Task D - Influenza Forecasting

Statistical Analysis of Big Data

Amanda Magzal 207608647

Contents

1 Introduction

1.1 Background

Flu activity forecasting involves predicting in advance when increases in influenza (flu) activity will occur. Unlike CDC's (Centers for Disease Control and Prevention) traditional influenza surveillance systems, which measure influenza activity after it has occurred, flu forecasting offers the possibility to look into the future and plan ahead. This is important because flu places a significant disease burden on the U.S. population each year. The potential benefits of flu forecasting are immense. When experts can accurately predict - similar to a weather forecast - when significant increases in flu activity will occur, the ability to plan ahead and more effectively implement disease mitigation strategies becomes possible. For example, disease forecasting could help determine when best to schedule vaccination clinics or educational campaigns; it could help decide the optimal time to distribute influenza antiviral medications; and it could help doctor's offices, hospitals, businesses and schools plan for the impact of flu on daily operations.

1.2 Data

Outpatient Illness Surveillance - Information on patient visits to health care providers for influenza-like illness is collected through the U.S. Outpatient Influenza-like Illness Surveillance Network (ILINet). ILINet consists of more than 2,900 outpatient healthcare providers in all 50 states, Puerto Rico, the district of Columbia and the U.S. Virgin Islands reporting over 36 million patient visits each year. Each week, approximately 2,000 outpatient healthcare providers around the country report data to CDC on the total number of patients seen and the number of those patients with influenza-like illness (ILI) by age group (0-4 years, 5-24 years, 25-49 years, 50-64 years, and over 65 years). For this system, ILI is defined as fever (temperature of 37.8 Celsius or greater) and a cough and/or a sore throat in the absence of a known cause other than influenza.

The data file used in this project includes weekly reports of ILI (Influenza Like Illness) in USA for the years 1997-2016. It was downloaded from here and is part of the CDC FluView report, which provides weekly influenza surveillance information in the United States.

The variables are YEAR, WEEK, AGE 0-4, AGE 25-49, AGE 25-64, AGE 5-24, AGE 50-64, AGE 65, ILITOTAL, TOTAL PATIENTS, and ILI percent.

The missing values of ILIp will be filled with the average ILIp of the corresponding week across all years. For instance, the ILIp of week 21 in 1998 will be the average ILIp of week 21 in all other years.

1.3 Main Goals

We define ILI percent to be (ILITOTAL/TOTAL PATIENTS) × 100% and refer to it as ILIp. We also define ILIp season to start from week 40 of the calendar year till week 39 of the next year. For instance, season 97-98 starts from week 40 of 1997 and ends on week 39 of 1998. There are 19 seasons in total.

There are two main goals:

1. Clustering ILIp seasons.
2. Forecasting ILIp 4 weeks ahead.

2 Clustering

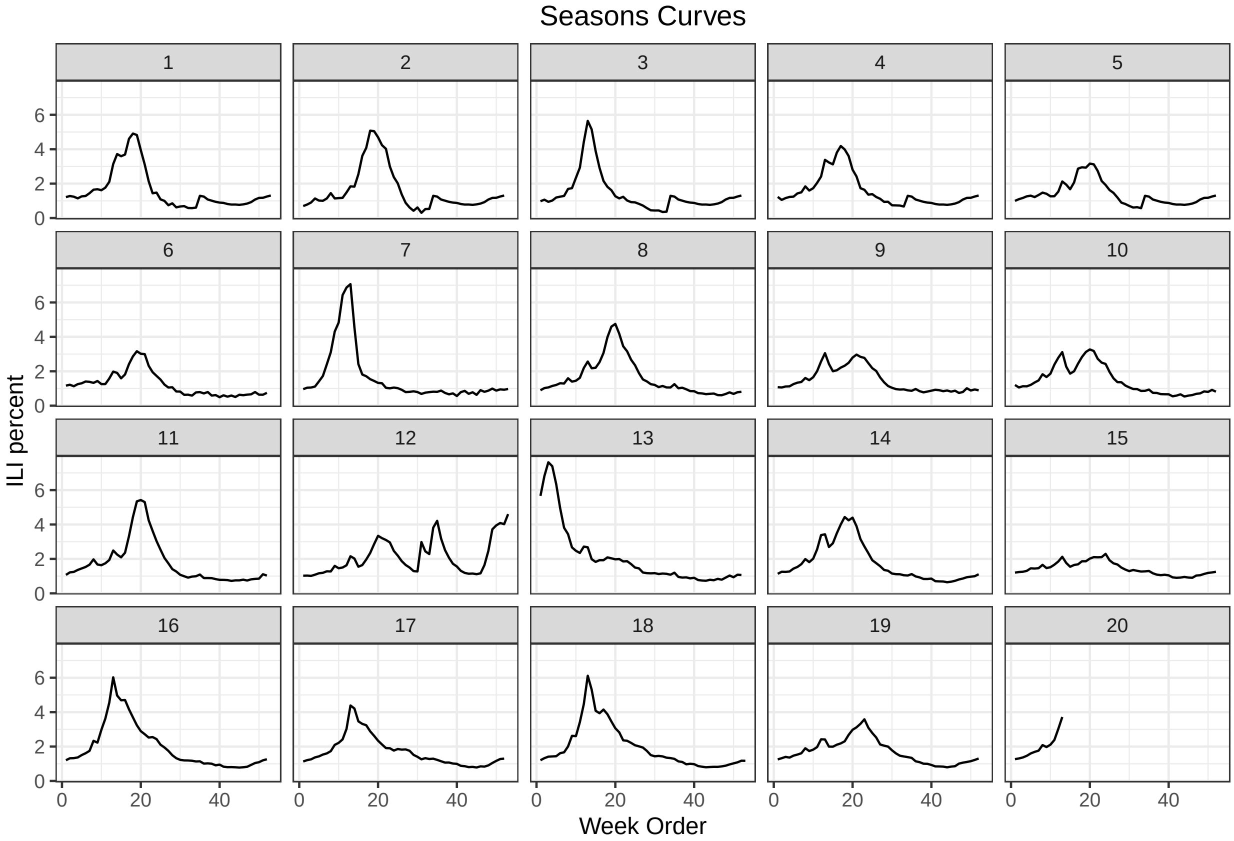
In this section, I attempt to find interesting clusters of the ILIp seasons.

