Published in Towards Data Science

Denyse Follow

Jul 28, 2021 · 6 min read · Listen

Time Series Clustering — Deriving Trends and Archetypes from Sequential Data

Using Machine Learning to automate time series clustering process

Source: Unsplash

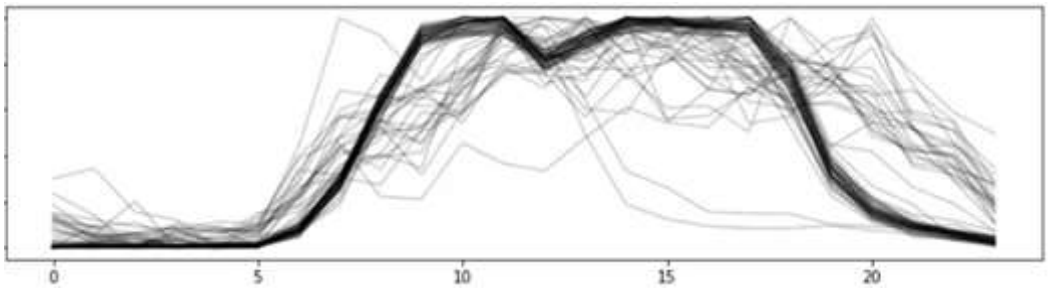
Background

As technology evolved over time, th collected in the world had increased exponentially too. Big da g information models, parking

transactions, and public transport transactions are rich and large datasets. However, existing applications of it tend to be very focused on **specific use cases.**

As part of my internship project with Urban Redevelopment Authority of Singapore (URA) Design and Planning Lab, I was tasked to derive trends from big data.

