Eden Landing Vegetation Data

Scott F. Jones

2022-09-16

Purpose

This document outlines the processing of USGS-collected plant data from Eden Landing in San Francisco Bay, California, USA. These data were collected as part of a San Francisco Bay Regional Monitoring Program funded project aiming to better understand sediment delivery from mudflats to salt marshes (co-PIs Drs. Karen Thorne [USGS WERC] and Jessie Lacy [USGS PCMSC]). Plots were measured in three different zones of the marsh (bay-side, channel-side, and interior). At each zone, plots were laid out along 3 replicate transects laid perpendicular to the water body, extending from 0.5 m to > 50 m from the edge.

Please see metadata tab included on final exported Excel spreadsheet for additional information on data use.

Data import and processing

Import raw plot-level field data. All plots had average vegetation height and visual cover estimates recorded in 1 m-2 quadrats. Additionally, a subset of plots had average vegetation height, visual cover estimates, live and dead stem widths (average of 5 measurements), live and dead stem percent by length, and total stem length measured in 0.1 m-2 quadrats.

```
data_plants_0.1 = read.csv("veg_density_small.csv")
data_plants_1 = read.csv("veg_density_large.csv")
```

Calculate vegetation volume

Total vegetation volume (cm-3) was estimated first using a simple approach: 10 cm * 10 cm * average plant height (cm) * percent plant cover. We expected this volume ('vol_estimate') to be much higher than actual volume since plants are not continuous.

To address this complication of succulent vegetation, we also calculated volume using measurements of stem length and width in 0.1 m-2 plots. We used these fine-scale measurements to calculate volume ('vol_measured') in the same areas that we estimated volume using the simple approach.

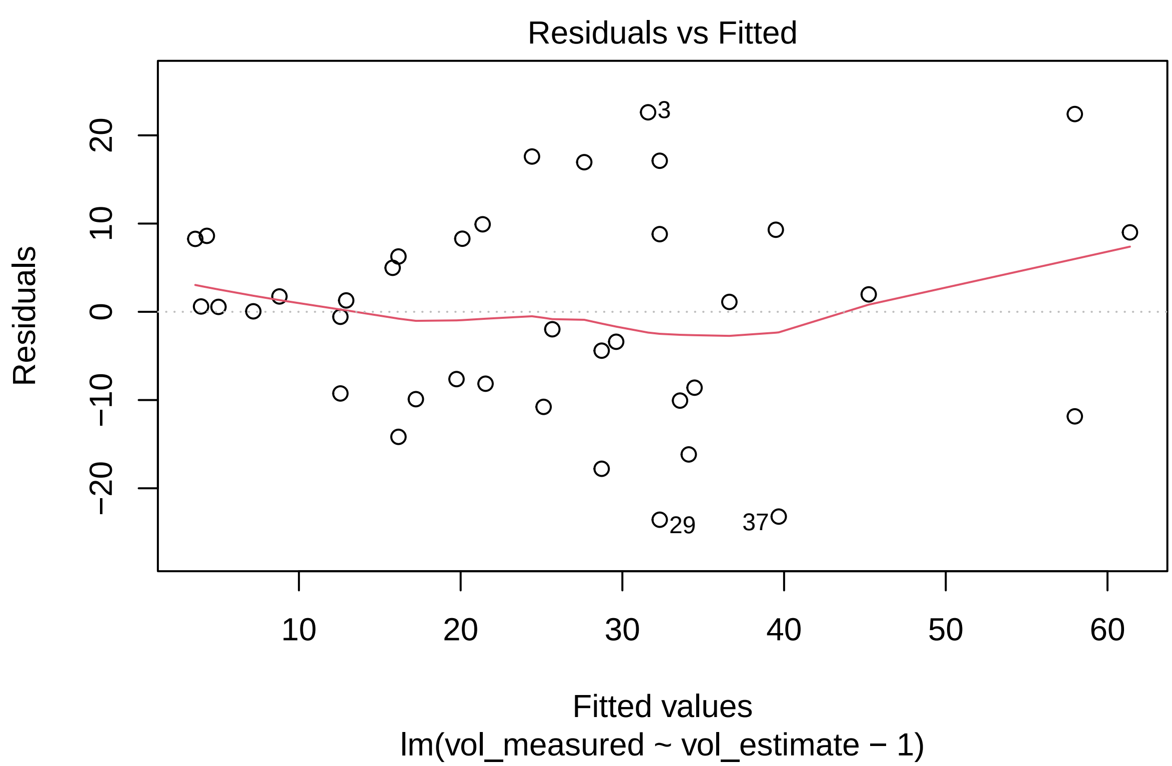
```
data_plants_small = data_plants_0.1 %>%
  mutate(vol_estimate = 10 * 10 * ave_ht * cover,
         vol_measured = ((pi * (stem_width_l/2) ^2) * stem_length * stem_perc_l) +
           ((pi * (stem_width_d/2) ^2) * stem_length * stem_perc_d))
```

