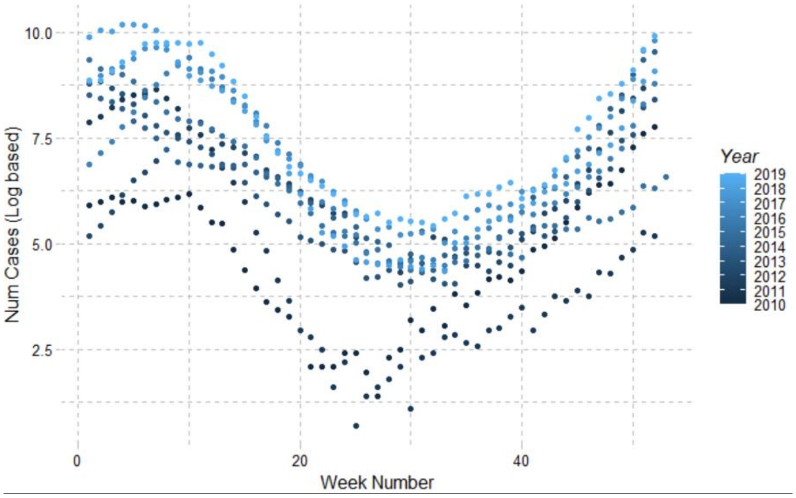


Fig. 3 Comparing number of cases of influence (log based) per week across years

Another tool we can use to measure the seasonality of the data is the 'Parzen Window' frequency chart. The Parzen Window allows us to estimate the frequency components in the data including the seasonal components. Fig. 4 shows the Parzen Window representation of the influenza cases. We notice a peak in the distribution at frequency near 0.02, equivalent to a period of around 52 weeks. This is a clear indication of the seasonality of 52 weeks in the data. This behavior is also confirmed by the autocorrelation chart in Fig. 4 where each period has a strong positive correlation with the 52 periods before it. If there was a peak in the 52nd prior week then there is a peak in the current week as well (and similarly for the valleys).

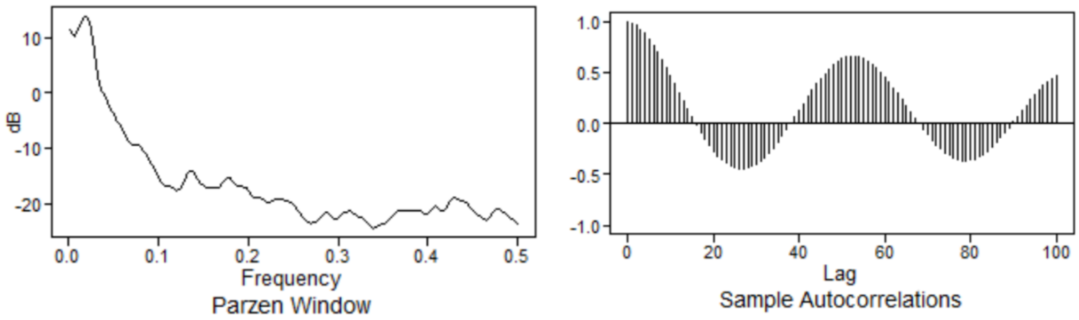


Fig. 4 Parzen Window and autocorrelation of the number of influence cases

The next step is to evaluate if there are other factors that make the distribution non-stationary. We need to remove the seasonal effect from the data in order to do so, and an 'autoregressive transformation' allows us to do this. This transformation is achieved by subtracting the data from the 52nd prior week from the current value. What is left is the non-seasonal data. Fig. 5 shows the comparison between the original seasonal data and the non-seasonal transformed data. We can see that the cyclic behavior has now disappeared. Fig. 6 shows the Parzen Window for the transformed data. It does not contain any frequency at 0.02 but has a new frequency component at ~0.010 (period of ~100 weeks). This is also confirmed by the sample correlation chart in Fig. 6 which has a cyclical behavior with a period of ~100. We believe this effect can be a remnant of the weekly seasonality in the data caused by the fact that a week is not a perfect factor of one year (one year is 52.14, or 52.29 weeks for lap-years). Hence, some years will have a seasonality of 52 week while other years will have a seasonality of 53 weeks.

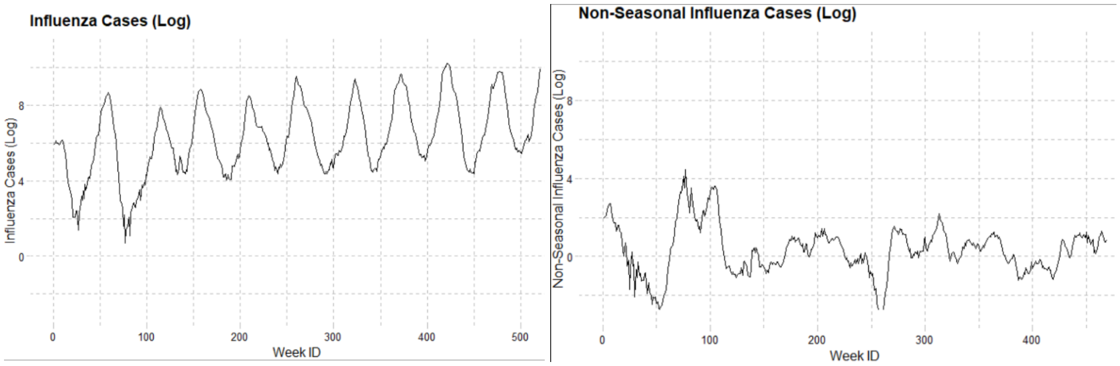


Fig. 5 Original vs. Non-Seasonal transformed influenza cases (log)