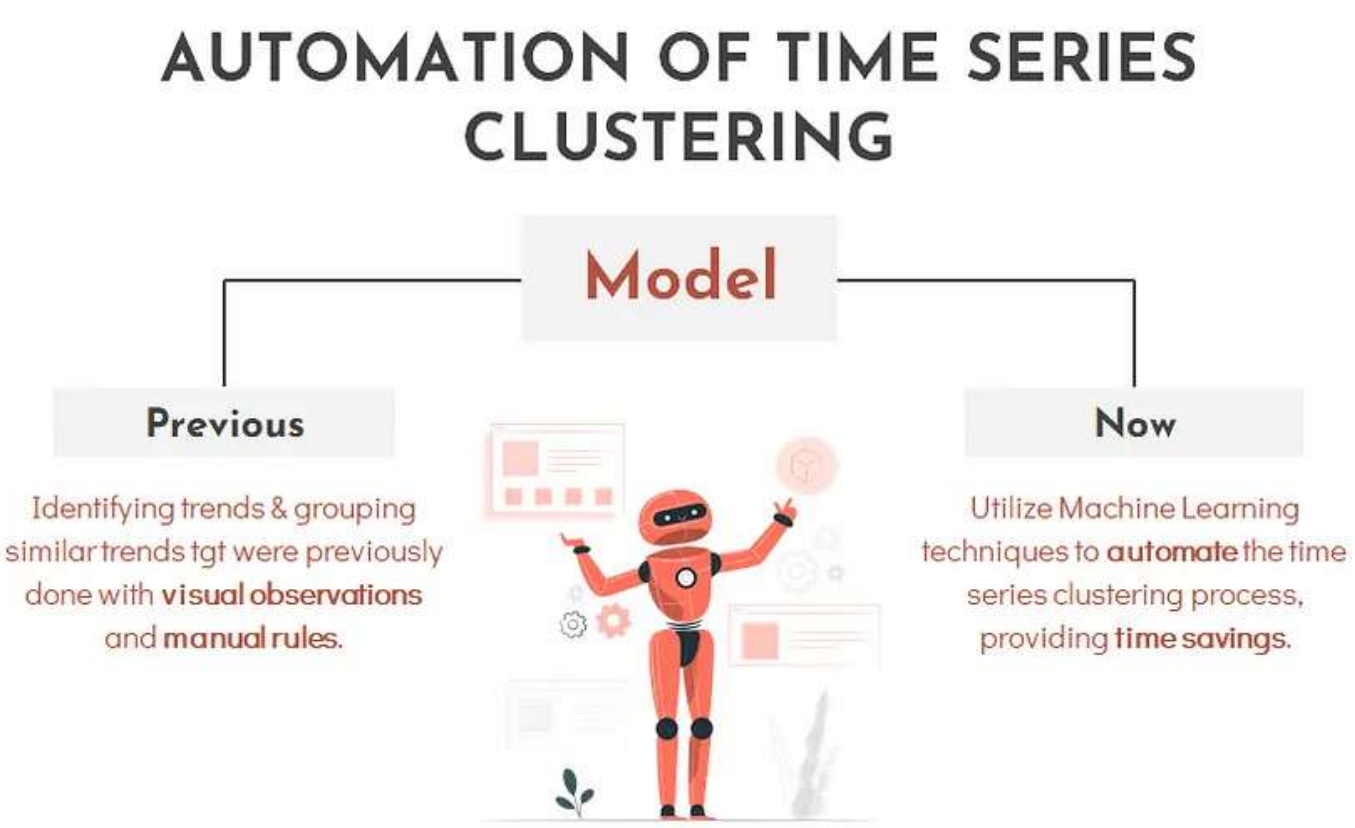
Motivation of Project



Challenges of analysing the data visually | Source: author

At present, it is **challenging** to analyse sequential data visually when plotted on the graph. It is **difficult** to identify and understand trends in data with millions of rows.

For example, the above chart shows the daily pattern over time for a few months — it is quite clear that there is at least one main trend with some outliers. However, it is not easy to know when (**temporal**) and where (**spatially**) these trends occur from a big dataset.



Automation of time series clustering | Source: author

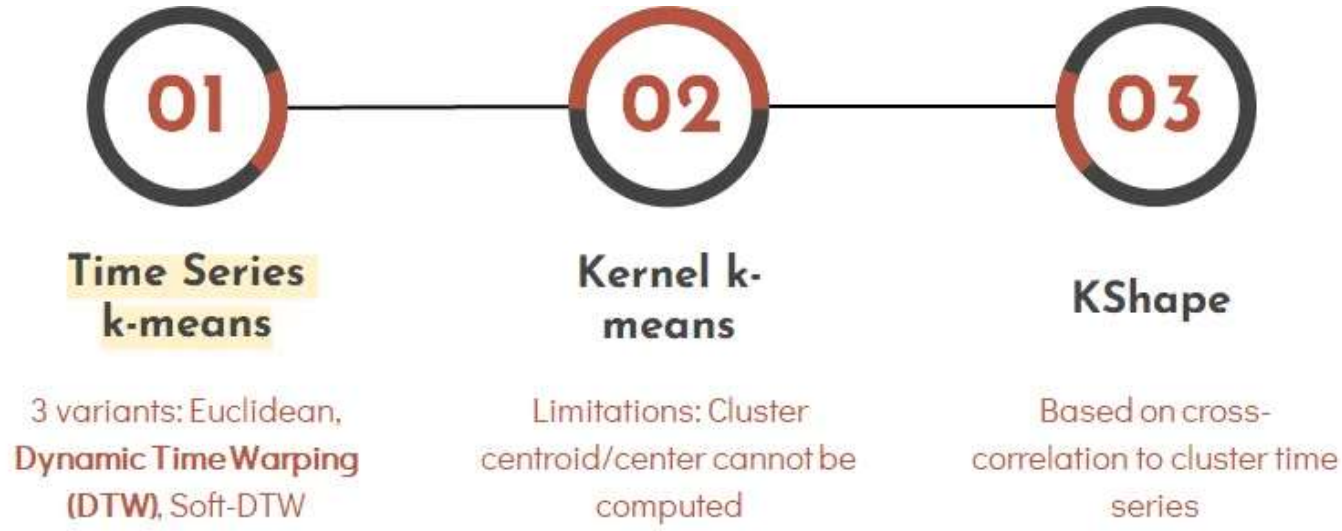
The project thus aims to utilise **Machine Learning clustering techniques** to automatically extract insights from big data and save time from manually analysing the trends.

Time Series Clustering

> ***Time Series Clustering*** *is an unsupervised data mining technique for organizing data points into groups based on their similarity. The objective is to maximize data similarity within clusters and minimize it across clusters.*

The project has 2 parts — temporal clustering and spatial clustering.

Time Series Clustering Algorithms



Source: author

I tested out many time series clustering algorithms on the sequential dataset. Upon closer analysis, **time series k-means with the dynamic time warping metric** produced the most accurate results. Hence, I used this model for subsequent analysis.