For convenience, I add two columns to the data frame:

- `Season`: The number of the season out of the 19 seasons in the data. For instance, season 97-98 as defined before will be season 1.
- `WEEK_ord`: The order of the weeks within a season. For instance, week 40 in season 97-98 will be week 1 and so on.

2.1 Seasons Curves

I create a grid of all the seasons' curves to visually identify similar patterns.



We see that most seasons have a similar pattern: in the beginning of the season, the ILIp is around 1. It increases as the weeks go by and reaches it's highest, between 4% to 6%, depending on the season. This usually happens somewhere between week 12 to week 22 (The winter months). Then it decreases again to around 1%.

There are a few interesting points worth mentioning:

- Season 7 (2003-2004) had a higher ILIp than the previous years. This seems to be because the predominant flu virus was A(H3N2).
- Season 12 (2008-2009) had a very unusual curve. This is due to the swine flu pandemic that occurred during this time.
- Season 15 (2011-2012) had a relatively flat curve with low ILIp throughout the entire season.

Note that we have only the beginning of the data for season 20, thus it will be dropped for now.

2.2 Identifying Clusters

In order to find interesting clusters of the ILIp seasons, I use the following procedure:

1. Create a matrix of distances D between the ILIp curves.
2. Get the first and second principal coordinates using **Multidimensional Scaling**.
3. Find Clusters using the **PAM Clustering Algorithm**.

First, I give a brief explanation of Multidimensional Scaling and the PAM Clustering Algorithm. Then, I apply the above procedure using two different distance measures: Euclidean and Manhattan Distance. For each measure I find $K = 2, 3, 4, 5$ clusters and compare the results.

2.2.1 MDS and PAM

Multidimensional Scaling (MDS)

MDS deals with "fitting" the data in a low-dimensional space with minimal distortion to the distances between original points.

The algorithm works as follows:

1. For given matrix of distances D, compute matrix B where

$$b_{ij} = -\frac{1}{2}\Big(d_{ij}^2 - \frac{1}{n}\sum_{j=1}^{n} d_{ij}^2 - \frac{1}{n}\sum_{i=1}^{n} d_{ij}^2 + \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}^2\Big)$$

2. Perform SVD of B, $B = V\Lambda V^T$; let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$.
3. Retain q largest eigenvalues, $q \leq p$, set $\Lambda_1 = \text{diag}\{\lambda_1, \ldots, \lambda_n, 0, \ldots, 0\}$.
4. The new q–dimensional data matrix representation is $Y = V\Lambda_1^{1/2}$. The rows of the matrix Y are called the **principal coordinates of X in q-dimensions**.

The PAM Clustering Algorithm

PAM stands for "partition around medoids". The algorithm is intended to find a sequence of objects called medoids that are centrally located in clusters.

The goal of the algorithm is to minimize the average dissimilarity of objects to their closest selected object. Equivalently, we can minimize the sum of the dissimilarities between object and their closest selected object.

The algorithm has two phases:

1. **BUILD** - a collection of k objects are selected for an initial set S.
2. **SWAP** - the algorithm tries to improve the quality of the clustering by exchanging selected objects with unselected objects.

2.2.2 Finding Clusters

Using the procedure explained above, I create clusters for $K = 2, 3, 4, 5$ using two different distance measures:

- Euclidean Distance

The euclidean distance between two points $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$ in n-dimensional space is given by

