Discharge_GageHeight

Sylvia Hipp

2025-12-01

Goal of this analysis is to visualize suspended-sediment concentrations before, during, and after dam removal.

Data Source: Ritchie et al, "Morphodynamic evolution following sediment release from the world's largest dam removal", https://www.nature.com/articles/s41598-018-30817-8

–> Daily sediment loads during and after dam removal in the Elwha River, Washington, 2011 to 2016 –> TO DO: Download daily sediment concentrations

Set Up Environment

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- 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 become errors
```

```
library(here)
```

```
## here() starts at /home/guest/KukulaHipp
```

```
library(janitor)
```

```
##
## Attaching package: 'janitor'
##
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(ggplot2)
library(glue)
library(zoo)
```

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

```
library(cowplot)
```

```
##
## Attaching package: 'cowplot'
##
## The following object is masked from 'package:lubridate':
##
##     stamp
```

```
# check current working directory
here()
```

```
## [1] "/home/guest/KukulaHipp"
```

Set theme

```
kukulahipp_theme <- theme_classic() +
  theme(plot.title = element_text(size = 11, hjust = 0.5, face = "bold"), #Adjust title
        plot.subtitle = element_text(size = 10, hjust = 0.5),   # Adjust subtitle
        axis.text.x = element_text(size = 9), # Adjust x-axis values
        axis.text.y = element_text(size = 9), # Adjust y-axis values
        axis.title.x = element_text(size = 10.5), # Adjust x-axis title
        axis.title.y = element_text(size = 10.5), # Adjust y-axis title
        legend.position = "bottom", # Define legend position
        legend.text = element_text(size = 9), # Define legend entry sizes
        legend.title = element_text(size = 9), # Define legend name sizes
        )
theme_set(kukulahipp_theme)
```

Key Dates

```
complete_removal <- ymd("2014-08-26")
```

Download Raw Data

```
filepath <- list.files(here("Data/Raw/Elwha_DailySediment_2011_2016/"),
                       pattern = "\\.csv", full.names = TRUE)

daily_sediment_raw <-read_csv(file = filepath) %>%
  clean_names() %>%
  mutate(date = mdy(day)) %>%
  select(date, contains("ssc"), daily_suspended_sediment_load_tonnes)
```

```
## Rows: 1843 Columns: 17
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): Day, Remarks, Release period, Project year
## dbl (13): Daily Discharge (m3/s), Daily SSC (mg/L), Upper SSC bound (+1SD), ...
##
## 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.
```

Clean Data: Daily Sediment Concentrations

```
# categorize daily sediments for histogram
daily_sediment_df <- daily_sediment_raw %>%
  arrange(date) %>%
  mutate(
    daily_ssc_mg_l = zoo::na.approx(daily_ssc_mg_l),   # linear approximation
    # group concentrations into buckets
    concentration_mgL = case_when(daily_ssc_mg_l < 100 ~ "<100",
                                  daily_ssc_mg_l %>% between(100, 499.99) ~ "100-499",
                                  daily_ssc_mg_l %>% between(500, 999.99) ~ "500-999",
                                  daily_ssc_mg_l %>% between(1000, 4999.99) ~ "1000-4999",
                                  daily_ssc_mg_l %>% between(5000, 9999.99) ~ "5000-9999",
                                  daily_ssc_mg_l >= 10000 ~ ">10000"),
    concentration_mgL = factor(concentration_mgL,
                               levels = c("<100", "100-499", "500-999",
                                          "1000-4999", "5000-9999", ">10000")))
```

Visualize Data

Plot weekly concentration levels

```
# get composition of daily concentration for each week
weekly_conc_to_plot <- daily_sediment_df %>%
  mutate(weekday = as.POSIXlt(date)$wday,   # group dates into weeks (starting on Sunday of each week)
         week = date - weekday) %>%
  group_by(week, concentration_mgL) %>%
  summarize(n_days = n()) %>%
  ungroup() %>%
  mutate(pct_of_week = n_days/7)
```

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

