coral_bleaching_V2

2022-11-22

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.5
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
##
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
library(stringr)
library(maptools)
```

```
## Loading required package: sp
## Checking rgeos availability: TRUE
## Please note that 'maptools' will be retired during 2023,
## plan transition at your earliest convenience;
## some functionality will be moved to 'sp'.
```

```
library(maps)
```

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

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
df <- read_csv("V2-global-bleaching-and-environmental-data.csv")
```

```
## Warning: One or more parsing issues, call `problems()` on your data frame for details,
## e.g.:
##   dat <- vroom(...)
##   problems(dat)
```

```
## Rows: 41361 Columns: 62
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (52): Data_Source, Ocean_Name, Reef_ID, Realm_Name, Ecoregion_Name, Cou...
## dbl   (9): Site_ID, Sample_ID, Latitude_Degrees, Longitude_Degrees, Turbidit...
## date  (1): Date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
df <- df[-1, ]
head(df)
```

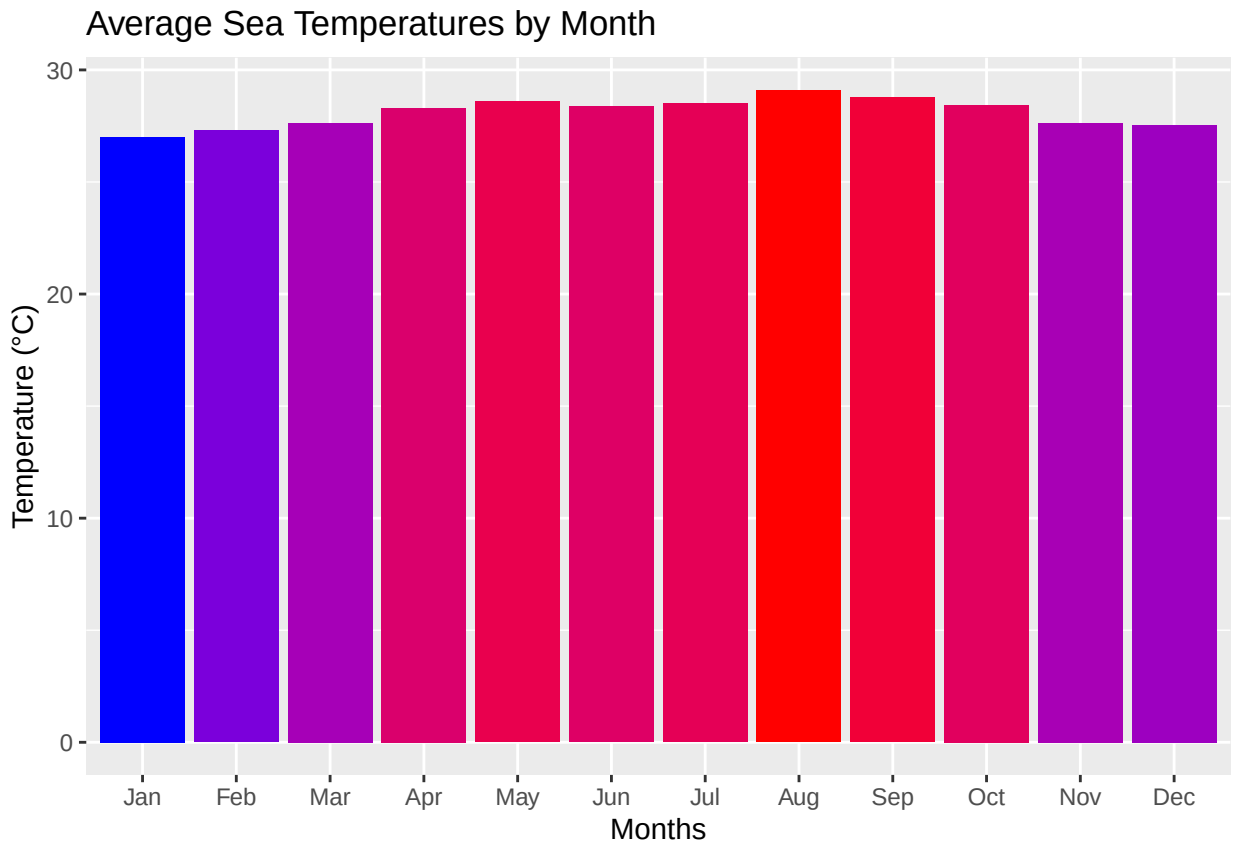
```
## # A tibble: 6 x 62
##   Site_ID Sample_ID Data_Source Latitu~1 Longi~2 Ocean~3 Reef_ID Realm~4 Ecore~5
##     <dbl>     <dbl> <chr>          <dbl>   <dbl> <chr>   <chr>   <chr>   <chr>
## 1    3467  10324754 Donner        -17.6   -150.  Pacific nd      Easter~ Societ~
## 2    1794  10323866 Donner         18.4    -64.6 Atlant~ nd      Tropic~ Hispan~
## 3    8647  10328028 Donner         17.8    -64.6 Atlant~ nd      Tropic~ Hispan~
## 4    8648  10328029 Donner         17.8    -64.6 Atlant~ nd      Tropic~ Hispan~
## 5    2180  10324021 Donner          9.82   -75.9 Atlant~ nd      Tropic~ Nether~
## 6    9298  10328657 Donner         17.8    -64.6 Atlant~ nd      Tropic~ Hispan~
## # ... with 53 more variables: Country_Name <chr>,
## #   State_Island_Province_Name <chr>, City_Town_Name <chr>, Site_Name <chr>,
## #   Distance_to_Shore <chr>, Exposure <chr>, Turbidity <dbl>,
## #   Cyclone_Frequency <dbl>, Date_Day <dbl>, Date_Month <dbl>, Date_Year <dbl>,
## #   Depth_m <chr>, Substrate_Name <chr>, Percent_Cover <chr>,
## #   Bleaching_Level <chr>, Percent_Bleaching <chr>, ClimSST <chr>,
## #   Temperature_Kelvin <chr>, Temperature_Mean <chr>, ...
```

```
df$month <- month(df$Date)
```

```
df$Temperature_Celcius <- as.double(df$Temperature_Kelvin) - 273
```

```
## Warning: NAs introduced by coercion
```

```
df %>% mutate(month2 = recode(month,"Jan","Feb", "Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","Nov",
  geom_bar(stat="identity") + theme(legend.position = 'none') +
  scale_fill_gradientn(colors=c("blue", "red")) + xlab("Months") + ylab("Temperature (°C)") + ggtitle("
```




```
df$Longitude_Degrees <- round((df$Longitude_Degrees), 0)
df$Latitude_Degrees <- round((df$Latitude_Degrees), 0)
```

```
df$bleach_per <- as.double(df$Percent_Bleaching)
```

```
## Warning: NAs introduced by coercion
```

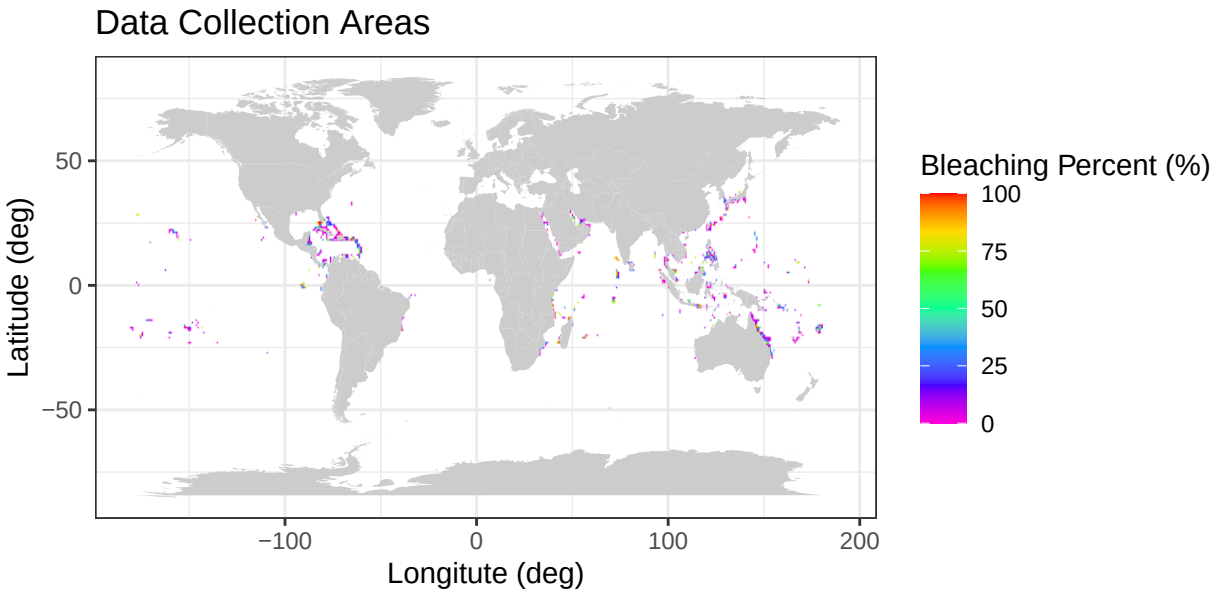
```
df$dis_shore <- as.double(df$Distance_to_Shore)
```

```
## Warning: NAs introduced by coercion
```

```
world <- map_data("world")


df %>% ggplot() + geom_raster(aes(x = Longitude_Degrees, y = Latitude_Degrees, fill=bleach_per), interp
```

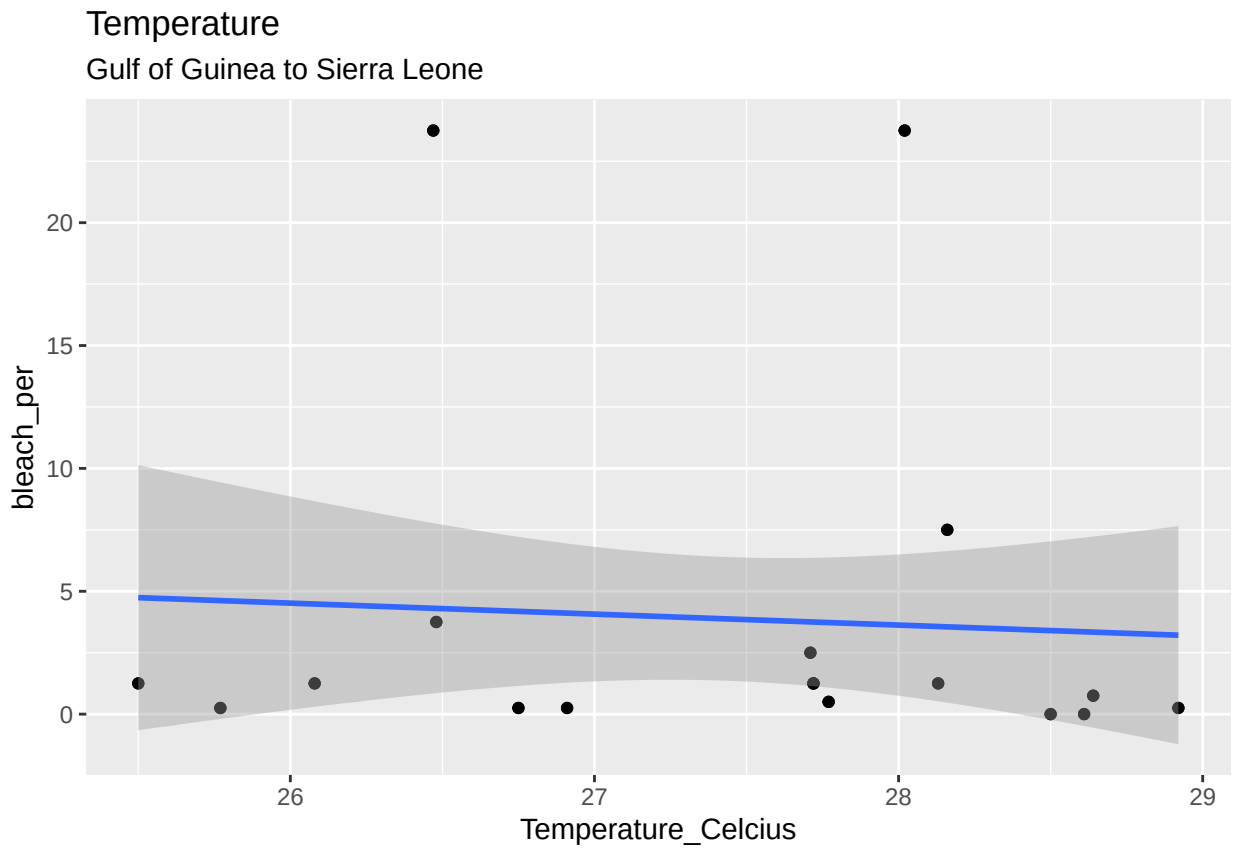
```
## Warning: Raster pixels are placed at uneven horizontal intervals and will be
## shifted. Consider using geom_tile() instead.
```

```
## Warning: Removed 6846 rows containing missing values (geom_raster).
```

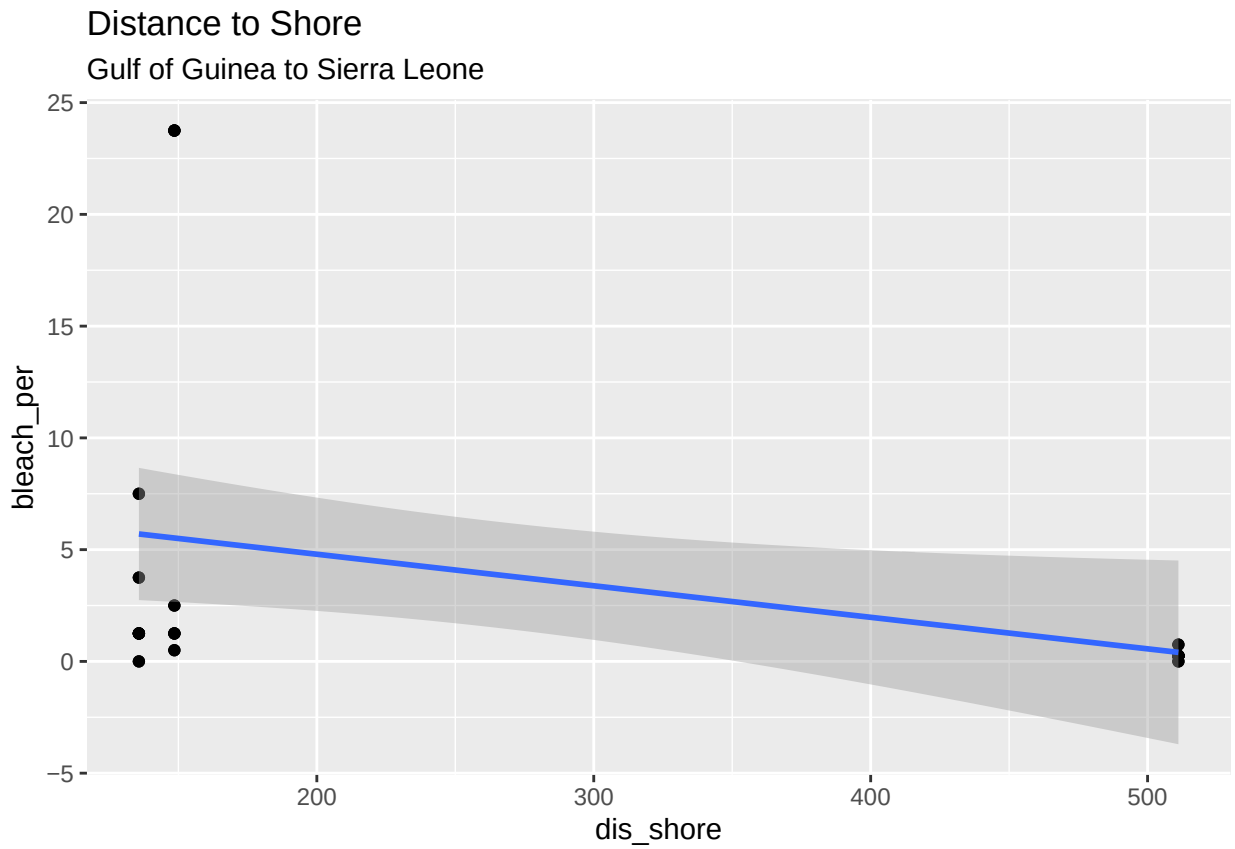



```
uni <- unique(df$Ecoregion_Name)

rand_uni <- sample(uni, 8, replace=FALSE)

for (i in rand_uni) {

  df_new <- df %>% filter(Ecoregion_Name == i)
  print(ggplot(df_new, aes(x=Temperature_Celcius, y=bleach_per)) + geom_point() + geom_smooth(method="lm
        + ggtitle("Temperature", i))

  print(ggplot(df_new, aes(x=dis_shore, y=bleach_per)) + geom_point() + geom_smooth(method="lm")
        + ggtitle("Distance to Shore", i))

  ml1 <- lm(Temperature_Celcius~bleach_per, data=df_new)
  r_sq1 <- summary(ml1)$r.squared
  print("Temp")
  print(r_sq1)

  ml2 <- lm(dis_shore~bleach_per, data=df_new)
  r_sq2 <- summary(ml2)$r.squared
  print("Dist")
  print(r_sq2)

}
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
## `geom_smooth()` using formula 'y ~ x'
```