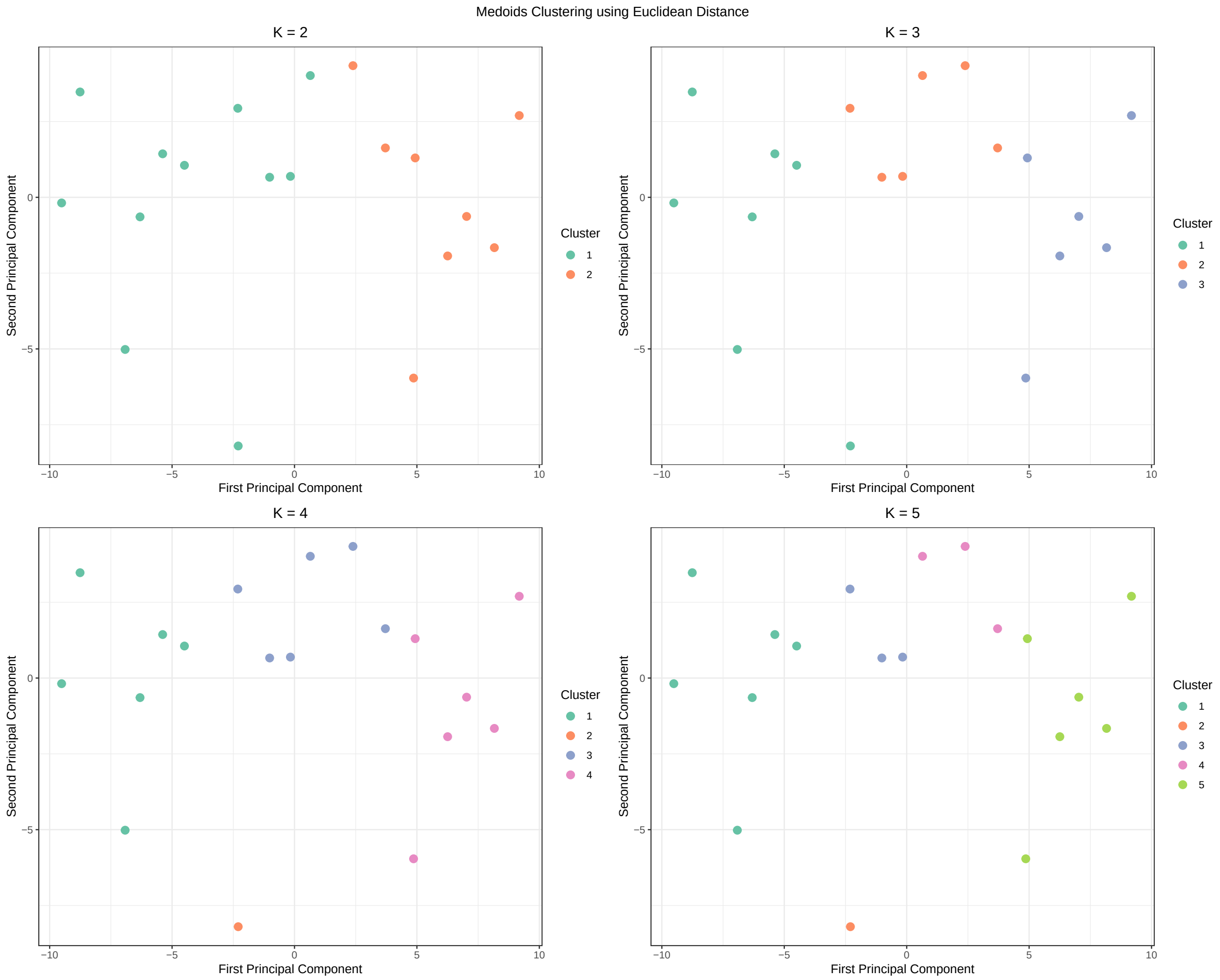
$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