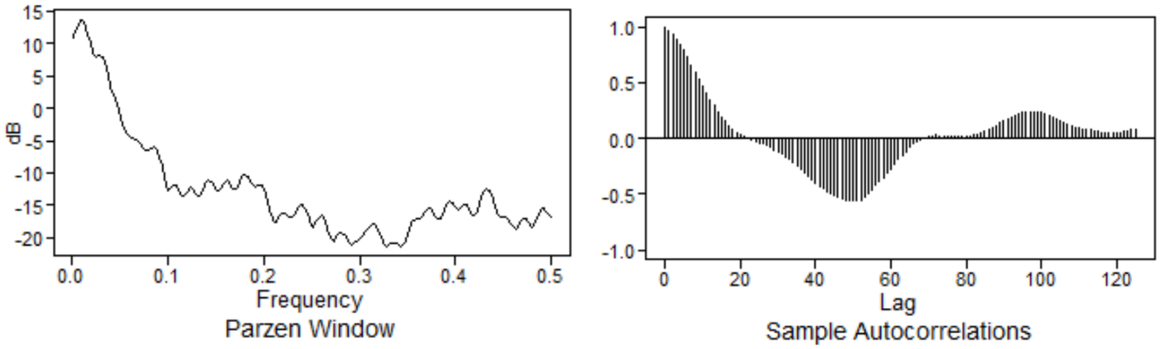
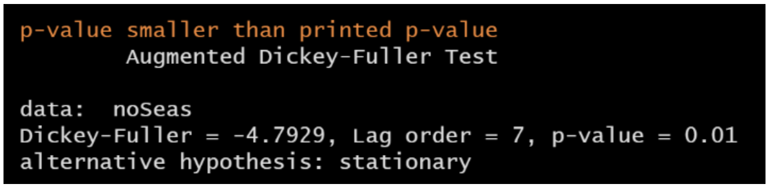


Fig. 6 Parzen Window and Sample Autocorrelation for the non-seasonal transformed influenza cases

Last step to prove if the transformed data is showing a stationary behavior is to perform a Dickey-Fuller Test. This test checks if a dataset satisfies the null hypothesis of having a unit root (which is equivalent to a non-stationary time series). Fig. 7 shows that the tests rejects the null hypothesis and hence we have evidence that the transformed data is stationary.



```
p-value smaller than printed p-value
        Augmented Dickey-Fuller Test

data:  noSeas
Dickey-Fuller = -4.7929, Lag order = 7, p-value = 0.01
alternative hypothesis: stationary
```

Fig. 7 Dickey-Fuller test on the transformed data

Based on these transformations and observations we can conclude that the logged version of the influenza cases from 2010 and 2019, for USA, are stationary after considering their 52-week seasonal behavior. We still have some evidence of a slight upward trend from Fig. 5 and hence we may consider evaluating this too during the model development process.

3. Modeling

3.1 ARIMA Models

There are many ways to model a time series. However, our focus was to build an **ARIMA** model. ARIMA stands for **A**uto **R**egressive **I**ntegrated **M**oving **A**verage. This model consists of three components: an **"AR"** component, an **"I"** component and a **"MA"** component. This section explains this model process in detail.

AR (Auto Regressive): An AR process is considered stationary and relies on "autoregression" which is a process of regressing a variable on its past values, as the past values of a time series can impact current and future time points. AR models tend to be more useful to describe a time series that progress in time. An AR model that considers past "p" time period values can be written as follows:

$$X_t = \beta + \phi_1 X_{t-1} + \phi_2 X_{t-2} + \cdots + \phi_p X_{t-p} + a_t$$

In the above equation, p is the order of AR model and this is interpreted as AR(p) model. β is a measure of the mean of the underlying process. ϕ_n terms are the autoregressive model coefficients associated with the lag terms. a_t is the white noise component not captured by the model. An Auto Regressive model of order 1, AR (1) can be written as:

$$X_t = \beta + \phi_1 X_{t-1} + a_t \qquad where\ \beta = (1 - \phi_1)\mu$$

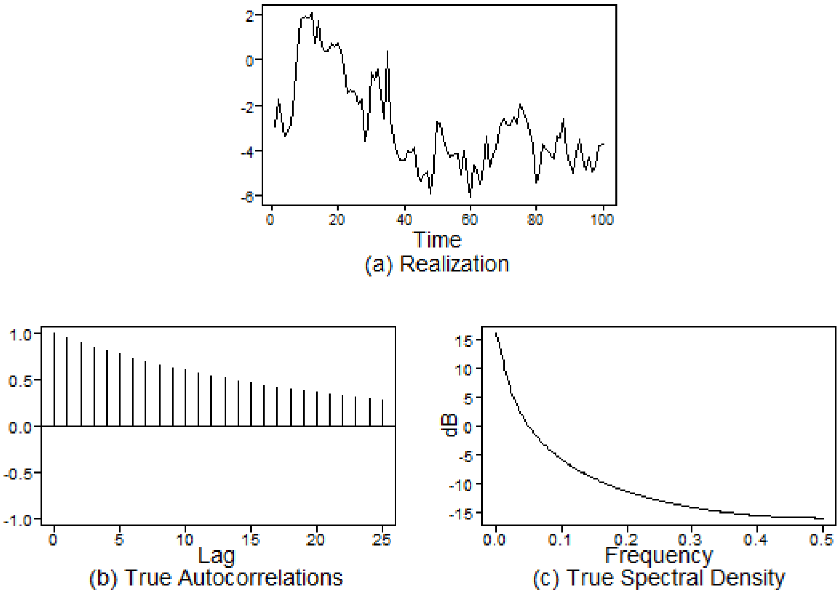
The AR (1) model says that the value of the process at time t (X_t) depends on the value of the process at time $t - 1$ plus a random noise component (and a constant). This is like simple linear regression model, but in this case, the "independent variable" is a value of the dependent variable at a prior time period. Above equation can be rewritten in zero mean form (considering $\mu = 0$) as follows:

$$X_t - \phi_1 X_{t-1} = a_t$$

Using the back-shift operator notation where $X_{t-1} = BX_t$, the above equation can be written as

$$X_t - \phi_1 BX_t = a_t \quad \Rightarrow \quad (1 - \phi_1 B)X_t = a_t \quad (or) \quad \phi(B)X_t = a_t \ \text{ where } \phi(B) = 1 - \phi_1 B$$

An example AR (1) process having a positive ϕ_1 value of 0.95 is shown in Fig. 8. This type of process is characterized by extended autocorrelations that don't die out quickly and by a peak in the Parzen Window at a frequency of 0. They are also characterized by wandering behavior as can be seen in the realization.



(a) Realization

(b) True Autocorrelations

(c) True Spectral Density

Fig. 8. Sample realization of an AR(1) process with positive ϕ_1.

MA (Moving Average): A MA process is considered stationary and this model defines that the value of a process at time t, say X_t is a linear combination of present and past noise components. This can be written as:

$$X_t = \mu + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \cdots - \theta_q a_{t-q}$$

Above equation can be interpreted as MA(q) model and in this equation, μ is the mean of the series, $a_t, a_{t-1}, \dots, a_{t-q}$ are the white noise error terms , θ_1, θ_2,..., θ_q are the parameters of the model which are non-zero, real and finite constants.