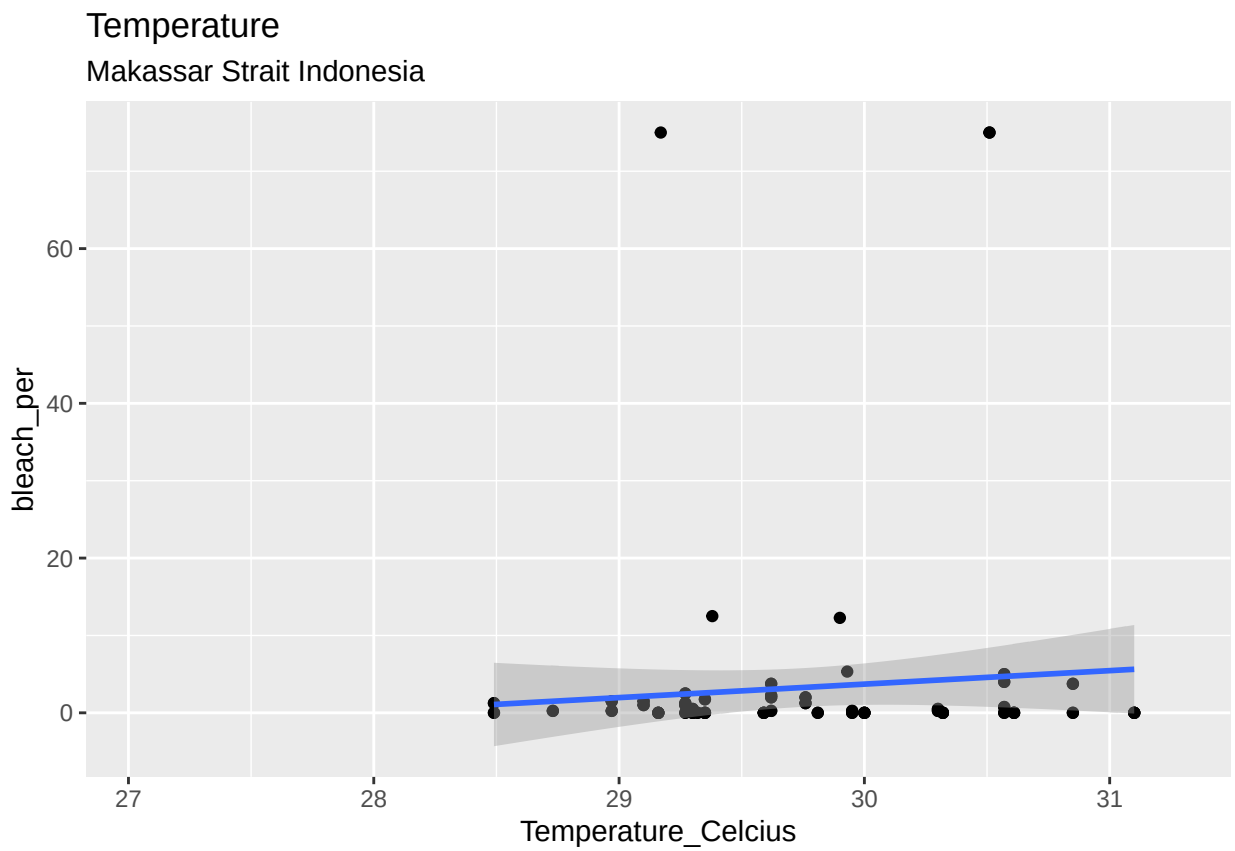



```
## [1] "Temp"
## [1] 0.003980684
## [1] "Dist"
## [1] 0.115011
```

```
## `geom_smooth()` using formula 'y ~ x'
```

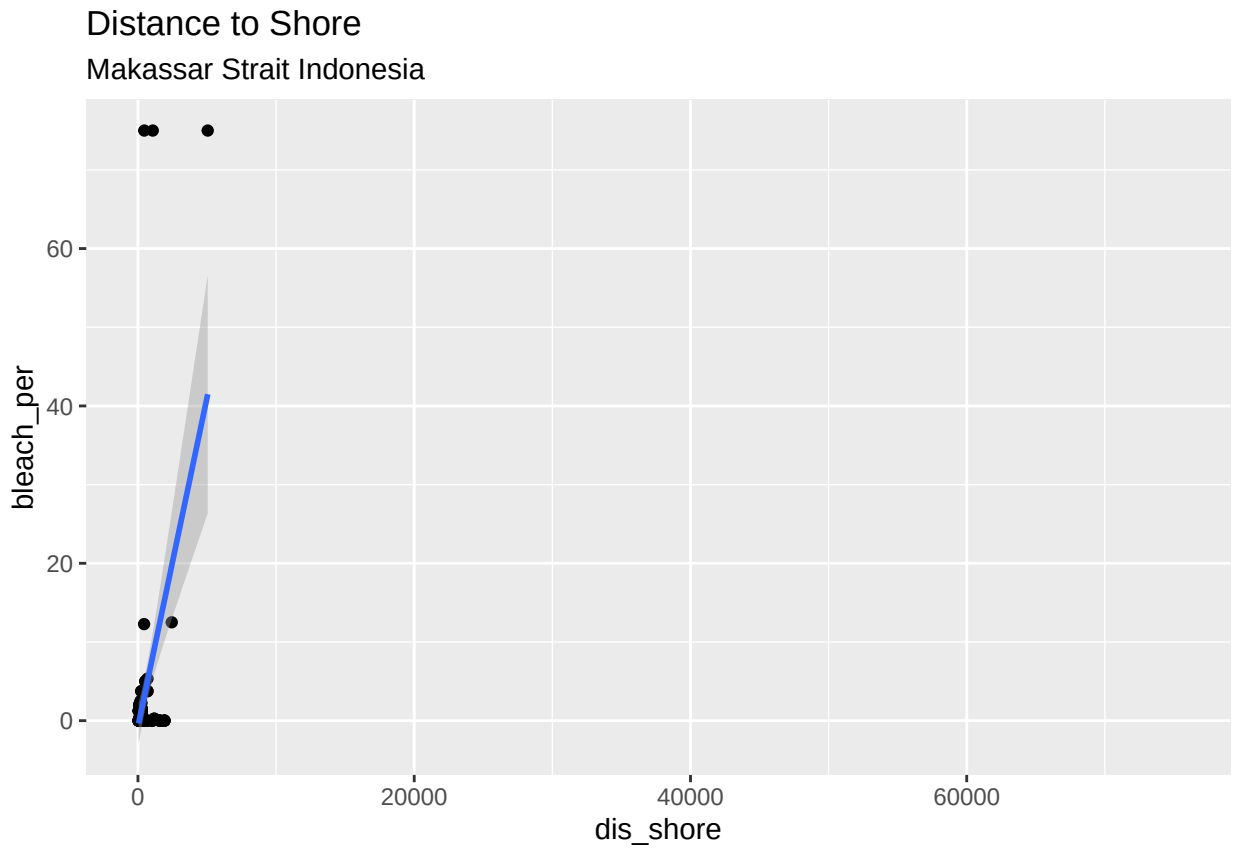
```
## Warning: Removed 164 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 164 rows containing missing values (geom_point).
```




```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 164 rows containing non-finite values (stat_smooth).
## Removed 164 rows containing missing values (geom_point).
```

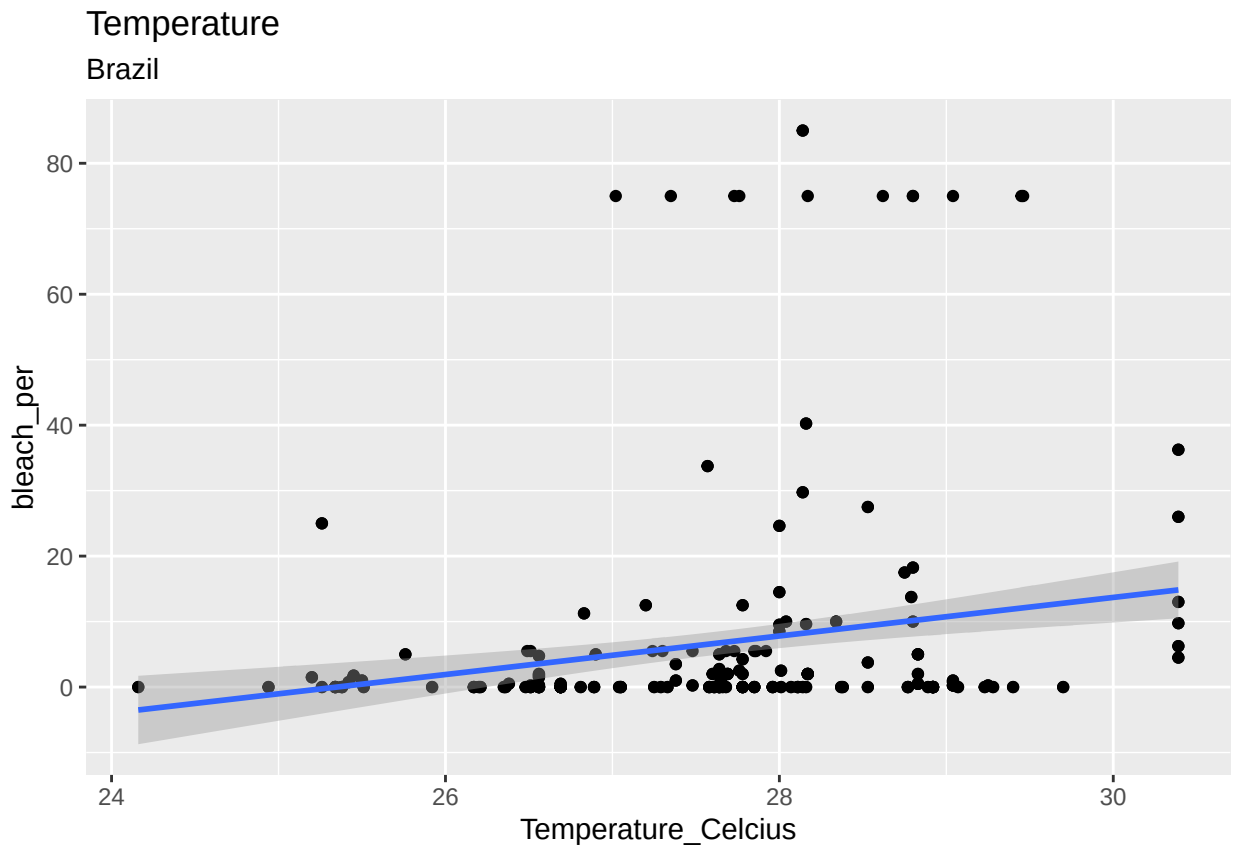



```
## [1] "Temp"
## [1] 0.008196515
## [1] "Dist"
## [1] 0.2043519
```

```
## `geom_smooth()` using formula 'y ~ x'
```

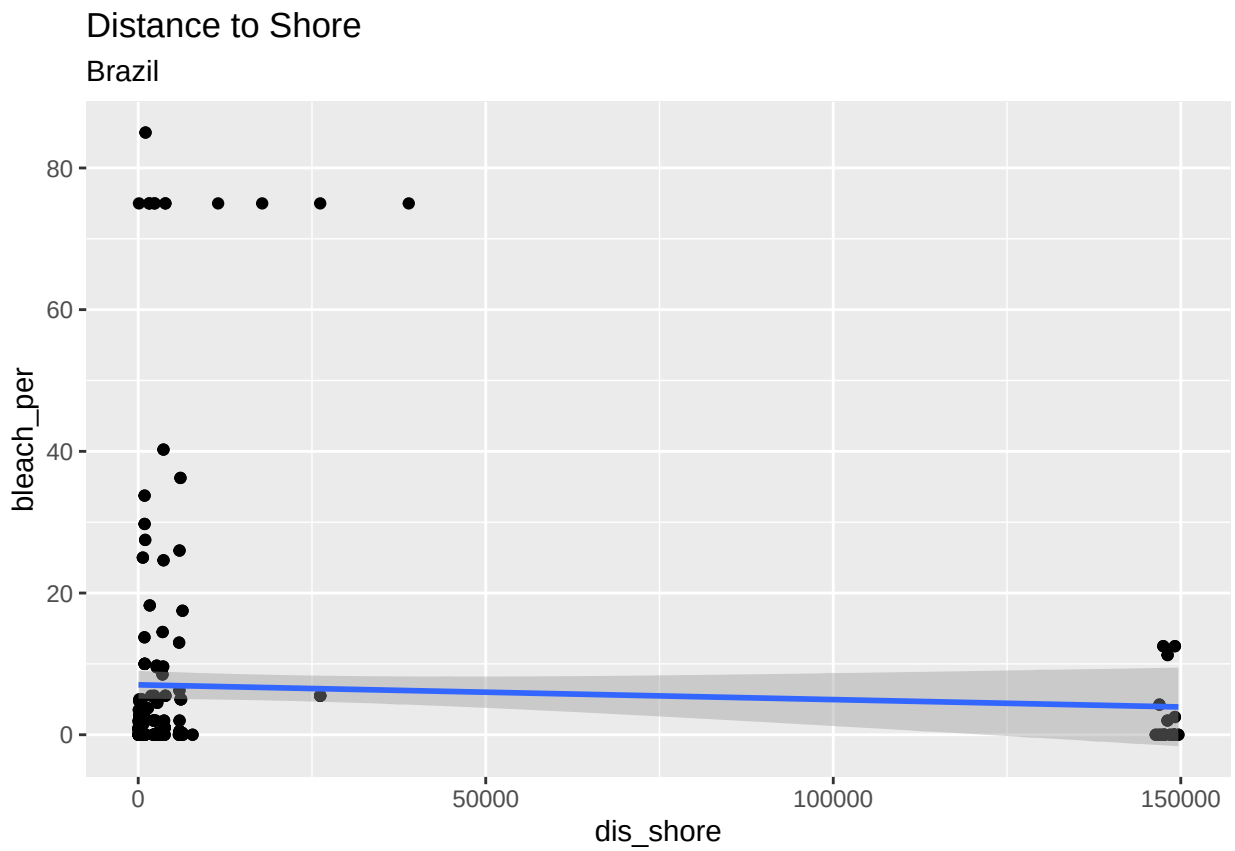
```
## Warning: Removed 71 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 71 rows containing missing values (geom_point).
```




```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 71 rows containing non-finite values (stat_smooth).
## Removed 71 rows containing missing values (geom_point).
```

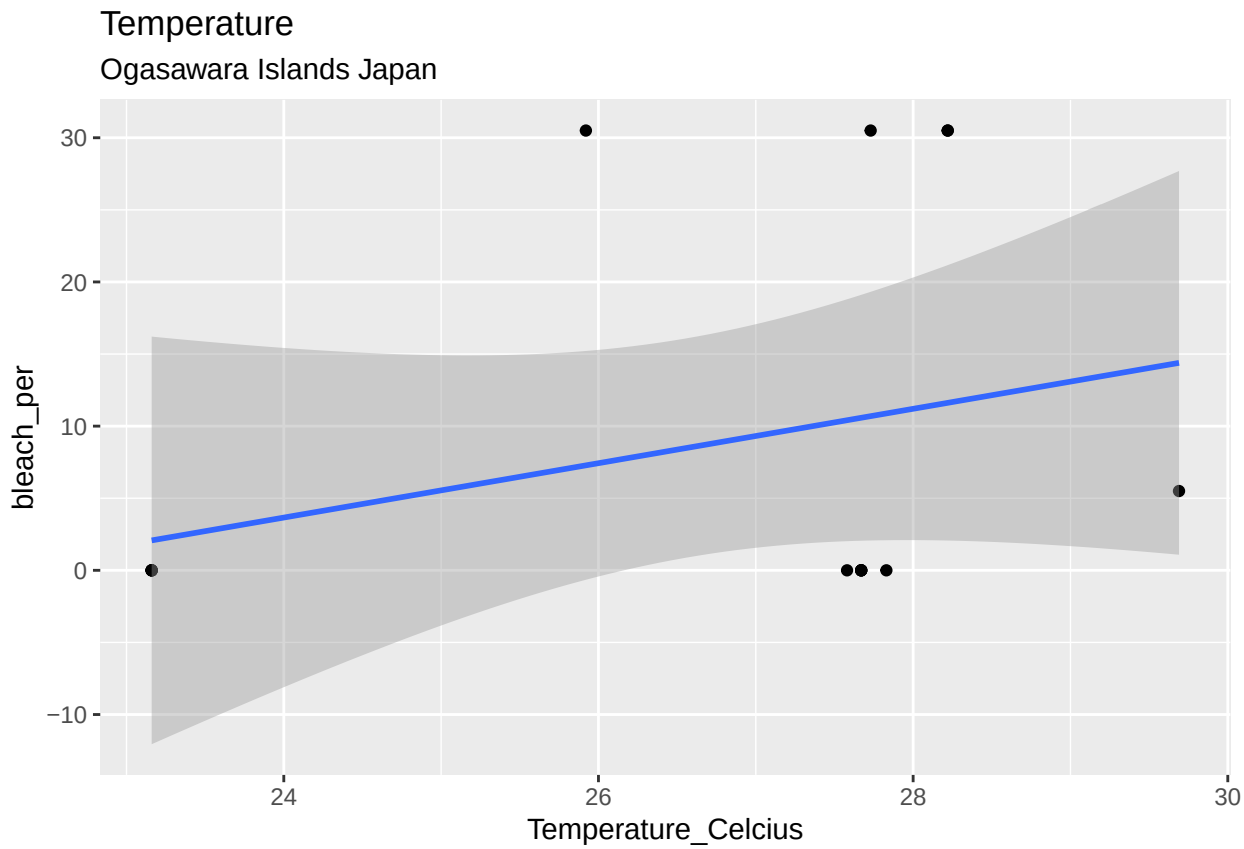



```
## [1] "Temp"
## [1] 0.05049036
## [1] "Dist"
## [1] 0.003378271
```

```
## `geom_smooth()` using formula 'y ~ x'
```

