Quantifying the impact of wind and wave action on seawater temperature along the South African coastline (Appendices)

Amieroh Abrahams *University of the Western Cape*

This file was generated in R using Rmarkdown, with a bit of LaTeX thrown in:

```r
sessionInfo()
```

```
## R version 3.5.0 (2018-04-23)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS  10.14.1
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     grDevices utils     datasets  graphics  methods   base
##
## loaded via a namespace (and not attached):
##  [1] compiler_3.5.0  backports_1.1.2 magrittr_1.5    rprojroot_1.3-2
##  [5] tools_3.5.0     htmltools_0.3.6 yaml_2.2.0      Rcpp_1.0.0
##  [9] stringi_1.2.4   rmarkdown_1.10  knitr_1.20      stringr_1.3.1
## [13] digest_0.6.18   evaluate_0.12
```

Appendix A

Background and code

The intention of this section is to show the approach and **R** scripts used to analyse the possible reasons for temperature variation between sites within the same cluster along the coast of South Africa. Specifically, I wish to determine if wave and wind action are the possible reasons for this variation.

The data

The SACTN dataset is the primary source of temperature data used in this study. This dataset consisted of *in situ* collected coastal seawater temperatures for 129 sites along the coast of South Africa, measured daily from 1972 until 2017. This study also make use of AVHRR, MUR, CMC and K10 satellite-derived SST datasets as well as wind and wave action data that were obtained from the South African Weather Service (SAWS), and were provided at three hour resolutions. This data was then used to model short–crested waves, generated by the wind into the coastal environment, using the wave model Simulating Waves in the Nearshore (SWAN).

Setting up the analysis

This is **R**, so first I need to find, install and load various packages. Some of the packages will be available on CRAN and can be accessed and installed in the usual way.

```r
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------------------------- tidy

## v ggplot2 3.1.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.8
## v tidyr   0.8.2     v stringr 1.3.1
## v readr   1.1.1     v forcats 0.3.0

## -- Conflicts ----------------------------------------------------------------------------------------------- tidyverse_c
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x dplyr::n()      masks .env::n()
```

```r
library(ggpubr)
```

```
## Loading required package: magrittr

##
## Attaching package: 'magrittr'

## The following object is masked from 'package:purrr':
##
##     set_names

## The following object is masked from 'package:tidyr':
##
##     extract
```

```r
library(zoo)
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

```r
library(ggrepel)
library(FNN)
library(stringr)
library(circular)
```

```
##
## Attaching package: 'circular'

## The following objects are masked from 'package:stats':
##
##     sd, var
```

```r
library(broom)
library(ggrepel)
library(purrr)
library(stlplus)
source("../functions/earthdist.R")
source("../functions/scale.bar.func.R")
```

```
##
## Attaching package: 'maps'

## The following object is masked from 'package:purrr':
##
##     map

## Loading required package: sp

## Checking rgeos availability: TRUE
```

```r
source("../functions/wind.rose.R")
```

```
## Loading required package: RColorBrewer
```

```r
source("../functions/waverose.diagram.R")
source("../functions/match_func.R")
source("../functions/seasonal_match.R")
source("../functions/anova_func.R")
source("../functions/Load_wave_function.R")
```

```
## Parsed with column specification:
## cols(
##   site = col_character(),
##   lon = col_double(),
##   lat = col_double(),
##   site_list = col_character(),
##   wave_7 = col_character(),
##   wave_15 = col_character()
## )
```

```
##
  |
  |                                                                 |   0%
  |
  |===                                                              |   5%
  |
  |======                                                           |  10%
  |
  |=========                                                        |  14%
  |
  |============                                                     |  19%
  |
  |===============                                                  |  24%
  |
  |===================                                              |  29%
  |
  |======================                                           |  33%
  |
  |=========================                                        |  38%
  |
  |============================                                     |  43%
  |
  |===============================                                  |  48%
  |
  |==================================                               |  52%
  |
  |=====================================                            |  57%
  |
  |========================================                         |  62%
  |
  |===========================================                      |  67%
  |
  |==============================================                   |  71%
  |
  |==================================================               |  76%
  |
  |=====================================================            |  81%
  |
  |========================================================         |  86%
  |
  |===========================================================      |  90%
  |
  |==============================================================   |  95%
  |
  |=================================================================| 100%
##
  |
  |                                                                 |   0%
```

```
  |                                                                 
  |====                                                             |   6%
  |                                                                 
  |========                                                         |  12%
  |                                                                 
  |===========                                                      |  18%
  |                                                                 
  |===============                                                  |  24%
  |                                                                 
  |===================                                              |  29%
  |                                                                 
  |=======================                                          |  35%
  |                                                                 
  |===========================                                      |  41%
  |                                                                 
  |===============================                                  |  47%
  |                                                                 
  |==================================                               |  53%
  |                                                                 
  |======================================                           |  59%
  |                                                                 
  |==========================================                       |  65%
  |                                                                 
  |==============================================                   |  71%
  |                                                                 
  |==================================================               |  76%
  |                                                                 
  |======================================================           |  82%
  |                                                                 
  |=========================================================        |  88%
  |                                                                 
  |=============================================================    |  94%
  |                                                                 
  |=================================================================| 100%
##
  |                                                                 
  |                                                                 |   0%
  |                                                                 
  |================================                                 |  50%
  |                                                                 
  |=================================================================| 100%
##
  |                                                                 
  |                                                                 |   0%
  |                                                                 
  |================================                                 |  50%
  |                                                                 
  |=================================================================| 100%
```

```
## Joining, by = "site"
```

```r
source("../functions/R_func.R")
source("../functions/temp_r2.R")
source("../functions/temp_wave_r2.R")
```

Appendix B

Now to get to the data. Here I load the site list dataset. This dataset comprise of the statistical properties of the seawater temperature representing the South African coastline, such as the mean, minimum and maximum temperatures. These values vary among coastal sections due to the influence of the cold Benguala and warm Agulhas currents. At the broad scale, this region exhibits a large variation in seawater temperatures along its coastline (Mead et al. 2013; Smit et al. 2013) and is divided into four bioregions, each with contrasting temperatures. These bioregions are the Benguela Marrine Province (BMP), Benguela-Agulhas Transition Zone (B-ATZ), the Agulhas Marine Province (AMP) and the East Coast Transition Zone (ECTZ) (Smit, Bolton, and Anderson 2017). The Benguela Marine Province (BMP) is located to the west of Cape Point and is characterized by the movement of cold water from the Southern Ocean moving north. The cold temperate west coast mentioned above is greatly affected by the upwelling caused by offshore winds, generated by the cold Benguela current. Thus the seawater temperatures found around South Africa's coastline exhibits a large variational range. My plotting functions partition the data into the clusters and colour code each point accordingly so that I am able to see patterns that exist and distictly group the different sites with similar temperatures along the three distinct coasts.