Dynamic Time Warping (DTW) Metric for Time Series Clustering

In time series analysis, dynamic time warping (DTW) is one of the algorithms for measuring **similarity** between **two temporal sequences** that do not align exactly in time, speed, or length.

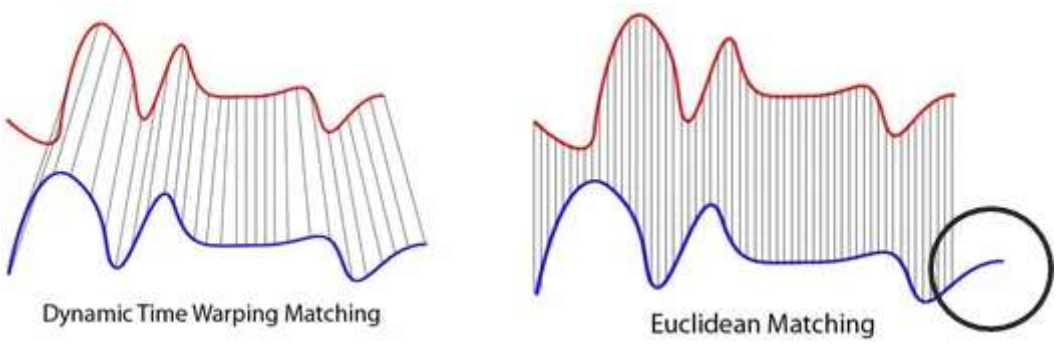
$$DTW(x, y) = \min_{\pi} \sqrt{\sum_{(i,j)\in\pi} d(x_i, y_j)^2}$$

where $\pi = [\pi_0, \ldots, \pi_K]$ is a path that satisfies the following properties:

- it is a list of index pairs $\pi_k = (i_k, j_k)$ with $0 \leq i_k < n$ and $0 \leq j_k < m$
- $\pi_0 = (0, 0)$ and $\pi_K = (n-1, m-1)$
- for all $k > 0$, $\pi_k = (i_k, j_k)$ is related to $\pi_{k-1} = (i_{k-1}, j_{k-1})$ as follows:
  - $i_{k-1} \leq i_k \leq i_{k-1} + 1$
  - $j_{k-1} \leq j_k \leq j_{k-1} + 1$

Source: tslearn documentation

> *To summarize the DTW equation: DTW is calculated as the squared root of the sum of squared distances between each element in X and* its nearest point in Y.



Source: Wikimedia Commons

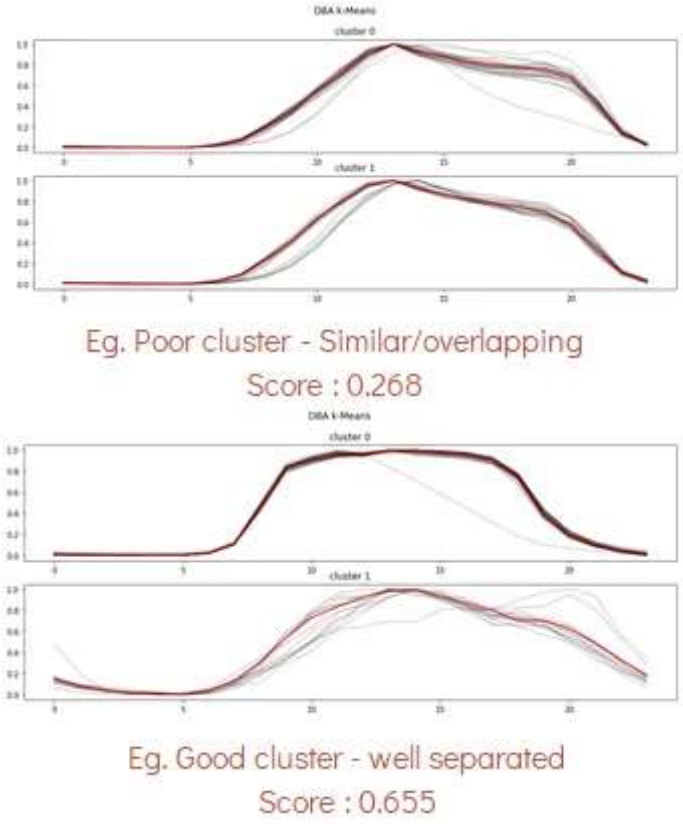
For instance, we have two different curves — red and blue with **different lengths**. The two curves follow the same pattern, however, the blue curve is longer than the red. If we apply the **one-to-one euclidean match** (shown on the right), the mapping is not perfectly synced up and the tail of the blue curve is being left out. DTW resolves this problem by developing a **one-to-many match** so that the same pattern is perfectly matched, and there is no left out for both curves (shown on the left).