A MA (1) model which can be written as:

$$X_t = \mu + a_t - \theta_1 a_{t-1}$$

The same can be rewritten using the back-shift operator notation as shown below where Ba_t = a_{t-1}

$$X_t - \mu = (1 - \theta_1 B)a_t$$

Using zero mean form ($\mu = 0$) MA (1) process can be rewritten as:

$$X_t = (1 - \theta_1 B)a_t \;\; (or) \;\; X_t = \theta(B)a_t \text{ where } \theta(B) = (1 - \theta_1 B)$$

I (Integrated): This process is considered non-stationary and indicates that the next value of the time series is the same as the last time point with a small white noise component added to it. This can be represented both in standard form and backshift notation as shown below and is also called differencing since we are left with white noise if we difference the data using the immediately preceding value of the data itself.

$$X_t = X_{t-1} + a_t \, (or) \;\; (1 - B)X_t = a_t$$

The process above can be repeated multiple times and the number of times the observations are differenced is also called as degree of differencing and is represented as "d". Fig. 9 shows the sample realization, ACF and Parzen window for an Integrated process with $d = 1$. We can see that the characteristics are like the AR(1) process described above with a large positive ϕ_1 value. We can see the wandering behavior in the realization, the extended autocorrelations in the ACF plots and the peak in the Parzen window at $f = 0$. In fact, we also see that if we use $\phi_1 = 1$ in the AR(1) equation, we get the Integrated process. Hence, it is sometimes difficult to decipher between an AR(1) process with high ϕ_1 value and an Integrated process with $d = 1$. The difference between the two comes from the fact that the AR(1) process is a stationary process while the Integrated process is non-stationary. Differences are also manifested in the shape of the forecasts. An AR(1) process will have forecasts that converge towards the mean over time while the Integrated process will just forecast the previous time point into the future.

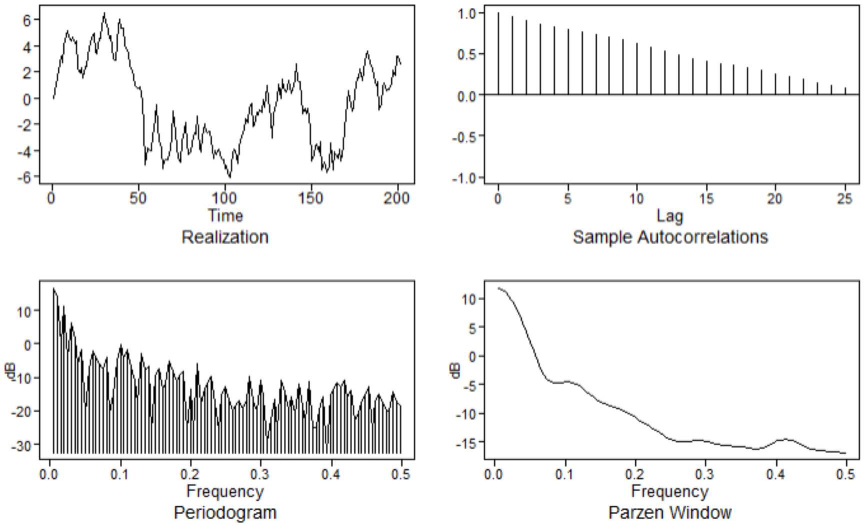


Fig. 9. Sample realization of an Integrated process of order d = 1.

Seasonal ARIMA model: Seasonality has already been discussed earlier and are another form of non-stationary process. They are also represented in the backshift notation form as shown below

$$(1 - B^s)X_t = a_t \text{ where s is the seasonal period}$$

However, real-life time series are generated from a combination of one or many of the above underlying processes. A generalized ARIMA(p, d, q) model with seasonality "s" can be represented as follows

$$\phi(B)(1 - \mathrm{B}^{\mathrm{s}})(1 - B)^d \, (X_t - \mu) = \theta(B)a_t$$

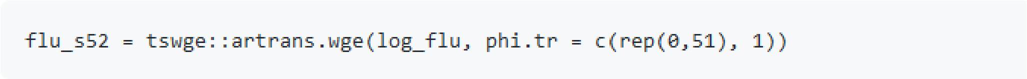
In this equation, $\phi(B)$ and $\theta(B)$ are the pth and qth order operators of AR, MA process respectively , $(1 - B)^d$ is the Integrated component where d determines degree of differencing and $(1 - \mathrm{B}^{\mathrm{s}})$ is the seasonal component. ARIMA models are considered powerful as they can capture the complex relationship in time series process. It considers the observation of lagged terms and error terms in the model and can also handle trend and seasonality in the data. This makes ARIMA the most preferred choice while modelling time series data.

3.2 Model Building

In this section, we will build an ARIMA model that best fits the flu data. In order to do this, we will dissect the data into its underlying AR, I, MA and Seasonal components described above. Using this model, we can forecast future flu outbreak numbers which might be helpful in preparing for the next flu season as outlined in Section 1. The steps to ID the right model for the data are outlined below.

Step 1: Identify if the data is stationary or not
Step 2: If the data is non-stationary, difference it to remove the non-stationary components (this could include seasonality "s" or differencing "d" or a combination of both)
Step 3: Once the data has been made stationary, the remainder can be modeled with a stationary ARMA process (combination of only AR "p" and MA "q" terms).
Step 4: Evaluate the model for goodness of fit.

The first step in developing an ARIMA model is to figure out whether the data is generated from a stationary process or a non-stationary process. As covered in 2.2, the model has a seasonal component with the behavior repeating every 52 weeks (one year). Hence, we must first remove this seasonality from the data in order to model it further. This is done using the code shown in Fig. 10 which is similar to what was done earlier in Fig. 6. The phi.tr parameter consists of 51 zeros followed by one. This essentially subtracts the value of the number of infections from 52 weeks back from each observation. What is left (bottom image in Fig. 10) is the non-seasonal flu infections i.e. the number of flu infections after taking the seasonal effect out. From the ACF plot in Fig. 10 (bottom right), we see that the non-seasonal data has an ACF has some extended autocorrelations. This can be representative of an ARIMA process with a large positive ϕ value with $d = 0$. It could also be representative of an ARIMA process with $d = 1$ since it still shows some signs of extended autocorrelation. We will decipher the best fit model next.



```
flu_s52 = tswge::artrans.wge(log_flu, phi.tr = c(rep(0,51), 1))
```