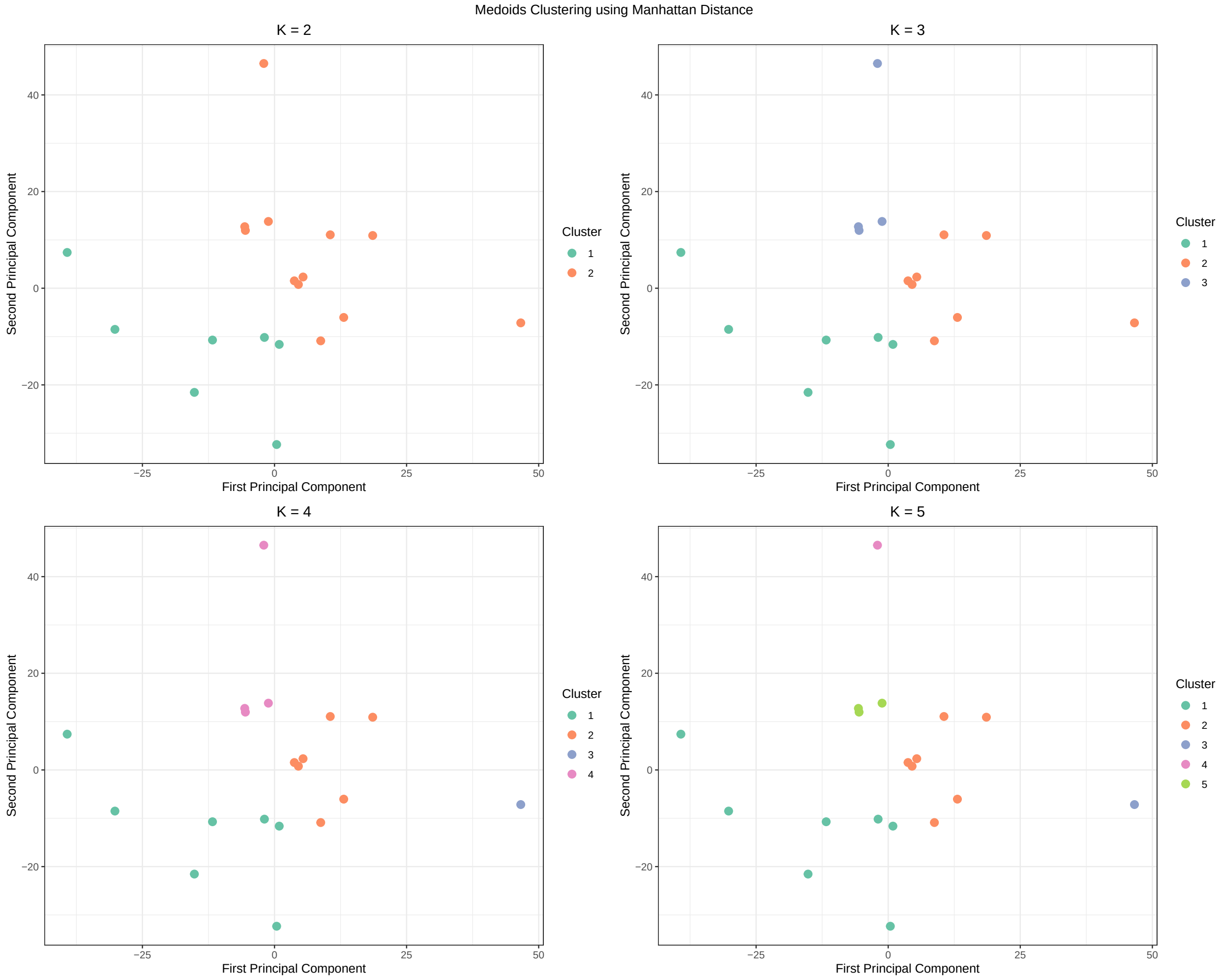
- Manhattan Distance

The manhattan distance between two points $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$ in n-dimensional space is given by