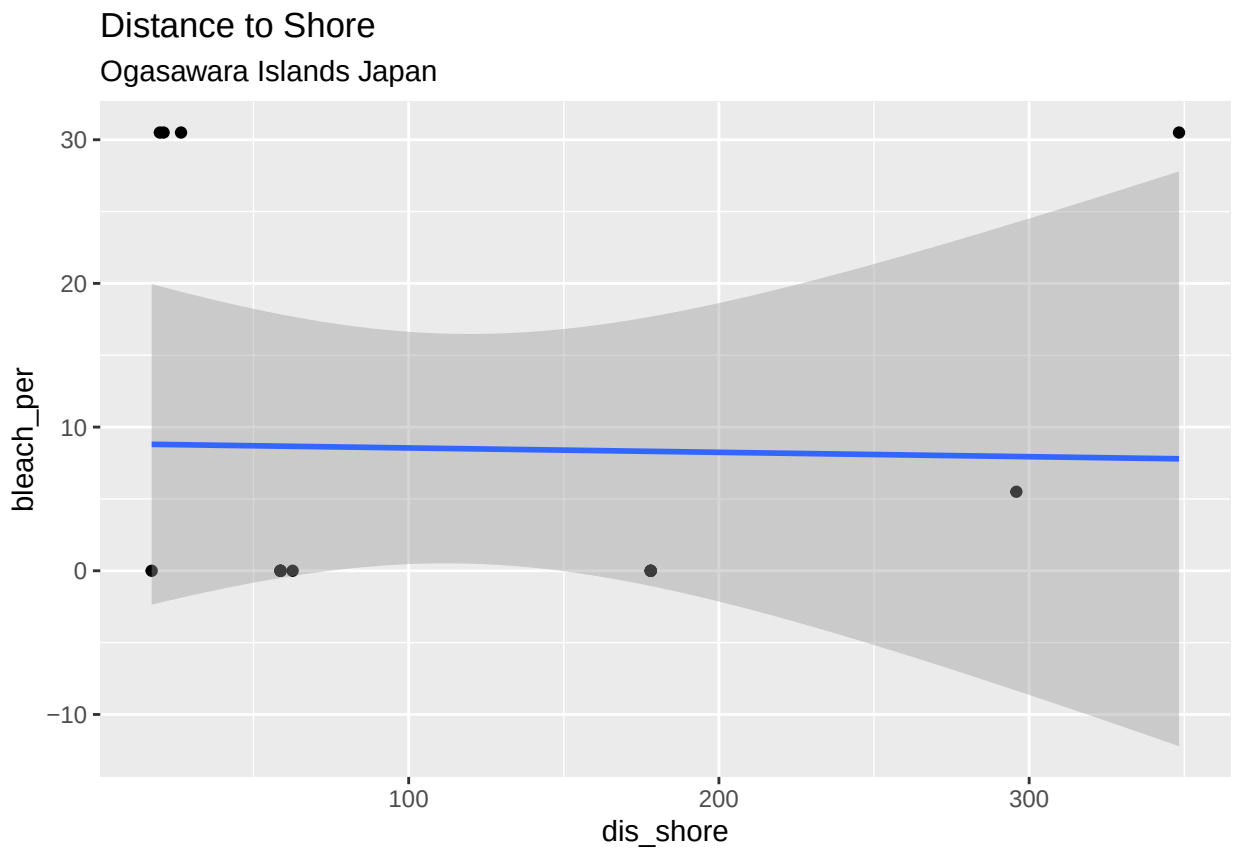
```
## Warning: Removed 12 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 12 rows containing missing values (geom_point).
```




```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 12 rows containing non-finite values (stat_smooth).
## Removed 12 rows containing missing values (geom_point).
```

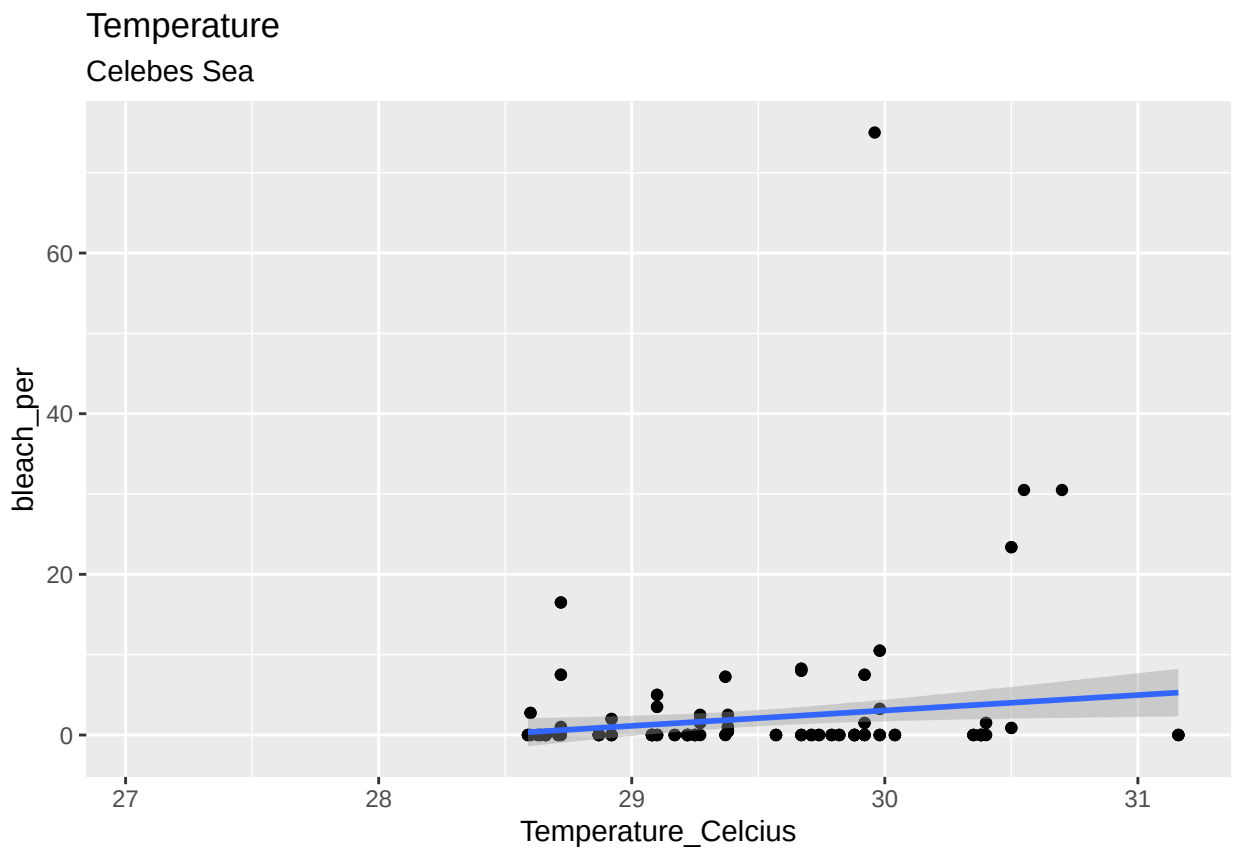



```
## [1] "Temp"
## [1] 0.09451456
## [1] "Dist"
## [1] 0.0005331207
```

```
## `geom_smooth()` using formula 'y ~ x'
```

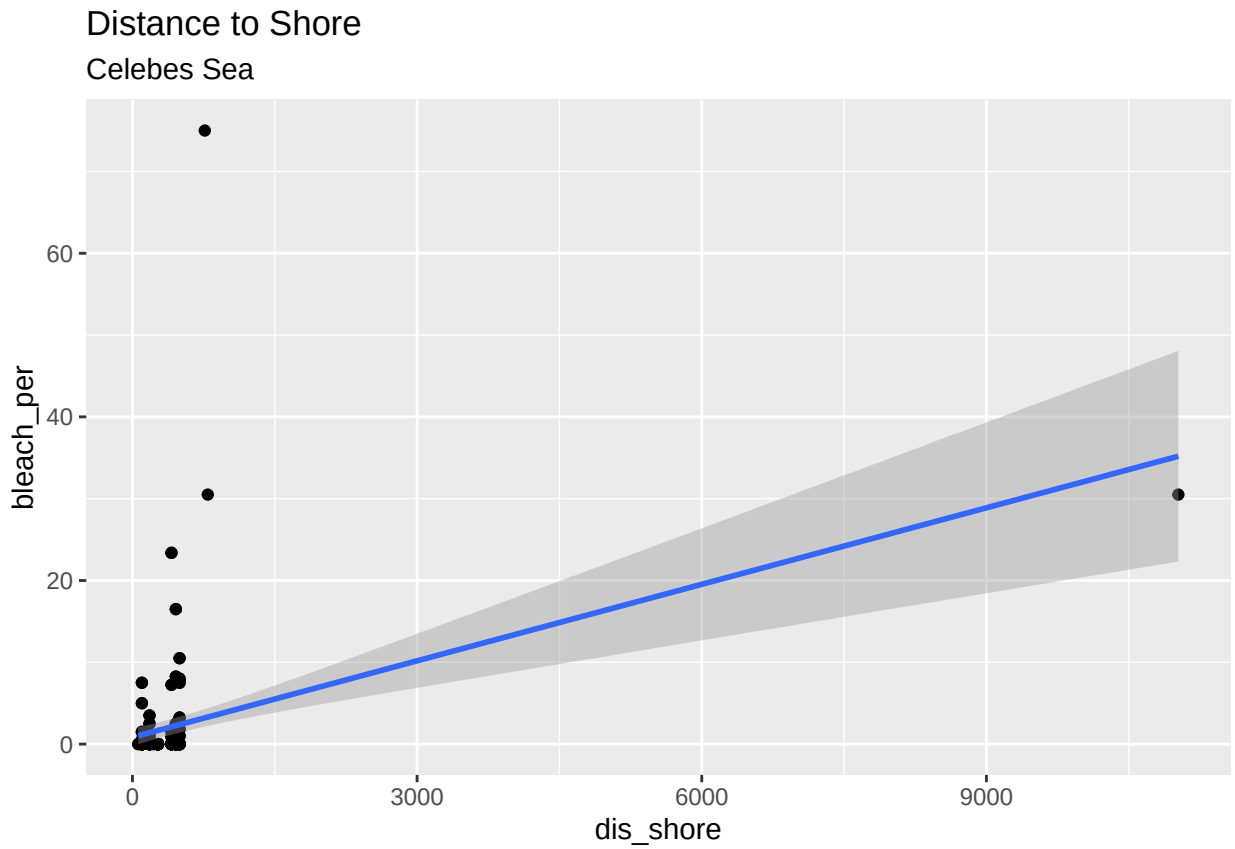
```
## Warning: Removed 22 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 22 rows containing missing values (geom_point).
```




```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 22 rows containing non-finite values (stat_smooth).
## Removed 22 rows containing missing values (geom_point).
```

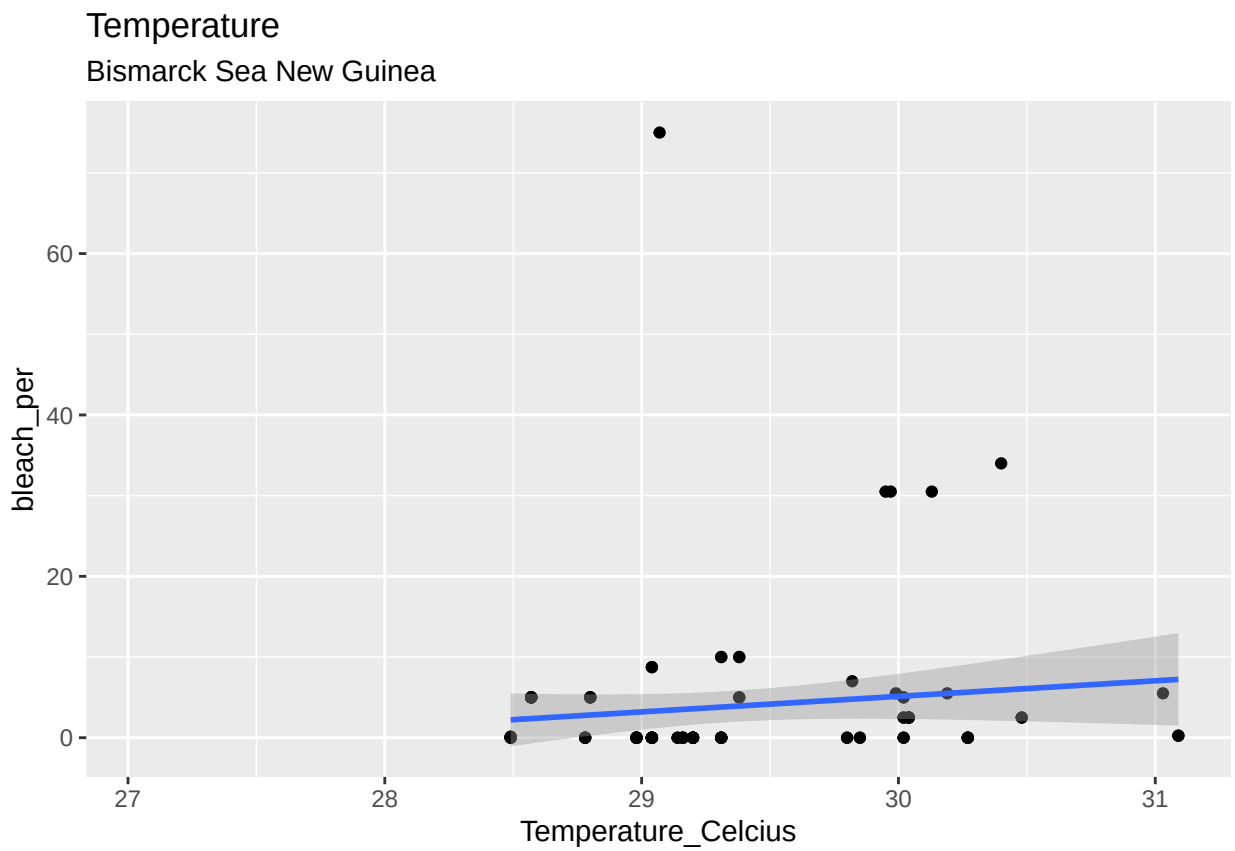



```
## [1] "Temp"
## [1] 0.02810705
## [1] "Dist"
## [1] 0.1232131
```

```
## `geom_smooth()` using formula 'y ~ x'
```

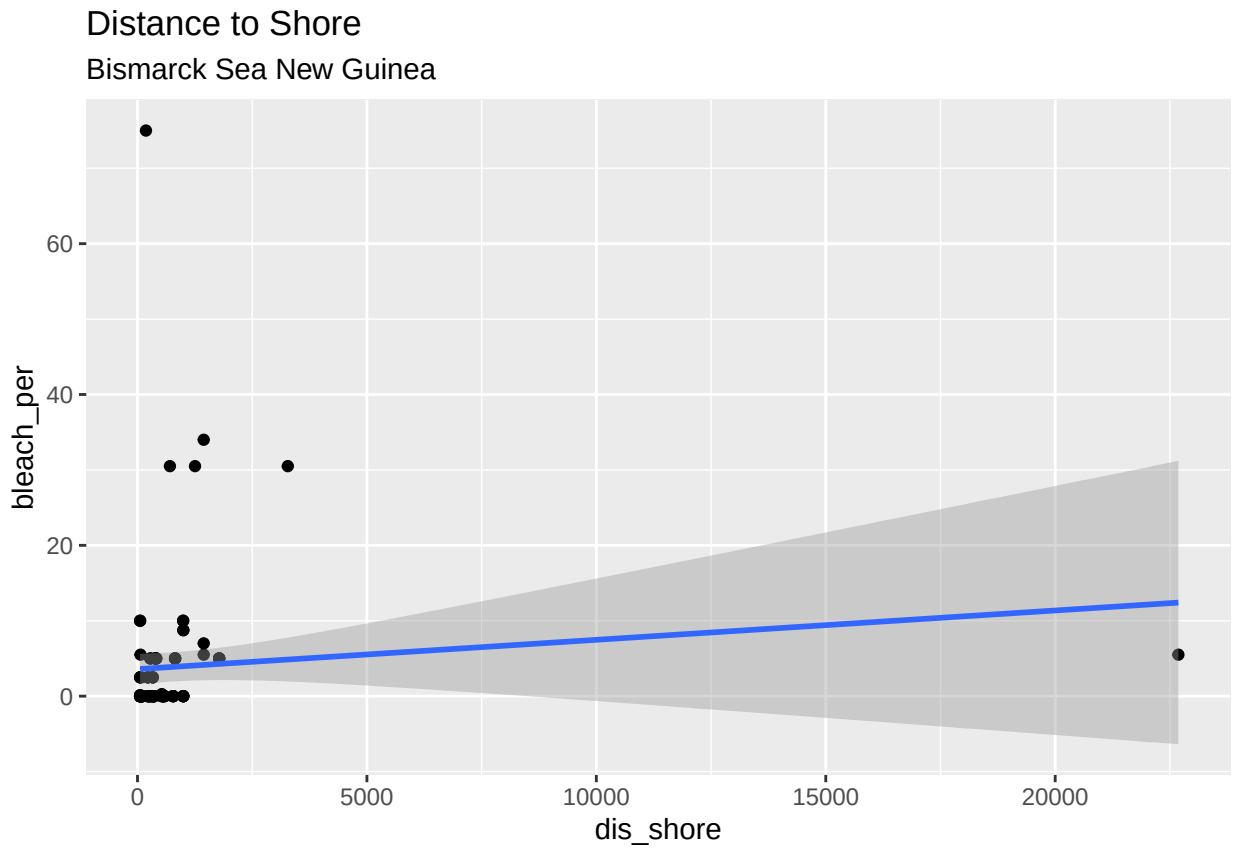
```
## Warning: Removed 86 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 86 rows containing missing values (geom_point).
```