```
load("../data/site_list_v4.2.RData")
```

The next step involves SACTN temperature dataset. First I compute a k-means clustering on a data matrix using the `kmeans()` function, which uses multiple random seeds to find a number of clusting solutions. This groups sites based on minimum, maximum and mean temperatures. Thereafter I select all the sites with a overlapping time series of at least 10 years of data and excluding temperatures collected deeper than 5m.

```
load("../SACTN_data/SACTN_daily_v4.2.RData")

set.seed(10)
site_list$cluster <- as.factor(kmeans(site_list[,c(15, 18:19)], 6)$cluster)

site_list_clusters <- site_list %>%
  filter(length >= 3650, depth <= 5)

SACTN_daily_clusters <- SACTN_daily_v4.2 %>%
  left_join(site_list[,c(4, 13, 21)], by = "index") %>%
  filter(index %in% site_list_clusters$index)
```

With the sites now split into eight clusters, the next step is to reduce the number of sites per cluster down to a manageable, but still representative, sub-sample of the whole. This is done primarily for two reasons. The first, simply, is to allow for the comparisons to be more readily interpretable by humans. The second more objective reason is to allow for an equal amount of sampling per cluster. The East coast is much more heavily sampled than the rest and so this imbalance must be addressed. The criteria that must be considered for the sub-samples are that both instrument types (UTR and thermometer) are present, and as many sources are included as are possible. This thermal data contain various variables, in this study I only make use of some of them. This resulted in 18 sites, three per cluster. My plotting functions partition the data into its respective groups and colour code the figures accordingly.

```
# Read in the species data
# Will be provided -- in the data folder as "site_list_sub.RData"

# save(site_list_sub, file = "data/site_list_sub.RData")
load("../data/site_list_sub.RData")
SACTN_daily_clusters_sub <- SACTN_daily_clusters %>%   #Subsetting the data
  filter(index %in% site_list_sub$index)
# save(SACTN_daily_clusters_sub, file = "data/SACTN_daily_clusters_sub.RData") #Saving the data for later use
```

Table 1: The 18 sites along the South African coastline, with approximate GPS coordinates, source and the cluster of which each site belong to based on maximum, minimum and mean temperature

Cluster	Site	SRC	Lon	Lat
1	Hamburg	DEA	27.49	-33.29
1	Eastern Beach	SAWS	27.92	-33.02
1	Orient Beach	SAWS	27.92	-33.02
2	Stilbaai	SAWS	21.44	-34.37
2	Mossel Bay	SAWS	22.13	-34.17
2	Knysna	SAWS	23.05	-34.09
3	Saldanha Bay	SAWS	17.95	-33.01
3	Bordjies	DAFF	18.46	-34.32
3	Gansbaai	SAWS	19.34	-34.59
4	Port Edward	KZNSB	30.23	-31.04
4	T.O. Strand	KZNSB	30.23	-31.03
4	Leisure Bay	KZNSB	30.25	-31.02
5	Port Nolloth	SAWS	16.87	-29.25
5	Lamberts Bay	SAWS	18.30	-32.09
5	Sea Point	SAWS	18.38	-33.92
6	Kalk Bay	SAWS	18.45	-34.13
6	Muizenberg	SAWS	18.48	-34.10
6	Gordons Bay	SAWS	18.86	-34.16

With the temperature data now loaded in, I am now able to calculate the distance between each of the sites. Here I set up a geographic or Euclidian distance matrix representing the pairwise distances between the n sites ($D = [d_{ij}]$). I already have a function for this and can be found in the folder titled functions.

```r
load("../data/sa_coast.Rdata") # Coastal data
options(scipen=999) # Forces R to not use exponential notation
sa_coast_prep <- sa_coast %>%
  mutate(site = 1:nrow(.),
         X = deg2rad(X),
         Y = deg2rad(Y)) %>%
  select(site, X, Y)

sa_coast_dist <- as.data.frame(round(PairsDists(sa_coast_prep), 2)) %>%
  mutate(lon = sa_coast$X,
         lat = sa_coast$Y) %>%
  dplyr::rename(dist = V1) %>%
  select(lon, lat, dist) %>%
  mutate(cum_dist = cumsum(dist)) %>%
  mutate(dist = lag(dist),
         cum_dist = lag(cum_dist))
```

Matching sites based on overlapping time series

Next I create a function to match the sites with each of the clusters. This allows me to compare whether or not a temperature variation exist between sites within the same cluster and to examine reasons for this variation. Here I also consider distance between sites.

```r
load("../data/SACTN_clust_1_match.RData")
load("../data/SACTN_clust_2_match.RData")
load("../data/SACTN_clust_3_match.RData")
load("../data/SACTN_clust_4_match.RData")
load("../data/SACTN_clust_5_match.RData")
load("../data/SACTN_clust_6_match.RData")
```

I then create a function which allows me to calculate the monthly mean temperature and standard deviation for each year and sites. Consequently we have a dataframe with seven columns and each of the rows representing each month for each of the years. By calculating the mean and SD for each site I am able identify trends in their temperatures and to compare these trends across sites within the same cluster. Mean and SD gives a broad look of the overall temperature for each of the sites. With each group of months representing the season.