$$d(x, y) = \sum_{i=1}^{n} |x_i - y_i|$$

Clustering Results



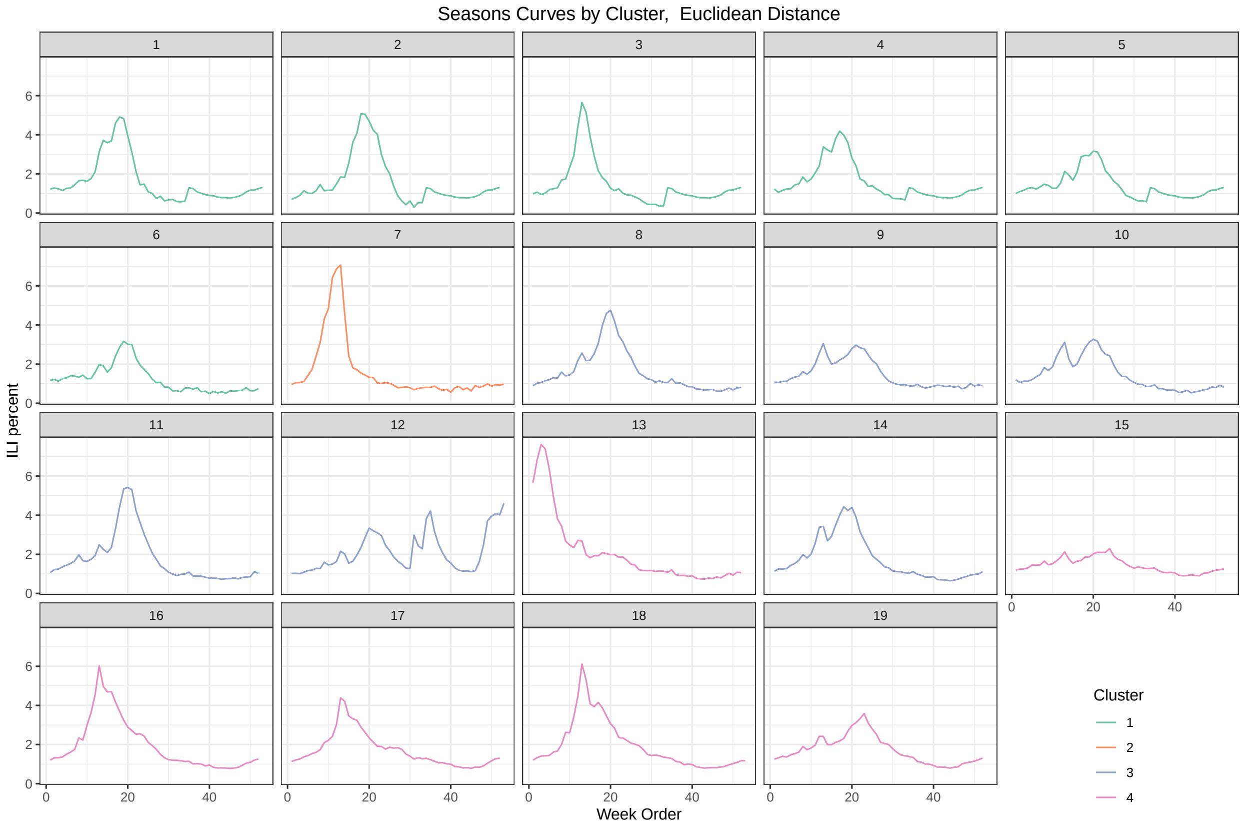


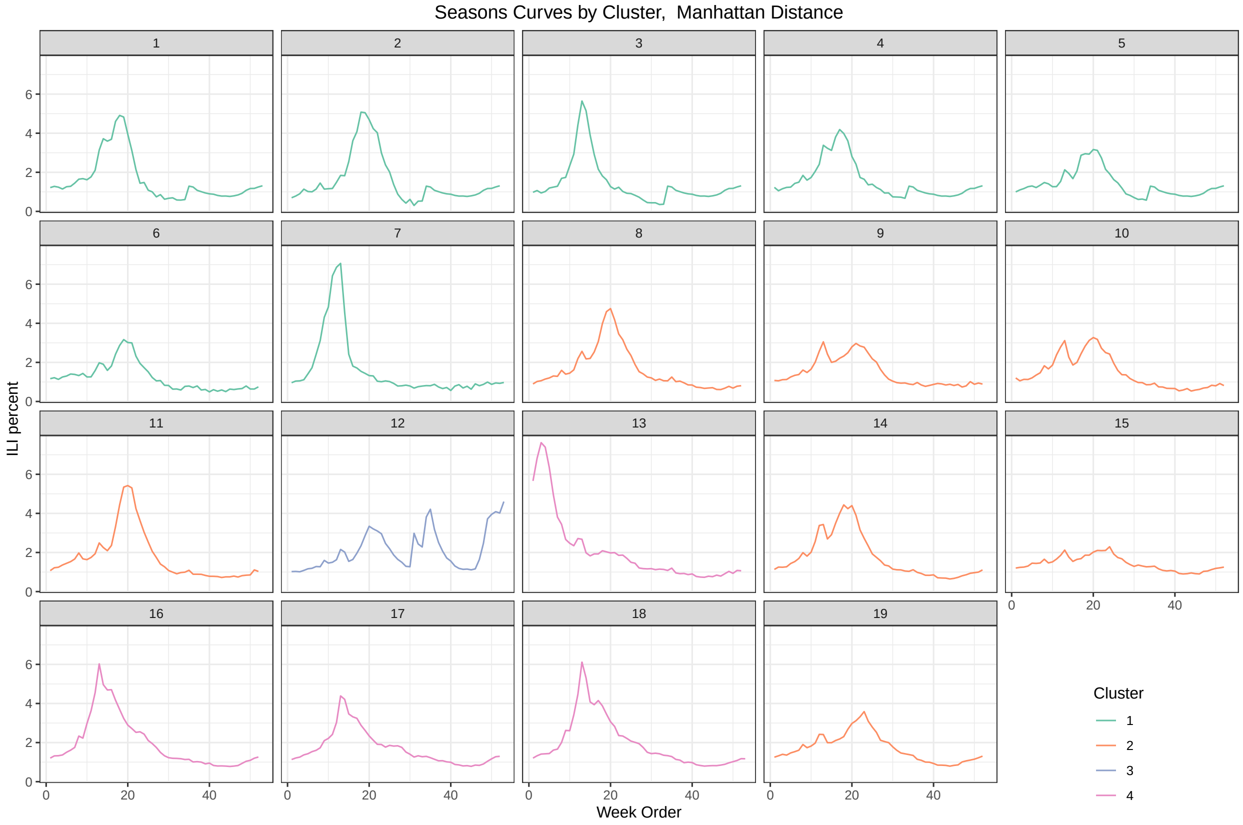
2.2.3 Comparing the Results

After analyzing the results, I found that the most interesting results were obtained for $K = 4$.

- We see that overall, consecutive seasons tend to be clustered together. For instance, both methods assigned the first 6 seasons to the same cluster.
- Using euclidean distance, season 7 was assigned to its own cluster. This might because it had a higher percentage at its peak, compared to other seasons.
- Using manhattan distance, season 12 was assigned to its own cluster. This is probably due to the swine flu pandemic that occurred during that time and resulted in a very unusual curve.
- Using manhattan distance, cluster 2 seems to include seasons where the ILIp peaked around week order 20 and had lower average percentages, while cluster 1 includes ones where it peaked around week 18 and had higher average percentages. Cluster 4, includes seasons that had their peak ILIp the earliest, around week 13.

The seasons' curves colored by cluster (for $K = 4$) are shown below.


Seasons Curves by Cluster, Euclidean Distance
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
ILI percent
0
2
4
6
0
20
40
Week Order
Cluster
1
2
3
4

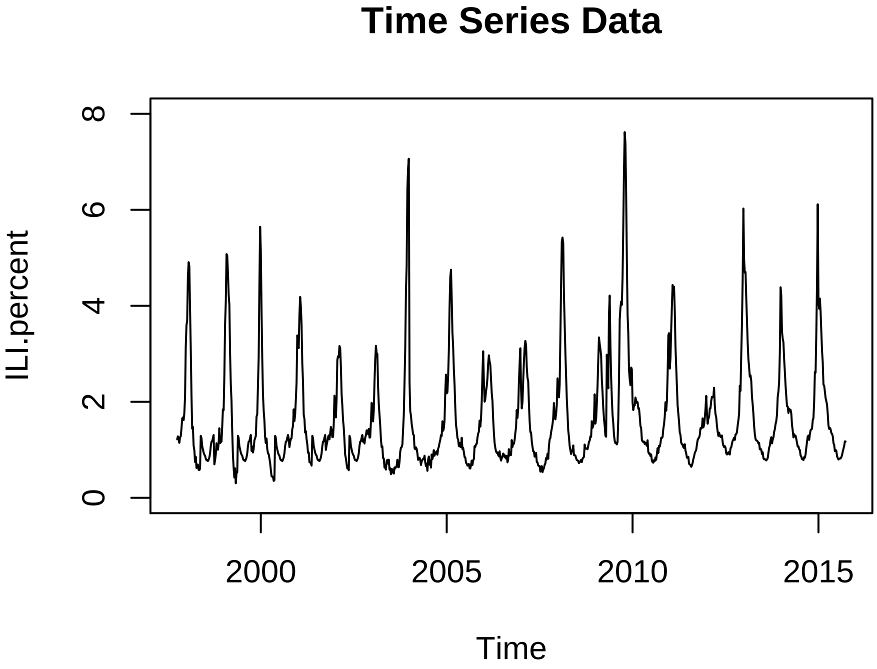

Seasons Curves by Cluster, Manhattan Distance
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
ILI percent
0
2
4
6
0
20
40
Week Order
Cluster
1
2
3
4

3 Forecasting

In this section, I attempt to develop a method that forecasts ILIp 4 weeks ahead. First, I show the predictions for an entire season and then I show for only 4 weeks.

