Visualizing and Clustering Data that Includes Circular Variables

Garland Will

Department of Mathematical Sciences

Montana State University

May 4, 2016

A writing project submitted in partial fulfillment
of the requirements for the degree

Master of Science in Statistics

APPROVAL

of a writing project submitted by

Garland Will

This writing project has been read by the writing project advisor and has been found to be satisfactory regarding content, English usage, format, citations, bibliographic style, and consistency, and is ready for submission to the Statistics Faculty.

Date

Mark Greenwood
Writing Project Advisor

Date

Mark Greenwood
Writing Projects Coordinator

Abstract

Circular variables are an interval variable type that occur in a variety of applications. Multivariate analyses that incorporate circular variables require special consideration. This paper explores how to visualize and analyze quantitative and circular variables together using meteorological data that include wind speed, temperature, and wind direction. Visualization is accomplished using a novel modification of a parallel coordinate plot. Gower's dissimilarity is used to create an overall distance matrix that includes the circular variable. Ward's method is applied to this distance matrix to develop a cluster solution, with the results displayed using the proposed parallel coordinate plot.

Acknowledgements

- A big thank you should be said to my advisor Dr. Mark Greenwood, for his substantial help and assistance.

- Bonneville Power Administration (BPA) should be mentioned here. Working as an intern for them the past year sparked my interest in wind, and provided much of the impetus for this project. The data set is also from BPA. While my work at Bonneville motivated my interest in wind, the work done here has no ties to Bonneville and this is a publicly available data set.

1 Introduction

Wind direction is an important component of meteorological data analysis. Circular data arise when responses are directions or time. Circular variables contrast with traditional quantitative variables in that there is only a certain range of possibles values the variable can take on. Also, directions of 0 and 360 degrees have the same meaning, or if using radians, the values 0 and 2π mean the same thing. These variables are very different from traditional continuous variables.

The data used for this analysis are meteorological data that consist of wind speed, wind direction, and temperature. The data set comes from a weather station on Cape Blanco, a cape protruding off of the southern Oregon coast near Port Orford (Bonneville, 1991). The Bonneville Power Administration, a branch of the US Department of Energy, historically had a weather station located on Cape Blanco. The data consist of hourly average values for the month of January, 1991.

The research question has multiple parts. The first part is concerned with creating a visual display that shows the relationship between a circular and multiple regular quantitative quantitative variables in an easy to understand fashion. The next part explores how to cluster a data set that includes a circular variable. The methods are illustrated with a display of the results of clustering using the novel display from the first part.

2 Circular Statistics

As mentioned above, circular data are unique, in that a value of 0 and 360 both represent the same thing when using degrees, or, if using radians, that the values 0 and 2π are the same. For time, 0 and 24 hours are the same time of day. These data are continuous and are interval scaled, which means ratios are not valid comparisons, but a ratio of differences is valid (Wikipedia, 2016b). Standard statistical methods must be modified to account for this aspect of circular data.

The two most common distributions when dealing with circular data are the wrapped normal and the Von Mises distributions. The probability density function of the wrapped normal distribution (Fisher, 1993) is

$$f(\theta) = \frac{1}{2\pi}(1 + 2\sum_{p=1}^{\infty} \rho^{p^2} \cos(p(\theta - \mu))) \text{ for } 0 \leq \theta < 2\pi \text{ and } 0 \leq \rho \leq 1.$$

The wrapped normal can be thought of as being obtained by wrapping a normal distribution around a circle, which is where the summation piece of the probability function comes into play. It is a symmetric, single peaked distribution that is centered at μ. The mean resultant length is ρ, with circular dispersion $\frac{1-\rho^4}{2\rho^2}$. In general, the wrapped normal is considered a complicated distribution to work with.

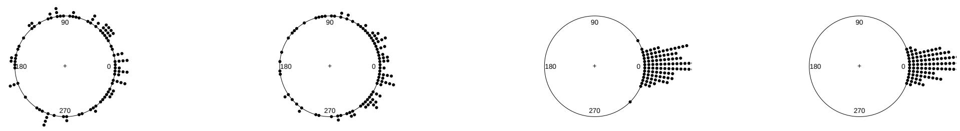
The Von Mises distribution is easier to work with than the wrapped normal and is the more commonly used circular distribution. The Von Mises Distribution (Fisher, 1993) is

$$f(\theta) = \frac{1}{2\pi I_0(\kappa)} \exp(\kappa * \cos(\theta - \mu)) \text{ where } 0 \leq \theta < 2\pi,\ 0 \leq \kappa < \infty \text{ and}$$

$$I_0(\kappa) = \frac{1}{2\pi} \int_0^{2\pi} \exp(\kappa * \cos(\phi - \mu)) d\phi.$$

$I_o(k)$ is the modified Bessel function of order 0. The mean direction is μ. κ is a concentration measure. $\frac{1}{\kappa}$ can be thought of as being analogous to σ^2 in the normal distribution (Wikipedia, 2015). As κ goes to 0 the Von Mises distribution converges to a wrapped normal distribution and as κ goes to infinity the limiting distribution is a point mass at μ. The Von Mises distribution is unimodal.

Figure 1 contains angular histograms that display four examples of simulated data from the Von Mises distribution obtained using the `rvonmises` function within the `circular` package (Agostinelli and Lund, 2013) in R (R Core Team, 2015). Angular histograms, are constructed the same way as a typical histogram, except that it is displayed on a 360 degree circle. Mutually exclusive and exhaustive bins, often of equal length, are defined for the the range of the variable and counts of observations in each bin are found and displayed. The histograms in Figure 1 illustrate the effect of various κ values. The angular



(a) $\kappa = .5$ (b) $\kappa = 1$ (c) $\kappa = 20$ (d) $\kappa = 50$

Figure 1: Simulated data from different Von Mises distributions.

histograms consist of 100 simulated observations, with $\mu = 0$ and κ =.5, 1, 20, and 50 going in sequence from left to right. It is pretty clear that as the concentration parameter is being increased, the spread of the observations is progressively narrowing. With $\kappa = 50$, all observations would appear to fall within approximately a 45 degree range, while with $\kappa = 1$ and especially when equal to .5, the distribution is spread around the entire circle.

In this paper, the primary use of the Von Mises distribution is enhancing visualization of the data through non-parametric density functions. Non-parametric kernel density curves work by looking at a selected observation and calculating the density centered at that observation for a given probability distribution, in this case the Von Mises distribution. Densities are obtained centered at all observations. The density curves are added and scaled to integrate to 1 as the non-parametric density estimator. The main challenge in applying this technique is selecting the spread or bandwidth for the individual densities. We will use visual assessment to select a bandwidth.

Three common circular summary statistics are the mean direction, mean resultant length, and circular variance. The mean direction coordinates are given by $\text{Cos}(\bar{\theta}) = \frac{C}{R}$ and $\text{Sin}(\bar{\theta}) = \frac{S}{R}$, where $C = \sum_{i=1}^{n} \text{Cos}(\theta_i)$, $S = \sum_{i=1}^{n} \text{Sin}(\theta_i)$, with $R^2 = C^2 + S^2$. The mean direction results from vector addition of observations (Fisher, 1993). The mean resultant length is $\bar{R} = \frac{R}{n}$, and represents the mean length of the vector resultant from calculating the mean direction. $\bar{R}$ is not necessarily a useful measure of dispersion, particularly if there are multiple groups present in the data (Fisher, 1993). Lastly, the circular variance is defined as $V = 1 - \bar{R}$. In Table 1 these summary statistics are displayed for the simulated data from Figure 1.

	A	B	C	D
Mean Direction	19.58	-1.12	0.04	1.18
Mean Resultant Length	0.28	0.46	0.98	0.99
Circular Variance	0.72	0.54	0.02	0.01

Table 1: Summary Statistics of simulated data from Figure 1.

3 An Example of Circular Data

3.1 Cape Blanco Data Summary Measures & Visualization

The Cape Blanco data set contains 743 observations. This contains hourly average values for the entire month of January 1991. Temperature is measured in Fahrenheit, Wind Speed is measured as miles per hour (MPH), and Wind Direction is measured in degrees. The hourly average values of wind direction may be being taken using the conventional arithmetic mean. As discussed previously, this could create unusual results when the wind orientation approaches 0° or 360° and they generate average directions for the hour from higher time resolution direction observations. Also, it should be noted that wind direction may be poorly measured and even defined when the observed wind speed is 0 MPH.

