Seafood Analysis

Connor Quiroz

2024-11-15

Data Cleaning

```
# Read in data file
consumption <- read.csv("example_consumption_eez_2024_11_15.csv")
```

```
## Warning in scan(file = file, what = what, sep = sep, quote = quote, dec = dec,
## : embedded nul(s) found in input
```

```
unique(consumption$year)
```

```
##  [1] "1996" "1997" "1998" "1999" "2000" "2001" "2002" "2003" "2004" "2005"
## [11] "2006" "2007" "2008" "2009" "2010" "2011" "2012" "2013" "2014" "2015"
## [21] "2016" "2017" "2018" "2019" "KWT"
```

```
names(consumption)
```

```
## [1] "year"           "eez_iso3c"      "eez_name"       "producer_iso3c"
## [5] "consumer_iso3c" "sciname"        "dwf"            "live_weight_t"
```

```
# Looking at total weights per country (aggregating species weight in 2019)
con_weight <- consumption %>%
  filter(year == 2019) %>%
  group_by(eez_name) %>%
  mutate(live_weight_t = as.numeric(live_weight_t)) %>%
  summarize(total_weight = sum(live_weight_t)) %>%
  arrange(desc(total_weight))
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `live_weight_t = as.numeric(live_weight_t)`.
## i In group 80: `eez_name = "Maldives"`.
## Caused by warning:
## ! NAs introduced by coercion
```

```
# Shannon diversity per country
con_shannon <- consumption %>%
  filter(year == 2019) %>%
  group_by(eez_name, sciname) %>%
```

```
  mutate(individual_abundance = length(sciname)) %>%
  distinct(eez_name, sciname, .keep_all = TRUE) %>%  #
  group_by(eez_name) %>%
  summarize(
    total_abundance = sum(individual_abundance),
    pi = individual_abundance / total_abundance,
    shannon = -sum(pi * log(pi))
  ) %>%
  distinct(eez_name, .keep_all = TRUE) %>%
  arrange(desc(shannon)) %>%
  ungroup() %>%
  select(eez_name, shannon)
```

```
## Warning: Returning more (or less) than 1 row per `summarise()` group was deprecated in
## dplyr 1.1.0.
## i Please use `reframe()` instead.
## i When switching from `summarise()` to `reframe()`, remember that `reframe()`
##   always returns an ungrouped data frame and adjust accordingly.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

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

```
# Join datasets
con_joined <- left_join(con_weight, con_shannon, by = "eez_name")

# Sort countries in alphabetical order
con_joined %>%
  arrange(-desc(eez_name))
```

```
## # A tibble: 149 x 3
##    eez_name          total_weight shannon
##    <chr>                    <dbl>   <dbl>
##  1 Albania                  5351.    2.06
##  2 Algeria                234490.    3.67
##  3 Angola                 579957.    3.87
##  4 Antigua & Barbuda        3741.    2.85
##  5 Argentina             1382789.    3.61
##  6 Australia              203603.    5.47
##  7 Bahamas                 18899.    3.42
##  8 Bahrain                 40712.    3.69
##  9 Bangladesh            1114556.    4.46
## 10 Barbados                3057.    3.56
## # i 139 more rows
```

```
# Read in extraneous Adaptive Capacity data
hdi <- read.csv("QuirozConnor_Chapter1_Data.csv")

# Add in hdi values to joined dataset
con_joined <- left_join(con_joined, hdi, by = "eez_name")
```

```r
# Update joined dataset to have log transformed total weight as a variable
con_joined <- con_joined %>%
  mutate(log_weight = log(total_weight))

# Add in region (e.g., North America, Asia, etc.)
con_joined <- con_joined %>%
  mutate(
    region = case_when(
      eez_name %in% c(
        "Algeria",
        "Angola",
        "Benin",
        "Cameroon",
        "Cape Verde",
        "Comoros",
        "Congo - Brazzaville",
        "Congo - Kinshasa",
        "Côte d'Ivoire",
        "Djibouti",
        "Egypt",
        "Equatorial Guinea",
        "Eritrea",
        "Gabon",
        "Gambia",
        "Ghana",
        "Guinea",
        "Guinea-Bissau",
        "Kenya",
        "Liberia",
        "Libya",
        "Madagascar",
        "Malawi",
        "Mali",
        "Mauritania",
        "Mauritius",
        "Morocco",
        "Mozambique",
        "Namibia",
        "Niger",
        "Nigeria",
        "São Tomé & Príncipe",
        "Senegal",
        "Seychelles",
        "Sierra Leone",
        "Somalia",
        "South Africa",
        "Sudan",
        "Tanzania",
        "Togo",
        "Tunisia",
        "Zambia",
        "Zimbabwe"
      ) ~
```