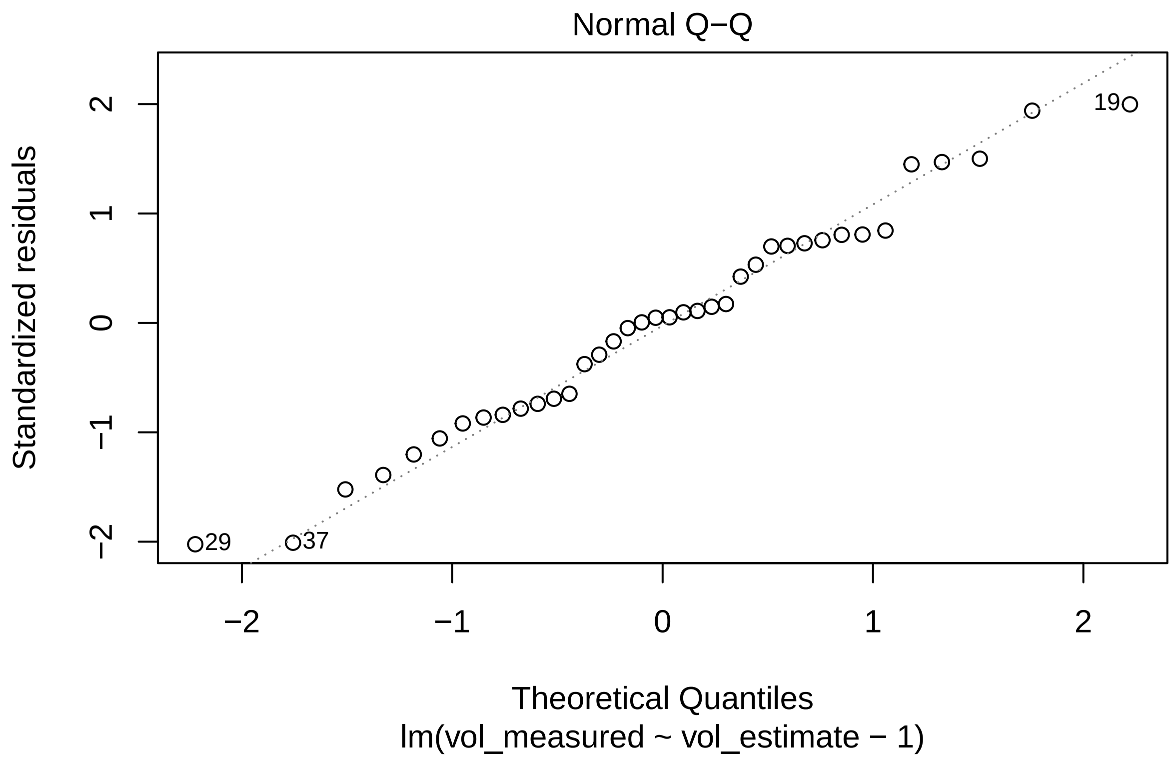
After calculating both methods for the same plot data, we fit a simple linear regression (summer + winter data combined) of calculated ~ estimated volume for small plots (n=39). We forced intercept of 0 so data could be fit to very small estimated volume plots at the Bay Edge site without negative values.

