

Change Analysis

Thomas Kincaid

April 4, 2019

Contents

1 Introduction	1
2 Preliminaries	2
3 Load the survey design and analytical variables data set	2
4 Change analysis	5

1 Introduction

This document presents change analysis of a GRTS survey design. The resource employed in the analysis is rivers and streams in the 48 contiguous United States. Data was obtained from two surveys conducted by the U.S. Environmental Protection Agency: (1) the Wadeable Streams Assessment (WSA) in 2004 ([U.S. Environmental Protection Agency 2006](#)) and (2) the National Rivers and Streams Survey (NRSA) in 2008 and 2009 ([U.S. Environmental Protection Agency 2016](#)). Change analysis measures the difference between response variables that were estimated in two surveys. Both continuous and categorical response variables can be employed for change analysis. For a categorical response variable, change is estimated by the difference in category estimates for the two surveys, where a category estimate is the estimated proportion of values in a category. Note that a separate change estimate is calculated for each category of a categorical response variable. For a continuous response variable, change can be estimated for the mean, the median, or for both the mean and median. For a continuous response variable using the mean, change is estimated by the difference in estimated mean values for the two surveys. For change estimates using the median, the first step is to calculate an estimate of the median for the first survey. The estimated median from the first survey is then used to define two categories: (1) values that are less than or equal to the estimated median and (2) values that are greater than the estimated median. Once

the categories are defined, change analysis for the median is identical to change analysis for a categorical variable, i.e., change is estimated by the difference in category estimates for the two surveys.

2 Preliminaries

The initial step is to use the library function to load the spsurvey package. After the package is loaded, a message is printed to the R console indicating that the spsurvey package was loaded successfully.

Load the spsurvey package

```
> # Load the spsurvey package
> library(spsurvey)
>
```

Version 3.5.0 of the spsurvey package was loaded successfully.

3 Load the survey design and analytical variables data set

The original data file contains more than 2,400 records for change estimation. To produce a more manageable number of records, rivers and streams located in the Western Mountains Level III Ecoregion ([Omernik 1987](#)) were retained in the data that will be analyzed, which produced a data set containing 668 records.

The next step is to load the data set, which includes both survey design variables and analytical variables. The data function is used to load the data set and assign it to a data frame named NRSA_2009. The nrow function is used to determine the number of rows in the NRSA_2009 data frame, and the resulting value is assigned to an object named nr. Finally, the initial six lines and the final six lines in the NRSA_2009 data frame are printed using the head and tail functions, respectively.

Load the survey design and analytical variables data set

```
> # Load the data set and determine the number of rows in the data frame
> data(NRSA_2009)
> nr <- nrow(NRSA_2009)
>
```

Display the initial six lines in the data file.

```
> # Display the initial six lines in the data file
> head(NRSA_2009)
```

	siteID	xcoord	ycoord	wgt	Survey	Revisit_Site	Stream_Size	NTL
1	WAZP99-0833	-1369578	1345072	17.17660	WSA	Y	Small	119
2	WAZP99-0545	-1221990	1291197	17.17660	WSA	Y	Small	324
3	WAZP99-0687	-1231441	1275327	13.70593	WSA	Y	Small	55
4	WCAP99-0991	-2038942	2002946	572.18276	WSA	Y	Small	55
5	WCAP99-0587	-2264511	2285650	100.41211	WSA	Y	Small	509
6	WCAP99-0503	-2035060	1880456	457.74621	WSA	Y	Small	63
	PTL	Benthic_MMI	NTL_Cond	PTL_Cond	Benthic_MMI_Cond			
1	8	46.78965	Good	Good		Fair		
2	92	27.15364	Poor	Poor		Poor		
3	39	19.12938	Good	Fair		Poor		
4	2	61.76663	Good	Good		Good		
5	27	49.32376	Poor	Fair		Fair		
6	1	68.57991	Good	Good		Good		

```
>
```

Display the final six lines in the data file.

```
> # Display the final six lines in the data file
> tail(NRSA_2009)
```

	siteID	xcoord	ycoord	wgt	Survey	Revisit_Site	Stream_Size	NTL
663	FW08WY041	-868099.9	2401874	1543.3105	NRSA	N	Small	452
664	FW08WY042	-688555.9	2408900	171.6259	NRSA	N	Large	343
665	FW08WY081	-1113106.7	2377400	4876.7431	NRSA	N	Small	301
666	FW08WY085	-1079717.5	2471045	936.4642	NRSA	N	Small	34
667	FW08WY089	-900074.7	2444794	1543.3105	NRSA	N	Small	124
668	FW08WY092	-1181999.6	2221639	1543.3105	NRSA	N	Small	540
	PTL	Benthic_MMI	NTL_Cond	PTL_Cond	Benthic_MMI_Cond			
663	48.4950	28.72	Poor	Poor		Poor		
664	13.3113	4.90	Poor	Good		Poor		
665	69.6469	38.50	Poor	Poor		Fair		
666	58.7781	44.43	Good	Poor		Fair		
667	45.3413	17.83	Good	Poor		Poor		
668	191.2206	10.92	Poor	Poor		Poor		

```
>
```

The location of rivers and streams that were sampled in the Western Mountains Ecoregion is displayed in Figure 1. The sample sites are displayed using a unique color for each survey.