```
    "Africa",

  eez_name %in% c(
    "Afghanistan",
    "Armenia",
    "Azerbaijan",
    "Bahrain",
    "Bangladesh",
    "Bhutan",
    "Brunei",
    "Cambodia",
    "China",
    "Cyprus",
    "Georgia",
    "India",
    "Indonesia",
    "Iran",
    "Iraq",
    "Israel",
    "Japan",
    "Jordan",
    "Kazakhstan",
    "Kuwait",
    "Kyrgyzstan",
    "Lebanon",
    "Malaysia",
    "Maldives",
    "Mongolia",
    "Myanmar (Burma)",
    "Nepal",
    "North Korea",
    "Oman",
    "Pakistan",
    "Palestinian Territories",
    "Philippines",
    "Qatar",
    "Russia",
    "Saudi Arabia",
    "Singapore",
    "South Korea",
    "Sri Lanka",
    "Syria",
    "Tajikistan",
    "Taiwan",
    "Thailand",
    "Timor-Leste",
    "Turkey",
    "United Arab Emirates",
    "Uzbekistan",
    "Vietnam",
    "Yemen"
  ) ~
    "Asia",
```

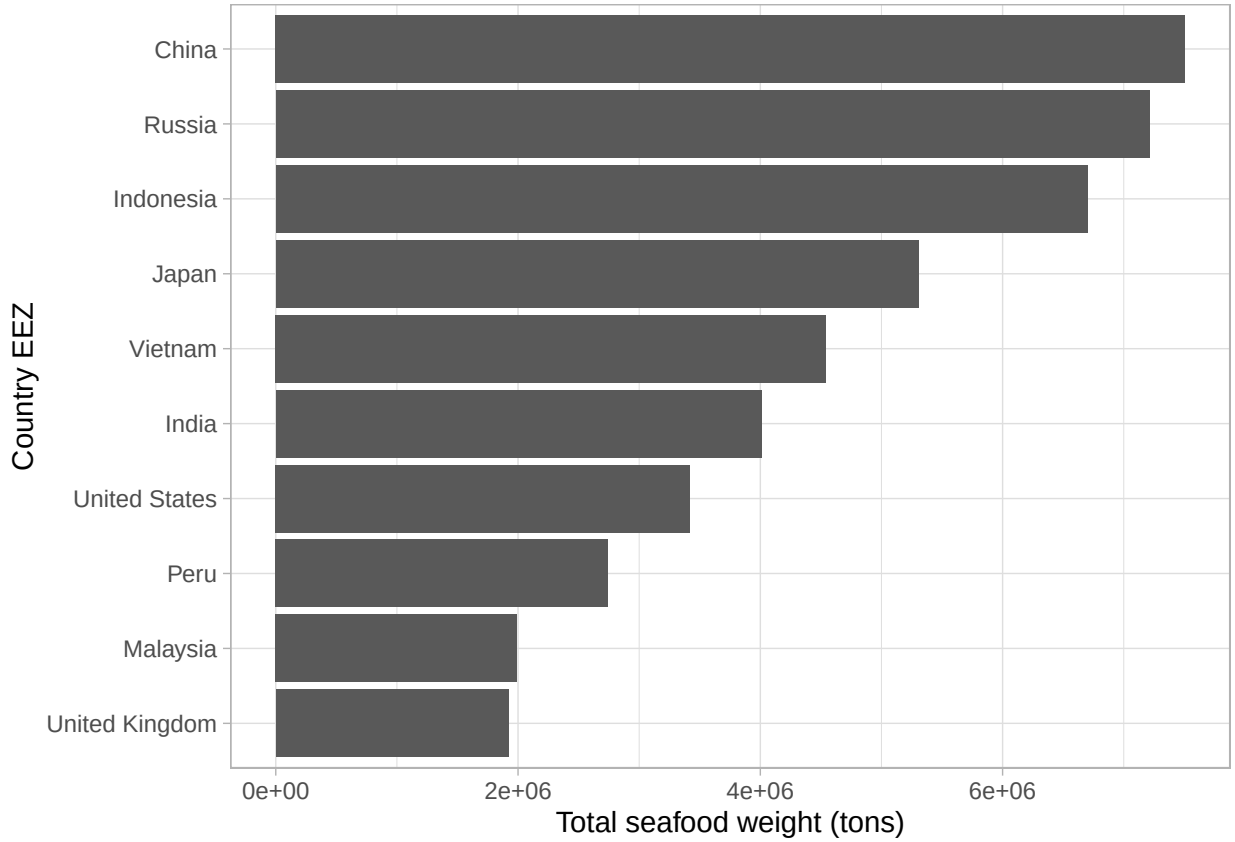
```
    eez_name %in% c(
      "Albania",
      "Belgium",
      "Bosnia & Herzegovina",
      "Bulgaria",
      "Croatia",
      "Cyprus",
      "Denmark",
      "Estonia",
      "Finland",
      "France",
      "Germany",
      "Greece",
      "Iceland",
      "Ireland",
      "Italy",
      "Latvia",
      "Lebanon",
      "Lithuania",
      "Luxembourg",
      "Malta",
      "Montenegro",
      "Netherlands",
      "Norway",
      "Poland",
      "Portugal",
      "Romania",
      "Russia",
      "Slovenia",
      "Spain",
      "Sweden",
      "Switzerland",
      "United Kingdom",
      "Ukraine"
    ) ~
      "Europe",

