UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

Wind pattern analysis applied to Tokyo 2020 Olympic Games (II)

Supervisor:

Alicia Ageno

Department of

Computer Science

Author:

Wangyang Ye

05/07/2019

A thesis submitted for the degree of

Master in Innovation and Research in Informatics

Specialty: Data Science

Facultat d'Informàtica de Barcelona (FIB)

Universitat Politècnica de Catalunya (UPC) – BarcelonaTech

Acknowledgements

I would first like to thank my thesis supervisor, Professor Alicia Ageno, for the willingness and constant support, the advice and guidance towards the right direction. The door to Prof. Ageno's office was always open whenever I got stuck on some problems or had a question about my research or writing.

I would also like to thank the experts who were involved in this research project: Elena Cristofori, Alessandro Demarchi, and Federico Saglia, the meteorologists from TriM company. Without their passionate participation and input, the validation and analysis of results could not have been successfully conducted.

Finally, last but not least, I must express my very profound gratitude to my family and my friends for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them.

Abstract

Meteorologists refer to repeating weather conditions as a weather pattern. A weather pattern is characterised by repeating values of weather variables, such as atmospheric pressure, cloud coverage, air temperature, wind speed and direction, etc. The ability to identify weather patterns and the evolution of a certain weather variable depending on the variation of other variables, is particularly important when weather conditions affect decision making.

Within this thesis, we refer to weather patterns helping decisions in Olympic sailing. The two most essential variables in sailing are wind speed and wind direction. The ability to predict the most likely evolution of these two variables based on previous experience and therefore, on a significant amount of collected weather variables for a specific location, is crucial.

Traditionally, wind patterns are identified based on meteorological experience and a manual methodology, with which the human interpretation is still the most important factor. However, the increasing amount of data makes manual inspection becomes harder and harder. For the magnitude of data that we have by now, it's even a mission impossible.

In this thesis, we have proposed a flexible and scalable framework that is able to perform clustering analysis of wind data and thus, to recognise details of significant wind patterns focused on specific area which consist of characteristic features of the wind speed and direction related with the other meteorological parameters and with the geographical position of the particular area.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This project is a continuation of a previous master thesis developed in the second term of the academic year 2017-18, in the framework of the Tokyo2020 Olympic Games Weather Project, led by TriM [1] and funded by the Austrian Sailing Federation [2] and by Croatia and Cyprus Laser Olympic classes.

Sailing strategy and performance are strongly related to environmental parameters such as weather, oceanic current and geographical data. A thorough prediction of the conditions expected during a sailing race is valuable information for a sailor, as it completely conditions his/her tactics during the race. With the aim of developing a decision support system valid for the Olympic Classes Sailing Venues, a big amount of data will be collected on the sea through several ribs that have been equipped with a weather system able to measure weather variables during trainings and racing in Enoshima Bay, the sailing venue of the next Olympic Games Tokyo 2020, and will be stored into a cloud database. Besides that, the following components will be developed and integrated together into one single web-based platform: 1. Wind component 2. Waves component 3. Oceanic current component 4. Boat Performance component. The present project is related to the wind component, in particular, it focus on applying wind pattern analysis to the Olympic sailing venues.

1.1 Motivation

Meteorologists refer to repeating weather conditions as a weather pattern. A weather pattern is characterised by repeating values of weather variables, such as atmospheric pressure, cloud coverage, air temperature, wind speed and direction, etc.

While it is pretty easy to identify patterns when we refer at a global scale or to long time periods: seasonal weather conditions, climatic weather conditions; it is much harder to identify patterns when we refer to local scale (in our case, Enoshima Bay where Olympic sailing takes place) and to short time periods, such as minutes or hours. The ability to identify weather patterns and the evolution of a certain weather variable depending on the variation of other variables, is particularly important when weather conditions affect decision making. Imagine how useful would it be, being able to identify a potential extreme rainfall based on the evolution of atmospheric pressure and cloud coverage observed previously during several flooding events.

Within this thesis, we refer to weather patterns helping decisions in Olympic sailing. The two most essential variables in sailing are wind speed and wind direction. The ability to predict the most likely evolution of these two variables based on previous experience and therefore, on a significant amount of collected weather variables for a specific location, is crucial.

Traditionally, wind patterns are identified based on meteorological experience and a manual methodology, which involves mainly:

1. The run of numerical prediction models.
2. The collection of weather parameters on the sea.
3. The analysis of predicted and measured values (semi-statistical).
4. A human interpretation of simulated and collected data.
5. A human identification of weather conditions similar to something observed in the past.

Step 5 is mainly carried out through the following methodology:

1. Splitting the wind directions into several sectors.

2. Identification of characteristic behaviour of the wind speed in relationship with the time evolution.

3. Identification of a characteristic behaviour of the wind in relationship with additional weather variables.

The human interpretation (points 4 and 5) is still the most important factor, due to the fact that:

- Weather patterns and wind evolution has to be used for very specific locations and for a very short time range.

- Often there is not the possibility of collecting a sufficient number of data to be used for automatic classification.

- There is no literature reporting the application of automatic identification of weather patterns for sailing.

However, the increasing amount of data makes manual inspection becomes harder and harder. For the magnitude of data that we have by now, it's even a mission impossible.

Therefore, it would be very useful to have approaches based on data mining techniques (especially clustering) that could automatically induce wind patterns based on collected data, as well as the characteristic features of these patterns and their evolution through the day. Clustering analysis might be a significant added value because machines can analyse big amount of data in a very short time and thus, it could help meteorologist and sailors in decision making within Olympic sailing. Moreover, because data measuring is being performed in different locations of the race areas, these clustering methods could also find different behaviours depending on the area for the same wind pattern. All these would allow:

- A detailed analysis to determine the representativeness of the wind fields encountered in the measuring period, their frequency of occurrence, timing, rate of evolution, and transition probabilities.

- Consequently, a more thorough prediction of the conditions expected before a sailing race, which is as mentioned a piece of highly valuable information for the sailor.

1.2 Goals

The main goal is to develop a methodology/procedure of clustering analysis to analyse the 'recorded wind dataset' and to recognize significant wind patterns, in other words, characteristic features of the wind speed and direction related with the other weather parameters of the day (e.g. air pressure, air and water temperature, etc.) and with the geographical position of the specific racing area. A similar analysis should be performed on the 'weather prediction model dataset'. And for instance, analysis of the wind parameters of the model is required to find correlations between predicted and measured values. This step is fundamental for the validation of the weather model that will be used daily during the Olympic Games.

The main opportunities derived by the fact of using combined collected and simulated data are:

1. A larger amount of data to be analysed and classified.

2. The possibility of comparing observed and predicted values and identifying differences in predicted or measured weather patterns.

The second point is particularly important, because, if the results related with predicted patterns are similar to the ones obtained using measured values, it would be possible to generalize the same methodology to areas where no measurements are collected but only predicted weather parameters are available.

Besides the development of the methodology, in this thesis we will validate it with a (still) limited set of data (as to size and as to quality), with the idea that, if a successful methodology is defined, more data can be aggregated to the existing database of a certain location contributing to an improvement of identified patterns.

During the previous master project, the clustering module (written in Python)

has been completely developed, using data coming from two different weather prediction models. The current project aims at:

- Correcting errors in the previous software.
- Applying the clustering module also to data collected on the sea (a previous data filtering and cleaning will be required to take out noise or errors from this dataset)
- Enriching the methodology by providing additional useful information about the clusters.
- Comparing/combining the results obtained with both data sets
- Extend the developed framework so that it can handle wind patterns of different granularities. For instance, to support hourly wind fields and daily sequential wind data.

1.3 Next chapters

The rest of the thesis is organized as follows: In Chapter 2, we present the background of this work, including the introduction of wind components and fundamental algorithms used in the related works. In Chapter 3, we describe the different datasets that will be used in this project. Then, in Chapter 4, the architecture and implementation of the developed framework will be explained. Analysis of obtained results will be discussed in Chapter 5. Finally, in Chapter 6, we offer conclusions and directions for future work.

Chapter 2

Background

In this chapter, essential concepts regarding wind components will be introduced. Afterwards, important previous researches on the wind pattern analysis in which our project mainly based on will be presented. Furthermore, the workflow and fundamental algorithms used in the previous project will be introduced.

2.1 Wind components

Wind (air flow) in the atmosphere has both a speed and direction. This is represented mathematically by a vector. In figure 2.1, the wind vector v_H is represented by the bold black line.

The wind vector can be expressed in two ways [3], either in terms of orthogonal velocity components, where u is the zonal velocity, that is, the component of the horizontal wind towards east. And v is the meridional velocity, i.e. the component of the horizontal wind towards north.

Or as true wind speed (TWS), $|v_H|$, and true wind direction (TWD), which is not simply the angle ϕ_{POLAR} between the wind vector and x-axis. Two commonly used representations of wind direction exist:

- ϕ_{VECT} is the wind vector azimuth representing the direction **towards** which the wind is blowing.
- ϕ_{MET} is the meteorological wind direction, i.e. the direction **from** which the

wind is blowing.

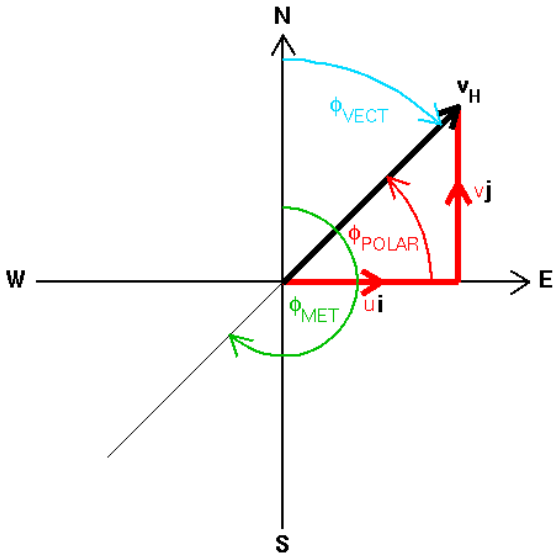

Figure 2.1: Example of wind components

By convention [4], winds are referred to the direction that they are coming from and thus, ϕ_{MET} is usually used. If one says "a north wind", then the wind is coming from the north. In such a situation, the wind barb would be pointed north, and the wind arrow would be pointed south, as we can see in the figure below. Recall that wind barbs point in the direction the wind is coming from, and wind arrows (vectors) point in the direction the wind is blowing towards.

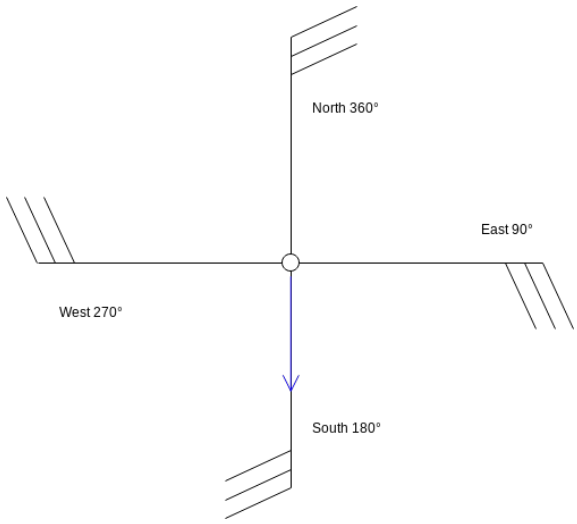

Figure 2.2: Meteorological direction and wind barbs

The direction is typically expressed in units of degrees, ranging from 0° to 360°. A north/south wind corresponds to be 360°/180°, respectively. And an east/west wind is 90°/270°, as indicated in figure 2.2.

Velocity components can be converted to TWS & TWD and vice versa. However, we have to bear in mind that when dealing with the conversion, the mathematical convention for the direction is used. To convert from meteorological direction to math direction, the following formula is applied:

$$\phi_{MATH} = 270 - \phi_{MET} \tag{2.1}$$

If the value is negative, then simply add 360 to ϕ_{MATH}. When using math wind direction, a due west wind will have a positive vector pointing along the x-axis and thus a direction of zero degrees. And a due south wind will have positive vector pointing along the y-axis and a then direction of 90 degrees.

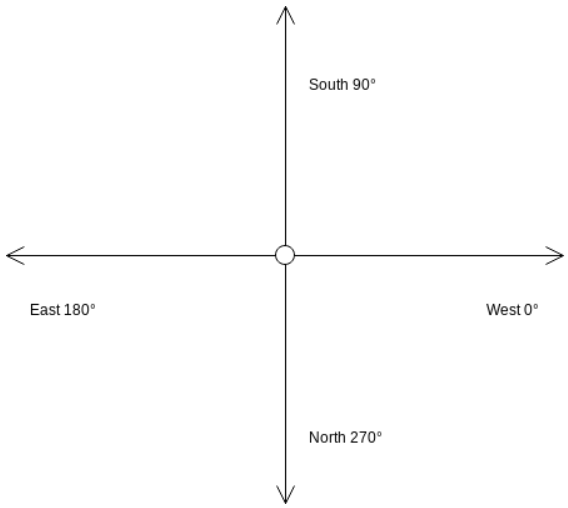

Figure 2.3: Math wind direction

On the one hand, in order to convert TWS&TWD to u, v components, the formulas below are applied:

$$\begin{aligned} u &= TWS \cdot cos(\phi_{MATH}) \\ v &= TWS \cdot sin(\phi_{MATH}) \end{aligned} \tag{2.2}$$

Note that for some programming languages and libraries, trigonometric functions

require that the angles are expressed in units of radians, rather than degrees. The conversion is through the below relationship:

$$\phi(rad) = \frac{\pi}{180} \cdot \phi(degree) \tag{2.3}$$

On the other hand, given velocity components u and v in scalars, the wind speed (or magnitude) is stated as the square root of the sum of the squares:

$$TWS = \sqrt{u^2 + v^2} \tag{2.4}$$

For the computation of wind direction, it requires the usage of the inverse trigonometric function of tan (i.e. arctan).

$$TWD = \begin{cases} arctan(\frac{v}{u}) & \text{if } u > 0 \\ arctan(\frac{v}{u}) + \pi & \text{if } v \geq 0, u < 0 \\ arctan(\frac{v}{u}) - \pi & \text{if } v < 0, u < 0 \\ +\frac{\pi}{2} & \text{if } v > 0, u = 0 \\ +\frac{\pi}{2} & \text{if } v < 0, u = 0 \\ undefined & \text{if } v = 0, u = 0 \end{cases} \tag{2.5}$$

In many programming languages, standard math libraries have the $atan2$ function which will do the above computation.

As in the current project, we keep using Python as the programming language, fortunately, MetPy [5], a powerful Python library for reading, visualising, and performing calculations with weather data, allowed us to handle all the mentioned conversions easily. Furthermore, it supports computation with the meteorological direction which is more intuitive for the analysis.

In the rest of the report, wind direction will always be meteorological direction ϕ_{MET}.

2.2 Related Work

As mentioned in the previous chapter, this project takes as basis the previous master thesis developed in the second term of the academic year 2017-18. Thus, it is necessary to present the previous work and the researches that the work is based on.

The substantial referenced work is carried out by Kaufmann and Whiteman in [6]. In this research, a two-stage classification scheme (proposed by Kaufmann and Weber in [7]) using the combination of Hierarchical clustering and K-means is applied to analyse wind patterns in the Gran Canyon, where the data source is the hourly wind data taken from 15 meteorological stations. This study matches very well with the scope of our current project since they proposed a suitable and robust solution for wind pattern analysis. The proposed approach is not location/time dependent, thus, it can be applied to anywhere of the world. In spite of that, it is important to emphasise that in our project, the area where wind patterns will be studied is much more specific, since the sailing competitions take place in a limited sea area.

The previous project [8] carried out by the former student, is mainly based on the work of Kaufmann and Whiteman, where wind patterns around the bay of Barcelona are studied. The data are collected through predictions on meteorological parameters produced by a numerical weather prediction model called WRF (see section 3.2). 16 points of the model have been chosen as virtual weather stations in order to be coherent with [7]. The locations can be seen in the figure below.

In the next sections, the workflow and essential algorithms used in the previous works will be introduced.

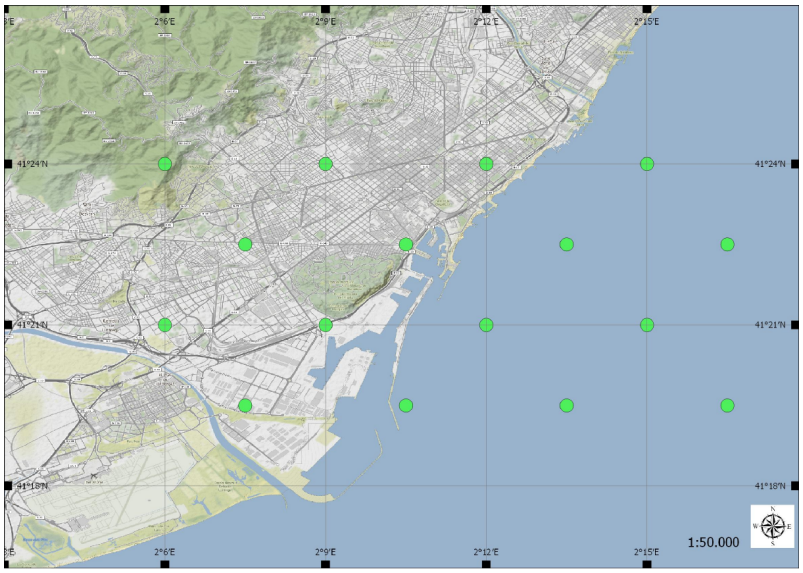

Figure 2.4: The 16 points around bay of Barcelona

2.3 Workflow

In the previous thesis, the workflow is as shown in the figure below. There are two possible ways when performing clustering. On the one hand, we have manual HC combined with K-means, which is identical to the two-stage classification that Kaufmann and Whiteman have used. With this combination, we have to choose the value of k for K-means clustering manually. On the other hand, an automatic clustering method is used, with which the number of clusters is decided automatically.

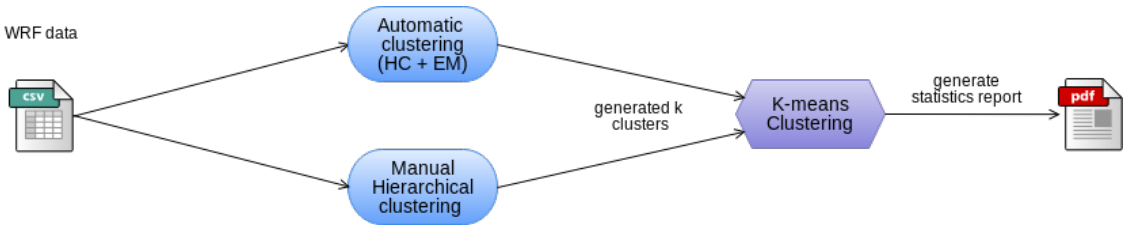

Figure 2.5: Main workflow of the previous project

2.4 Clustering

By **clustering**, we refer to a category of *unsupervised learning* which consists of grouping a set of objects into various clusters in the way that the elements within

the same group are homogeneous and are distinct among different groups.

In the scope of our project, **a cluster is identified by values of wind velocity components u and v**, so that similar winds, in terms of speed and direction, will be situated in the same cluster. Together with the statistics report of the corresponding meteorological parameters, wind pattern can be easily detected by the meteorologists.

There are many kinds of clustering which include:

- Density-based clustering: the most common algorithm is Density-based spatial clustering of applications with noise (DBSCAN). Given a set of points in some space, it groups together points that locate together closely, marking as outliers points that lie alone in low-density regions (whose nearest neighbours are too far away).

- Distribution-based clustering groups objects based on the underlying distribution models of the data belong to. For example, expectation–maximization (EM) algorithm iteratively find maximum posteriori estimates of parameters in statistical models, then the clustering is based on the values of these parameters.

- Hierarchical clustering: it seeks to build a hierarchy of clusters.

- Partitioning-based clustering: clusters are represented by a centroid, which may not necessarily be a member of the data set. A typical example is the K-means clustering.

Since the clustering algorithms used in the previous works are Hierarchical clustering and K-means, we are going to describe them in a more detailed way.

2.4.1 Hierarchical clustering

Hierarchical clustering (HC) is a method of cluster analysis which seeks to build a hierarchy of clusters. Strategies for Hierarchical clustering generally fall into two groups:

1. Agglomerative: This is a "bottom-up" approach: each observation starts in its own cluster, and pairs of clusters are merged recursively as one moves up the hierarchy. This is the method used in the previous works.

2. Divisive: This is a "top-down" approach: all observations start in a single cluster, and splits are performed recursively as one moves down the hierarchy.

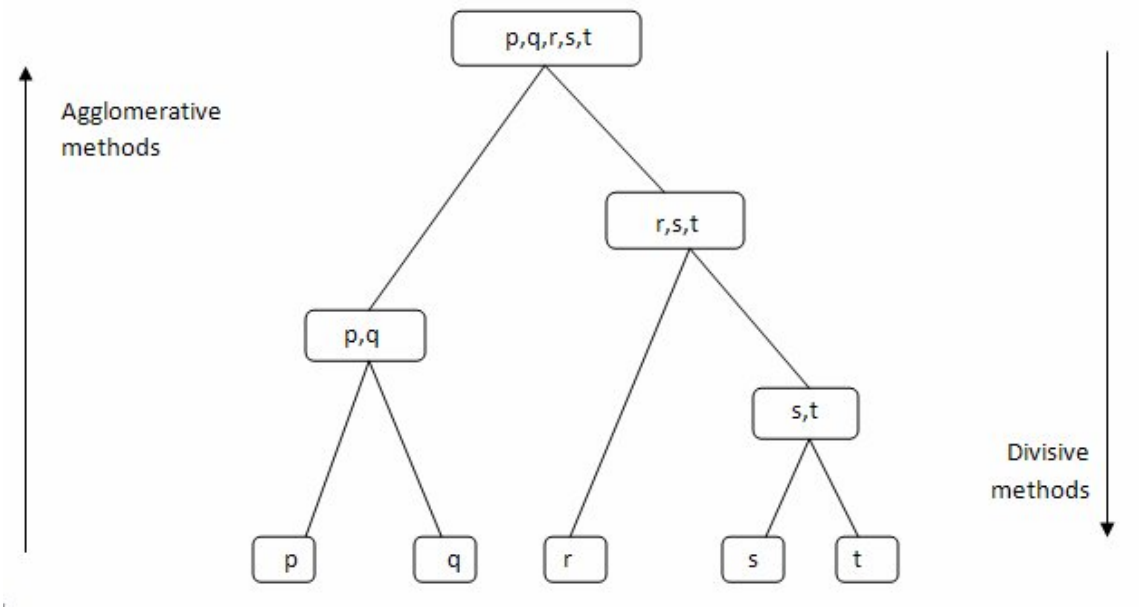

Figure 2.6: Hierarchical clustering: Agglomerative versus Divisive [9]

From figure 2.6, we can clearly see the difference between the mechanism of both approaches.

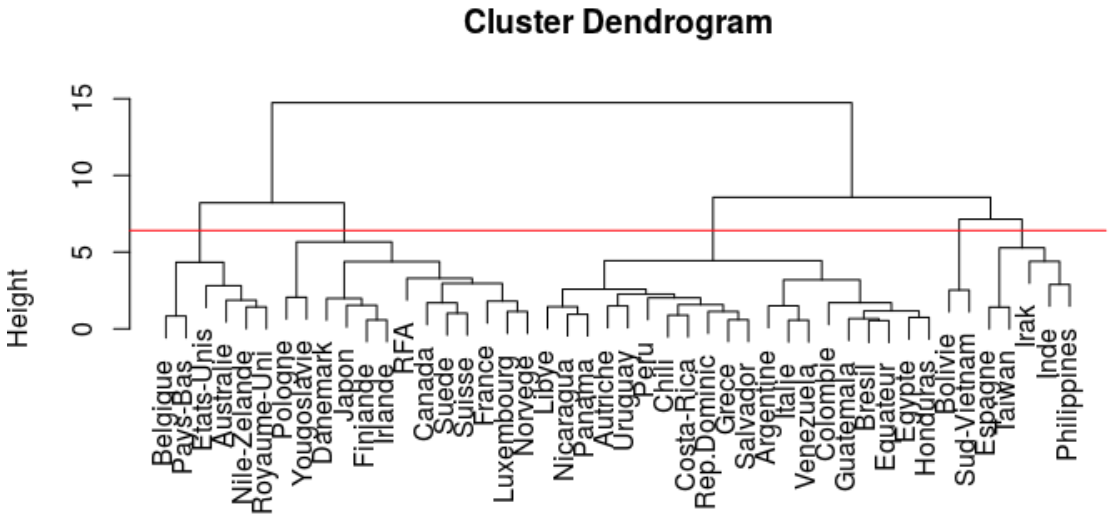

Figure 2.7: Example of dendrogram

Figure 2.7 is a dendrogram taken from an analysis on the Russett data set [10], from which we can clearly see the hierarchies formed when merging the clusters. This dataset has defined three blocks of variables that relate to "Agricultural Inequality", "Industrial Development", "Political Instability", correspondingly, for a total of 47 countries. An additional variable describes the political regime: stable democracy, unstable democracy or dictatorship. Russett collected this data to study relationships between Agricultural Inequality, Industrial Development and Political Instability. Russett's hypothesis can be formulated as follows: It is difficult for a country to escape dictatorship when its agricultural inequality is above-average and its industrial development below-average. And this hypothesis was supported by the results of cluster analysis.

The procedure of agglomerative HC is quite straightforward, as illustrated in the algorithm 2.1. We first initialize each object as a cluster and compute the pairwise distance matrix between them. Then, we recursively merge the two clusters with the minimum pairwise distance and update the distance matrix, until there's only one single cluster.