Cluster Evaluation: Silhouette Score

$$s = \frac{b - a}{max(a, b)}$$

a: The mean distance between a sample and all other points in the same class. **b:** The mean distance between a sample and all other points in the next nearest cluster. Source: tslearn

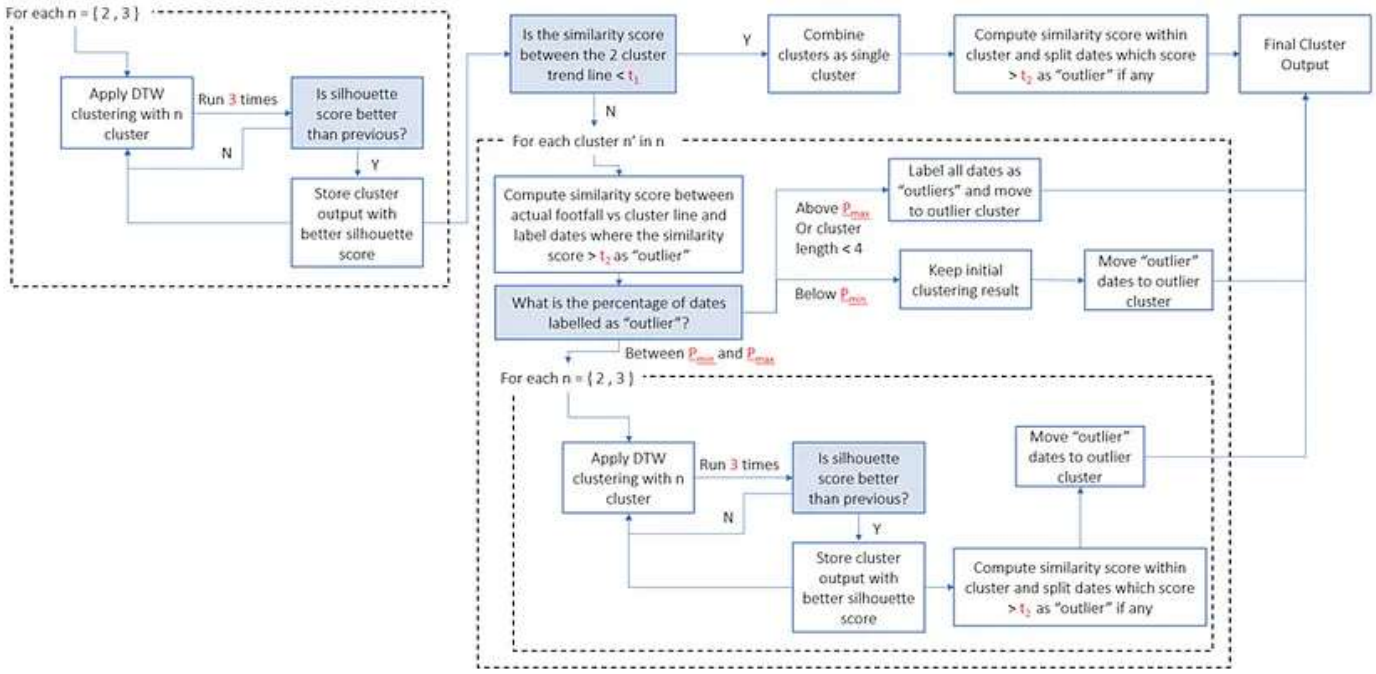
For the evaluation of cluster performance, **silhouette score** was used as the metric. The score is bounded between **-1** for incorrect clustering and **+1** for highly dense clustering. Scores around zero indicate overlapping clusters. The score is **higher** when clusters are **dense** and **well separated**, which relates to a standard concept of a cluster.



