Cluster Analysis and Modeling

-- Team Invincibles

Motivation

There are 72 buoys measuring climate changes in the equatorial Pacific in our dataset. Among them, the buoy that recorded the most data has roughly 3000 rows of data in more than 10 years. Within the period of more than one decade, buoys traveled to different locations away from where they were initially planted. The latitude values stay within one degree from an approximate location for one buoy; however, the longitude values for one buoy sometimes are as far as five degrees away from its original position. Data collected from one buoy might account for discrepancy resulting from location changes, which poses challenges if we would like to investigate the change of climate at a location over time. In addition, if a buoy traveled to an approximate location where another buoy sat, it is reasonable to combine data collected by these two buoys together.

As a result, we would like to cluster data collected by buoys based on relative distance among data points. During this process, we will create clusters, and data points within a cluster are more close to each other and hopefully more consistent in factors apart from time than they were within a buoy before. We can use data points in a cluster later if we would like to investigate climate change over time.

However, aiming to re-group data that were previously divided by buoys and to remove location related noise from one cluster to investigate climate change over time at a location requires one hypothesis: data collected in neighboring locations were more related than that collected from regions far away, and factors apart from locations did not have as much influence in determining clustering. We bear this hypothesis in mind and will check its validity once the clustering results are generated. In fact, this hypothesis does not hold and as a result we explore other causes for the clustering results.

Last but not least, we are aware of computing powers clustering usually take. The dataset to record the distance among all data points in the El Nino generates a huge amount of data; clustering such an amount of data needs an extremely strong computing power. Instead, we take 200 data points from each of the 10 buoys with most data and congregate the 2000 data points into a sample dataset. By omitting the nulls, we have 1990 left. We then apply clustering methods to this sample dataset. Since this dataset is a random sample from the population, results and interpretation generated from clustering the sample can be extrapolated to the population.

After we find the optimal number of clusters for the sample dataset, we will label data points based on clusters and apply classification analysis to the dataset. We would like to examine if results given by classification align with what we achieve in clustering.

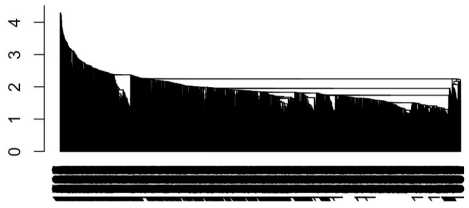
Clustering Analysis

Hierarchical Clustering

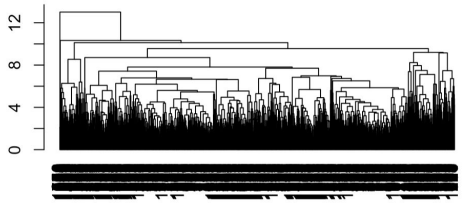
We first scaled data, removed null values (There are 10 null cells) to prepare for clustering. Then we calculated Euclidean distance among 1990 data points, and applied hierarchical clustering methods.

We tried hierarchical clustering in different ways, including single, average, median, complete, centroid and ward.D methods. However, all dendrograms look messy and do not show clear structures of hierarchy as displayed below.

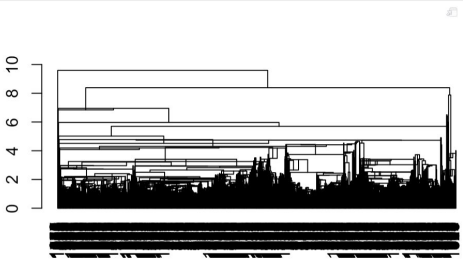
```r
hc1.2 <- hclust(d, method = "single")
plot(hc1.2, hang = -1, ann = FALSE)
```



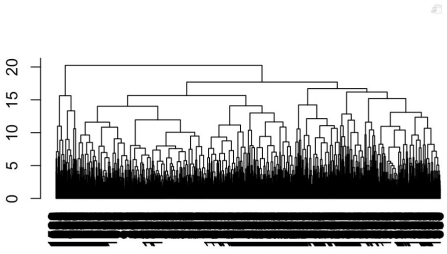

```r
hc2.2 <- hclust(d, method = "average")
plot(hc2.2, hang = -1, ann = FALSE)
```



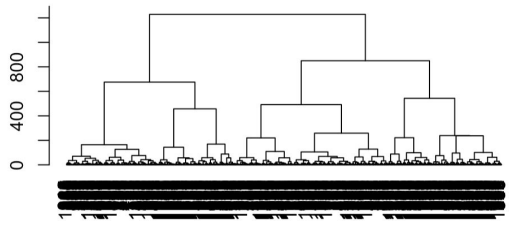

```r
hc3.2 <- hclust(d, method = "median")
plot(hc3.2, hang = -1, ann = FALSE)
```



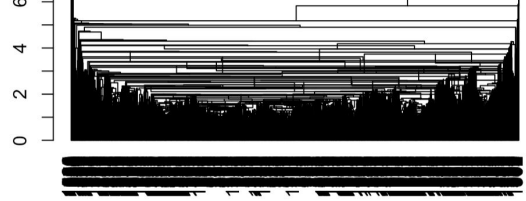

```r
hc4.2 <- hclust(d, method = "complete")
plot(hc4.2, hang = -1, ann = FALSE)
```




```r
hc6.2 <- hclust(d, method = "ward.D")
plot(hc6.2, hang = -1, ann = FALSE)
```