Algorithm 2.1 Hierarchical clustering

Function *Hierarchical_clustering(D, lf)* **:**
Input: D: dataset, lf: linkage function
Output: dendogram representing the aggregation

```
1  Initialization:
2  E ← set of objects to cluster, every single object represents a cluster
3  D ← the distance matrix for each pair of clusters
4  while cardinality(E) > 1 do
5      Find the closest clusters (a, b) in D
6      Set h = a ∪ b ;                          // merge the closest clusters
7      Update E = E − {a, b} + {h} ; // replace the original clusters with
        the merged one
8      Update the matrix of distances of E in D
9  end
10 return dendogram
end
```

2.4.1.1 Linkage criteria

Another essential distinction between the different Hierarchical clustering algorithms is due to the **linkage criteria**. Roughly saying, it determines how the distance between each cluster is measured when there are more than one element within the cluster. Given a distance function d, some commonly used linkage criteria are:

- Single-linkage: it takes the minimum distance between the clusters. Formally stating:

$$D_{single}(A, B) = min\{d(a, b) : a \in A, b \in B\} \tag{2.6}$$

- Complete-linkage: it takes the maximum distance between the clusters:

$$D_{complete}(A, B) = max\{d(a, b) : a \in A, b \in B\} \tag{2.7}$$

- Average-linkage: it takes the average pairwise distance between the clusters:

$$D_{average}(A, B) = \frac{1}{|A| \cdot |B|} \sum_{a \in A} \sum_{b \in B} d(a, b) \tag{2.8}$$

- Centroid-linkage: the distance is as the distance between the centroids of the clusters:

$$D_{centroid}(A, B) = d(a, b), a, b \in centroids \tag{2.9}$$

- Ward's method [11]: this criterion seeks for the minimization the total within-cluster variance. At each iteration, it finds the pair of clusters that leads to a minimum increase in total within-cluster variance after merging so that logically the elements will be more equally distributed. This mentioned increase is a weighted squared distance between cluster centres. This method requires that the initial distance between individual objects must be proportional to squared Euclidean distance. At the outset, all clusters are singletons (clusters containing a single point). Therefore, the initial cluster distances are defined to be the squared Euclidean distance between points:

$$d_{i,j} = d(X_i, X_j) = ||X_i - X_j||^2 \tag{2.10}$$

For the choice of the number of clusters, Kaufmann and Whiteman used a simple yet effective criterion. Since in HC, two clusters are merged at each step, the dissimilarities of the merged clusters could be a good indicator. With complete-linkage, they correspond to the maximum dissimilarity within the newly formed cluster. If the dissimilarity of a merged cluster is high, it means that two heterogeneous clusters were merged. Thus it is better to stop the merging before such a leap takes place. To facilitate this decision, the dissimilarities are represented in a plot. On the x-axis, the numbers of clusters are presented decreasingly, and on the y-axis, the maximum dissimilarities of each iteration are plotted. The preferred choices are the points before the leaps. The figure below is an example of dissimilarities plot, we would say that for instance, 10 and 6 could be a good choice.

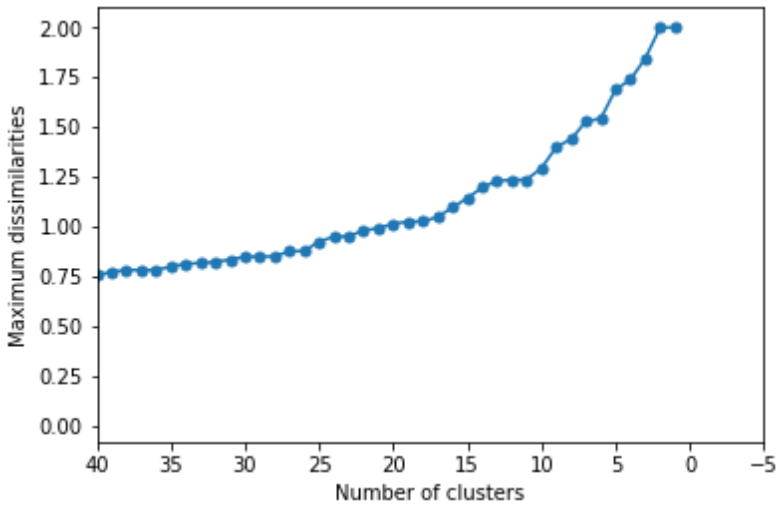

Figure 2.8: Example of dissimilarities plot

An important part of the previous thesis focused on an approach for selecting the number of clusters automatically. However, this approach is not applied in the current project since the result was not promising in the context of the current project.

2.4.2 K-means clustering

K-means clustering is probably the most commonly used partitional-based clustering algorithm. It partitions n observations into k non-overlapping clusters in which each observation belongs to the cluster represented by the closest centroid, which serves as the prototype of a cluster. The algorithm is based on the concept that a good partition should make the within-cluster distances as small as possible. For the measurement of distance, the Euclidean distance is often used. Naturally, the distance between each pair of observations can be considered as a criterion for similarity evaluation, the closer the distance, the higher the similarity is.

The most famous and fundamental description regarding the main process of the algorithm, illustrated by J. B. MacQueen [12], could be found in the following pseudocode:

Algorithm 2.2 K-means clustering

Function *K-means(D, k)* **:**

Input: D: dataset, k: number of clusters

Output: final clusters

1 Randomly choose k observation as initial cluster centroids (seeds)

2 Compute the distance between each observation to each cluster centroid, then assign the observation to the cluster represented by the closest centroid

3 Recompute the centroid for each of the clusters. This is done by averaging the elements within the cluster.

4 Repeat steps 2 and 3 until convergence

end

The problem of this algorithm is the fact that the choice of the initial seeds and the number of clusters could have a huge impact on the result of clustering. In practice, a technique called Consolidation is widely applied, which takes advantages of both Hierarchical clustering (the dendrogram informs about the whole process of aggregation and gives clues concerning the number of distinct groups in the data) and K-means clustering (linear cost, local optimal partition). The process is as follows:

1. Perform Hierarchical clustering.
2. Decide the number of classes present in the data and compute the corresponding centroids.
3. Perform K-means clustering taking as seeds the previously calculated centroids.

2.4.2.1 Wishart's variant

Kaufmann and Whiteman have used a variant of K-means clustering proposed by Wishart [13] [14] [15] that uses additional user-defined parameters ϵ, n_min, and K_max. Like classical K-means, the number of clusters is given by the user, but this variant is able to modify the provided value through decisions within the algorithm in order to archive a more sophisticated clustering result.

On the outset, the procedure starts with an initial partition of the data, for example, by following a random assignment procedure. Then the centroid of each cluster is computed based on the initial composition. The distance from each element to each centroid is computed, and if the smallest distance exceeds the threshold, the element will be moved to the outlier set and the cluster centroids will be updated. Otherwise, the element is assigned to its nearest cluster, that is, the one whose centroid has the least distance towards the element. If at any time, due to the updates of the cluster centroids, an object that has been set aside is now closer to a cluster centroid than the threshold, then the object is assigned to that cluster. After this procedure converges, if the number of elements within any given cluster is less than n_min, all its elements will be removed to the outlier set and the previous steps will be repeated. When both of these steps converge, the most similar clusters are merged until there are at most K_max clusters.

Algorithm 2.3 K-means clustering - Wishart's variant

Function *Wishart(D, k, threshold, n_min, K_max)* **:**

Input: D: dataset, k: initial number of clusters, ϵ: threshold, n_min: minimum number of elements required per cluster, K_max: maximum number of clusters

Output: final clusters

```
1  clusters ← randomly_create_k_clusters(D, k)
2  outliers ← empty set
3  while there are changes in the clusters do
4      centroids ← compute_centroid(clusters)
5      for element e in clusters do
6          min_dist ← min_distance(e, centroids)
7          if min_dist > ε then
8              remove e from the current cluster to outliers
9              update_centroids(clusters)
10             for outlier o in outliers do
11                 if min_distance(o, centroids) ≤ ε then
12                     move o to the closest cluster
13             end
14         else
15             move e to the closest cluster
16         end
17     end
18     for outlier o in outliers do
19         if min_distance(o, centroids) ≤ ε then
20             move o to the closest cluster
21     end
22     for cluster c in clusters do
23         if size of cluster < n_min then
24             move the elements of the cluster c to outliers
25     end
26 end
27 while number of clusters > K_max do
28     merge two clusters that are the most similar
29 end
30 return clusters
end
```

2.4.3 Centroid computation

In [6] [8], the centroid of a cluster is done by averaging the values of velocity components u, v for each of the stations within the cluster. That is, for each station j, we compute the corresponding average of u, v components taking into account all the timestamp of the cluster:

$$u_{centroid,j} = \frac{\sum_{i=1}^{n} u_{i,j}}{n} \qquad v_{centroid,j} = \frac{\sum_{i=1}^{n} v_{i,j}}{n} \tag{2.11}$$

where n is the number of elements (hourly wind fields) of the cluster. Then, the centroid would be an artificial hourly wind field which contains all the averaged velocity components for each of the stations.

2.5 Normalization

Before diving to the clustering procedure, Kaufmann and Whiteman [6] did an additional procedure called normalization, which has been followed in the previous thesis [8] as well. In order to prevent the stations with generally high speeds from getting an overweighting in the distance measure [defined later in Eq. 2.19, the wind components u_{ij} and v_{ij} at each time i and station j were firstly normalized by dividing by the time-average speed s_j at each site to obtain

$$u'_{ij} = \frac{u_{ij}}{s_j}, \qquad v'_{ij} = \frac{v_{ij}}{s_j} \tag{2.12}$$

where the time-average speed at site j is computed as:

$$s_j = \frac{1}{M_j} \sum_{i=1}^{M_j} (u_{ij}^2 + v_{ij}^2)^{1/2} \tag{2.13}$$

and M_j is the total number of hourly winds at the site.

After this normalization of the hourly wind measurements at each of the sites,

the individual wind patterns were normalized. The wind components u'_{ij} and v'_{ij} are divided by the spatial-average speed s'_i at each time i, that is:

$$\tilde{u}_{ij} = \frac{u'_{ij}}{s'_i}, \qquad \tilde{v}_{ij} = \frac{v'_{ij}}{s'_i} \tag{2.14}$$

where the spatial-average speed at each time i is calculated in the following way:

$$s'_i = \frac{1}{N_i} \sum_{j=1}^{N_i} ({u'_{ij}}^2 + {v'_{ij}}^2)^{1/2} \tag{2.15}$$

and N_i is the total number of sites at time i.

2.6 Distance measure

For any clustering scheme to work properly, a measure of dissimilarity or distance between wind patterns is required. There exist many distance measures, some commonly used metrics are [16]:

- Euclidean distance: in mathematics, it is the straight-line distance between two points in Euclidean space. It can be formally stated as:

$$||x - y||_2 = \sqrt{\sum_i (x_i - y_i)^2} \tag{2.16}$$

- Manhattan distance: the distance between two points is the sum of the absolute differences of their Cartesian coordinates. It can be defined mathematically as:

$$||x - y||_1 = \sum_i |x_i - y_i| \tag{2.17}$$

- Mahalanobis distance (MD): it is the distance between two points in multivariate space. The Mahalanobis distance measures distance relative to the centroid, which can be considered as an overall mean for multivariate data. It takes into account the correlations of the data set. The most common use of

MD is outliers detection.

$$\sqrt{(x-y)^T S^{-1}(x-y)}, \text{ where S is the covariance matrix} \tag{2.18}$$

- Dynamic Time Warping (DTW): measures similarity between two time series, which may vary in length.
- Hamming distance: it measures the minimum number of substitutions required to change one string into the other. For example, the distance between '10101' and '11001' is two.
- Levenshtein distance: it is the minimum number of single-character edits (insertions, deletions or substitutions) required to change one word into the other. For instance, the distance between 'cat' and 'fat' is one since we only have to substitute 'c' with 'f'.

An appropriate choice of metric will have an impact on the shape of the clusters, since some elements may be closer to a cluster according to one metric and farther away according to another. For example, in a 2-dimensional space, the distance between the point (1,0) and the origin (0,0) is always 1 according to the usual norms, but the distance between the origin (0,0) and the point (1,1) can be 2 under Manhattan distance, and $\sqrt{2}$ under Euclidean distance.

In the previous sections that describe the clustering algorithms, the distance measure referred is the Euclidean distance. However, in this work, it was not suitable to apply it directly due to the underlying data. The velocity components explained in section 2.1, are the two parameters that determine the similarities between winds. And it is necessary to take into account the number of stations that are available at both timestamps of the two hourly wind data to compare.

2.6.1 Measure of Kaufmann and Whiteman

Kaufmann and Whiteman defined the distance between two wind patterns at arbitrary times a and b as:

$$d_{ab} = \frac{1}{N_{ab}} \sum_{j}^{N_{ab}} [(\tilde{u}_{aj} - \tilde{u}_{bj})^2 + (\tilde{v}_{aj} - \tilde{v}_{bj})^2]^{1/2} \tag{2.19}$$

where N_{ab} is the total number of sites that are available at both times a and b. And $\tilde{u} and \tilde{v}$ are the normalized velocity components. This measure is adopted by the previous student as well.

2.7 Comparison between Clusterings

Since the previous project adopted two clustering procedures (manual and automatic decision on the number of clusters) for the same data, it was important to know how they perform and how similar the results are. For this reason, it was relevant to compare the resulting clustering not only in a qualitative way (which could be subjective), but also in a quantitative way that one can assess the similarity of the two clusterings numerically, and thus the effectiveness of the automatic clustering.

2.7.1 Maximum-Match-Measure

The method adopted was Maximum-Match-Measure $\mathcal{M}(\mathcal{C}, \mathcal{C}')$ [17], which tries to match clusters that have a maximum absolute or relative overlap. The method can be summarized in the following way: it looks for the largest entry m_{ab} of the confusion matrix M (which consists of the numbers of matching elements for each cluster pair in the two clusterings to compare) and match the corresponding clusters C_a and C'_b (this is the cluster pair with the largest (absolute) number of matched elements). Then, cross out the a-th row and the b-th column and repeat this step (searching for the maximum entry, matching the corresponding clusters and deleting the corresponding row and column) until the matrix M has size 0. Afterwards,

simply sum up the matches and divide it by the total number of elements:

$$\mathcal{MM}(\mathcal{C}, \mathcal{C}') = \frac{\sum_{i=1}^{min\{k,l\}} m_{ii'}}{n} \tag{2.20}$$

where i' is the index of the cluster in $\mathcal{C}'$ that is matched to cluster i of clustering $\mathcal{C}$. Notice that in the case of $k \neq l$ (the number of clusters in each clustering result), this measure completely discards the $\|k - l\|$ remaining clusters in the clustering with a higher cardinality (i.e. the number of clusters).

2.8 Time series and clustering

Because of the nature of the boat data (measurements recorded with a certain frequency), they can be considered as time series data [18]. Generally speaking, a time series is a sequence of data point indexed in time order. Commonly, the data is taken at successive equally spaced points in time (for instance: 1 second, 10 minutes, 24 hours, 7 days, 1 year), so the time series can be analyzed and processed as discrete-time data.

The domain of application of time series data includes weather forecasting, statistics, signal processing, pattern recognition, econometrics, mathematical finance, control engineering, communications engineering, aeronautics, earthquake prediction, electroencephalography and most applications involving temporal measurements. In fact, time series data is widely used in any domain of applied science and engineering which involves temporal measurements.

Time series analysis comprises methods for analyzing time series data in order to extract meaningful statistics and other characteristics of the data. A common and typical task of time series analysis is to compare the similarity between two sequences. In terms of time series, the lengths of two sequences that need to be compared may not be equal. For example, in the speech recognition field, the speeds of speech of different people could vary significantly. Because of the considerable randomness that the voice signal has, even if the same person repeats the same word several times, the sequences produced would not have the same length. Moreover,

the phonemes within the same word may have different pronunciation speeds for different person. For instance, some people may drawl the 'a' or shorten the 'i'. All these issues make the comparison between time series a hard task, and it requires a subtle algorithm addressed for these issues.

2.8.1 Time series clustering

In the last two decades, as part of the effort in temporal data mining research, an increasing interest in time series clustering has been sparked, as it has been shown effective in providing useful information in various domains.

Time series clustering is mostly used for discovering interesting patterns among the sequences, which include frequently appearing and surprising patterns. These tasks are also called motif discovery and anomaly detection/discord detection, respectively. As stated in [19], finding clusters among the time series could also be useful in different domains for:

- Recognizing dynamic changes in time-series: for example, in financial datasets, it can be used to find the companies with similar movements of the stock price (detection of correlation between time series) or, to find days with similar evolution of wind, in the context of this project.

- Prediction and recommendation: a hybrid technique combining clustering and function approximation per cluster can help the user to predict and recommend. For example, in scientific databases, it can address problems such as finding the patterns of solar magnetic wind to predict today's pattern.

Clustering time series data has been used in diverse scientific areas to discover interesting patterns which empower data analysts to extract valuable information from complex and massive datasets. A bunch of time series clustering works that have been published in the open literature are presented in the review and survey [19] [20]. Some popular applications for common fields are:

- Medicine: it is applied to functional MRI data (univariate time series of equal length) in order (i) to provide the functional maps of human brain activity

on the application of a stimulus, (ii) to unveil regional abnormalities of brain perfusion characterized by differences of signal magnitude and dynamics in contrast-enhanced cerebral perfusion MRI, and (iii) to analyze suspicious lesions in patients with breast cancer in dynamic MRI mammography data. Also, time series clustering is used to detect diseases like *Hypokalemia* (a disease where the heart system is deficient in potassium), which is often diagnosed by examining electrocardiograms for increased amplitude and width of the P-wave. These applications allow the automatic diagnosis of the potential diseases a patient may have, without too much human intervention.

- Environment and urban: foreshocks, aftershocks, triggered earthquakes detection by identifying repeated patterns (motifs); clustering population distribution; identification of similar velocity flows.

- Finance: it is applied to find seasonality patterns of retail data; discovery patterns from stock time series; personal income pattern.

- Energy: analysis of a country's energy consumption; consumer's daily power consumption patterns for the segmentation of markets.

- Speech/voice recognition: speaker verification.

In the literature, there are mainly three categories to which time series clustering methods belong:

- **Whole sequence clustering** is similar to that of conventional clustering of discrete objects. Given a set of individual time series data, the goal is to group similar time series into the same cluster. Although it is quite similar to conventional clustering, it requires further processing on the data.

- **Subsequence clustering** means clustering on a set of subsequences of a time series that are extracted via a sliding window, that is, clustering of segments from a single long sequence. However, Keogh et al. [21] claimed that applying clustering approaches to discover motifs is meaningless when focusing on time series subsequence. This is because when using a sliding window to split the

long time series into subsequences in fixed window size, patterns, which are derivations from sine curve, are always resulted no matter how the shape of the given time series is.

- **Time point clustering** is a clustering of time points based on a combination of their temporal proximity and the similarity of the corresponding values. This approach is similar to time series segmentation whose typical application is speech diarization, which is the process of partitioning an input audio stream into homogeneous segments according to the speaker identity. However, it is different from segmentation since not all points need to be assigned to clusters, i.e. some of them are considered as noise. The objective of time point clustering is finding the clusters of time points instead of clusters of time series data.

In this project, we want to explore the evolution of wind during the entire days. Hence, whole sequence clustering approach would be applied. The fundamental components of time series clustering are:

- Dimensionality reduction/time series representation: the objective is to represent the raw time series in another space by transforming sequences to a lower dimensional space or by feature extraction. Applying dimensionality reduction is important because i) it reduces the memory requirements for storing the whole raw sequences, ii) it can significantly speed up the distance measurements and iii) it can reduce the effect of noises in the data.

- Distance measure: as in conventional clustering, a similarity measure is required for the computation of distances between sequences. If all time series are of equal length (which rarely happens), standard clustering techniques can be applied by considering each time series as a long vector using Euclidean distance. Nevertheless, this approach would not take into account the similarities in shape that different sequences could possess. Distance measures between time series objects can be divided into three categories, namely:

1. Shape-based: shapes of two sequences are matched as well as possible, by a non-linear stretching and contracting of the time axes (also called time warping). This approach usually uses conventional clustering methods with a modified distance measure.
2. Feature-based: the raw time series is converted into a feature vector of lower dimension. Later, a conventional clustering algorithm is applied to the extracted feature vectors. Usually, in this approach, feature vectors with equal length will be extracted and then Euclidean distance based measurement can be applied.
3. Model-based: raw sequences are transformed into model parameters (a parametric model for each sequence) and then an appropriate model distance is applied.

Some popular measures are Dynamic Time Warping, Hidden Markov model (HMM) based distance, Longest Common Subsequence, etc.

- Prototype definition: finding the representative of a cluster is an essential part of clustering approaches, especially in partitioning-based clustering algorithms like K-means, K-medoids and Fuzzy C-means.

- Clustering algorithm: just like conventional data clustering, time series clustering requires a clustering algorithm or procedure to form clusters given a set of unlabeled data objects and the choice of clustering algorithm depends both on the type of available data and on the particular purpose and application. There are generally three different approaches to cluster time series data, namely:

  1. Customizing the existing conventional clustering algorithms (which work with static equal length data) such that they become compatible with the nature of time series data. The distance measures are usually modified in this approach so that they are compatible with sequences which vary in length.

2. Converting time-series data into simple objects (static data) as the input of conventional clustering algorithms.

3. Using multi resolutions of time series as the input of a multi-step approach.

There are many researches that focus on time series clustering algorithms. As mentioned in [20], the mainstream is to adapt existing conventional clustering algorithms (like Hierarchical and K-means clustering) so that they can deal with time series data, this is usually done by defining a distance measure that is compatible with time series of unequal length. In this project, we adopted the solution in point 1 (customizing the existing conventional clustering algorithms) to make use of the developed framework, and at the same time, we applied the necessary processes and modifications that correspond to the four components as described above. Concretely, we will use Principal Component Analysis and Piecewise Linear Representation for dimensionality reduction, Dynamic Time Warping as our distance measure and we apply Local Search for Prototyping for the computation of centroid. The mentioned techniques will be described in chapter 4.

Chapter 3

Datasets

3.1 Data sources

For this project, two kinds of data source are available. On the one hand, we have data come from Numerical Weather Prediction (NWP) [22] [23] which uses mathematical models of the oceans and atmosphere to predict the weather based on current weather conditions. Both current state of the weather and the numeric model play an important role in the prediction. Current weather observations, after applying a process called data assimilation, are the input to the mathematical models to produce outputs of meteorological parameters such as wind speed, wind direction, temperature, pressure, and hundreds of other parameters from the oceans to the top of the atmosphere. On the other hand, real sensors based data are gathered by boats that move around the sea area where the competition will take place. In the following sections, a more detailed description of the two data sources is provided.

3.2 WRF Japan

The Weather Research and Forecasting (WRF) Model [24] [25] is a next-generation mesoscale numerical weather prediction system, started in the late 1990s, which is designed for both atmospheric research and operational forecasting applications. It was a collaborative partnership of many American research centres. It features two

dynamical (computational) cores, a data assimilation system, supporting parallel computation and system extension. The model serves a wide range of meteorological applications across scales from tens of meters to thousands of kilometres. The WRF data is usually not freely available to the public due to the computational resources the model consumes. In our project, the WRF data for Japan was provided by TriM. In total, 13 meteorological parameters (in contrast with the 6 parameters in the previous project) were predicted by the model, which are:

- Wind u, v components in knots.
- Temperature at 2 meters and at the ground, the unit is degree Celsius.
- Mean sea level pressure in hectopascal (hPa).
- Wind gust in knots.
- Low/high/medium/total cloud coverage in percentage.
- Land cover surface, in m^2.
- Relative humidity in percentage.
- Total precipitation, the unit used is mm/h.

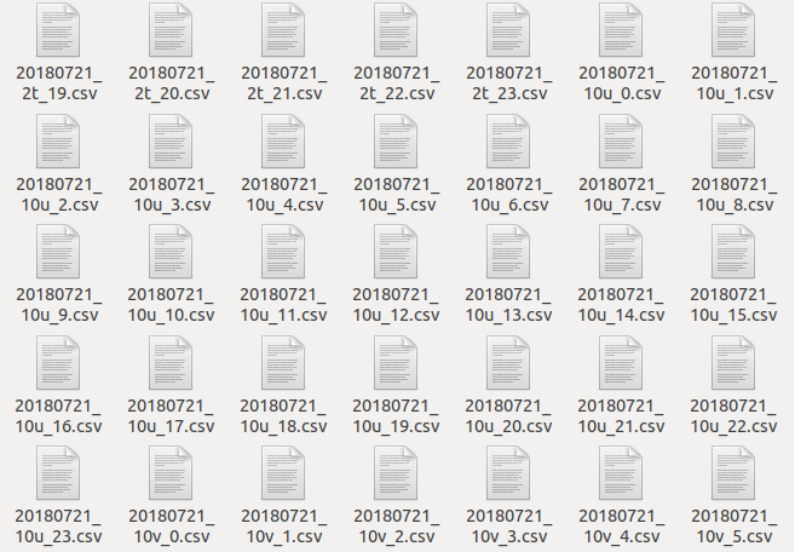

Figure 3.1: A glance at WRF Japan files

Figure 3.1 represents part of the CSV files that compose the WRF Japan dataset. As we can see, the filename consists of 3 parts: the date, the code associated with the meteorological parameter, and the hour. The data set covers dates from 21 July 2018 to 15 September 2018. For each date and hour, we have 13 files that contain the predicted value of the mentioned weather parameters for various stations. Figure 3.2 shows partial content that a file will have. The first two columns correspond to latitude and longitude of each station. The last column is then, the forecasted value.