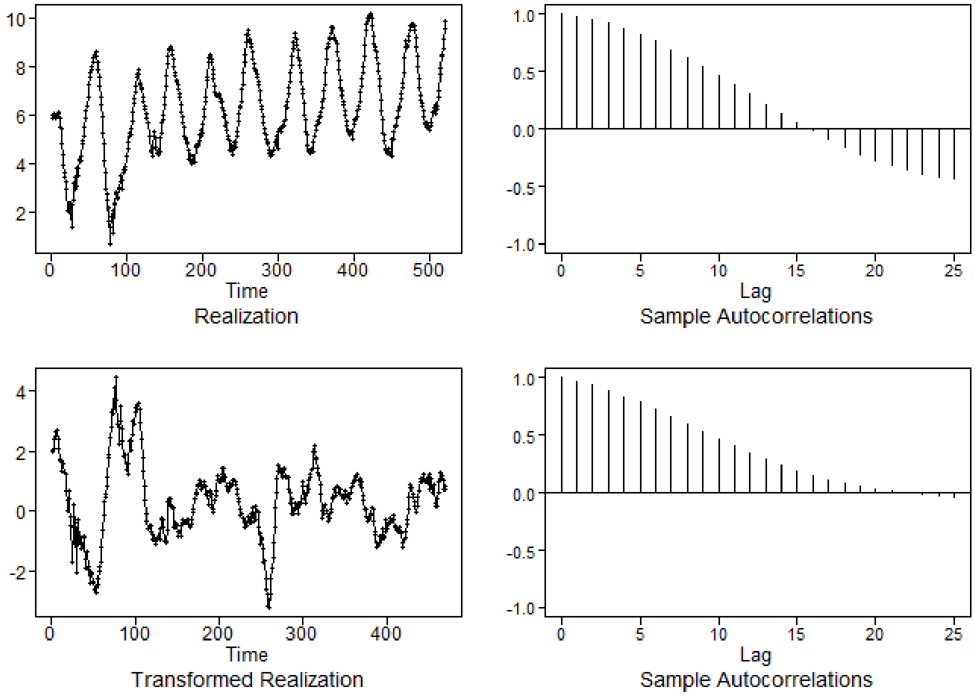
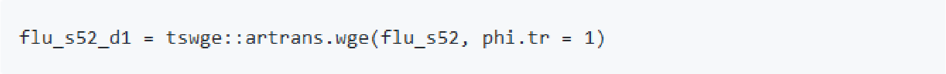


Fig. 10. Removing Seasonality from the Data

In order to determine the right fit (whether the model is an ARIMA model with $d = 1$ or an ARIMA model with $d = 0$), we will build both models and determine the right model based on the fit statistics and white noise characteristics. To build the first candidate model (i.e. ARIMA with $d = 1$), we must take the difference term out of the non-seasonal data so that we can model the remainder of the data with a stationary ARMA process. This can we done in a similar manner to what we did when removing the seasonality and is shown in Fig. 11. This time, phi.tr is simply "one" since we are just removing the different with a lag of "one". The ACF plots comparing the non-seasonal data (top right) to the differenced non-seasonal data (bottom right) show that the "extended" auto-correlation that was observed has been removed. The resultant data looks very close to a stationary process and can be modeled with an ARMA model.



```
flu_s52_d1 = tswge::artrans.wge(flu_s52, phi.tr = 1)
```

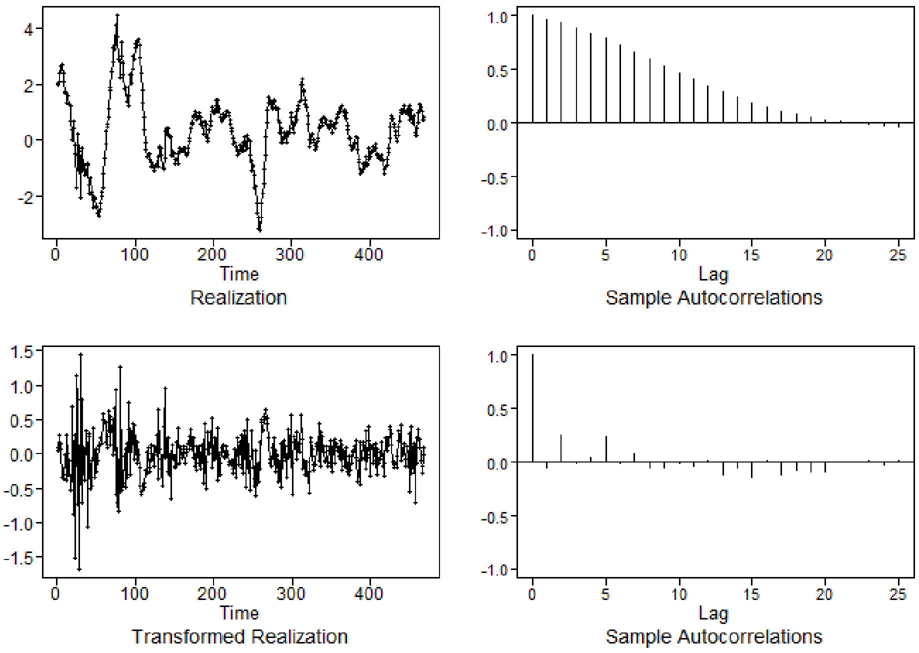


Fig. 11. Removing First Difference from the Non-Seasonal Flu Data

The second candidate model (ARIMA with $d = 0$) was already shown to be non-stationary in 2.2 after removing the seasonal term. Since no further transformation is needed for both our candidate models in order to make them stationary, we can proceed to the next step (step 3) of identifying the stationary ARMA components (p, and q). We can use a grid search approach for this by varying both p and q and then picking the best model based on an evaluation metric. The traditional evaluation metrics that are used in Time Series Models are called AIC and BIC. These terms essentially balance underfitting and overfitting by considering both the residuals or errors in the model (underfitting) and the complexity of the models (overfitting). Both use slightly different penalty terms for overfitting, and in general the BIC metric favors a more persimmons model that leans towards underfitting vs. the AIC metric which favors a more complex model leaning more towards overfitting.