3.1 Time Series Data

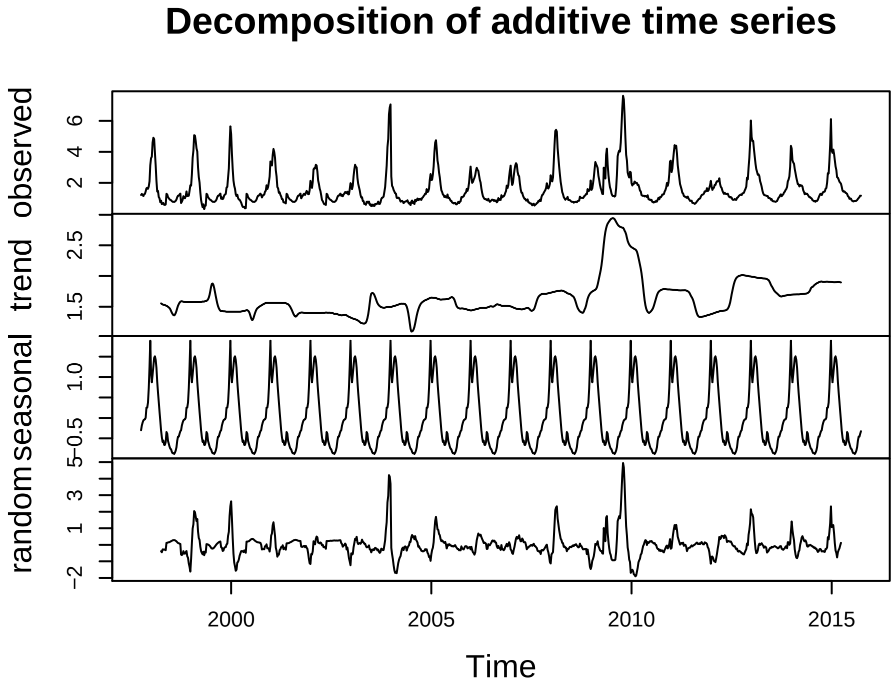
The train data includes the first 18 season and the rest (season 19 and part of season 20) is part of the test data.



A seasonal time series consists of the following components:

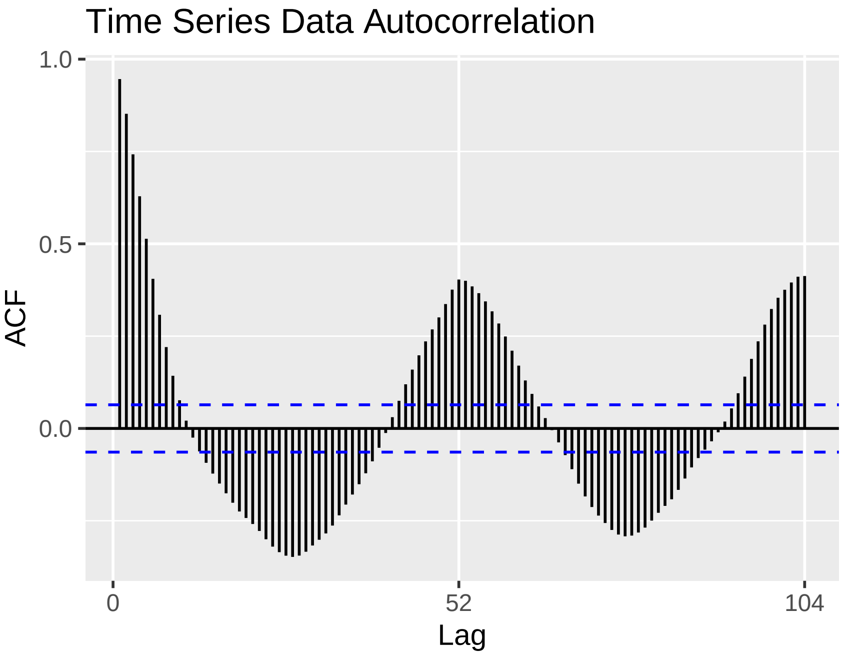
- Trend: represents the gradual change in the time series data. The trend pattern depicts long-term growth or decline.
- Seasonality: represents the short-term patterns that occur within a single unit of time and repeats indefinitely.
- Noise (Random Behavior): represents irregular variations and is purely random. These fluctuations are unforeseen, unpredictable, and cannot be explained by the model.

The estimated values of these components (using the `decmpose` function in `R`) are shown in the following plot:



We can see that the trend component shows an irregular peak around 2009 with is when the swine flu pandemic occurred.

Using the `acf` function in `R` (with `lag`=104, i.e. two seasons), we can also see how the seasonality is reflected. The correlation decreases as the weeks go by and almost disappears as we reach the first quarter of the year, then increases again till it reaches a peak in the middle of the year. The pattern is repeated in the second half of the year.



3.2 Forecasting Models

3.2.1 Baseline Predictor

For each test observation, the baseline predictor is the average of the same week over the previous years. For example, when predicting ILIp for week 34, 2012, the prediction will be the average of the ILIp values in weeks 34 of all previous years.