```r
load("../data/SACTN_clust_1_matched.RData")
load("../data/SACTN_clust_2_matched.RData")
load("../data/SACTN_clust_3_matched.RData")
load("../data/SACTN_clust_4_matched.RData")
load("../data/SACTN_clust_5_matched.RData")
load("../data/SACTN_clust_6_matched.RData")
```

Analyses

Here I do a three way ANOVA analayses. This allows me to compare one variable in two or more groups taking into account the variability of other variables. This analysis of covariance is used to test the main effect of variables on a continuous variable. In this case we specifically analyse the relationship between the average temperature as a function of index pair (Paired sites), year and season. For this I create a function and so I replicate the same test for each of the six clusters.

```r
anova_clus_1_func1 <- anova_func(df = SACTN_clust_1_matched)
summary(anova_clus_1_func1)
```

```
##                            Df Sum Sq Mean Sq F value      Pr(>F)
## index_pair                  2  15.28   7.642  12.071 0.00000876 ***
## year                        1   0.07   0.072   0.114    0.73553
```

```
## season                  3   9.53   3.177   5.019    0.00205 **
## index_pair:year         2   1.75   0.876   1.384    0.25198
## index_pair:season       6  13.07   2.179   3.442    0.00261 **
## year:season             3   4.01   1.338   2.113    0.09844 .
## index_pair:year:season  6   7.93   1.322   2.088    0.05426 .
## Residuals             326 206.39   0.633
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 32 observations deleted due to missingness
```

```r
anova_clus_2_func1 <- anova_func(df = SACTN_clust_2_matched)
summary(anova_clus_2_func1)
```

```
##                          Df Sum Sq Mean Sq F value              Pr(>F)
## index_pair                2  418.6  209.29 166.848 < 0.0000000000000002
## year                      1   41.7   41.66  33.211    0.000000010818789
## season                    3  144.1   48.04  38.296 < 0.0000000000000002
## index_pair:year           2   71.4   35.70  28.463    0.000000000000907
## index_pair:season         6  125.9   20.98  16.722 < 0.0000000000000002
## year:season               3    7.7    2.57   2.050                0.105
## index_pair:year:season    6   10.6    1.76   1.403                0.210
## Residuals              1065 1335.9    1.25
##
## index_pair             ***
## year                   ***
## season                 ***
## index_pair:year        ***
## index_pair:season      ***
## year:season
## index_pair:year:season
## Residuals
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 103 observations deleted due to missingness
```

```r
anova_clus_3_func1 <- anova_func(df = SACTN_clust_3_matched)
summary(anova_clus_3_func1)
```

```
##                         Df Sum Sq Mean Sq F value              Pr(>F)
## index_pair               2    2.9    1.45   1.165              0.3127
## year                     1    0.3    0.27   0.218              0.6405
## season                   3  598.7  199.58 159.866 < 0.0000000000000002 ***
## index_pair:year          2   18.8    9.38   7.514              0.0006 ***
## index_pair:season        6   91.6   15.26  12.223    0.000000000000564 ***
## year:season              3    3.5    1.18   0.942              0.4201
## index_pair:year:season   6   19.7    3.29   2.635              0.0157 *
## Residuals              584  729.1    1.25
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 148 observations deleted due to missingness
```

```r
anova_clus_4_func1 <- anova_func(df = SACTN_clust_4_matched)
summary(anova_clus_4_func1)
```

```
##                          Df Sum Sq   Mean Sq F value Pr(>F)
## index_pair                2 0.0028 0.0013828   0.736  0.479
```

```
## year                       1 0.0009 0.0009298   0.495  0.482
## season                     3 0.0042 0.0014164   0.754  0.520
## index_pair:year            2 0.0003 0.0001743   0.093  0.911
## index_pair:season          6 0.0062 0.0010282   0.547  0.772
## year:season                3 0.0091 0.0030444   1.621  0.183
## index_pair:year:season     6 0.0047 0.0007898   0.420  0.866
## Residuals               1208 2.2693 0.0018786
## 28 observations deleted due to missingness
```

```r
anova_clus_5_func1 <- anova_func(df = SACTN_clust_5_matched)
summary(anova_clus_5_func1)
```

```
##                          Df Sum Sq Mean Sq F value               Pr(>F)
## index_pair                2  196.7   98.34  77.103 < 0.0000000000000002
## year                      1  220.4  220.39 172.803 < 0.0000000000000002
## season                    3  114.5   38.15  29.915 < 0.0000000000000002
## index_pair:year           2   35.5   17.74  13.913       0.000001028948
## index_pair:season         6   23.2    3.86   3.029              0.00601
## year:season               3  135.2   45.06  35.333 < 0.0000000000000002
## index_pair:year:season    6   71.0   11.83   9.279       0.000000000522
## Residuals              1515 1932.2    1.28
##
## index_pair             ***
## year                   ***
## season                 ***
## index_pair:year        ***
## index_pair:season      **
## year:season            ***
## index_pair:year:season ***
## Residuals
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 49 observations deleted due to missingness
```

```r
anova_clus_6_func1 <- anova_func(df = SACTN_clust_6_matched)
summary(anova_clus_6_func1)
```

```
##                          Df Sum Sq Mean Sq F value               Pr(>F)
## index_pair                2  419.7  209.83 132.044 < 0.0000000000000002
## year                      1    0.2    0.18   0.115              0.73446
## season                    3   24.8    8.26   5.196              0.00144
## index_pair:year           2  347.4  173.70 109.308 < 0.0000000000000002
## index_pair:season         6  485.8   80.97  50.950 < 0.0000000000000002
## year:season               3    1.9    0.63   0.397              0.75486
## index_pair:year:season    6    7.9    1.31   0.826              0.54998
## Residuals              1223 1943.5    1.59
##
## index_pair             ***
## year
## season                 **
## index_pair:year        ***
## index_pair:season      ***
## year:season
## index_pair:year:season
## Residuals
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 336 observations deleted due to missingness
```

Hereafter, I create a box plot also allowing me to observe whether or not a variation exist between sites within the same cluster. Box plots determine the strength of correlation of temperature between sites found within the same clusters along the coast. Similarly, I create a line plot, using the gam function in R. This allows me to assess whether or not an average temperature differences exist between sites on a seasonal basis and to examine the intensity of this variation over the past years. Due to the length of the ouput I have prevented the scrript from returning here. The scipt however may be found in my GitHub repository, and the full code can be run in its entiety.

Appendix C

Wind and wave data

Since the analysis prove that temperature variation exist between sites within the same cluster, I now continue with the analyses to determine if wind and wave action are the possible causes for this variation. Here I created a function to load in the wind and wave data. This data is presented in single folders for each of the 18 sites and as a result I will not be adding the code here. As mentioned earlier, this code is present in my Github repository.