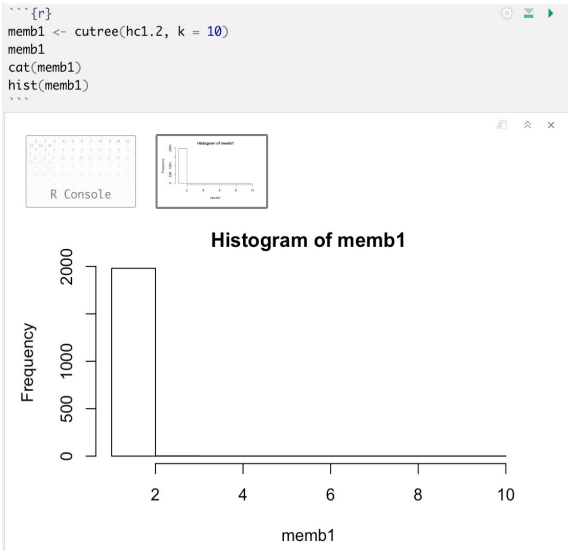


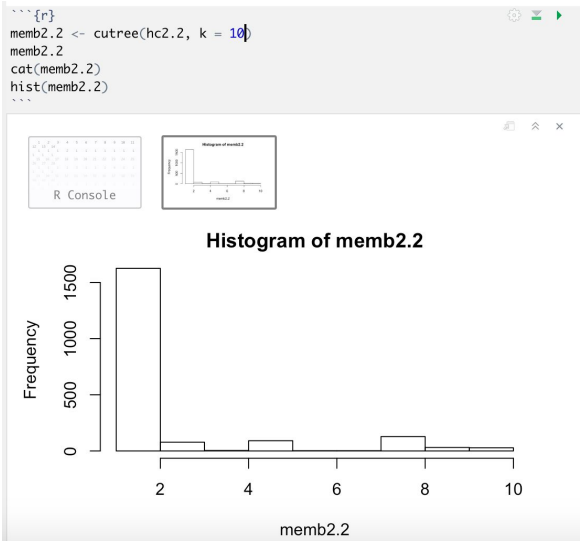

```r
hc5.2 <- hclust(d, method = "centroid")
plot(hc5.2, hang = -1, ann = FALSE)
```

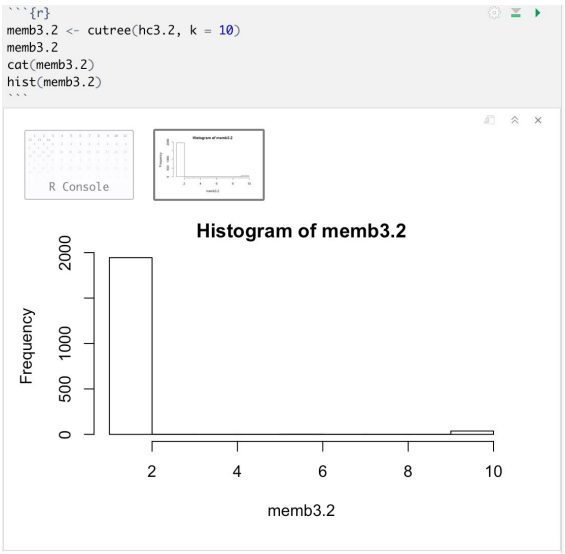


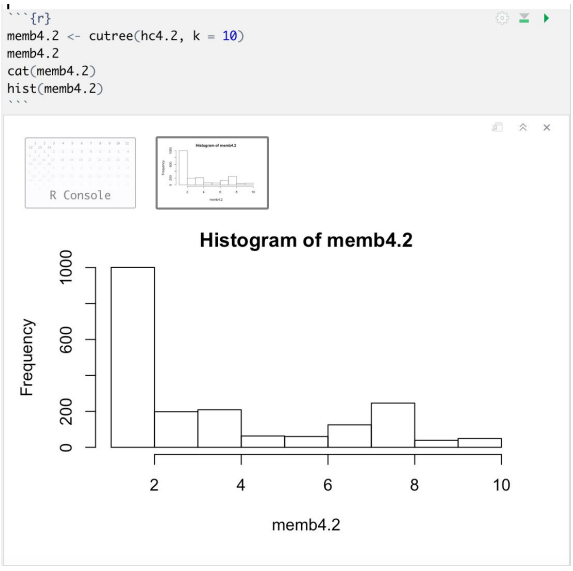
Though methods such as ward.D or complete linkage deliver reasonable results and graphs, the clustering still makes no sense to us. We expected buoys from nearby locations to cluster, however, none of the pictures above shows such a trend. In the extreme case of single linkage method, data collected by the first buoy is at the deepest branch, which looks counter-intuitive.

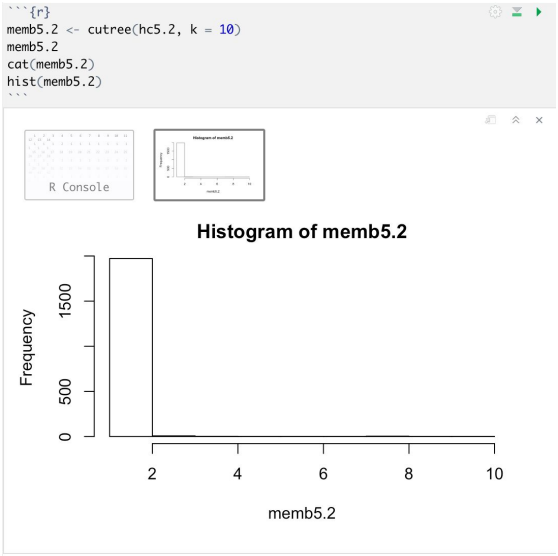
We then specified the number of clusters desired for dendrograms displayed above. Since we have 10 buoys in this case, we chose 10 as the default number for this value. We plotted histograms to show the distribution of data in each clustering analysis, and they are displayed below.