Source: author

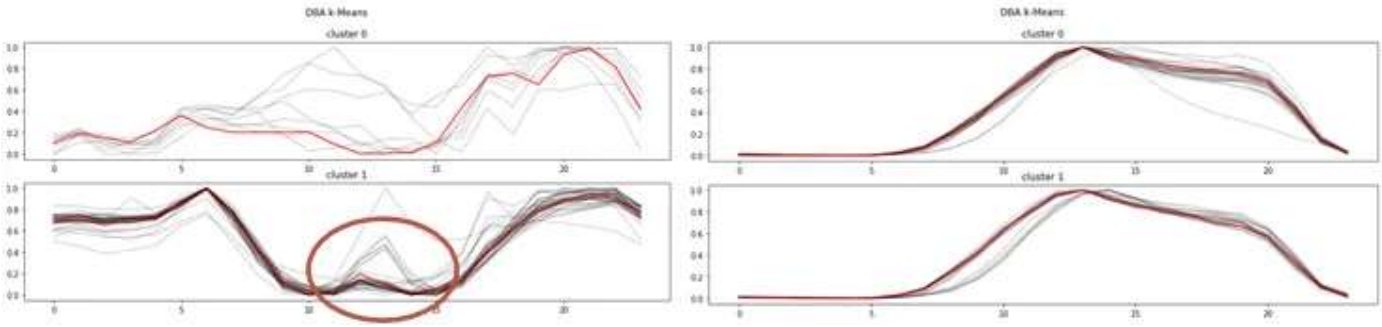
Looking at the top graph, both clusters look similar, resulting in a low score of **0.268.** Whereas for the bottom right graph, the clusters are well-separated and have 2 distinct trends, leading to a high score of **0.655.**

Part 1: Temporal Clustering



Temporal Clustering Framework | Source: author

The goal of temporal clustering is to create a method that can self-cluster identical mobility trends. A **clustering framework** was thus developed to achieve this task.



Results from 1 level of clustering | Source: author

The first level of clustering would usually produce cluster outputs that were not well-separated. As seen from the top-left graph, the model is unable to differentiate the slight peak as a separate cluster. Additionally, the 2 clusters on the right look similar and should be combined.

Therefore, **cluster fine-tuning** was required. The framework consists of multiple functions to enhance the results. Some **functions** include:

1. Checking if it's possible to combine 2 clusters into a single cluster based on the similarity score against a threshold (**t1**) between the cluster main trend lines.

2. Detecting any outlier dates from the cluster using similarity score against a threshold (**t2**), which will be extracted and moved into the outlier" cluster.

3. Identifying any cluster with inconsistent trends. The framework will label all dates in the cluster as outliers if the percentage of outlier dates in a cluster is more (**Pmax**).

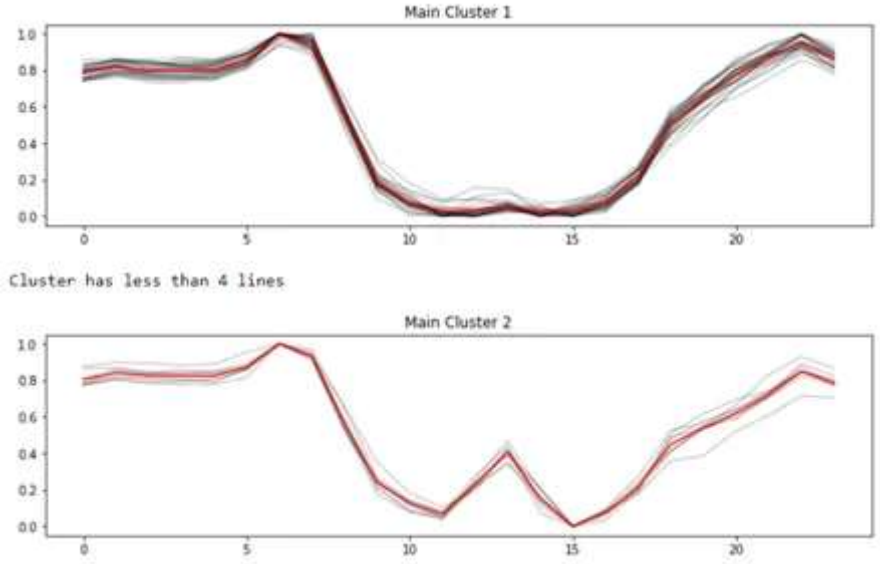
4. Identifying a single date as an outlier. If the percentage of outlier dates in a cluster is below (**Pmin**), the initial clustering results will be kept and we would move any outlier dates detected into the "outlier" cluster.