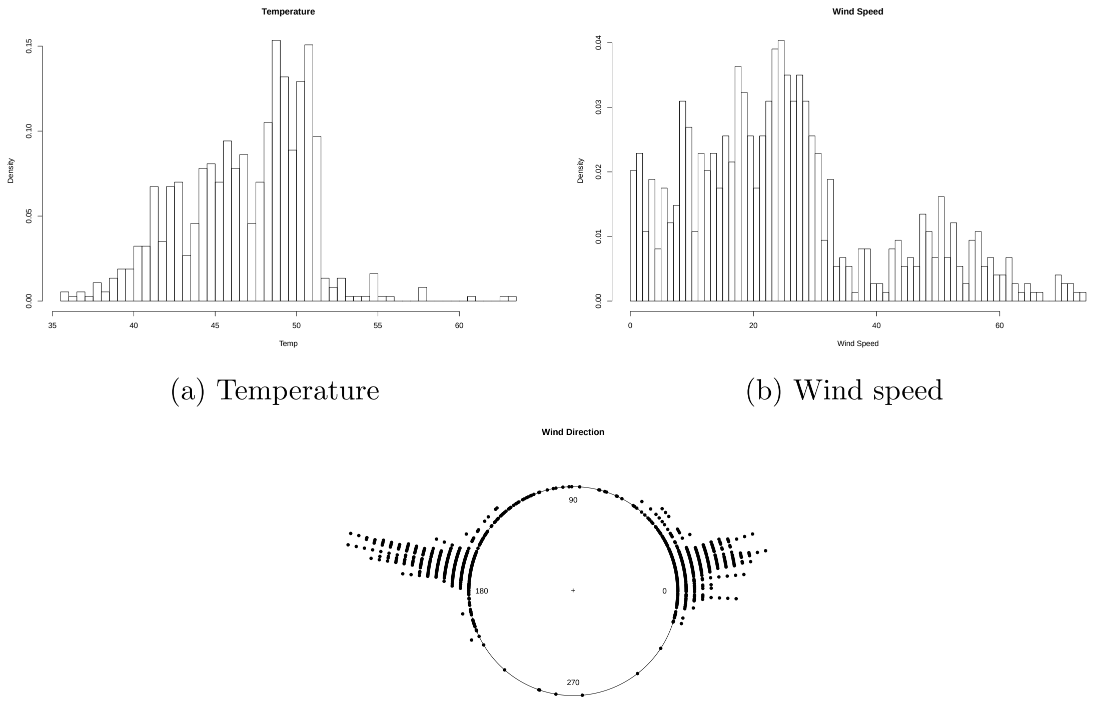
In Tables 2 & 3, summary measures of the Cape Blanco data set are provided. Figure 2 provides univariate summaries of the three variables with histograms in (a) and (b), while (c) is an angular histogram. Temperature seems to be a unimodal distribution with few observations above 52° F. Wind Speed appears to be a bi-modal distribution with peaks around 25 and 50 mph. Most of the observations fall in the lower wind speed group. There is no information provided with the data set regarding the definitions of degrees and rotation direction relative to East, South, North, and West. We chose to treat 0° as north, 90° as west, 180° as south and 270° as east. From subject matter knowledge combined with the plots of Figures 6, 10, and 11 (discussed later), we feel fairly confident about 0° and 180° as being north and south respectively, but are less certain with regards to east and west angle assignments (the rotation direction). The plots of Figure 2 provide univariate information about each of the three variables, but do not provide any information about relationships among the three variables.

	Temperature (F)	Wind Speed (mph)
Min.	35.66	0.00
1st Qu.	44.31	13.39
Median	47.54	23.14
Mean	46.87	25.09
3rd Qu.	49.76	30.92
Max.	63.10	73.95

Table 2: Cape Blanco summary statistics for temperature and wind speed.

	Wind Direction (Deg)
Mean Direction	84.23
Mean Resultant Length	0.23
Circular Variance	0.77

Table 3: Cape Blanco wind direction summary statistics.



(a) Temperature

(b) Wind speed

(c) Wind direction

Figure 2: Cape Blanco univariate summary plots.

The angular histogram in Figure 2(c) is helpful for visualizing where the observations are located on the circle. As mentioned above, the Von Mises distribution is a unimodal distribution. The wind data would appear to be bimodal so do not appear to follow a von Mises distribution when considered all together.

Another visual aid is to include a nonparametric density estimator with the angular histogram. By using a non-parametric density curve, multi-modal estimated densities are

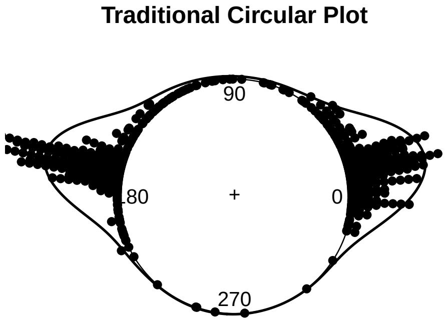


Figure 3: Circular density curve with angular histogram for Cape Blanco wind directions.

possible. To use the nonparametric density estimator, one has to select the bandwidth (window size) to use. For the Von Mises distribution, the bandwidth is the concentration parameter. This can be implemented in R using the `density.circular` function found in the `circular` package (Agostinelli and Lund, 2013). There are no hard and fast rules for choosing the parameter, and it is largely a judgement call. The bandwidth was selected by visual inspection, comparing the density curve for various bandwidths to the angular histogram. A bandwidth of 20 seemed to offer a reasonably good representation of the data. This choice can also be impacted by the amount of separation used in displaying the dots in the angular histogram. In Figure 3, the results of adding a density curve to the plot from Figure 2(c) can be seen. It reinforces the interpretation of a bimodal distribution of wind directions. The use of density curves will be revisited later.

3.2 Quantitative versus Circular Scatterplot

A potential method for visualizing the relationship between a circular and quantitative variable would be with a scatterplot. To avoid the boundary issues at 0° and 360°, data at the edges can be plotted twice, showing observations at $-180°$ and up to $540°$ as suggested by Fisher (1993). In Figure 4, an example of this plot is provided using the Wind Speed and Wind Direction variables from the Cape Blanco data set. The plot is problematic, as it is still difficult to grasp the relationship between Wind Speed and Wind Direction from this plot. Other disadvantages are that observations are plotted twice and the plot doesn't extend to higher dimensional displays. Another solution is needed.

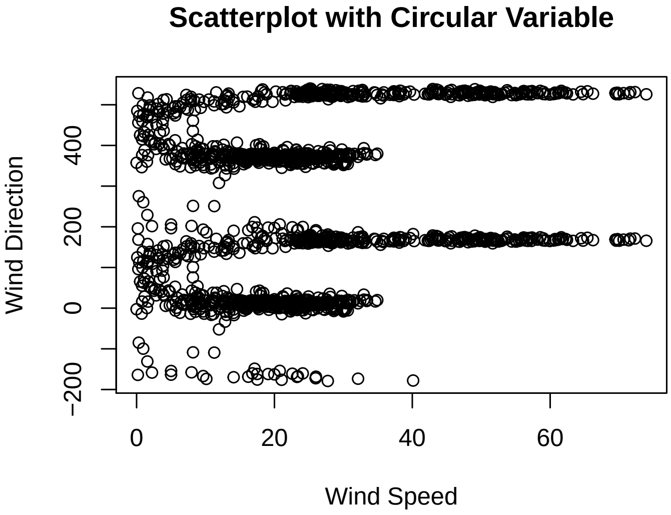


Figure 4: Scatterplot of wind direction and wind speed for Cape Blanco data.

3.3 Parallel Coordinate Plot

A parallel coordinate plot (PCP) is a technique for the visualization of multivariate quantitative data. All variables are scaled to have a maximum of 1 and and minimum of 0. The variables are located equidistant apart on the horizontal axis, and connecting lines are drawn for each observation across all of the variables. This enables the visualization of relationships across multiple variables, and can be helpful for detecting sub-groups (Hardle and Simar, 2012).

Figure 5 is a parallel coordinate plot of the Cape Blanco data that treats wind direction as a regular continuous quantitative variable. When looking at wind direction in Figure 5, observations near the top and bottom appear very different. However, this is not the the case as values of 0 and 1 are in fact the same. After rescaling for making a parallel coordinate plot, 0 and 1 represent 0 and 360 degrees respectively, which both mean the same thing in terms of degrees. Thus, it is clearly evident that plotting a circular variable as a traditional quantitative variable in a PCP is not a viable solution. A solution will be

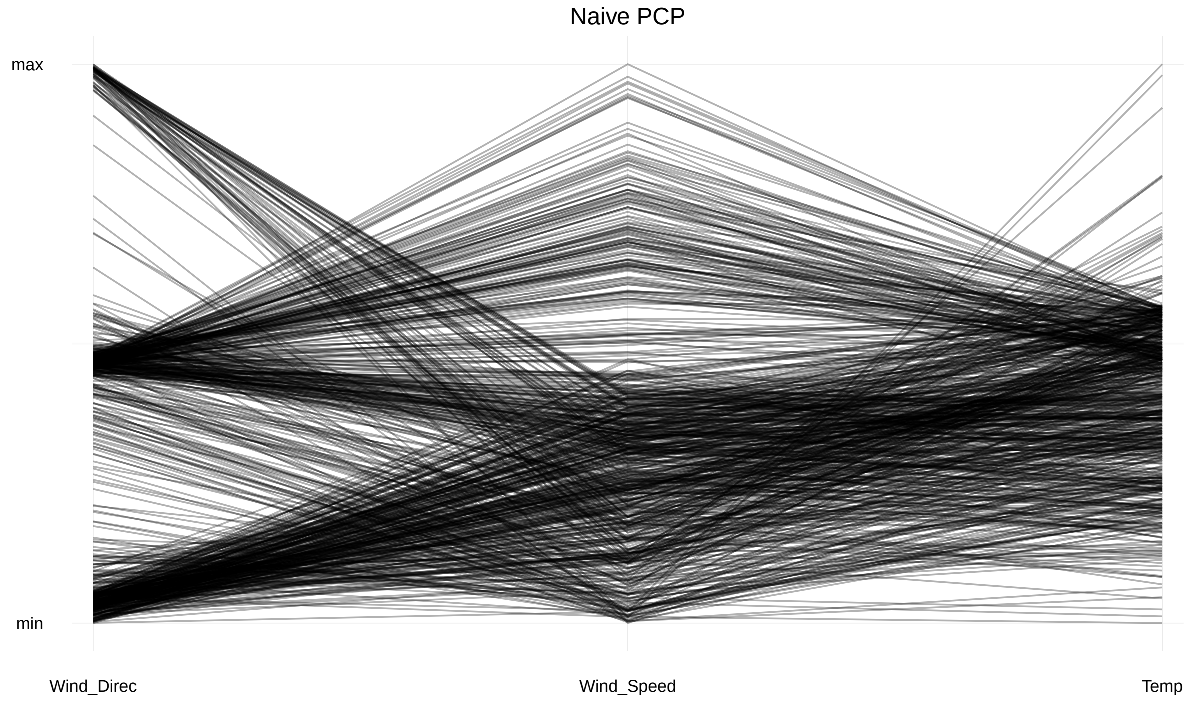


Figure 5: Naive parallel coordinate plot of Cape Blanco data.

proposed in the next section.