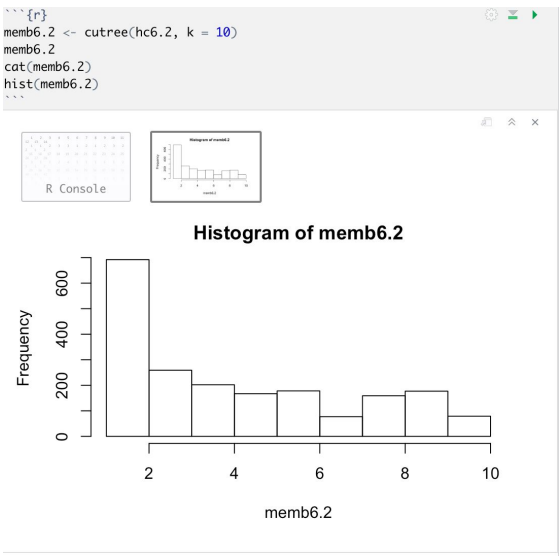












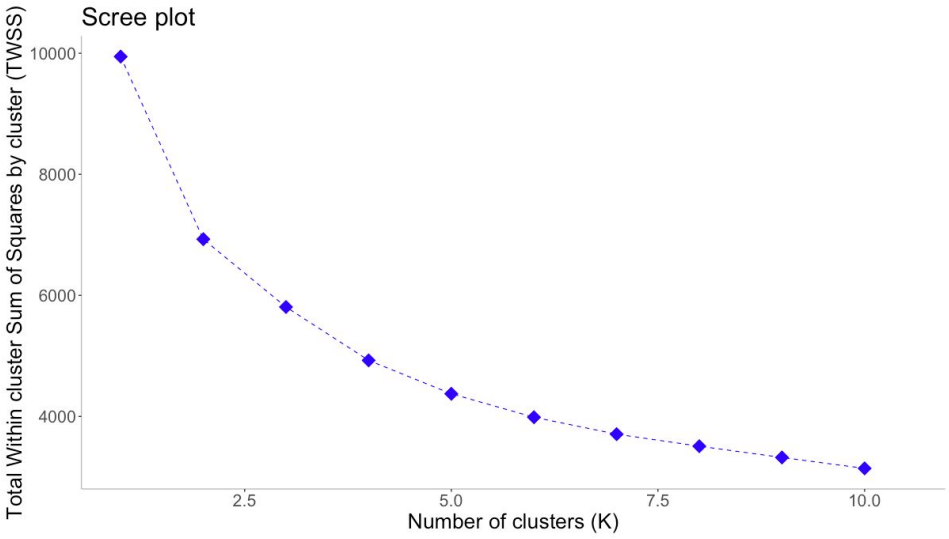
Given the number of clusters equals to 10, the histograms show that data are more closely related to each other than we thought. In the first 5 diagrams, almost all data falls into one group. The ward.D method generates a more even distribution among clusters, though the density still bias towards cluster one.

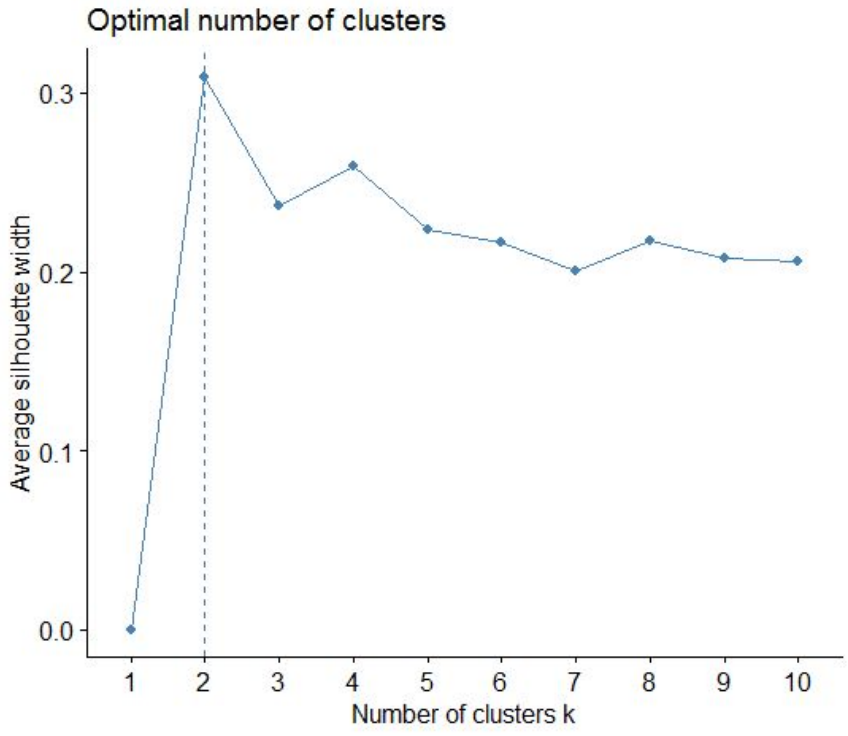
As a result, we decided to examine the actual clustering results of ward.D method. However, they showed that our hypothesis that data collected within nearby regions is way off from what really happens in clustering, and the results appear to be random. Data collected by a buoy should have consistently fallen into one or two groups, however, in reality they are sparsely distributed across all 10 clusters and we failed to generate a convincing interpretation. This dataset has no innate hierarchical structure, and thus does not fit in any hierarchical explanation.

