06_site_specific_analysis.R

laurenkay

2025-08-05

```
# Site-Specific Analysis: Detailed Coral Response Characterization
# Comprehensive analysis of individual site responses and recovery patterns

library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

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

```
library(readr)
library(tidyr)
library(knitr)

# Load all processed data
extent_means <- read_csv("01_extent_site_means.csv")
```

```
## Rows: 165 Columns: 8

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): site, period
## dbl (6): year, replicate, ext_bleached, ext_verypale, ext_anybleaching, ext_...
##
## 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.
```

```
temp_metrics <- read_csv("02_temperature_metrics_2023_2024.csv")
```

```
## Rows: 66 Columns: 16
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): site
## dbl (15): year, max_dhw, max_weekly_temp, mean_weekly_temp, temp_range, temp...
##
## 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.
```

```
predictive_data <- read_csv("04_predictive_dataset.csv")
```

```
## Rows: 32 Columns: 19
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): site
## dbl (18): baseline_2024_annual, outcome_2025_pbl, predictor_2023_annual, per...
##
## 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.
```

```r
site_patterns <- read_csv("03_site_response_patterns.csv")
```

```
## Rows: 33 Columns: 16
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): site, period_p1, period_p2, response_pattern
## dbl (12): initial_bleaching_p1, final_bleaching_p1, recovery_rate_p1, recove...
##
## 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.
```

```r
cat("=== COMPREHENSIVE SITE-SPECIFIC CORAL BLEACHING ANALYSIS ===\n")
```

```
## === COMPREHENSIVE SITE-SPECIFIC CORAL BLEACHING ANALYSIS ===
```

```r
cat("Analysis Period: 2023-2025 Bleaching Events\n")
```

```
## Analysis Period: 2023-2025 Bleaching Events
```

```r
cat("Focus: 2024 Annual → 2025 PBL Response Prediction\n\n")
```

```
## Focus: 2024 Annual → 2025 PBL Response Prediction
```

```r
# Create comprehensive site dataset
site_comprehensive <- predictive_data %>%
  left_join(
    site_patterns %>% select(site, response_pattern),
    by = "site"
  ) %>%
  left_join(
    temp_metrics %>% filter(year == 2024) %>%
    select(site, weeks_above_30, total_dhw_accumulation, cv_temperature),
    by = "site"
  ) %>%
  mutate(
    # Response classifications
    recovery_category = case_when(
      recovery_achieved > 30 ~ "Exceptional_Recovery",
      recovery_achieved > 15 ~ "Strong_Recovery",
      recovery_achieved > 5 ~ "Moderate_Recovery",
      abs(response_magnitude) <= 5 ~ "Stable",
      response_magnitude > 15 ~ "Strong_Worsening",
      response_magnitude > 5 ~ "Moderate_Worsening",
      TRUE ~ "Minimal_Change"
    ),

    # Thermal stress classification
    thermal_stress_level = case_when(
```

```r
      dhw_2024 > 12 ~ "Extreme_Stress",
      dhw_2024 > 8 ~ "High_Stress",
      dhw_2024 > 4 ~ "Moderate_Stress",
      TRUE ~ "Low_Stress"
    ),

    # Previous impact classification
    previous_impact_level = case_when(
      predictor_2023_annual > 60 ~ "Severe_2023_Impact",
      predictor_2023_annual > 30 ~ "Moderate_2023_Impact",
      predictor_2023_annual > 10 ~ "Low_2023_Impact",
      TRUE ~ "Minimal_2023_Impact"
    ),

    # Combined vulnerability index
    vulnerability_score =
      (predictor_2023_annual / 100) * 0.4 +  # 40% weight to previous impact
      (dhw_2024 / 16) * 0.4 +                # 40% weight to current thermal stress
      (temp_instability / max(temp_instability, na.rm = TRUE)) * 0.2  # 20% weight to instability
  ) %>%
  arrange(desc(recovery_achieved))