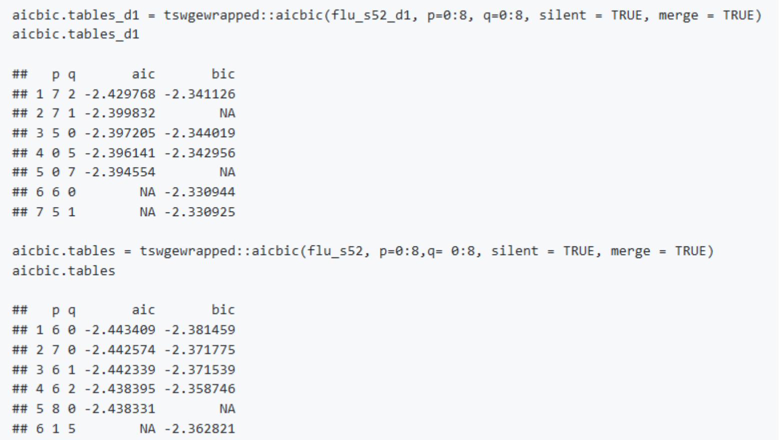
We ran the AIC and BIC calculations for both our candidate models and the results are shown in Fig. 12. The results are sorted by AIC and only show the top 5 ARMA fits for each candidate model. The model with differencing ($d = 1$) can be best modeled with $p = 7$ and $q = 2$ resulting in a final model ARIMA(7, 1, 2) with $s = 52$. The model without differencing ($d = 0$) can be best modeled with $p = 6$ and $q = 0$ resulting in a final model ARIMA(6, 0 , 0) with $s = 52$. The resulting model coefficients are shown in Fig. 14 and Fig. 14. The equation for the final models is as follows:

ARIMA(7, 1, 2) with s = 52

$$\mathrm{X_t}\ (1-\mathrm{B}^{52})\ (1-\mathrm{B})(1-0.0538\ \mathrm{B}-1.1120\ \mathrm{B}^2+0.0465\ \mathrm{B}^3+0.2309\ \mathrm{B}^4-0.3143\ \mathrm{B}^5+0.0195\ \mathrm{B}^6+0.2811\ \mathrm{B}^7=1-0.1261\mathrm{B}-0.8739\mathrm{B}^2)\mathrm{a_t}\ \text{with}\ \mathrm{var}(\mathrm{a_t})=0.0844$$

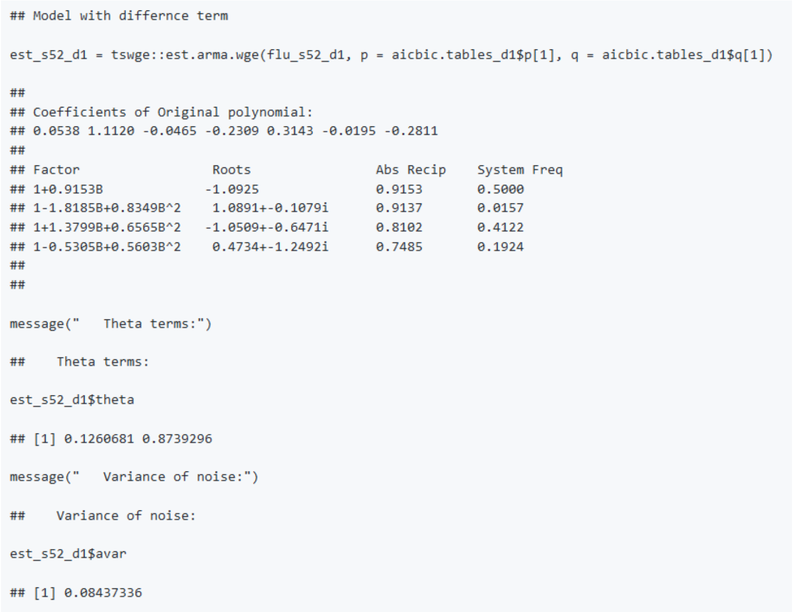
ARIMA(6, 0, 0) with s = 52

$$\mathrm{X_t}\ (1-\mathrm{B}^{52})\ (1-0.9321\mathrm{B}-0.2911\ \mathrm{B}^2+0.3006\ \mathrm{B}^3-0.0413\ \mathrm{B}^4-0.2686\ \mathrm{B}^5+0.2860\mathrm{B}^6)=\mathrm{a_t}\ \text{with}\ \mathrm{var}(\mathrm{a_t})=0.0844$$



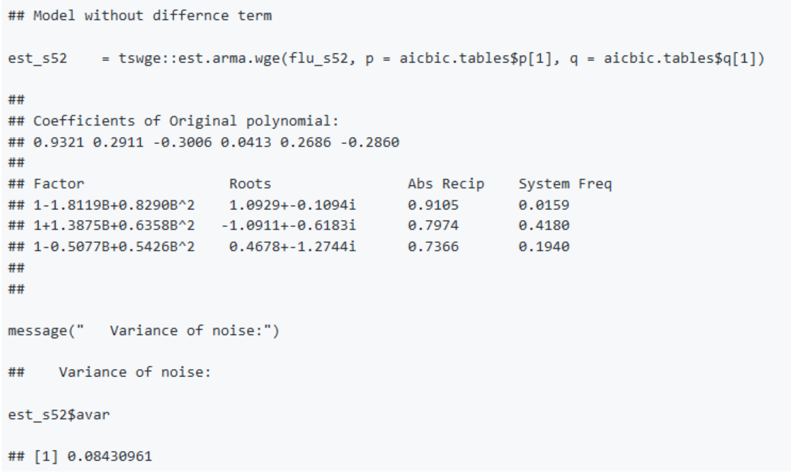
```
aicbic.tables_d1 = tswgewrapped::aicbic(flu_s52_d1, p=0:8, q=0:8, silent = TRUE, merge = TRUE)
aicbic.tables_d1

##   p q       aic       bic
## 1 7 2 -2.429768 -2.341126
## 2 7 1 -2.399832        NA
## 3 5 0 -2.397205 -2.344019
## 4 0 5 -2.396141 -2.342956
## 5 0 7 -2.394554        NA
## 6 6 0        NA -2.330944
## 7 5 1        NA -2.330925

aicbic.tables = tswgewrapped::aicbic(flu_s52, p=0:8,q= 0:8, silent = TRUE, merge = TRUE)
aicbic.tables

##   p q       aic       bic
## 1 6 0 -2.443409 -2.381459
## 2 7 0 -2.442574 -2.371775
## 3 6 1 -2.442339 -2.371539
## 4 6 2 -2.438395 -2.358746
## 5 8 0 -2.438331        NA
## 6 1 5        NA -2.362821
```