    eez_name %in% c(
      "Antigua & Barbuda",
      "Bahamas",
      "Barbados",
      "Belize",
      "Canada",
      "Costa Rica",
      "Cuba",
      "Dominica",
      "Dominican Republic",
      "Ecuador",
      "El Salvador",
      "Grenada",
      "Guatemala",
      "Haiti",
      "Honduras",
```

```
        "Jamaica",
        "Mexico",
        "Montserrat",
        "Nicaragua",
        "Panama",
        "St. Kitts & Nevis",
        "St. Lucia",
        "St. Vincent & Grenadines",
        "Trinidad & Tobago",
        "United States"
      ) ~
        "North America",

      eez_name %in% c(
        "Argentina",
        "Brazil",
        "Chile",
        "Colombia",
        "Ecuador",
        "Guyana",
        "Paraguay",
        "Peru",
        "Suriname",
        "Uruguay",
        "Venezuela"
      ) ~
        "South America",

      eez_name %in% c(
        "Australia",
        "Fiji",
        "Kiribati",
        "Micronesia (Federated States of)",
        "Nauru",
        "New Zealand",
        "Palau",
        "Papua New Guinea",
        "Samoa",
        "Solomon Islands",
        "Tonga",
        "Tuvalu",
        "Vanuatu"
      ) ~
        "Oceania",

      TRUE ~ "Unknown"
    )
  )

# Consumption data without NA's (for correlation plots)
con_na_removed <- con_joined %>%
  drop_na()
```

Data visualization + Analysis

```
# Top 10 total seafood weights by country
(plot_1 <- con_joined %>%
  arrange(desc(total_weight)) %>%
  select(eez_name, total_weight) %>%
  top_n(10) %>%
  ggplot(aes(x = total_weight,
             y = fct_reorder(eez_name, desc(-total_weight)))) +
  geom_col() +
  labs(x = "Total seafood weight (tons)", y = "Country EEZ") +
  theme_light())
```

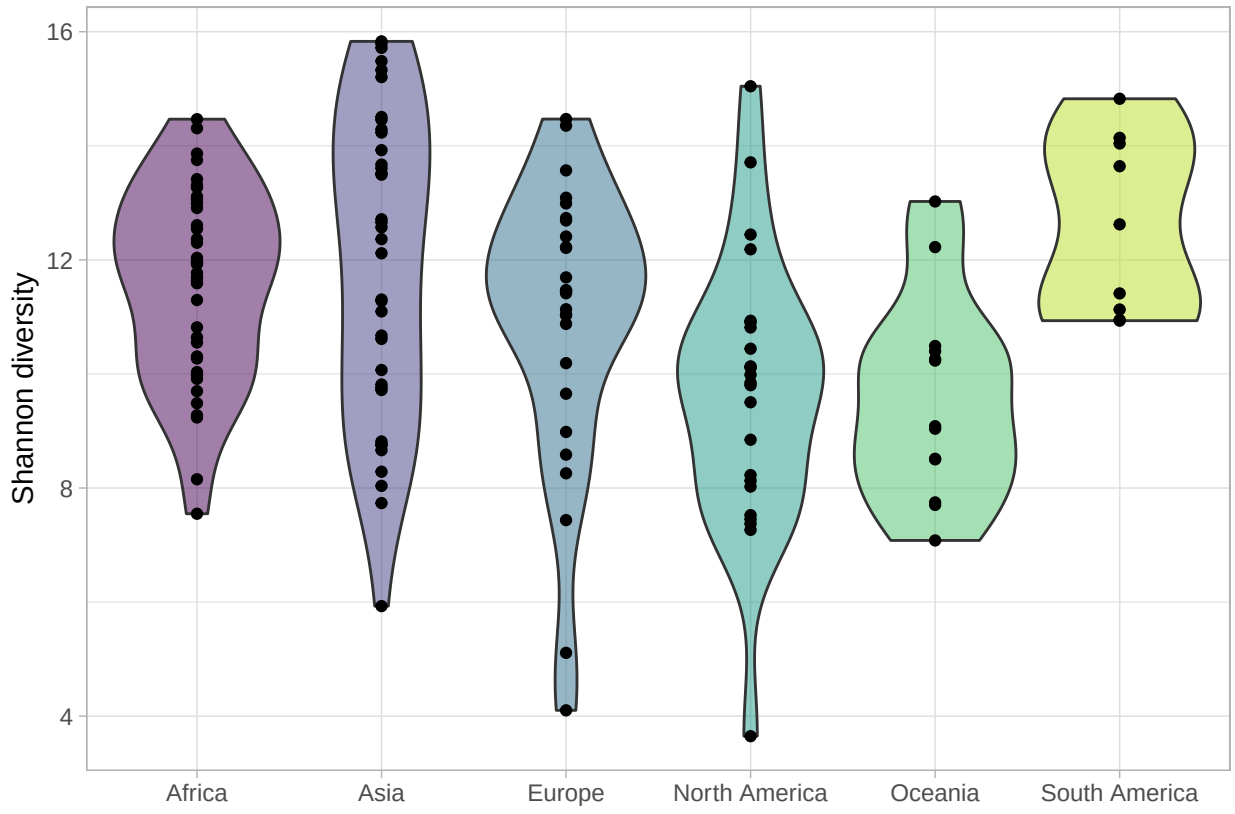
```
## Selecting by total_weight
```