4 Visualizing Multivariate Data with a Circular Variable

The parallel coordinate plot in Figure 5 accurately represents the relationship between temperature and wind speed but the plot did not correctly depict wind direction. In Figure 2(c), an angular histogram of wind direction was displayed. This plot nicely represented the wind direction variable. From looking at Figure 2(c) it is apparent that using a circle is very beneficial for displaying circular data, and would appear to be a desirable characteristic to include in a multivariate plot. The question is whether it would be possible to use a circle to represent a circular variable within a PCP. We will use the Cape Blanco data to illustrate that this is possible and explore its utility.

With wind speed and temperature scaled to have a range of 1, the height of each part of the plot is one. While a PCP does not plot an x-axis grid, there is in fact a grid being used for the creation of the plot. Three variables are being plotted, and a default PCP uses equal spacing between each variable. For the modified PCP, the underlying x-axis is set to have a range of -0.5 to 7, so that the traditional quantitative variables could be plotted at 3.5 and 7, while the circular variable is centered at 0 on the x-axis. The actual values used for the range are not important, but the relative placement of variables with regard to one another is important. The particular values used here were chosen as they worked well given other scaling choices. A circle of radius 0.5 was used for the the circular variable as this maximized the available y-axis range of the plot. The underlying dimensions of the plot are 7.5 x 1. When constructing the plot, the circle will appear as an ellipse. An aspect ratio of 1 would make the representation of the circular variable accurate as a circle, but would come at the cost of being unable to use most of the available area of the plot and this would make the visualization of relationships across variables difficult. The mechanics of the parallel coordinate plot remains the same in that observations are plotted by variable, with a connecting line used to denote each observation across the variables. Figure 6 displays this modified parallel coordinate plot.

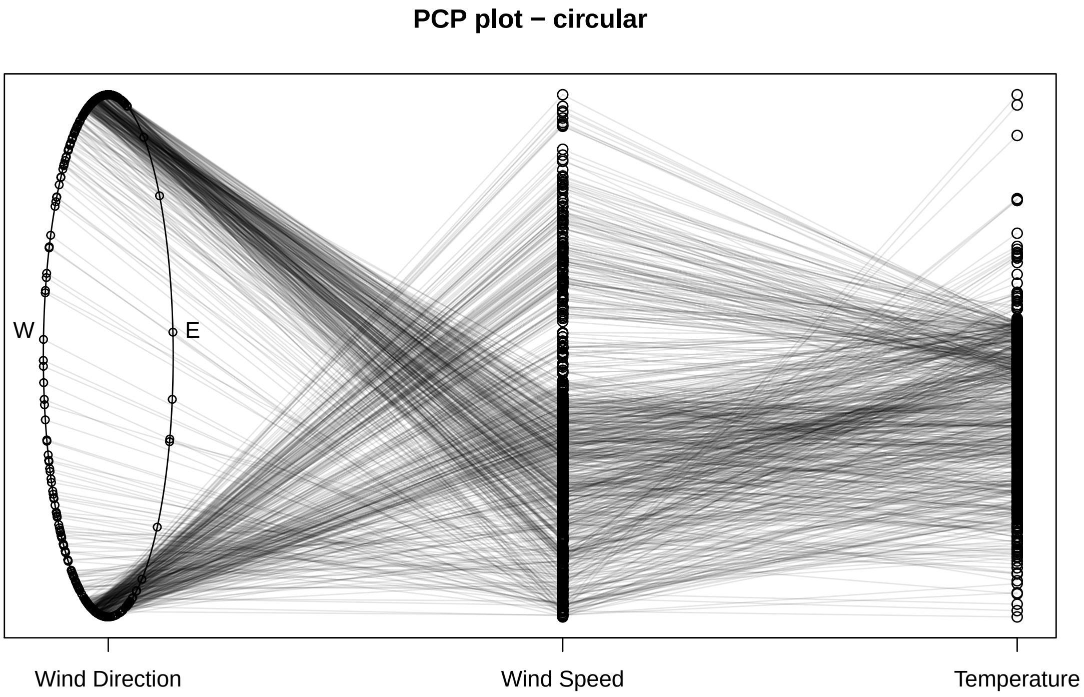


Figure 6: Circular Parallel Coordinate Plot.

With the ellipse, the connecting lines to the next variable from the top and bottom of the ellipse are fairly parallel and tightly bunched, as can be seen in Figure 6. This can make it difficult to discern whether the observed density at the top and bottom of the ellipse is truly indicating higher density at those points or an artifact of using a elliptical representation. The density curve discussed in Section 3.1 is useful for determining whether higher density levels are in fact being observed at the top and bottom of the ellipse.

It is also useful to limit the number of variables included in the circular PCP plot. If many variables are included, the pinching effect at the top and bottom of the ellipse becomes worse. The current opinion of the author is that having 3 to 4 variables in a circular PCP is a maximum for practical use. But the physical size of the plot is important for determining a maximum number of variables to include. If the plot was being displayed on a very large window, more variables could be included. Some of these issues discussed here are not that noticeable in this particular plot, but can become more acute depending on the pattern of the observations across the variables.

Figure 7 is a circular PCP of the Cape Blanco data, with a density curve added for the wind direction variable. With the density curve added, it is clear that the bulk of the observations do in fact lie around 0 and 180 degrees. (Note, the density curve looks slightly different here than the single density curve from Figure 3 as 0° is located at the top of the circle in this case rather on the right side, and plotted around what appears to be an ellipse instead of a circle.) When adding the density curve it is necessary to adjust the axis limits on the plot to fit the density plot curve on the plot. This allows a choice for the traditional quantitative variables that are included on the plot. In order to fit the density curve, 0.5 was added to the range on the top and bottom of the plot so that the y-axis now goes from -.5 to 1.5, with a resulting total range of two units. With the height of the plot being two units, it was thought reasonable to also expand the range of the traditional quantitative variables to be two units.

PCPs are a comparison of relative positions, so shifting the scale as shown in Figure 7 doesn't impact the ability to see patterns and relationships. Now the connecting lines go outward from the ellipse to the next variable over, but this appears to still effectively convey relationships while utilizing the full area of the plot. It would also be possible to continue to have the height of the regular quantitative variables match the major axis height of the ellipse in the display. This would more closely match the spirit of the conventional PCP but information may not be as easily extracted with this relative scaling. Whether adding the density curve produced a superior plot compared to the circular PCP without the density curve is hard to say. In this case, both plots seem to effectively convey information. The benefit of either approach would appear to depend largely on the particular data set being worked with.

As with any PCP, the order of the variables is important in the visualization of the relationships among the variables. This is even more critical as the number of variables displayed grows. Making the PCP with different variables as neighbors or with the circular variable shifting between central and edge locations could help to understand different pairwise relationships better.

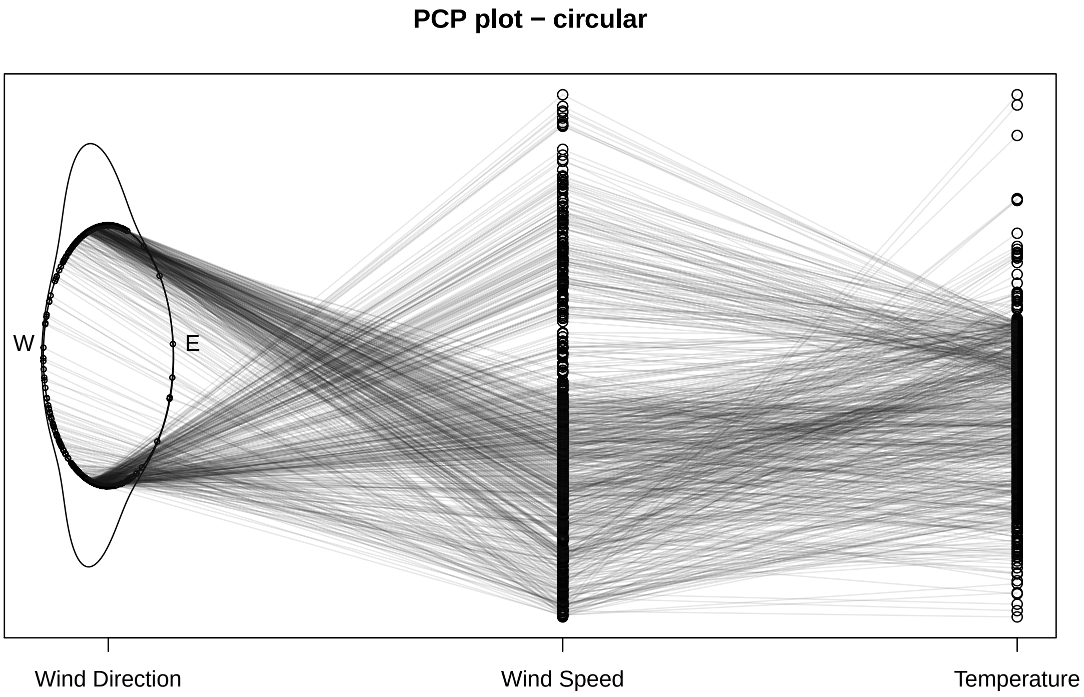


Figure 7: Circular PCP with density curve.

