

# Package ‘elfgen’

November 28, 2025

**Title** Ecological Limit Function Model Generation and Analysis Toolkit

**Version** 2.3.4

**Maintainer** Connor Brogan <connor.brogan@deq.virginia.gov>

**Description** A toolset for generating Ecological Limit Function (ELF) models and evaluating potential species loss resulting from flow change, based on the ‘elfgen’ framework. ELFs describe the relation between aquatic species richness (fish or benthic macroinvertebrates) and stream size characteristics (streamflow or drainage area). Journal publications are available outlining framework methodology (Kleiner et al. (2020) <[doi:10.1111/1752-1688.12876](https://doi.org/10.1111/1752-1688.12876)>) and application (Rapp et al. (2020) <[doi:10.1111/1752-1688.12877](https://doi.org/10.1111/1752-1688.12877)>).

**Imports** utils, stringr, quantreg, stats, ggplot2, testit, scales,  
sqldf, curl, sbtools, nhdplusTools

**License** MIT + file LICENSE

**URL** <https://github.com/HARPgroup/elfgen>

**RoxygenNote** 7.3.2

**Suggests** testthat (>= 2.1.0)

**NeedsCompilation** no

**Author** Joseph Kleiner [aut] (ORCID: <<https://orcid.org/0000-0003-4837-7678>>),  
Robert Burgholzer [ctb] (ORCID:  
<<https://orcid.org/0000-0002-7290-4928>>),  
Connor Brogan [cre] (ORCID: <<https://orcid.org/0000-0001-9377-1805>>)

**Repository** CRAN

**Date/Publication** 2025-11-28 09:50:12 UTC

## Contents

bkpt_pwit . . . . .	2
bkpt_ymax . . . . .	3
clean_vahydro . . . . .	3
elfchange . . . . .	4
elfdata . . . . .	5
elfdata_vahydro . . . . .	6
elfgen . . . . .	7
richness_change . . . . .	8

<b>Index</b>	<b>10</b>
--------------	-----------

---

<b>bkpt_pwit</b>	<i>Identify breakpoint location with PWIT</i>
------------------	---

---

## Description

This applies the Piecewise Iterative elfgen method. This approach uses an iterative algorithm to identify shifts in the relation between maximum richness and stream size. A user specifies a "quantile" for isolating an upper subset of the data. A user also identifies a bounding range between two x-values ("blo" = "bound low", "bhi" = "bound high") in which the upper subset of data is believed to contain a breakpoint. (Note: Small datasets may not contain a breakpoint)

## Usage

```
bkpt_pwit(watershed.df, quantile, blo, bhi)
```

## Arguments

watershed.df	A datafram of sites with ecological and hydrologic data
quantile	Specified value for the quantile of interest - 0.95 refers to the 95th percentile
blo	A "bound low" value, or the lower bound of the piecewise range
bhi	A "bound high" value, or the upper bound of the piecewise range

## Details

See: Lemoine, N. 2012. "R for Ecologists: Putting Together a Piecewise Regression." <https://www.r-bloggers.com/r-for-ecologists-putting-together-a-piecewise-regression/> The R Book, Second Edition. Michael J. Crawley. 2013 John Wiley & Sons, Ltd. Published 2013 by John Wiley & Sons, Ltd.

## Value

Breakpoint value is returned

## Examples

```
# We don't run this example by R CMD check, because it takes >10s

watershed.df <- elfdatal(watershed.code = '0208020104',
  ichthy.localpath = tempdir(), use_cache = FALSE)
bkpt_pwit(watershed.df, 0.85, 100, 300)
```

---

bkpt_ymax	<i>Identify breakpoint location with Ymax</i>
-----------	---

---

### Description

This applies the Ymax elfgen method. This approach treats the maximum observed species richness value as the breakpoint. This function begins by locating the point with the highest y-value in the full dataset, then utilizing the associated x-value as the breakpoint.