	A	B	C
1	34.9450000000014	139.259999999997	7.9518359375
2	34.9450000000014	139.309999999997	8.2498359375
3	34.9450000000014	139.359999999996	9.1318359375
4	34.9450000000014	139.409999999996	11.0938359375
5	34.9450000000014	139.459999999996	13.7118359375
6	34.9450000000014	139.509999999996	15.6458359375
7	34.9450000000014	139.564999999996	14.5538359375
8	34.9450000000014	139.614999999996	9.9138359375
9	34.9450000000014	139.664999999996	7.3848359375
10	34.9450000000014	139.714999999996	9.5518359375
11	34.9850000000014	139.259999999997	7.4508359375
12	34.9850000000014	139.309999999997	7.6708359375
13	34.9850000000014	139.359999999996	7.9348359375
14	34.9850000000014	139.409999999996	10.0578359375
15	34.9850000000014	139.459999999996	12.5998359375
16	34.9850000000014	139.509999999996	14.4428359375
17	34.9850000000014	139.564999999996	14.7678359375
18	34.9850000000014	139.614999999996	11.9178359375
19	34.9850000000014	139.664999999996	7.2468359375
20	34.9850000000014	139.714999999996	8.0878359375

Figure 3.2: Content of WRF Japan file

3.3 Boat data

As mentioned before, another important data source consists of real sensors based data. Within this project, several rigid inflatable boats (ribs) have been equipped with a weather system able to measure weather variables on the sea during trainings and racing in Enoshima Bay, the sailing venue of the next Olympic Games Tokyo 2020.

Collected data refer to the periods from 21 July 2018 to 10 August 2018 and from 25 August 2018 to 15 September 2018. These data are being collected from the Austrian federation's moving ribs of the coaches that follow the sailors while they train. Additional data come from the Olympic Committee Tokyo 2020 and

have been collected by 6 stationary British ribs during the period from 25 July 2018 to 6 August 2018.

From figure 3.3, we can see that we have 6 folders which correspond to the six ribs that move around, and a csv file that corresponds to the dataset of anchored/stationary ribs. Inside each folder, the files are represented as in figure 3.4. The filename indicates the date that the data corresponds to.

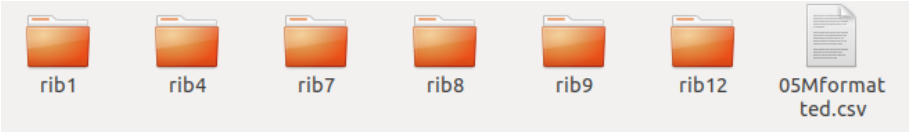

Figure 3.3: A glance at Boat datasets

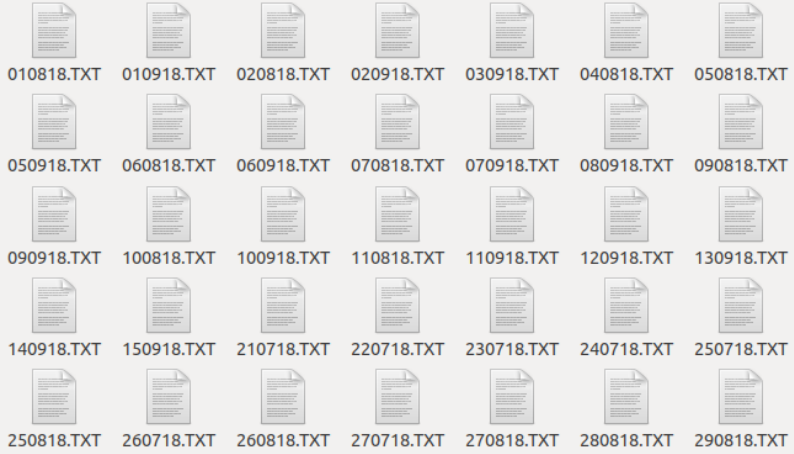

Figure 3.4: Composition of moving boats dataset

3.3.1 Moving boats dataset

The format that moving boats dataset follows is **Texys Marine frame format** which contains the below information that will be used for the analysis:

1. DATE: DDMMYY and HOUR: HHMMSS.CC (centiseconds)
2. LAT: DDMM,SS (latitude in DMS)
3. LONG: DDMM.SS (longitute in DMS)
4. TWS: NN.NN (knots) and TWD : DDD.DD (0 - 360 degrees)

5. Windflag:
    - 0: Valid data
    - 1, 2, 4: excessive speed in the case of upwind, downwind and low relative speed
    - 8: In case of curvature

Take the following record as example:
$TEXYS,**20817,112332.2,4734.3466796875**,N,**300.8052368164**,W,201.84,
11.07,201.84,56.26,12.49,3.4,132.04,1.7,0,0,**0*6E**

We can extract information regarding date (02/08/17), time (11h 23mn 32.2sec), latitude (47°34′34.66796875), longitude (3°00′80.52368164), TWS (11.07 knots), and TWD (201.84 degrees). We also know the record is valid since its wind flag is 0.

The frequency of data collection is 5 Hz, therefore, the total amount of records is tremendous.

3.3.1.1 Issues

However, after exploring the datasets, we found that there are many issues that we need to handle, whether they are human-caused or sensor-caused. For example, as we would expect, there could be sensors failures or, humans (the coaches in this case) can forget to switch the data logger on. The records that we considered as invalid data are:

- Records with non-zero windflag. For instance:

  $TEXYS,8,010818,074108.80,4735.4838867188,N,301.7031555176,W,180.00,
  0.63,180.00,0.00,0.00,0.92,0.00,0.00,0.0,0.0,**6*4E**

- Records with static speed, directions, or invalid latitude/longitude. For example:

  $TEXYS,8,010818,073601.40,**0.0000000000**,S,**0.0000000000**,W,0.00,**0.00**,
  0.00,0.00,0.00,0.00,0.00,0.00,0.0,0.0,0*66

Moreover, we could occasionally find the case that no records are registered between two timestamps. We call this situation as *gap* if the difference between the timestamps of two consecutive records is bigger than 10 minutes. Also, we have found cases such that the value of TWD or TWS changes radically in one second (at timestamp $peak_{begin}$). We say it's a *peak* if the difference of TWD is greater than 100 degrees or the difference of TWS is greater than 7 knots, for a difference of timestamp of 1 second. These peaks are normally caused by sudden accelerations, decelerations, violent rotations in the trajectory of the boats, movement caused by big waves, etc. Theoretically, they should be detected by the software of the data-logger (Windflag field), but we have found that it is not always the case.

If a peak appears, we check if the value come back to a safe range (that is, difference of TWD is less than 30 degrees and difference of TWS is less than 5 knots) in 10 minutes (the threshold to be considered as gap) to decide whether it's a real peak or just because the wind changes substantially. The cases we would find when seeking for safe value are:

- Case 1: Safe value is found in 10 minutes (at timestamp $peak_{end}$). What we do in this case is remove the records between $peak_{begin}$ and $peak_{end}$.
- Case 2: When looking for a safe value, a gap happens (at timestamp x). In this case, the readings restart in whatever value after the gap and the records between the timestamps $peak_{begin}$ and x will be discarded. This is a rare case, actually, we didn't find any of it.
- Case 3: Safe value cannot be found in 10 minutes, which means the wind conditions has changed radically (rare but could happen). In this case, the records between the timestamps are preserved.

3.3.2 Static boats: 05M dataset

The format of data recorded through anchored ribs is **05Mformatted**. In the 05M dataset, the data are recorded simultaneously by 1 to 6 stationary ribs (the number of active boats is not a constant). Collected data refer to the periods from 25 July 2018 to 06 August 2018 (except 28 July 2018). Table 3.1 shows an example of the

records. The column deciveId is the numbering of the boat. The columns lat & lon represent the position of the anchored boat. The units of true wind direction and true wind speed are degree and knots, respectively. The frequency of sampling is one record per 5 minutes. This kind of data is quite similar to the WRF Japan dataset as the anchored boats can be considered as fixed stations.

username	source	deviceId	lat	lon	timeUTC	twd	tws
trim	05M	1	35.29166	139.49333	2018-07-25 02:40:00	132	6.1
trim	05M	4	35.29	139.51666	2018-07-25 02:40:00	111	5.5
trim	05M	5	35.27	139.54333	2018-07-25 02:40:00	90	5.5

Table 3.1: Example of records of 05M dataset

3.4 Inputs for clustering algorithm

Taking all the mentioned types of data into account, in the end, we have mainly three kinds of input for the clustering algorithms:

1. For WRF Japan dataset, as in the previous work, hourly weather data of various station for a whole period is provided.

2. For Boat data, similar to the WRF Japan dataset, the ribs are considered as virtual stations and the data recorded by sensors will be split by an interval of half an hour (i.e. starting from minute 0 and 30), through averaging all the records within the interval.

3. Additionally, for the newly introduced clustering type described in section 2.8, time series will be constructed for each rib and date, by means of using the entire readings of a boat for one day. This new kind of input is used for extracting patterns for the evolution of each day as a whole. We compare the set of the data from each day so that we can capture if the evolution of different days is similar, as usual, according to the wind components. Note that each element is going to be identified by the pair (day, rib). We could have decided to average somehow the data from different ribs for one day, but

in the end, we have preferred to consider the data from each rib as different elements because of the following facts:

- This might allow us to detect different daily evolution depending on the areas where the ribs are located.
- This will allow us to evaluate in some way the quality of the clustering: one would guess that the ribs for the same days should be mainly located in the same clusters.

In summary, types 1 and 2 will be used for conventional clustering (like in the previous project) using hourly wind fields. And input type 3 will be used for time series clustering.

3.5 Area of measurements

This project focus on the analysis of wind patterns of Enoshima Bay, Fujisawa, where the sailing race will take place. As Kaufmann and Whiteman stated in [6], data of meteorological stations were used. In this project, one hundred coordinates of the WRF model distributed over an area of 45 km^2 were taken to simulate meteorological stations (we will refer them as stations as well). The locations of the stations can be seen in figure 3.5.

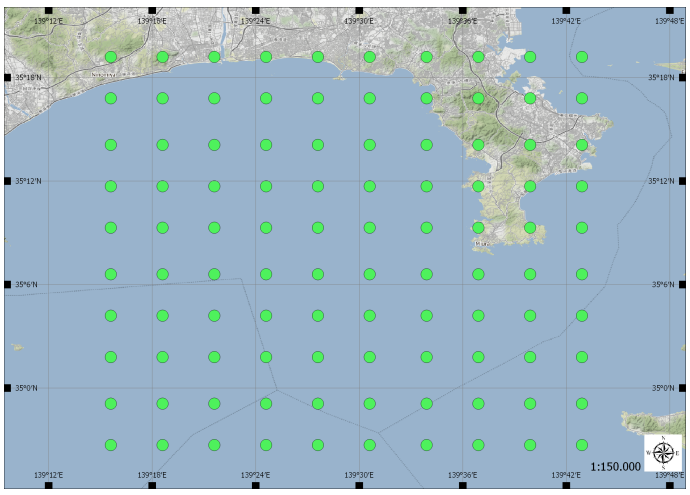

Figure 3.5: Stations of WRF Japan dataset

In a similar way, as the ribs of 05M dataset are anchored, they can be considered as stations as well. We can see the locations of the stationary ribs in figure 3.6.

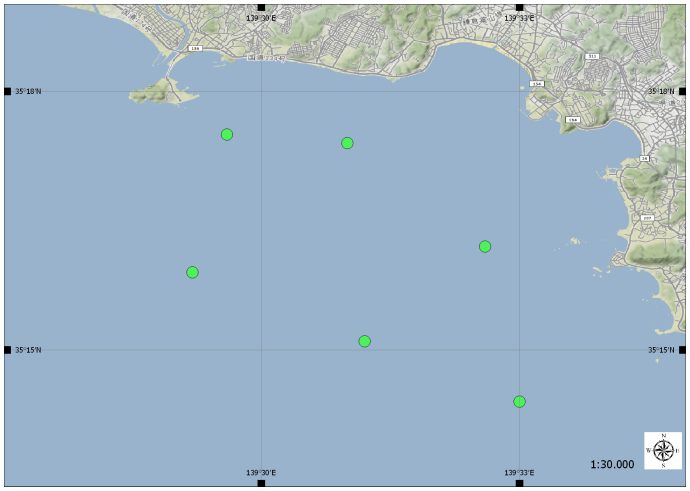

Figure 3.6: Position of stationary ribs

Finally, for moving boats dataset, since the ribs are moving around constantly, we don't have fixed stations. The area of measurement can be found in the figure below, where the moving ribs will mainly move around the racing areas surrounded by red circles. And in fact, each of the 6 stationary ribs is located in each of the 6 racing areas.

In the Olympic games, these racing areas are fixed and they are assigned to a different class of sailings boat for each day of the competition.

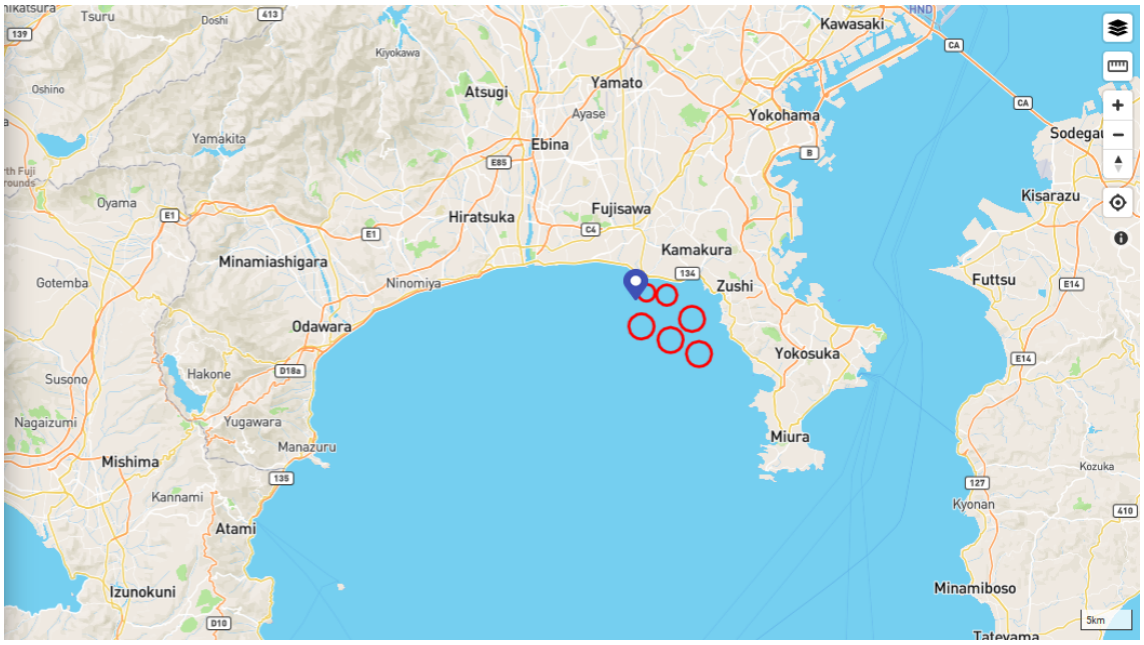

Figure 3.7: Area of measurements of the moving ribs

Chapter 4

Architecture and Implementation

4.1 Architecture

Similar to the workflow mentioned in section 2.3, the procedure that we have followed in this project is as shown in the figure below.

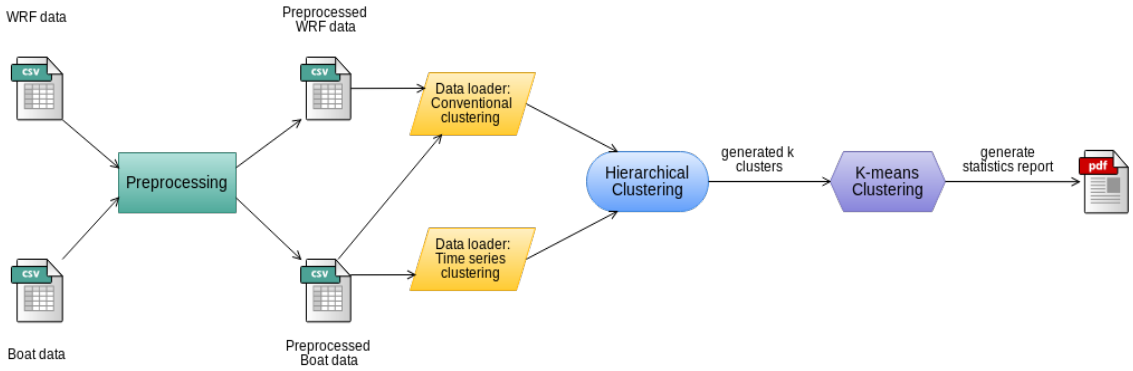

Figure 4.1: Work flow of the current project.

As we have described in chapter 3, our sources of data include **WRF Japan** and **Boat data**. Once we obtain the data, the next immediate step is to perform **Data Preprocessing**, mainly due to the discussed issues of Boat data. Afterwards, data will be loaded differently, depending on the analysis we want to perform. For time series clustering, data would require further processing so that can be fed to the clustering module. More specifically, as described in section 2.8.1, dimensionality reduction techniques like **Principal Components Analysis** and **Piecewise Linear Representation** will be used. In the next sections, a detailed description

of these methods will be provided. Then, as the basis of the clustering framework, **Hierarchical Clustering** and **K-means Clustering** will be performed sequentially to produce clustering results. Note that for time series clustering, a delicate distance measure called **Dynamic Time Warping** which supports unequal length sequences, is used. We also have to bear in mind that the computation of centroid is different for the two types of clustering. Subsequently, **statistics report** will be automatically generated so that the meteorologists can analyze on the obtained wind patterns.

4.2 Preprocessing

For **WRF Japan**, the meteorologists are interested in some specific points of the marine zone where the competition is carried out. Since these points may not match exactly the points provided by the WRF model (for simplicity we named them as WRF points), it is necessary to find the closest points between the interested points and the WRF points, using geodesic distance [26]. The library that we have adopted for computing geodesic distance can be found on the site [27], which is a well known Geocoding library for Python. After knowing which are the WRF points that we would use for the next steps, we use them to filter out the rows that do not correspond to those points, for each of the WRF files.

The pseudocode of this process can be found in algorithm 4.1. From line 1 to 3, we create the list of interested points and WRF points, correspondingly, and we create an empty set for closest points. From line 4 to 13, for each interested point, we iterate over the list of WRF points to find its closest point using geodesic distance. From line 15 to 22, we filter the files in the folder called $directory\_to\_filter$ by selecting only the rows that correspond to the closest points and store the filtered files in the folder called $directory\_to\_store$.

The famous phrase states: "Garbage in, garbage out". **Boat data**, in particular, data collected by moving ribs, as discussed in section 3.3.1.1, contains many invalid records that would have a great impact on the quality of clustering. Therefore, data cleansing is a must for this kind of data. First we have to remove the records that

have non-zero Windflag, invalid latitude/longitude, incorrect number of variables, incorrect format, etc. Then we have to identify the possible gaps and peaks. In the meanwhile, auxiliary files that register the gaps and peaks, the starting and ending timestamp of the dates are generated automatically. With these files, we finally decide which are the dates and hours we should consider for the next steps.

Algorithm 4.1 Filter WRF data

Function *filter_WRF_files(file_ips, directory_to_filter, directory_to_store)* **:**

Input: file_ips: the file that contains the interested points, directory_to_filter: directory that contains the files to be filtered, directory_to_store: directory where the filtered files are stored

Output: filtered files stored in directory_to_store

```
1  interested_points ← create_ineterested_points(file_ips)
2  WRF_points ← create_WRF_points(arbitrary file within directory_to_filter)
3  closest_points ← set()
4  for ip in interested_points do
5      min_distance_point = None
6      min_distance = inf
7      for wrfp in WRF_points do
8          actual_distance = geodesic_distance(rp, wrfp)
9          if actual_distance < min_distance then
10             min_distance = actual_distance
11             min_distance_point = wrfp
12     end
13     closest_points.add(min_distance_point)
14 end
15 for file in directory_to_filter do
16     filtered_rows = []
17     for row in file do
18         if (row.lat, row.long) in closest_points then
19             filtered_rows.append(row)
20     end
21     save_file(directory_to_store, file.filename, filtered_rows)
22 end
end
```

From line 1 to 4 of the algorithm 4.2, we iterate over the files within the directory.

Then, for each row of a file, we check its validity (line 5 to 6) according to the conditions mentioned in section 3.3.1.1. If the row is considered as valid, from line 7 to 10, we look for the potential gap and/or peak that may take place. From line 11 to 19, we deal with different cases that could happen when a peak is found. On the one hand, if a gap is found in the same time, we could find one of the following cases, depending on whether a peak has been encountered in the previous readings (more details are explained in the section 3.3.1.1):

- Case 2: this case happens when a gap is found when looking for safe value, given that a peak is found previously.
- Case 3: this is the case when the value of TWD/TWS does not back to a safe value in 10 minutes.

Then, we register the peak according to the identified case and update the related variables. On the other hand, if a gap is not presented at the same time, then it will be the situation that a real peak is encountered. Hence, we update the related variables and goes directly to the next iteration. From line 20 - 24, we deal with the Case 1, in which the value of TWD/TWS goes back to safe value respect the value before the peak takes place. From line 25 - 27, we update the information regarding the beginning and ending time, the last row, and add the current row to the array *valid_rows*. Finally, after iterating all the rows of a file, we register the corresponding information to the CSV file that records the valid dates. Meanwhile, we also store the filtered files in the *dir_to_store* directory with the same filename.

Algorithm 4.2 Filter Boat data

Function *filter_boat_files(source_dir, dir_to_store, min_hour, max_hour)* **:**

Input: source_dir: the folder that contains the boat data, dir_to_store: directory where the filtered files are stored, min_hour: desired starting hour, max_hour: desired ending hour

Output: filtered files stored in directory_to_store

```
1  initialization()
2  for rib_dir in source_dir do
3      for file in rib_dir do
4          initialization_per_file()
5          for row in file do
6              if is_valid(row) then
7                  if row_timestamp - last_row_timestamp ≥ gap then
8                      gap_appears ← True
9                      register_gap_to_csv()
10                 peak_appears ← check_if_peak_appears()
11                 if peak_appears then
12                     if gap_appears then
13                         peak_case ← determine_peak_case()
14                         register_peak_to_csv(peak_case)
15                         update_peak_related_variables()
16                     else
17                         update_peak_related_variables()
18                         continue
19                     end
20                 else
21                     if peak is already detected before then
22                         register_peak_to_csv(1) ; // case 1, values went back
                            to safe range
23                         update_peak_related_variables()
24                 end
25                 update_beginning_ending_time()
26                 update_last_row_info()
27                 valid_rows.append(row)
28         end
29         register_valid_date()
30         save_file(dir_to_store, file.filename, valid_rows)
31     end
32 end
end
```

4.3 Data loading

When we want to proceed with any analysis, we load first the preprocessed data into the corresponding data structure, depending on the dataset and the type of clustering.

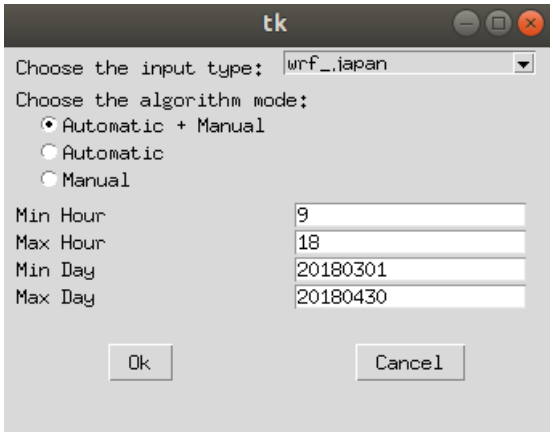

Figure 4.2: Initial screen of the application.

From the pop-up window above, we can choose the algorithm mode which corresponds to the manual/automatic procedure for the decision of the number of clusters. However, as mentioned in the previous sections, we only use the manual procedure in this project. Also, we are able to choose the type of input for the clustering module described in section 3.4 (in this case, the chosen input is of type 1 for WRF conventional clustering. Other options that can be selected in the drop-down list are: "m5" which corresponds to conventional clustering for stationary ribs; "boat" that corresponds to Boat conventional clustering; and "boat_series_complete" which corresponds to Boat time series clustering), and specify filters on dates and hours. The default interval of hours is from 1:00 a.m. to 10:00 a.m. UTC (which corresponds to 10:00 a.m. - 7:00 p.m. Japan time). Since the sailing race will take place during the daytime and afternoon, it was decided to discard the data correspond to nighttime that could have an influence on the results.

To represent a timestamp and its measurements, a custom class *HourData* was implemented. It is characterised by date time and the data structure representing the measurements of weather. In case of conventional clustering, it is a list of objects of the custom class *Parameters*, which contains information regarding the coordinates

of the stations and the meteorological parameters like velocity components and pressure, humidity, etc. In case of time series clustering, the data structure will contain a list that includes all the readings for a date, and the id of the boat.

Note that the ideal situation would be that in the ribs we also have sensors to measure all the weather parameters. However, the ribs are only equipped with sensors for the measurements of TWS and TWD, by now. That is the reason why we had to use the meteorological parameters of WRF data. Unlike the wind, these other weather parameters are far less problematic to predict. Therefore, of course we are not sticking to the actual data (hopefully more meteorological parameters for Boat data will be available in the future) but they should be quite close.

As a new type of clustering (i.e. time series clustering) is introduced, we have to process the raw sequences firstly so that they can be used properly as inputs for the clustering algorithms. In the following sections, two methods applied for dimensionality reduction and time series representing are presented.

4.3.1 Principal Component Analysis

The purpose of Principal Component Analysis (PCA) is to project the cloud of points upon a subspace (a plan) to retain the maximum of the original cloud information. It is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. And it is a sophisticated method to overcome several issues related to high dimensionality. Tanaka et al. [28] represent a method for transforming multi-dimensional time-series data into 1-dimensional time-series data. This method is very suitable for our purpose, as we will see later in section 4.3.2, the method used for representing time series only supports 1-dimensional sequences.