```
# Seafood weight by region
(plot_2 <- con_joined %>%
  ggplot(aes(x = region, y = log(total_weight), fill = region)) +
  geom_violin(alpha = 0.5) +
  geom_point() +
  scale_fill_viridis_d(end = 0.9) +
  theme_light() +
  guides(fill = "none") +
  labs(x = "", y = "Shannon diversity"))
```

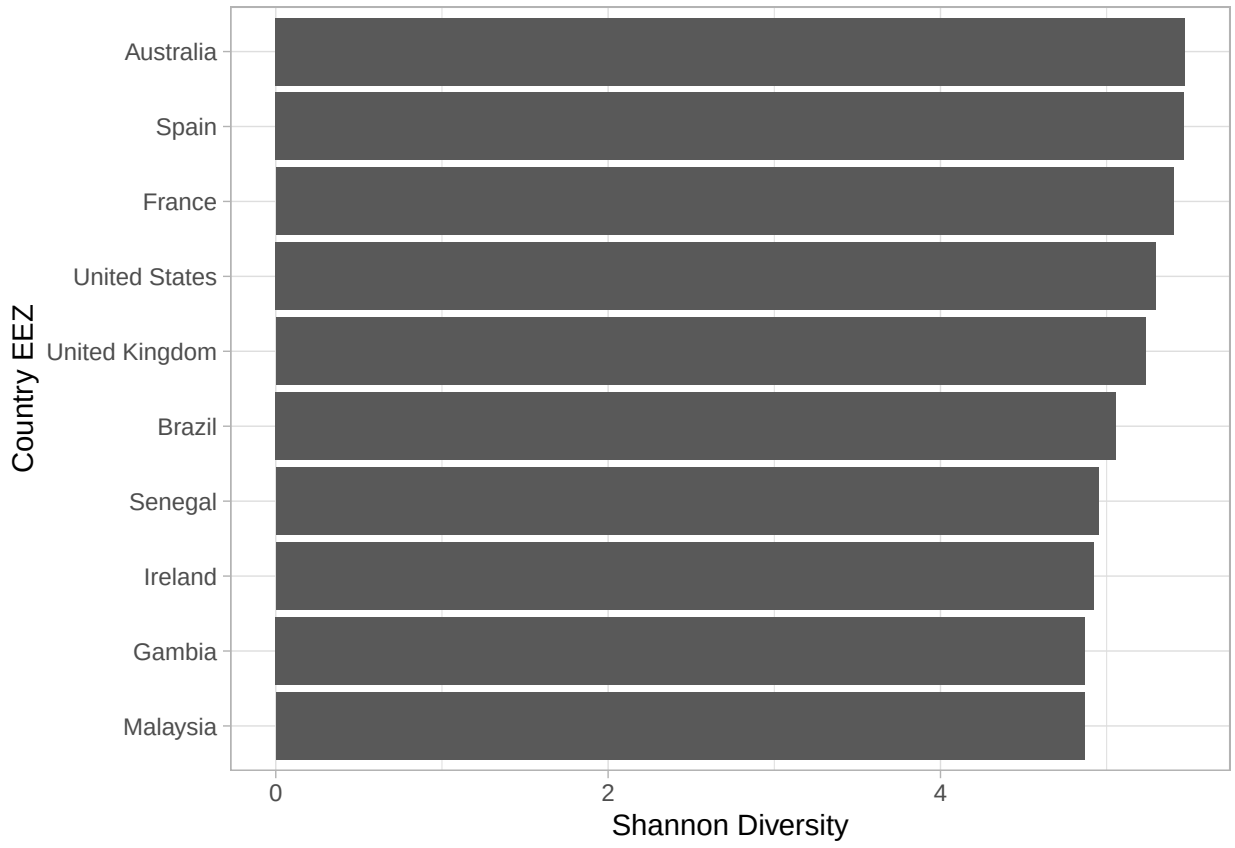
```
## Warning: Removed 1 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```

