

Single Metric Assignment

SOLUTION KEY

Justin Pomeranz

This is the KEY

Problem 1

Single-site Richness Metrics

For all sub-problems in problem 1, you will only be using the `site1` data object.

Problem 1.1

- Calculate the richness (number of taxa) which are in the Order Plecoptera
- Save this result as a data object and print it to the console.

```
## # A tibble: 1 x 2
##   site_code p_rich
##   <chr>     <int>
## 1 A75CC         7
```

Problem 1.2

- Calculate the richness (number of taxa) which are in the Order Trichoptera
- Save this result as a data object
- join this result with the result calculated in 1.1 above and print it to the console.

```
## Joining with 'by = join_by(site_code)'
```

```
## # A tibble: 1 x 3
##   site_code p_rich t_rich
##   <chr>     <int> <int>
## 1 A75CC         7       4
```

Problem 1.3

- Calculate the richness (number of taxa) which are in the scraper FFG

- Save this result as a data object
- join this result with the results calculated above and print it to the console.

```
## Joining with 'by = join_by(site_code)'
```

```
## # A tibble: 1 x 4
##   site_code p_rich t_rich scrap_rich
##   <chr>      <int> <int>      <int>
## 1 A75CC          7     4          7
```

Problem 2

Single-site Metrics with individuals

For all sub-problems in problem 2, you will only be using the `site1` data object.

Problem 2.1

- Calculate the percent of individuals which are in the Chironomidae Family
- Save this result as a data object and print it to the console.

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

```
## # A tibble: 1 x 2
## # Groups:   site_code [1]
##   site_code c_prop
##   <chr>      <dbl>
## 1 A75CC      0.0827
```

Problem 2.2

- Calculate the percent of individuals which are in the Scraper FFG
- Save this result as a data object.
- join this result with the result calculated in 2.1 above and print it to the console.

```
## 'summarise()' has grouped output by 'site_code'. You can override using the
## '.groups' argument.
## Joining with 'by = join_by(site_code)'
```

```
## # A tibble: 1 x 3
## # Groups:   site_code [1]
##   site_code c_prop sc_prop
##   <chr>      <dbl> <dbl>
## 1 A75CC      0.0827  0.260
```

Problem 3

Multiple-site Metrics

All sub-problems in problem 3 will work with the `site1` and `site2` data objects which have been combined using `bind_rows()`.

Problem 3.1

- Calculate the EPT richness for your two sites
- Save this result as a new data object and print it to the screen

```
## # A tibble: 2 x 2
##   site_code EPT_rich
##   <chr>      <int>
## 1 A72         6
## 2 A75CC      19
```

Problem 3.2

- Calculate the per cent of individuals which are in the shredder FFG.
- Save this result as a new data object
- Join it to your result from 3.1 and print it to the screen

```
## 'summarise()' has grouped output by 'site_code'. You can override using the
## '.groups' argument.
## Joining with 'by = join_by(site_code)'
```

```
## # A tibble: 2 x 3
##   site_code EPT_rich shred_percent_ind
##   <chr>      <int>          <dbl>
## 1 A72         6          0.121
## 2 A75CC      19          0.111
```

Problem 3.3

- Calculate the Shannon Diversity for these two sites.
- Save this result as a new data object
- Join it to your previous results from problem 3 and print it to the screen

```
## Joining with 'by = join_by(site_code)'
```

```
## # A tibble: 2 x 4
##   site_code EPT_rich shred_percent_ind      H
##   <chr>      <int>          <dbl> <dbl>
## 1 A72         6          0.121 0.525
## 2 A75CC      19          0.111 2.60
```

Problem 3.4

- Calculate the percent of individuals which are in the Chironomidae Family
- Save this result as a new data object
- Join it to your previous results from problem 3 and print it to the screen

```
## 'summarise()' has grouped output by 'site_code'. You can override using the
## '.groups' argument.
## Joining with 'by = join_by(site_code)'
```

```
## # A tibble: 2 x 5
##   site_code EPT_rich shred_percent_ind   H c_percent_ind
##   <chr>      <int>          <dbl> <dbl>          <dbl>
## 1 A72        6          0.121 0.525          0.0182
## 2 A75CC     19          0.111 2.60           0.0678
```

Problem 3.5 Challenge (optional) This sub-problem is optional. Problem number 4 is not optional