5 Analyzing Multivariate Data with Circular Variables

Some traditional statistical analysis tools are available when circular variables are present in a data set. For example, there is a framework for doing circular - linear regression (Fisher, 1993) for which some functionality is available in the `circular` (Agostinelli and Lund, 2013) package within R. Another approach is clustering to look for groups of observations. This approach is explored here, incorporating relationships among two regular quantitative variables and a circular one.

5.1 Disimilarity Matrix

When looking at a PCP, one of the goals is to look for patterns or groups of observations among the variables. We can formalize this exploration using cluster analysis, which is most easily performed using a distance or dissimilarity matrix. A dissimilarity matrix is made up of the pairwise differences among the observations. If a dissimilarity measure is always greater than or equal to 0, the dissimilarity between two identical points is 0, it is symmetric, and it satisfies the triangle equality, then it is a distance metric (Wikipedia, 2016a). A distance matrix is made from a dissimilarity measure that satisfies the above criteria. We will use Gower's distance as a starting point for our dissimilarity measure that incorporates a circular variable.

Gower's coefficient is a non-Euclidean metric that can be used on various variable types (Gower, 1971), including quantitative, categorical, and, as explained here circular variables. For a single quantitative variable q, Gower's similarity between observation i and j on variable x_q is measured as

$$s_{ijq} = 1 - \frac{|x_{iq} - x_{jq}|}{max|x_{iq} - x_{iq}|},$$

where $max|x_{iq} - x_{jq}|$ is the maximum difference. For a single variable variable q, Gower's dissimilarity, d_{ijq} is calculated as

$$d_{ijq} = \frac{|x_{iq} - x_{jq}|}{max|x_{iq} - x_{iq}|}.$$

Gower's dissimilarity over all Q variables is calculated as $d_{ij} = \frac{1}{Q}\sum_{q=1}^{Q} d_{ijq}$. As can be seen in the dissimilarity formula, Gower's distance scales comparisons among observations i and j to be between 0 and 1 where 0 is for no difference and 1 is for the maximum distance. It then averages these differences across all the Q dimensions being compared. Any time one uses Gower's dissimilarity, there is a choice in the denominator of whether to use the maximum distance observed or the maximum possible distance. For the Cape Blanco data, the maximum observed distance is used. For Temperature and Wind Speed, the maximum possible difference is unknown, making the maximum observed range the only choice. The decision for wind direction took more consideration.

When looking at a circle, distance could be measured in either direction around a circle. The minimum distance around the circle between observations makes sense in most cases. This involves finding the distance in both directions and using the smaller of the values. The largest possible difference one can observe is then 180 degrees. The choice of using the largest observed or possible difference for the Cape Blanco wind directions didn't matter as observations were observed in all areas of the circle, and both maximum the possible distance and maximum observed distance were 180 degrees. For other situations where observations are only obtained in part of the circle the maximum possible difference could be constrained further. The denominator is then 180 for calculating the contribution to the distance for the wind direction variable. It should be noted here that being consistent with the choice of using maximum observed or possible distance across all variables is recommended.

Once the dissimilarity for the individual variables are available, they can be combined into Gower's overall dissimilarity. For the Cape Blanco data with three variables, this is done as

$$d_{ij} = \frac{|Temp_i - Temp_j|}{max|Temp_i - Temp_j|} + \frac{|Speed_i - Speed_j|}{max|Speed_i - Speed_j|} + \frac{min(|Direc_i - Direc_j|, (360 - |Direc_i - Direc_j|))}{180}.$$

We chose to not rescale the distances by Q as this will not impact the clustering that follows. Dividing by the number of variables used is most necessary when Gower's is used for partially observed responses.

For implementing a dissimilarity metric with a circular variable, there are two options. We created a function for implementing Gower's dissimilarity metric as described above, that is provided in the appendix. There is also implementation of Gower's general coefficient in the `ade4` (Dray et al., 2007) package that permits circular variables. Our function is built to use degrees for the circular measurement while the `ade4` version requires the circular data in radians. The `ade4` implementation for a circular variable through the `dist.ktab` function uses a slightly different approach. From looking in the source code, it would appear to be working in the following manner:

1. Divide all values by the maximum observed difference, generating $x^*_{iq} = \frac{x_{iq}}{max|x_{iq}-x_{jq}|}$ and $x^*_{jq} = \frac{x_{jq}}{max|x_{iq}-x_{jq}|}$.

2. Calculate the dissimilarity with,

$$d_{ij} = \sqrt{min(|x^*_{iq} - x^*_{jq}|, 1 - |x^*_{iq} - x^q_{jq}|) * 2}.$$

The `dist.ktab` function has a maximum possible value of $\sqrt{2}$ while our function has 1 as the maximum possible value for a circular variable. For a single quantitative variable, the Gower's dissimilarity formula is used in the ade4 version.

When working through a simple example for 5 wind direction observations, the `dist.ktab` function produced results that were different from expected and what our function produced. We think it possible the `dist.ktab` function is doing an internal sorting of the variables, which makes comparison of distance matrices and use of the `dist.ktab` matrix problematic. We used our implementation of Gower's dissimilarity for all analyses in this paper.

5.2 Clustering

Clustering represents the use of quantitative methods to find groups of observations that are alike and different from other groups of observations (Everitt and Hothorn, 2011). Having created a distance matrix that includes the circular variable, it is now possible to look for clusters. Ward's hierarchical agglomerative clustering method will be used as implemented in the `hclust` function within R. As a bottom-up hierarchical agglomeration method, initially individual observations each make up a cluster. Then for each step, all possible pairs of clusters are evaluated with the cluster that results in the smallest increase in error sums of squares is used (Everitt and Hothorn, 2011). When using circular

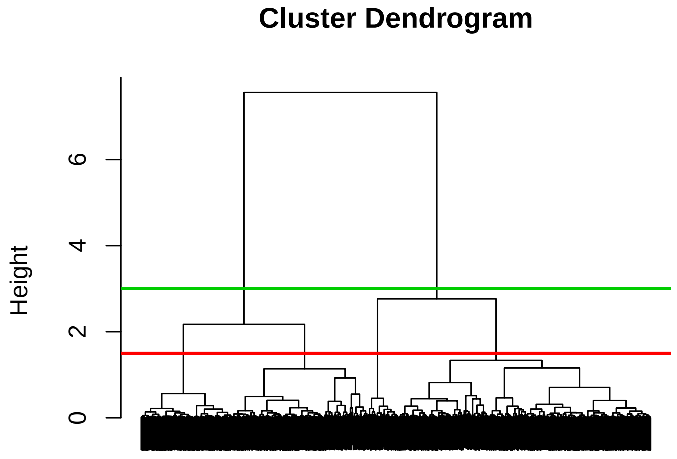


Figure 8: Dendrogram of cluster solution.

variables and Gower's dissimilarity matrix, the distance measurement is non-Euclidean, so it is not precisely the error sums of squares being measured but rather a pseudo-error sums of squares. This process is then repeated over and over, looking at what additional clusters can be combined to minimize the increase in pseudo-error sums of squares until all observations are in one large cluster. One drawback of hierarchical clustering is that once observations are assigned to a cluster, they can't be reassigned to a different cluster.

A good starting point for deciding upon the appropriate number of clusters to use is looking at the dendrogram. The height in the dendrogram is a relative measure of distance between clusters, and the number and composition of the clusters is chosen by drawing a horizontal line across the plot. However many vertical lines the horizontal line intersects determines the number of clusters. Determining where to draw the horizontal line is not necessarily clear cut. The main idea is that when the vertical distance between splitting a group into more clusters is small in magnitude, one probably doesn't believe that the additional clusters are truly present, and the horizontal line for the number of clusters should be above this point.

When looking at the dendrogram for the Cape Blanco data in Figure 8, it appears like there are probably either 2 or 4 clusters, as represented by the green and red lines on the

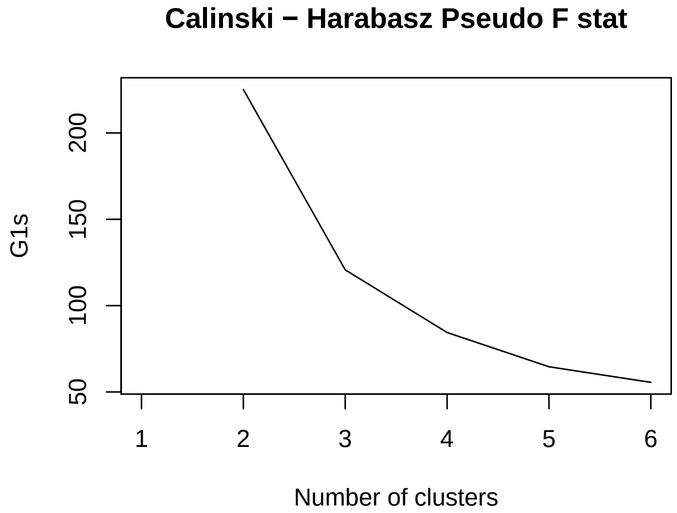


Figure 9: G1 measure.

plot. The vertical distance between two and four clusters is less than the distance from one to two clusters, but it still seems like four clusters might be a viable solution. There does not appear to be any clear evidence for more than four clusters.