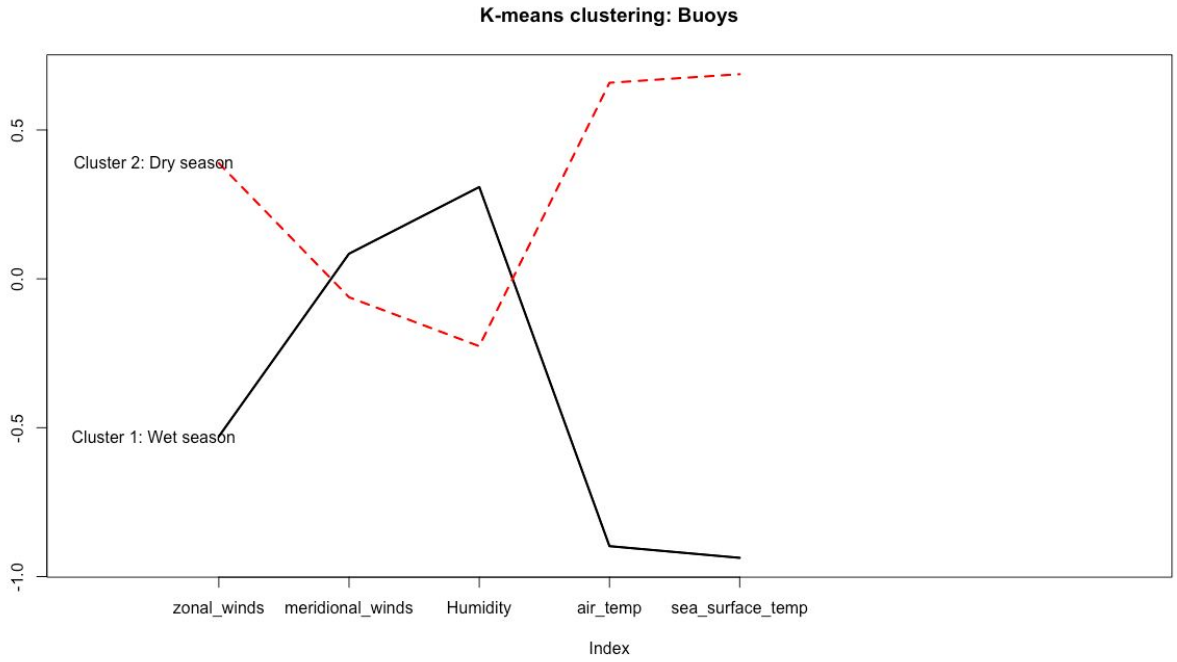
Therefore, we decided to move on K-Means Clustering.

K-Means Clustering

Diagrams to show the performance of clustering are shown below and facilitate us to choose the optimal K.







The determined optimal number of clusters: 2

The best number of the clusters (k) is defined by the smallest within sum of squared errors (wsse). However, the wsse approaches zero as the k increases; it's not meaningful to choose a large k. The efficient way is to choose a small k that has a low SSE. The "bent" point is where the optimal k locates. According to the scree plot, two clusters are optimal. We also used the average silhouette method to determine the optimal number of clusters, and the result is two..

By clustering the data into two clusters, we found that there are patterns between the clusters. One cluster has high humidity and low temperature, whereas the other cluster has high temperature and low humidity, and values of all the other variables are opposite. As a matter of fact, in tropical areas, the dry season had lower humidity but higher temperatures, and wet seasons had higher humidity but lower temperatures. Therefore, it is highly probable that characteristics of the two clusters reflect the seasonality in equatorial Pacific -- wet and dry seasons. It can be seen from the last picture above that air_temp (air temperature) and sea_surface_temp(sea surface temperature) contribute the most to the clustering. We bear that in mind, and once we run into overfitting of models later in classification analysis, we will consider removing these two variables which utterly divides the clusters.

Meanwhile, the clustering results deviated from what we set to achieve in the beginning. We expected clusters to be parted by geographical locations, i.e. data collected from nearby regions tend to cluster. However, this appears not to be the case for reasons stated in previous paragraphs. Seasonality plays a more important role than locations.

Going forward from the clustering analysis, we labeled the two clusters as “Wet Season” and “Dry Season”, and revisited logistic regression, K nearest neighbors and decision trees to investigate how these models perform on the dichotomous data we just created.

Descriptive Statistics

Wet Season Statistics

	ZonalWinds	MeridionalWinds	Humidity	AirTemp	SeaSurfaceTemp
Min	-9.60	-7.90	67.30	22.32	22.38
Max	-0.70	6.00	96.30	27.52	28.23
Mean	-6.27	-0.37	82.88	25.63	25.93
Median	-6.30	-0.25	82.40	25.85	26.18
Sd	1.35	2.22	4.85	1.00	1.13
Var	1.82	4.92	23.55	1.00	1.28

Wet Season	ZonalWinds	Meridional Winds	Humidity	AirTemp	SeaSurfaceTemp
Mode	**-6.8, -6.1**	**0.1**	**82.4, 82.8, 83.6, 86.1**	**26.33, 26.7**	**26.76**

Dry Season Statistics

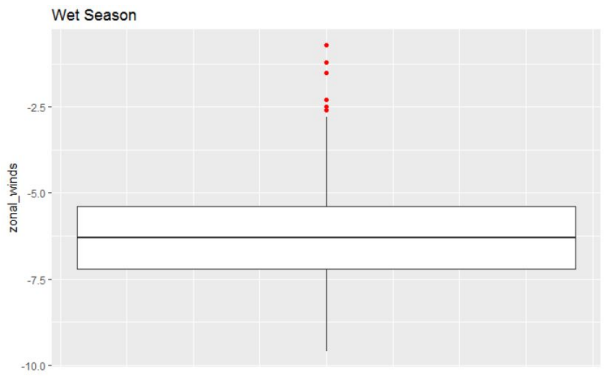
	ZonalWinds	MeridionalWinds	Humidity	AirTemp	SeaSurfaceTemp
Min	-9.80	-8.60	62.60	25.57	26.73
Max	7.60	6.20	94.30	29.97	30.54
Mean	-3.75	-0.72	80.17	27.68	28.47
Median	-4.35	-0.70	80.50	27.68	28.43
Sd	3.00	2.59	4.93	0.71	0.76
Var	9.01	6.69	24.31	0.51	0.57