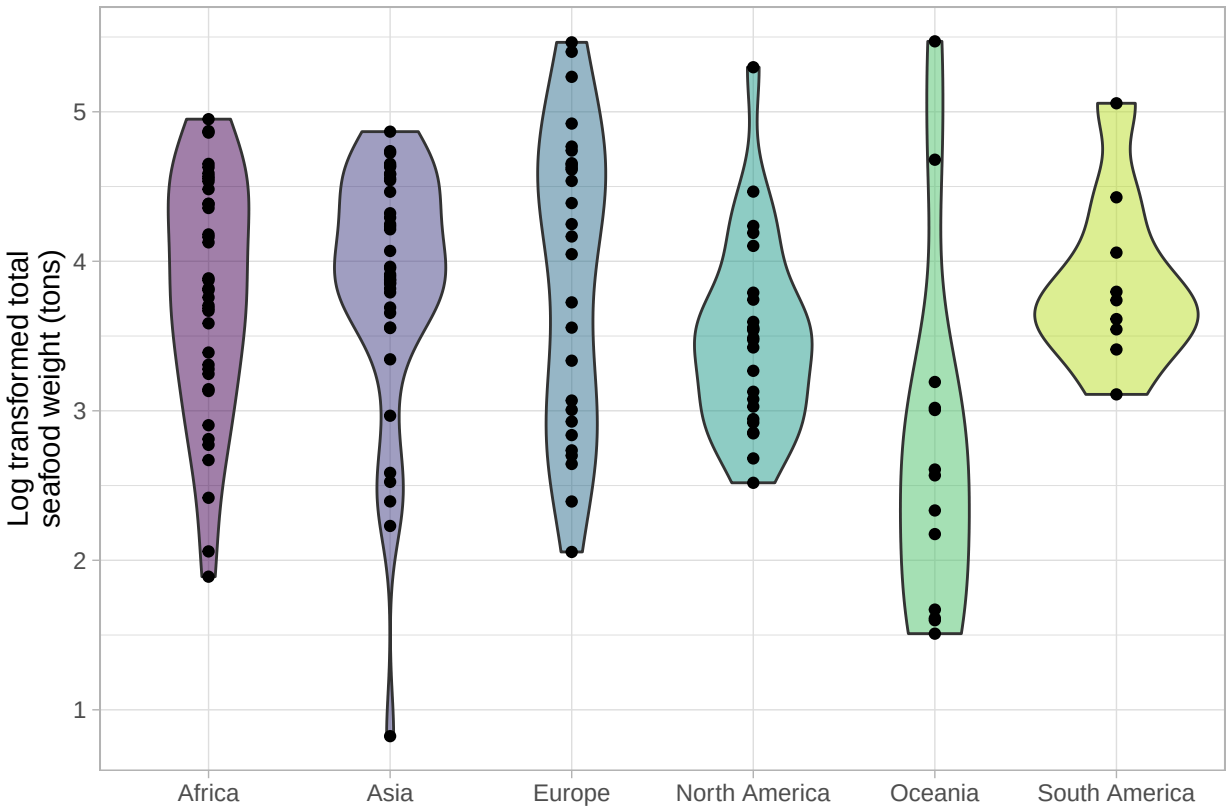



```
# Top 10 most diverse seafood by country
(plot_3 <- con_joined %>%
  arrange(desc(shannon)) %>%
  select(eez_name, shannon) %>%
  top_n(10) %>%
  ggplot(aes(x = shannon,
             y = fct_reorder(eez_name, desc(-shannon)))) +
  geom_col() +
  labs(x = "Shannon Diversity", y = "Country EEZ") +
  theme_light())
```

```
## Selecting by shannon
```




```
# Seafood diversity by region
(plot_4 <- con_joined %>%
  ggplot(aes(x = region, y = shannon, fill = region)) +
  geom_violin(alpha = 0.5) +
  geom_point() +
  scale_fill_viridis_d(end = 0.9) +
  theme_light() +
  guides(fill = "none") +
  labs(x = "", y = "Log transformed total\nseafood weight (tons)"))
```