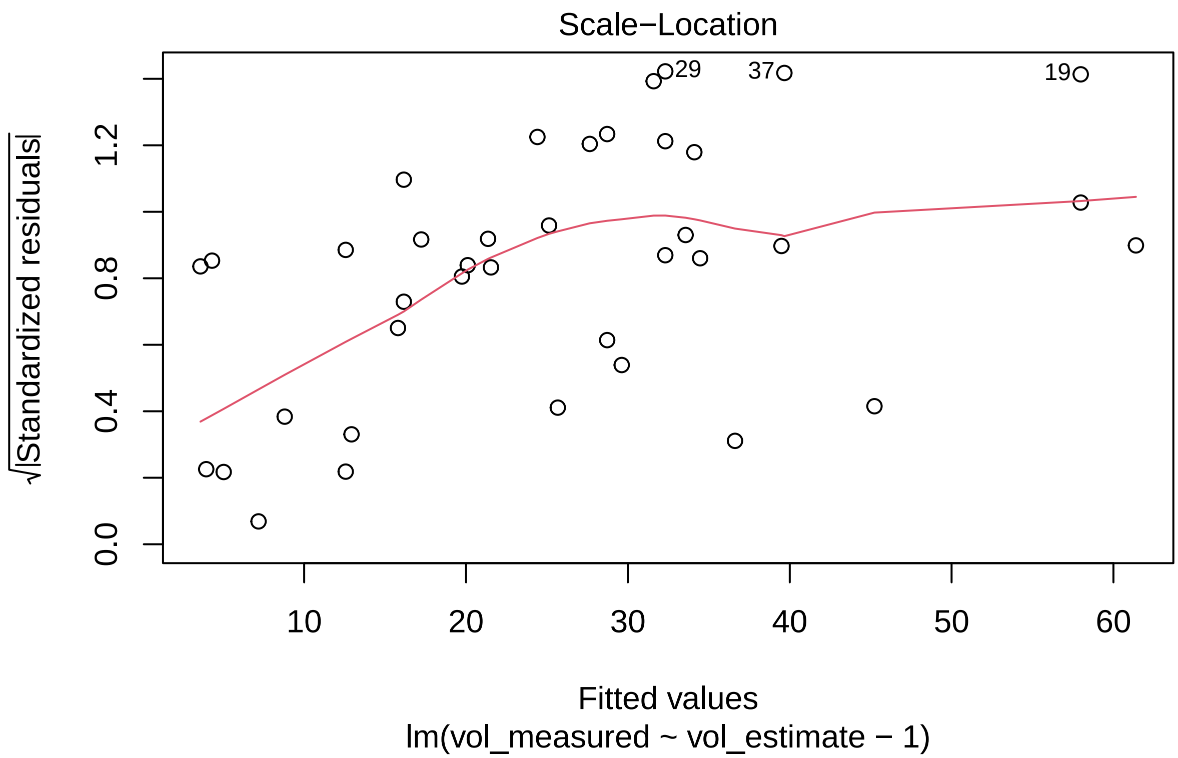
```
lm_volume = lm(data=data_plants_small, vol_measured ~ vol_estimate - 1)
summary(lm_volume)
```

```
##
## Call:
## lm(formula = vol_measured ~ vol_estimate - 1, data = data_plants_small)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -23.5639  -9.0895   0.5808   8.5278  22.6100
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## vol_estimate 0.035900   0.002329   15.41   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.83 on 37 degrees of freedom
## Multiple R-squared:  0.8652, Adjusted R-squared:  0.8616
## F-statistic: 237.6 on 1 and 37 DF,  p-value: < 2.2e-16
```