### Usage

```
bkpt_ymax(watershed.df)
```

### Arguments

watershed.df A dataframe of sites with ecological and hydrologic data

### Value

Breakpoint value is returned

### Examples

```
# We don't run this example by R CMD check, because it takes >10s

watershed.df <- elfdatal(watershed.code = '0208020104',
  ichthy.localpath = tempdir(), use_cache = FALSE)
bkpt_ymax(watershed.df)
```

---

clean_vahydro	<i>Clean dataset of ecological and hydrologic data</i>
---------------	--

---

### Description

Given a dataframe of flow metric and richness metric data (Typically retrieved from the DEQ VAHydro database), removes all sites where the ratio of Drainage Area:Mean Annual Flow is greater than 1000, also aggregates to the maximum richness value at each x-metric value

### Usage

```
clean_vahydro(watershed.df)
```

### Arguments

watershed.df A dataframe of sites with ecological and hydrologic data

**Value**

A cleaned dataframe of sites with ecological and hydrologic data

**Examples**

```
# We don't run this example by R CMD check, because it takes >10s

# Retrieve dataset of interest
watershed.df <- data.frame(
  MAF = c(100, 200, 300, 400, 526, 600, 700, 800, 400, 900, 1000, 100, 100),
  NT.TOTAL.UNIQUE = c(10, 20, 30, 40, 50, 40, 30, 20, 50, 10, 10, 99999, 87),
  watershed.code = "test_testcode",
  hydrocode = c("t1", "t2", "t3", "t4", "t5", "t6", "t7", "t8", "t9", "t10", "t11", "t12", "t13"),
  DA_SQMI = c(110, 220000, 280, 360, 530, 604, 712, 698, 40000, 905, 1087, 98, 87),
  x.metric = c(100, 200, 300, 400, 526, 600, 700, 800, 400, 900, 1000, 100, 100)
)
# Clean the dataset
clean_vahydro(watershed.df)
```

**elfchange***Plot percent richness change for various percent flow reductions***Description**

Calculates and plots percent richness change resulting from streamflow reductions

**Usage**

```
elfchange(stats, yaxis_thresh, xlabel = FALSE, ylabel = FALSE)
```

**Arguments**

<code>stats</code>	A dataframe of ELF statistics
<code>yaxis_thresh</code>	Value used for specifying y-axis max limit
<code>xlabel</code>	Used to overwrite default x-axis label
<code>ylabel</code>	Used to overwrite default y-axis label

**Value**

Plot of percent decreases in richness from flow reductions

## Examples

```
# We don't run this example by R CMD check, because it takes >10s

# Generate plot of percent richness change for various percent flow reductions
watershed.df <- elfdata(watershed.code = '0208020104',
  ichthy.localpath = tempdir(), use_cache = FALSE)
breakpt <- 500
elf <- elfgen(
  "watershed.df" = watershed.df,
  "quantile" = 0.95,
  "breakpt" = breakpt,
  "xlabel" = "Mean Annual Flow (ft3/s)",
  "ylabel" = "Fish Species Richness"
)
elfchange(elf$stats, "yaxis_thresh" = 25)
```

elfdata

*Retrieve and format data for ELF generation*

## Description

Given a HUC code, provides a datafram of all contained nhdplus segments and their individual NT Total and Mean Annual Flow MAF values

## Usage

```
elfdata(
  watershed.code,
  ichthy.localpath,
  use_cache = TRUE,
  update_cache = FALSE
)
```

## Arguments

watershed.code	Hydrologic unit code, either HUC6, HUC8, HUC10, or HUC12 (e.g. HUC10 code '0208020101').
ichthy.localpath	Local file path for storing downloaded ichthy data. Defaults to a temp directory.
use_cache	Should the function look for a file with the same name in the file directory? This allows users to use the same Ichty dataset each time rather than needing to download separately when running multiple analyses
update_cache	Should the file be written out to use for future caching?

## Value

A datafram of nhdplus segments containing species richness data (NT Total values) and mean annual flow (MAF) data.

## Examples

```
# We don't run this example by R CMD check, because it takes >10s

# Retrieve dataset of interest
# You may enter either a 6, 8, 10, or 12-digit HUC code.
# By default the ichthy dataset is downloaded to a temp directory, however this may be overridden by
# supplying a local path of interest using the input parameter 'ichthy.localpath'
watershed_df <- elfdata(watershed.code = '0208020104',
  ichthy.localpath = tempdir(), use_cache = FALSE)
head(watershed_df)
```

### elfdata\_vahydro

*Retrieve data from DEQ VAHydro database and format data for ELF generation. Contact Virginia DEQ Office of Water Supply to request access to the VAHydro database.*

## Description

Given a set of VAHydro input parameters, outputs a dataframe of flow metric and richness metric data for hydrologic unit supplied

## Usage

```
elfdata_vahydro(
  watershed.code,
  watershed.bundle,
  watershed.ftype,
  x.metric,
  y.metric,
  y.sampres,
  datasite,
  EDAS.localpath = tempdir()
)
```

## Arguments

watershed.code	Hydrologic unit code, either HUC6, HUC8, HUC10, or HUC12 (e.g. HUC10 code '0208020101').
watershed.bundle	dH bundle of hydrologic unit
watershed.ftype	dH ftype of hydrologic unit
x.metric	x-metric, i.e. streamflow or drainage area
y.metric	y-metric, most commonly species richness
y.sampres	Sample resolution of y.metric (e.g. 'species')
datasite	VAHydro database URL
EDAS.localpath	Local file path for storing downloaded EDAS data. Defaults to a temp directory.