Fig. 12. Model ID for Stationary Component of the two Candidate Models



```
## Model with differnce term

est_s52_d1 = tswge::est.arma.wge(flu_s52_d1, p = aicbic.tables_d1$p[1], q = aicbic.tables_d1$q[1])

##
## Coefficients of Original polynomial:
## 0.0538 1.1120 -0.0465 -0.2309 0.3143 -0.0195 -0.2811
##
## Factor                 Roots                Abs Recip    System Freq
## 1+0.9153B             -1.0925               0.9153       0.5000
## 1-1.8185B+0.8349B^2    1.0891+-0.1079i      0.9137       0.0157
## 1+1.3799B+0.6565B^2   -1.0509+-0.6471i      0.8102       0.4122
## 1-0.5305B+0.5603B^2    0.4734+-1.2492i      0.7485       0.1924
##
##

message("   Theta terms:")

##    Theta terms:

est_s52_d1$theta

## [1] 0.1260681 0.8739296

message("   Variance of noise:")

##    Variance of noise:

est_s52_d1$avar

## [1] 0.08437336
```

Fig. 13. Stationary Model Coefficients for the ARIMA(7, 1, 2) model with s = 52



```
## Model without differnce term

est_s52    = tswge::est.arma.wge(flu_s52, p = aicbic.tables$p[1], q = aicbic.tables$q[1])

##
## Coefficients of Original polynomial:
## 0.9321 0.2911 -0.3006 0.0413 0.2686 -0.2860
##
## Factor                 Roots                Abs Recip    System Freq
## 1-1.8119B+0.8290B^2    1.0929+-0.1094i      0.9105       0.0159
## 1+1.3875B+0.6358B^2   -1.0911+-0.6183i      0.7974       0.4180
## 1-0.5077B+0.5426B^2    0.4678+-1.2744i      0.7366       0.1940
##
##

message("   Variance of noise:")

##    Variance of noise:

est_s52$avar

## [1] 0.08430961
```

Fig. 14. Stationary Model Coefficients for the ARIMA(6, 0, 0) model with s = 52

3.3 Model Evaluation

Next, we will evaluate the models for goodness of fit (step 4 in the process) using two evaluation metrics. First, we will look at the residuals from the models and see if they are consistent with white noise. If they are, then it means that we have captured all the signal from the data and that the fit is good. Figure 13 shows the residuals for the two candidate models along with their ACF plots. Both the models show that the residuals are consistent with white noise (the ACF values for any lag greater than 0 is less than the significance limit marked by the dark blue region). We also conducted a statistical (Ljung-Box) test to check for white noise and none of these tests could reject the null hypothesis that the residuals were consistent with white noise. Hence both models passes this assessment.

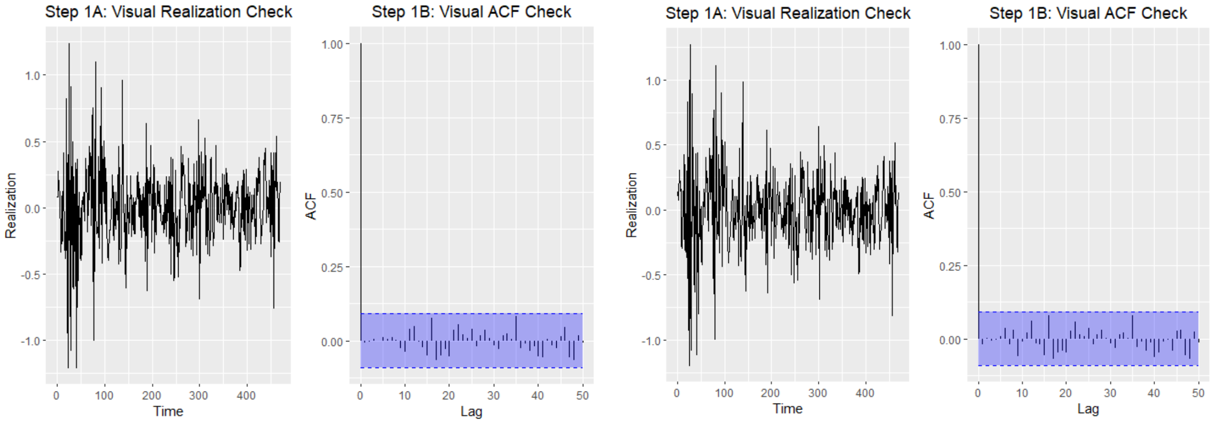


Fig. 15. White Noise Test for the Residuals: ARIMA(7, 1, 2) with s = 52 (left), ARIMA(6, 0, 0) with s = 52 (right)

Next, we will evaluate the two candidate models based on how accurately they are able forecast future values. Like regular machine learning models, we want to be use a cross-validation technique (such as k-fold) so that we can get a more generalized sense of the model's performance rather than basing it on only comparing to a single region of data. However, unlike regular machine learning models, we can not randomly divide the data into folds since the data is serially correlated and we want to preserve the correlation structure. In addition, we want to make sure that we don't use future data to train the model since that would be akin to data leakage. Hence, we use a windowed approach to the cross-validation as outlined in Fig. 16. In this technique, we choose to train the model on a small window of continuous data (blue sections in the figure) and then forecast the next few data points (red sections in the figure). These forecasted data points are then compared to the known actuals from that period and the error is calculated. The error metric that was used in our case study is called Average Square Error or ASE (another name for Mean Squared Error or MSE). The training window is then be moved (rolled) and the process is repeated for another period. This can be repeated a few times (k) and we can get an equivalent of k-fold cross validation for time series that preserves the serial correlation and prevents data leakage.