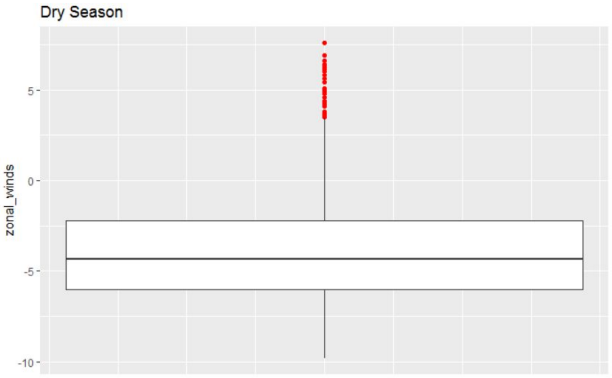
Dry Season	ZonalWinds	MeridionalWinds	Humidity	AirTemp	SeaSurfaceTemp

Mode	**-4.6**	**-0.6**	**82**	**27.75**	**28.38, 28.44, 29.35**

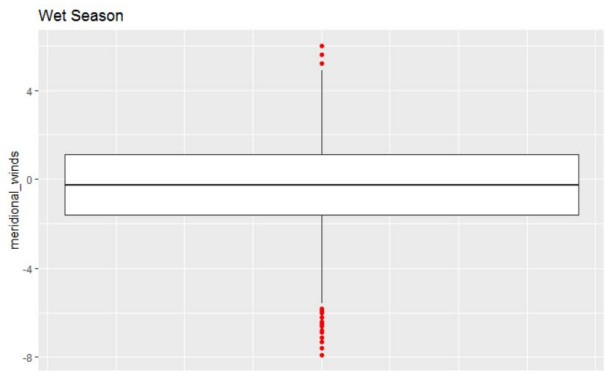
Plots

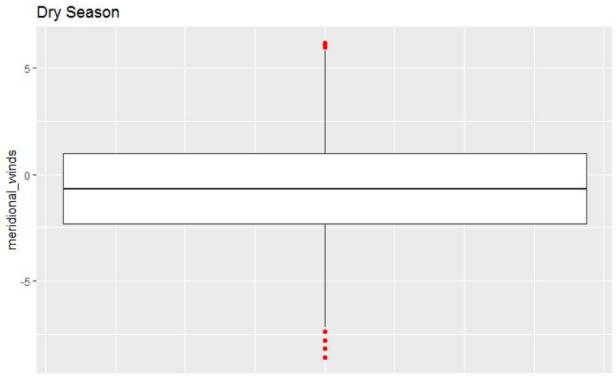
Zonal Winds



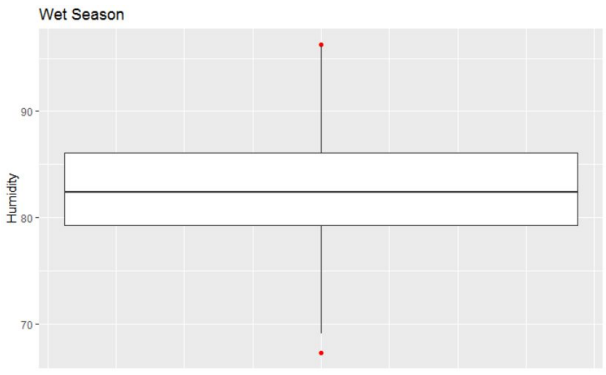


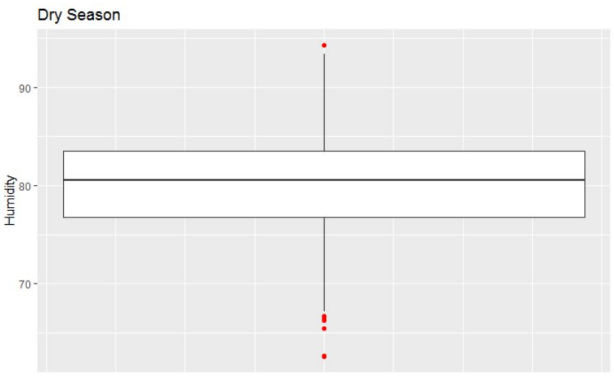
Meridional Winds



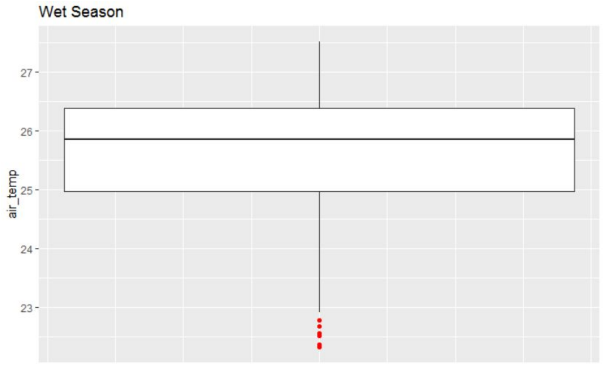


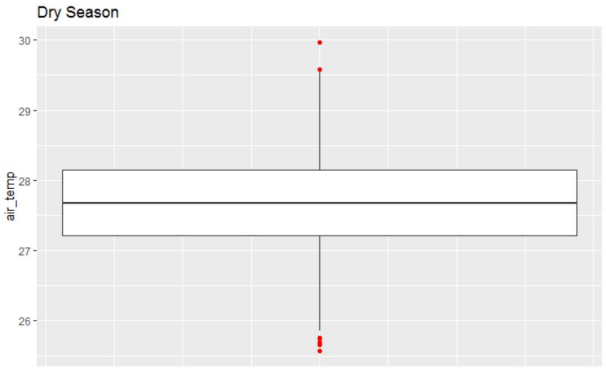
Humidity



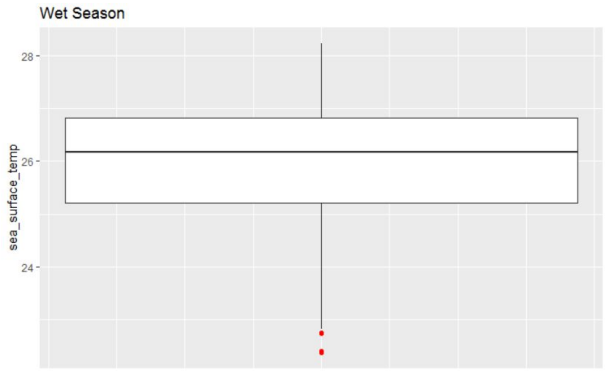


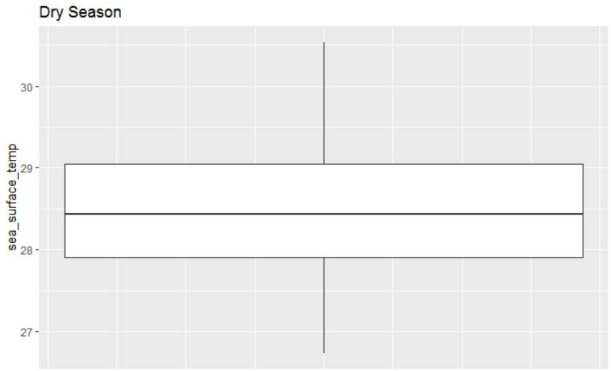
Air Temp



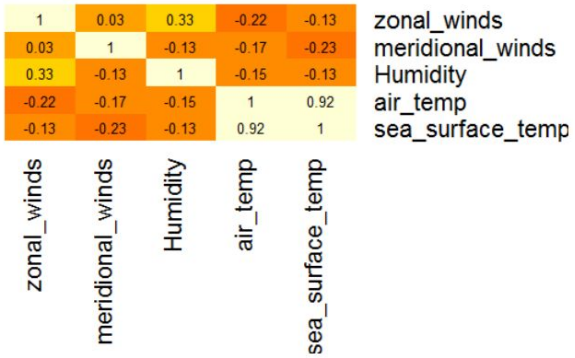


Sea Surface Temp

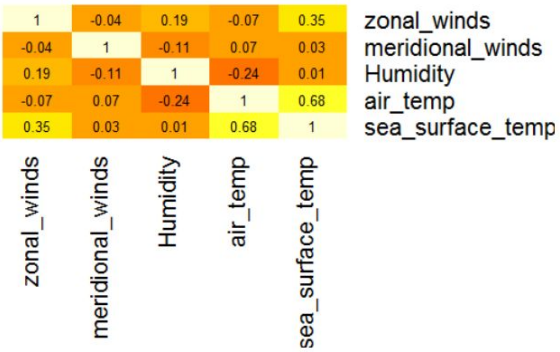




Correlation Heatmap (Wet Season)



Correlation Heatmap (Dry Season)



Outlier detection

To detect the outliers in our dataset, we calculated the Euclidean distance of each point to its cluster center. Here are the 100 farthest distances from the points to the centers:

26345 25090 25743 26451 25436 26133 25622 26082 25636 25801 25571 25885 26182 26040 25420 25468 26004 26204 26440 25281 25735 25415 26117 25296 25462 26075 25161 25544 26271 25312 25900 25519 25135 25216 25284 26241 26247 26464 26027 25893 26094 25570 25765 25336 26197 25764 25463 26236 25219 25867 25882 