```
# source("functions/Load wave function.R") # I load this function in the code above
load("../data/wave_data.RData")
```

Wave and wind data were modelled at three hour resolutions and at 7 and 15m isobaths. I now converted the data into daily data points in order to compare them with the temperature data, which is collected daily. The `circular()` function in R software is then used to create circular objects around the wave data in order to calculate the daily wave and wind parameters.

```
load("../data/wave_daily.RData")
```

Now that I have the daily values for the wind and wave variables, I am able to split the dataset into depths of 7 and 15m.

```
sites_complete_7a <- wave_daily %>%
  filter(depth == "7m")

sites_complete_15a <- wave_daily %>%
  filter(depth == "15m")
```

Next I want to show the relationship between temperature and wind and wave action. I create a function matching the wind and wave data with the SACTN temperature data. This allows for further analyses.

```
SACTN_daily_clusters_sub_1 <- SACTN_daily_clusters_sub %>%
  separate(index, into = c("site", "src"), sep = "/", remove = FALSE)

try1 <- function(df) {
  func1 <- df %>%
    left_join(sites_complete_7a, by = c("site", "date")) %>%
    na.trim() %>%
    group_by(site)
    return(func1)
}
try2 <- function(df) {
  func2 <- df %>%
    left_join(sites_complete_15a, by = c("site", "date")) %>%
    na.trim() %>%
    group_by(site)
    return(func2)
}

SACTN_daily_clusters_7a <- try1(df = SACTN_daily_clusters_sub_1)
SACTN_daily_clusters_15a <- try2(df = SACTN_daily_clusters_sub_1)
```

With the wind and wave data now matching the temperature data based on the date and site at which temperature was collected, I am now able to do a regression analyses. Here I create a function which calculates the coefficient of determination (R^2). The coeffcient of determination was calculated to determine how differences in one variable can be explained by a difference in the second variable.

```
# source("functions/R_func.R")
SACTN_7_R2 <- SACTN_daily_clusters_7a %>%
 temp_wave_R2_()
SACTN_15_R2 <- SACTN_daily_clusters_15a %>%
  temp_wave_R2_()

load("../data/SACTN_7_R2.RData")
load("../data/SACTN_15_R2.RData")
```

I repeat the above code, allowing me to match the wind and wave data with the satellite collected SST data. Due to the length of script I do not include this here. The scipt however may be found in my GitHub repository, as previously mentioned. I now read in the R^2 value results for each of the satellite datasets.

```
load("../data/AVHRR_temp_7_R2.RData")
load("../data/AVHRR_temp_15_R2.RData")
load("../data/CMC_7_R2.RData")
load("../data/CMC_15_R2.RData")
load("../data/K10_7_R2.RData")
load("../data/K10_15_R2.RData")
load("../data/MUR_7_R2.RData")
load("../data/MUR_15_R2.RData")
```

The R^2 values indicate that wind and wave action has no significant impact on temperature variation along the South African coastline, my next step is to determine the most predominant wind and wave direction for each of the 18 sites. By doing this I am able to determine whether or not predominant wind and wave direction may have a greater effect on temperature at each of the sites. Here I again make use of the `circular()` function which calculates the circular statistic for wind and wave direction only. I then split these dataset into two seperate datasets comprising of wind and wave data collected at 7 and 15m respectively.

```
wave_daily_dir <- wave_data %>%
  mutate(date = as.Date(date)) %>%
  group_by(site, num, depth, date) %>%
  summarise(dir_circ = mean.circular(circular(dir, units = "degrees")),
            dirw_circ = mean.circular(circular(dirw, units = "degrees")))

wave_daily <- wave_data %>%
  mutate(date = as.Date(date)) %>%
  group_by(site, num, depth, date) %>%
  summarise_all(funs(mean = mean, sd = sd), na.rm = T) %>%
  ungroup() %>%
  left_join(wave_daily_dir)
```

```
## Joining, by = c("site", "num", "depth", "date")
```

```
sites_complete_7 <- wave_daily %>%
  filter(depth == "7m")

sites_complete_15 <- wave_daily %>%
  filter(depth == "15m")
```

Splitting the waves up into two different dataframes requires me to write a bit more code, but ultimately makes it easier to read. The argument `s.window` controls how rapidly the seasonal component can change. Small values allow more rapid change. Setting the seasonal window to be infinite is equivalent to forcing the seasonal component to be periodic. The stlplus package for R has a plot_seasonal function that can be used to generate a cycle-subseries plot in order to visually calibrate `s.window`.

```
# Create site column from the index column
SACTN_daily_clusters_sub <- SACTN_daily_clusters_sub %>%
  separate(index, into = c("site", "src"), sep = "/", remove = FALSE)

# Combine the temperature and 7 m depth wave data
SACTN_daily_clusters_7 <- SACTN_daily_clusters_sub %>%
  left_join(sites_complete_7, by = c("site", "date")) %>%
  na.trim() %>%
  group_by(site) %>%
  mutate(temp_flat = stlplus(x = temp, n.p = 365, s.window = "periodic")[[1]]$remainder)

# Combine the temperature and 15 m depth wave data
SACTN_daily_clusters_15 <- SACTN_daily_clusters_sub %>%
  left_join(sites_complete_15, by = c("site", "date")) %>%
  na.trim() %>%
  group_by(site) %>%
  mutate(temp_flat = stlplus(x = temp, n.p = 365, s.window = "periodic")[[1]]$remainder)
```

Comparing waves and temperatures

With our temperature and wave values paired up we now want to investigate what the relationship between these values may be. In order to do so we will use linear models, as this produces for us the coefficient of determination (R^2). We are going to use **`purrr`** to do this as it will allow us to compare multiple variables at once: The following example uses purrr to solve a fairly realistic problem: split a data frame into pieces, fit a model to each piece, compute the summary, then extract the R^2. The R^2 value is a statistical measure of how close the data are to the fitted regression line. The R^2 is the percentage of the response variable variation that is explained by a linear model.

Due to the rather low R^2 values found above, the follow up is to ascertain what the potential relationship is at each site only during the prevailing wind/wave directions. Here I create a function which selects the most predominant wind and wave direction at each of the sites.