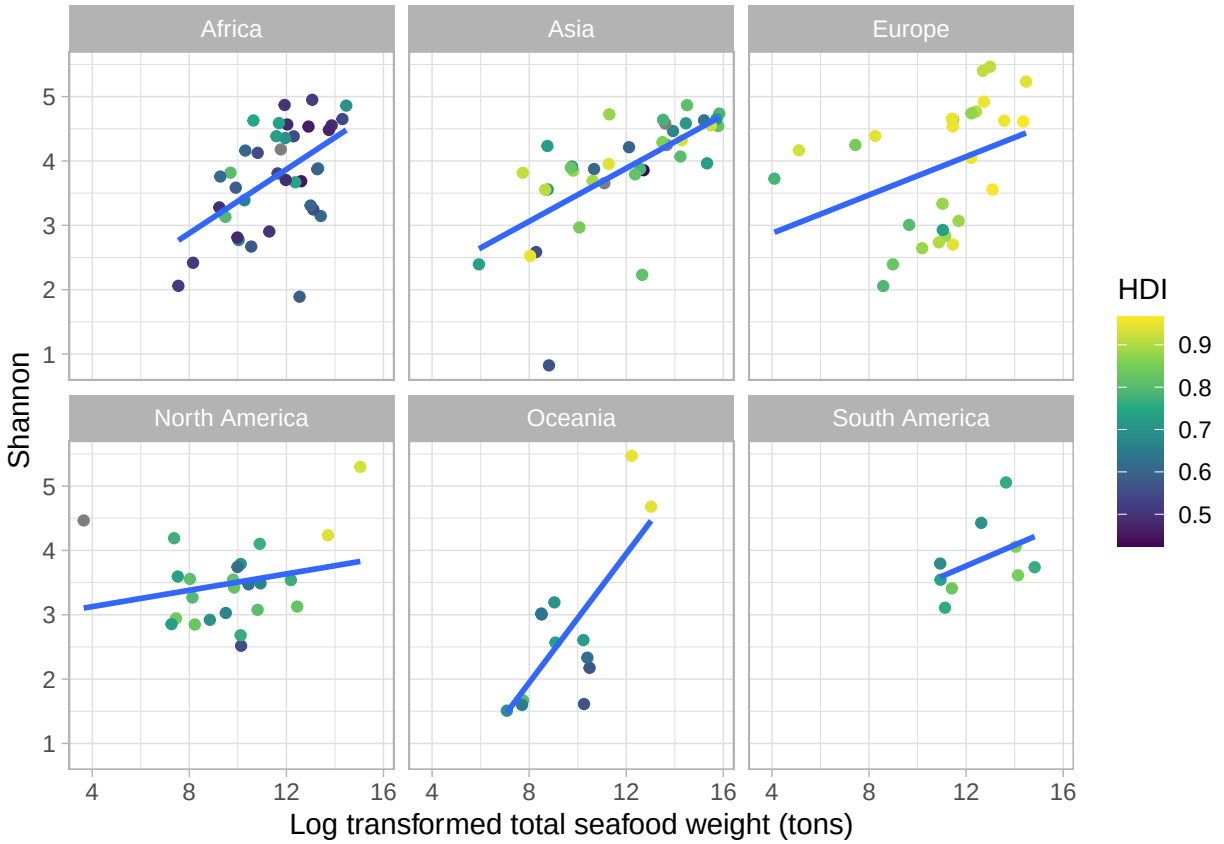



```
# Relationship between shannon diversity and the total weight by country
(plot_5 <- con_joined %>%
  ggplot(aes(x = log(total_weight), y = shannon, color = hdi)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  scale_color_viridis_c() +
  facet_wrap(~ region) +
  theme_light() +
  labs(x = "Log transformed total seafood weight (tons)", y = "Shannon", color = "HDI"))
```