# Generate detailed site summaries
cat("=== TOP PERFORMING SITES (HIGHEST RECOVERY) ===\n")
```

```
## === TOP PERFORMING SITES (HIGHEST RECOVERY) ===
```

```r
top_recovery <- site_comprehensive %>%
  filter(recovery_achieved > 10) %>%
  arrange(desc(recovery_achieved)) %>%
  slice_head(n = 10)

for(i in 1:nrow(top_recovery)) {
  site_data <- top_recovery[i, ]
  cat(sprintf("\n%d. %s\n", i, site_data$site))
  cat(sprintf("   Recovery: %.1f%% reduction (%.1f%% → %.1f%%)\n",
              site_data$recovery_achieved, site_data$baseline_2024_annual, site_data$outcome_2025_pbl))
  cat(sprintf("   2023 Bleaching: %.1f%%, 2024 DHW: %.1f\n",
              site_data$predictor_2023_annual, site_data$dhw_2024))
  cat(sprintf("   Thermal Stress: %s, Previous Impact: %s\n",
              site_data$thermal_stress_level, site_data$previous_impact_level))
  cat(sprintf("   Vulnerability Score: %.3f\n", site_data$vulnerability_score))
}
```

```
##
## 1. Great Pond
##    Recovery: 81.4% reduction (81.4% → 0.0%)
##    2023 Bleaching: 14.1%, 2024 DHW: 23.1
##    Thermal Stress: Extreme_Stress, Previous Impact: Low_2023_Impact
##    Vulnerability Score: 0.822
##
## 2. Sprat Hole
##    Recovery: 64.2% reduction (64.2% → 0.0%)
##    2023 Bleaching: 33.3%, 2024 DHW: 21.3
##    Thermal Stress: Extreme_Stress, Previous Impact: Moderate_2023_Impact
```

```
##    Vulnerability Score: 0.840
##
## 3. Magens Bay
##    Recovery: 52.4% reduction (60.6% → 8.2%)
##    2023 Bleaching: 35.8%, 2024 DHW: 22.5
##    Thermal Stress: Extreme_Stress, Previous Impact: Moderate_2023_Impact
##    Vulnerability Score: 0.891
##
## 4. Black Point
##    Recovery: 52.2% reduction (61.3% → 9.2%)
##    2023 Bleaching: 49.3%, 2024 DHW: 21.4
##    Thermal Stress: Extreme_Stress, Previous Impact: Moderate_2023_Impact
##    Vulnerability Score: 0.917
##
## 5. Salt River West
##    Recovery: 47.5% reduction (47.5% → 0.0%)
##    2023 Bleaching: 38.0%, 2024 DHW: 3.3
##    Thermal Stress: Low_Stress, Previous Impact: Moderate_2023_Impact
##    Vulnerability Score: 0.399
##
## 6. Cane Bay
##    Recovery: 44.8% reduction (47.3% → 2.5%)
##    2023 Bleaching: 25.8%, 2024 DHW: 20.5
##    Thermal Stress: Extreme_Stress, Previous Impact: Low_2023_Impact
##    Vulnerability Score: 0.794
##
## 7. Savana
##    Recovery: 42.3% reduction (42.3% → 0.0%)
##    2023 Bleaching: 23.1%, 2024 DHW: 21.4
##    Thermal Stress: Extreme_Stress, Previous Impact: Low_2023_Impact
##    Vulnerability Score: 0.801
##
## 8. Seahorse Cottage Shoal
##    Recovery: 37.9% reduction (40.4% → 2.5%)
##    2023 Bleaching: 55.1%, 2024 DHW: 22.5
##    Thermal Stress: Extreme_Stress, Previous Impact: Moderate_2023_Impact
##    Vulnerability Score: 0.963
##
## 9. Coculus Rock
##    Recovery: 37.7% reduction (65.5% → 27.8%)
##    2023 Bleaching: 80.5%, 2024 DHW: 21.9
##    Thermal Stress: Extreme_Stress, Previous Impact: Severe_2023_Impact
##    Vulnerability Score: 1.059
##
## 10. Eagle Ray
##    Recovery: 33.3% reduction (33.3% → 0.0%)
##    2023 Bleaching: 33.3%, 2024 DHW: 21.9
##    Thermal Stress: Extreme_Stress, Previous Impact: Moderate_2023_Impact
##    Vulnerability Score: 0.861
```

```r
cat("\n=== WORST PERFORMING SITES (HIGHEST WORSENING) ===\n")
```

```
##
## === WORST PERFORMING SITES (HIGHEST WORSENING) ===
```

```r
worst_response <- site_comprehensive %>%
  filter(response_magnitude > 10) %>%
  arrange(desc(response_magnitude)) %>%
  slice_head(n = 8)

for(i in 1:nrow(worst_response)) {
  site_data <- worst_response[i, ]
  cat(sprintf("\n%d. %s\n", i, site_data$site))
  cat(sprintf("   Worsening: +%.1f%% increase (%.1f%% → %.1f%%)\n",
              site_data$response_magnitude, site_data$baseline_2024_annual, site_data$outcome_2025_pbl)
  cat(sprintf("   2023 Bleaching: %.1f%%, 2024 DHW: %.1f\n",
              site_data$predictor_2023_annual, site_data$dhw_2024))
  cat(sprintf("   Thermal Stress: %s, Previous Impact: %s\n",
              site_data$thermal_stress_level, site_data$previous_impact_level))
  cat(sprintf("   Vulnerability Score: %.3f\n", site_data$vulnerability_score))
}
```