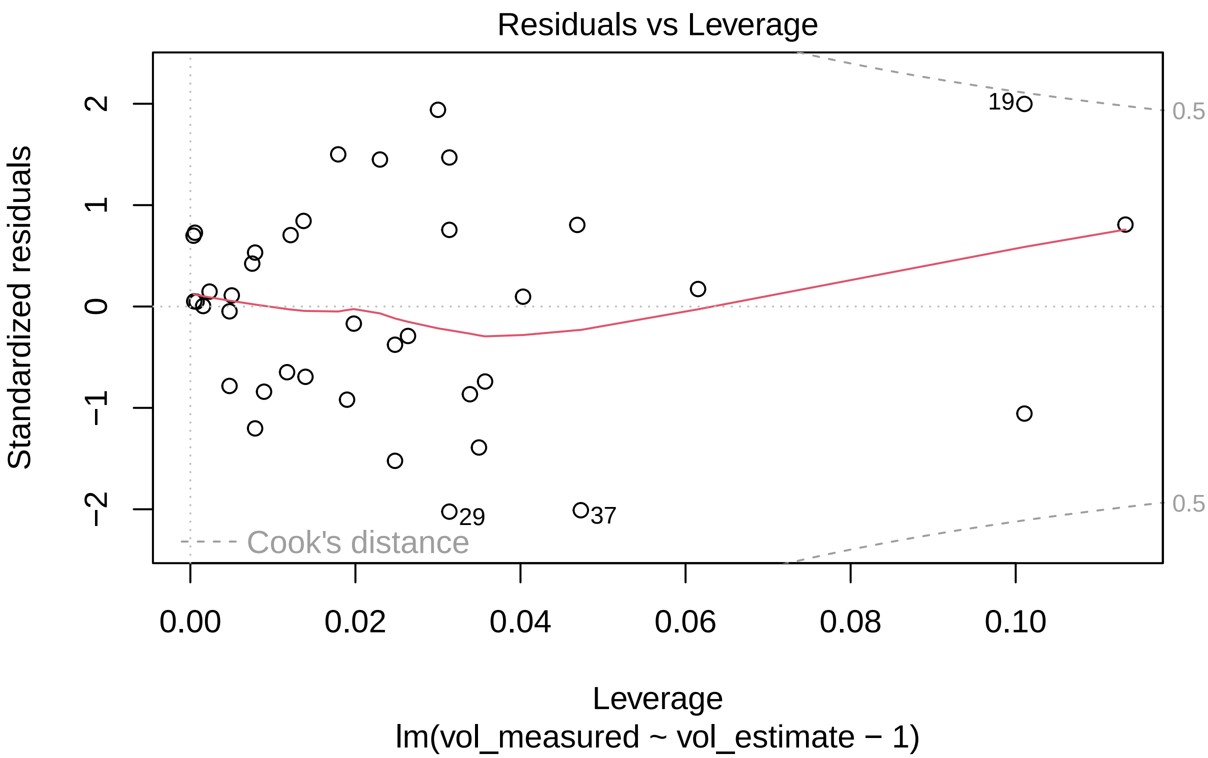
```
plot(lm_volume)
```




Normal Q−Q
Standardized residuals
−2
−1
0
1
2
29
37
19
−2
−1
0
1
2
Theoretical Quantiles
lm(vol_measured ~ vol_estimate − 1)



Scale−Location
√|Standardized residuals|
0.0
0.4
0.8
1.2
29
37
19
10
20
30
40
50
60
Fitted values
lm(vol_measured ~ vol_estimate − 1)



The regression model fits reasonably well to the data, with fairly homoscedastic variation and normally distributed residuals; model explains 86% of the variation in measured volume given estimated volume.

Due to the good fit, we applied the relationship between measured and estimated volumes to the large plot data. First we estimated volumes as before ('vol_est' in cm-3). Applying the regression relationship gives calculated volumes ('vol_calc' in cm-3) for each large plot in the study.

```
slope_regression = lm_volume$coefficients

data_plants_large = data_plants_1 %>%
  mutate(vol_est = 100 * 100 * ave_ht * cover,
         vol_calc = vol_est * slope_regression)
```

Calculate vegetation density

Vegetation density was determined from calculated volumes at each 1 m-2 plot. We calculated plant density using two approaches. First, we created an archetypal stem using the average live stem radius (cm) and average stem height (cm) across all plots. We then calculated density (# m-2) by dividing the calculated volume by the archetypal stem dimensions (assuming stems were cylinders)('density_naive').

We also calculated plant density ('density_final') using plot-specific plant heights, allowing each plot to have stems of the same width, but varying heights. This provides a more plot-specific view of density.

```
stem_radius = mean(data_plants_small$stem_width_l[data_plants_small$stem_width_l > 0]) / 2
stem_height = mean(data_plants_large$ave_ht)
```

```
data_plants_large = data_plants_large %>%
  mutate(density_naive = vol_calc / (pi * (stem_radius ^2) * stem_height),
         density_final = vol_calc / (pi * (stem_radius ^2) * ave_ht))
```

Calculate frontal area

To convert to frontal area (cm2 m-2), we multiplied the final density by plot-specific height and stem width (this is equivalent to multiplying calculated volume by 2 and then dividing by pi * stem radius).

We also calculated frontal area by individual strata, taking total area every 5 cm in height up to max height. This assumes vertical vegetation volume is distributed evenly across strata. Each strata contains only the frontal area calculated in that strata; to get larger strata sum together strata of interest.

```
data_plants_large = data_plants_large %>%
  mutate(frontal_total = density_final * (stem_radius*2) * ave_ht,
         frontal_5 = ifelse(ave_ht > 5, density_final * (stem_radius*2) * 5, density_final * (stem_radi
         frontal_10 = ifelse(ave_ht > 10, density_final * (stem_radius*2) * 5,
                             ifelse(ave_ht < 5, 0, density_final * (stem_radius*2) * (ave_ht-5))),
         frontal_15 = ifelse(ave_ht > 15, density_final * (stem_radius*2) * 5,
                             ifelse(ave_ht < 10, 0, density_final * (stem_radius*2) * (ave_ht-10))),
         frontal_20 = ifelse(ave_ht > 20, density_final * (stem_radius*2) * 5,
                             ifelse(ave_ht < 15, 0, density_final * (stem_radius*2) * (ave_ht-15))),
         frontal_25 = ifelse(ave_ht > 25, density_final * (stem_radius*2) * 5,
                             ifelse(ave_ht < 20, 0, density_final * (stem_radius*2) * (ave_ht-20))),
         frontal_30 = ifelse(ave_ht > 30, density_final * (stem_radius*2) * 5,
                             ifelse(ave_ht < 25, 0, density_final * (stem_radius*2) * (ave_ht-25))))
```

Export dataframe

Export final dataframe with all calculated values for further analysis/modeling efforts.

```
write.csv(data_plants_large, "USGS_veg_density_09.16.2022.csv", row.names = FALSE)
```