5. If the percentage of outlier dates is between (**Pmin**) and (**Pmax**), the cluster will then go through another round of clustering.

Note: Due to the complexity of the framework, I will not explain the full methodology in this article.

Example of output

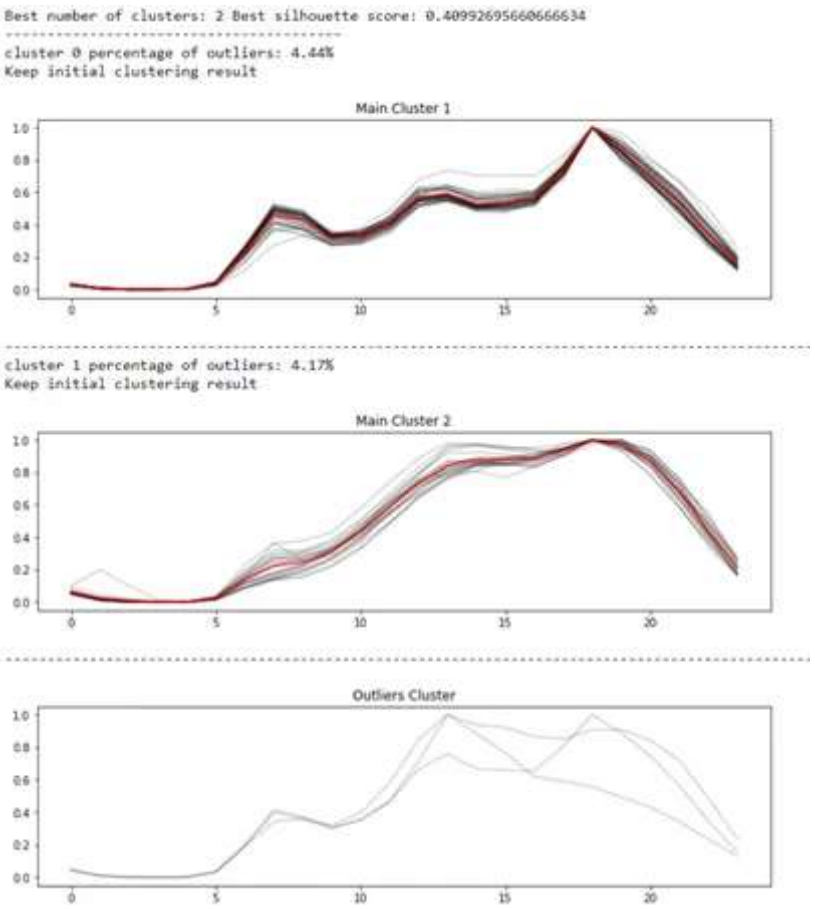
With the framework developed, the results have improved significantly compared to just having 1 level of clustering. We are now able to observe distinct trend lines derived from the time series data.



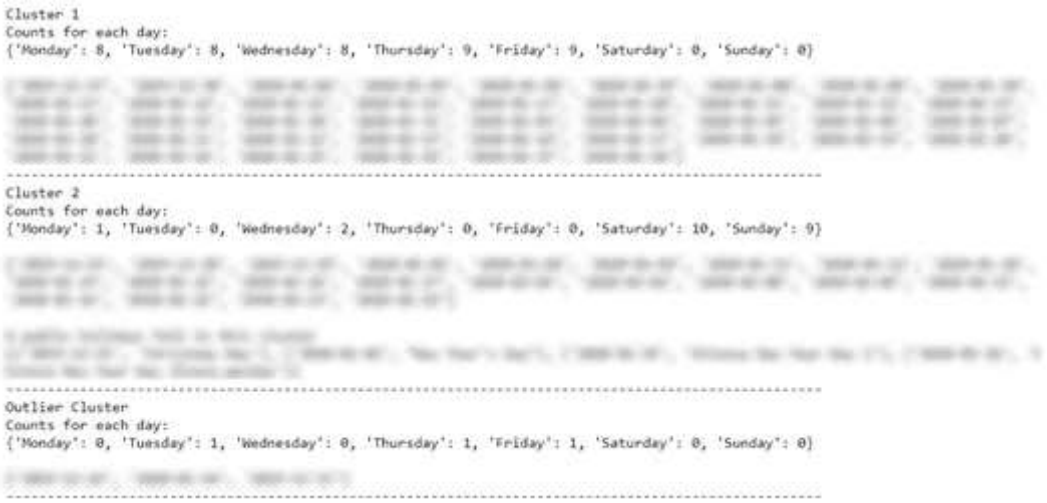
Results improved significantly compared to just having 1 level of clustering | Source: author