```
##
## 1. NA
##    Worsening: +NA% increase (NA% → NA%)
##    2023 Bleaching: NA%, 2024 DHW: NA
##    Thermal Stress: NA, Previous Impact: NA
##    Vulnerability Score: NA
```

```r
# Statistical summaries by response category
cat("\n=== RESPONSE CATEGORY ANALYSIS ===\n")
```

```
##
## === RESPONSE CATEGORY ANALYSIS ===
```

```r
category_analysis <- site_comprehensive %>%
  group_by(recovery_category) %>%
  summarise(
    n_sites = n(),
    mean_2023_bleaching = round(mean(predictor_2023_annual, na.rm = TRUE), 1),
    mean_2024_dhw = round(mean(dhw_2024, na.rm = TRUE), 1),
    mean_temp_instability = round(mean(temp_instability, na.rm = TRUE), 2),
    mean_vulnerability = round(mean(vulnerability_score, na.rm = TRUE), 3),
    .groups = "drop"
  ) %>%
  arrange(desc(n_sites))

print(category_analysis)
```

```
## # A tibble: 5 x 6
##   recovery_category    n_sites mean_2023_bleaching mean_2024_dhw
##   <chr>                  <int>               <dbl>         <dbl>
## 1 Exceptional_Recovery      13                36.5          20.4
## 2 Stable                    10                20            22.1
## 3 Strong_Recovery            6                28.6          22.4
## 4 Moderate_Worsening         2                41.3          18.4
## 5 Moderate_Recovery          1                15            25.4
## # i 2 more variables: mean_temp_instability <dbl>, mean_vulnerability <dbl>
```

```r
# Thermal stress pattern analysis
cat("\n=== THERMAL STRESS PATTERN EFFECTS ===\n")
```

```
##
## === THERMAL STRESS PATTERN EFFECTS ===
```

```r
thermal_analysis <- site_comprehensive %>%
  group_by(thermal_stress_level) %>%
  summarise(
    n_sites = n(),
    mean_response = round(mean(response_magnitude, na.rm = TRUE), 1),
    mean_recovery = round(mean(recovery_achieved, na.rm = TRUE), 1),
    recovery_sites = sum(recovery_achieved > 5, na.rm = TRUE),
    worsening_sites = sum(response_magnitude > 5, na.rm = TRUE),
    .groups = "drop"
  )

print(thermal_analysis)
```

```
## # A tibble: 2 x 6
##   thermal_stress_level n_sites mean_response mean_recovery recovery_sites
##   <chr>                  <int>         <dbl>         <dbl>          <int>
## 1 Extreme_Stress            31         -21.2          21.7             19
## 2 Low_Stress                 1         -47.5          47.5              1
## # i 1 more variable: worsening_sites <int>
```

```r
# Previous impact pattern analysis
cat("\n=== PREVIOUS YEAR IMPACT EFFECTS ===\n")
```

```
##
## === PREVIOUS YEAR IMPACT EFFECTS ===
```

```r
previous_impact_analysis <- site_comprehensive %>%
  group_by(previous_impact_level) %>%
  summarise(
    n_sites = n(),
    mean_response = round(mean(response_magnitude, na.rm = TRUE), 1),
    mean_recovery = round(mean(recovery_achieved, na.rm = TRUE), 1),
    proportion_recovering = round(mean(recovery_achieved > 5, na.rm = TRUE), 2),
    .groups = "drop"
  )

print(previous_impact_analysis)
```

```
## # A tibble: 4 x 5
##   previous_impact_level n_sites mean_response mean_recovery
##   <chr>                   <int>         <dbl>         <dbl>
## 1 Low_2023_Impact            10         -22.4          23.2
## 2 Minimal_2023_Impact         5          -6.4           6.6
## 3 Moderate_2023_Impact       14         -28.7          28.7
## 4 Severe_2023_Impact          3         -16.1          17.8
## # i 1 more variable: proportion_recovering <dbl>
```