The procedure computes firstly the covariance matrix A_{T_m}:

$$A_{T_m} = \begin{bmatrix} \sum_t x_{1t}x_{1t} & \sum_t x_{1t}x_{2t} & \cdots & \sum_t x_{1t}x_{mt} \\ \sum_t x_{2t}x_{1t} & \sum_t x_{2t}x_{2t} & \cdots & \sum_t x_{2t}x_{mt} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_t x_{mt}x_{1t} & \sum_t x_{mt}x_{2t} & \cdots & \sum_t x_{mt}x_{mt} \end{bmatrix} \tag{4.1}$$

The eigenvalues λ_i of the above matrix is ordered as $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m$. And the eigenvector is represented as $[e_1\lambda_i \; e_2\lambda_i \; \cdots \; e_m\lambda_i]$. Then, the i-th principal component pc_{t,λ_i} is calculated by using means of each time series x_1, x_2, $\cdots$, x_m.

$$pc_{t,\lambda_i} = e_{1\lambda_i}(x_{1t} - \overline{x}_1) + e_{2\lambda_i}(x_{2t} - \overline{x}_2) + \cdots + e_{m\lambda_i}(x_{mt} - \overline{x}_m) \tag{4.2}$$

In their approach, the first principal component, which explains the most of variability of the data, is used to transform the multi-dimensional time series data into 1-d time series data in an effective way. The final 1-d data T is as follows:

$$T = x_1, x_2, \cdots, x_n \tag{4.3}$$

$$x_t = e_{1\lambda_1}(x_{1t} - \overline{x}_1) + e_{2\lambda_1}(x_{2t} - \overline{x}_2) + \cdots + e_{m\lambda_1}(x_{mt} - \overline{x}_m) \tag{4.4}$$

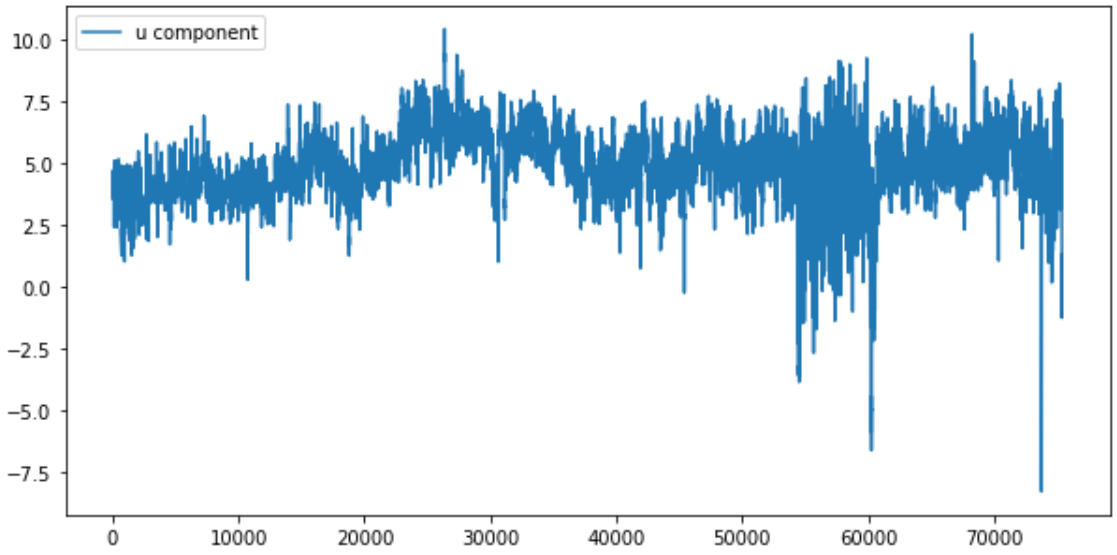

Figure 4.3: u component of the daily time series of a boat.

Figures 4.3 and 4.4 represent the evolution of u, v components, measured by the sensors of a boat on a specific day. From figure 4.5, we can observe that the characteristic patterns are preserved, as PCA dynamically detects the significant coordinates of the original data.

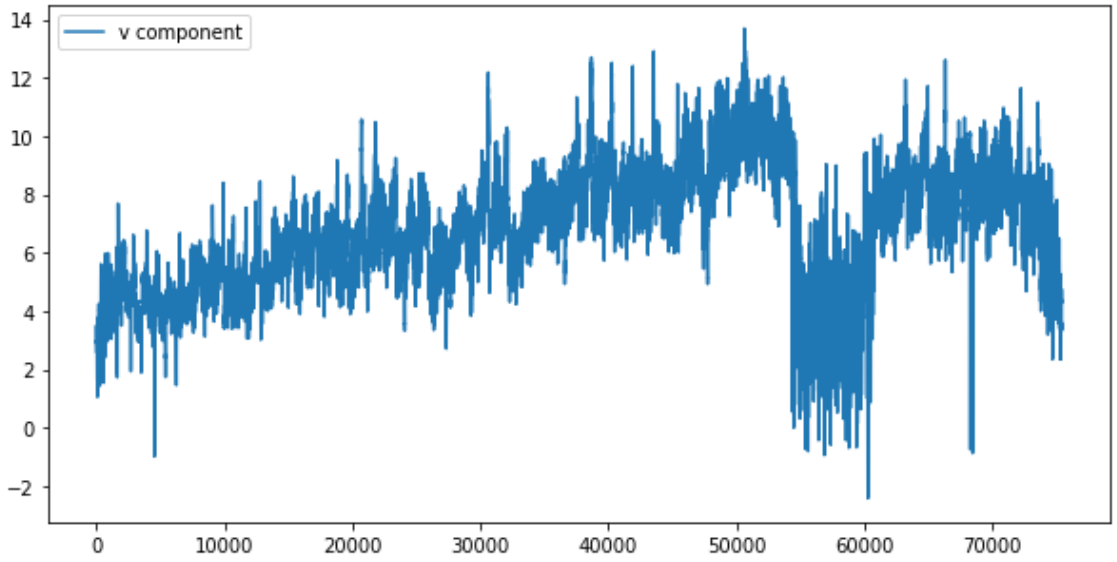

Figure 4.4: v component of the daily time series of a boat.

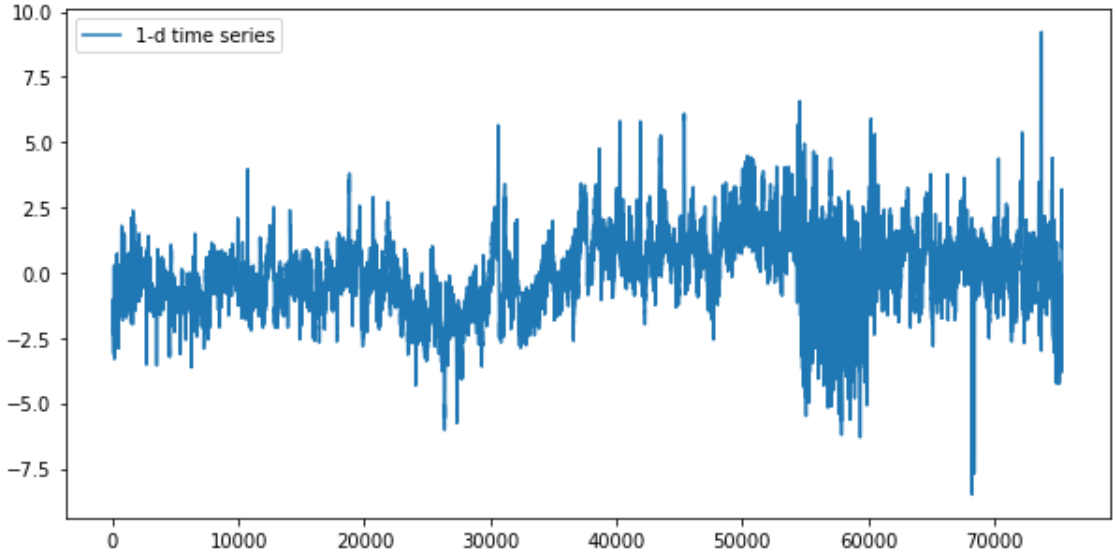

Figure 4.5: Obtained 1-D time series after applying PCA.

4.3.2 Piecewise Linear Representation

Due to the huge amount of boat's sensor data, the sizes of the time series that we deal with are indeed, very large. Consequently, it causes that the distance measurement for sequences becomes extremely slow. Therefore, dimensional reduction (i.e. the number of data points) is a must for dealing with this kind of data. In Fu's survey [29] regarding time series data mining, a bunch of proposals for time series representation are reviewed. Among the large variety of time series representation related researches, an approach is to approximate a time series with straight lines. Two major categories are involved. The first one is using linear interpolation and the other one is preserving the salient points, called as perceptually important points (PIP). In this project, we have adopted the common method proposed by Eamonn Keogh [30], which uses piecewise linear representation (PLR). The algorithm works as follows (algorithm 4.3):

We begin by approximating the given time series T of n points, with j linear segments, where $j = \lfloor n/3 \rfloor$ (line 1). At this stage, each segment contains 3 points (allowing the last segment to contains more points). Each segment is the best-fit line through its collection of points, determined by using the classic linear regression

$$y - \overline{y} = \frac{S_{xy}}{{S_x}^2}(x - \overline{x}) \tag{4.5}$$

Naturally, the segment will not fit the data perfectly. Therefore, they will produce some amount of residual error for each point they approximate, which is defined as the vertical distance from the data points to the best-fit line $(d_1, d_2, \cdots, d_j)$. The normalized residual error, measuring how good a segment is approximating its collection of data (line 2 to 4), is defined as:

$$e_i = \frac{\sum_{m=1}^{j} {d_m}^2}{j} \tag{4.6}$$

In general, e_i's will show noticeable variance. This variability is captured by defining the Balance of error for k segments in the following way (line 5):

$$B_k = std(e_1, e_2, \cdots, e_k) \tag{4.7}$$

The algorithm begins by merging two neighbouring segments to produce a new approximation of the time series with $j - 1$ segments until there's only one single line approximation. The criteria for selecting the pair to merge is that merging them will give the minimum value of B_k in the next iteration. And the same idea applies when selecting the approximation to use (recall that we have one approximation for each value of j from $\lfloor n/3 \rfloor$ to 1).

Balance of error is a useful heuristic for determining whether an approximation is good or not. In the first iteration of our algorithm, the balance of error is mostly probably randomly distributed. In each subsequent iteration, the algorithm attempts to redress this until the number of segments remaining equals the true K. At that moment, all segments will have almost equal error norms. In the next iteration two of those segments must be merged, resulting in a larger segment which will have a very large error norm.

Algorithm 4.3 Piecewise linear representation

Function *PLR(T)* **:**
  Input: T: original time series of length n
  Output: segment representation of T
1 Approximate T with $\lfloor n/3 \rfloor$ segments, fit the segments using linear regression
2 **for** *each segment* **do**
3 compute normalized residual error e_i
4 **end**
5 $B_k \leftarrow$ the standard deviation of the residual errors
6 **while** *number of segments > 1* **do**
7 Merge the two consecutive segments which give the minimum B_k in the next iteration
8 Update then best-fit line
9 **end**
end

As we can observe from the figure 4.6 below, which is an example of one of our day-boat sequences corresponding to 31 July 2018 and rib number 12, from 10:00 a.m. to 3:00 p.m., Japan time. The segmented model appears to capture the essential shape of the underlying time series (with respect to the sequence in figure 4.5 which has 75404 data points) using 74 segments.

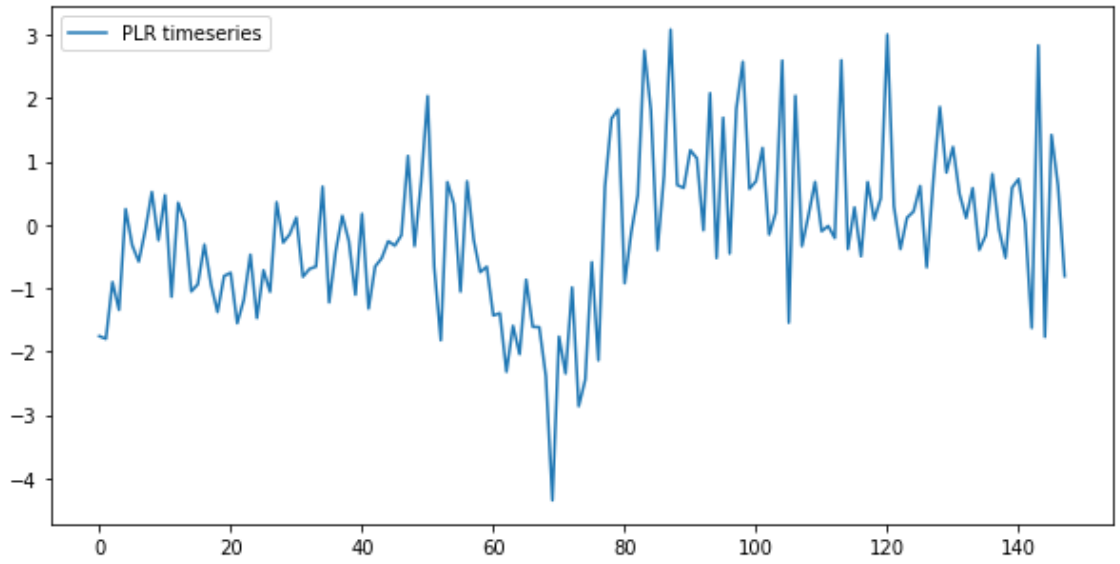

Figure 4.6: Time series representation using PLR.

Figures 4.7 and 4.8 represent the segmented u, v components by using the information of the segmentation (i.e. the starting and ending point). As we can see, the substantial shape is preserved with respect to the original u, v components, represented by the figures 4.3 and 4.4.

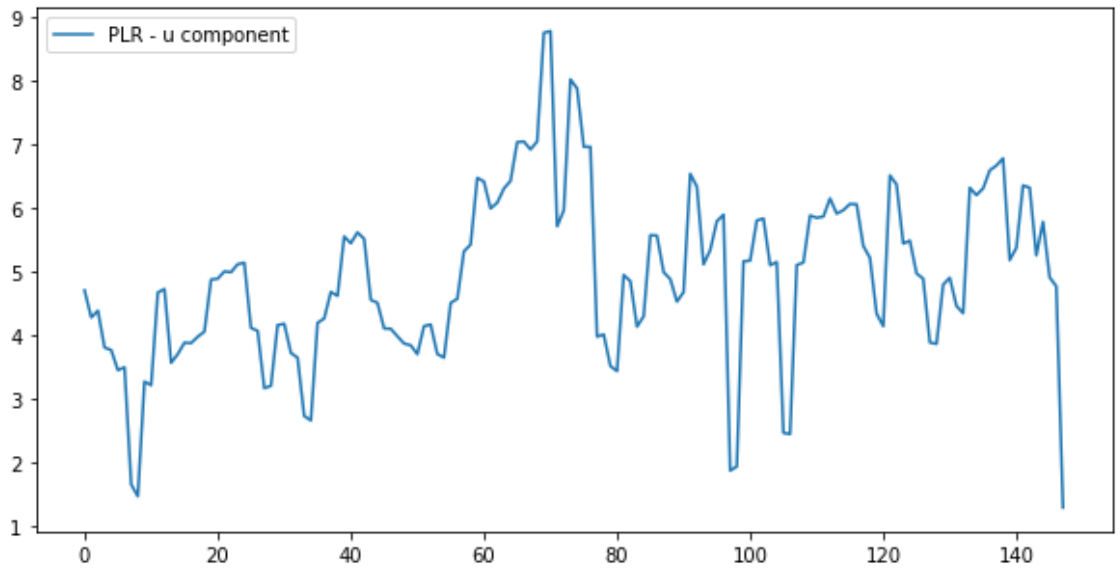

Figure 4.7: u component representation using PLR.

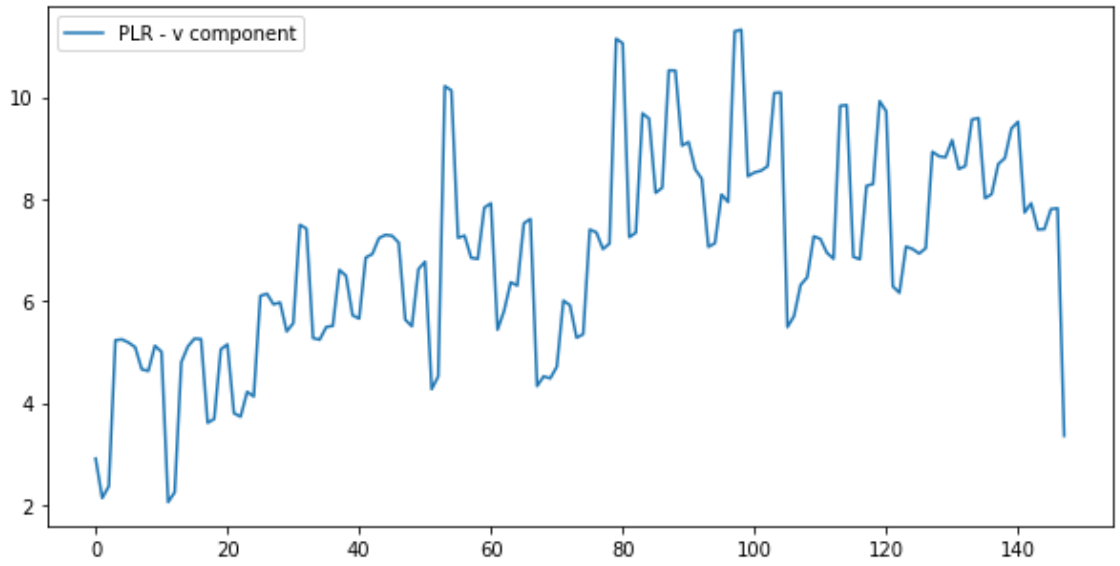

Figure 4.8: v component representation using PLR.

4.4 Normalization

In this project, we follow the same normalization procedure for **WRF Japan conventional clustering** as described in section 2.5. For **Boat conventional clustering**, since the boats move constantly, we do not have a fixed station as with WRF data. Therefore, we considered that it is not needed to do the time-average normalization and only spatial-average normalization are applied. With regard to **time series clustering**, the same reason for not doing time-average normalization is taken into account. However, the spatial-average normalization procedure changes a bit. That is, the wind components of each frame of the time series are divided by the overall spatial-average speed:

$$\tilde{u}_{ij} = \frac{u_{ij}}{s'_{overall}}, \qquad \tilde{v}_{ij} = \frac{v_{ij}}{s'_{overall}} \tag{4.8}$$

where the overall spatial-average speed is calculated in the following way:

$$s'_{overall} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{T_i}\sum_{j=1}^{T_i}({u_{ij}}^2 + {v_{ij}}^2)^{1/2}\right) \tag{4.9}$$

being N the number of time series in the dataset, and T_i is the length of the time series i.

4.5 Distance measure

In this project, different distance measures are used, depending on the type of clustering that we want to proceed. On the one hand, for WRF Japan conventional clustering, we continue using the measure proposed by Kaufmann and Whiteman, as described previously in section 2.6.1. On the other hand, for Boat conventional clustering, since we don't always have the same number of virtual stations (boats in this case) for two timestamps, it was decided to average firstly the wind components for each timestamp. Then the equation 2.19 is applied with the averaged components as if there's only one station for all timestamps.

Special attention should be paid for time series clustering. Under the complex circumstances introduces by time series data (mainly due to variant length and potential time displacement), the distance (or similarity) between two time series cannot be effectively computed using the conventional Euclidean distance. Taking into account these problems, and the fact that we want to **allow time displacement** since wind patterns of one day might happen in other days even if they are out of phase in the time axis (i.e. appear earlier or later), an algorithm called Dynamic Time Warping (DTW) [31] [32] [33] was chosen to overcome these issues. In the next section, the main concept and procedure of the DTW algorithm will be explained in a more precise way.

4.5.1 Dynamic Time Warping

DTW is one of the most used similarity measures for sequences that may vary in length, originally designed for the treatment of automatic speech recognition. And the main idea of DTW is to find the optimal global alignment between time series by exploiting temporal distortions between them.

In simple terms, given two discrete sequences (actually not necessarily related to time), the DTW can measure the degree of similarity or the distance between the

two sequences. At the same time, DTW can adapt to the extension or compression of the two sequences. For example, similarities between walking paths can be detected even if one person was walking faster than the other, or if there were acceleration and deceleration during the course of an observation. Since DTW is insensitive to the extension and compression of the sequence, therefore, it has been widely applied to temporal sequences of video, audio, and graphics data. Indeed, any data that can be turned into a linear sequence can be analyzed with DTW. Due to the simplicity and flexibility of DTW, it can solve many discrete time sequence matching problems, and it has many applications in a bunch of fields such as motion recognition, biological information comparison, etc.

For example, in figure 4.9, there are two regular sinusoidal sequences, in which the blue sequence is slightly elongated. We can observe that their overall waveform is similar, but they are not aligned respect the x-axis. Visually, we can say that the two sinusoids are highly similar, but it is obviously not reliable calculating the similarity by computing the Euclidean distances between the two series in a conventional way since they are not coincident in the x-axis. Fortunately, DTW can compute the optimal (with the least cumulative distance) alignment between points of two time series.

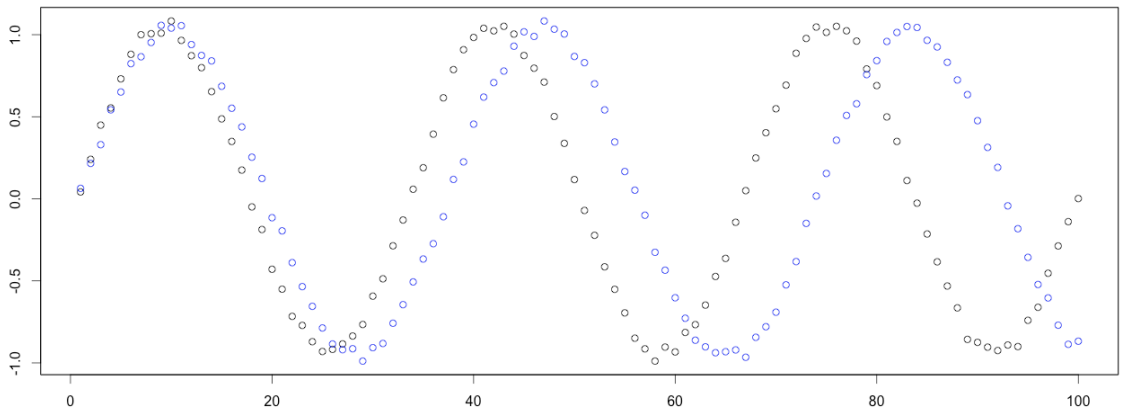

Figure 4.9: Regular sinusoids.

The dashed lines in figure 4.10, which connect the black and red curves, represent the alignment (or mapping) between the most similar points. DTW uses the sum of the distances between all these similar points, called the Warp Path Distance, to

measure the similarity between two time series.

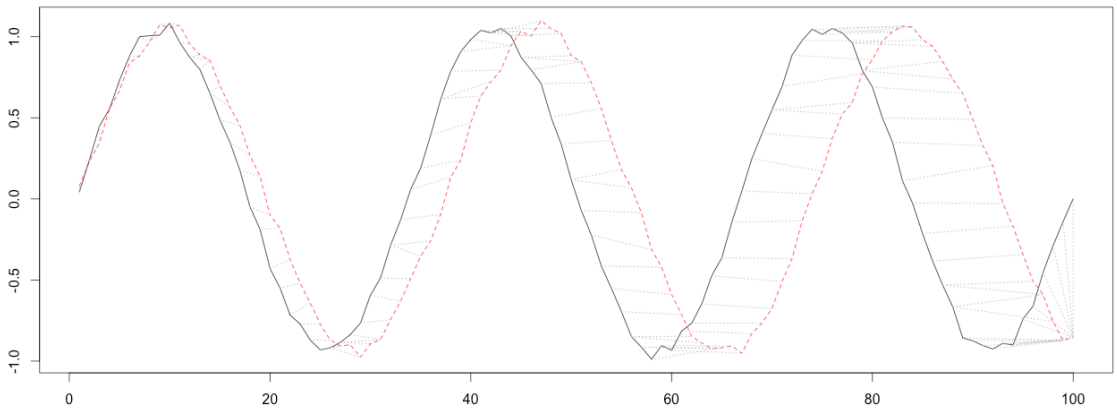

Figure 4.10: Alignment between the points of the sinusoids.

That is, in many cases, two time series may have similar shapes. However, their morphological feature points (peaks, troughs) may not be aligned one by one respect the x-axis (timeline). Nevertheless, if we permit temporal extension and contraction in the time of matching, the result will be significantly enhanced. So before we compute the similarity between them, we need to warp one of the two (or both) sequences in the timeline in order to achieve a better alignment. DTW is actually an effective way to accomplish this warping, it provides a matching method that allows the extension and contraction of the time series on the time axis.

4.5.1.1 Definition

There is a question: how do we know that the two time series are well aligned? In other words, what kind of warping is correct? Intuitively, we expect that after warping the sequences could coincide with each other, or the sum of the distances between all corresponding points in the two sequences is the smallest.

Suppose we have two time series X and Y whose length is m and n, respectively:

$$X = \{x_1, x_2, ..., x_m\}$$

$$Y = \{y_1, y_2, ..., y_n\}$$

In order to align these two sequences, DTW constructs a $m \cdot n$ matrix. The

elements of the matrix $m(i, j)$ represent the distance $d(x_i, y_j)$ between x_i and y_j (that is, the degree of similarity between each point of sequences X and Y, the smaller the distance is, the higher the similarity is), Commonly, Euclidean distance is taken, but it is also possible to take other measures. The core of DTW is to find a warping path that passes through the grids of the matrix, id est to solve the correspondences of the points of two sequences. We express the warping path as:

$$\phi(k) = (\phi_x(k), \phi_y(k)) \tag{4.10}$$

in which the value of $\phi_x(k)$ may be 1, 2, ..., m, the value of $\phi_y(k)$ may be 1, 2, ..., n, and k = 1, 2, ..., K, where $max(m, n) <= K <= m + n - 1$. For example, if $\phi(k) = (1, 1)$, it means that the first point of X corresponds to the first point of Y.