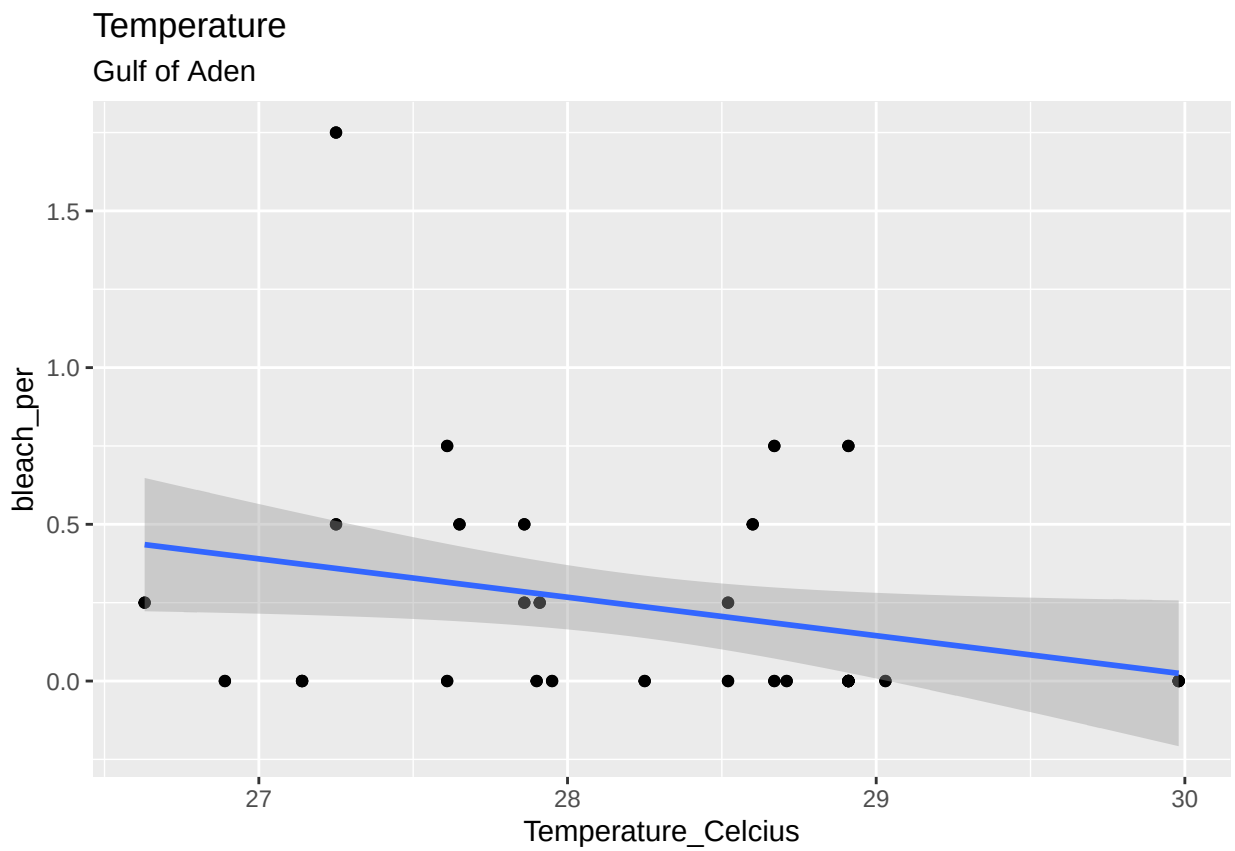

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 86 rows containing non-finite values (stat_smooth).
## Removed 86 rows containing missing values (geom_point).
```

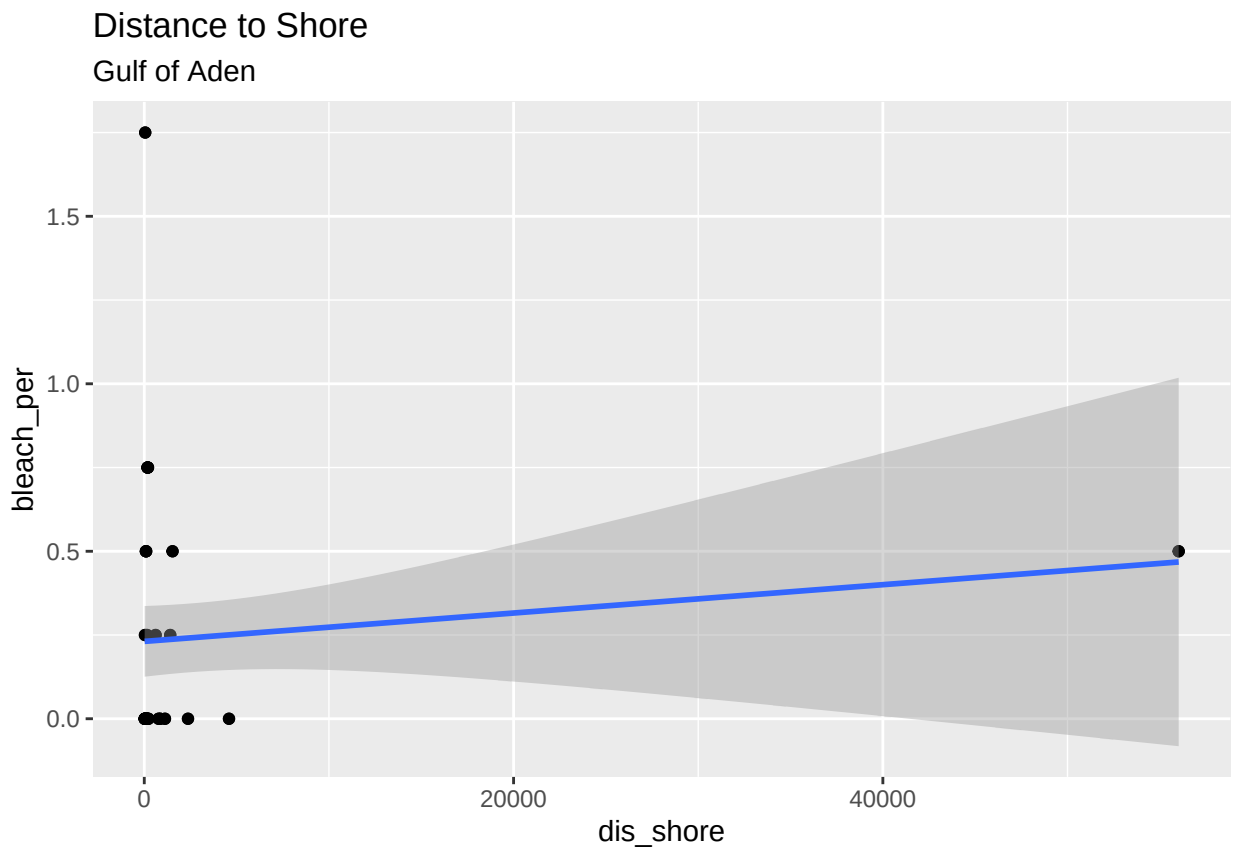



```
## [1] "Temp"
## [1] 0.01555175
## [1] "Dist"
## [1] 0.008423277
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
## `geom_smooth()` using formula 'y ~ x'
```

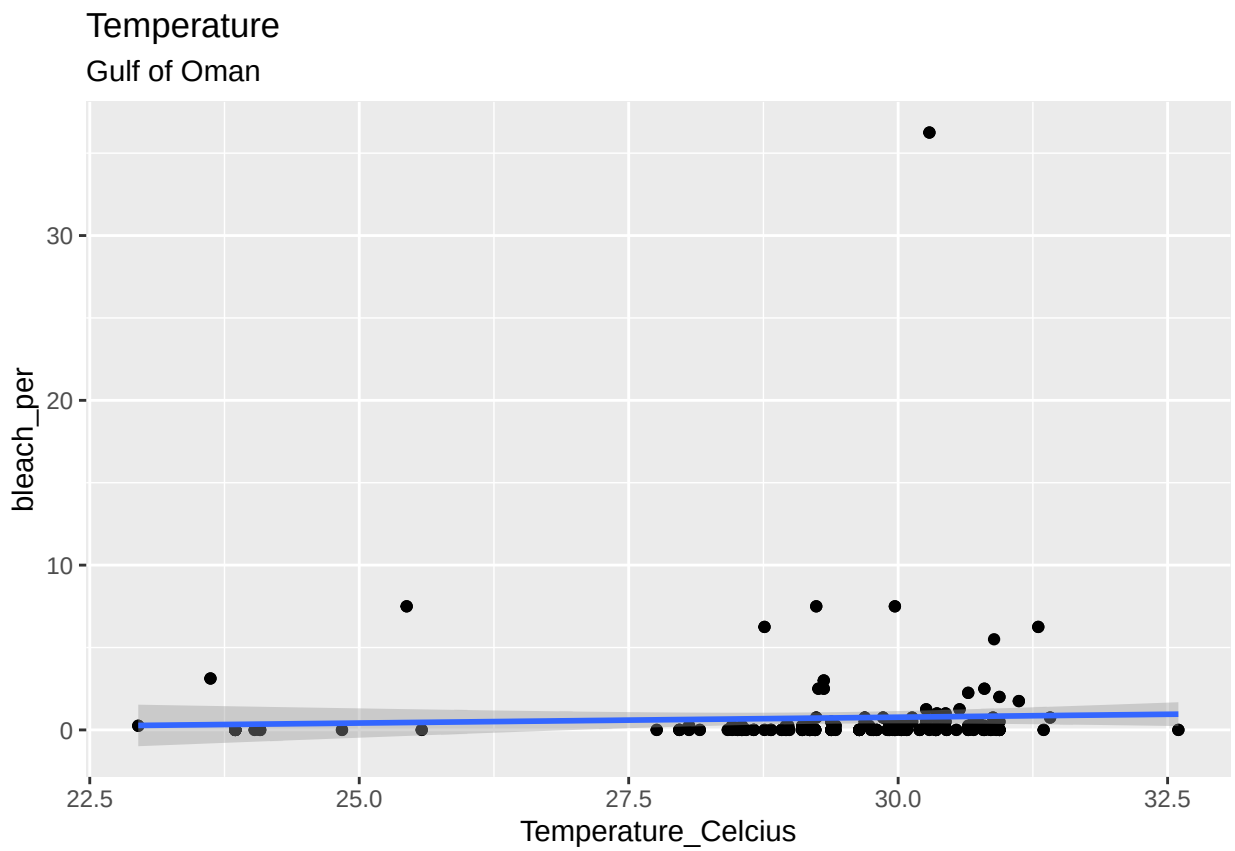



```
## [1] "Temp"
## [1] 0.07079909
## [1] "Dist"
## [1] 0.0124494
```

```
## `geom_smooth()` using formula 'y ~ x'
```

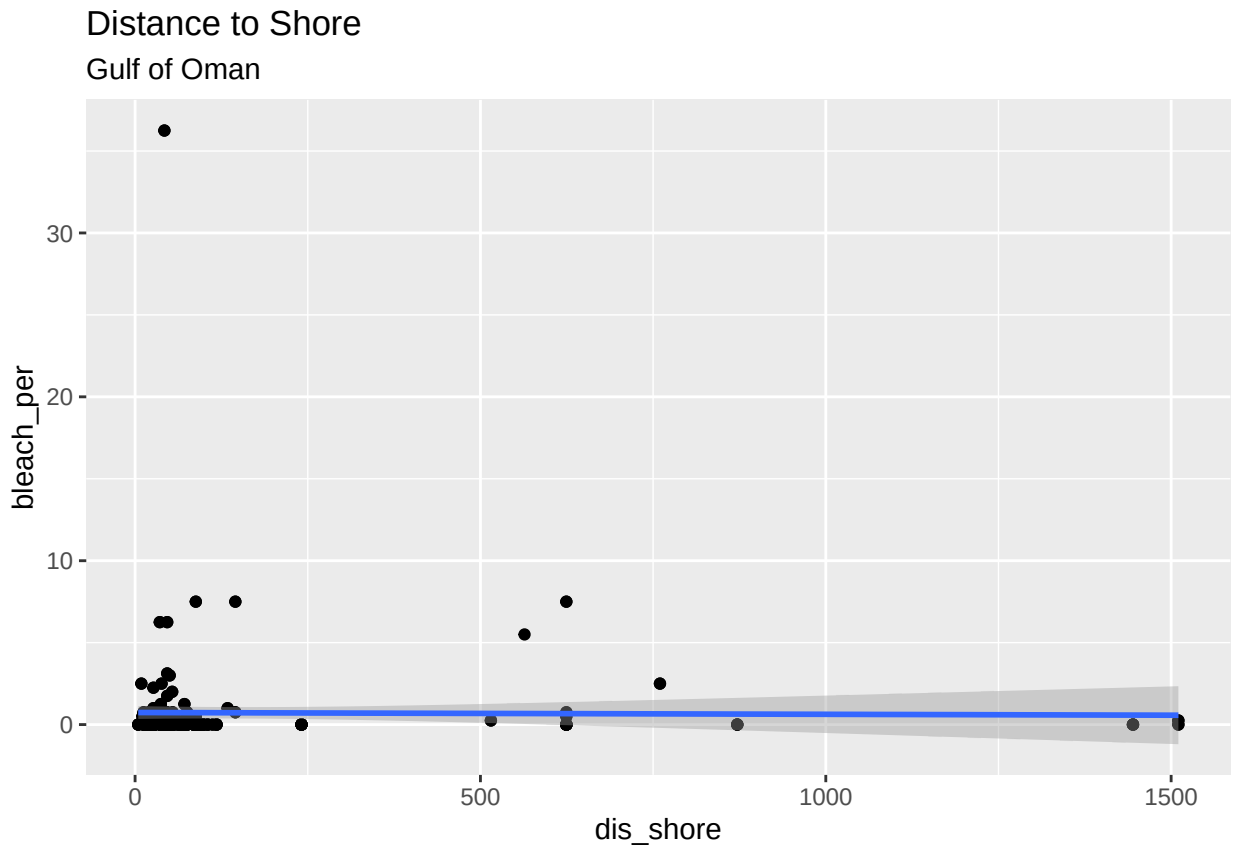
```
## Warning: Removed 9 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 9 rows containing missing values (geom_point).
```




```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 9 rows containing non-finite values (stat_smooth).
## Removed 9 rows containing missing values (geom_point).
```




```
## [1] "Temp"
## [1] 0.001692942
## [1] "Dist"
## [1] 9.099784e-05
```

```
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
library(readr)


#cleaning the graph
update_bleach <- df %>%
  filter(!(Percent_Bleaching=='nd')) %>%
  filter(!(Temperature_Mean=='nd')) %>%
  mutate_at('Percent_Bleaching', as.numeric) %>%
  mutate_at('Temperature_Mean', as.numeric) %>%
  mutate_at('Temperature_Kelvin', as.numeric) %>%
```

```
  mutate_at('Distance_to_Shore', as.numeric)
```

```
## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion
```

```
## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion
```

```
#Computations/Checking my Work
max(update_bleach$Temperature_Mean)
```

```
## [1] 303.52
```

```
min(update_bleach$Temperature_Mean)
```

```
## [1] 290.88
```

```
max(update_bleach$Percent_Bleaching)
```

```
## [1] 100
```

```
min(update_bleach$Percent_Bleaching)
```

```
## [1] 0
```

```
class(update_bleach$Temperature_Mean)
```

```
## [1] "numeric"
```

```
class(update_bleach$Percent_Bleaching)
```

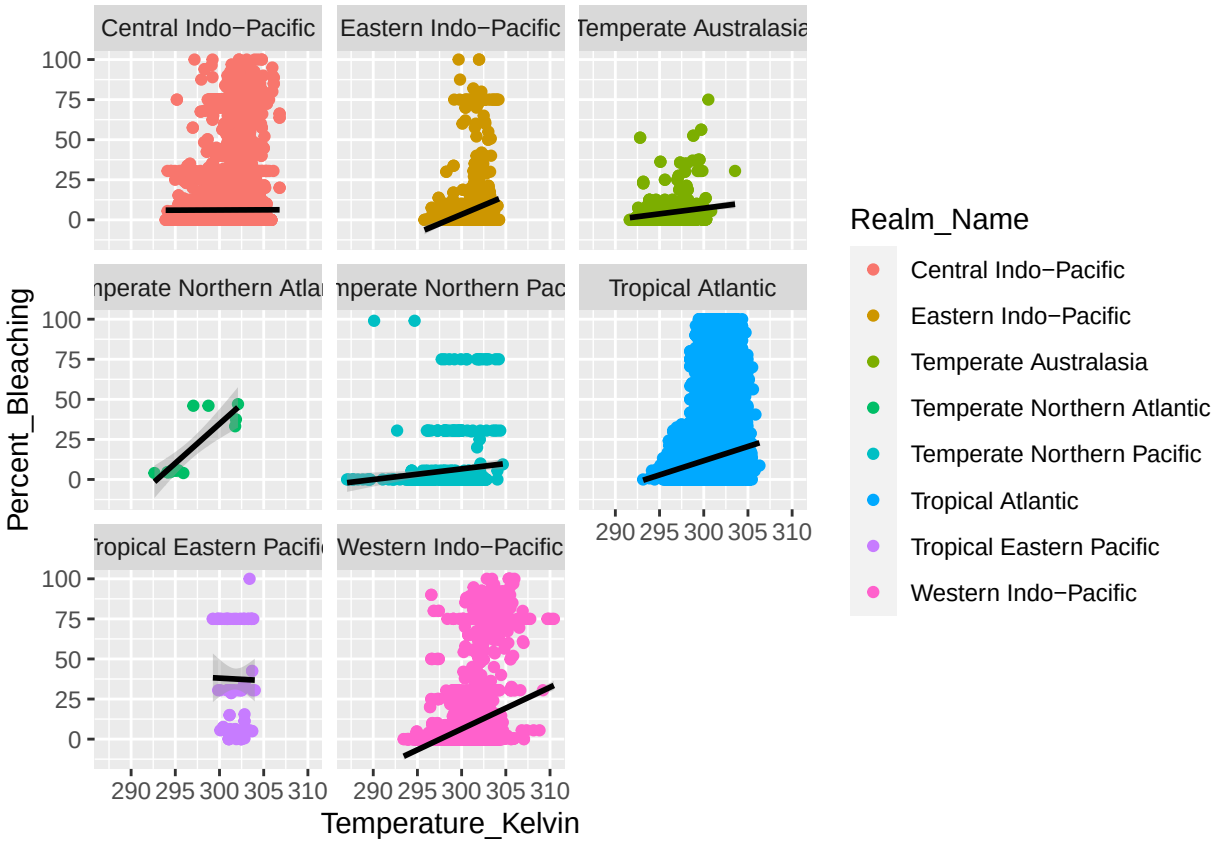
```
## [1] "numeric"
```