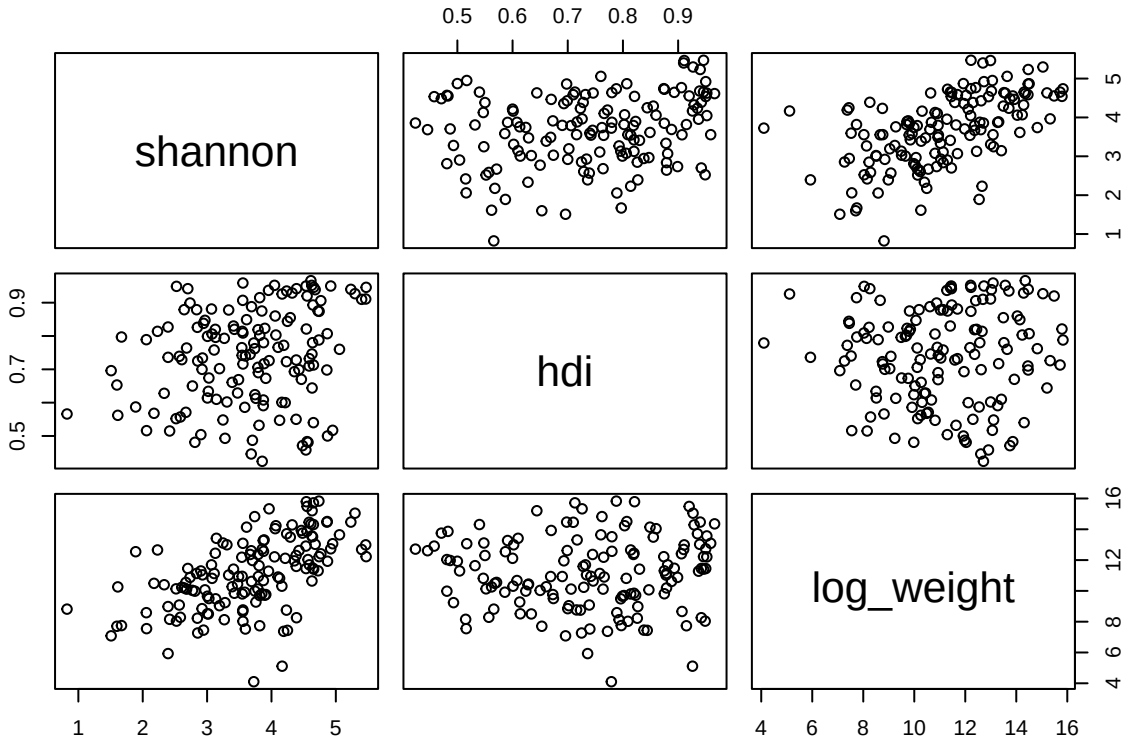
```
## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 1 rows containing non-finite values (`stat_smooth()`).

## Warning: The following aesthetics were dropped during statistical transformation: colour
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a `group` aesthetic or to convert a numerical
##   variable into a factor?
## The following aesthetics were dropped during statistical transformation: colour
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a `group` aesthetic or to convert a numerical
##   variable into a factor?
## The following aesthetics were dropped during statistical transformation: colour
```

```
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a ‘group‘ aesthetic or to convert a numerical
##   variable into a factor?
## The following aesthetics were dropped during statistical transformation: colour
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a ‘group‘ aesthetic or to convert a numerical
##   variable into a factor?
## The following aesthetics were dropped during statistical transformation: colour
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a ‘group‘ aesthetic or to convert a numerical
##   variable into a factor?
## The following aesthetics were dropped during statistical transformation: colour
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a ‘group‘ aesthetic or to convert a numerical
##   variable into a factor?
```

```
## Warning: Removed 1 rows containing missing values (‘geom_point()‘).
```




```
# Correlations between variables (plots + pearson)
pairs(con_na_removed[c(3:5)])
```




```
cor(con_na_removed[c(3:5)])
```

