report for erosion

Yingmai Chen

```
library(ggplot2)
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.1     v readr     2.1.4
v forcats   1.0.0     v stringr   1.5.0
v lubridate 1.9.2     v tibble    3.2.1
v purrr     1.0.2     v tidyr     1.3.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```
library(readr)
library(gridExtra)
```

```
Attaching package: 'gridExtra'

The following object is masked from 'package:dplyr':

    combine
```

```
library(car)
```

```
Loading required package: carData

Attaching package: 'car'
```

```
The following object is masked from 'package:dplyr':

    recode

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

    some
```

```r
library(grid)
library(stats)
```

read data and normalize the data

```r
stan <- function(x) {
  return((x - min(x)) / (max(x) - min(x)))
}
data1 <- read_csv("erosion.csv")# data include max wave height
```

```
Rows: 31 Columns: 8
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (1): Bluff
dbl (7): Orientation (deg), RR (m/yr), Max Wave Height (m), Mud (%), BaseEl ...

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
names(data1) <- gsub(" ", "_", names(data1))
names(data1) <- gsub("\\(", "", names(data1))
names(data1) <- gsub("\\)", "", names(data1))
names(data1) <- gsub("/", "_per_", names(data1))
names(data1) <- gsub("%", "percent", names(data1))
if ("RR (m/yr)" %in% names(data1)) {
  names(data1)[names(data1) == "RR (m/yr)"] <- "RR_m_per_yr"
}

numeric_columns <- sapply(data1, is.numeric) & !names(data1) %in% c("Bluff", "RR_m_per_yr"
data1[numeric_columns] <- lapply(data1[numeric_columns], stan)
```

```r
data1$RR_m_per_yr <- stan(data1$RR_m_per_yr)
```

```r
data2 <- read_csv("erosionnne15.csv")# data include NNE 15 m/s of max waveheight
```

```
Rows: 31 Columns: 8
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (1): Bluff
dbl (7): Orientation (deg), RR (m/yr), Wave Height for NNE wind 15 m/s (m), ...

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
names(data2) <- gsub(" ", "_", names(data2))
names(data2) <- gsub("\\(", "", names(data2))
names(data2) <- gsub("\\)", "", names(data2))
names(data2) <- gsub("/", "_per_", names(data2))
names(data2) <- gsub("%", "percent", names(data2))
if ("RR (m/yr)" %in% names(data2)) {
  names(data2)[names(data2) == "RR (m/yr)"] <- "RR_m_per_yr"
}
numeric_columns <- sapply(data2, is.numeric) & !names(data2) %in% c("Bluff", "RR_m_per_yr"
data2[numeric_columns] <- lapply(data2[numeric_columns], stan)
data2$RR_m_per_yr <- stan(data2$RR_m_per_yr)
```

Scatterplot for each variable and boxplot for categorical variable seaWall

correlated data

linear regression for data1(with max wave height)

```r
target <- "RR_m_per_yr"
predictors <- setdiff(names(data1), c("Bluff", target))
model_formula <- as.formula(paste(target, "~", paste(predictors, collapse = " + ")))

model1 <- lm(model_formula, data = data1)
print(summary(model1))
```

```
Call:
lm(formula = model_formula, data = data1)

Residuals:
     Min       1Q   Median       3Q      Max
-0.29687 -0.08353 -0.02441  0.06200  0.48000

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        0.14438    0.10996   1.313  0.20159
Orientation_deg    0.04229    0.10612   0.399  0.69376
Max_Wave_Height_m  0.52523    0.16321   3.218  0.00368 **
Mud_percent       -0.14023    0.12099  -1.159  0.25785
BaseEl_m          -0.05829    0.20675  -0.282  0.78043
BluffEl_m         -0.31922    0.14579  -2.190  0.03851 *
Seawall           -0.03885    0.10377  -0.374  0.71143
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1738 on 24 degrees of freedom
Multiple R-squared:  0.478, Adjusted R-squared:  0.3475
F-statistic: 3.662 on 6 and 24 DF,  p-value: 0.01006
```

```r
predictions <- predict(model1, data1)
mse <- mean((data1[[target]] - predictions)^2)
rsquared <- summary(model1)$r.squared
cat("(MSE):", mse, "\n")
```

```
(MSE): 0.02337344
```

```r
cat("R square:", rsquared, "\n")
```

```
R square: 0.4779682
```