- Calculate the percent of individuals which are in the EPT orders but remove all individuals in the Baetidae Family
- Save this result as a new data object
- Join it to your previous results from problem 3 and print it to the screen

```
## 'summarise()' has grouped output by 'site_code'. You can override using the
## '.groups' argument.
## Joining with 'by = join_by(site_code)'
```

```
## # A tibble: 2 x 6
##   site_code EPT_rich shred_percent_ind   H c_percent_ind ept_no_b_percent_ind
##   <chr>      <int>          <dbl> <dbl>          <dbl>          <dbl>
## 1 A72        6          0.121 0.525          0.0182          0.157
## 2 A75CC     19          0.111 2.60           0.0678          0.526
```

Problem 4

Plots

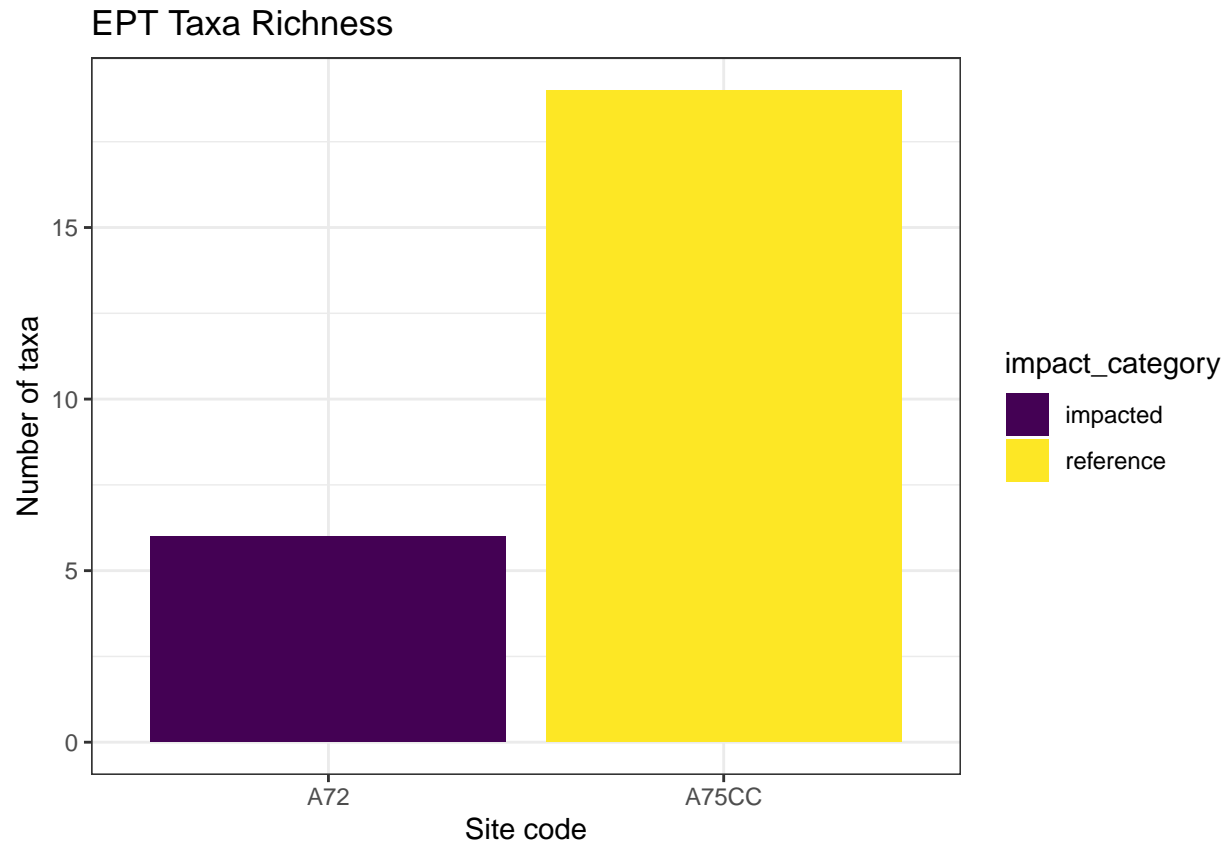
- All sub-problems in problem 4 will be working with the combined results from problem 3. i.e., the results from the two sites.
- All plots made in part 4 need to have the following:
 - A short, informative title.
 - x- and y-axis labels
 - Fill colors need to be based on the impact category variable.
 - The color palettes that you use are up to you

- Before you can plot, you will need to join the `site_info` to your results using the `left_join()` function.

```
## Joining with 'by = join_by(site_code)'
```