```
sediment_conc_plot <- ggplot() +
  geom_col(data = weekly_conc_to_plot,
           aes(x = week, y = pct_of_week, fill = concentration_mgL), width = 8) +
  geom_vline(xintercept = complete_removal, color = "red", linetype = 2, linewidth = 0.75) +
  scale_fill_manual(values = c("#ECAD7C","#DE924F", "#B25900",
                               "#964B00", "#663300", "#433100")) +
  scale_x_date(expand = c(0,0))+
  scale_y_continuous(labels = scales::percent) +
  labs(x = NULL,
       y = "Share of Week",
       fill = "Sediment Concentration (mg/L)")
```

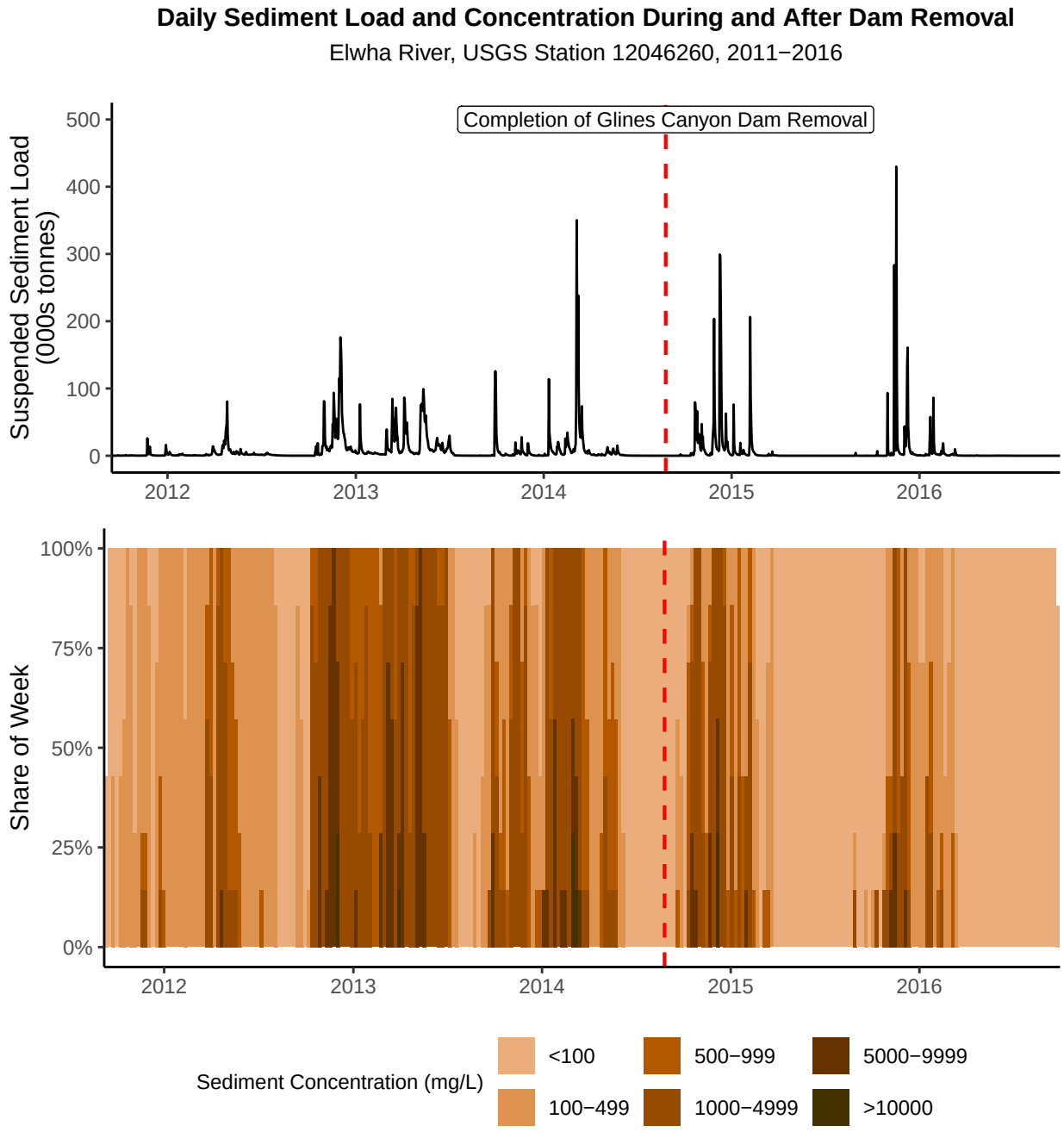
Plot daily suspended sendiment loads

```
sediment_load_plot <- ggplot() +
  geom_line(data = daily_sediment_df,
            aes(x = date, y = daily_suspended_sediment_load_tonnes/1000)) +
  geom_vline(xintercept = complete_removal, color = "red", linetype = 2, linewidth = 0.75) +
  geom_label(aes(x = complete_removal, y = 500,
                 label = "Completion of Glines Canyon Dam Removal"),
             fill = "white", size = 3) +
  scale_x_date(expand = c(0,0)) +
  labs(x = NULL,
       y = "Suspended Sediment Load\n(000s tonnes)",
       title = "Daily Sediment Load and Concentration During and After Dam Removal",
       subtitle = "Elwha River, USGS Station 12046260, 2011-2016\n")
```

Combine plots

```
plot_grid(plotlist = list(sediment_load_plot, sediment_conc_plot),
          ncol = 1, nrow = 2,
          rel_heights = c(0.45, 0.55))
```

```
## Warning: `position_stack()` requires non-overlapping x intervals.
```



Statistical Analysis: Sediment Concentration During and After Removal

Perform a t-test on average sediment concentration during and after dam removal to determine if there's a statistically significant difference

```
sediment_conc_during <- daily_sediment_df %>%
  filter(date <= complete_removal) %>%
  select(date, daily_ssc_mg_l)
```

```r
sediment_conc_after <- daily_sediment_df %>%
  filter(date > complete_removal) %>%
  select(date, daily_ssc_mg_l)

mean(sediment_conc_during$daily_ssc_mg_l)
```

```
## [1] 1267.684
```

```r
mean(sediment_conc_after$daily_ssc_mg_l)
```

```
## [1] 489.0397
```

```r
# statistically significant that the true mean sediment concentration
# during removal is greater than the mean after removal
# p-value < 2.2e-16
t.test(sediment_conc_during$daily_ssc_mg_l,
       sediment_conc_after$daily_ssc_mg_l,
       alternative = "greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  sediment_conc_during$daily_ssc_mg_l and sediment_conc_after$daily_ssc_mg_l
## t = 10.258, df = 1813.7, p-value < 2.2e-16
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  653.725     Inf
## sample estimates:
## mean of x mean of y
## 1267.6843  489.0397
```

Prepare a table

```r
ssc_means_table <- tibble(
  "Period" = c("During Removal", "After Removal"),
  "Average SSC (mg/L)" = c(round(mean(sediment_conc_during$daily_ssc_mg_l), digits=2),
                           round(mean(sediment_conc_after$daily_ssc_mg_l), digits=2))
)

knitr::kable(ssc_means_table,
             caption = "Suspended Sediment Concentration Before and After Dam Removal")
```

Table 1: Suspended Sediment Concentration Before and After Dam Removal

Period	Average SSC (mg/L)
During Removal	1267.68
After Removal	489.04