25688 25773 25875 26032 25517 25786 25529 26116 25194 25562 25568 26398 25506 26031 26050 26319 25385 26437 26181 25332 25701 25408 25927 25187 26112 26160 26225 25222 25445 26065 26153 26028 25951 26106 25744 25909 26343 25934 25117 25831 26230 26256 25704 25471 25758 25075 25845 26093 25266

Because there are no significant differences between the distances, we determined that there is no outlier to remove in this dataset.

Logistic Regression

Because we clustered the data into 2 clusters “wet season” and “dry season”, Logistic regression serves as a reasonable model to predict the cluster of the data since it is used to model dichotomous outcome variables.

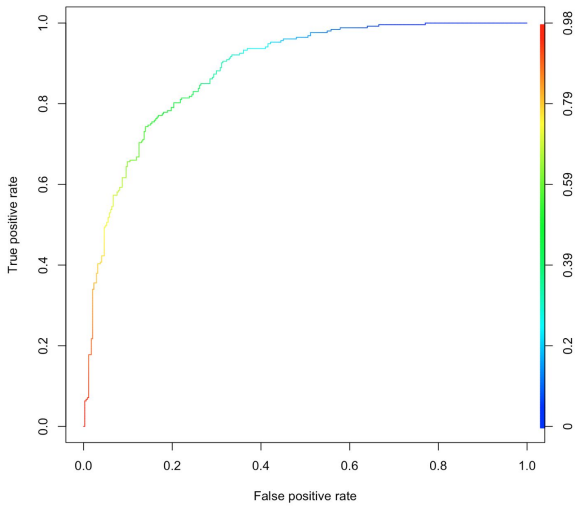
We initially run a logistic regression of “Cluster_type” on all variables, their squared terms, interaction terms, and monthly dummies. However, this regression was overfitting the data because the accuracy, sensitivity and specificity reached 1. Therefore, we took out “air_temp” and “sea_surface_temp”, not just because we want to avoid overfitting, but also because we are interested in investigating the winds and humidity’s correlation with wet and dry seasons. We also iteratively removed squared terms, interaction terms, and monthly dummies.