```
# A function that runs a linear model on each metric of the combined data and get the R^2 value
temp_wave_predom_R2 <- function(df){
  # Find predominant wave direction
  predom_dir <- df %>%
    mutate(dir_round = round(dir_mean, -1)) %>%
    group_by(cluster, site, dir_round) %>%
    summarise(dir_mean_n = n()) %>%
    na.omit() %>%
    filter(dir_mean_n == max(dir_mean_n)) %>%
    select(-dir_mean_n)
  # Find predominant wind direction
  predom_dirw <- df %>%
    mutate(dirw_round = round(dirw_mean, -1)) %>%
    group_by(cluster, site, dirw_round) %>%
    summarise(dirw_mean_n = n()) %>%
    na.omit() %>%
    filter(dirw_mean_n == max(dirw_mean_n))%>%
    select(-dirw_mean_n)

  # Extract just those values
  ## waves
  df_dir <- df %>%
    mutate(dir_round = round(dir_mean, -1)) %>%
    right_join(predom_dir, by = c("site", "cluster", "dir_round")) %>%
```

```r
    select(-dir_round) %>%
    temp_wave_R2() %>%
    right_join(predom_dir, by = c("site", "cluster")) %>%
    dplyr::rename(predom = dir_round) %>%
    mutate(predom_var = "dir")
  ## wind
  df_dirw <- df %>%
    mutate(dirw_round = round(dirw_mean, -1)) %>%
    right_join(predom_dirw, by = c("site", "cluster", "dirw_round")) %>%
    select(-dirw_round) %>%
    temp_wave_R2() %>%
    right_join(predom_dirw, by = c("site", "cluster")) %>%
    dplyr::rename(predom = dirw_round) %>%
    mutate(predom_var = "dirw")

  # Wrap it up
  results <- rbind(df_dir, df_dirw)
  return(results)
}

SACTN_7_predom_R2 <- SACTN_daily_clusters_7 %>%
  temp_wave_predom_R2()
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped

## Warning: attributes are not identical across measure variables;
## they will be dropped
```

```r
SACTN_15_predom_R2 <- SACTN_daily_clusters_15 %>%
  temp_wave_predom_R2()
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped

## Warning: attributes are not identical across measure variables;
## they will be dropped
```

```r
# save(SACTN_7_predom_R2, file = "data/SACTN_7_predom_R2.RData")
# save(SACTN_15_predom_R2, file = "data/SACTN_15_predom_R2.RData")

load("../data/SACTN_7_predom_R2.RData")
load("../data/SACTN_15_predom_R2.RData")
```

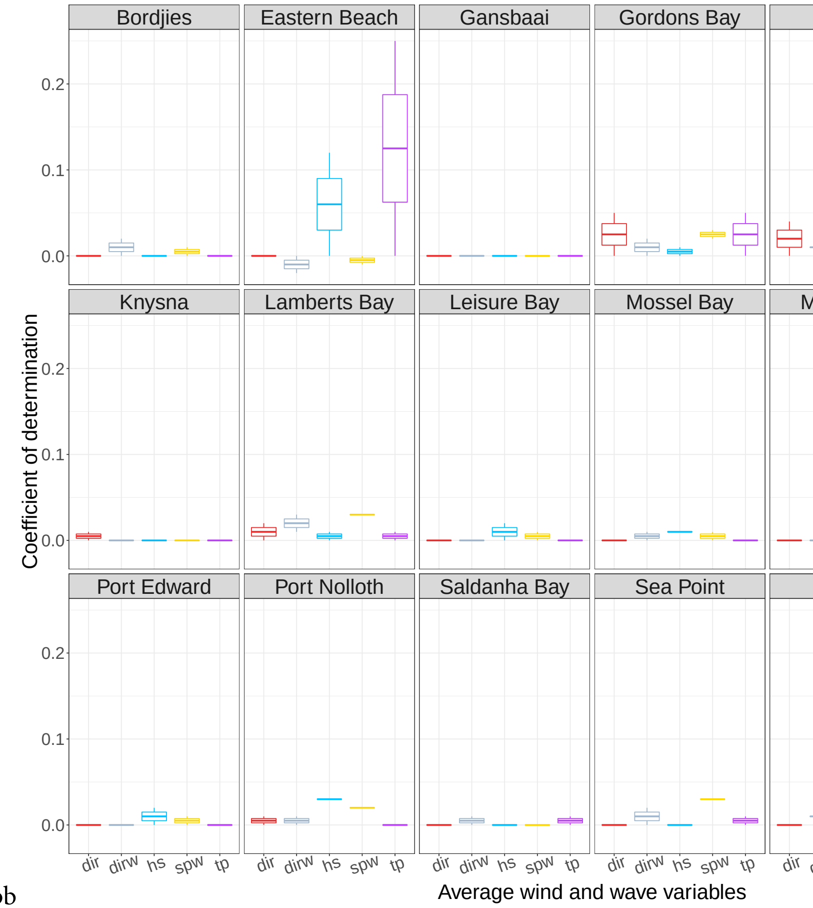
Now I make a visualisation to reveal the R^2 values obtained when comparing the most prevailing wind and wave direction on temperature.