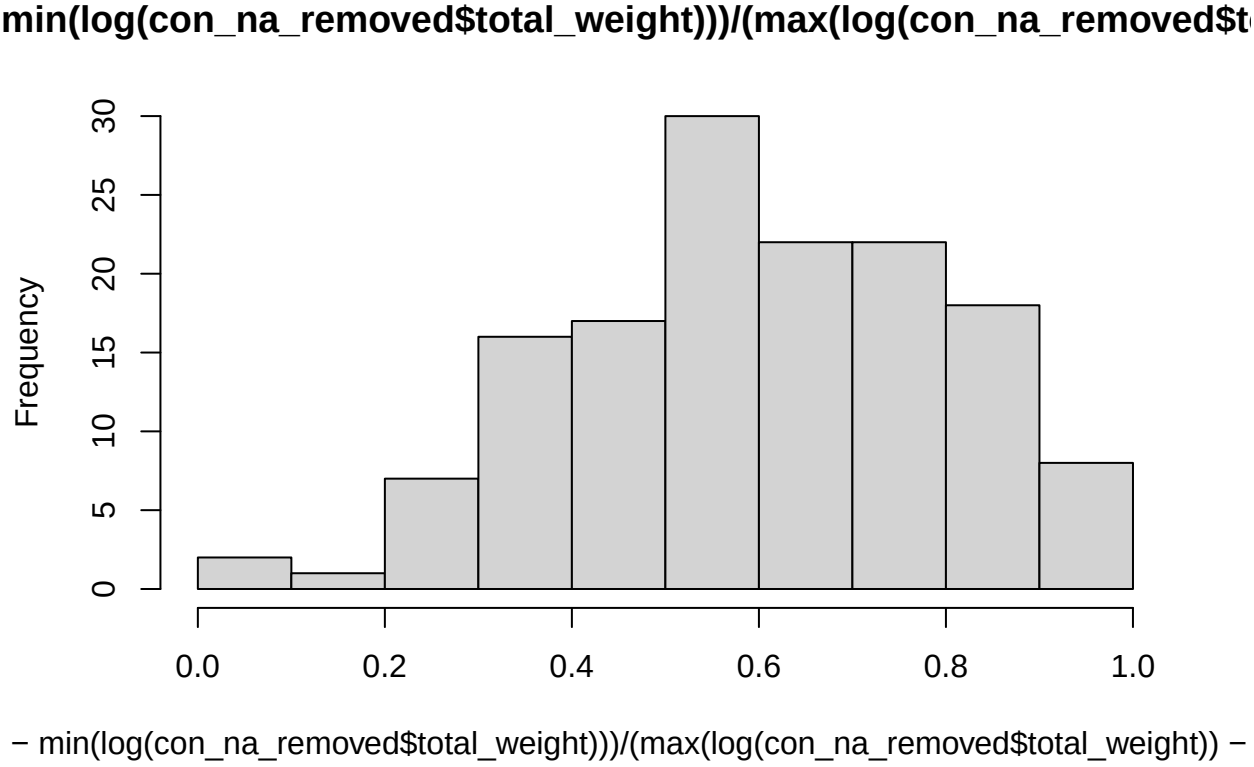
```
##                shannon        hdi log_weight
## shannon    1.0000000 0.23947808 0.56012176
## hdi        0.2394781 1.00000000 0.04360658
## log_weight 0.5601218 0.04360658 1.00000000
```

```
# Linear model between weight
weight_shannon_lm <- lm(shannon ~ log(total_weight), data = con_joined)
summary(weight_shannon_lm)
```

```
##
## Call:
## lm(formula = shannon ~ log(total_weight), data = con_joined)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.40564 -0.54173  0.01274  0.48515  2.26080
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        1.48186    0.30118   4.920 2.30e-06 ***
## log(total_weight)  0.19822    0.02647   7.489 6.08e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.7751 on 146 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.2775, Adjusted R-squared:  0.2726
## F-statistic: 56.08 on 1 and 146 DF,  p-value: 6.08e-12
```

```
# Testing out normalizing data - potentially might need to regularize if data is not normally distrubte
hist((log(con_na_removed$total_weight) - min(log(con_na_removed$total_weight))) / (max(log(con_na_remove
```



Takeaways: Little correlation between hdi and total weight + shannon diversity, but stronger, positive correlation between weight and shannon diversity (as seen in correlatnion + scatter plots)

Save Images

Dummy data testing to work with big joined datset

```
data.frame(x = c("bird", "bird", "fish", "fish", "fish", "goat", "goat"), y = c("US", "US", "US", "China
  group_by(x, y) %>%
  mutate(individual_abundance = length(x)) %>%
  distinct(x, y, .keep_all = TRUE) %>%  # Remove duplicate species-country combinations
  group_by(y) %>%  # Group by country
  mutate(total_abundance = sum(individual_abundance), pi = individual_abundance / total_abundance, shann
```

```
## # A tibble: 4 x 6
## # Groups:   y [2]
##   x     y     individual_abundance total_abundance    pi shannon
##   <chr> <chr>                <int>           <int> <dbl>   <dbl>
## 1 bird  US                       2               3 0.667   0.637
## 2 fish  US                       1               3 0.333   0.637
## 3 fish  China                    2               4 0.5     0.693
## 4 goat  China                    2               4 0.5     0.693
```