```r
# Site clustering by vulnerability and response
cat("\n=== SITE VULNERABILITY CLUSTERS ===\n")
```

```
##
## === SITE VULNERABILITY CLUSTERS ===
```

```r
# High vulnerability, different outcomes
high_vulnerability <- site_comprehensive %>%
```

```r
  filter(vulnerability_score > 0.6) %>%
  select(site, vulnerability_score, recovery_achieved, response_magnitude,
         predictor_2023_annual, dhw_2024) %>%
  arrange(desc(vulnerability_score))

cat("High Vulnerability Sites (>0.6):\n")
```

```
## High Vulnerability Sites (>0.6):
```

```r
print(high_vulnerability, n = Inf)
```

```
## # A tibble: 31 x 6
##    site                     vulnerability_score recovery_achieved response_magnitude
##    <chr>                                  <dbl>             <dbl>              <dbl>
##  1 Coculus Rock                           1.06             37.7             -3.77e+ 1
##  2 Lang Bank EEMP                         1.04             15.8             -1.58e+ 1
##  3 Coral Bay                              1.01             21.3             -2.13e+ 1
##  4 Seahorse Cottage Sh~                   0.963            37.9             -3.79e+ 1
##  5 Flat Cay                               0.957             0                5.14e+ 0
##  6 Black Point                            0.917            52.2             -5.22e+ 1
##  7 Castle                                 0.902            28.3             -2.83e+ 1
##  8 Magens Bay                             0.891            52.4             -5.24e+ 1
##  9 Fish Bay                               0.889            31.7             -3.17e+ 1
## 10 Lang Bank Red Hind ~                   0.884             0                0
## 11 Salt River Deep                        0.871             9               -9    e+ 0
## 12 Eagle Ray                              0.861            33.3             -3.33e+ 1
## 13 Jacks Bay                              0.852             0                0
## 14 Botany Bay                             0.850            32.6             -3.26e+ 1
## 15 Sprat Hole                             0.840            64.2             -6.42e+ 1
## 16 Brewers Bay                            0.839             0                3.55e-15
## 17 Great Pond                             0.822            81.4             -8.14e+ 1
## 18 St James                               0.822             0                0
## 19 South Capella                          0.815             0.833           -8.33e- 1
## 20 Meri Shoal                             0.804             4.17            -4.17e+ 0
## 21 Savana                                 0.801            42.3             -4.23e+ 1
## 22 Cane Bay                               0.794            44.8             -4.48e+ 1
## 23 Grammanik Tiger FSA                    0.788             0.222           -2.22e- 1
## 24 Buck Island STX Deep                   0.782             0.833           -8.33e- 1
## 25 Kings Corner                           0.770            32.5             -3.25e+ 1
## 26 Cane Bay Deep                          0.764             0                0
## 27 Buck Island STT                        0.763            17.2             -1.72e+ 1
## 28 Mutton Snapper FSA                     0.706            16.7             -1.67e+ 1
## 29 South Water                            0.659            15.6             -1.56e+ 1
## 30 College Shoal East                     0.628             0                8.33e+ 0
## 31 Hind Bank East FSA                     0.608             0                1.17e+ 0
## # i 2 more variables: predictor_2023_annual <dbl>, dhw_2024 <dbl>
```

```r
# Low vulnerability, different outcomes
low_vulnerability <- site_comprehensive %>%
  filter(vulnerability_score < 0.3) %>%
  select(site, vulnerability_score, recovery_achieved, response_magnitude,
         predictor_2023_annual, dhw_2024) %>%
  arrange(vulnerability_score)

cat("\nLow Vulnerability Sites (<0.3):\n")
```

```
##
## Low Vulnerability Sites (<0.3):
```

```r
print(low_vulnerability, n = Inf)
```

```
## # A tibble: 0 x 6
## # i 6 variables: site <chr>, vulnerability_score <dbl>,
## #   recovery_achieved <dbl>, response_magnitude <dbl>,
## #   predictor_2023_annual <dbl>, dhw_2024 <dbl>
```