Alternatively, one can use a measure called the Calinski - Harabasz Pseudo F-statistic available in the clusterSim package (Walesiak and Dudek, 2015) in R. It is calculated as

$$G1 = \frac{SS_B * (N - k)}{SS_W * (k - 1)},$$

where SS_B is the between cluster sums of squares, SS_W is the within cluster sums of squares, N is the total number of observations and k is the number of clusters (Calinski and Harabasz, 1974). This measure looks at the ratio of between clusters sums of squares relative to within cluster sums of squares across different numbers of clusters (k). Although pseudo sums of squares are being used in this case, it would seem a reasonable measure to look at given the use of Ward's method. While this is a pseudo measure with this dissimilarity, the ratio looks very similar to an F-statistic from an ANOVA procedure.

The largest value of G1 (ratio of between cluster variability over within cluster variability) is considered to be the best choice for number of clusters from this measure. In Figure 9, G1 is maximized at two clusters. The possible choice of there being one cluster is included, but there is no measured value for only having one cluster as that calculation is not possible. Thus, if this method suggests that two clusters should be used, as it does here, one does not know if that is correct or whether the optimal solution is really to have only one cluster (Tran and Greenwood, 2015).

Subject matter knowledge is also very important when determining what an appropriate number of clusters should be. This holds for the Cape Blanco data set where local knowledge suggests that there are just a few "types" of weather patterns. Figures 10 and 11 are parallel coordinate plots of the 2 and 4 cluster solutions respectively. These solutions are discussed further in Section 5.4.

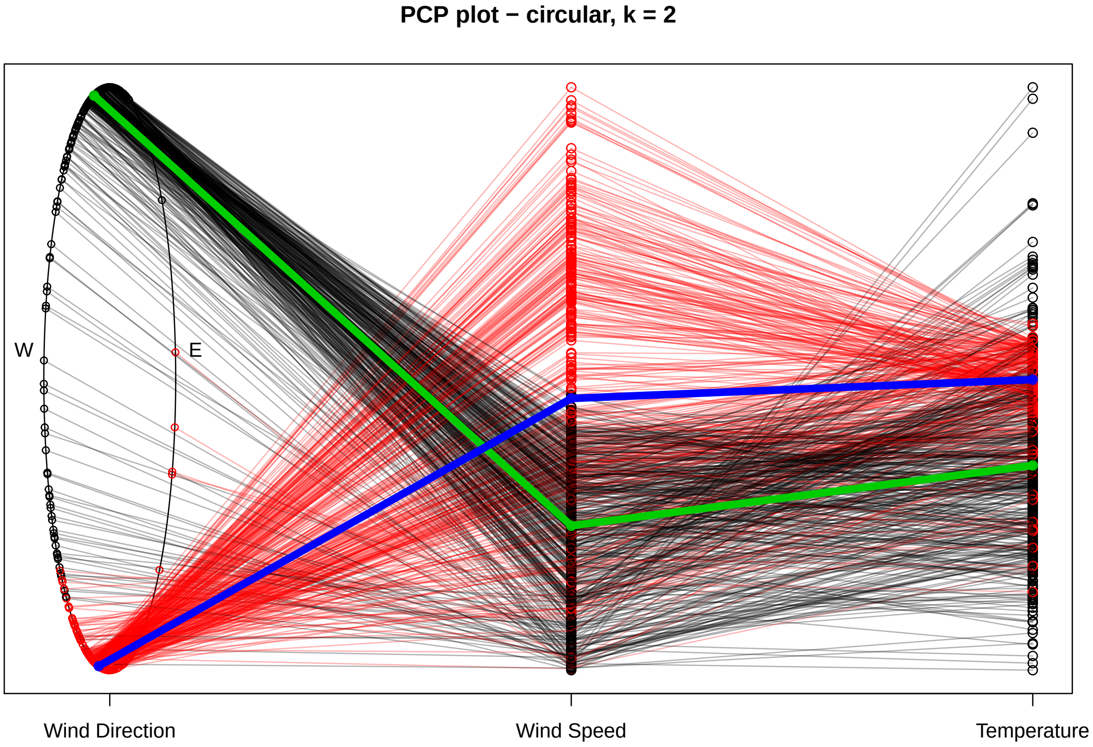


Figure 10: Two cluster solution for Cape Blanco data with proposed Gower's distance and Ward's method.

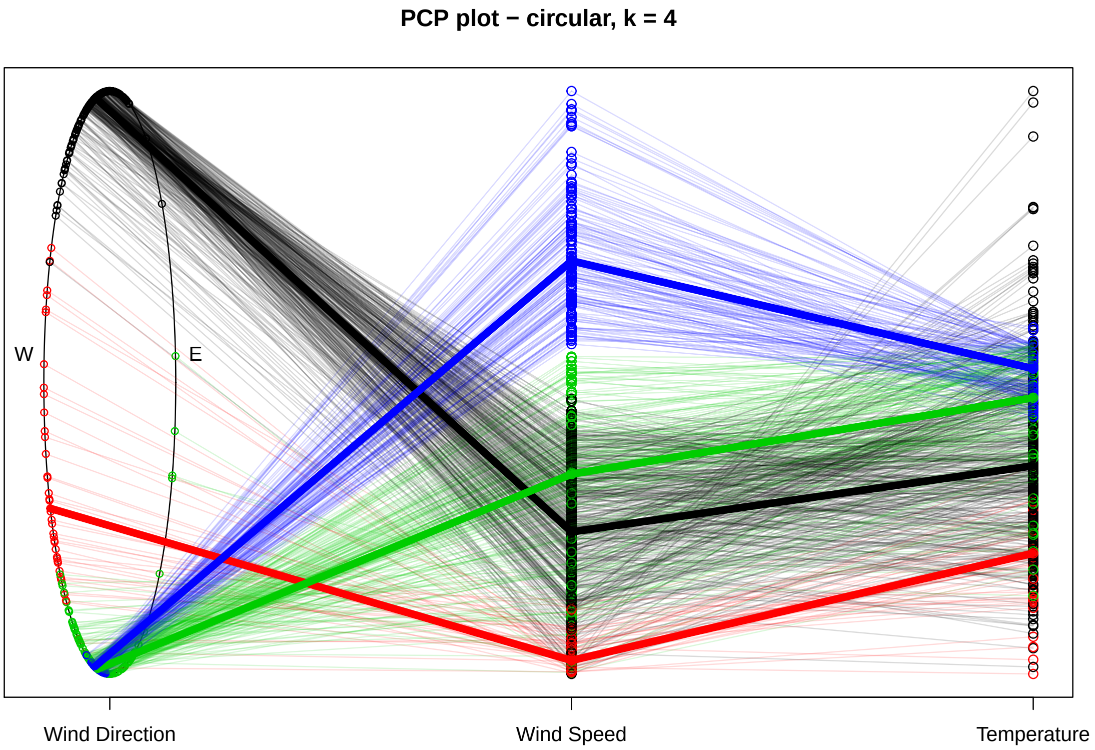


Figure 11: Four cluster solution for Cape Blanco data with proposed Gower's distance and Ward's method.

5.3 Medoids

It can be useful for visualization and understanding of the cluster solution to highlight an individual observation that represents each cluster. A medoid of a cluster is an observation that has the smallest average dissimilarity with all of the other observations in the cluster (Kaufman and Rousseeuw, 1990). The medoid for each cluster can then be used to succinctly represent a cluster. The medoids for each cluster are the highlighted observations on Figures 10 and 11.

Tables 4 and 5 show the group representative observations for the two and four cluster solutions. Cluster names are also included, and will be discussed in the following section.

	Observation ID	Temp	Wind_Direc	Wind_Speed
North Wind	507	45.30	13.74	18.33
South Wind	199	49.34	170.40	34.48

Table 4: Two cluster solution medoids for Cape Blanco data using proposed Gower's distance and Ward's method.

	Observation ID	Temp	Wind_Direc	Wind_Speed
North Wind 2.0	357	45.46	12.46	18.02
Cold & Calm	50	41.34	115.60	1.71
Moderation	163	48.66	169.40	25.35
Gusty	239	50.03	167.00	52.40

Table 5: Four cluster solution medoids for Cape Blanco data using proposed Gower's distance and Ward's method.

5.4 Cluster Results

5.4.1 Two Cluster Solution

With wind direction seemingly being the driver for the two cluster solution (Figure 10), these two clusters will be called the North Wind & South Wind clusters. Table 6 shows the group means for the two cluster solution, using a circular mean for Wind Direction and traditional arithmetic mean for Temperature and Wind Speed. Looking at Figure 10 and Table 6, along with the cluster medoids (Table 4) , we see that the North Wind cluster has substantially lower wind speeds and slightly lower Temperatures than the South Wind cluster.

	Temp	Wind_Speed	Wind Direction
North Wind	45.28	16.73	18.72
South Wind	48.86	35.50	169.12

Table 6: Two Cluster Solution Group Means for Cape Blanco data using proposed Gower's distance and Ward's method

5.4.2 Four Cluster Solution

From the dendrogram (Figure 8) and because we are using a hierarchical clustering algorithm, we know that the four cluster solution here is created by breaking apart each cluster from the two cluster solution discussed in the previous section. This helps with interpretation of this more complicated solution. Table 7 provides the variable means by cluster for the four cluster solution. The Temperature and Wind Speed means represent traditional arithmetic means, while the circular mean is used for Wind Direction.

	Temp	Wind_Speed	Wind Direction
North Wind 2.0	45.81	18.33	12.35
Cold & Calm	40.91	3.34	115.90
Moderation	48.25	23.58	169.35
Gusty	49.76	53.26	168.78

Table 7: Four Cluster Solution Group Means for Cape Blnaco data using proposed Gower's distance and Ward's method.