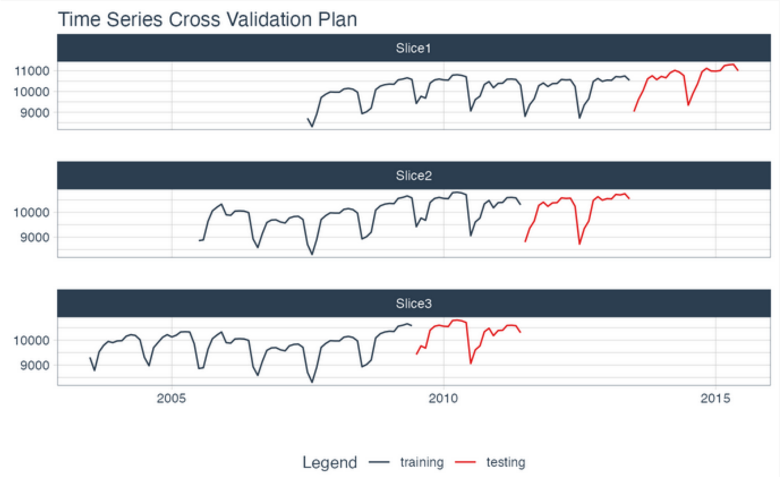


Fig. 16. Example of Rolling Window Cross Validation Technique for Time Series Model Evaluation [3]

The training window in our case study was three years (156 weeks) and the forecast horizon was one year (52 weeks). Given the size of our dataset, we could repeat this process seven times giving us an equivalent of 7-fold cross validation. The forecasts from these seven windows and their comparison to the actual data is shown in Fig. 17. Clearly, the ARIMA(6, 0, 0) with s = 52 models is performing better than the ARIMA(7, 1, 2) with s = 52 model. The addition of the difference term in the second model adds a trend to the forecast which is taking the forecasts away from the actual values. Fig. 18 shows the same behavior from the point of the error metric. Again, we clearly see that the model without the difference terms is performing much better (lower ASE) compared to the model with the difference term. Given this performance comparison, we will choose the model without differencing as our final model.

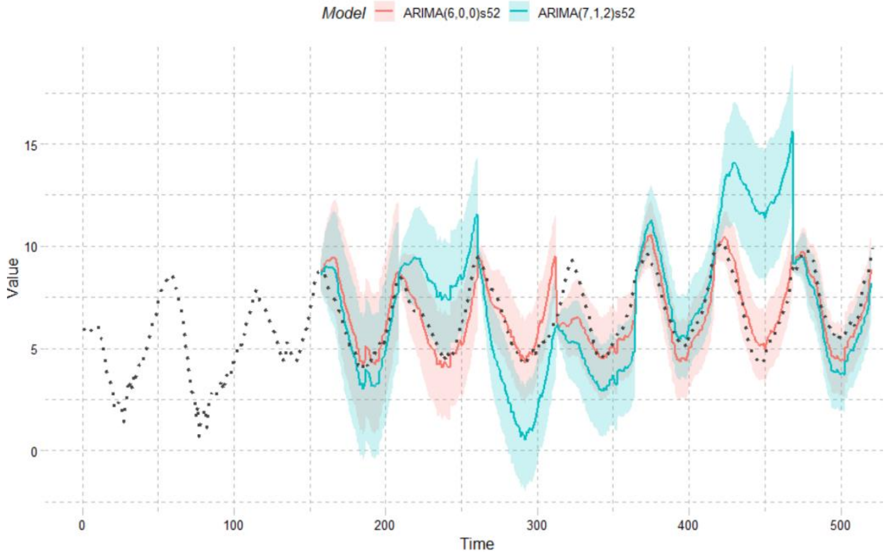


Fig. 17. Window Forecast Comparison of the two Candidate Models to the Actual Data (dotted).

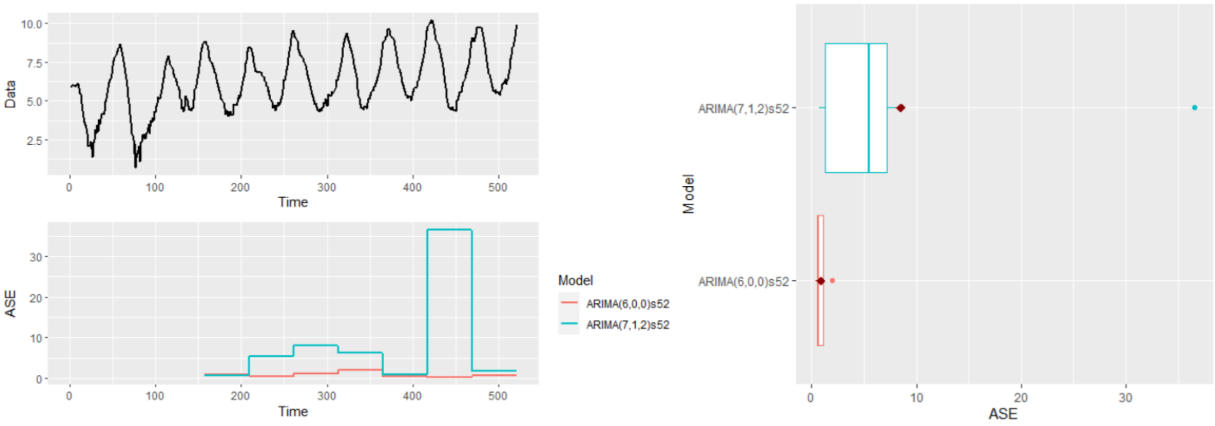


Fig. 18. The Averaged Square Error (ASE) of the two Candidate Models: Over time (left), Boxplot (right)

4. Forecasting

We can now utilize the select ARIMA(6, 0, 0), with s = 52, model to predict the number of influenza cases for the future weeks. Usually, the performance of the predictions from a statistical model degrades as we forecast further into the future. Hence, we want to balance out how far ahead we forecast with the business need. We think that a one-year forecast window should be enough for this purpose as it will give the authorities enough time to make the necessary preparations for the upcoming flu season.

Fig. 19 shows the prediction of the log of the influenza cases for year 2020. The red line shows the prediction based on the ARIMA(6,0,0) model with seasonality of 52 weeks. The red shaded area represents the 95% confidence interval for the model's prediction. We can notice the predictions are following a pattern that is consistent with the historical cyclic pattern.