```r
addline_format <- function(x,...){
  gsub('-', '\n', x)
}

renamed_SACTN_7_predom_R2 <- SACTN_7_predom_R2 %>%
  dplyr::rename(Variables = variable)

predominant_ww <- ggplot(data = renamed_SACTN_7_predom_R2, aes(x = addline_format(Variables), y = adj.r.squared)) +
  geom_boxplot(aes(colour = Variables)) +
```

```
  facet_wrap(~site, ncol = 6, nrow = 3) +
  labs(y = "Coefficient of determination", x = "Average wind and wave variables") +
  scale_x_discrete(labels = c('dir','dirw','hs', 'spw', 'tp')) +
  scale_color_manual(labels = c("dir", "dirw", "hs", "spw","tp"), values = c("firebrick2", "slate gray3","deepskyblue","gold
  theme_bw() +
  theme(strip.text.x = element_text(size = 25)) +
  theme(axis.text.x = element_text(angle = 20)) +
      theme(axis.text = element_text(size = 20),
          axis.title = element_text(size = 25),
          legend.text = element_text(size = 20),
          legend.title = element_text(size = 25))
predominant_ww
```



representing R2 values-1.pdf representing R2 values-1.bb

I now create a wind and wave rose diagram. Wind and wave diagrams help visualise the patterns present at a particular site. Moving outward on the radial scale, the frequency associated with wind and waves coming from a partiular direction increases. Here I create a function to create snooty wind and wave plots. As before the functions are ran in the setup chunk of code at the begining of this document.

```
# source("functions/wind.rose.R")

# Now for creating the wind rose diagram

wave_daily_renamed <- wave_daily %>%
  dplyr::rename(spd = spw_mean) %>%
  dplyr::rename(dir = dirw_mean)

p.wr2 <- plot.windrose(data = wave_daily_renamed,
              spd = "spd",
              dir = "dir")
```

```
## Hadley broke my code
```

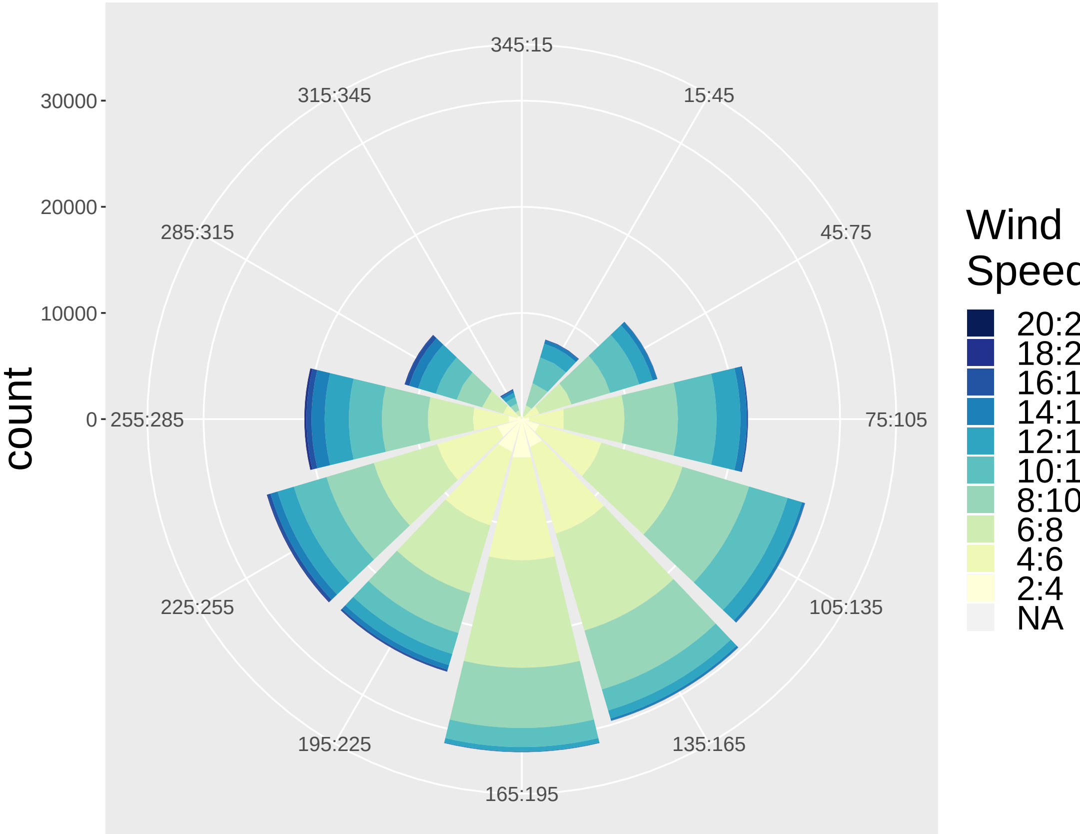


diagram -1.pdf diagram -1.bb