Looking at the four cluster summaries (Figure 11, Tables 5 & 7), we see one cluster with a strong nexus towards a northern wind direction. This cluster will be called North Wind 2.0. The observations in this cluster are a subset of the observations from the original North Wind cluster of the two cluster solution. The remaining observations from the original North Wind cluster tended to have a western wind direction, along with very low wind speeds and the lowest average temperatures of any cluster. This cluster solution will be called Cold & Calm. The other two clusters of the four cluster solution represent the division of the original South Wind cluster. In Figure 11, we see that all of the highest wind speeds have been grouped into a single cluster (blue observations on Figure 11). This would appear to be a defining feature of the cluster, and this cluster will called Gusty, to represent having the highest wind speeds, or possibly gusts. The remaining observations of the the South Wind cluster from the two cluster solution tend to have moderate wind speed and average to slightly above average temperatures. As a single unique feature does not stand out about this cluster, it will be called Moderation.

5.4.3 Preferred Cluster Solution

From personal knowledge having grown up on the Oregon coast, I think of there being two largely dominant winter weather patterns. It is unknown exactly how accurate this personal recollection is. Nonetheless, the author would tend go with a two cluster situation based off of personal experience and the diagnostic tools seeming to back up that idea. Having said that, the author also found the four cluster solution interesting and illustrative of subtler patterns. The question would be whether the four cluster solution was overfitting based on a very particular month. The two cluster solution would seem to generally be an accurate, if not complete solution.

5.4.4 Scope of Inference

These observations are hourly average average values for the entire month of January, 1991. They are not a random sample, so inference should be limited to the sample, in this case, hourly averages for January, 1991. There is no random assignment, so all of the relationships found in the clustering solutions are associative in nature.

6 Discussion

Circular variables occur in many applications, and it is important to be able to incorporate them into multivariate statistical approaches. Dealing with a circular variable requires estimators specific to circular variables for correct estimates of centers and spread. For example, the conventional arithmetic average is incorrect for directional variables. It is possible that the circular variable we were analyzing here was generated as an hourly observation using the regular mean which would lead to a biased estimate of the mean direction if it was not correctly handled as a circular variable. For measuring distance between observations, we used a custom implementation of Gower's method to create a distance measure that includes a circular variable. This enabled clustering of observations with Ward's hierarchical agglomeration method.

The circular PCP was informative for visualizing multivariate data that includes a circular variable as seen with the Cape Blanco data. This was very useful for the visualization of our clustering solutions. In this case, there are largely two groups of wind directions. Wind either came out of the north or south. This was helpful for visualization using the plot, as the clusters were fairly clear. It is unclear how the plot would look if there was a large data set and no clear groupings on the circular variable. One aspect that would be interesting, but was outside the scope of this paper, would be how the plot would work for showing the relationship between two circular variables. This would potentially seem like an interesting plot, especially if made in 3D where it was possible for the user to rotate the plot, or even even rotate the circles in a 2D plot.

Overall, the circular parallel coordinate plot would appear to be useful tool in the statistical toolkit. It has limitations, but for visualizing multivariate data that include circular variables, it is currently the best known tool available.

References

Agostinelli, C. and Lund, U. (2013). R package `circular`: Circular statistics (version 0.4-7). `https://r-forge.r-project.org/projects/circular`.

Bonneville (1991). Historical meterological readings. `http://transmission.bpa.gov/business/operations/wind/MetData/monthly/CapeBlanco`.

Calinski, T. and Harabasz, J. (1974). *A dendrite method for cluster analysis*, volume 3. Communications in Statistics.

Dray, S., Dufour, A., and Chessel, D. (2007). The ade4 package-ii: Two-table and k-table methods. R News. R package ade4.

Everitt, B. and Hothorn, T. (2011). *An Introduction to Applied Multivariate Analysis with R*. Springer.

Fisher, N. I. (1993). *Statistical Analysis of Circular Data*. Cambridge University Press.

Gower, J. C. (1971). *A General Coefficient of Similarity and Some of Its Properties*, volume 27. International Biometric Society. pp. 857 - 871.

Greenwood, M. (2016). Statistics 537 course notes. Multi-Variate Course Notes.

Hardle, W. K. and Simar, L. (2012). *Applied Multivariate Statistical Analysis*. Springer, third edition.

Kaufman, L. and Rousseeuw, P. (1990). *Finding Groups in Data, An Introduction to Cluster Analysis*. John Wiley & Sons Inc., Hoboken, New Jersey.

Maechler, M., Rousseeuw, P., Struyf, A., Hubert, M., and Hornik, K. (2015). *cluster: Cluster Analysis Basics and Extensions*.

R Core Team (2015). R: A language and environment for statistical computing. `https://www.R-project.org`.

Schloerke, B., Crowley, J., Cook, D., Hofmann, H., Wickham, H., Briatte, F., Marbach, M., and Thoen, E. (2014). Ggally: Extension to ggplot2. `https://CRAN.R-project.org/package=GGally`. R package version 0.5.0.

Tran, T. and Greenwood, M. (2015). Choosing the number of clusters in monothetic clustering. JSM Proceedings, American Statistical Association.

Walesiak, M. and Dudek, A. (2015). *clusterSim: Searching for Optimal Clustering Procedure for a Data Set*. R package version 0.44-2.

Wickham, H. (2009). ggplot2: elegant graphics for data analysis. `http://had.co.nz/ggplot2/book`.

Wikipedia (2015). Von mises distribution — wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Von_Mises_distribution&oldid=694701474`. [Online; accessed 4-March-2016].

Wikipedia (2016a). Distance matrix — wikipedia, the free encyclopedia. [Online; accessed 5-April-2016].

Wikipedia (2016b). Level of measurement — wikipedia, the free encyclopedia. [Online; accessed 25-March-2016].