Given $\phi(k)$, we can solve the cumulative distance between 2 sequences as:

$$d_\phi(X, Y) = \sum_{k=1}^{K} d(\phi_x(k), \phi_y(k)) \tag{4.11}$$

The final output of DTW is the best-fit warping path which minimizes the cumulative distance:

$$DTW(X, Y) = min(d_\phi(X, Y)) \tag{4.12}$$

In other words, given the distance matrix, we have to find a path that comes from the upper left corner and goes to the bottom right corner, in which the sum of the value of the passed grids is the smallest. And the path must fulfil the following constraints:

1. Continuity:

$$\phi_x(k+1) - \phi_x(k) \leq 1 \quad and \quad \phi_y(k+1) - \phi_y(k) \leq 1 \tag{4.13}$$

That is, the path cannot skip points and must be continuous, ensuring that all points in the two sequences are matched.

2. Monotonicity:

$$\phi_x(k+1) \geq \phi_x(k) \quad and \quad \phi_y(k+1) \geq \phi_y(k) \tag{4.14}$$

In other words, the path cannot go backwards or upwards, otherwise there will be a meaningless cycle.

$$\phi_x(1) = \phi_y(1) = 1, \quad \phi_x(K) = m, \quad \phi_y(K) = n \tag{4.15}$$

3. Boundary Condition:. That it, the path must start at the upper left corner and end at the bottom right corner.

Combine continuity and monotonicity, there are only three possible directions for each grid to choose. For example, if the path has passed the grid (i, j), then the next grid can only be one of the following three options: $(i+1, j)$, $(i, j+1)$ or $(i+1, j+1)$.

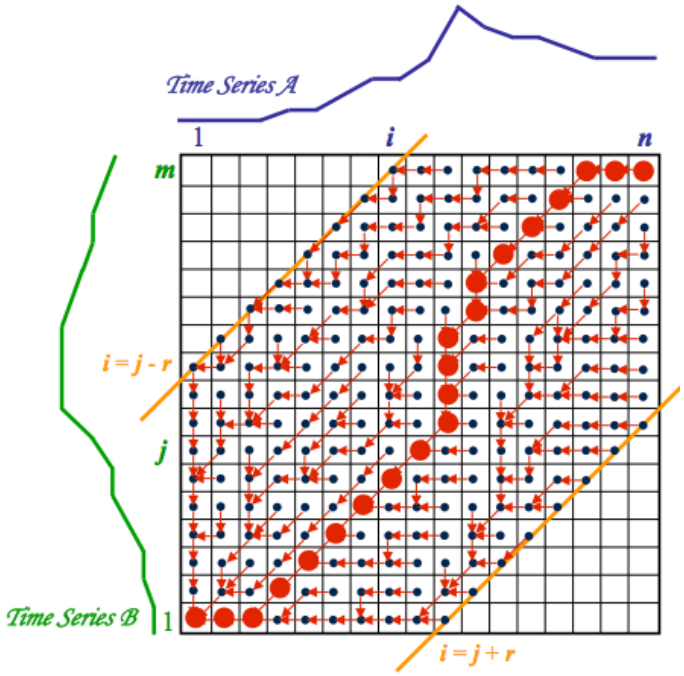

Figure 4.11: An example of DTW.

In the figure above, we have two similar sequences for comparison. We can see that there are several possible warping paths, in which the bold one has the least

cost.

Algorithm 4.4 that describes the procedure of DTW is quite intuitive, we first create an $m \cdot n$ matrix and set the first element (i.e. $m[1,1]$) of the matrix to the distance of the first element of each sequence. Then we initialize the first column starting from the second row, in each element we put the cumulative distance which is the sum of the value of the upper element and the respective distance ($d(X[i], Y[1])$). Then we initialize the first row starting from the second column, in each element we put the cumulative distance which is the sum of the value of the element in the left side and the respective distance ($d(X[1], Y[j])$). Then for each i from 2 to m, and for each j from 2 to n, we calculate the distance between the points x_i and y_j, and sum up the minimum of $m[i-1, j]$, $m[i, j-1]$ and $m[i-1, j-1]$. Afterwards, we return the last element of the matrix, that is, the distance between the two series.

Algorithm 4.4 Dynamic Time Warping

Function *DTW(X, Y)* **is**
    Data: X: array [1..m], Y: array [1..n]
    Result: Distance between X and Y applying DTW
1   $DTW \leftarrow array[1..m, 1..n]$
2   $DTW[1,1] \leftarrow d(X[1], Y[1])$
3   **for** $i \leftarrow 2$ **to** m **do**
4       $DTW[i,1] \leftarrow DTW[i-1,1] + d(X[i], Y[1])$
5   **end**
6   **for** $j \leftarrow 2$ **to** n **do**
7       $DTW[1,j] \leftarrow DTW[1,j-1] + d(X[1], Y[j])$
8   **end**
9   **for** $i \leftarrow 2$ **to** m **do**
10   **for** $j \leftarrow 2$ **to** n **do**
11          $cost \leftarrow d(X[i], Y[j])$
12          $DTW[i,j] \leftarrow cost + min(DTW[i-1,j], DTW[i,j-1], DTW[i-1,j-1])$
13   **end**
14 **end**
15 **return** $DTW[m,n]$
end

4.6 Clustering

4.6.1 Hierarchical clustering revisited

In the current project, manual Hierarchical clustering is used. The first step is the creation of the distance matrix based on an initial setting of clusters which consist of one element per cluster (lines 1-2 of the algorithm below). Then, dissimilarities are calculated at every merge (lines 3-6), in order to make a dissimilarities plot that helps the user to make a decision on the number of clusters.

Unfortunately, the automatic procedure for the choice of the number of clusters did not work properly as in the previous project. It gives either a high number of clusters or an tiny k which are discarded by the experts. Since we needed to keep on advancing on the newly introduced topics of this thesis, we have decided to stick to the manual Hierarchical clustering.

Algorithm 4.5 Manual Hierarchical clustering

Function *manual_HC(D, k)* **is**
    Data: D: dataset, k: number of clusters to create, default value is 1
    Result: k clusters with all the elements in D
1   Initialize the clusters where each of them contain 1 element of D
2   Compute the initial distance matrix
3   **while** *number of clusters* $> k$ **do**
4   Find two closest clusters and merge them
5   Update then distance matrix
6   Replace the old clusters with the merged one
7   **end**
8   **if** $k == 1$ **then**
9   Show the dissimilarities graph and set k to the value entered by user
10   Back to line 1 (i.e. rerun the procedure with the user-provided k)
11 **return** *k clusters*
end

4.6.2 Threshold computation

Before we proceed with the K-means algorithm, a user-defined threshold is required by the Wishart's variant as explained in section 2.4.2.1. The procedure to choose

an appropriate threshold comes from Kaufmann and Weber [7]. For each cluster, the distances from its centroid to all the elements of the cluster are calculated, and then the frequencies distribution of the distances are collected. The frequencies are then plotted so that the user can decide which threshold to use, normally a local minima will be chosen. The clusters obtained from HC and the distance threshold chosen will be the inputs for the incoming K-means clustering.

Algorithm 4.6 Threshold computation

```
Function threshold_computation(C) is
    Data: C: clusters obtained from HC
    Result: threshold chosen by user
1   for each cluster c do
2       Compute the centroid
3       for each element e of cluster c do
4           Measure the distance between e and centroid
5           Collect information on the distribution of the distances
6       end
7   end
8   Plot the distance distribution
9   return threshold chosen by user
end
```

In this project, we continue using Wishart's variant of K-means clustering to keep us aligned with the work of Kaufmann and Whiteman. The procedure is represented in algorithm 2.3, a slight difference is that the initial clusters are the result of the Hierarchical clustering in the previous steps, rather than randomly generated.

4.6.3 Centroid for time series clustering

For conventional clustering, the centroid of a cluster is computed as described in section 2.4.3, whether the data source is WRF Japan or Boat data. However, for Boat conventional clustering, a slight change is made. Since the number of available boats for each timestamp is different, the n in equation 2.11 should be changed correspondingly, depends on the available boats at a timestamp.

Unlike conventional clustering, sequence clustering poses some problems which

do not exist in Euclidean space. Specifically, the problem remains on how the cluster centroid (prototype) is calculated. For this kind of clustering, averaging the elements of a cluster like in classical K-means algorithm is no longer suitable due to the variance on the sequences' lengths. Given sequences in a cluster, it is clear that the cluster's centroid c minimizes its total distances towards the rest of the elements within the cluster, such that

$$E(S_j, c) = \sum_{s_i \in S_j} d_{DTW}(s_i, c) \tag{4.16}$$

is minimized. The sequence c that minimizes $E(S_j, c)$ is called a Steiner sequence [34].

Taking this concept into account, the most common way for the definition of cluster centroid for time series clustering is to use cluster medoid as the prototype, which is a real sequence of the cluster. It is defined in the following way:

$$c_j = \arg\min_{s_j \in S_i} \sum_{s_k \in S_i \setminus s_j} d_{DTW}(s_k, s_j) \tag{4.17}$$

Other common methods are:

- Optimal prototype: it computes the Steiner sequence using n-dimensional dynamic time warping, being n the number of sequences in the cluster. The optimal prototype is the time-series of length K, where each vector is the average of the original ones given by the warping path. However, the drawback of this approach is that the search space grows exponentially as a function of n, and it has been proven to be NP-complete finding the Steiner sequence in the discrete case.

- Averaging method: it combines two sequences using DTW, until only one time-series is left. Unfortunately, the order in which the pairing is performed affects the final prototype significantly. Abdulla et al. proposed *Cross-words reference template* [35] which is invariant to the order of processing sequences. It first computes the average length of the sequences is calculated. Afterwards, the sequence with the length nearest to the average length is chosen to be the

initial reference prototype. Next, the other sequences are aligned by the DTW process such that their lengths will be equal to the chosen initial prototype. Finally, the final time series will be created by averaging the time-aligned sequences across each frame.

In [36], a method called *prototype by local search* is proposed. This method starts from the medoid, then iterate between the mapping stage and averaging stage: *i*) compute averaged prototype based on warping paths and *ii*) calculate new warping paths to the averaged prototype.

Algorithm 4.7 Local search prototype

Function *LS(TS)* **:**
  Input: TS: the set of time series of a cluster
  Output: centroid of the cluster based on local search
1 $c_{old} \leftarrow$ Compute medoid of the cluster
2 **repeat**
3 $c_{old} \leftarrow c_{new}$ if not first iteration
4 Compute warping paths to c_{old}
5 $c_{new} \leftarrow$ new averaged time series using paths
6 **until** $E(S, c_{new}) \geq E(S, c_{old})$;
end

We have adopted prototype by local search as our centroid computation algorithm. Thus, for each of the clusters, the medoid will be firstly computed by applying equation 4.17 (line 1). Note that each element contains a time series composed by u, v components, as can be seen separately in figures 4.7 and 4.8. Then, we iteratively compute the new averaged prototype until convergence (line 2 - 6). In the end, the centroid will have a sequence whose length is as same as the length of the sequence of the medoid. Thus, we can assign the timestamps of the medoid to the centroid. This allows us to have a tracking of the representative hours of a cluster, which is especially useful for the generation of statistics report where the evolution of weather data among time will be represented.

4.7 Clustering comparison

Since this project works with various kind of input data (WRF/Boat data) and clustering configurations (conventional/time series clustering), it would be helpful for the meteorologists if they can have quantitative measurements to have a first impression of the equivalences in two clustering results. Based on these measurements, further delicate comparisons can be carried out, together with the generated statistic reports.

Since there is no "gold standard" that defines what are the correct clusters for our data, we had to come up with other ways to evaluate our clustering.

On the one hand, the criterion of the meteorologists using their previous knowledge about the area plays a vital role in the comparison.

On the other hand, Maximum-Match-Measure (explained in section 2.7.1) is adopted as the previous project did. Another simple measure we used to quantify the quality of the time series clustering is counting the percentage of available boats of a day that are in the same cluster. Despite the fact that it is a straightforward method, actually, it allows us to have immediate insight on the quality of the time series clusterings.

The are several comparisons that we could perform, for example:

- Boat conventional - WRF conventional: it allows us to verify if there is a significant difference between the predicted data and real data. In this case, since we use a period of half an hour for compacting Boat conventional data, naturally, the hourly wind fields of WRF conventional data could match 2 elements of Boat data. For instance, say we have 2 elements in cluster 1 of clustering $\mathcal{C}_{Boat}$ which correspond to the data of timestamps 10:00 and 10:30 of the day d, and we have hourly wind field of 10h of day d in cluster 1 of clustering $\mathcal{C}_{WRF}$. Then we say that the number of matched elements of the two clusters is 2, since when searching for the matching, only the hours are taken into account.

- Boat conventional - Boat time series: it can identify the correspondence of time series clustering with the previously analysed conventional clustering by using

Maximum-Match-Measure, and then manually analysing if it makes sense. For this specific situation, some adaptions are needed. Since the boats of a specific day may be assigned to a different class, first we have to determine which is the cluster where most of the boats are assigned to. Then, when searching for the matching, only the dates are taken into account. For example, say we have again 2 elements in cluster 1 of clustering $\mathcal{C}_{BoatConv}$ which correspond to the data of day d, and cluster 1 of clustering $\mathcal{C}_{BoatSeries}$ is the cluster to which most of the boats of the day d belong. In this case, we say that the number of matched elements is 2.

Notice that although these measures are very useful for the analysis, however, they depend on the quality of data that we have which can affect the clustering result significantly, especially for Boat data.

4.8 Generation of report

After the clustering procedure is done, based on the results of clustering, statistics report is generated. This report is what we considered the phase of "prototyping": in the previous work, a simple report had been defined. Now, this report has been gradually enriched with the comments and needs of the meteorologists, so it contains relevant info (both global and detailed) to define the contents and behaviour (including transitions) of each cluster. The meteorologists need to find the explanation of certain phenomena, and we will try to provide it by using correlation information. Therefore, the report is dense, but we have to consider that the meteorologists will go directly to look up the relevant information for their interest. Note that even if we are going to enumerate the different components of the reports here, we think their motivation will be more clear when they are explained in the results/discussions of chapter 5.

First of all, in the report, there will be basic information regarding the chosen filters for the input data. For instance, the selected range of days and hours, etc. Also, the execution time of the procedures like normalization and clustering algorithms will be reported. The dissimilarities plot, the user-defined k and threshold,

as well as the threshold plot, will be displayed.

The meteorological parameters that have been taken into account for the analysis and thus, the computation of maximum/minimum/average values are:

- Wind speed and direction.
- Temperature at 2 meters and at the ground, mean sea level pressure.
- Relative humidity and total precipitation.
- Low/high/medium/total cloud coverage.
- Wind gust and land cover surface.

The following content of the report consists of a series of tables, plots and descriptive values which differs from one clustering type to another. To make it clear, they will be presented individually in the following sections.

4.8.1 Statistics: conventional clustering

On the one hand, for conventional clustering, a series of tables that contains information regarding the composition of the clusters and statistic values of the meteorological values are shown. The information includes a table that contains the number of elements and the relative frequency, and tables with the maximum and minimum values of each meteorological parameter (pressure, humidity), as well as their averages. With this information meteorologists could have an immediate overall insight regarding the clusters. An example can be found in figures 4.12 and 4.13, note that the cluster whose name involves asterisks represents the set of outliers.

Particular attention should be paid when computing the means of wind directions since it should be computed using the mean of circular quantities [37], rather than the simple arithmetic mean of all the directions. For example, the arithmetic average of directions of 355° and 15° would be 185°, but actually, it should be 5°.

Cluster	N° elements	Relative freq
1	113	34.24
2	60	18.18
3	52	15.76
4	39	11.82
5	29	8.79
6	17	5.15
7	3	0.91
8	2	0.61
9	15	4.55

Figure 4.12: Table representing the composition of the clusters

Cluster	Min speed	Avg speed	Max speed	Min direction	Avg direction	Max direction
1	0.421	18.107	31.257	3.137	210.168	356.169
2	1.344	12.786	27.416	52.532	191.550	358.424
3	0.559	9.537	15.510	1.295	160.408	248.642
4	0.929	13.000	26.299	0.138	55.293	359.215
5	0.233	10.375	22.292	0.005	20.470	359.957
6	0.323	9.254	18.623	0.941	132.069	355.917
7	0.213	3.084	8.898	1.783	228.310	359.693
8	0.651	11.287	23.410	2.232	323.192	353.520
9	0.076	5.069	13.837	0.250	126.784	359.391

Figure 4.13: Table representing the min/average/max value of TWS/TWD

Subsequently, there are some tables that represent frequencies of ranges of weather parameters (precipitation, temperatures, etc.): the minimum and maximum values of the complete dataset are taken, and the interval is divided into 5 ranges, except for the TWD which is split into 16 ranges. For each of the clusters, the percentage of each range is shown. These tables were useful for the meteorologist to analyze the clustering and identify the underlying wind pattern, especially by checking wind direction and speed. Figure 4.14 shows an example of the ranges of temperatures. The reason of adding these frequencies of ranges of values is: even if we have the min/max/average values, it is important to know the detailed distribution of each of the weather parameters.

Table 1: Ranges of TEMPERATURE (in °C):

Cluster	[20.178, 23.253)	[23.253, 26.328)	[26.328, 29.404)	[29.404, 32.479)	[32.479, 35.554]
1	0.00%	7.61%	83.16%	8.49%	0.74%
2	0.00%	9.30%	83.40%	6.38%	0.92%
3	0.00%	7.23%	84.21%	7.90%	0.65%
4	2.77%	69.59%	26.67%	0.97%	0.00%
5	28.31%	68.97%	2.72%	0.00%	0.00%
6	0.35%	25.76%	69.59%	4.29%	0.00%
7	0.00%	6.33%	75.33%	17.33%	1.00%
8	0.00%	0.00%	80.00%	17.50%	2.50%
9	0.87%	34.40%	53.53%	9.87%	1.33%

Figure 4.14: Ranges of temperatures

In conventional clustering with fixed stations/ribs, our methodology generates a representation of the behaviour of the local wind in each position in each cluster. Using a tool provided by the meteorologists, we can generate plots like the figure below, from which different behaviours in different locations are observed (the colours of the arrows express the intensity of the wind in knots). This is a massive help for visualizing the behaviours within each cluster.

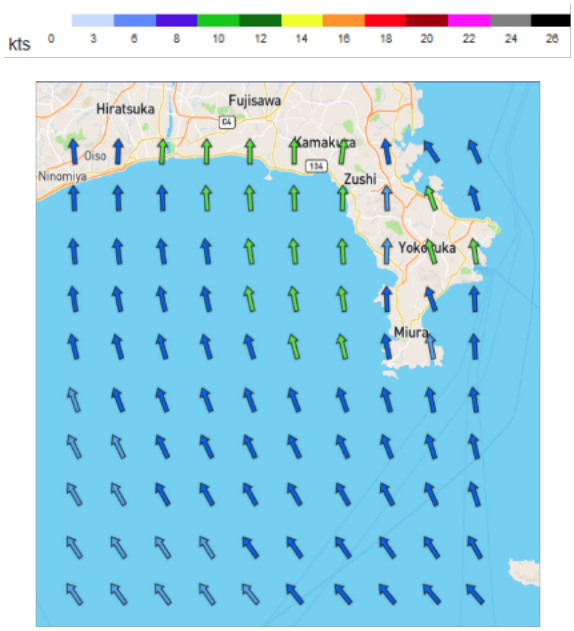

Figure 4.15: Example of average wind for the 100 stations of WRF model, for a specific cluster

Afterwards, more information is provided by the transition matrix which shows how the transitions take place between clusters. The reason for introducing this kind of information is that it gives an idea about the dynamic behaviour of the patterns: whether one pattern is quite stationary (once the day starts in this pattern, we

can forecast that it is probably going to stay) or if it evolves during the day going through different patterns.

To do so, given an element of a cluster, it checks to which cluster the following timestamp (i.e. the element whose date-time if one hour after) belongs to. The transitions are expressed in percentages and the representation of them is a square matrix M with as many rows and columns as the number of clusters. Each cell M_{ab} represents the percentage of elements that change from cluster a to cluster b.

Furthermore, relative frequencies of occurrence of hours that the hourly wind fields correspond to, are represented by a bar plot for each of the clusters. These plots allow us to have an immediate insight regarding the time distribution of the winds and hence detect potential day/afternoon wind pattern.

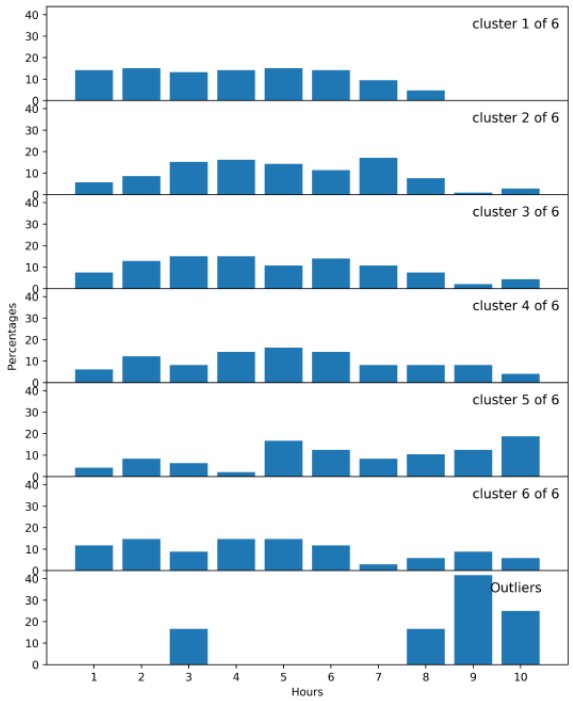

Figure 4.16: Occurrence of hours for each of the clusters

In general, one can assume that the most probable behaviour is that one pattern is going to remain during a day. Then, whenever a pattern changes, it is very relevant to study what weather parameters might motivate this transition. That is why, for these cases, we add more detailed information including how the weather parameters change during these transitions (tables of deltas which show the differences of

parameters between the clusters, statistics values, etc.).

But the meteorologists found that for some cases, they needed to get into even more detail and study each of the transitions according to the what had been the change in wind direction, which is considered the most critical parameter. Therefore, for each transition and possible shift in wind direction (using the 16 ranges in which we have divided the 360° wind direction), we study the average deltas in the weather parameters that may have caused this change.

In order to provide this information, for each meteorological parameter, frequency distribution of ranges of differences on these parameters (i.e. value of h_i - value of h_{i+1}) will be shown in a table for each transition we have (e.g. cluster 1 - cluster 1, cluster 1 - cluster 2, etc.), as can be seen in the figure below.

Table 15: Ranges of DIFF. TEMPERATURE (in °C):

Transition	[-0.666, -0.390)	[-0.390, -0.115)	[-0.115, 0.161)	[0.161, 0.437)	[0.437, 0.713]
1-1	0.00%	30.77%	58.24%	9.89%	1.10%
1-2	0.00%	42.86%	42.86%	14.29%	0.00%
1-8	0.00%	100.00%	0.00%	0.00%	0.00%
2-1	0.00%	0.00%	77.78%	11.11%	11.11%
2-2	0.00%	11.11%	75.56%	11.11%	2.22%
2-3	0.00%	66.67%	33.33%	0.00%	0.00%
3-2	0.00%	0.00%	100.00%	0.00%	0.00%
3-3	0.00%	39.53%	58.14%	2.33%	0.00%
3-6	0.00%	100.00%	0.00%	0.00%	0.00%
4-4	16.13%	22.58%	25.81%	25.81%	9.68%
4-6	0.00%	0.00%	0.00%	100.00%	0.00%
5-4	0.00%	0.00%	66.67%	0.00%	33.33%
5-5	0.00%	4.17%	29.17%	41.67%	25.00%
6-3	0.00%	0.00%	66.67%	33.33%	0.00%
6-6	0.00%	45.45%	45.45%	9.09%	0.00%
7-1	0.00%	0.00%	100.00%	0.00%	0.00%
7-7	0.00%	0.00%	100.00%	0.00%	0.00%

Figure 4.17: Ranges of difference of temperatures for the transitions

After that, for each possible transition, a table that represents the changes of TWD from one hour to the next, is shown. Each row and column of the table corresponds to a range of TWD (360° split in 16 ranges). For example, in the figure below, the value of cell corresponds to the row that represents the range [202.5-225) degrees and the column which represents the range [180-202.5) means that: for 100% of the elements of transition from cluster 1 to cluster 2 that had a TWD range of [202.5-225) degrees at hour h_i, their value of TWD had decreased slightly to a range of range [180-202.5) degrees in the in the next hour h_{i+1}.

Table 28: Transition 1-2: Changes of TWD

TWD Range	[0-22.5)	[22.5-45)	[45-67.5)	[67.5-90)	[90-112.5)	[112.5-135)	[135-157.5)	[157.5-180)	[180-202.5)	[202.5-225)	[225-247.5)	[247.5-270)	[270-292.5)	[292.5-315)	[315-337.5)	[337.5-360]
[0-22.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[22.5-45)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[45-67.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[67.5-90)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[90-112.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[112.5-135)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[135-157.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[157.5-180)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[180-202.5)	0%	0%	0%	0%	0%	0%	0%	12.50 (1)	87.50 (7)	0%	0%	0%	0%	0%	0%	0%
[202.5-225)	0%	0%	0%	0%	0%	0%	0%	0%	100.00 (16)	0%	0%	0%	0%	0%	0%	0%
[225-247.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[247.5-270)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[270-292.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[292.5-315)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[315-337.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[337.5-360]	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

Figure 4.18: An example of table that represents the changes of TWD

Then, for each of the non-zero TWD range transition, average differences on the meteorological parameters are computed. In a similar way, the overall average difference for each of the available meteorological parameters is calculated. An example of this kind of information is shown in the figure below.