```
#Realm_Name Plot (1)
update_bleach %>%
  ggplot(aes(Temperature_Kelvin,Percent_Bleaching,color=Realm_Name)) +
  geom_point() +
  geom_smooth(method="lm",color="black") +
  scale_x_continuous(n.break=6) +
  facet_wrap(~Realm_Name)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

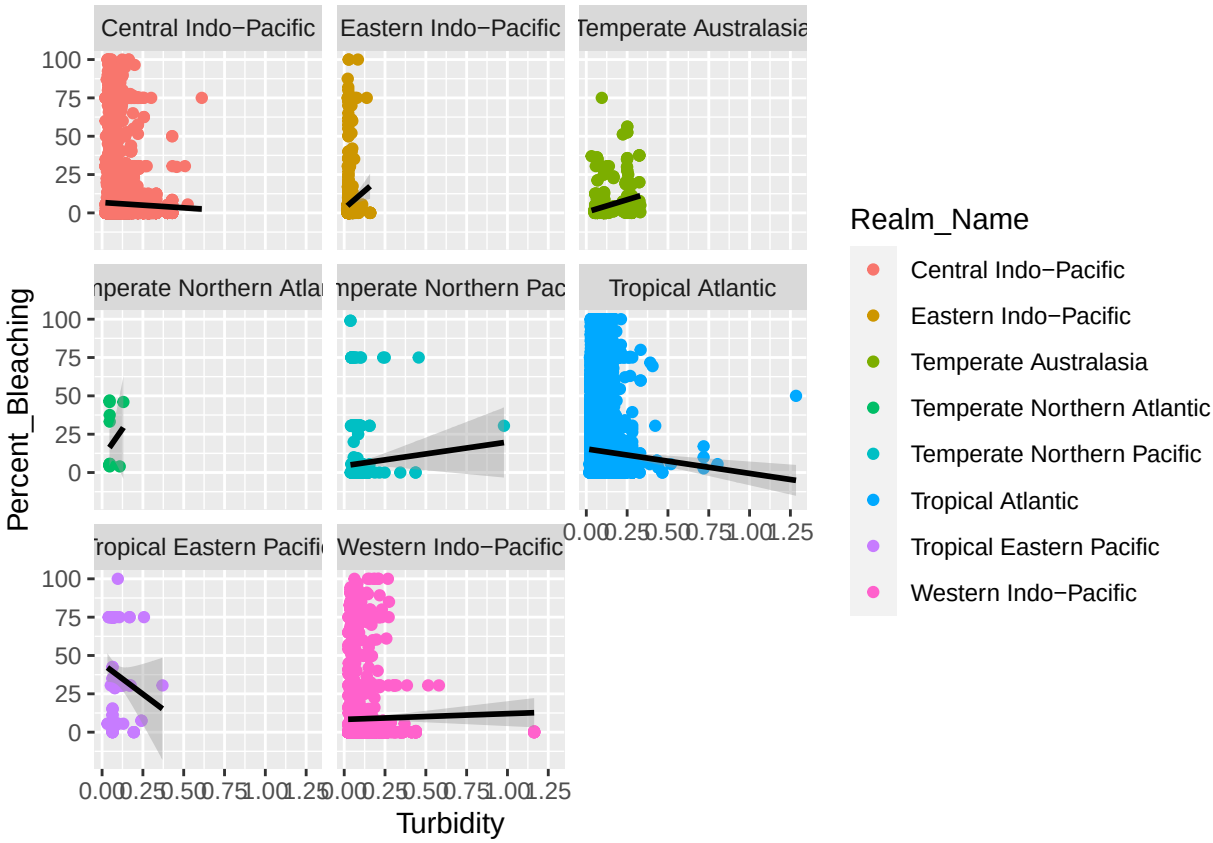
```
## Warning: Removed 16 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 16 rows containing missing values (geom_point).
```



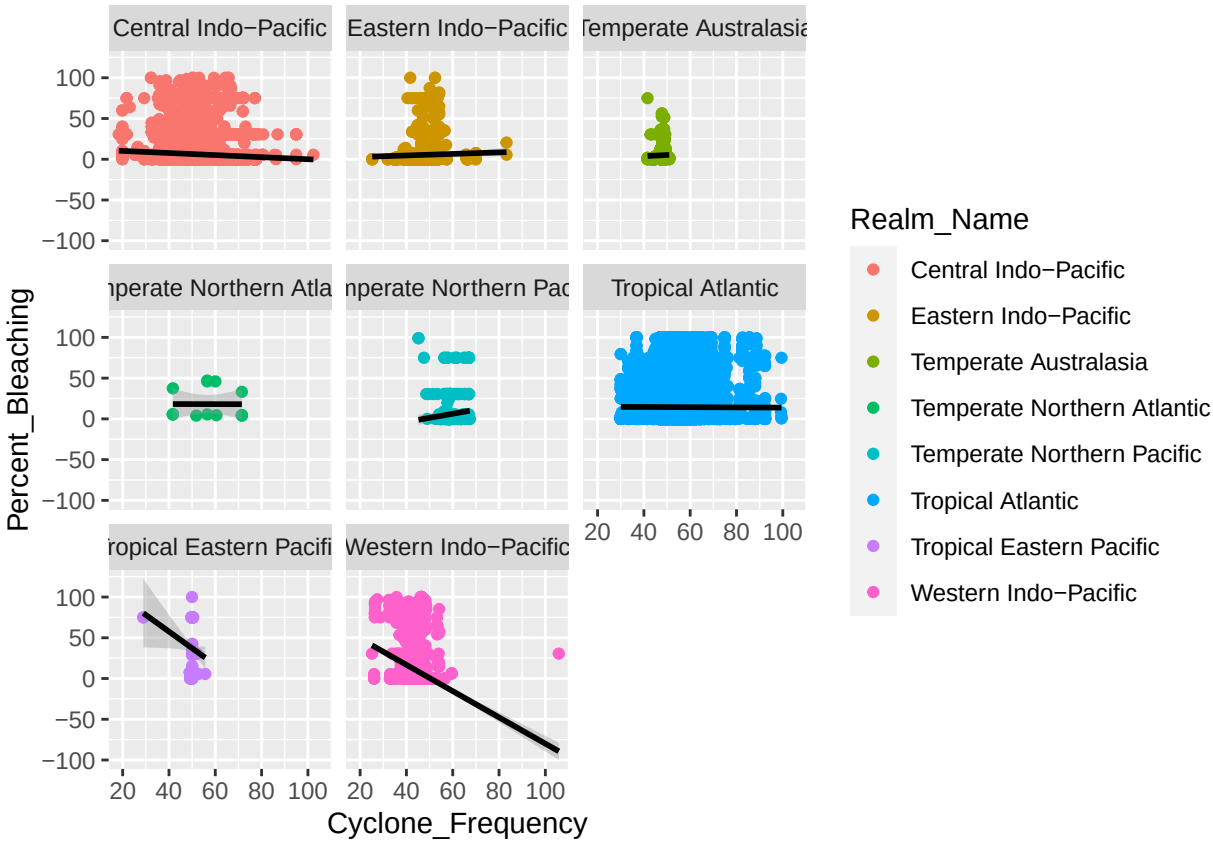

```
#Turbidity | Realm facet wrapped (2)
update_bleach %>%
  ggplot(aes(Turbidity,Percent_Bleaching,color=Realm_Name)) +
  geom_point() +
  geom_smooth(method="lm",color="black") +
  scale_x_continuous(n.break=6) +
  facet_wrap(~Realm_Name)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



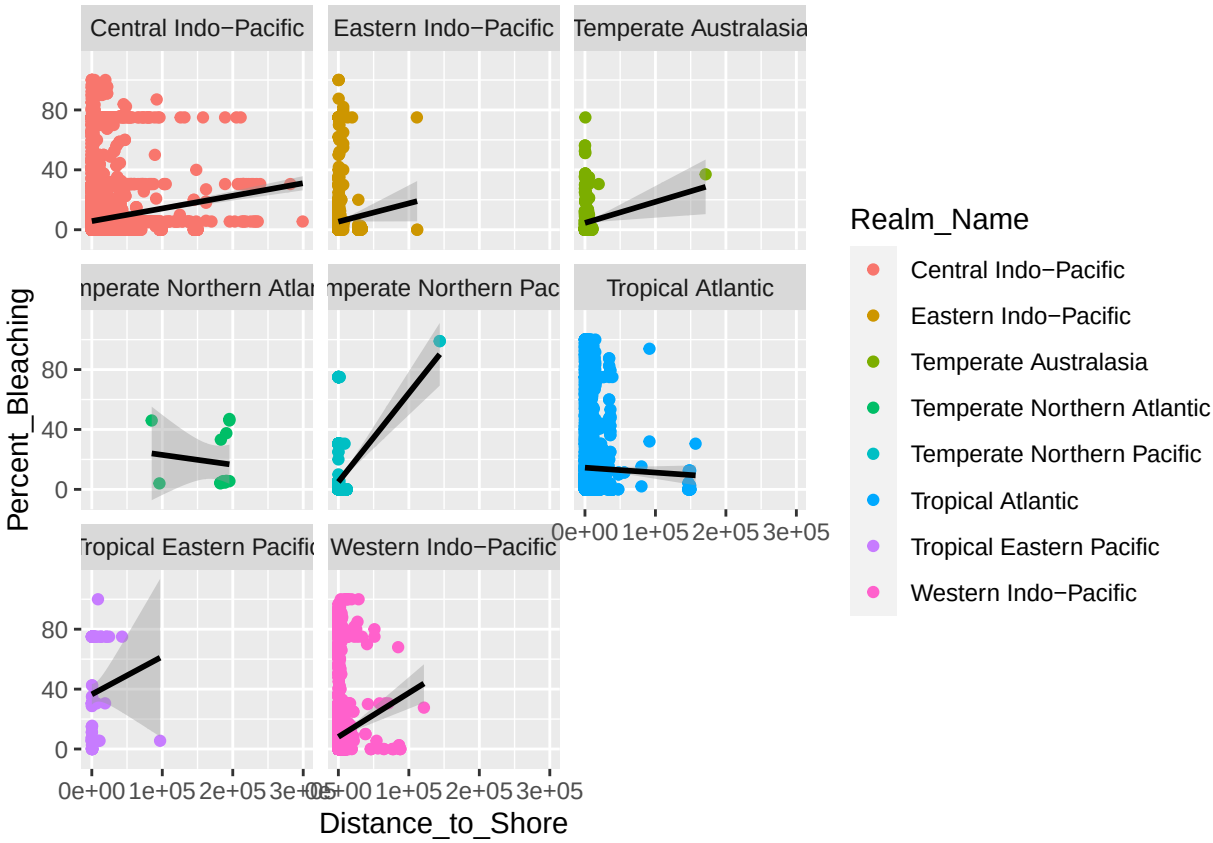

```
#Cyclone Frequency | Realm Facet Wrapped (3)
update_bleach %>%
  ggplot(aes(Cyclone_Frequency,Percent_Bleaching,color=Realm_Name)) +
  geom_point() +
  geom_smooth(method="lm",color="black") +
  scale_x_continuous(n.break=6) +
  facet_wrap(~Realm_Name)
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
#Distance to Shore | Realm Facet Wrapped (4)
update_bleach %>%
  ggplot(aes(Distance_to_Shore,Percent_Bleaching,color=Realm_Name)) +
  geom_point() +
  geom_smooth(method="lm",color="black") +
  scale_x_continuous(n.break=3) +
  facet_wrap(~Realm_Name)
```

```
## `geom_smooth()` using formula 'y ~ x'
## Warning: Removed 2 rows containing non-finite values (stat_smooth).
## Warning: Removed 2 rows containing missing values (geom_point).
```




```
# While reading from csv file, replacing all "nd" with NA.
# Once, done all numeric columns like Temperature_Kelvin, etc. automatically
# got converted to dbl.

df <- read_csv("V2-global-bleaching-and-environmental-data.csv", na = "nd")
```

```
## Rows: 41361 Columns: 62
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (15): Data_Source, Ocean_Name, Reef_ID, Realm_Name, Ecoregion_Name, Cou...
## dbl  (46): Site_ID, Sample_ID, Latitude_Degrees, Longitude_Degrees, Distance...
## date  (1): Date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(df)
```

```
## # A tibble: 6 x 62
##   Site_ID Sample_ID Data_Source Latitu~1 Longi~2 Ocean~3 Reef_ID Realm~4 Ecore~5
##     <dbl>     <dbl> <chr>          <dbl>   <dbl> <chr>   <chr>   <chr>   <chr>
## 1    2501  10324336 Donner         23.2    -82.5 Atlant~ <NA>    Tropic~ Cuba a~
## 2    3467  10324754 Donner        -17.6   -150.  Pacific <NA>    Easter~ Societ~
## 3    1794  10323866 Donner         18.4    -64.6 Atlant~ <NA>    Tropic~ Hispan~
## 4    8647  10328028 Donner         17.8    -64.6 Atlant~ <NA>    Tropic~ Hispan~
## 5    8648  10328029 Donner         17.8    -64.6 Atlant~ <NA>    Tropic~ Hispan~
## 6    2180  10324021 Donner          9.82   -75.9 Atlant~ <NA>    Tropic~ Nether~
```

```
## # ... with 53 more variables: Country_Name <chr>,
## #   State_Island_Province_Name <chr>, City_Town_Name <chr>, Site_Name <chr>,
## #   Distance_to_Shore <dbl>, Exposure <chr>, Turbidity <dbl>,
## #   Cyclone_Frequency <dbl>, Date_Day <dbl>, Date_Month <dbl>, Date_Year <dbl>,
## #   Depth_m <dbl>, Substrate_Name <chr>, Percent_Cover <dbl>,
## #   Bleaching_Level <chr>, Percent_Bleaching <dbl>, ClimSST <dbl>,
## #   Temperature_Kelvin <dbl>, Temperature_Mean <dbl>, ...
```

```
colnames(df)
```

```
##  [1] "Site_ID"
##  [2] "Sample_ID"
##  [3] "Data_Source"
##  [4] "Latitude_Degrees"
##  [5] "Longitude_Degrees"
##  [6] "Ocean_Name"
##  [7] "Reef_ID"
##  [8] "Realm_Name"
##  [9] "Ecoregion_Name"
## [10] "Country_Name"
## [11] "State_Island_Province_Name"
## [12] "City_Town_Name"
## [13] "Site_Name"
## [14] "Distance_to_Shore"
## [15] "Exposure"
## [16] "Turbidity"
## [17] "Cyclone_Frequency"
## [18] "Date_Day"
## [19] "Date_Month"
## [20] "Date_Year"
## [21] "Depth_m"
## [22] "Substrate_Name"
## [23] "Percent_Cover"
## [24] "Bleaching_Level"
## [25] "Percent_Bleaching"
## [26] "ClimSST"
## [27] "Temperature_Kelvin"
## [28] "Temperature_Mean"
## [29] "Temperature_Minimum"
## [30] "Temperature_Maximum"
## [31] "Temperature_Kelvin_Standard_Deviation"
## [32] "Windspeed"
## [33] "SSTA"
## [34] "SSTA_Standard_Deviation"
## [35] "SSTA_Mean"
## [36] "SSTA_Minimum"
## [37] "SSTA_Maximum"
## [38] "SSTA_Frequency"
## [39] "SSTA_Frequency_Standard_Deviation"
## [40] "SSTA_FrequencyMax"
## [41] "SSTA_FrequencyMean"
## [42] "SSTA_DHW"
## [43] "SSTA_DHW_Standard_Deviation"
## [44] "SSTA_DHWMax"
## [45] "SSTA_DHWMean"
```