R Code Appendix

```
cape_blanco2 <- read.csv("CapeBlanco_1991_january.csv")
#alternatively, the data is available using
#cape_blanco2 <- read.csv("http://www.transmission.bpa.gov/business/
#operations/Wind/MetData/monthly/CapeBlanco/CapeBlanco_1991_01.csv")
#if using this route, it would be necessary to use one additional
#step of subsetting all observations outside of January, as January
#was the only month used
new_names <- c("Date", "Barometric", "Temp", "Wind_Direc",
                        "Wind_Speed", "std_wind_direc",
                        "std_wind_speed", "peak_speed")
names(cape_blanco2) <- new_names
wind_direc2 <- cape_blanco2$Wind_Direc
```

```
set.seed(08)
library(circular, quietly = T)
sim_dat0 <-  rvonmises(100, mu = circular(0), kappa = .5,
                        control.circular = list(units = "degrees"))
 plot(sim_dat0, stack = T, sep = .08, shrink = 1.5, main = "")

 sim_dat1 <-  rvonmises(100, mu = circular(0), kappa = 1,
                        control.circular = list(units = "degrees"))
 plot(sim_dat1, stack = T, sep = .08, shrink = 1.5, main = "")

 sim_dat2 <- rvonmises(100, mu = circular(0), kappa = 20,
                       control.circular = list(units = 'degrees'))
 plot(sim_dat2, stack = T, sep = .08, shrink = 1.5, main = "")

 sim_dat3 <- rvonmises(100, mu = circular(0), kappa = 50,
                       control.circular = list(units = 'degrees'))
plot(sim_dat3, stack = T, sep = .08, shrink = 1.5, main = "")
```

```
mean_0 <- mean(sim_dat0)
rho_0 <- rho.circular(sim_dat0)
var_0 <- var.circular(sim_dat0)
sum_0 <- data.frame(mean_0, rho_0, var_0)
#sum_0

mean_1 <- mean(sim_dat1)
rho_1 <- rho.circular(sim_dat1)
var_1 <- var.circular(sim_dat1)
```

```
sum_1 <- data.frame(mean_1, rho_1, var_1)
#sum_1

mean_2 <- mean(sim_dat2)
rho_2 <- rho.circular(sim_dat2)
var_2 <- var.circular(sim_dat2)
sum_2 <- data.frame(mean_2, rho_2, var_2)
#sum_2

mean_3 <- mean(sim_dat3)
rho_3 <- rho.circular(sim_dat3)
var_3 <- var.circular(sim_dat3)
sum_3 <- data.frame(mean_3, rho_3, var_3)
#sum_3

sum_dat <- data.frame(matrix(c(sum_0, sum_1, sum_2, sum_3), ncol = 4, byrow = F))
names(sum_dat) <- c("A", "B", "C", "D")
rownames(sum_dat) <- c("Mean Direction",  "Mean Resultant Length",
                       "Circular Variance")
print(xtable(sum_dat, caption = "Summary Statistics of simulated data from Figure 1."), f
      table.placement = "H")
```

```
temp1 <- summary(cape_blanco2$Temp)
wind_speed1 <- summary(cape_blanco2$Wind_Speed)
sum1 <- cbind(temp1, wind_speed1)
colnames(sum1) <- c("Temperature (F)", "Wind Speed (mph)")
print(xtable(sum1,
             caption =
               "Cape Blanco summary statistics for temperature and wind speed."),
      floating = T, table.placement = "H", size = '\\footnotesize')

direc_sum <- summary(circular(cape_blanco2$Wind_Direc, type = "angles",
                              units = "degrees"))

mean1 <- direc_sum[5]
var1 <- var.circular(circular(cape_blanco2$Wind_Direc, type = "angles",
                              units = "degrees"))
rho1 <- rho.circular(circular(cape_blanco2$Wind_Direc, type = "angles",
                              units = "degrees"))

dat1 <- t(data.frame(mean1, rho1, var1))
rownames(dat1) <- c("Mean Direction", "Mean Resultant Length", "Circular Variance")
colnames(dat1) <- "Wind Direction (Deg)"
print(xtable(dat1, caption = "Cape Blanco wind direction summary statistics."),
      floating = T, table.placement = "H", size = '\\footnotesize')
```

```
hist(cape_blanco2$Temp, breaks = 40, main = "Temperature", xlab = "Temp", freq = F)
hist(cape_blanco2$Wind_Speed, breaks = 60, main = "Wind Speed", xlab = "Wind Speed",
     freq = F)
library(circular, quietly = T)
direc3 <- circular(wind_direc2, units = "degrees", type = "angles")
plot(direc3, stack = T, sep = .08, shrink = 1.2, main = "Wind Direction")
```

```
less_180 <-which(cape_blanco2$Wind_Direc < 180, arr.ind = T)
greater_180 <- which(cape_blanco2$Wind_Direc > 180, arr.ind = T)

less_180_dat <- cape_blanco2[less_180,]
less_180_dat$Wind_Direc <- less_180_dat$Wind_Direc + 360

greater_180_dat <- cape_blanco2[greater_180,]
greater_180_dat$Wind_Direc <- greater_180_dat$Wind_Direc - 360

new_dat <- rbind(less_180_dat, cape_blanco2, greater_180_dat)

plot(Wind_Direc ~ Wind_Speed, data = new_dat, ylim = c(-180, 540),
     main = "Scatterplot with Circular Variable", ylab = "Wind Direction",
     xlab = "Wind Speed")
```

```
library(GGally)
ggparcoord(cape_blanco2, columns = c(4, 5, 3), scale = "uniminmax",
           alphaLines = .3) +
  theme_minimal() + scale_x_discrete(expand = c(.02, .02)) +
  ggtitle("Naive PCP") + xlab("") + ylab("")+
  theme(axis.ticks = element_blank()) +
  scale_y_continuous(breaks = c(0,1), labels=c("min", "max"))
# ggparcoord(cape_blanco2, columns = c(4, 5, 3), scale = "uniminmax",
#            alphaLines = .3) +
#   theme_minimal() + scale_x_discrete(expand = c(.02, .02)) +
#   ggtitle("Naive PCP plot") + xlab("") + ylab("")+
#   theme(axis.ticks = element_blank(), axis.text.y = element_blank()) +
#   scale_y_continuous(breaks = c(0,1), labels=c("min", "max"))
```

```
############################################
#work on circular plot
#set up data for making plot
#################################################
theta <- seq(0, 360, .01)
```

```
x1 <- cos(pi*theta/180)
y1 <- sin(pi*theta/180)
y1_stand <- y1/2 + .5
x1_stand <- x1/2
#dont make data into circular object
wind_rad <- pi*(wind_direc2)/180 + pi/2
wind_direc2
wind_rad
wind_speed <- cape_blanco2$Wind_Speed
wind_speed_stand <- wind_speed / max(wind_speed)
speed_vec <- rep(3.5, length(wind_speed_stand))
#barom_vec <- rep(5, length(wind_speed_stand))
temp_vec <- rep(7, length(wind_speed_stand))
temp_stand <- (cape_blanco2$Temp - min(cape_blanco2$Temp)) /
              (max(cape_blanco2$Temp) - min(cape_blanco2$Temp))

seg_dat <- data.frame(cbind((cos(wind_rad)/2), (sin(wind_rad)/2 + .5),
                            speed_vec, wind_speed_stand))
seg_dat1 <- data.frame(cbind(speed_vec, wind_speed_stand, temp_vec,
                             temp_stand))
```

```
#actually make the plot
plot(y1_stand ~ x1_stand, type = "l", xlim = c(-.5, 7), ylim = c(0, 1),
     xaxt = "none", xlab = "", yaxt = 'none', ylab = "", main = "PCP plot - circular")
labs <- c("Wind Direction", "Wind Speed", "Temperature")
axis(1,  at = c(0, 3.5, 7), labels = labs, las = 1)
#axis(2, at = c(0, 1), labels = c("Min", "Max"), las = 1)
#to angle labels, not really a fan
#text(c(0, 4, 7), par("usr")[3] - 0.15, labels = labs, srt = 45, pos = 1, xpd = TRUE)
points(cos(wind_rad)/2, sin(wind_rad)/2 + .5, cex = .75, col = 1)
points(wind_speed_stand ~ speed_vec, col = 1)
points(temp_stand ~ temp_vec, col = 1)
segments(seg_dat[,1], seg_dat[,2], seg_dat[,3], seg_dat[,4],
         col = adjustcolor(col = 1, alpha.f = .1))
segments(seg_dat1[,1], seg_dat1[,2], seg_dat1[,3], seg_dat1[,4],
         col = adjustcolor(col = 1, alpha.f = .1))
text( .65, .55, "E")
text(-.65, .55, "W")
```

```
library(circular, quietly = T)
direc3 <- circular(wind_direc2, units = "degrees", type = "angles")
#plot(density(direc3, kernel = "vonmises", bw = 20))
dens_circ <- density.circular(direc3, kernel = "vonmises", bw = 20)
```

```
p2 <- plot(density(direc3, kernel = "vonmises", bw = 20), zero = pi/2,
            rotation = 'counter', shrink = 1)

#never ended up using this
#rotate plot by 90 degrees
#angle <- -pi/2
#M <- matrix( c(cos(angle), -sin(angle), sin(angle), cos(angle)), 2, 2 )
#plot(as.matrix(data.frame(p2£x, p2£y)) %*% M)
```

```
#make circular plot
plot(y1_stand ~ x1_stand, type = "l", xlim = c(-.5, 7), ylim = c(-.5, 1.5),
     xaxt = "none", xlab = "", ylab = "", yaxt = 'none', main = "PCP plot - circular")
lines(I(p2$y/2 + .5) ~ I(p2$x/2))
axis(1,  at = c(0, 3.5, 7), labels = labs, las = 1)
#axis(2, at = c(0, 1), labels = c("Min", "Max"), las = 1)
points(cos(wind_rad)/2, sin(wind_rad)/2 + .5, cex = .5)
points(wind_speed_stand*2 -.5 ~ speed_vec)
points(temp_stand*2 -.5 ~ temp_vec)

segments(seg_dat[,1], seg_dat[,2], seg_dat[,3], seg_dat[,4]*2 -.5,
         col = rgb(.1, .1, .1, .1))
segments(seg_dat1[,1], seg_dat1[,2]*2 -.5, seg_dat1[,3], seg_dat1[,4]*2 - .5,
         col = rgb(.1, .1, .1, .1))
text( .65, .55, "E")
text(-.65, .55, "W")
```

```
# Inclusion of distance around a circle (in degrees?) in Gowers
xc <- cape_blanco2$Wind_Direc
#xr <- data.frame(scale(cape_blanco2£Temp), scale(cape_blanco2£Wind_Speed))
xr <- data.frame(cape_blanco2$Temp, cape_blanco2$Wind_Speed)
#Calculate Gower's first and multiply by number of variables considered, excluding the