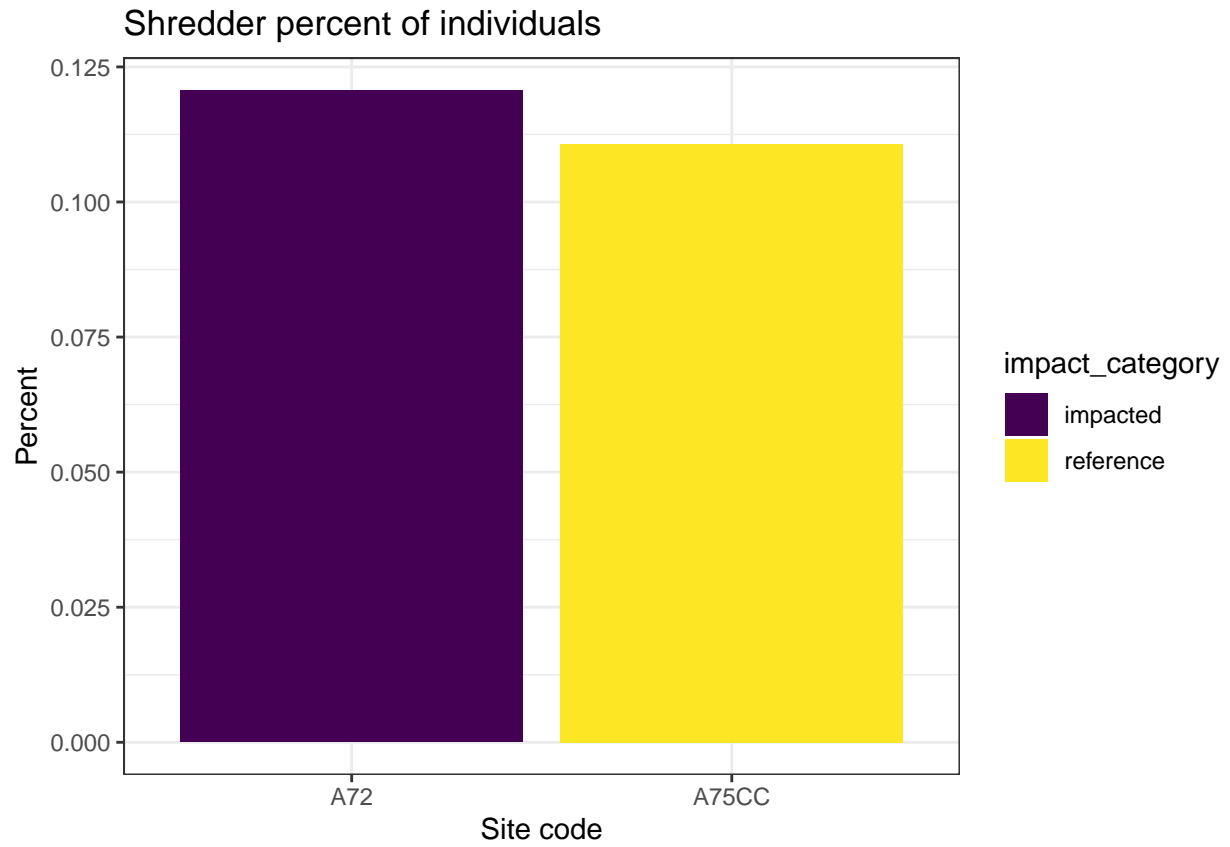
Problem 4.1

- Make a bar plot of the EPT richness results



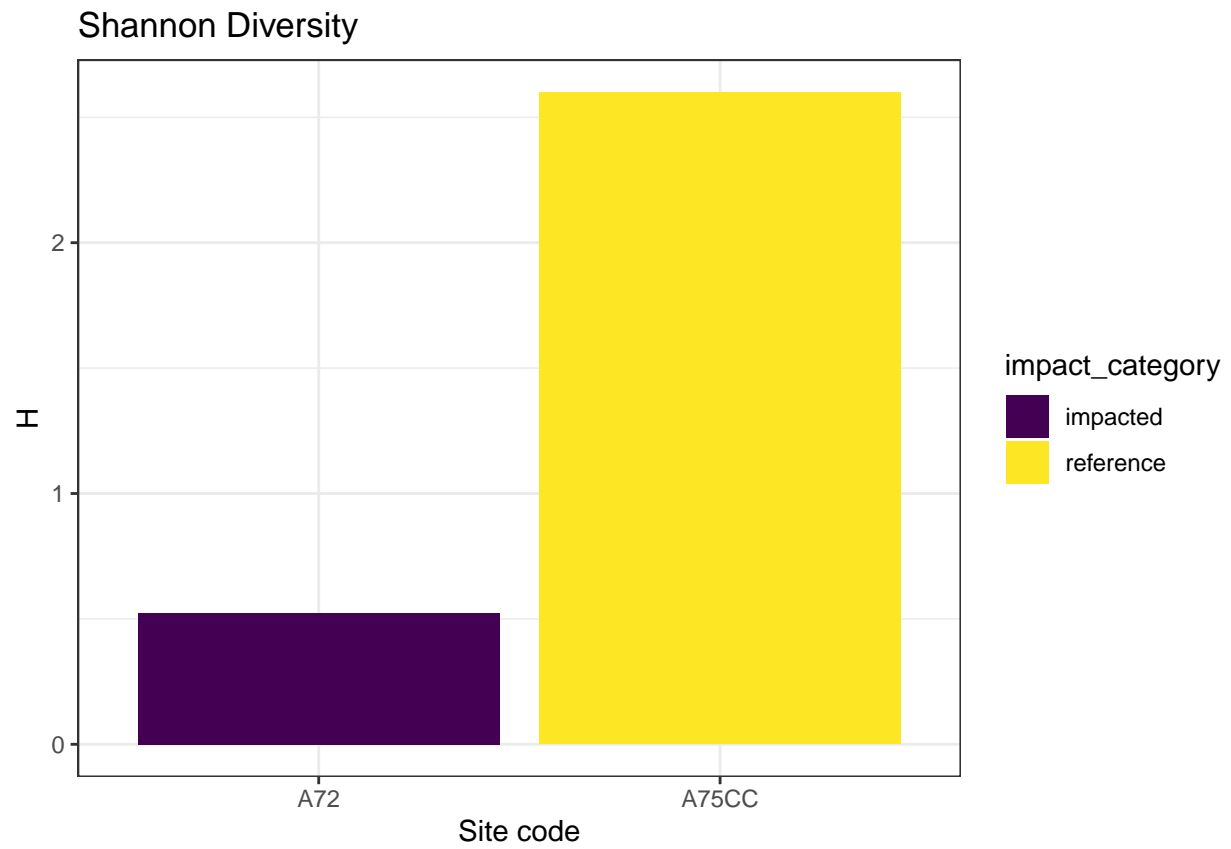
Problem 4.2

- Make a bar plot of the percent of individuals in the shredder FFG