```
## [46] "TSA"
## [47] "TSA_Standard_Deviation"
## [48] "TSA_Minimum"
## [49] "TSA_Maximum"
## [50] "TSA_Mean"
## [51] "TSA_Frequency"
## [52] "TSA_Frequency_Standard_Deviation"
## [53] "TSA_FrequencyMax"
## [54] "TSA_FrequencyMean"
## [55] "TSA_DHW"
## [56] "TSA_DHW_Standard_Deviation"
## [57] "TSA_DHWMax"
## [58] "TSA_DHWMean"
## [59] "Date"
## [60] "Site_Comments"
## [61] "Sample_Comments"
## [62] "Bleaching_Comments"
```

```
# For our processing from case to case basis, we have to omit all rows with NA
df_month_temp <- df  %>%
  select(Date_Month, Temperature_Kelvin) %>%
  na.omit()

head(df_month_temp)
```

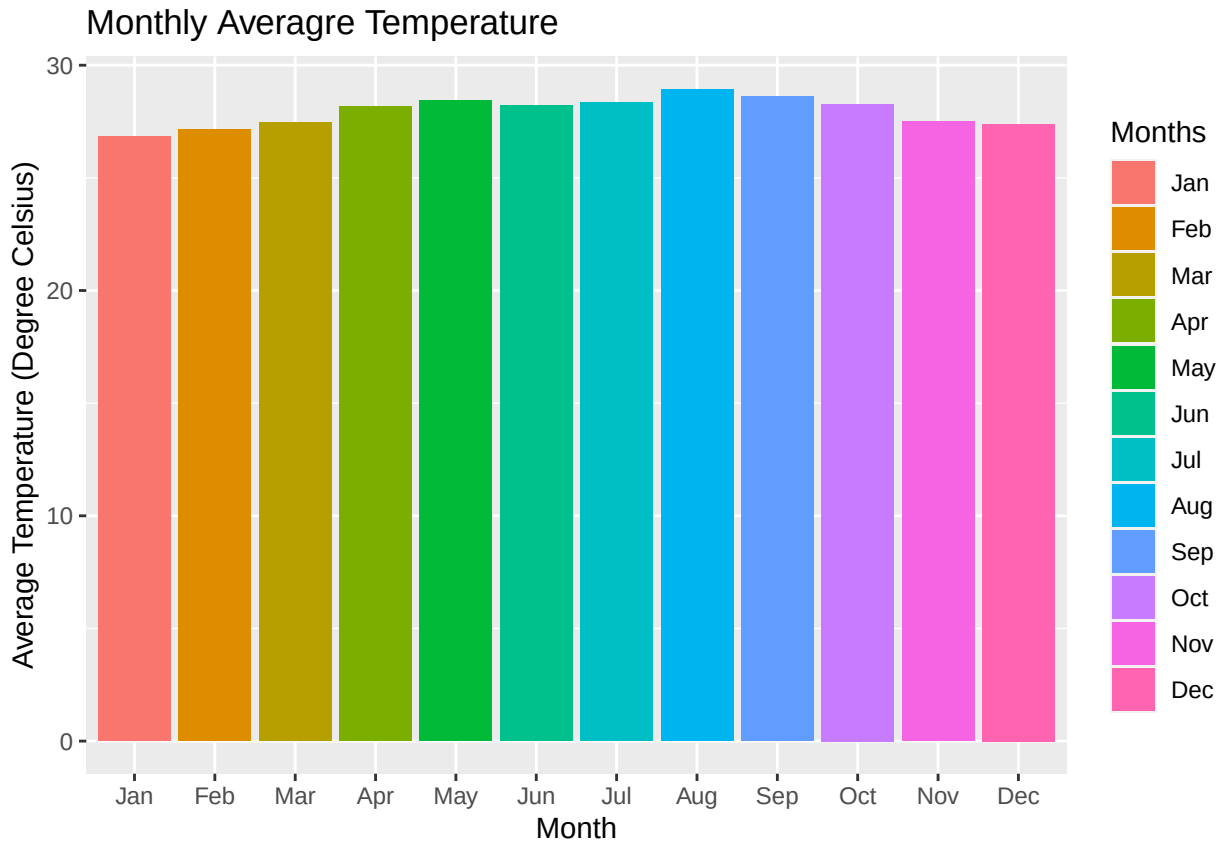
```
## # A tibble: 6 x 2
##   Date_Month Temperature_Kelvin
##        <dbl>              <dbl>
## 1          9               302.
## 2          3               303.
## 3          1               299.
## 4          4               300.
## 5          4               300.
## 6          8               303.
```

```
# In df_month_and_avg_temp dataset, adding another column Months(i.e. Jan, Feb, ...)
df_month_and_avg_temp <- df_month_temp %>%
  group_by(Date_Month) %>%
  summarise(Average_Temperature_Centrigrate = mean(Temperature_Kelvin, na.rm=T) - 273.15) %>%
  mutate(Months = factor(month.abb[Date_Month], levels = month.abb))

print(df_month_and_avg_temp)
```

```
## # A tibble: 12 x 3
##    Date_Month Average_Temperature_Centrigrate Months
##         <dbl>                           <dbl> <fct>
##  1          1                            26.8 Jan
##  2          2                            27.1 Feb
##  3          3                            27.5 Mar
##  4          4                            28.2 Apr
##  5          5                            28.4 May
##  6          6                            28.2 Jun
##  7          7                            28.4 Jul
##  8          8                            28.9 Aug
##  9          9                            28.6 Sep
```

```
## 10          10                          28.3 Oct
## 11          11                          27.5 Nov
## 12          12                          27.4 Dec
```

```
# Bar Plot (geom_bar) of Months Vs. Average Temperature
ggplot(df_month_and_avg_temp, aes(x = Months, y = Average_Temperature_Centrigrate, fill = Months)) +
  geom_bar(na.rm=T, stat = "identity", width = 0.90) +
  labs(x = 'Month', y = 'Average Temperature (Degree Celsius)', title = 'Monthly Averagre Temperature')
```




```
# To know all Realm_Name
unique(df$Realm_Name)
```

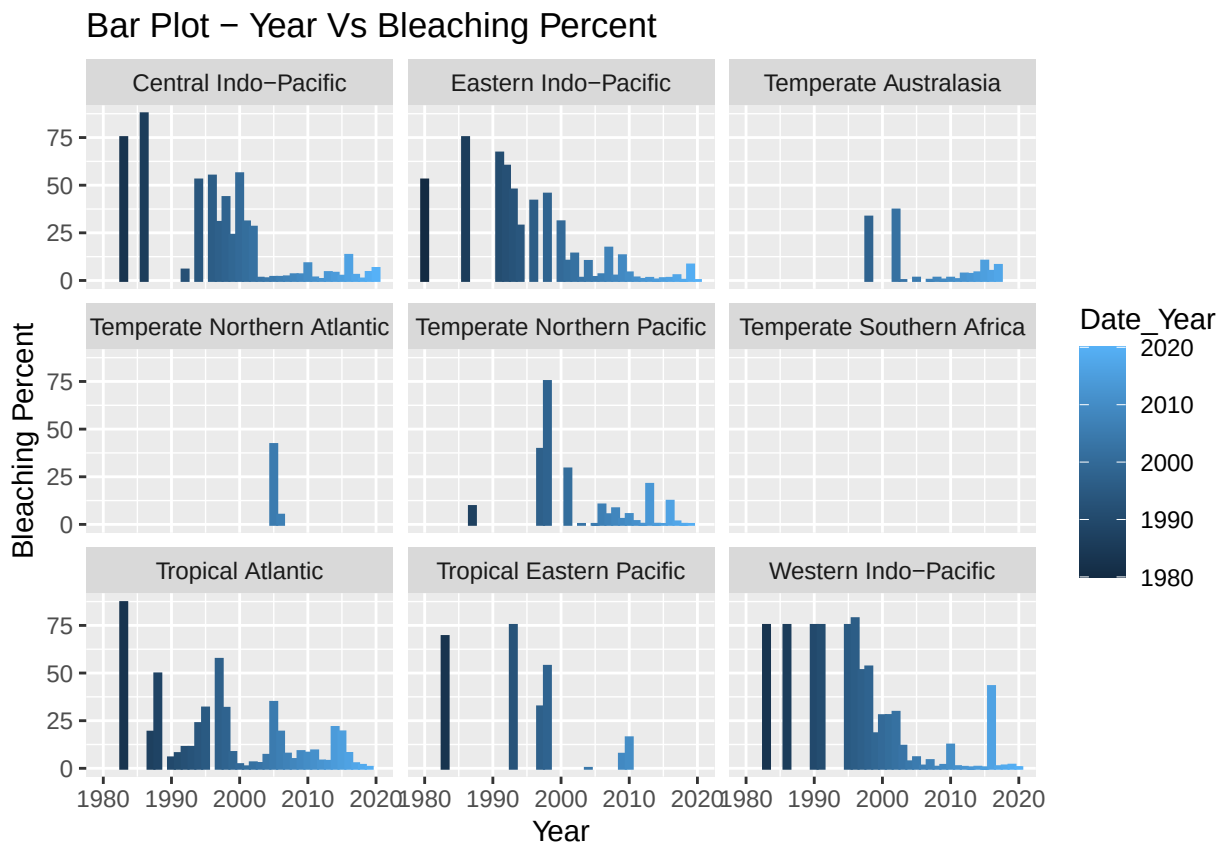
```
## [1] "Tropical Atlantic"           "Eastern Indo-Pacific"
## [3] "Western Indo-Pacific"        "Central Indo-Pacific"
## [5] "Temperate Northern Pacific"  "Tropical Eastern Pacific"
## [7] "Temperate Australasia"       "Temperate Northern Atlantic"
## [9] "Temperate Southern Africa"
```

```
# Selecting Realm_Name, Date_Year and Percent_Bleaching.
# Then taking the average (ignoring NAs) of Percent_Bleaching for a combination of Realm_Name and Date_
df_year_percent_bleach_realm <- df %>%
  select(Realm_Name, Date_Year, Percent_Bleaching) %>%
  group_by(Realm_Name, Date_Year) %>%
  summarize(Average_Percent_Bleaching = mean(Percent_Bleaching, na.rm=T))
```

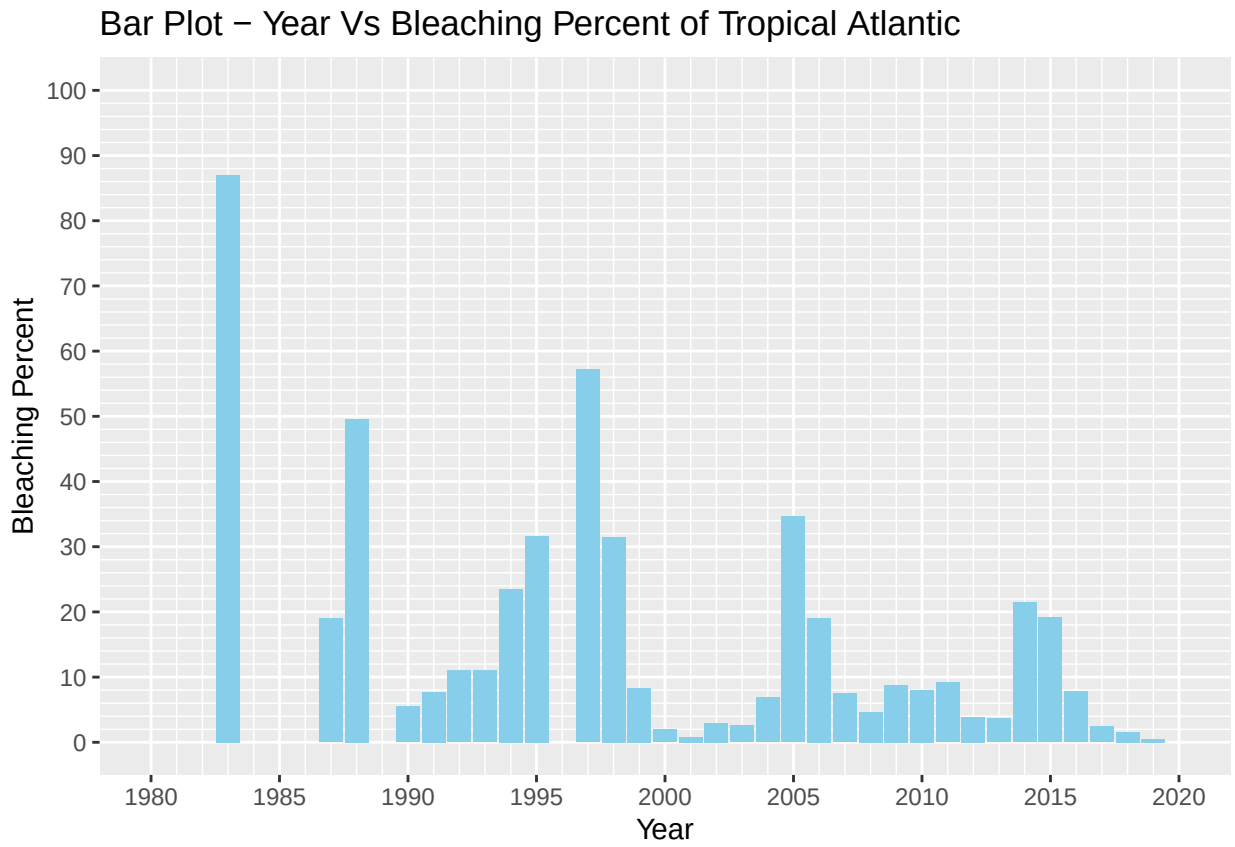
```
## `summarise()` has grouped output by 'Realm_Name'. You can override using the
## `.groups` argument.
```

```
# Bar Plot (geom_bar) of Percent Bleaching Vs Year for each Realm_Name
ggplot(data=df_year_percent_bleach_realm, aes(x=Date_Year, y=Average_Percent_Bleaching, fill = Date_Yea
  geom_bar(na.rm=T, stat="identity") +
  facet_wrap(~Realm_Name) +
  labs(x = 'Year', y = 'Bleaching Percent', title = 'Bar Plot - Year Vs Bleaching Percent')
```

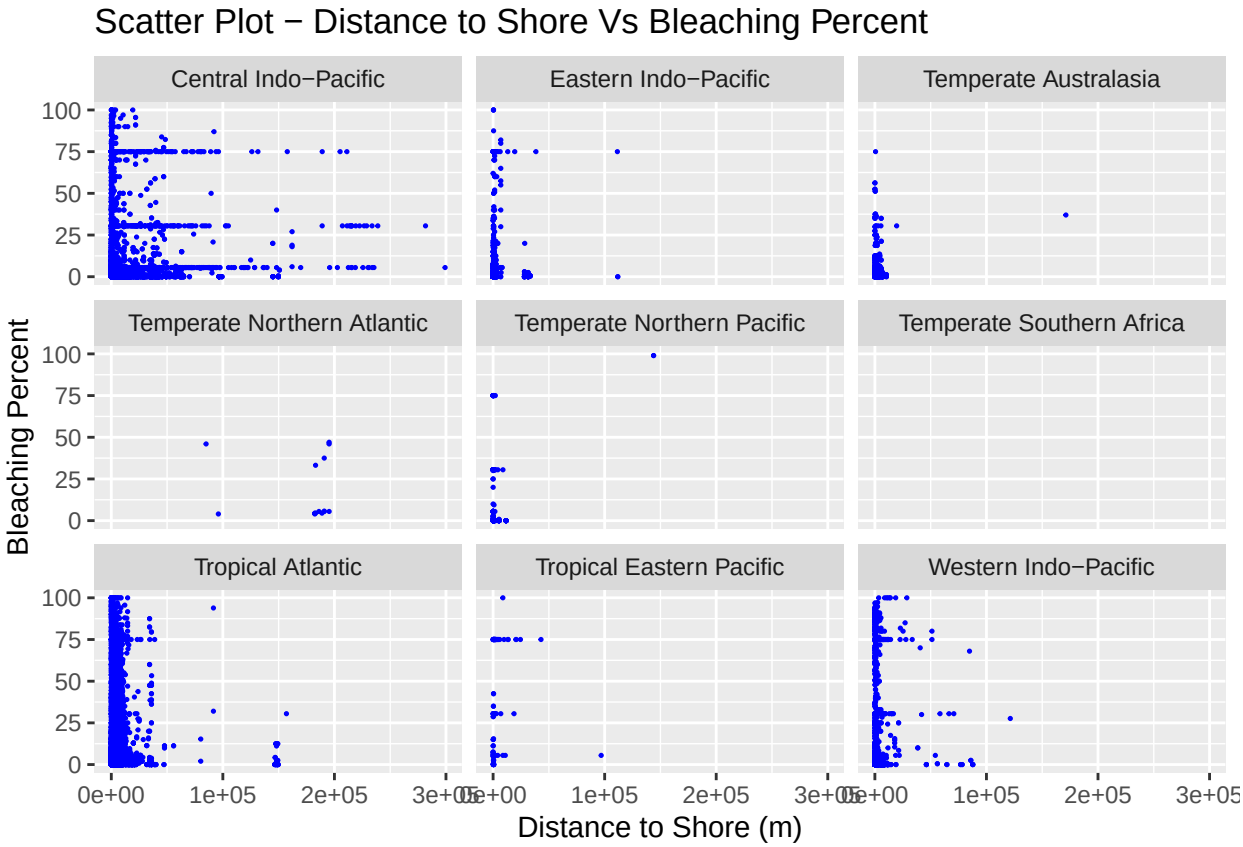
```
## Warning: Removed 19 rows containing missing values (position_stack).
```



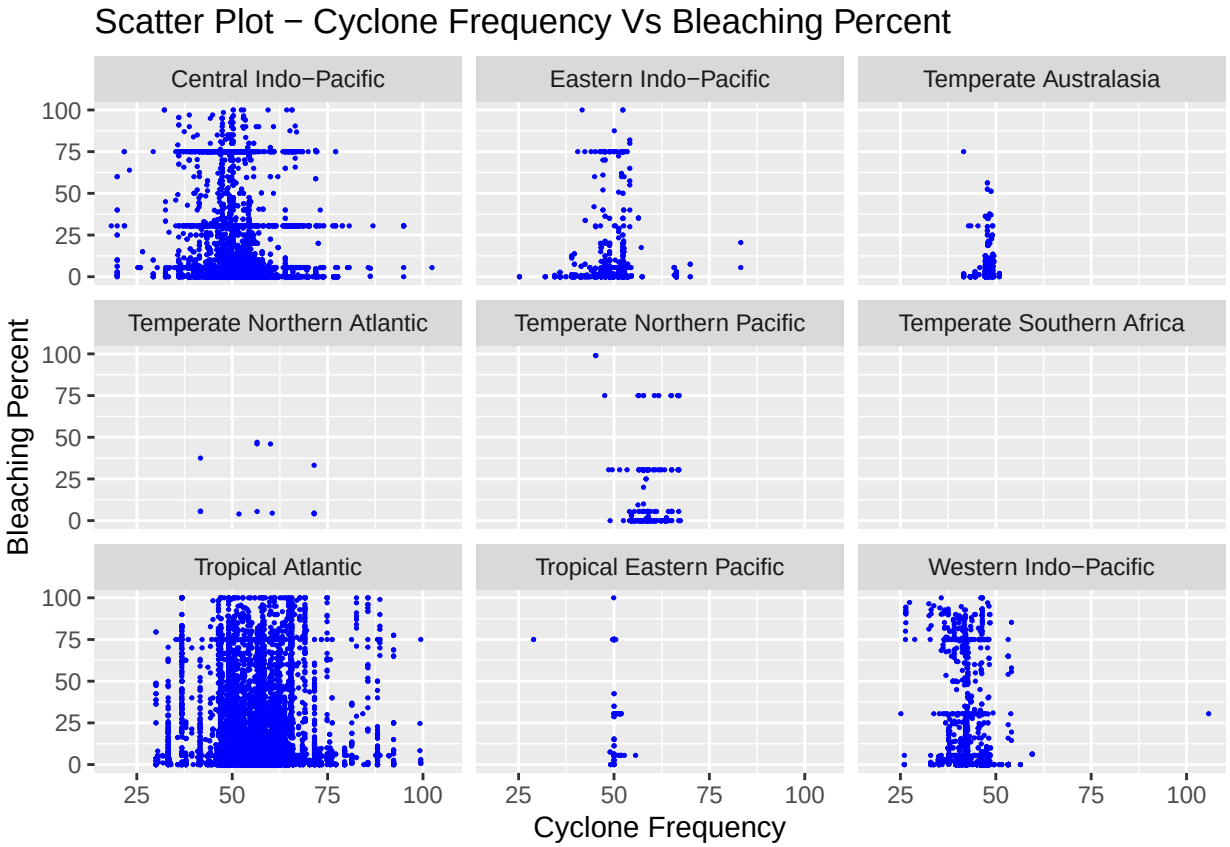