#circular variable.
library(cluster)
#Put variables that aren't circular one into xr
d1<-as.dist(as.matrix(daisy(xr,"gower")))*dim(xr)[2]
# Times number of variables that aren't circular

circd <- function(x){
      #Assumes x is just a single variable
      dist1<-matrix(0,nrow=length(x),ncol=length(x))
      for (i in (1:(length(x)-1))){
          for (j in i:length(x)){
               dist1[j,i]=min(abs(x[i]-x[j]), (360 - abs(x[i]-x[j])))/180
          }
        }
     return(as.dist(dist1))
}
#circd(xc)
dc<-(d1+circd(xc))/(dim(xr)[2]+1)
#Divide by total number of variables (assumes no missing values)
```

```
#greenwood - circle only
circ_dist <- circd(cape_blanco2$Wind_Direc)
clust_circle <- hclust(circ_dist, method = 'ward.D2')
#plot(clust_circle)
cuts_circle <- factor(cutree(clust_circle, k = 2))

library(ade4)
speed <- data.frame(scale(cape_blanco2$Wind_Speed))
temp <- data.frame(scale(cape_blanco2$Temp))
direc <- data.frame(cape_blanco2$Wind_Direc)*(pi/180)
direc5 <- prep.circular(direc)
ktab1 <- ktab.list.df(list(speed, temp, direc5))
dist5 <- dist.ktab(ktab1, type = c("Q", "Q", "C"))
clust5 <- hclust(dist5, method = 'ward.D2')
#plot(clust5)
cuts5 <- cutree(clust5, k = 2)

#wind_direc only
ktab2 <- ktab.list.df(list(direc5))
dist2 <- dist.ktab(ktab2, type = 'C')

clus2 <- hclust(dist2, method = 'ward.D2')
cut2 <- cutree(clus2, k= 2)
```

```
clust_one <- hclust(dc, method = 'ward.D2')
#plot(clust_one)
cuts_2 <- factor(cutree(clust_one, k = 2))
cuts_4 <- factor(cutree(clust_one, k = 4))
cuts_3 <- factor(cutree(clust_one, k = 3))
```

```r
med <- function(members,Dist){
  if(length(members)==1){return(members)}
  else{
    if(length(members)==0){return(0)}
    dists<-apply(Dist[members,members],1,sum)
    medoid<-members[which(dists==min(dists))]
    return(medoid[1])
  }
}

ids <- 1:nrow(cape_blanco2)
#medoids 2 cluster solution
k_2_1 <- med(members = ids[cuts_2 == 1], Dist = as.matrix(dc))  #507
k_2_2 <-med(members = ids[cuts_2 == 2], Dist = as.matrix(dc)) #199
meds_2 <- c(k_2_1, k_2_2)
#medoids 4 cluster solution
k_4_1 <- med(members = ids[cuts_4 == 1], Dist = as.matrix(dc)) #357
k_4_2 <- med(members = ids[cuts_4 == 2], Dist = as.matrix(dc)) #50
k_4_3 <- med(members = ids[cuts_4 == 3], Dist = as.matrix(dc)) #163
k_4_4 <- med(members = ids[cuts_4 == 4], Dist = as.matrix(dc)) #239
meds_4 <- c(k_4_1, k_4_2, k_4_3, k_4_4)
 #Need to pass observation IDs that relate to the rows in the distance matrix
```

```r
par(mfrow = c(1,1))
old_par <- par(mar = c(5.1, 4.1, 4.1, 2.1))
par(mar= c(2, 4.1, 4.1, 2.1))
plot(clust_one, labels = F, xlab = "", sub = "")
abline(h = 3, lwd = 2, col = 3)
abline(h = 1.5, lwd = 2, col = 2)
par(mar = old_par)
```

```r
scale_cape <- data.frame(apply(cape_blanco2[, c(3, 5)], 2, scale), wind_direc2)
#names(cape_blanco2)
noscale_cape <- data.frame(cape_blanco2[,c(3,5) ], wind_direc2)
library(clusterSim)
G1s <- numeric(0)

for(j in 1:6){
  G1s[j] <- index.G1(x = noscale_cape, cl = cutree(clust_one, k = j))
}
plot(1:6, G1s, type = 'l', xlab = "Number of clusters",
     main = "Calinski - Harabasz Pseudo F stat")
```

```r
#greenwood method
plot(y1_stand ~ x1_stand, type = "l", xlim = c(-.5, 7), ylim = c(0, 1),
     xaxt = "none", xlab = "", ylab = "", yaxt = "none",
     main = "PCP plot - circular, k = 2")
labs <- c("Wind Direction", "Wind Speed", "Temperature")
axis(1,  at = c(0, 3.5, 7), labels = labs, las = 1)
#axis(2, at = c(0, 1), labels = c("Min", "Max"), las = 2)

points(cos(wind_rad)/2, sin(wind_rad)/2 + .5, cex = .75, col = cuts_2)
points(wind_speed_stand ~ speed_vec, col = cuts_2)
points(temp_stand ~ temp_vec, col = cuts_2)
segments(seg_dat[,1], seg_dat[,2], seg_dat[,3], seg_dat[,4],
         col = adjustcolor(col = cuts_2, alpha.f = .3))
segments(seg_dat1[,1], seg_dat1[,2], seg_dat1[,3], seg_dat1[,4],
         col = adjustcolor(col = cuts_2, alpha.f = .3))

#group reps
points(cos(wind_rad[c(507, 199)])/2, sin(wind_rad[c(507, 199)])/2 + .5,
       cex = 1, col = c(3, 4))
points(wind_speed_stand[c(507, 199)] ~ speed_vec[c(507, 199)], col = c(3, 4))
points(temp_stand[c(507, 199)] ~ temp_vec[c(507, 199)], col = c(3, 4))
segments(seg_dat[c(507, 199), 1], seg_dat[c(507, 199), 2],
         seg_dat[c(507, 199), 3], seg_dat[c(507, 199), 4],
         col = c(3, 4), lwd = 6)
segments(seg_dat1[c(507, 199), 1], seg_dat1[c(507, 199), 2],
         seg_dat1[c(507, 199), 3], seg_dat1[c(507, 199), 4],
         col = c(3, 4), lwd = 6)

text( .65, .55, "E")
text(-.65, .55, "W")


# #k = 3
# plot(y1_stand ~ x1_stand, type = "l", xlim = c(-.5, 7), ylim = c(0, 1),
#      xaxt = "none", xlab = "", ylab = "", main = "PCP plot - circular, k = 3")
# labs <- c("Wind Direction", "Wind Speed", "Temperature")
# axis(1,  at = c(0, 3.5, 7), labels = labs, las = 1)
#
# points(cos(wind_rad)/2, sin(wind_rad)/2 + .5, cex = .75, col = cuts_3)
# points(wind_speed_stand ~ speed_vec, col = cuts_3)
# points(temp_stand ~ temp_vec, col = cuts_3)
# segments(seg_dat[,1], seg_dat[,2], seg_dat[,3], seg_dat[,4],
#          col = adjustcolor(col = cuts_3, alpha.f = .3))
# segments(seg_dat1[,1], seg_dat1[,2], seg_dat1[,3], seg_dat1[,4],
#          col = adjustcolor(col = cuts_3, alpha.f = .3))
```

```
#k = 4
# plot(y1_stand ~ x1_stand, type = "l", xlim = c(-.5, 7), ylim = c(0, 1),
#      xaxt = "none", yaxt = "none", xlab = "", ylab = "",
#      main = "PCP plot - circular, k = 4")
# labs <- c("Wind Direction", "Wind Speed", "Temperature")
# axis(1,  at = c(0, 3.5, 7), labels = labs, las = 1)
# #axis(2, at = c(0, 1), labels = c("Min", "Max"), las = 2)
#
#
# points(cos(wind_rad)/2, sin(wind_rad)/2 + .5, cex = .75, col = cuts_4)
# points(wind_speed_stand ~ speed_vec, col = cuts_4)
# points(temp_stand ~ temp_vec, col = cuts_4)
# segments(seg_dat[,1], seg_dat[,2], seg_dat[,3], seg_dat[,4],
#          col = adjustcolor(col = cuts_4, alpha.f = .3))
# segments(seg_dat1[,1], seg_dat1[,2], seg_dat1[,3], seg_dat1[,4],
#          col = adjustcolor(col = cuts_4, alpha.f = .3))
#
# #group reps
# points(cos(wind_rad[meds_4])/2, sin(wind_rad[meds_4])/2 + .5,
#        cex = .75, col = c(5, 6, 7, 8))
# points(wind_speed_stand[meds_4] ~ speed_vec[meds_4], col = c(5, 6, 7, 8))
# points(temp_stand[meds_4] ~ temp_vec[meds_4], col = c(5, 6, 7, 8))
# segments(seg_dat[meds_4,1], seg_dat[meds_4,2],
#          seg_dat[meds_4,3], seg_dat[meds_4,4],
#          col = c(5, 6, 7, 8), lwd = 3)
# segments(seg_dat1[meds_4, 1], seg_dat1[meds_4, 2], seg_dat1[meds_4, 3],
#          seg_dat1[meds_4, 4],
#          col = c(5, 6, 7, 8), lwd = 3)
#
# text( .65, .55, "E")
# text(-.65, .55, "W")

#k = 4
plot(y1_stand ~ x1_stand, type = "l", xlim = c(-.5, 7), ylim = c(0, 1),
     xaxt = "none", yaxt = "none", xlab = "", ylab = "",
     main = "PCP plot - circular, k = 4")
labs <- c("Wind Direction", "Wind Speed", "Temperature")
axis(1,  at = c(0, 3.5, 7), labels = labs, las = 1)

points(cos(wind_rad)/2, sin(wind_rad)/2 + .5, cex = .75, col = cuts_4)
points(wind_speed_stand ~ speed_vec, col = cuts_4)
points(temp_stand ~ temp_vec, col = cuts_4)
segments(seg_dat[,1], seg_dat[,2], seg_dat[,3], seg_dat[,4],
         col = adjustcolor(col = cuts_4, alpha.f = .15))
segments(seg_dat1[,1], seg_dat1[,2], seg_dat1[,3], seg_dat1[,4],
         col = adjustcolor(col = cuts_4, alpha.f = .15))
```

```r
text( .65, .55, "E")
text(-.65, .55, "W")

#group reps
points(cos(wind_rad[meds_4])/2, sin(wind_rad[meds_4])/2 + .5,
       cex = .75, col = c(1, 2, 3, 4))
points(wind_speed_stand[meds_4] ~ speed_vec[meds_4], col = c(1, 2, 3, 4))
points(temp_stand[meds_4] ~ temp_vec[meds_4], col = c(1, 2, 3, 4))
segments(seg_dat[meds_4,1], seg_dat[meds_4,2],
         seg_dat[meds_4,3], seg_dat[meds_4,4],
         col = adjustcolor(col = c(1, 2, 3, 4), alpha = 1), lwd = 6)
segments(seg_dat1[meds_4, 1], seg_dat1[meds_4, 2], seg_dat1[meds_4, 3],
         seg_dat1[meds_4, 4],
         col = adjustcolor(col = c(1, 2, 3, 4), alpha = 1), lwd = 6)
```

```r
med2 <- cape_blanco2[meds_2, c(3:5)]
rownames(med2) <- c("North Wind", "South Wind")
med2 <- data.frame(meds_2, med2)
names(med2)[1] <- "Observation ID"
print(xtable(med2, caption = "Two cluster solution medoids for Cape Blanco data using pro
      floating = T, table.placement = "H")
```

```r
med4 <- cape_blanco2[meds_4, c(3:5)]
rownames(med4) <- c("North Wind 2.0", "Cold & Calm", "Moderation", "Gusty")
med4 <- data.frame(meds_4, med4)
names(med4)[1] <- "Observation ID"
print(xtable(med4, caption = "Four cluster solution medoids for Cape Blanco data using pr
      floating = T, table.placement = "H")
```