**Value**

A dataframe of sites containing species richness data (NT Total values) and mean annual flow (MAF) data.

**Examples**

```
# We don't run this example by R CMD check, because it takes >10s

# Retrieve dataset of interest
watershed.df <- elfdatal_vahydro(
  'nhd_huc8_02080201',
  'watershed',
  'nhd_huc8',
  'nhdp_drainage_sqmi',
  'aqbio_nt_total',
  'species'
)
elfdata_vahydro(watershed.df)
```

elfgen

*Generate Ecological Limit Function (ELF)***Description**

Generate ELF models by supplying a dataframe of richness and stream size data (streamflow or drainage area), a quantile for evaluating the ecological limit, and a breakpoint threshold.

**Usage**

```
elfgen(
  watershed.df,
  quantile,
  breakpt,
  yaxis_thresh,
  xlabel = FALSE,
  ylabel = FALSE,
  plot_title = FALSE,
  break_var = "x_var"
)
```

**Arguments**

watershed.df	A dataframe of sites with ecological and hydrologic data
quantile	A specified value for the quantile of interest - 0.95 equals the 95th percentile
breakpt	A breakpoint - either user-supplied fixed value or derived using elfgen breakpoint functions bkpt_pwit() or bkpt_ymax

<code>yaxis_thresh</code>	Value used for specifying y-axis max limit
<code>xlabel</code>	Used to overwrite default x-axis label
<code>ylabel</code>	Used to overwrite default y-axis label
<code>plot_title</code>	A plot title used to title the ggplot. If left as the default (FALSE), the plot will be titled using the name of the HUC from nhdPlus
<code>break_var</code>	The name of the variable in watershed.df to be filtered for by the breakpt, defaults to the expected name "x_var"

**Value**

Object containing plot image and dataframe of ELF statistics

**Examples**

```
# We don't run this example by R CMD check, because it takes >10s

watershed.df <- elfdata(watershed.code = '0208020104',
  ichthy.localpath = tempdir(), use_cache = FALSE)
breakpt <- 500
elfgen(
  "watershed.df" = watershed.df,
  "quantile" = 0.80,
  "breakpt" = breakpt,
  "xlabel" = "Mean Annual Flow (ft3/s)",
  "ylabel" = "Fish Species Richness"
)
```

**richness\_change**      *Calculate change in richness resulting from a percent reduction in flow*

**Description**

Calculates absolute or percent richness change from streamflow reduction

**Usage**

```
richness_change(stats, pctchg, xval = FALSE)
```

**Arguments**

<code>stats</code>	A dataframe of ELF statistics
<code>pctchg</code>	Decrease in flow as a percent (e.g. 10 equals 10 percent reduction in flow).
<code>xval</code>	x-axis value for assessing percent change in richness. When supplied, the function will calculate percent change in richness at a specific stream size (e.g. 50 equals a stream size with mean annual flow of 50 cfs).

**Value**

Richness change value is returned

**Examples**

```
# We don't run this example by R CMD check, because it takes >10s

watershed.df <- elfdata(watershed.code = '0208020104',
  ichthy.localpath = tempdir(), use_cache = FALSE)
breakpt <- 500
elf <- elfgen(
  "watershed.df" = watershed.df,
  "quantile" = 0.95,
  "breakpt" = breakpt,
  " xlabel" = "Mean Annual Flow (ft3/s)",
  " ylabel" = "Fish Species Richness"
)
# Calculate absolute richness change
richness_change(elf$stats, "pctchg" = 10)
# Calculate percent richness change at a specific stream size
richness_change(elf$stats, "pctchg" = 10, "xval" = 50)
```

# Index

`bkpt_pwit`, [2](#)  
`bkpt_ymax`, [3](#)  
`clean_vahydro`, [3](#)  
`elfchange`, [4](#)  
`elfdata`, [5](#)  
`elfdata_vahydro`, [6](#)  
`elfgen`, [7](#)  
`richness_change`, [8](#)