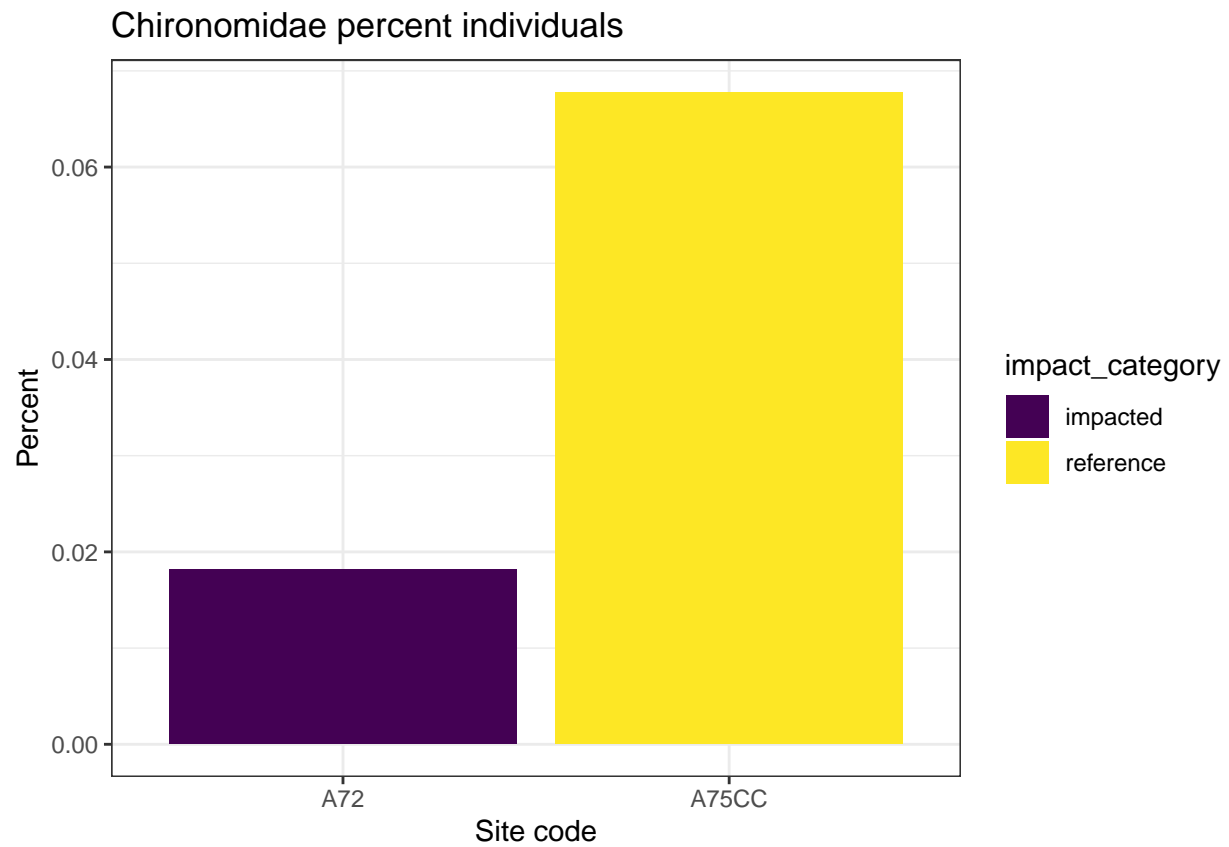
Problem 4.3

- Make a bar plot of the Shannon Diversity results



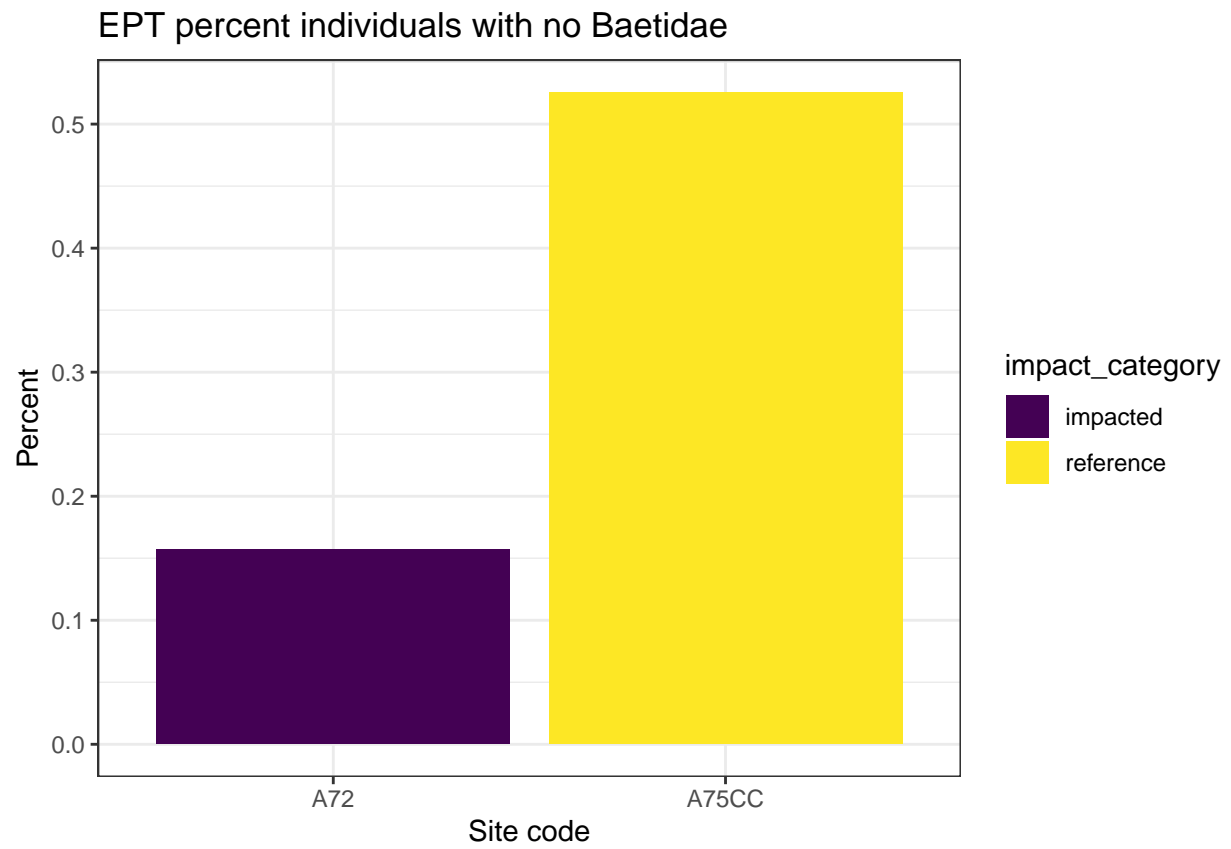
Problem 4.4

- Make a bar plot of the percent of individuals in the Chironomidae Family results



Problem 4.5 (Optional, if you did problem 3.5)

- Make a bar plot of the percent of individuals in EPT orders without individuals which are in Baetidae results



This concludes this R-assignment