The output of the clustering framework will be in the following manner:

1. Graphs will be plotted — main clusters followed by outlier cluster
2. Summary of each cluster will be printed at the bottom — counts for each day and dates falling in each cluster



Graphs will be plotted — main clusters followed by outlier cluster | Source: author

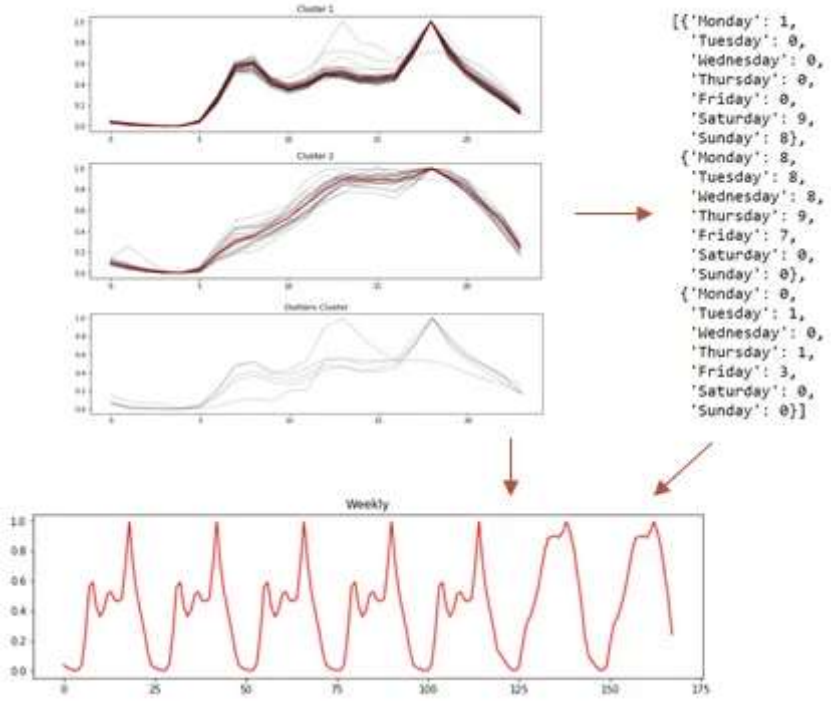


Summary of each cluster will be printed at the bottom — counts for each day and dates falling in each cluster | Source: author

Note: I will not be specifying the trends identified in this article

Part 2: Spatial Clustering

Spatial clustering aims to identify places of the same trend for each time cluster. With the time clusters produced from Part 1, I would obtain the **weekly trend line** for each cell and perform spatial clustering using similar framework for temporal.



How the weekly trend line is derived from each time cluster | Source: author

For this task, I had conducted spatial clustering on a specific area and the cluster results are visualised on ArcGIS Pro.

Note: Similar to part 1, I will not be highlighting the output obtained from the spatial clustering.

Concluding Notes

Deriving Trends and Archetypes from data is just one of the many applications of Time Series Clustering. The framework developed could also be transferrable to other business use cases with sequential data to harness valuable insights.

Internship Reflections

URA Internship experience | Source: author

My experience with URA for the past 5 months is summarised with the above 4 points. I am immensely grateful for the opportunity to apply my analytical skills and contribute to AI/ML adoption in this field. Moreover, I am also thankful to my supervisors — Songyu, Blossom, and Zhongwen, for entrusting me with the project and enabling me to present it to the upper management and external professor. I really appreciate all the guidance and feedback given.

Do feel free to reach out to me on LinkedIn if you would like to find out more about the project or my experiences with URA Design and Planning Lab :)

References

Time Series Clustering — tslearn 0.5.1.0 documentation

Clustering is the task of grouping together similar objects. This task hence heavily relies on the notion of similarity...

tslearn.readthedocs.io

How to Apply K-means Clustering to Time Series Data

Theory and code for adapting the algorithm to time series

towardsdatascience.com

Dynamic Time Warping

Explanation and Code Implementation

towardsdatascience.com

Time Series Clustering Time Series Analysis Machine Learning Data Science

Clustering