```r
# Key numerical insights
cat("\n=== KEY NUMERICAL INSIGHTS ===\n")
```

```
##
## === KEY NUMERICAL INSIGHTS ===
```

```r
total_sites <- nrow(site_comprehensive)
sites_with_recovery <- sum(site_comprehensive$recovery_achieved > 5, na.rm = TRUE)
sites_with_worsening <- sum(site_comprehensive$response_magnitude > 5, na.rm = TRUE)
sites_stable <- sum(abs(site_comprehensive$response_magnitude) <= 5, na.rm = TRUE)

strong_recovery_sites <- sum(site_comprehensive$recovery_achieved > 15, na.rm = TRUE)
exceptional_recovery_sites <- sum(site_comprehensive$recovery_achieved > 30, na.rm = TRUE)

mean_2023_bleaching <- round(mean(site_comprehensive$predictor_2023_annual, na.rm = TRUE), 1)
mean_2024_dhw <- round(mean(site_comprehensive$dhw_2024, na.rm = TRUE), 1)
mean_response <- round(mean(site_comprehensive$response_magnitude, na.rm = TRUE), 1)

cat(sprintf("Total analyzed sites: %d\n", total_sites))
```

```
## Total analyzed sites: 32
```

```r
cat(sprintf("Sites showing recovery (>5%% reduction): %d (%.1f%%)\n",
            sites_with_recovery, sites_with_recovery/total_sites*100))
```

```
## Sites showing recovery (>5% reduction): 20 (62.5%)
```

```r
cat(sprintf("Sites showing worsening (>5%% increase): %d (%.1f%%)\n",
            sites_with_worsening, sites_with_worsening/total_sites*100))
```

```
## Sites showing worsening (>5% increase): 2 (6.2%)
```

```r
cat(sprintf("Stable sites (±5%% change): %d (%.1f%%)\n",
            sites_stable, sites_stable/total_sites*100))
```

```
## Stable sites (±5% change): 10 (31.2%)
```

```r
cat(sprintf("Strong recovery sites (>15%% reduction): %d\n", strong_recovery_sites))
```

```
## Strong recovery sites (>15% reduction): 19
```

```r
cat(sprintf("Exceptional recovery sites (>30%% reduction): %d\n", exceptional_recovery_sites))
```

```
## Exceptional recovery sites (>30% reduction): 13
```

```r
cat(sprintf("Mean 2023 bleaching extent: %.1f%%\n", mean_2023_bleaching))
```

```
## Mean 2023 bleaching extent: 29.5%
```

```r
cat(sprintf("Mean 2024 maximum DHW: %.1f\n", mean_2024_dhw))
```

```
## Mean 2024 maximum DHW: 21.3
```

```r
cat(sprintf("Mean 2024→2025 response: %.1f%%\n", mean_response))
```

```
## Mean 2024→2025 response: -22.1%
```

```r
# Generate comprehensive site ranking
site_ranking <- site_comprehensive %>%
  mutate(
    resilience_score =
      ifelse(recovery_achieved > 0, recovery_achieved, 0) * 0.6 +  # 60% recovery achieved
      pmax(0, -response_magnitude) * 0.4,  # 40% resistance to worsening

    rank_recovery = rank(desc(recovery_achieved), ties.method = "min"),
    rank_resilience = rank(desc(resilience_score), ties.method = "min"),
    rank_vulnerability = rank(vulnerability_score, ties.method = "min")
  ) %>%
  select(site, recovery_achieved, response_magnitude, resilience_score, vulnerability_score,
         rank_recovery, rank_resilience, rank_vulnerability, thermal_stress_level,
         previous_impact_level, predictor_2023_annual, dhw_2024) %>%
  arrange(rank_resilience)

# Save all analysis results
write_csv(site_comprehensive, "06_site_comprehensive_analysis.csv")
write_csv(site_ranking, "06_site_resilience_ranking.csv")
write_csv(category_analysis, "06_response_category_summary.csv")
write_csv(thermal_analysis, "06_thermal_stress_analysis.csv")
write_csv(previous_impact_analysis, "06_previous_impact_analysis.csv")

cat("\n=== ANALYSIS COMPLETE ===\n")
```

```
##
## === ANALYSIS COMPLETE ===
```

```r
cat("Site-specific analysis files saved:\n")
```

```
## Site-specific analysis files saved:
```

```r
cat("- 06_site_comprehensive_analysis.csv\n")
```

```
## - 06_site_comprehensive_analysis.csv
```

```r
cat("- 06_site_resilience_ranking.csv\n")
```

```
## - 06_site_resilience_ranking.csv
```

```r
cat("- 06_response_category_summary.csv\n")
```

```
## - 06_response_category_summary.csv
```

```r
cat("- 06_thermal_stress_analysis.csv\n")
```

```
## - 06_thermal_stress_analysis.csv
```

```r
cat("- 06_previous_impact_analysis.csv\n")
```

```
## - 06_previous_impact_analysis.csv
```