```
df_Tropical_Atlantic <- df_year_percent_bleach_realm %>%
  filter(Realm_Name == 'Tropical Atlantic')

# Bar Plot (geom_bar) of Percent Bleaching Vs Year for Tropical Atlantic
ggplot(data=df_Tropical_Atlantic, aes(x=Date_Year, y=Average_Percent_Bleaching)) +
  geom_bar(na.rm=T, stat="identity", fill = 'skyblue') +
  scale_x_continuous(limits = c(1980,2020), breaks = seq(1980,2020,5), minor_breaks = seq(1980, 2020,1)
  scale_y_continuous(limits = c(0,100), breaks = seq(0,100,10), minor_breaks = seq(0,100,2)) +
  labs(x = 'Year', y = 'Bleaching Percent', title = 'Bar Plot - Year Vs Bleaching Percent of Tropical A
```



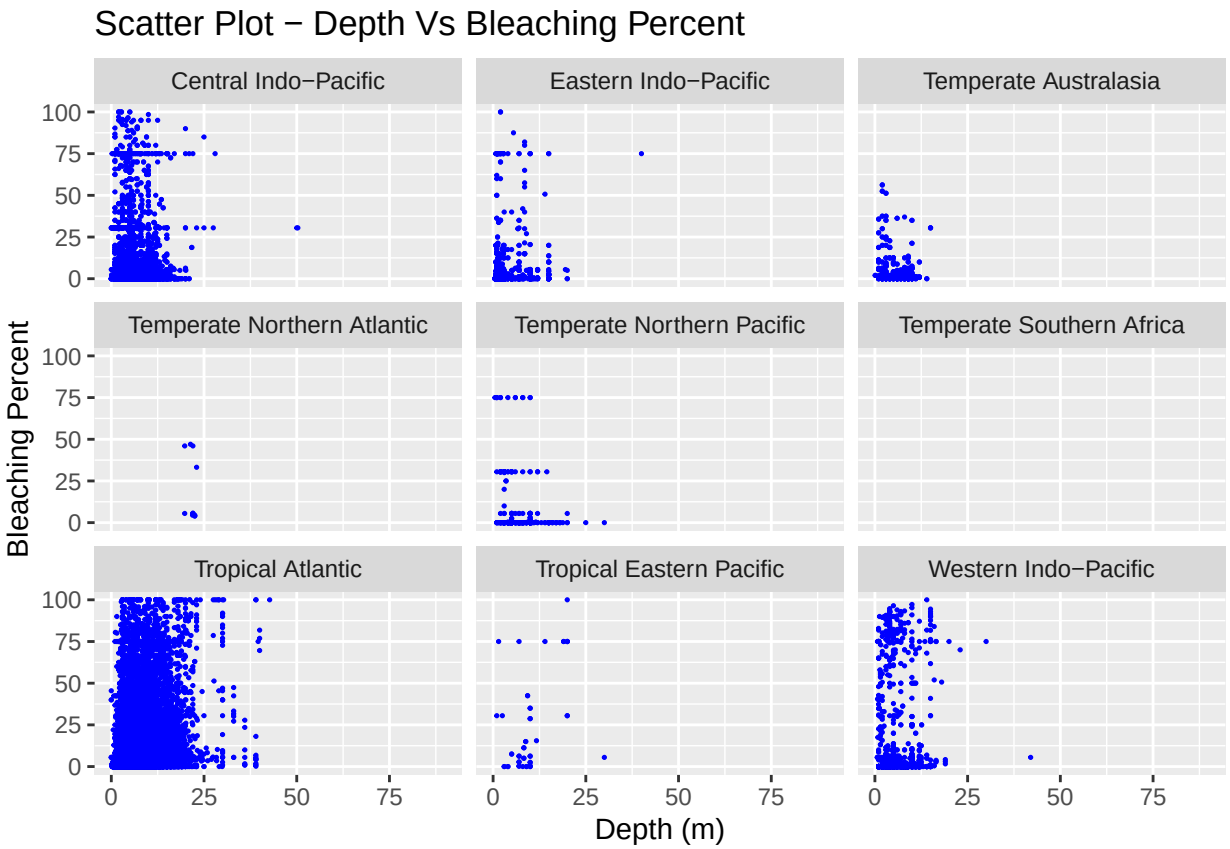

```
# Scatter Plot of Distance_to_Shore Vs Percent_Bleaching for all Realm_Name
ggplot(df, aes(x=Distance_to_Shore, y=Percent_Bleaching)) +
  geom_point(na.rm=T, color = 'blue', size = 0.25) +
  facet_wrap(~Realm_Name) +
  labs(x = 'Distance to Shore (m)', y = 'Bleaching Percent', title = 'Scatter Plot - Distance to Shore
```



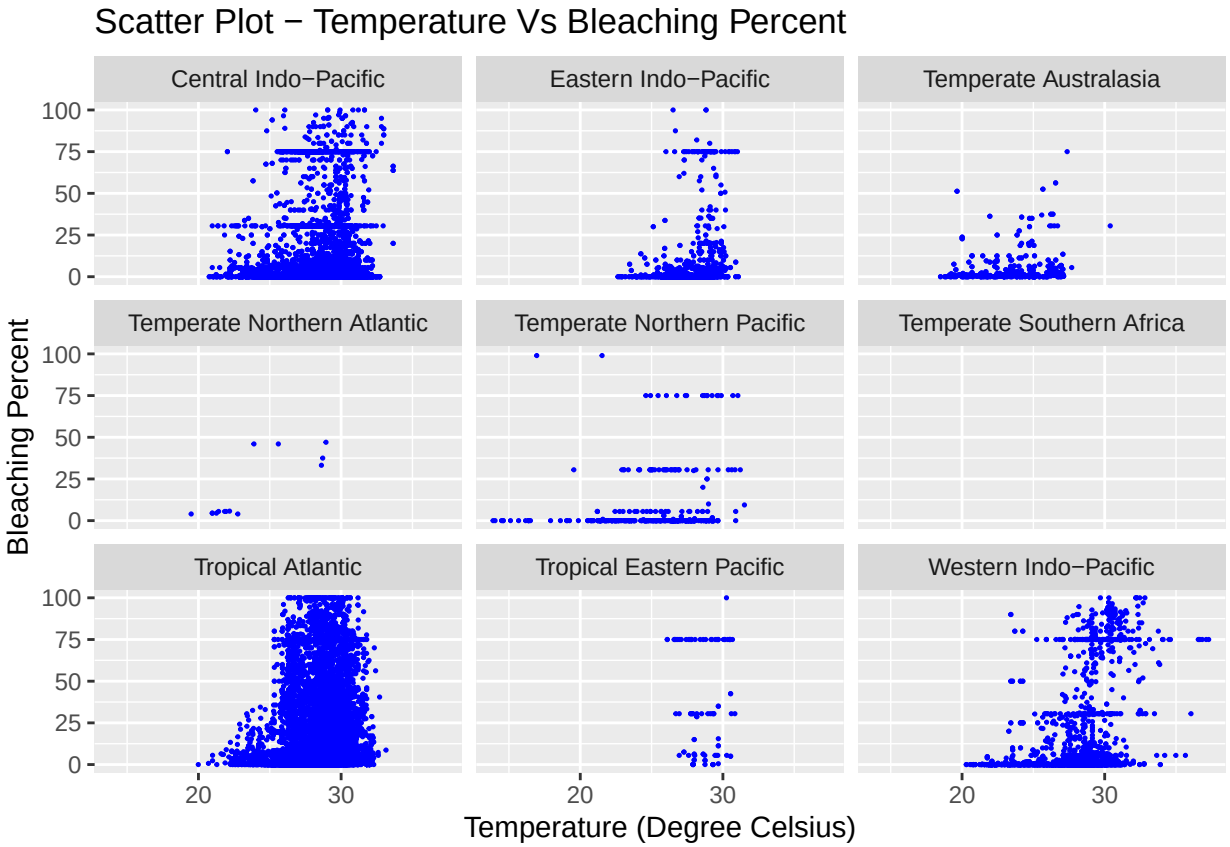

```
# Scatter Plot of Cyclone_Frequency Vs Percent_Bleaching for all Realm_Name
ggplot(df, aes(x=Cyclone_Frequency, y=Percent_Bleaching)) +
  geom_point(na.rm=T, color = 'blue', size = 0.25) +
  facet_wrap(~Realm_Name) +
  labs(x = 'Cyclone Frequency', y = 'Bleaching Percent', title = 'Scatter Plot - Cyclone Frequency Vs B
```



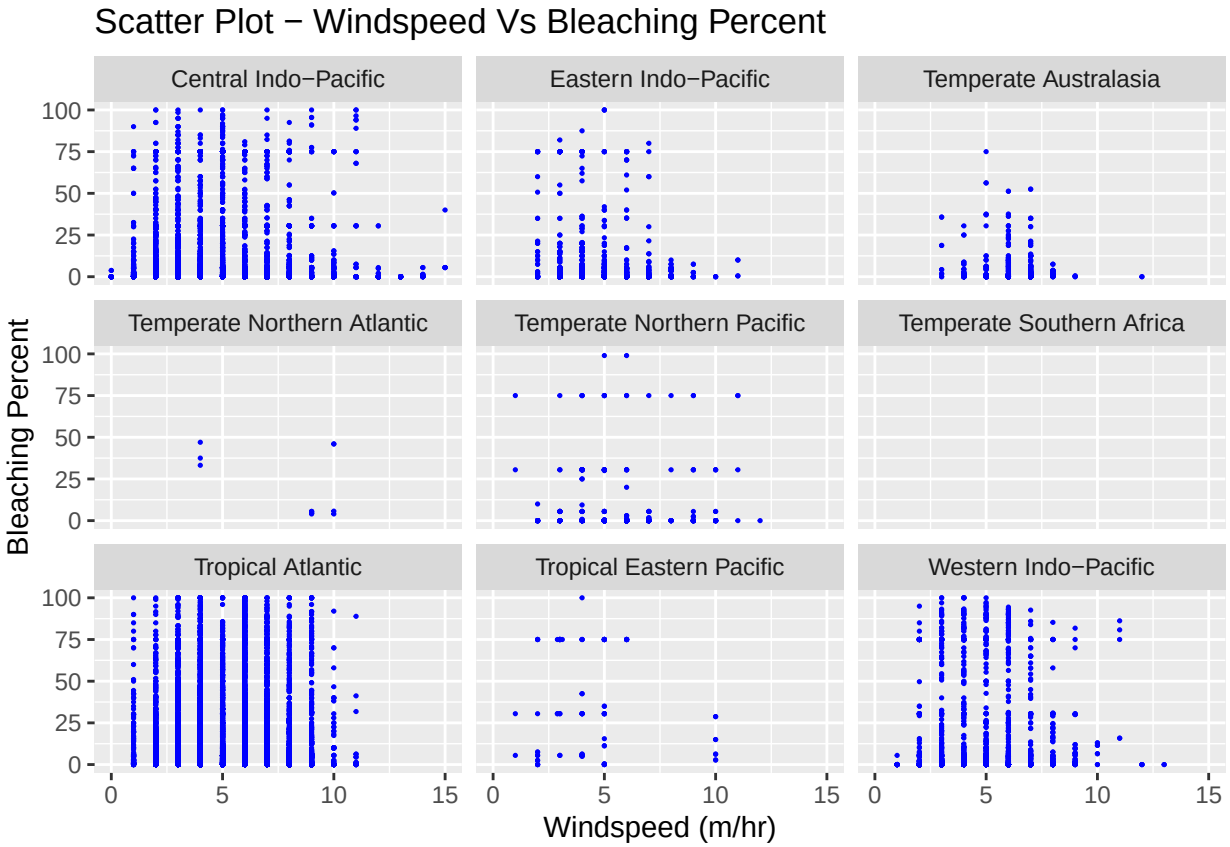

```
# Scatter Plot of Depth_m Vs Percent_Bleaching for all Realm_Name
ggplot(df, aes(x=Depth_m, y=Percent_Bleaching)) +
  geom_point(na.rm=T, color = 'blue', size = 0.25) +
  facet_wrap(~Realm_Name) +
  labs(x = 'Depth (m)', y = 'Bleaching Percent', title = 'Scatter Plot - Depth Vs Bleaching Percent')
```



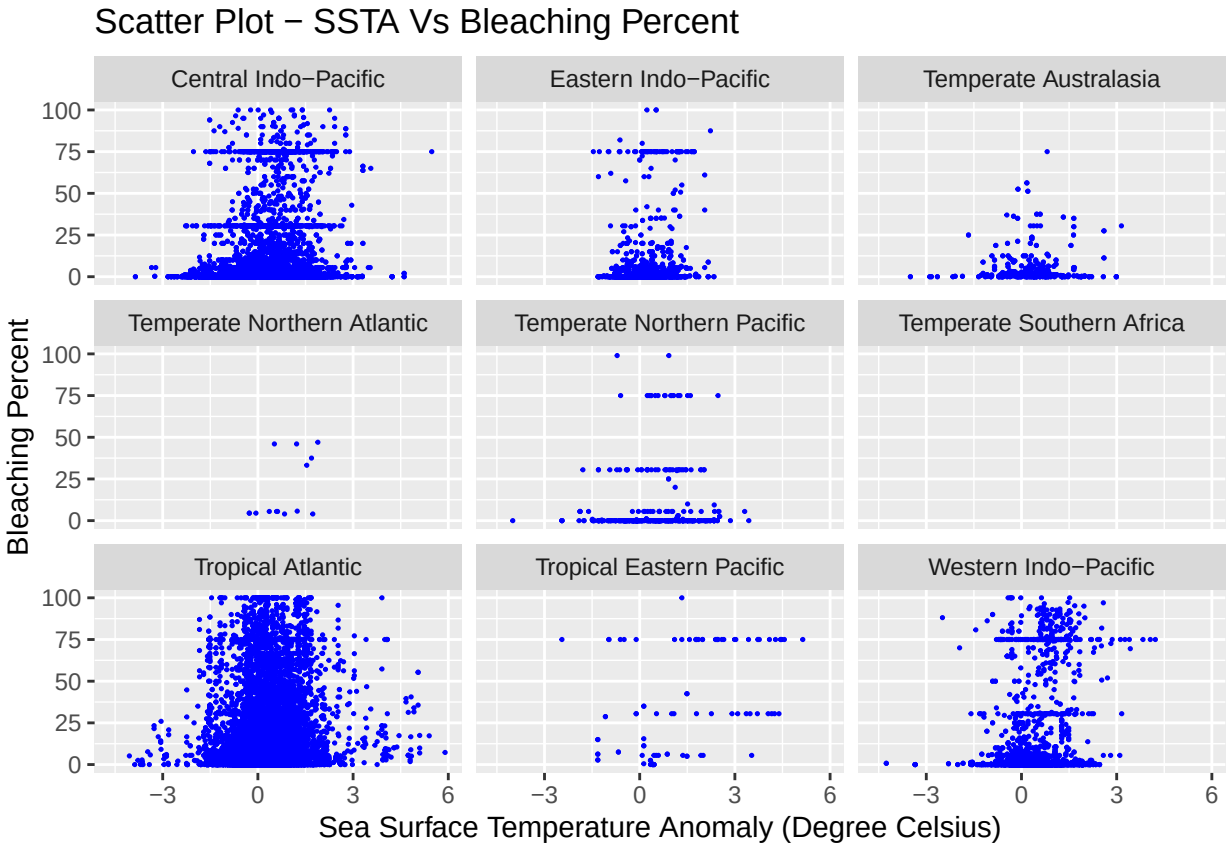