Combining all the newly added information allow the meteorologists to analyze and understand the reason for the transitions.

Transition 6-6 ([135-157.5) - [135-157.5)): Range level average differences
diff_temp: 0.0003056666666666115
diff_humidity: -0.11393333333333344
diff_precipitation: 0.08333333333333331
diff_pressure: 0.14880716959635418
diff_gust: 0.34084876302083345
diff_high_cloud_coverage: 2.8214666666666663
diff_low_cloud_coverage: -0.27086666666666664
diff_land_cover_surface: 0.0
diff_medium_cloud_coverage: 0.011016666666666675
diff_temperature_at_ground: -0.00438000000000011
diff_total_cloud_coverage: 2.3500666666666663
diff_tws: 0.5223127379600102
diff_twd: 2.995442175789479

Transition 6-6: Overall level average differences
diff_temp: -0.05655972727272726
diff_humidity: -0.08587272727272727
diff_precipitation: 0.022919090909090913
diff_pressure: 0.02302764892578125
diff_gust: 0.28985142957513993
diff_high_cloud_coverage: -3.0632272727272736
diff_low_cloud_coverage: 0.19001636363636362
diff_land_cover_surface: 0.0
diff_medium_cloud_coverage: 0.003004545454545457
diff_temperature_at_ground: -0.11448245960582391
diff_total_cloud_coverage: -2.8769436363636367
diff_tws: 0.3115383140824422
diff_twd: 2.0445301368745095

Figure 4.19: Range/overall level average differences

4.8.2 Statistics: time series clustering

On the other hand, for time series clustering, the percentage of ribs of the same day in the same cluster is shown at the beginning (as mentioned, this is one of our quality criteria).

After that, a table is shown indicating the number of elements in each cluster and their relative frequency. The tables with the information regarding the min/max/average values of each meteorological parameter are also presented, as for conventional clustering. However, the computation is done more delicately.

Since for this kind of clustering, the clusters are composed by elements with sequences and timestamp info (i.e. to which timestamp a frame of the sequence corresponds, after applying PLR), the centroids of each cluster are computed firstly. Note that these centroids will have the same timestamps as the timestamps of the medoids, as mentioned in section 4.6.3.

As we are using the hourly meteorological parameters coming from WRF model, we have to use the hours derived from the centroid's timestamp, and also the available dates of the cluster by counting the dates of the ribs within the cluster. Then for each of the clusters, we follow the procedure summarized in the algorithm below. First, we initialize some dictionaries (line 1) in order to store the min/max/average value for the meteorological parameters. Then for each hour, we iterate over the dates of the cluster to collect values of weather data and, in the meantime, check if they correspond to a minimum/maximum (from line 2 to 13). Afterwards, we compute the hourly average value for the weather parameters (lines 14-16). Finally, the daily averaged value will be computed (lines 18-20).

Algorithm 4.8 Statistics for time series clustering

Function *gather_cluster_statistics(hours, cluster_dates)* **:**
Input: hours: hours associated to the timestamps of the centroid, cluster_dates: the dates of the ribs in the cluster)
Output: segment representation of T

```
1  Initialize dictionaries for storing min/max/average of each parameter
2  for hour in hours do
3      auxiliary_dict ← {}
4      for date in cluster_dates do
5          weather_parameters ← get_WRF_weather_params(date, hour)
6          for param_name, value in weather_parameters do
7              auxiliary_dict[param_name].append(value)
8              if value < min[param_name] then
9                  min[param_name] ← value
10             if value max[param_name] then
11                 max[param_name] ← value
12         end
13     end
14     for param_name, values in auxiliary_dict do
15         averages_dict[param_name].append(mean(values))
16     end
17 end
18 for param_name, values in averages_dict do
19     averages_dict[param_name] = mean(values)
20 end
end
```

For the statistics of TWD/TWS, we use the sequences of the centroid since it is the averaged prototype of a cluster, thus, it should be representative. We have decided to compact the u, v components within a period of 30 minutes since with a smoother gap, the changes of wind could be reflected in a better way. With the compact u, v components, we first calculate the mean values of the components for each period. Then we convert them to TWS and TWD. Afterwards, minimum/-maximum/average TWD/TWS can be easily derived.

After the statistics table, for each of the clusters, the composition is shown by displaying the date and boat id of each element of the cluster. And for each of the

meteorological parameters, the evolution of the parameter of cluster's centroid is plotted, as can be seen in the following figure.

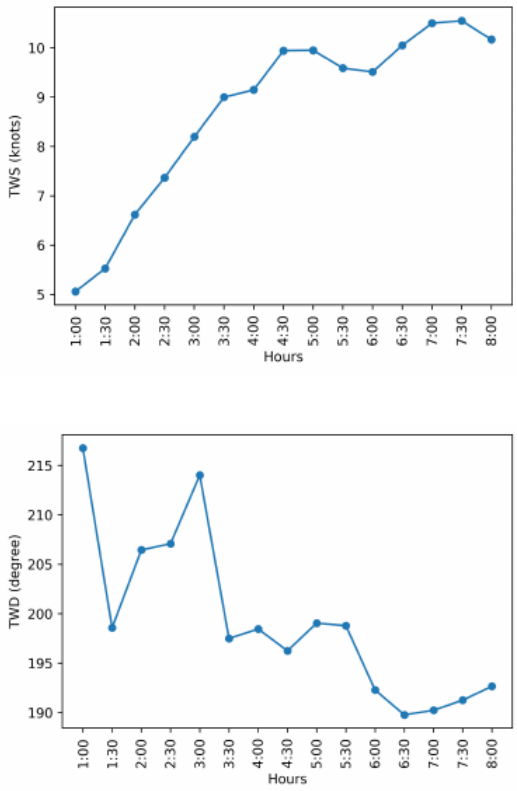

Figure 4.20: Evolution of TWS and TWD through the day

All the mentioned information (except the plots generated by a specific application) are gathered in an auto-generated PDF file that will be analyzed by experts to validate the effectiveness of the clustering results.

Chapter 5

Results Analysis and Discussion

As already briefly mentioned in section 1.1, a wind pattern useful for decision making within sailing is mainly identified through the following methodology:

1. Splitting the wind directions into several sectors (the number and size of each sector depend on the geography of the bay and the climatic wind flows).

2. Identifying a characteristic behaviour of the wind speed within each sector in relationship with the time evolution (i.e. is the wind speed gradually increasing or decreasing through the day?).

3. Identifying a characteristic behaviour of the wind in relationship with additional weather variables (i.e. if the air temperature is increasing the wind speed is increasing and the wind direction is switching to another direction sectors).

Usually, this work is done from a very theoretical point of view just by watching at the geography of the bay and climatic values. Once some first measurements are recorded on site, the theoretical patterns are validated and refined. The higher the number of collected data and the experience on the specific place, the more precise the patterns are.

The idea is therefore to sort out several clusters by using the combined values of wind speed and direction and to compare these clusters with additional weather variables to understand if the clusters are relevant for the place and the sailors.

Moreover, one fundamental analysis is to analyse the transition from one cluster to another and the variation of different weather parameters during this transition. This information can be a key to predict the potential evolution from one wind direction to another by watching the evolution of additional parameters such as air temperature, cloud coverage or atmospheric pressure.

In the next sections, we provide the analyses derived from the statistics reports, for different data source and type of clustering. The complete auto-generated statistics reports are included in the Annexes. Since each of them is indeed a large PDF file, it's inconvenient to cover the entire content in this report. For simplicity, only the most relevant part of the analysis are highlighted and presented.

These analyses have been performed by the author of this thesis together with an expert (a meteorologist) who has also previous knowledge of the Olympic training area. It does not pretend to be an exhaustive analysis (this would be completely out of the scope of this thesis), but to show how the different results provided by our methodology can help drawing conclusions and making decisions.

5.1 WRF Japan

5.1.1 First attempt

As an initial attempt, the first clustering has been performed on the WRF data, using 100 points distributed in the Bay. The area covered by these points is much bigger than the area where Olympic racing will take place. However, the decision was made to have a bigger overview of weather conditions that might generate specific patterns within the racing area but coming from far away. This is particularly important, in a place like Japan, where Tropical Cyclones coming from far away significantly affect wind conditions at a local scale.

WRF data were covering the whole period from 21 July 2018 to 15 September 2018 (including days where no rib data are present), as a preliminary analysis of the period in which the Olympics will take place. The chosen hours start from 10:00 a.m. to 7:00 p.m. in Japan time, which correspond to the usual period for training and competition.

Four clusters have been used during this first test, using the hierarchical dissimilarities graph as a guide, as well as the homogeneity and distribution of the resulting clusters.

Cluster	N° elements	Relative freq	Min temp	Avg temp	Max temp	Min humid.	Avg humid.	Max humid.
1	197	51.84	23.81	27.91	35.91	31.43	84.52	100.00
2	90	23.68	20.18	25.88	32.24	39.64	76.76	99.56
3	79	20.79	21.86	27.93	34.83	43.35	84.19	100.00
4	11	2.89	25.47	28.15	33.47	49.81	78.74	93.89
5	3	0.79	24.48	27.65	33.58	40.56	76.39	94.61

Table 5.1: First attempt - Statistics (1)

Cluster	Min speed	Avg speed	Max speed	Min direction	Avg direction	Max direction
1	0.253	15.663	31.257	3.137	203.955	358.424
2	0.076	17.794	59.977	0.005	32.435	359.960
3	0.383	9.700	18.623	0.941	158.175	355.917
4	0.091	4.745	13.837	0.380	122.922	359.693
5	0.651	9.003	23.410	2.232	341.513	359.391

Table 5.2: First attempt - Statistics (2)

Cluster	Min precipit	Avg precipit	Max Precipit	Min pressure	Avg pressure	Max pressure
1	0.000	0.160	29.400	1000.042	1008.805	1016.173
2	0.000	4.594	61.310	979.359	1010.017	1021.891
3	0.000	1.393	44.060	1002.709	1008.507	1020.462
4	0.000	0.055	10.000	1005.958	1009.799	1013.442
5	0.000	0.045	4.074	1004.766	1012.277	1019.951

Table 5.3: First attempt - Statistics (3)

By watching at the tables 5.1, 5.2 and 5.3, the following results are derived. Four main directions are identified:

1. 200 meaning SSW wind.

2. 30 meaning NE wind.

3. 160 meaning SE wind.

4. 120 meaning again SE wind.

Since clusters 3 and 4 have similar direction, it is important to see what the difference in terms of additional weather variables is.

Concerning the wind speed, cluster 3 seems to have an average speed higher than cluster 4. However, if we looking at the frequency table 5.1, we can notice that cluster 4 has only 2.89% of occurrence, so a very low number of cases.

By watching at the wind speed we can also notice that clusters 1 and 2 having SSW and NE wind, are the strongest, with maximum speed going up to 31 and 59 knots (kts). This is particularly true since the Tropical Cyclones are mainly generating SSW-SW flows becoming NE.

After this first test, it was clear that strong wind days were a limited number of cases but affecting the clusters' characteristics. Considering that normally no sailing is performed with wind speed higher than 25 kts, it has been decided to conduct a second test by using the 100 WRF points but considering only sailable days and in particular, days where measurements from Austrian ribs and/or from British ribs had been collected.

Moreover, since the variability of direction was not detailed enough compared with the behaviour of the wind observed during trainings, it has been decided to increase the number of clusters. All these results are described in the next section.

5.1.2 Clustering with filters

(The report used for this analysis corresponds to file WRF100-k8-th0.6.pdf in the Annexes).

Regarding the sailable days, we have first discarded 28 July 2018, in which the wind speed is extremely high (up to 80 kts). And the following days for which we have not enough boat data, are discarded as well:

- 21-24 July 2018.
- 7, 8, 10, 11 August 2018.
- 4-5 September 2018.

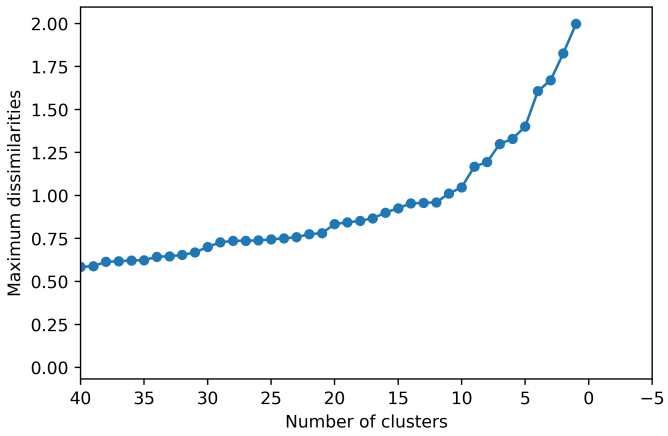

Figure 5.1: Filtered WRF - Dissimilarities plot

According to the above dissimilarities plot, the preferred choices of the number of clusters (denoted as k) are 5, 8 and 10, since they are followed by leaps. Out of these clear possibilities, we have considered having 8 clusters since, in the previous analysis, it was not clear enough to determine the behaviours of wind with only 4 clusters.

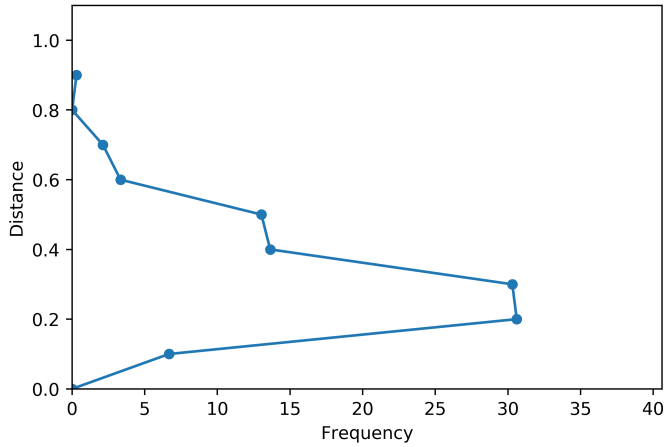

Figure 5.2: Filtered WRF - Thresholds plot

Then, with the help of the thresholds plot, we decided to use a value of 0.6, as most of the distances are less or equal than this value.

However, when we observed from table 5.4 that clusters 7 and 8 were so irrelevant (we can see that they contain only 3 and 2 elements), we decided to use only the first 6 clusters.

Cluster	N° elements	Relative freq	Min temp	Avg temp	Max temp	Min humid.	Avg humid.	Max humid.
1	113	34.24	24.48	27.83	35.55	40.71	83.20	99.96
2	60	18.18	23.81	27.56	35.23	43.38	86.90	100.00
3	52	15.76	24.11	27.85	34.76	44.87	85.69	100.00
4	39	11.82	20.94	25.31	31.12	39.64	69.53	95.24
5	29	8.79	20.18	24.09	27.80	47.38	77.04	99.56
6	17	5.15	21.86	27.03	31.98	46.61	78.62	98.78
7	3	0.91	26.10	28.16	33.01	49.81	79.16	93.88
8	2	0.61	26.50	28.84	33.58	40.56	80.12	94.61
9	15	4.55	20.39	27.17	34.35	44.27	77.40	95.59

Table 5.4: Filtered WRF - Statistics (1)

As explained above, the first parameter to analyse is the wind direction. We can deduce from table 5.5 that the average wind direction is now distributed as follows:

- Cluster 1: 210 – SSW.
- Cluster 2: 190 – S.
- Cluster 3: 160 – SSE.
- Cluster 4: 050 – NE.
- Cluster 5: 020 – NNE.
- Cluster 6: 130 – SE.

Cluster	Min speed	Avg speed	Max speed	Min direction	Avg direction	Max direction
1	0.421	18.107	31.257	3.137	210.168	356.169
2	1.344	12.786	27.416	52.532	191.550	358.424
3	0.559	9.537	15.510	1.295	160.408	248.642
4	0.929	13.000	26.299	0.138	55.293	359.215
5	0.233	10.375	22.292	0.005	20.470	359.957
6	0.323	9.254	18.623	0.941	132.069	355.917

Table 5.5: Filtered WRF - Statistics (2)

Since we considered only the sailable days and the dates for which we have enough data measured with ribs' sensors, now the strongest wind speed result to be about 31 kts.

Cluster 1, with SSW wind, results to be the strongest one, with an average speed around 18 kts. Actually, this direction represents the most likely wind associated with the passage of Tropical Cyclones.

Moreover, a direction about 220°, is considered as the direction of the so-called 'fully developed sea breeze'. The sea breeze is the wind enhanced by thermal effect. This wind often starts from a different direction than 220°, and increases going towards 220°. Therefore having the 210° as the higher speed cluster makes particular sense.

Considering cluster 2, having an average direction around 190 degrees, could represent either a southerly gradient with a moderate to strong speed associated with frontal systems coming from the South-West, or a light southerly gradient that could then develop in a stronger sea breeze from SW.

Clusters 4 and 5 represent winds from NE to NNE. By observing the wind speed average ranges it is clear that the NE winds seem in average stronger than the NNE ones.

Clusters 3 and 6 represent winds from SSE to SE sectors. By watching the wind speed average ranges it is hard to notice any significant difference between the two patterns. We can remark, using the information about distribution of the patterns during the day (figure 5.3, the hours are in UTC), that cluster 5 corresponds mainly to the first hours of the day, cluster 4 to the first and last hours of the day and cluster 3 to the middle and last hours of the day. This is particularly useful information because actually the NNE-NE winds, mainly represent winds not affected by any thermal effect. While the SSE winds, so cluster 3, appears typically after 11:00 a.m. or 12:00 a.m. The distribution of cluster 6 reinforces this theory, having a higher probability during the last hours of the day. The medium-high frequency of cluster 6 during the first hours of the day might correspond to some early morning south-easterly winds, that normally die and go to South or South-West.

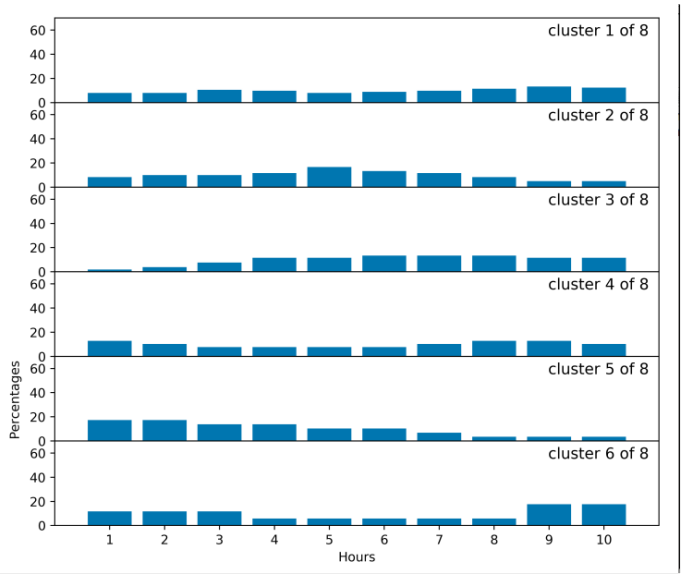

Figure 5.3: Filtered WRF - Occurrence of hours

Another interesting parameter to look at is the cloud coverage. In a country like Japan, where the humidity of the air is often very high, the cloud coverage can play a significant role in identifying differences from cluster to cluster.

Cluster	Min TG	Avg TG	Max TG	Min TCC	Avg TCC	Max TCC
1	24.193	27.853	53.256	0.000	11.235	98.500
2	23.397	28.379	52.371	0.000	7.362	95.700
3	23.802	28.869	52.239	0.000	9.350	100.000
4	19.088	26.788	45.282	0.000	38.690	100.000
5	20.184	26.103	41.782	0.250	51.680	100.000
6	21.286	28.125	45.793	0.000	13.513	92.100

Table 5.6: Filtered WRF - Statistics (3)

No cluster has a minimum total cloud coverage different from 0% (except cluster 5 with a tiny percentage), and every cluster has a maximum cloud coverage very close to 100%. The only information that we can clearly derive is that cluster 5, so winds from NE, have the higher average cloud coverage. This information, being derived from a weather prediction model, could not represent the reality but just an interpretation from the model.

In general, considering the tables representing the average ranges of the additional parameters such as precipitation, air pressure, humidity, etc., is not particularly significant to identify additional characteristics that can help in recognizing specific patterns. That's why it is very useful to look at the second set of tables: the ranges of each variable.

Cluster	[20.178, 23.253)	[23.253, 26.328)	[26.328, 29.404)	[29.404, 32.479)	[32.479, 35.554]
1	0.00%	7.61%	83.16%	8.49%	0.74%
2	0.00%	9.30%	83.40%	6.38%	0.92%
3	0.00%	7.23%	84.21%	7.90%	0.65%
4	2.77%	69.59%	26.67%	0.97%	0.00%
5	28.31%	68.97%	2.72%	0.00%	0.00%
6	0.35%	25.76%	69.59%	4.29%	0.00%

Table 5.7: Filtered WRF - Ranges of TEMPERATURE (in °C)

For instance, by watching the above table of ranges of temperature, it is immediately clear that cluster 5, so the winds closer to the North direction, are the ones having the lowest air temperature. That's why one conclusion that one can derive is that any increase in temperature would be associated with a shift of the direction to another cluster more 'right' than the cluster 5 (more 'right' meaning rotating in a clockwise direction).

On the other hand, cluster 4, the NE winds so a bit more right than cluster 5, result to be the one with a lower level of humidity. Therefore, going on from the previous reasoning, one can assume that an increase in temperature and a decrease in humidity would lead to a transition from cluster 5 to cluster 4. This is particularly useful in sailing, as explained above, because thanks to evidence-based signs such as an increase in temperature or a decrease in humidity we are able to predict the evolution of the wind.

Cluster	[39.640, 51.712)	[51.712, 63.784)	[63.784, 75.856)	[75.856, 87.928)	[87.928, 100.000]
1	0.31%	2.39%	10.68%	58.70%	27.92%
2	0.30%	2.30%	5.47%	38.25%	53.68%
3	0.40%	2.63%	6.67%	42.63%	47.65%
4	2.49%	24.74%	48.97%	22.95%	0.85%
5	0.41%	6.48%	40.10%	39.28%	13.72%
6	0.82%	4.88%	42.76%	27.41%	24.12%

Table 5.8: Filtered WRF - Ranges of HUMIDITY (in %)

Another interesting feature observed during the trainings in Japan is the transition from the NE winds to the SE ones. Cluster 3, so the SSE winds, are characterised by higher ranges of air temperature and higher ranges of humidity as well as by lower values of cloud coverage, compared with clusters 4 and 5. In reality the transition from NE to SE happens when the NE gradient effect weakens and some thermal component tries to fill in. This would result, as shown by the model, in a decrease of cloud coverage, an increase of humidity and an increase in temperature. So these results, confirm even more the ones derived from the distribution of patterns analysed above. The same happens if one compares cluster 4 with cluster 6. So the conclusion is that an increase in temperature and in humidity and a decrease in cloud coverage would result in a transition from cluster 4 to cluster 3 or 6.

As mentioned in the previous chapter, an important option of our methodology is the generation of the vector averages of all hourly winds at each specific location for each cluster. Figure 5.4 shows graphically these data. Arrows indicate wind direction and the colours indicate wind intensity in knots. This representation helps in immediately identifying relevant features of the wind related to the geographical distribution around the bay.

While in some clusters (2) the wind behaves homogeneously in the different locations, all the others show interesting geographic features as to wind intensity and/or direction.

For example, clusters 1, 2, 3 and 5 are examples of differences in wind speed: wind from the land, as cluster 5, causes that the areas closer to the land present far less speed. The fact that wind direction is almost perpendicular to the coastline produces

a wind which is more unsteady in pressure. This is a widely known phenomenon, especially if the coast is high. But it is not so obvious the fact that for certain patterns of wind from the sea (cluster 1), we can expect the wind to be stronger offshore, while for others (clusters 2 and 3), we can expect the wind to be stronger in certain areas closer to the land. So the graphical representation with a map, supports theory with evidence, enhancing the level of confidence that a sailor can have in making a strategical decision during an Olympic race.

On the other hand, clusters 3 and 6 are examples of what we call "bending along the coast": in certain patterns, the geography affects enormously the wind direction, which can be quite different according to the specific location. Finally, we observe in cluster 4 that the wind manages to become close to the parallel to the northern coastline and manages to accelerate better than cluster 5. This confirms our impression that with NE winds, the pressure is better in the left.