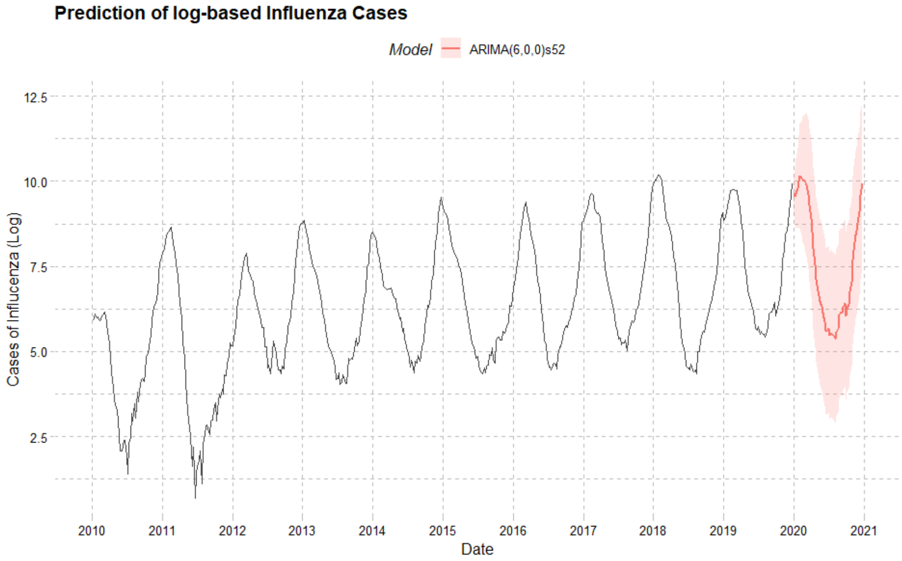


Fig. 19 Prediction of log-based influenza cases

As the log values are difficult to interpret in a real-world situation, we processed the final predictions by removing the log transformation. Fig. 20 shows the number of influenza cases for year 2020 on the original scale. We can notice the model shows a peak at about 25,000 cases that is slightly lower than the peak happened year 2018 and much higher than the peak for year 2019.

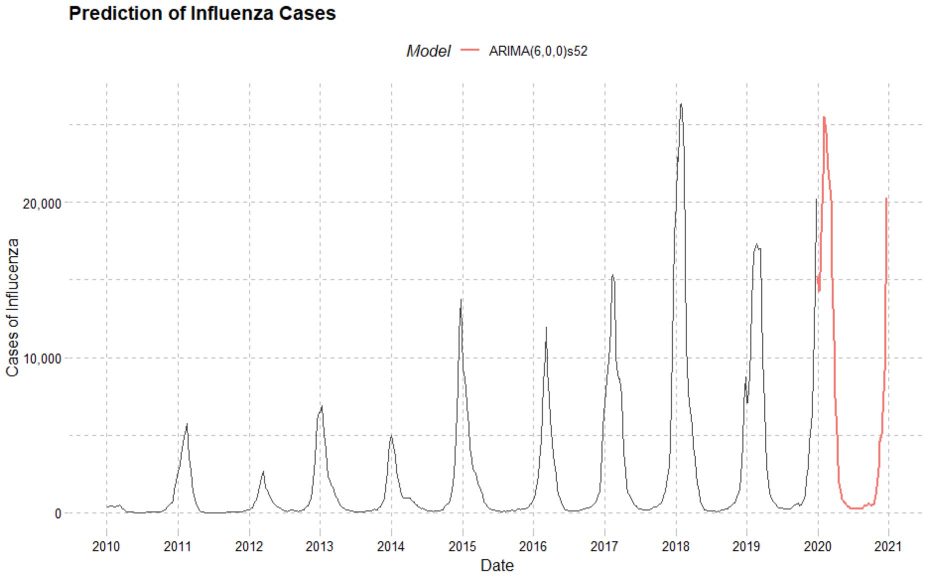


Fig. 20 Prediction of influenza cases

Table 2 shows the comparison of predicted cases for 2020 with the actual cases in 2019, in groups of four weeks. The table shows how the model predicts that there will be more influenza cases in 2020 compared to 2019. We can also notice that the ending number of cases of 2019 (44,016) is higher than the number of cases at the beginning of 2019 (40,272). This is an indication of harsher influenza season for 2020 and the model was able to pick up on this pattern. In contrast, the model forecasts slight lower number of cases of the summer of 2020 compared to the summer of 2019.

Table 2 Comparing 2020 predicted cases with actual cases in 2019

Week Range	2019 Actual Cases	2020 Predicted Cases	% Change
1-4	40,272	69,644	+72.9%
5-8	68,204	95,432	+39.9%
9-12	57,249	67,345	+17.6%
13-16	16,556	17,380	+5.0%
17-20	3,617	3,536	-2.2%
21-24	1,734	1,651	-4.8%
25-28	1,099	1,044	-5.0%
29-32	991	951	-4.0%
33-36	1,702	1,656	-2.7%
37-40	2,141	2,106	-1.6%
41-44	4,796	4,766	-0.6%
45-48	19,012	18,991	-0.1%
49-52	44,016	44,081	+0.1%

5. Conclusion

The goal of this case study was to use historical data for influenza to develop a model that could be used to forecast the future values. This would eventually help the authorities in planning the supply of vaccines, hand sanitizers, masks, hospital beds etc. This study used the influenza data for the USA from 2010 to 2019 and this indicated strong seasonality. There was some evidence of an upward trend as well (though minor). Both these were considered during the model building process and after careful evaluation, the model with seasonality only was chosen to be the best fit model.

Using this best fit model, a forecast for the number of cases was made for 2020. It is predicted that the start of 2020 will have a harsher influenza outbreak than the start of 2019. Hence the authorities should plan to have a larger stock of vaccines and medical supplies at hand as it might very well be needed to reduce the severity of the outbreak. The summer months and the winter months of 2020 are expected to have roughly the same number of outbreaks as 2019, hence the supplies can be maintained at the same rate as 2019 during these periods.

Finally, it is important to again reiterate that the assumption made in this study is that the underlying process that causes the flu is not changing drastically from 2019 to 2020. If, however, this does happen (possibly due to a pandemic outbreak or another reason), then the model forecast may not be accurate and representative of the actual numbers that are likely to be observed. Also, we should not use this model to predict too far out into 2021 and beyond as predictions will not be accurate so far out into the future. Instead, we should use the data from 2020 (as it becomes available) to start building the model to forecast the outbreaks in 2021. This would give us the most accurate estimate for future years (beyond 2020) and will also be able to consider the likely underlying changes in the influenza generation process in 2020 if there are any.

6. References

[1] World Health Organization, "Flumart," [Online]. Available: https://apps.who.int/flumart. [Accessed 01 10 2020].

[2] World Health Organization, "fluiID," 08 12 2010. [Online]. Available: https://www.who.int/influenza/surveillance_monitoring/fluid/FluID_Flyer.pdf. [Accessed 01 10 2020].

[3] M. Dancho, "Business Science," [Online]. Available: https://business-science.github.io/timetk/reference/time_series_cv.html. [Accessed 17 10 2020].

7. Appendix

The entire code used for this case study is available in the following GitHub repository:
https://github.com/ngupta23/ds7333_qtw/tree/master/case_study_4/analysis/master