Our final logistic regression model was “Cluster type” against “Zonal.Winds”, “Meridional.Winds”, “Humidity”, “Month” dummy variables from April to September, because the dry season in equatorial Pacific is generally from April to September. Here is the summary of the logistic regression:

```
Call:
glm(formula = Cluster_type ~ meridional_winds + zonal_winds +
    Humidity + month_4 + month_5 + month_6 + month_7 + month_8 +
    month_9, family = binomial(link = "logit"), data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6497  -0.6891  -0.1657   0.7571   2.8148

Coefficients:
                 Estimate Std. Error z value             Pr(>|z|)
(Intercept)      -0.322998   0.102899  -3.139             0.001695 **
meridional_winds  0.259361   0.076905   3.372             0.000745 ***
zonal_winds      -1.947211   0.127292 -15.297 < 0.0000000000000002 ***
Humidity          1.087717   0.086544  12.568 < 0.0000000000000002 ***
month_4          -2.011188   0.353335  -5.692     0.00000001255469 ***
month_5          -1.760004   0.254418  -6.918     0.00000000000459 ***
month_6          -0.988744   0.250937  -3.940     0.00008140920347 ***
month_7          -0.971126   0.259356  -3.744             0.000181 ***
month_8          -0.008131   0.264020  -0.031             0.975433
month_9           0.569943   0.279997   2.036             0.041797 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1897.8  on 1392  degrees of freedom
Residual deviance: 1237.4  on 1383  degrees of freedom
AIC: 1257.4

Number of Fisher Scoring iterations: 5
```

The regression is a fit to the classification so we used this regression to predict the cluster type of the validation data. Here are the confusion matrix of the prediction and ROC curve:

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 287  60
         1  57 193

               Accuracy : 0.804
                 95% CI : (0.7699, 0.8351)
    No Information Rate : 0.5762
    P-Value [Acc > NIR] : <0.0000000000000002

                  Kappa : 0.5981

 Mcnemar's Test P-Value : 0.8533

            Sensitivity : 0.8343
            Specificity : 0.7628
         Pos Pred Value : 0.8271
         Neg Pred Value : 0.7720
             Prevalence : 0.5762
         Detection Rate : 0.4807
   Detection Prevalence : 0.5812
      Balanced Accuracy : 0.7986

       'Positive' Class : 0
```



From the results we notice that the following:

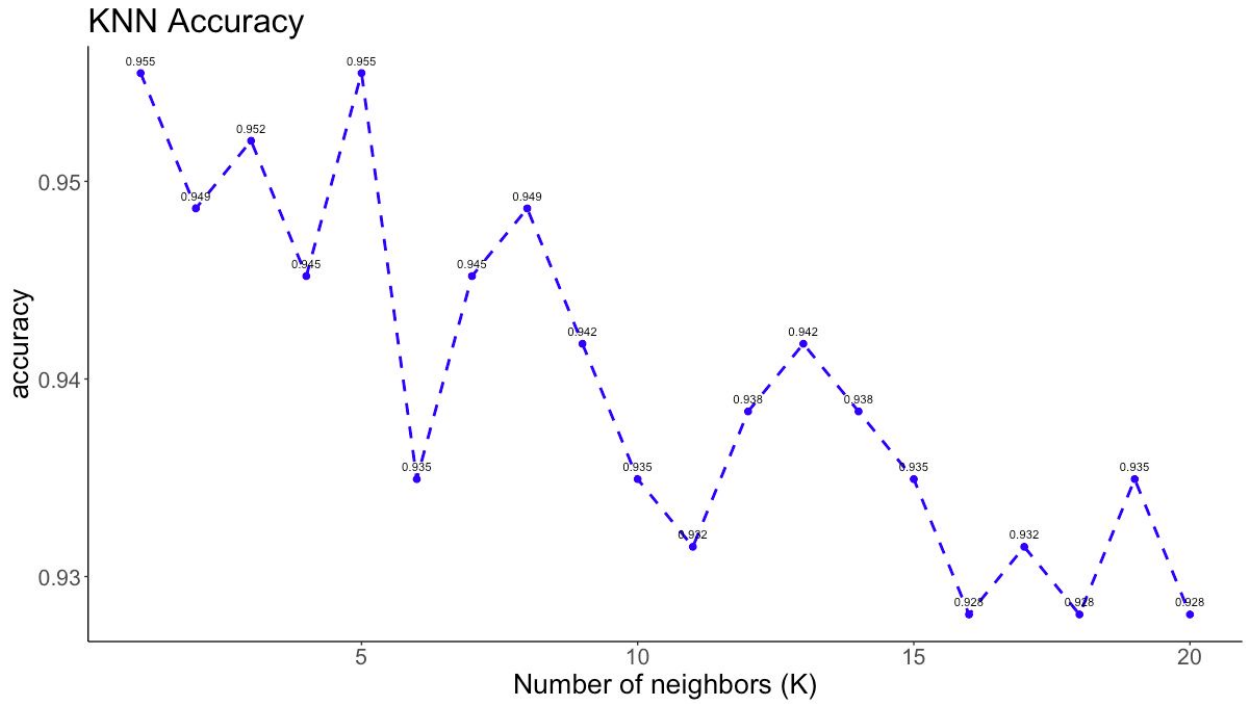
- Accuracy = 0.804
- Sensitivity = 0.8343
- Specificity = 0.7628

Because of the values above, logistic regression is an effective model to predict whether the data belongs to “wet season” or “dry season”. However, we cannot compare the result to our last assignment due to the change of datasets.

From this model, we can classify the data according to winds, humidity and months to dry and wet seasons, so that in future when we want to investigate the El Nino effect in a specific season, we classify the data by this model to generate more accurate insights.

KNN

We also used the K-Nearest Neighbor (KNN) algorithm for the classification of the two clusters. We split the data into 70% training data, 15% validating data and 15% testing data. We then calculated the distance between two data points using the Euclidean distance method. Inorder to choose the best *"K"*, we run the KNN algorithm on the validating data for different value of *"K"* from 1 to 20. Below is a graph showing the accuracy (true positives and true negatives) derived from the confusion matrix for different values of *"K"*. The graph clearly shows that a *"K"* of **1** yields the highest accuracy of 76.73%.



Using a *"K"* = 1, we run the KNN algorithm on the testing data and this yields a marginally higher accuracy of 94.94%. We used the same procedure as outlined in the logistic regression section of iteratively removing squared terms, interaction terms, and monthly dummies to minimize overfitting in our preliminary models. This final model neither underfits nor overfits. Below is a snippet showing the confusion matrix from this model.

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 156   5
         1  11 144

               Accuracy : 0.9494
                 95% CI : (0.9191, 0.9708)
    No Information Rate : 0.5285
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.8986

 Mcnemar's Test P-Value : 0.2113

            Sensitivity : 0.9341
            Specificity : 0.9664
         Pos Pred Value : 0.9689
         Neg Pred Value : 0.9290
             Prevalence : 0.5285
         Detection Rate : 0.4937
   Detection Prevalence : 0.5095
      Balanced Accuracy : 0.9503

       'Positive' Class : 0
```