3.2.2 Holt-Winters Exponential Smoothing

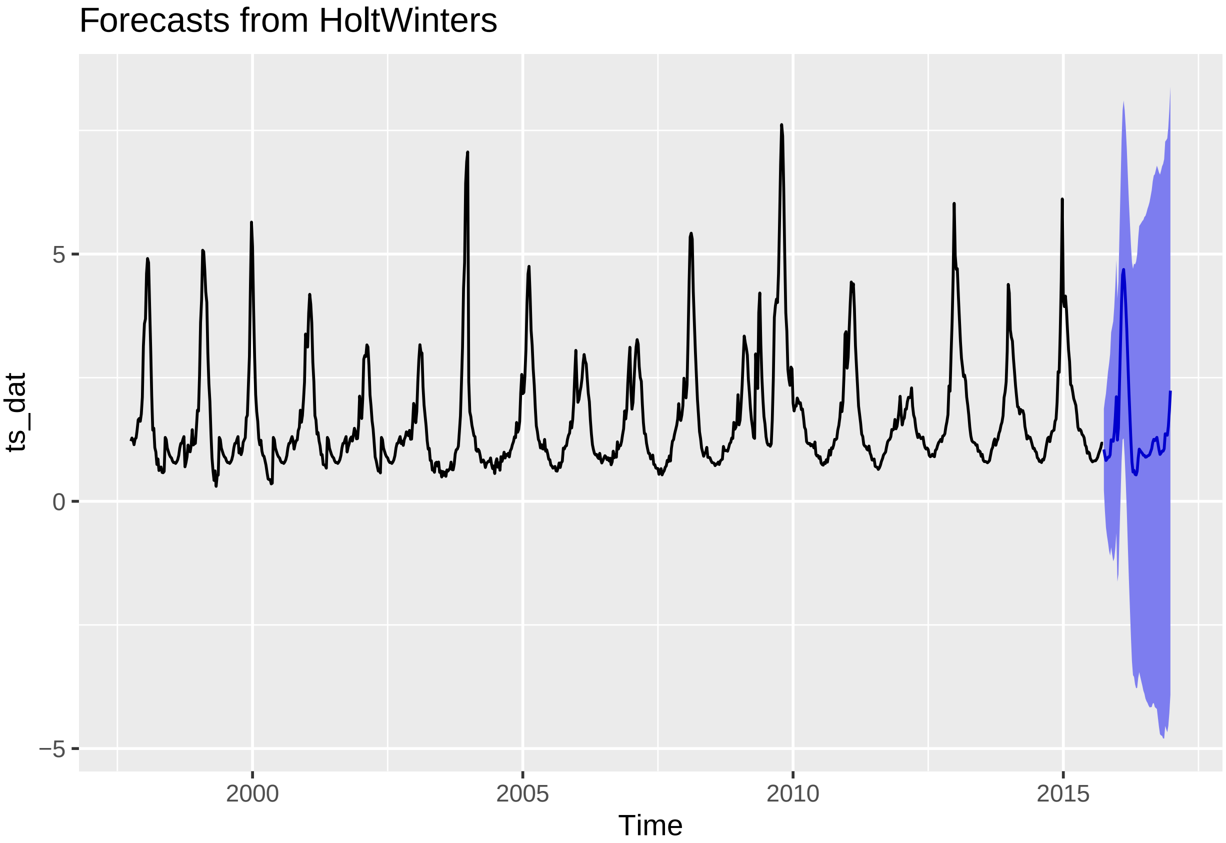
Holt-Winters exponential smoothing estimates the level, slope and seasonal component at the current time point.

Smoothing is controlled by three parameters:

- **alpha** - estimates the level.
- **beta** - estimates the slope of the trend component.
- **gamma** - estimates the seasonal component.

All the parameters have values between 0 and 1. If the values that are close to 0, it mean that relatively little weight is placed on the most recent observations when making forecasts of future values.

The estimated values of the parameters are $\alpha = 0.924, \beta = 0, \gamma = 1$. As the time series is stationary, we see that the trend parameter β is equal to 0.