```r
p.wr3 <- p.wr2 + facet_wrap(.~ site, ncol = 4, nrow = 5) +
  theme(strip.text.x = element_text(size = 25))
p.wr3
```

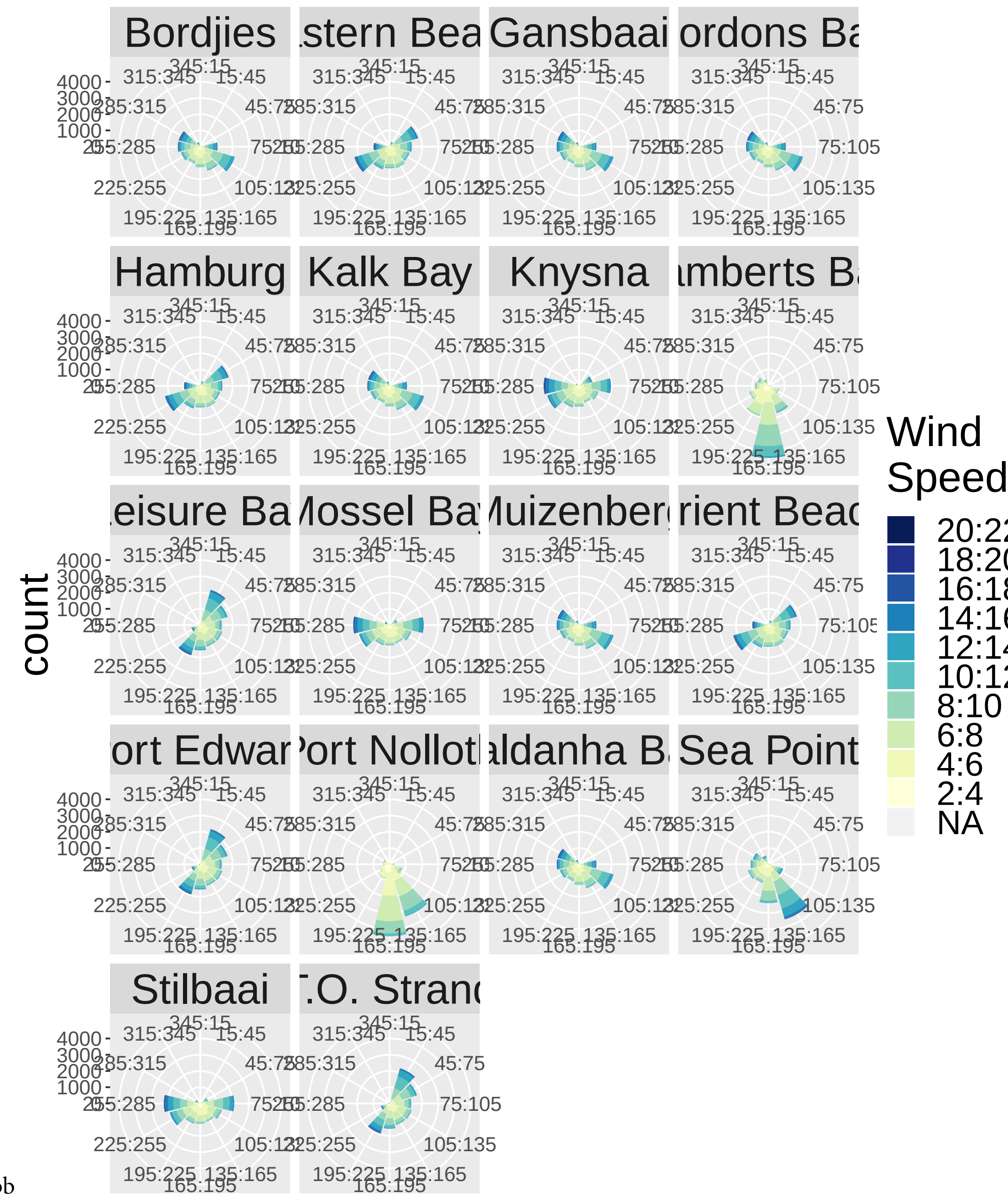


diagram -2.pdf diagram -2.bb

```r
# ggsave(plot = p.wr3, filename = "figures/p.wr3.png")
```

Now for visualising the wave rose diagram

```r
# source("functions/waverose.diagram.R")

# Now for creating the wave rose diagram

wave_daily_renamed <- wave_daily %>%
  dplyr::rename(spd = hs_mean) %>%
  dplyr::rename(dir = dir_mean)

p.wr2 <- plot.waverose(data = wave_daily_renamed,
              spd = "spd",
              dir = "dir")
```

```
## Warning in brewer.pal(min(max(0, n.colors.in.range), min(2, n.colors.in.range)), : minimal value for n is 3, returning rec
```

```
## Hadley broke my code
```

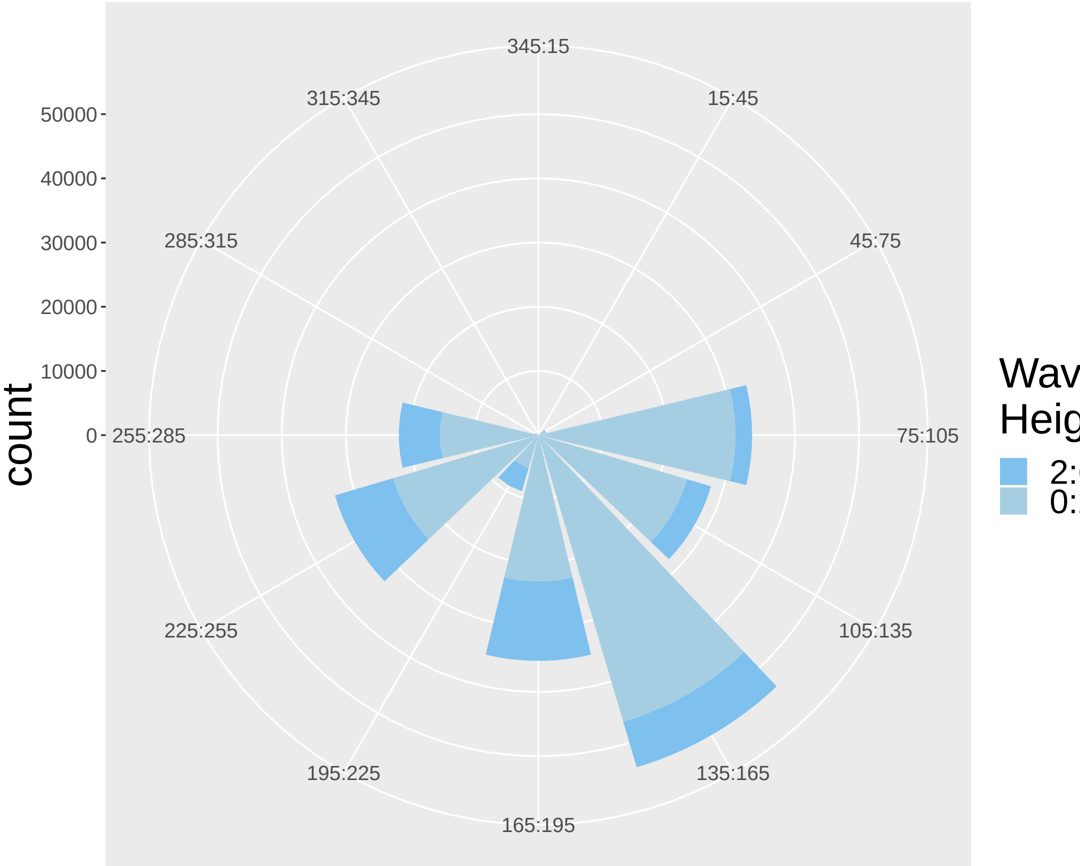


diagram -1.pdf diagram -1.bb