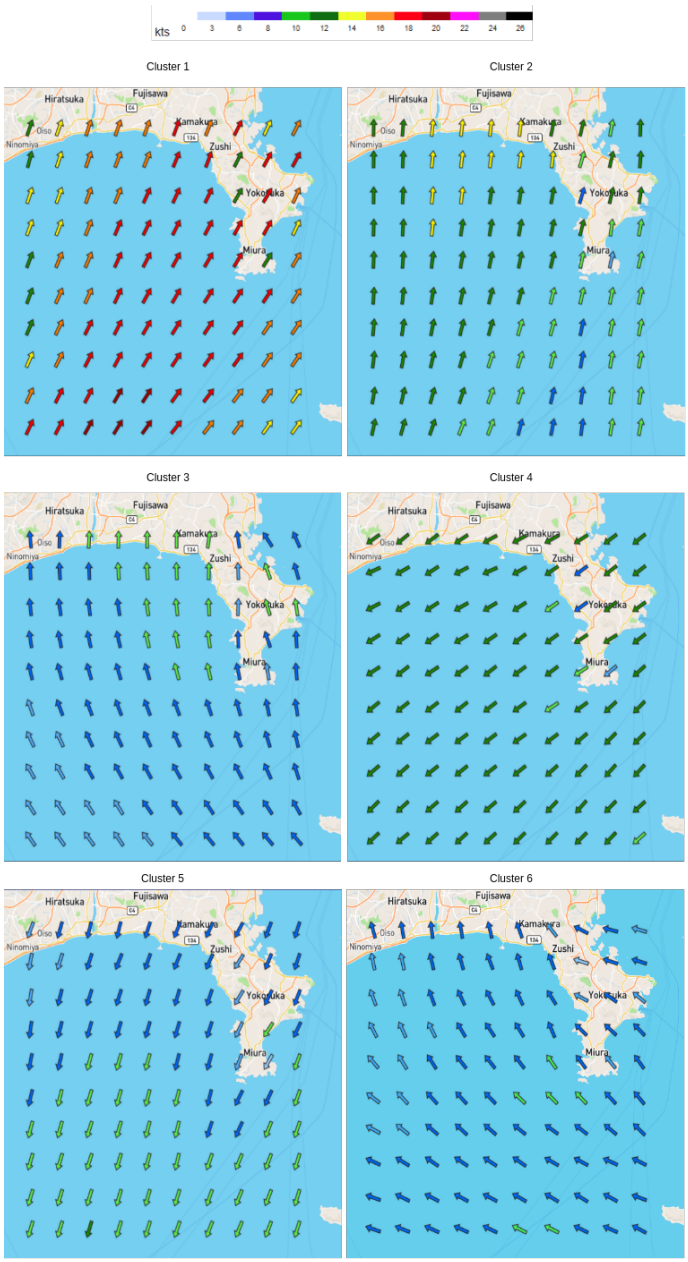

Figure 5.4: Filtered WRF - Average directions and speeds

Considering that this first analysis is done just with values derived from the numerical prediction model, which gives an interpretation of a possible evolution of reality, it is fundamental to analyse the results obtained using the real measurement on the sea. This is done in the next sections.

5.2 Boat data: Conventional clustering

Note that we denote "**conventional**" the clustering of timestamps (half an hour) versus the clustering of sequences (**time series/sequential clustering**) in the following sections.

5.2.1 All ribs

(The reports used for this analysis correspond to files Boat-k6-th0.6.pdf and comparison.pdf in the Annexes).

For Boat data, first we want to perform an analysis using all the actual data that we have for both types of ribs (moving/stationary). Regarding the days to filter out, they are the days for which we have not enough data (mainly, the ones that only contain data corresponds to night hours, which is not so relevant), they are:

- 21-24 July 2018.
- 7, 8, 10 August 2018.
- 5 September 2018.

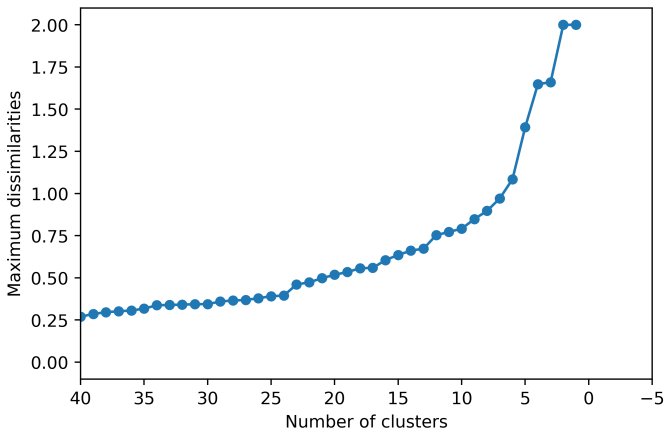

Figure 5.5: Boat conventional (all ribs) - Dissimilarities plot

According to the above dissimilarities plot, the preferred choices of the number of clusters are 5 and 6. Out of these clear possibilities, we have considered having 6 clusters to keep us aligned with the previous analysis.

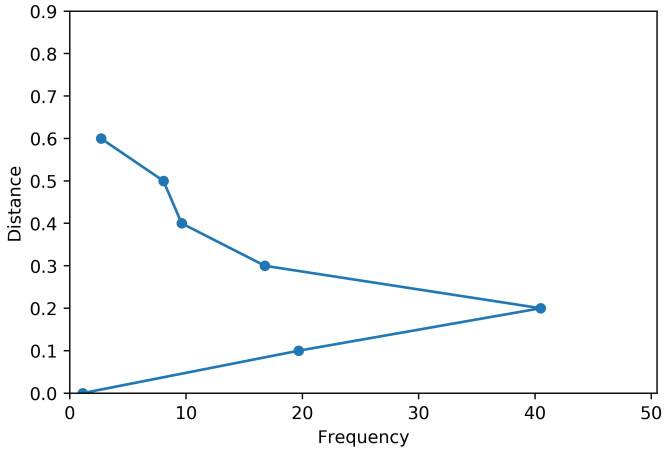

Figure 5.6: Boat conventional (all ribs) - Thresholds plot

Then, with the help of the thresholds plot, we decided to use a value of 0.6, as most of the distances are less or equal than this value. As we can deduce from the table below, the average wind direction is now distributed as follows:

- Cluster 1: 210 – SSW.
- Cluster 2: 190 – S.
- Cluster 3: 070 – E.
- Cluster 4: 240 – SW.
- Cluster 5: 130 – SE.
- Cluster 6: 020 – NE.

Cluster	Min speed	Avg speed	Max speed	Min direction	Avg direction	Max direction
1	4.182	10.198	20.627	181.544	211.061	346.361
2	3.013	9.671	20.251	15.774	188.923	352.419
3	2.161	8.962	16.967	0.691	69.604	343.895
4	2.254	10.520	18.978	4.427	241.386	356.053
5	1.934	5.816	10.650	13.149	132.283	348.913
6	3.232	6.804	13.915	4.052	18.442	348.701

Table 5.9: Boat conventional (all ribs) - Statistics (1)

In order to have a preliminary equivalence between the clustering obtained from the numerical model and the one from the actual measured data, we can apply the Maximum Match measure defined in section 2.7.1 (obtaining a global MM measure of 0.430). In table 5.10, we can see the number of each cluster (with its number of elements in parentheses) and its more probable equivalent in the other clustering.

The first evident element is that the real data show one different cluster compared with the WRF model, the 240-SW wind. This information is very useful to validate the performance of the numerical weather prediction model.

Considering that the SW direction is either the direction associated with Tropical Cyclones or with the fully developed sea breeze, it is evident that the model underestimates this direction, giving as the most right wind average the 210 direction. The other clusters seem to be similar to the ones observed by the model.

Cluster Boat	Cluster WRF JAPAN	Common elements
1 (106)	1 (113)	49
3 (93)	4 (39)	47
2 (105)	2 (60)	39
6 (34)	5 (29)	25
5 (48)	3 (52)	16
4 (49)	7 (3)	1

Table 5.10: Clusters matching (Boat conventional (all ribs) - Filtered WRF)

By examining the table of humidity, we can see that cluster 3, so the easterly winds, are characterized by the lower values of humidity and that the clusters 1, 2 and 4, by the highest. Actually this is particularly true, since the S to SW wind, come from the open ocean.

Cluster	[59.700, 66.282)	[66.282, 72.863)	[72.863, 79.444)	[79.444, 86.025)	[86.025, 92.607]
1	0.00%	0.00%	15.45%	51.21%	33.33%
2	0.00%	0.00%	5.49%	57.80%	36.71%
3	31.46%	45.17%	3.12%	18.07%	2.18%
4	0.00%	0.00%	11.57%	67.77%	20.66%
5	0.00%	8.18%	9.09%	55.45%	27.27%
6	0.00%	42.03%	11.59%	31.88%	14.49%

Table 5.11: Boat conventional (all ribs) - Ranges of HUMIDITY (in %)

Considering the difference in humidity between cluster 3 and 5, so between the E and SE wind, we can confirm, as observed previously by analysing the weather model's data, that the SE wind has higher values of humidity than cluster 3.

In this case, it is also interesting to observe the real values of precipitation. Clusters 2 and 5, so S and SE winds, are the ones associated with the highest values of precipitation. Unfortunately, in the case of the weather model, it was not very easy the identification of clear differences in precipitation among the different clusters.

Cluster	[0.000, 2.800)	[2.800, 5.600)	[5.600, 8.400)	[8.400, 11.200)	[11.200, 14.000]
1	100.00%	0.00%	0.00%	0.00%	0.00%
2	90.17%	0.00%	0.00%	9.83%	0.00%
3	81.62%	10.59%	6.85%	0.31%	0.62%
4	100.00%	0.00%	0.00%	0.00%	0.00%
5	89.09%	0.00%	0.00%	3.64%	7.27%
6	68.12%	0.00%	26.09%	5.80%	0.00%

Table 5.12: Boat conventional (all ribs) - Ranges of PRECIPITATION (in mm/h)

Considering air pressure values, cluster 3, so the easterly winds, are associated with the highest values of pressure. While cluster 6, so the NE winds have also high pressure values but in average lower compared with cluster 3. This distribution of air pressure is very similar to the one observed during the analysis of the weather prediction model where cluster 5, so the NNE winds, were having lower values of pressure than cluster 4 (NE). Therefore it seems particularly likely that, in case of

a NNE-NE wind, an increase in air pressure would lead to a rotation to the right towards ENE or E.

Cluster	[998, 1003)	[1003, 1008)	[1008, 1013)	[1013, 1018)	[1018, 1023]
1	0.00%	53.94%	42.12%	3.94%	0.00%
2	0.00%	69.65%	28.90%	1.45%	0.00%
3	0.62%	11.53%	45.48%	3.12%	39.25%
4	0.00%	18.18%	77.69%	4.13%	0.00%
5	7.27%	28.18%	63.64%	0.00%	0.91%
6	0.00%	8.70%	13.04%	69.57%	8.70%

Table 5.13: Boat conventional (all ribs) - Ranges of PRESSURE (in hPa)

These are just some examples of the very useful analysis that can be derived from the observation of the clusters and of the associated weather parameters. However, as stated in the previous chapters, one of the most important information that can be derived from the analysis of such data is the probability of transition from one cluster to another. Moreover, it would be fundamental to understand why and what is happening during the transition from one cluster to another. That's why it has been decided to represent the transition tables.

Cluster	1	2	3	4	5	6	7	8
1	91.92	7.07	0.00	0.00	0.00	0.00	0.00	1.01
2	15.79	78.95	5.26	0.00	0.00	0.00	0.00	0.00
3	0.00	4.35	93.48	0.00	0.00	2.17	0.00	0.00
4	0.00	0.00	0.00	96.88	0.00	3.12	0.00	0.00
5	0.00	0.00	0.00	11.11	88.89	0.00	0.00	0.00
6	0.00	0.00	21.43	0.00	0.00	78.57	0.00	0.00
7	66.67	0.00	0.00	0.00	0.00	0.00	33.33	0.00
8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 5.14: Filtered WRF - Transition matrix

Cluster	1	2	3	4	5	6
1	58.51	25.53	1.06	14.89	0.00	0.00
2	21.11	60.00	1.11	8.89	6.67	2.22
3	0.00	2.56	80.77	0.00	14.10	2.56
4	26.32	13.16	7.89	36.84	7.89	7.89
5	0.00	24.24	15.15	3.03	57.58	0.00
6	0.00	3.70	33.33	0.00	3.70	59.26

Table 5.15: Boat conventional (all ribs) - Transition matrix

Tables 5.14 and 5.15 depict transition information in both clusterings. The first information that can be derived is that in most of the cases, the cluster stays inside the same cluster. This information is not particularly useful for the purpose of understanding why and when a certain wind direction is switching to another one.

Therefore the analysis should be deepened on the little cases when there is a transition from one cluster to another. In both clusterings, clusters 1 and 2 are equivalent and represent respectively the SSW and S winds. We can notice that, considering WRF model, cluster 1, when changing cluster, goes to cluster 2, that means a transition from SSW to S. The same happens considering the measured data. On the other hand, it is very interesting to notice that the ribs data give additional crucial information: cluster 1 can also switch to cluster 4, that is, to the SW direction, which is the one not forecasted by the weather model.

Cluster 2, so the S wind, can change to cluster 1 both considering WRF and Boat data. On the other hand, WRF data show a possible transition to cluster 3, so SSE, while the ribs data show a possible transition to any other cluster, with a higher percentage to clusters 4 and 5 so either SW or SE.

Considering the application of such results to decision making in sailing, one can realize that further analysis should be performed. Indeed, if the start of an Olympic race is given with the wind inside cluster 2, so southerly wind, the sailor has the following information:

1. The most likely is that the wind stays from South.

2. The wind can switch to SSW.

3. The wind can also switch to SE or to SW.

This information is not usable for making a proper decision. That's why it has been decided to deepen even more the analysis of transitions, considering first of all the values of differences in each meteorological variable during the transitions from one cluster to another or even when the cluster was not changing. This is essential information to identify the behaviour of additional weather parameters and therefore to give to the decision maker evidence-based signs in order to make a better strategy.

As an example of the usability of this more detailed information let's consider the transition from cluster 2, S wind, to cluster 4, SW, or to cluster 5, SE, in the case of Boat data.

Transition	Temperature	Humidity	Wind Speed	Wind Direction
2 to 5	no change or decrease	no change in 100% of the cases	No change or decrease	No change or shift to the left
2 to 4	no change or mainly increase	mainly decrease	No change or significant increase	Most likely shift to the right
2 to 2	no change or slight increase	mainly decrease but less significant than 2 to 4	No change or slight increase	No change

Table 5.16: Boat conventional (all ribs) - Detailed transition information

As reported in the table above, we can conclude that in case of decrease of temperature, no change of humidity and decrease in speed, we can expect a likely shift from S to SE direction. On the other hand, with an increase in temperature, a decrease in humidity, a significant increase in speed, the wind would most likely shift to SW.

It should be noticed that this analysis is made with a limited number of records, therefore no sure conclusion can be derived. On the other hand, the methodology is very promising and would lead to more accurate results when the number of records increase.

A final more in-depth analysis has been then performed. Indeed, the cluster represents pretty big ranges of wind direction. Therefore, saying that for example

cluster 2 will change to cluster 4 or 5, might give an indication on the clockwise or anticlockwise change but not on the specific numbers of direction.

That is why we have decided to introduce the more detailed transition tables explained in section 4.8: as seen in figure 5.7, if one looks at transition 2 to 4, it seems interesting to notice that winds from 135 to 180 go to the range 200-225 in the 100% of the cases. On the other hand winds from 180 to 200 can switch to 225 up to 270. Finally winds from 202 to 225 change to 225-245.

Table 34: Transition 2-4: Changes of TWD

TWD Range	[0-22.5)	[22.5-45)	[45-67.5)	[67.5-90)	[90-112.5)	[112.5-135)	[135-157.5)	[157.5-180)	[180-202.5)	[202.5-225)	[225-247.5)	[247.5-270)	[270-292.5)	[292.5-315)	[315-337.5)	[337.5-360]
[0-22.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[22.5-45)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[45-67.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[67.5-90)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[90-112.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[112.5-135)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[135-157.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	100.00 (1)	0%	0%	0%	0%	0%	0%
[157.5-180)	0%	0%	0%	0%	0%	0%	0%	0%	0%	100.00 (1)	0%	0%	0%	0%	0%	0%
[180-202.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	50.00 (2)	50.00 (2)	0%	0%	0%	0%
[202.5-225)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	100.00 (2)	0%	0%	0%	0%	0%
[225-247.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[247.5-270)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[270-292.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[292.5-315)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[315-337.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[337.5-360]	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

Figure 5.7: Boat conventional (all ribs) - Detail of TWD in transitions from cluster 2 to cluster 4

Now, it would be very useful to know specific characteristics of these transitions, especially concerning the other weather parameters: if we consider transition from 180-202 to 225-247 (figure 5.8), we immediately see that this happens with an increase of wind speed of about 3 knots, no changes in temperature and a decrease in humidity. While the change to 247-270 (figure 5.9) happens with a slight decrease in wind speed, a slight decrease in temperature and an increase in humidity.

Transition 2-4 ([180-202.5) - [225-247.5)): Range level average differences
diff_temp: 0.32172900000000304
diff_humidity: -1.9764000000000124
diff_precipitation: 0.0
diff_pressure: -0.44837127685542555
diff_gust: 2.4979692382812475
diff_high_cloud_coverage: -0.1821500000000003
diff_low_cloud_coverage: 0.0
diff_medium_cloud_coverage: 0.0
diff_temperature_at_ground: 0.0012999999999987466
diff_total_cloud_coverage: -0.1821500000000003
diff_tws: 3.3287446043833517
diff_twd: 43.513540741867374

Figure 5.8: Boat conventional (all ribs) - Transition 2-4 ([180-202.5)-[225-247.5))

This is very logical since the 210-220 direction is the direction of the fully de-

veloped sea breeze, so we find wind having high values of the speed, high values of temperature and low values of humidity.

Transition 2-4 ([180-202.5) - [247.5-270)): Range level average differences
diff_temp: -0.15214200000001554
diff_humidity: 2.058850000000014
diff_precipitation: 0.0
diff_pressure: -0.07314727783204944
diff_gust: 0.041161929321294366
diff_high_cloud_coverage: -0.8987500000000003
diff_low_cloud_coverage: 0.0
diff_medium_cloud_coverage: 0.0
diff_temperature_at_ground: -0.3568900000000035
diff_total_cloud_coverage: -0.8987500000000003
diff_tws: -0.24452412267598003
diff_twd: 60.42113736788963

Figure 5.9: Boat conventional (all ribs) - Transition 2-4 ([180-202.5)-[247.5-270))

Another example can be done by considering the transition from cluster 2 to 5 (figure 5.10). This transition mainly happens when the wind is changing from 157-180 to 135-157 or from 180-202 to 135-157 (see figures 5.11 and 5.12). The very interesting information is that the transition from 180-202 to 135-157 is characterized by a higher decrease in wind speed compared with the transition from 157-180 to 135-157.

Table 37: Transition 2-5: Changes of TWD

TWD Range	[0-22.5)	[22.5-45)	[45-67.5)	[67.5-90)	[90-112.5)	[112.5-135)	[135-157.5)	[157.5-180)	[180-202.5)	[202.5-225)	[225-247.5)	[247.5-270)	[270-292.5)	[292.5-315)	[315-337.5)	[337.5-360]
[0-22.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[22.5-45)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[45-67.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[67.5-90)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[90-112.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[112.5-135)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[135-157.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[157.5-180)	0%	0%	0%	0%	0%	0%	100.00% 1	0%	0%	0%	0%	0%	0%	0%	0%	0%
[180-202.5)	0%	0%	0%	0%	0%	0%	80.00% 4	20.00% 1	0%	0%	0%	0%	0%	0%	0%	0%
[202.5-225)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[225-247.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[247.5-270)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[270-292.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[292.5-315)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[315-337.5)	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[337.5-360]	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

Figure 5.10: Boat conventional (all ribs) - Detail of TWD in transitions from cluster 2 to cluster 5

Transition 2-5 ([157.5-180) - [135-157.5)): Range level average differences
diff_temp: -0.16490000000001004
diff_humidity: 1.4861999999999682
diff_precipitation: 0.0
diff_pressure: 0.07155181884763806
diff_gust: 0.0751615234374956
diff_high_cloud_coverage: 0.0
diff_low_cloud_coverage: -0.03287
diff_medium_cloud_coverage: 0.0
diff_temperature_at_ground: -0.39250000000000895
diff_total_cloud_coverage: -0.03287
diff_tws: 0.3861522021684074
diff_twd: -28.251243243600555

Figure 5.11: Boat conventional (all ribs) - Transition 2-5 ([157.5-180)-[135-157.5))

Transition 2-5 ([180-202.5) - [135-157.5)): Range level average differences
diff_temp: -0.027133462890622795
diff_humidity: 1.0415749999999662
diff_precipitation: 0.0
diff_pressure: -0.2853866577148949
diff_gust: 1.1102904785156245
diff_high_cloud_coverage: 0.3897999999999997
diff_low_cloud_coverage: 0.0082175
diff_medium_cloud_coverage: 0.0
diff_total_cloud_coverage: 0.3980174999999997
diff_tws: -1.4209083949438834
diff_twd: -46.029135240234616

Figure 5.12: Boat conventional (all ribs) - Transition 2-5 ([180-202.5)-[135-157.5))

This result combined with the one of transition from cluster 2 to 4, can lead to the following conclusion: a wind from 180-202 will most likely turn towards 230-240 if the wind speed increases and to 140-150 if the wind speed decreases. This is key information that supports with evidence the theory on sea breeze behaviour, applying to the specific characteristics of the Enoshima Bay.

Many more conclusions could be derived from the analysis of the produced tables. However, as mentioned, the main goal of the presented work is not to describe in detail each table but to show how the developed methodology can be generalized and scaled up to other places just by having predicted or measured data about meteorological variables.

5.2.2 British ribs

(The reports used for this analysis correspond to files EM5-k6-th0.4.pdf and comparison.pdf in the Annexes).

After seeing the results obtained from the previous analysis, we wanted to perform an additional analysis using only stationary ribs. In this section, the clusters obtained from the static ribs are compared with the clusters obtained with WRF data considering only the closest points to each rib. The points are the 3 WRF points which are the closest to the coordinates of ribs 1, 4 and 5, according to the geodesic distance.

We have decided to examine this subset of Boat data in more detail for three reasons:

1. Because they are measured by static boats, one can consider that these data are going to be more "clean".

2. Because the boats have static positions, even if it is not much data, we can show the sort of geography we can obtain (equivalent to the one we already commented for the numerical model in section 5.1.2).

3. Because we can once more try and detect failures in the prediction of the numerical model by comparing the patterns obtained with actual data with the ones obtained, this time both globally and locally.

By checking the records, we have found that the stationary ribs 2, 3 and 6 have recorded very few data. Hence, it was decided to use only the data coming from ribs 1, 4 and 5. As to the hours to be considered, we have observed that most of the data of a day were recorded from 2:00 a.m. to 7:00 a.m. UTC. Although for some days some ribs have more data, to be consistent, we only considered the data recorded in the mentioned period (from 2:00 a.m. to 7:00 a.m. UTC, both included). Additionally, the pruned dataset covers the period from 25 July 2018 to 6 August 2018 except the 29th of July 2018 and the 1st of August 2018. We selected this data because we are looking for days/hours where all the 3 ribs have readings since this is important for our clustering methodology to work properly.

First of all, as we did in section 5.1.2, we use the Maximum Match measure to compute the equivalences between clusters. The table below represents the comparison (the global measure obtained is 0.483).

Cluster 5M	Cluster WRF JAPAN (3 closest points)	Common elements
3 (25)	4 (8)	16
1 (32)	1 (18)	15
2 (31)	3 (14)	15
4 (17)	2 (15)	9
5 (7)	5 (3)	1
6 (4)	6 (2)	0

Table 5.17: Clusters matching (Boat conventional (static ribs) - WRF (3 closest points))

We have used the arrows (barbs) maps to graphically compare the local behaviour of the wind for each of the clusters for both the 3 5M ribs and the 3 chosen points of WRF model. The 6 different racing areas have been depicted (each of the 5M ribs is located in one of them).

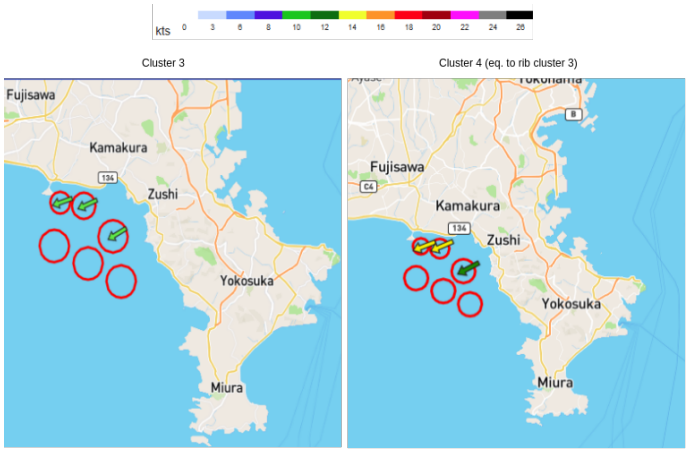

Figure 5.13: Boat conventional (static ribs) - WRF (3 closest points): equivalence of cluster 3

Considering cluster 3 from ribs and cluster 4 from the weather model (figure 5.13), we notice that the TWD is similar while the weather model predicts a higher speed, around 12-14 kts, while the measured one is around 10 kts.

Also, in the case of cluster 1, SSW wind (figure 5.14, the upper part), the WRF model predicts a wind speed higher than the real one, by 2-3 kts.

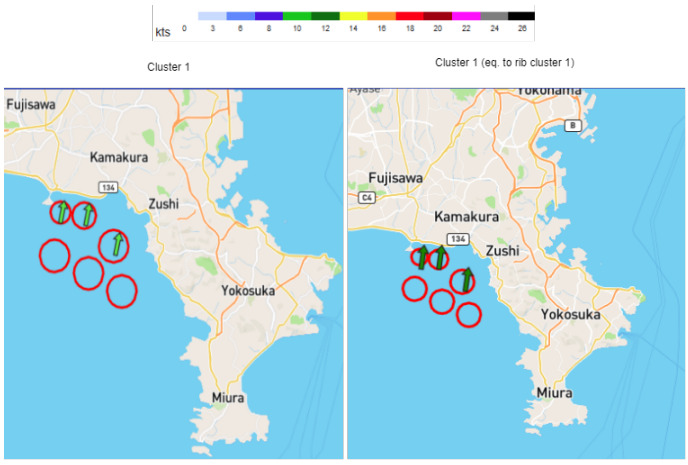

Figure 5.14: Boat conventional (static ribs) - WRF (3 closest points): equivalence of cluster 1

Cluster 2 from ribs, associated with cluster 3 of WRF model, do not present significant differences from prediction to the reality (figure 5.15, the middle part).