3.2.3 Autoregressive Integrated Moving Average (ARIMA)

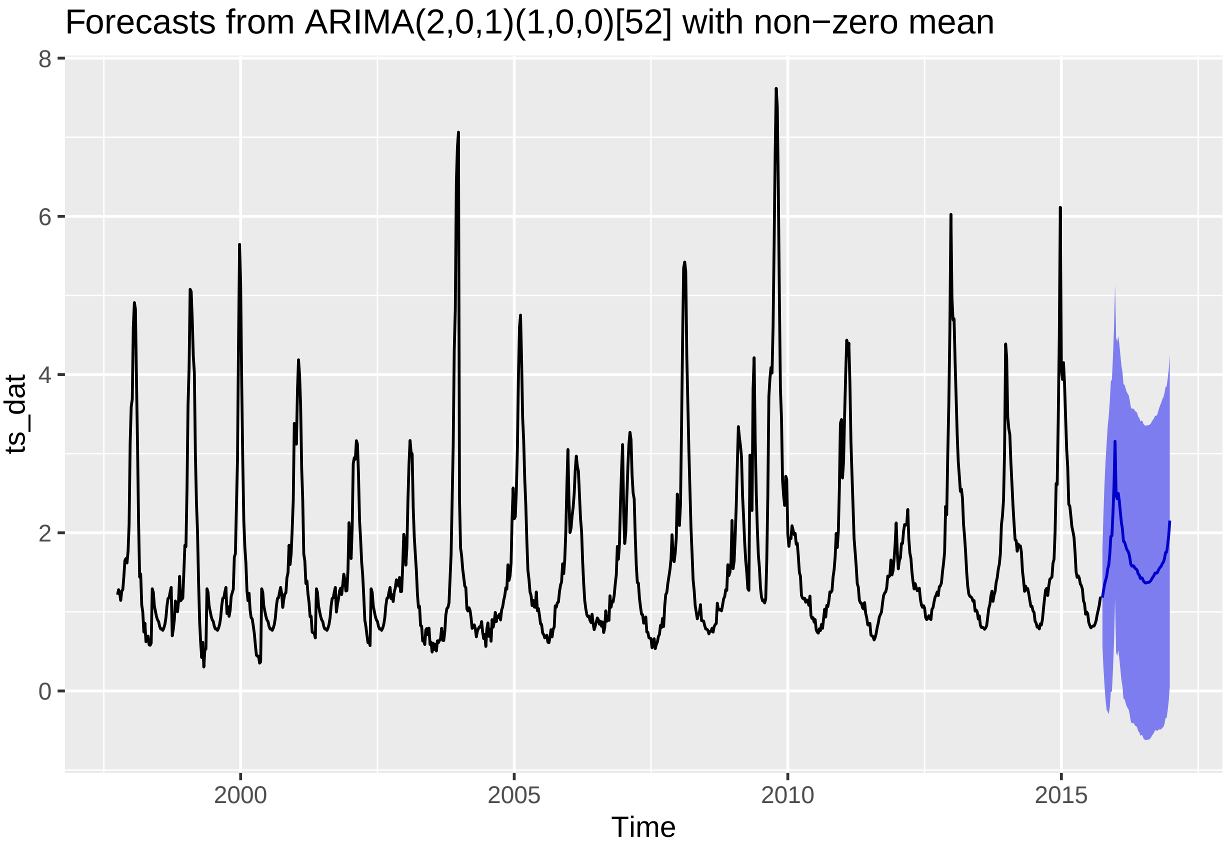
ARIMA models are classified by three factors:

- **p** - Number of autoregressive terms (AR).
- **d** - How many non-seasonal differences are needed to achieve stationarity (I).
- **q** - Number of lagged forecast errors in the prediction equation (MA).

I used the function `auto.arima` in `R` that uses a variation of the Hyndman-Khandakar algorithm, which combines unit root tests, minimization of the AICc, and MLE to obtain an ARIMA model.

1. The number of differences $0 \leq d \leq 2$ is determined using repeated KPSS tests.
2. The values of p and q are then chosen by minimizing the AICc after differencing the data d times. Rather than considering every possible combination of p and q, the algorithm uses a stepwise search to traverse the model space.

The selected parameters are $p = 2, d = 0, q = 1$. Again, as the time series is stationary, we see that there was no need to difference.



3.3 Comparison and Evaluation of the Models

Let y stand for the observed ILIp and $\hat{y}$ stand for the corresponding predicted values.

To compare the different models, the following accuracy measures will be provided:

1. **Pearson Correlation** - a measure of the linear dependence between two variables, defined as:

$$r = \frac{\sum_{i=1}^{n}(y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2(\hat{y}_i - \bar{\hat{y}})^2}}$$

2. **Root Mean Squared Error (RMSE)** - a measure of the difference between predicted and true values, defined as:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

3. **Root Mean Squared Percent Error (RMSPE)** - a measure of the percent difference between predicted and true values, defined as:

$$\text{RMSPE} = \sqrt{\frac{1}{n}\sum_{i-1}^{n}\left(\frac{y_i - \hat{y}_i}{y_i}\right)^2} \cdot 100$$

4. **Maximum Absolute Percent Error (MAPE)** - a measure of the magnitude of the maximum percent difference between predicted and true values, defined as:

$$\text{MAPE} = \left(\max_{i=1,\ldots,n} \frac{|y_i - \hat{y}_i|}{y_i}\right) \cdot 100$$

5. **Hit Rate (HR)** - a measure of how well the algorithm predicts the direction of change in the signal (independently of the magnitude of the change), defined as:

$$\text{HR} = \frac{\sum_{i=2}^{n}\big(\text{sign}(y_i - y_{i-1}) == \text{sign}(\hat{y}_i - \hat{y}_{i-1})\big)}{n-1} \cdot 100$$

where the symbol $(a == b)$ denotes an if statement that returns the value 1, if a (here the sign of the observed changes) and b (here the sign of the predicted changes) are the same, and 0 otherwise.

The following table shows the measures for each model.

Table 1: Accuracy Measures of the Models - Season Prediction

	r	RMSE	RMSPE	MAPE	HR
Baseline Predictor	0.7974	0.475	0.2079	0.1518	1
Holt-Winters ES	0.7351	0.7343	0.3532	0.2945	1
ARIMA	0.6385	0.5636	0.301	0.2369	1

It seems like the ARIMA model had the overall best results.

I perform the same procedure to forecast 4 weeks ahead. The measures for each model are shown in the following table.

Table 2: Accuracy Measures of the Models - 4 Weeks Prediction

	r	RMSE	RMSPE	MAPE	HR
Baseline Predictor	0.9236	0.112	0.08299	0.07949	1
Holt-Winters ES	-0.895	0.3231	0.229	0.1998	1
ARIMA	0.9049	0.1256	0.09454	0.09079	1

Again, it seems like ARIMA model had the overall best results.