```
p.wr2
```

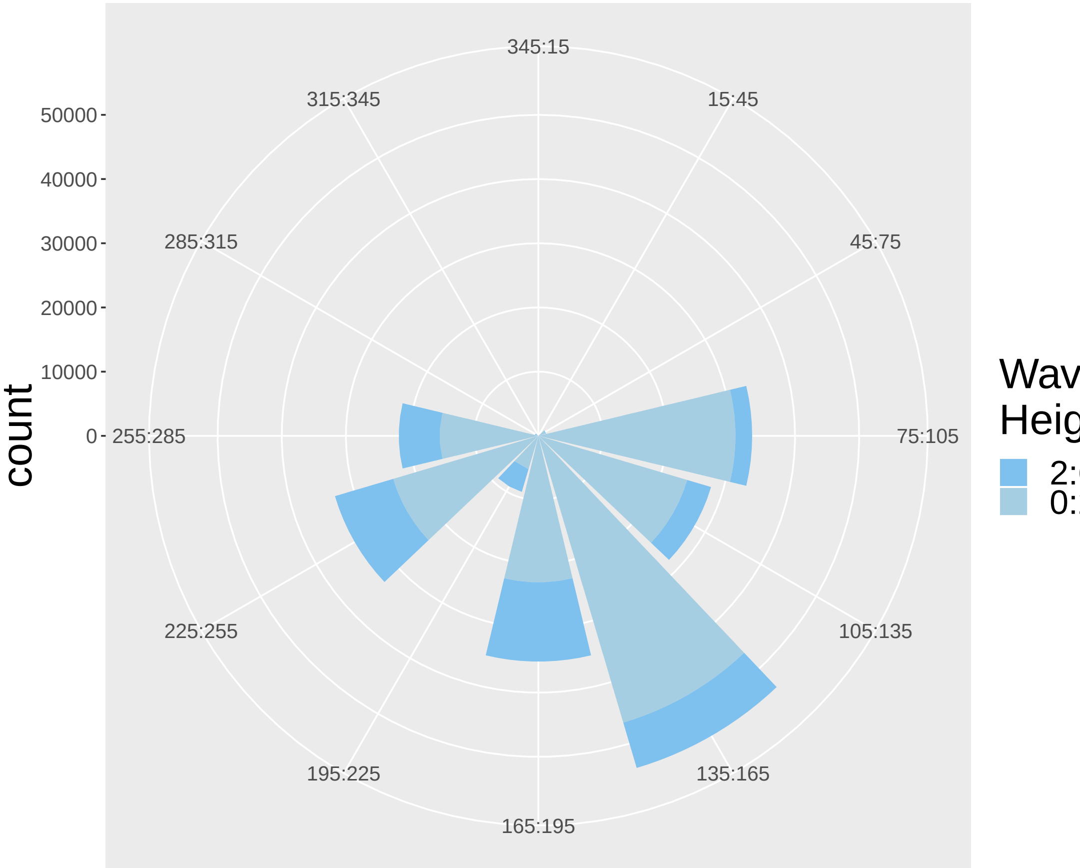


diagram -2.pdf diagram -2.bb

```r
p.wave <- p.wr2 + facet_wrap(~site,
                             ncol = 6, nrow = 3)
p.wave
```

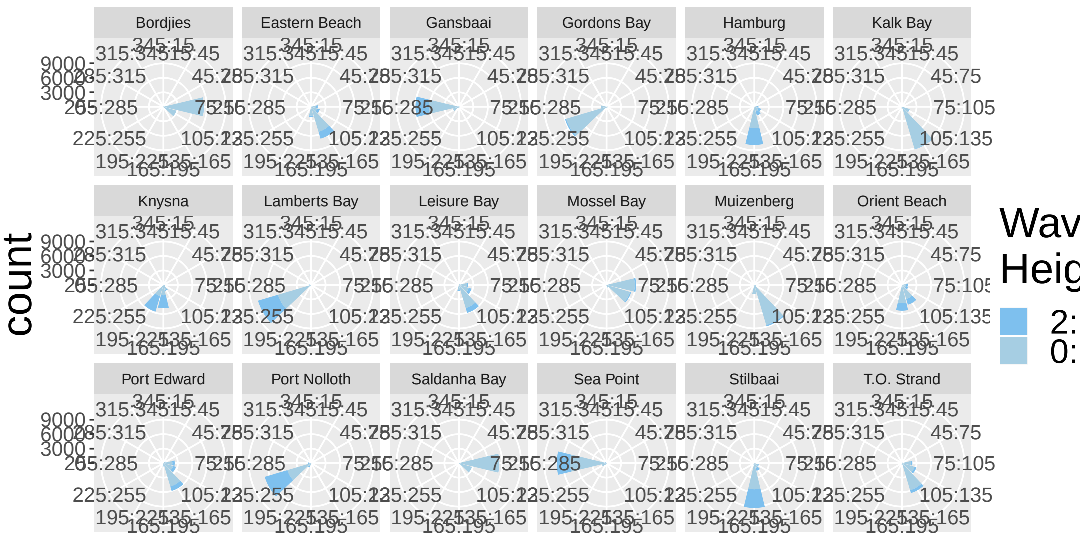


diagram -3.pdf diagram -3.bb

```r
# ggsave(plot = p.wave, filename = "figures/p.wave.pdf")
```

Mead, A, CL Griffiths, GM Branch, CD McQuaid, LK Blamey, JJ Bolton, RJ Anderson, et al. 2013. "Human-Mediated Drivers of Change—impacts on Coastal Ecosystems and Marine Biota of South Africa." *African Journal of Marine Science* 35 (3). Taylor & Francis: 403–25.

Smit, Albertus J, John J Bolton, and Robert J Anderson. 2017. "Seaweeds in Two Oceans: Beta-Diversity." *Frontiers* *in Marine Science* 4. Frontiers: 404.

Smit, Albertus J, Michael Roberts, Robert J Anderson, Francois Dufois, Sheldon FJ Dudley, Thomas G Bornman, Jennifer Olbers, and John J Bolton. 2013. "A Coastal Seawater Temperature Dataset for Biogeographical Studies: Large Biases Between in Situ and Remotely-Sensed Data Sets Around the Coast of South Africa." *PLoS One* 8 (12). Public Library of Science: e81944.