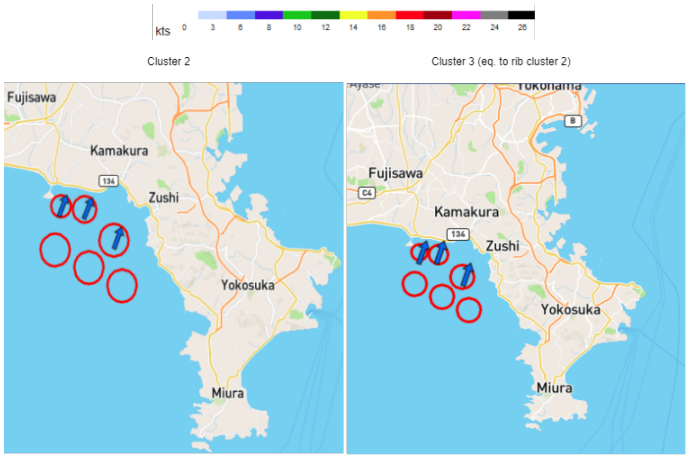

Figure 5.15: Boat conventional (static ribs) - WRF (3 closest points): equivalence of cluster 2

Cluster 4 of ribs is associated with cluster 2 of WRF model (figure 5.16, the lower part). This association results particularly strange since cluster 4 corresponds to SW winds while cluster 2 of WRF model, corresponds to SSE winds. It indicates a disagreement between forecasted winds by the model and the actual data: in spite of what's predicted, the occurrence of SW ends up being much higher. This is a phenomenon we have already observed in the analysis with complete Boat dataset (section 5.2.1).

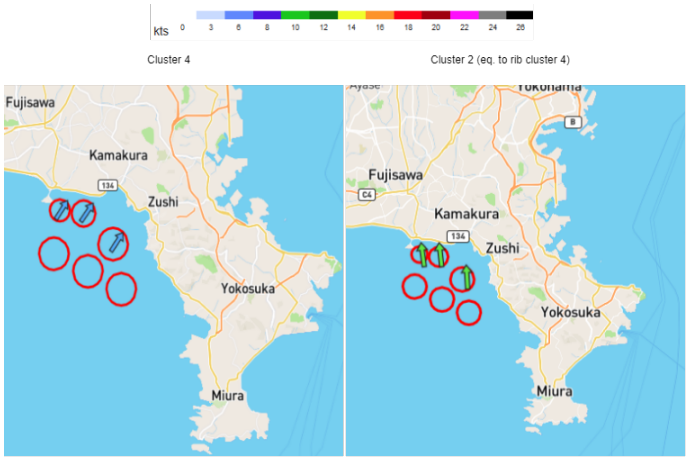

Figure 5.16: Boat conventional (static ribs) - WRF (3 closest points): equivalence of cluster 4

Finally, clusters 5 and 6 have very few numbers of association so they are not taken into account.

The association of measurement to weather prediction model's data and the comparison of clusters is an advantage in identifying persistent errors of the model. Based on such observation, if confirmed by a bigger number of observations on the sea, one can consider adjusting the model output to the measured values. For instance, by this analysis, it can be assumed that in the case of NE and SSW winds the model should be adjusted with a decrease of about 2-3 knots from the predicted values.

5.3 Boat data: Time series clustering

(The reports used for this analysis correspond to files BoatSequentialPLR-k6-th0.5.pdf and comparison.pdf in the Annexes).

For the time series clustering, the first thing we had to do is to decide which data to use. We found that for the current Boat dataset, some ribs did not record enough data in terms of the number of hours that the data covers. Therefore, it was decided to use only, for each of the days that we have records excluding the dates mentioned in section 5.2.1, the ribs whose data covers more than 4 hours of that day. The data is then transformed into corresponding sequences using PCA

and PLR, as explained in the previous section. After applying data processing, we observed that the global average number of segments for the sequences is 42. More specifically, an average of 64 segments for moving ribs and 18 segments for stationary ribs, is observed. Moreover, we have seen that the average time covered by a segment is around 13 minutes for moving ribs and 37 minutes for static ribs, which leads to a global average of 25 minutes. We considered that the difference of behaviour between the Austrian data and the British data is due to the different sampling frequency. Recall that the sampling frequency of the Austrian ribs is 5 Hz while the British ribs record once every five minutes.

Then, from the figure below we can see that the preferred choices of the number of clusters could be 6, 5 and 4. Since we think it is few having only 4 clusters and 6 is a good option, in order to see the equivalence between time series clustering and conventional clustering, we have decided to use 6 clusters as well.

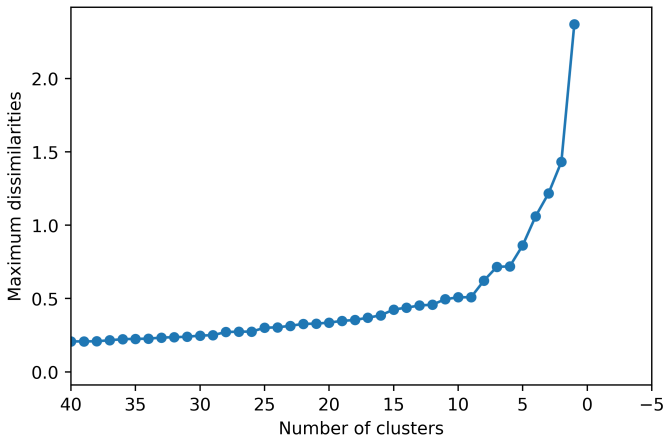

Figure 5.17: Boat time series - Dissimilarities plot

Since we have few data (92 pairs of day and rib) for time series clustering after filtering, by observing figure 5.18, for the threshold we took the value of 0.5 to keep most of the elements.

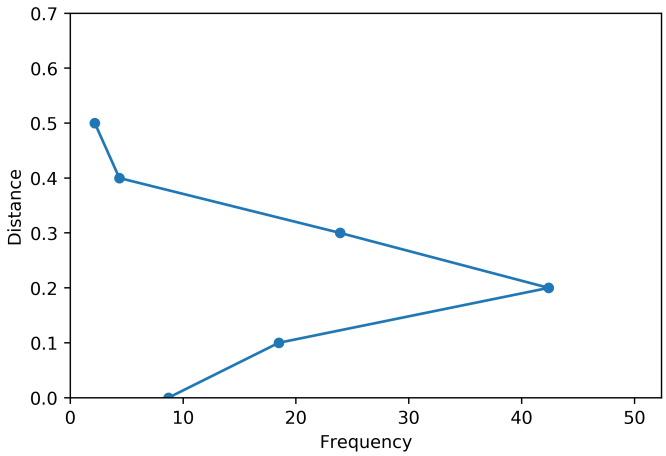

Figure 5.18: Boat time series - Thresholds plot

We are aware that so far, we do not have enough data to extract relevant information, but again, the idea is to present the methodology. In this case, the percentage of "ribs of the same day in the same cluster" is about 86% (the measure is explained in section 4.7), we have noticed that most of the "wrongly classified" sequences contain relatively few data, i.e. cover fewer hours, which reminds us again that more data should be provided in order to get more accurate results. Regarding the computation of centroids, in general, it converges quickly.

Data derived from time series clustering present another key perspective in data interpretation. The first key point is that days belonging to the same cluster are grouped together. The information concerning which day belongs to which cluster is very useful to approach the pattern identification by starting from groups of days having a similar wind behaviour.

For instance, considering cluster 2, one can see from figure 5.19 that days 11th, 12th and 13th of September belong to the same cluster. On the other hand, information derived from the human interpretation of the meteorologist reports that these three days were classified as belonging to the same pattern, named 'NE to E wind'. That means that, after observing the wind data on the sea and after discussing the relevant wind features with sailors, the three days were considered as having similar behaviour.

Cluster 2 with 22 elements:
20180726 by boat 12
20180726 by boat 5m5
20180726 by boat 5m4
20180726 by boat 5m1
20180726 by boat 5m2
20180727 by boat 5m5
20180727 by boat 5m1
20180727 by boat 5m4
20180730 by boat 5m4
20180730 by boat 5m5
20180911 by boat 9
20180911 by boat 12
20180911 by boat 8
20180911 by boat 1
20180911 by boat 4
20180912 by boat 9
20180912 by boat 1
20180912 by boat 12
20180913 by boat 12
20180913 by boat 4
20180913 by boat 8
20180913 by boat 9

Figure 5.19: Time series clustering - Composition of of Clusters 2

The second key information is the representation into a graph of the evolution hour by hour of the variation of each meteorological parameter.

Again concerning the above mentioned days, the information gathered by the meteorologist, and reported manually in a book, says that the wind has tended to start from a stronger and more consistent NE (040-060), rotating clockwise to E (070-090). The easterly wind was associated with generally lighter wind speed.

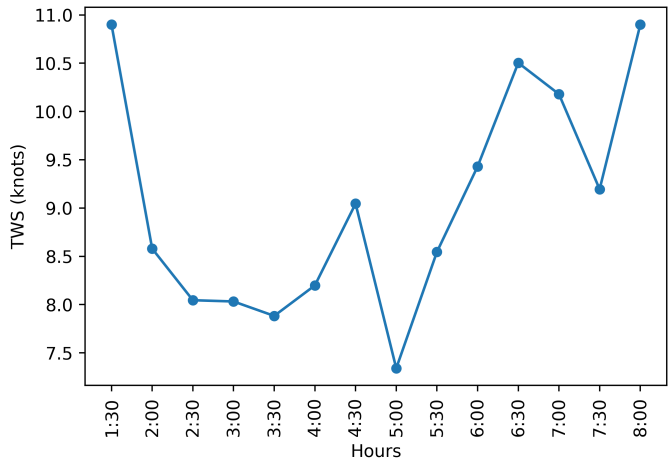

Figure 5.20: Time series clustering - TWS of Clusters 2's centroid

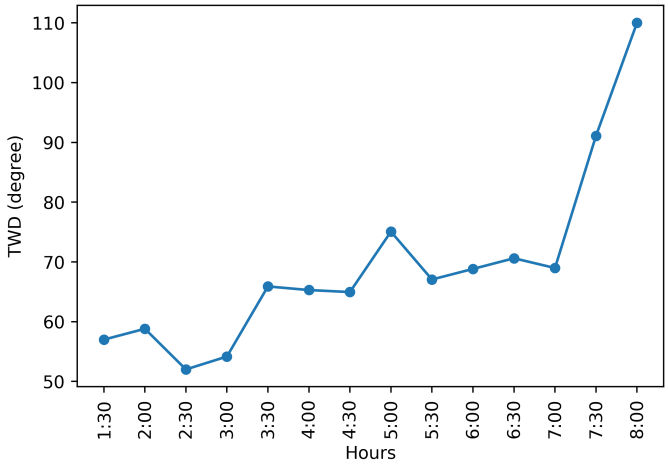

Figure 5.21: Time series clustering - TWD of Clusters 2's centroid

Considering the graphics of TWD and TWS (see figures 5.20 and 5.21, recall that TWD and TWS stand for True Wind Direction and True Wind Speed), one can see that from 1:30 a.m. to 3:30 a.m. UTC the trend is to rotate clockwise and decrease in speed. Then from 3:30 a.m to 4:30 a.m. the TWD is pretty stable with an increase of the speed, after that, from 4:30 a.m. to 5:00 a.m there is another clockwise rotation with a decrease of speed. This behaviour from 1:30 a.m to 5:00 a.m. totally confirms qualitative observations recorded by the meteorologist during the 11th, 12th and 13th of September 2018. The very interesting point is that this time series clustering behaviour is derived not only with those three days but considering as well the 26th, 27th and 30th of July 2018, when no observations from the meteorologist are available. This can give a positive sign regarding:

1. The good match between manual patterns identification and automatic clustering.

2. The possibility of generalizing this approach to other areas.

Of course, having a much bigger database would lead to an increase in the accuracy and reliability of the methodology. Moreover, the same approach could be applied to data coming from the numerical prediction models, giving the meteorologist a very important input concerning the type of cluster of the day, starting from the early morning and therefore from the crucial moment when the team plans the best strategy for the races.

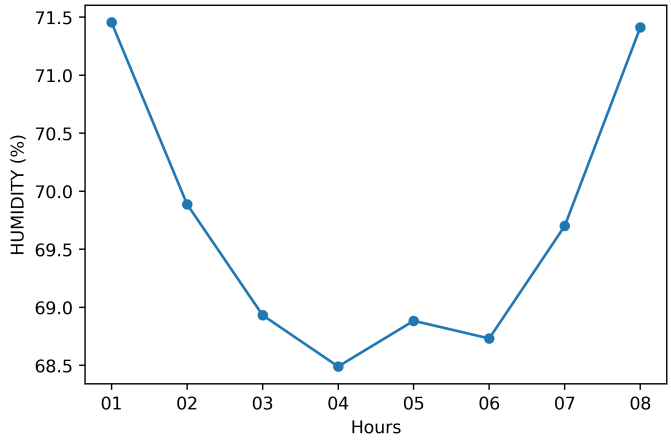

Figure 5.22: Time series clustering - average Humidity of Clusters 2

Additional input gathered by the observation of graphic of humidity (figure 5.22), is that a decrease of humidity characterizes the behaviour observed by the meteorologist and confirmed by the time series clustering. On the other hand, when the humidity starts to rise, the wind continues the clockwise rotation but with an increase of the wind speed.

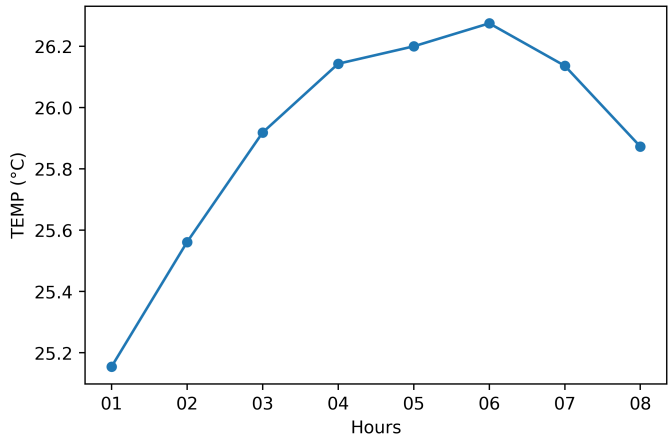

Figure 5.23: Time series clustering - average Temperature of Clusters 2

Finally, we can observe from figure 5.23 that the air temperature increases together with the decrease in humidity and decreases when the humidity starts to rise. The variation of these two parameters, apparently not reported by the meteorologist, can be easily measured, giving a big advantage in the ability to predict evolution of the wind in near real time. This last consideration shows how much the ability of a

machine to analyse a big amount of data in a short time can positively complement the qualitative interpretation of a human expert, who is essential to make sense out of data and to translate them to users, but is not able to take into account all the information available from weather stations.

Chapter 6

Conclusions

In this thesis, we have extended and developed a framework that is able to perform clustering analysis of wind data and thus, to recognise details of significant wind patterns focused on specific area which consist of characteristic features of the wind speed and direction related with the other meteorological parameters and with the geographical position of the particular area.

Our framework has the following characteristics:

- Flexibility: it is a flexible methodology which can deal with different types of clustering and with multiple data sources. On the one hand, it supports conventional clustering using data produced by the numerical prediction model or collected by the ribs. On the other hand, time series clustering is also permitted. Moreover, temporal/spatial criteria can be applied both in clustering (for normalisation, see section 4.4) and analysis (for analysing the wind behaviours of different time/location).

- Scalability: even though we have focused on the Tokyo Olympic Games, in fact, the proposed methodology is scalable and can be applied to any area where an event is going to take place.

In spite of the difficulties we have encountered such as understanding the background and previous works, dealing with a raw dataset that needs to be pruned carefully, working with final users' changing requirements, etc., many promising results have been derived from the analyses. On the one hand, we have found and

confirmed the relations between the evolution of winds and other weather parameters such as humidity, temperature and cloud coverage. On the other hand, we have noticed that there exist disagreements between winds forecasted by the WRF model and the real winds. Based on such disputes, if confirmed by a bigger number of observations on the sea, one can consider adjusting the model output to the measured values.

Since it was the first time that the meteorologists collect real data over Enoshima Bay, we have faced many input data issue as explained in section 3.3.1.1. Nevertheless, it is remarkable that even with this limited dataset, we have already been able to draw conclusions and the final users (the meteorologists) are very satisfied with the environment and looking forward to working with it using more data (they will be collecting more data this summer 2019 and then finally in the previous weeks to the Olympics in 2020). They have already been able to identify potential reasons for transitions (evolution of the wind according to each wind pattern), and they are confident that with more data, they will be able to identify more different behaviours (both static and dynamic) in the different parts of Enoshima Bay according to each pattern.

6.1 Future work

In the short-term future, it would be interesting to perform more analysis of the current data. Two immediate tasks could be carried out:

1. Applying time series clustering with WRF data. The idea would be taking the same days we considered for Boat sequential clustering, and as WRF points, take the 6 points closer to the positions of British ribs since each one represents the centre of each racing area. In this way, we can compare even more the two datasets and potentially detect more similarities/differences.

2. Applying sequential clustering for Boat data with area filtering. Given an input squared area (delimited as max/min latitude and longitude), we would only take as input data those pairs of day and rib, for which a high percentage

of the records are inside this area. The idea is to try to detect different behaviours according to the areas.

Additionally, the coaches can add environmental and/or strategical impressions to each day when they go out to sea: wind behaviour (such as periodical shifts, vertical profile, etc.), side of the race course which is favoured, visibility, etc. If we manage to categorise this data, they might also be considered in profiling when doing the clustering.

On the other hand, in the long-term future, several directions of future work could be addressed:

- **Automatic Hierarchical clustering**: as mentioned, the automatic hierarchical clustering developed in the previous thesis did not work properly with the current data. Therefore, it would be interesting having a procedure that automatically chooses the "best" number of clusters and then compare it with the current approach.

- **Classification**: we would like to have an approach that could automatically classify the incoming data (hourly wind fields or sequences) into the existing clusters. This goal should not be hard to gain due to the nature of our methodology. For example, based on the clusters that we have, when new data comes in, by using only the distance measure, we could quickly identify the nearest cluster to which the new data should belong.

- **Matrix Profile**: a recently proposed near-universal time series data mining tool, called Matrix Profile (MP) [38] [39], has caught the attention of us. The MP computes and stores the all-pairs-similarity-search information in an efficient and easy-to-access fashion, and this information can be used in a variety of data mining tasks ranging from well-defined tasks (e.g., motif discovery) to more open-ended tasks (e.g., representation learning). Some works related to time series clustering is presented in [40] [41] [42]. We were amazed by the power of MP and seek to incorporate it into our framework, for example, find relevant motifs in the time series.

Bibliography

[1] TriM - Toyko 2020. http://www.trimweb.it/project_tokyo2020.

[2] Australia Sailing. https://www.sailing.org.au/home/.

[3] Meteorological wind direction. http://tornado.sfsu.edu/geosciences/classes/m430/Wind/WindDirection.html.

[4] Wind direction. http://colaweb.gmu.edu/dev/clim301/lectures/wind/wind-uv.

[5] Ryan May et al. *MetPy: A Python Package for Meteorological Data*. https://github.com/Unidata/MetPy. Version 0.4.3. Boulder, Colorado: Unidata, 2008 - 2017. DOI: 10.5065/D6WW7G29.

[6] Pirmin Kaufmann and C. David Whiteman. "Cluster-Analysis Classification of Wintertime Wind Patterns in the Grand Canyon Region". In: *Journal of Applied Meteorology* 38.8 (1999), pp. 1131–1147.

[7] Pirmin Kaufmann and Rudolf O Weber. "Classification of mesoscale wind fields in the MISTRAL field experiment". In: *Journal of Applied Meteorology* 35.11 (1996), pp. 1963–1979.

[8] Fabio Di Francesco. "Wind pattern analysis applied to Tokyo 2020 Olympic Games". MA thesis. Universitat Politècnica de Catalunya, 2018.

[9] Nusa Erman, Ales Korosec, and Jana Suklan. "PERFORMANCE OF SELECTED AGGLOMERATIVE HIERARCHICAL CLUSTERING METHODS". In: *Innovative Issues and Approaches in Social Sciences* 8.1 (2015), pp. 180–204.

[10] Bruce M Russett. "Inequality and instability: The relation of land tenure to politics". In: *World Politics* 16.3 (1964), pp. 442–454.

[11] Joe H Ward Jr. "Hierarchical grouping to optimize an objective function". In: *Journal of the American statistical association* 58.301 (1963), pp. 236–244.

[12] James MacQueen et al. "Some methods for classification and analysis of multivariate observations". In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.

[13] David Wishart. *Fortran II programs for 8 methods of cluster analysis (CLUSTAN I)*. State Geological Survey, 1969.

[14] Michael R Anderberg. *Cluster analysis for applications*. Academic press, 1973, p. 359.

[15] Douglas Steinley. "K-means clustering: a half-century synthesis". In: *British Journal of Mathematical and Statistical Psychology* 59.1 (2006), pp. 1–34.

[16] Wikipedia contributors. Hierarchical clustering — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Hierarchical_clustering&oldid=901301136. [Online; accessed 12-June-2019]. 2019.

[17] Silke Wagner and Dorothea Wagner. *Comparing clusterings: an overview*. Universität Karlsruhe, Fakultät für Informatik Karlsruhe, 2007.

[18] Wikipedia contributors. Time series — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Time_series&oldid=900896085. [Online; accessed 20-June-2019]. 2019.

[19] Saeed Aghabozorgi, Ali Seyed Shirkhorshidi, and Teh Ying Wah. "Time-series clustering–A decade review". In: *Information Systems* 53 (2015), pp. 16–38.

[20] T Warren Liao. "Clustering of time series data—a survey". In: *Pattern recognition* 38.11 (2005), pp. 1857–1874.

[21] Eamonn Keogh and Jessica Lin. "Clustering of time-series subsequences is meaningless: implications for previous and future research". In: *Knowledge and information systems* 8.2 (2005), pp. 154–177.

[22] Peter Lynch. "The origins of computer weather prediction and climate modeling". In: *Journal of Computational Physics* 227.7 (2008), pp. 3431–3444.

[23] Numerical Weather Prediction - National Center for Environmental Information. https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/numerical-weather-prediction.

[24] WRF Model site. https://www.mmm.ucar.edu/weather-research-and-forecasting-model.

[25] Wikipedia contributors. Weather Research and Forecasting Model — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Weather_Research_and_Forecasting_Model&oldid=892598039. [Online; accessed 13-June-2019]. 2019.

[26] Wikipedia contributors. Geodesics on an ellipsoid — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Geodesics_on_an_ellipsoid&oldid=900205143. [Online; accessed 9-June-2019]. 2019.

[27] GeoPy. https://github.com/geopy/geopy.

[28] Yoshiki Tanaka, Kazuhisa Iwamoto, and Kuniaki Uehara. "Discovery of time-series motif from multi-dimensional data based on MDL principle". In: *Machine Learning* 58.2-3 (2005), pp. 269–300.

[29] Tak-chung Fu. "A review on time series data mining". In: *Engineering Applications of Artificial Intelligence* 24.1 (2011), pp. 164–181.

[30] Eamonn Keogh. "Fast similarity search in the presence of longitudinal scaling in time series databases". In: *Proceedings Ninth IEEE International Conference on Tools with Artificial Intelligence*. IEEE. 1997, pp. 578–584.

[31] Donald J Berndt and James Clifford. "Using dynamic time warping to find patterns in time series." In: *KDD workshop*. Vol. 10. 16. Seattle, WA. 1994, pp. 359–370.

[32] Eamonn J Keogh and Michael J Pazzani. "Derivative dynamic time warping". In: *Proceedings of the 2001 SIAM international conference on data mining*. SIAM. 2001, pp. 1–11.

[33] Eamonn Keogh and Chotirat Ann Ratanamahatana. "Exact indexing of dynamic time warping". In: *Knowledge and information systems* 7.3 (2005), pp. 358–386.

[34] Dan Gusfield. *Algorithms on strings, trees, and sequences: computer science and computational biology*. Cambridge university press, 1997.

[35] Waleed H Abdulla, David Chow, and Gary Sin. "Cross-words reference template for DTW-based speech recognition systems". In: *TENCON 2003. Conference on convergent technologies for Asia-Pacific region*. Vol. 4. IEEE. 2003, pp. 1576–1579.

[36] Ville Hautamaki, Pekka Nykanen, and Pasi Franti. "Time-series clustering by approximate prototypes". In: *2008 19th International Conference on Pattern Recognition*. IEEE. 2008, pp. 1–4.

[37] Wikipedia contributors. Mean of circular quantities — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Mean_of_circular_quantities&oldid=885307281. [Online; accessed 14-June-2019]. 2019.

[38] Eamonn Keogh. The ucr matrix profile page. http://www.cs.ucr.edu/~eamonn/MatrixProfile.html.

[39] Chin-Chia Michael Yeh. "Towards a Near Universal Time Series Data Mining Tool: Introducing the Matrix Profile". PhD thesis. UCR, 2018.

[40] Chin-Chia Michael Yeh, Helga Van Herle, and Eamonn Keogh. "Matrix profile III: the matrix profile allows visualization of salient subsequences in massive time series". In: *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE. 2016, pp. 579–588.

[41] Chin-Chia Michael Yeh, Nickolas Kavantzas, and Eamonn Keogh. "Matrix profile VI: Meaningful multidimensional motif discovery". In: *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE. 2017, pp. 565–574.

[42] Shaghayegh Gharghabi et al. "Matrix profile VIII: domain agnostic online semantic segmentation at superhuman performance levels". In: *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE. 2017, pp. 117–126.