```
# Scatter Plot of Temperature Vs Percent_Bleaching for all Realm_Name
ggplot(df, aes(x=Temperature_Kelvin - 273.15, y=Percent_Bleaching)) +
  geom_point(na.rm=T, color = 'blue', size = 0.25) +
  facet_wrap(~Realm_Name) +
  labs(x = 'Temperature (Degree Celsius)', y = 'Bleaching Percent', title = 'Scatter Plot - Temperature
```



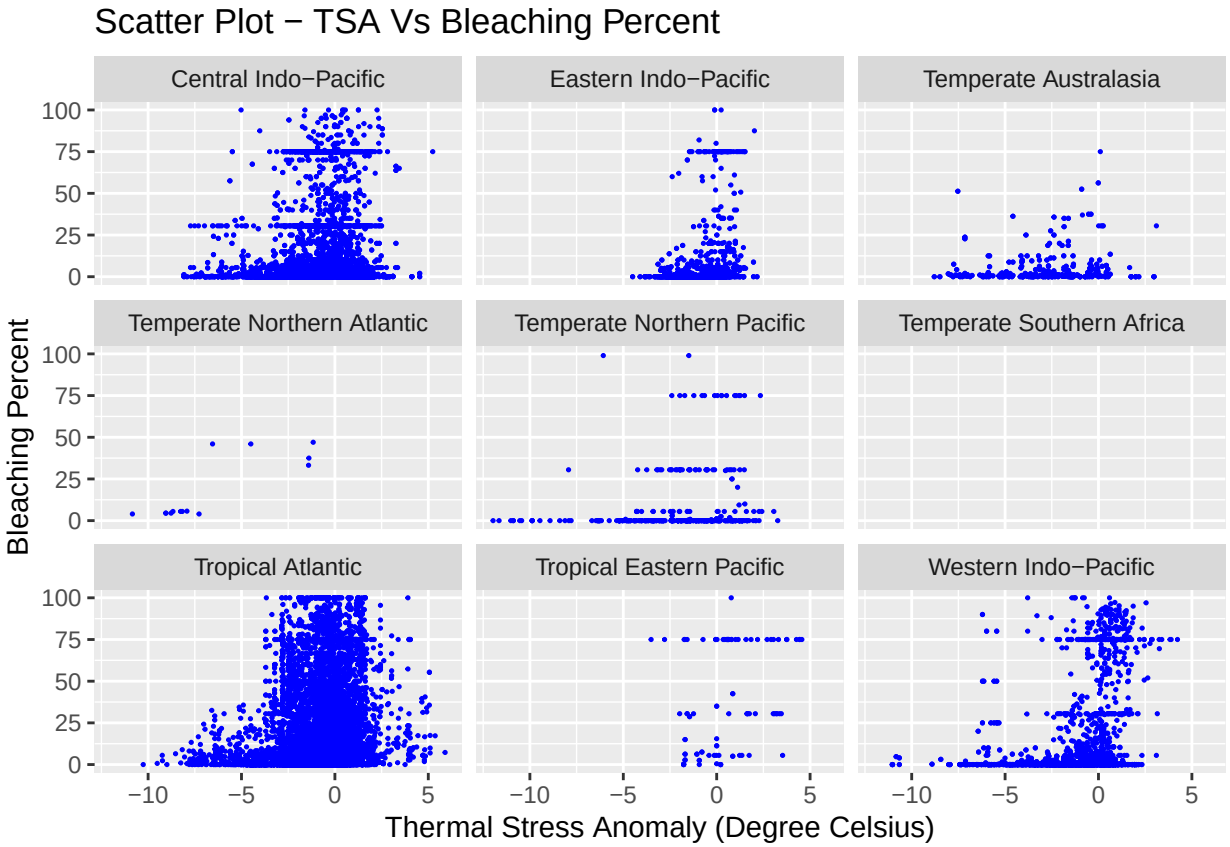

```
# Scatter Plot of Windspeed Vs Percent_Bleaching for all Realm_Name
ggplot(df, aes(x=Windspeed, y=Percent_Bleaching)) +
  geom_point(na.rm=T, color = 'blue', size = 0.25) +
  facet_wrap(~Realm_Name) +
  labs(x = 'Windspeed (m/hr)', y = 'Bleaching Percent', title = 'Scatter Plot - Windspeed Vs Bleaching
```



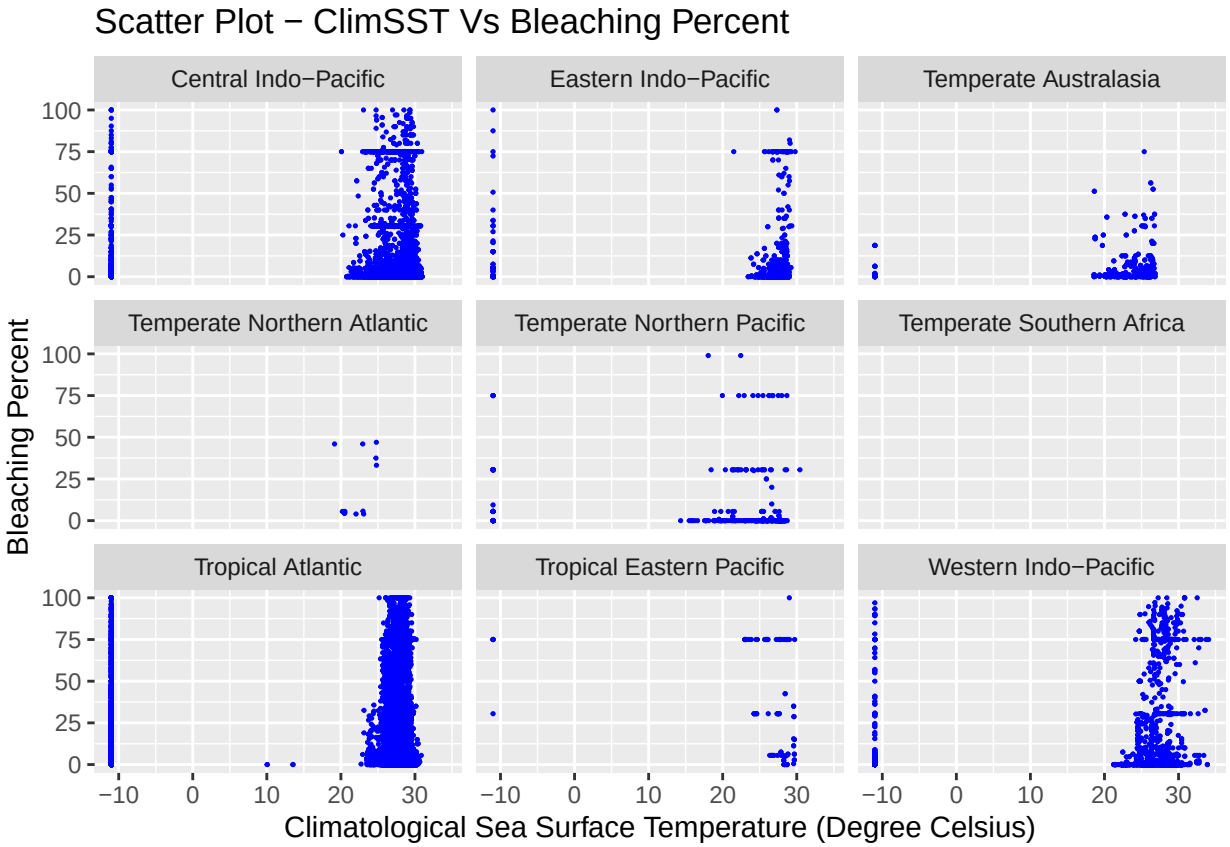

```
# Scatter Plot of SSTA Vs Percent_Bleaching for all Realm_Name
ggplot(df, aes(x=SSTA, y=Percent_Bleaching)) +
  geom_point(na.rm=T, color = 'blue', size = 0.25) +
  facet_wrap(~Realm_Name) +
  labs(x = 'Sea Surface Temperature Anomaly (Degree Celsius)', y = 'Bleaching Percent', title = 'Scatte
```



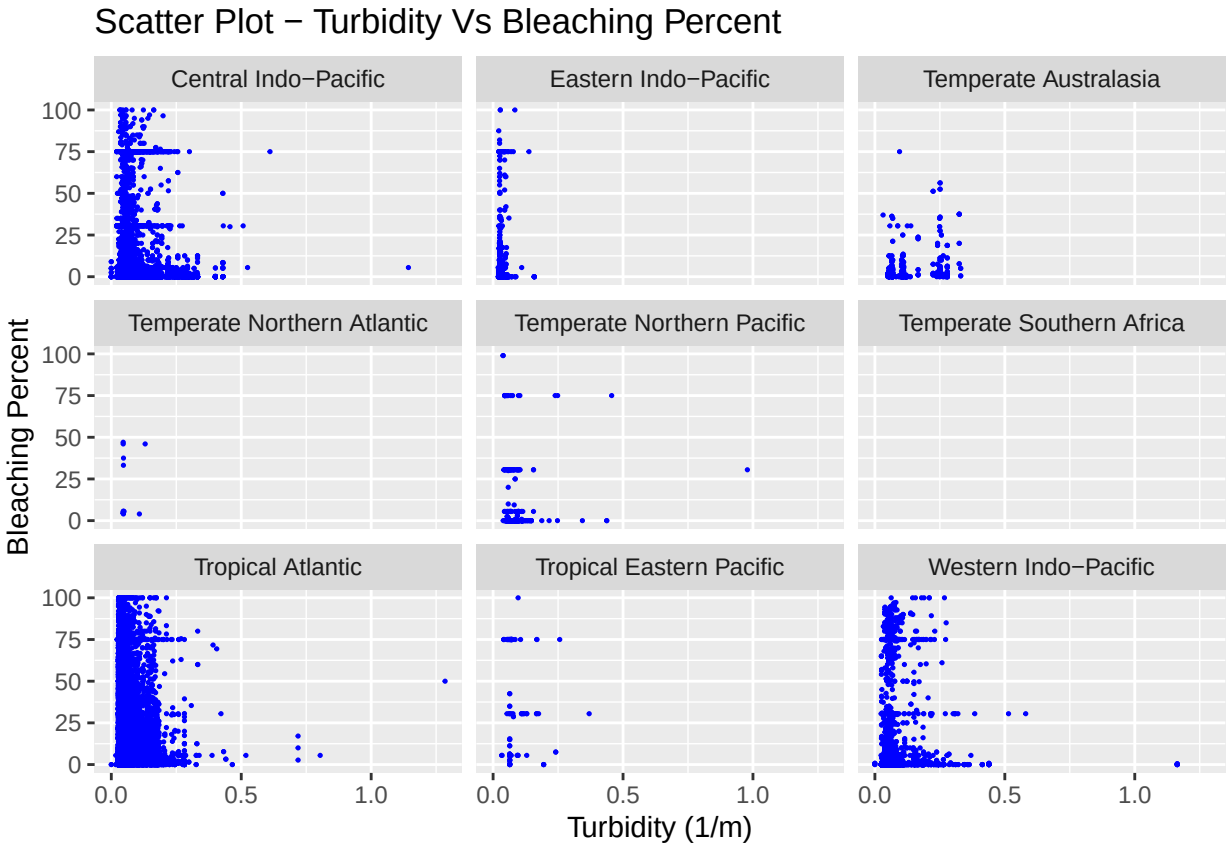

```
# Scatter Plot of TSA Vs Percent_Bleaching for all Realm_Name
ggplot(df, aes(x=TSA, y=Percent_Bleaching)) +
  geom_point(na.rm=T, color = 'blue', size = 0.25) +
  facet_wrap(~Realm_Name) +
  labs(x = 'Thermal Stress Anomaly (Degree Celsius)', y = 'Bleaching Percent', title = 'Scatter Plot - '
```



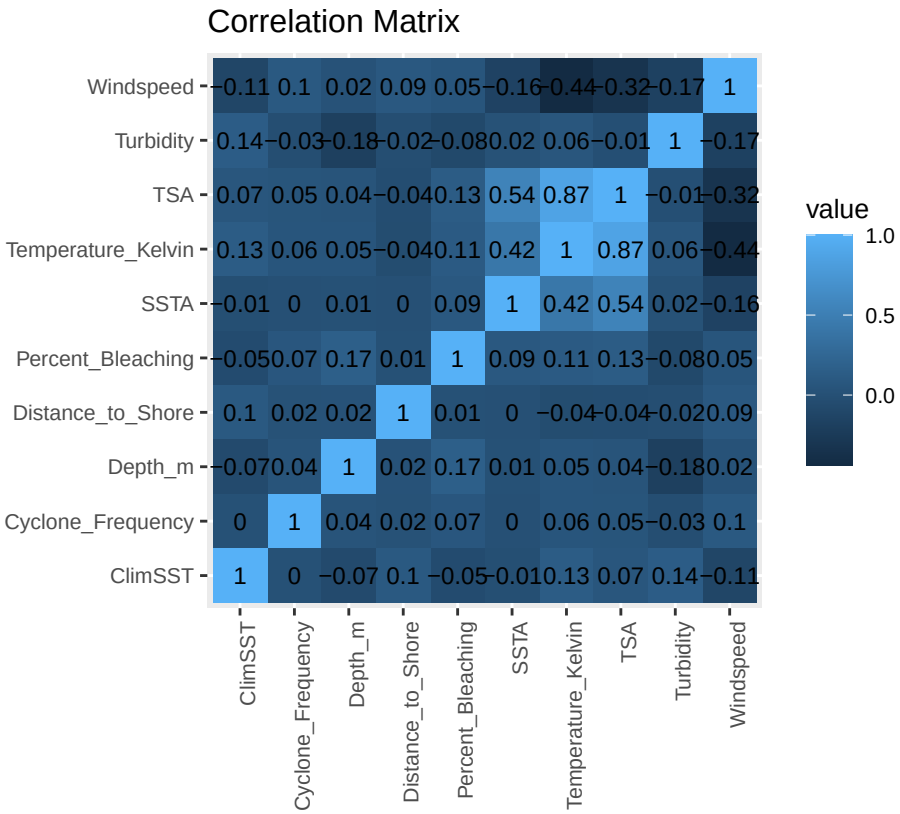

```
# Scatter Plot of ClimSST Vs Percent_Bleaching for all Realm_Name
ggplot(df, aes(x=ClimSST - 273.15, y=Percent_Bleaching)) +
  geom_point(na.rm=T, color = 'blue', size = 0.25) +
  facet_wrap(~Realm_Name) +
  labs(x = 'Climatological Sea Surface Temperature (Degree Celsius)', y = 'Bleaching Percent', title =
```




```
# Scatter Plot of Turbidity Vs Percent_Bleaching for all Realm_Name
ggplot(df, aes(x=Turbidity, y=Percent_Bleaching)) +
  geom_point(na.rm=T, color = 'blue', size = 0.25) +
  facet_wrap(~Realm_Name) +
  labs(x = 'Turbidity (1/m)', y = 'Bleaching Percent', title = 'Scatter Plot - Turbidity Vs Bleaching P
```




```
# Visualizing Correlation Matrix for important columns
df_cor <- df %>% select(Percent_Bleaching, Distance_to_Shore,
                        Cyclone_Frequency, Depth_m, Temperature_Kelvin,
                        Windspeed, SSTA, TSA, ClimSST, Turbidity)

df_cor %>% na.omit() %>% select_if(is.numeric) %>% cor %>%
  as.data.frame %>% rownames_to_column %>% pivot_longer(-1) %>%
  ggplot(aes(rowname, name, fill=value)) + geom_tile() +
  geom_text(aes(label=round(value,2)), size=3.3) +
  theme(text = element_text(size=10), axis.text.x = element_text(angle=90, hjust=1)) +
  labs(x = '', y = '', title = 'Correlation Matrix') +
  coord_fixed()
```


Correlation Matrix
Windspeed
Turbidity
TSA
Temperature_Kelvin
SSTA
Percent_Bleaching
Distance_to_Shore
Depth_m
Cyclone_Frequency
ClimSST
ClimSST
Cyclone_Frequency
Depth_m
Distance_to_Shore
Percent_Bleaching
SSTA
Temperature_Kelvin
TSA
Turbidity
Windspeed
value
1.0
0.5
0.0