From the results we notice that the following results:

- Accuracy = 0.9494
- Specificity = 0.9664
- Sensitivity = 0.9341

With KNN, the Accuracy is slightly higher than in logistic regression. Based on these results, we can conclude that KNN is as effective in classifying the clusters.

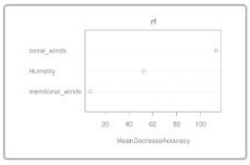
Random Forest

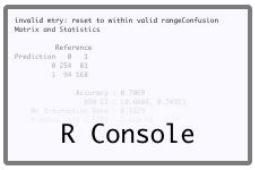
Similar to the case in the above two classification analysis, we encounter the overfitting problem while regressing Cluster_type on all five variables plus month dummy variables using random forest method. The accuracy, sensitivity and specificity all approach 0.99. To mitigate this problem, similar to the approach used in logistic regression, we removed air_temp(air temperature), sea_surface_temp(sea temperature) and month dummy variables. As discussed

in clustering analysis, values of sea_sufrace_temp and air_temp variables are opposite for the two clusters and contribute a lot to delineate the borders between the two clusters. Moreover, month dummy variables indicate the month that the data point is collected, and helps clusters to identify if a data point is literally in dry or wet season. After removing these variables, we hope that the overfitting problem will be solved.

The summary of a random forest model trained on 70% of the data and three variables is attached below:

```{r}
rf <- randomForest(Cluster_type ~ ., data = df_train, ntree = 500,
                   mtry = 4, nodesize = 5, importance = TRUE)
varImpPlot(rf, type = 1)
rf.pred <- predict(rf, df_test)
confusionMatrix(rf.pred, as.factor(df_test$Cluster_type))
```






```
               Accuracy : 0.7069
                 95% CI : (0.6686, 0.7431)
    No Information Rate : 0.5829
    P-Value [Acc > NIR] : 2.46e-10

                  Kappa : 0.4016

 Mcnemar's Test P-Value : 0.3643

            Sensitivity : 0.7299
            Specificity : 0.6747
         Pos Pred Value : 0.7582
         Neg Pred Value : 0.6412
             Prevalence : 0.5829
         Detection Rate : 0.4255
   Detection Prevalence : 0.5611
      Balanced Accuracy : 0.7023
```

Accuracy retreats to 70 percent level. This may not be a perfect result, given that a random guess will produce an accuracy of 50%. However, this model improves a lot in terms of overfitting and is good enough to be implemented on a large training dataset.

Interpretation of Classification Models

After trying three classification models to fit the clustered dataset, we found that accuracy values of these models approach perfection (100%) if we used the same variables that are used in clustering models in classification models. In our case, air_temp, sea_surface_temp and month dummy variables, which differentiate drastically in the two clusters, contribute to this phenomenon. As a result, we were more concerned with overfitting than outlier issues.

Later when we took out air_temp, sea_surface_temp and month dummy variables, the classification models generated better and more “flawed” results, i.e. accuracy retreated to normal levels. In fact, it is hard to decide in this case which classification model works the best. All three models that we tried run into overfitting issues, and only the removal of some over significant variables alleviate the situation. The results afterwards of these three models cannot be compared directly either, since once we pushed sufficiently hard for any model with the remaining variables, it will soon overfit the data absurdly.

Though overfitting is not welcome in the classification, it shows that K-Means Clustering actually works well on the original dataset. A return to normalcy of the classification models after the removal of temperature and month dummy variables also justifies our interpretation for the clustering results. Values of temperature variables change significantly in the two clusters and determine the borders of the two, corresponding to seasonal cycles in real life. One cluster reflects the climate in “wet season”, and the other “dry season.”