Plot of WSA and NRSA Sample Sites Color-Coded by Survey

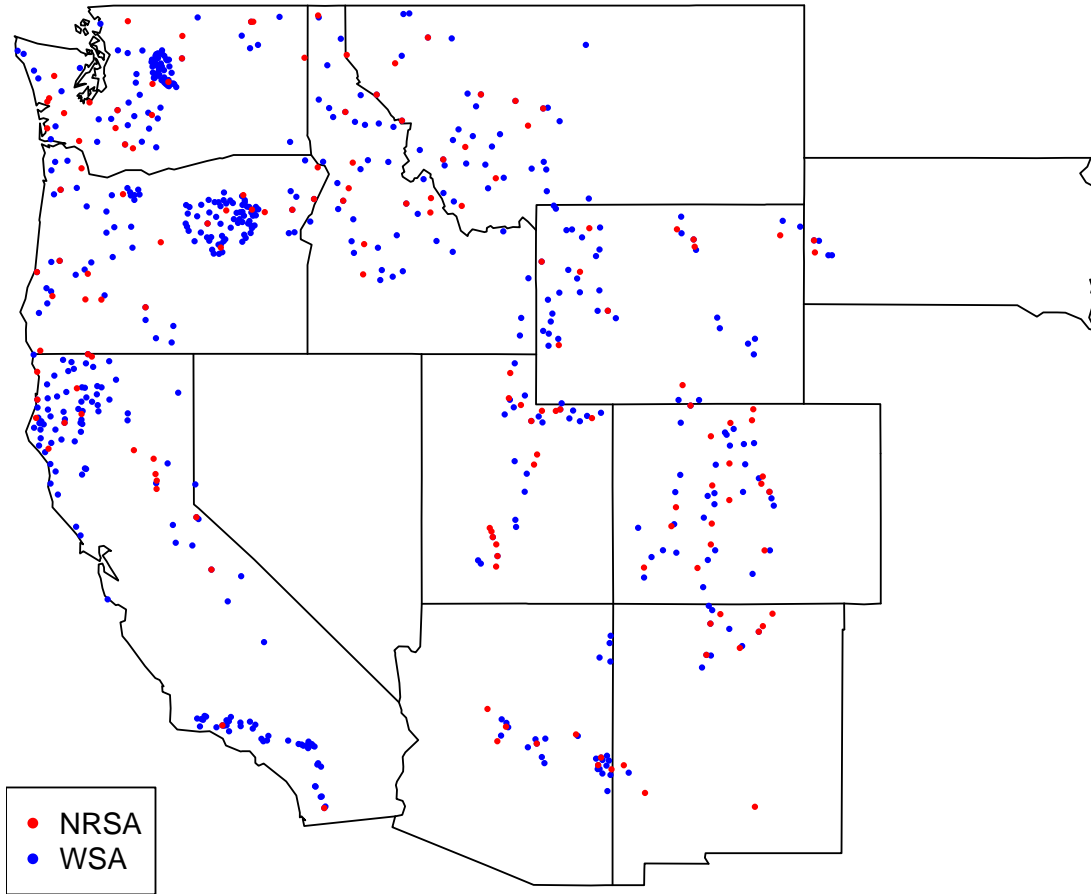


Figure 1: Location of rivers and streams that were sampled in the Western Mountains Ecoregion by the U.S. Environmental Protection Agency during the Wadeable Streams Assessment (WSA) conducted in 2004 and the National Rivers and Streams Assessment (NRSA) conducted in 2008-2009.

4 Change analysis

Change analysis will be investigated by examining three continuous response variables and three categorical response variables. The continuous response variables are total phosphorus concentration, total nitrogen concentration, and benthic macroinvertebrate multimetric index (MMI) score. The categorical response variables are condition class variables for each of the continuous response variables. Condition classes are created by grouping values for a continuous response variable into categories that reflect the impact of a response value on the overall ecological condition of a site. Categories used for condition classes are: "Good", "Fair" and "Poor".

The `change.analysis` function will be used to calculate change estimates. Six data frames constitute the primary input to the `change.analysis` function. The site ID provides the unique identifier for each sample site and is used to connect records among the data frames. The `siteID` variable in the `NRSA_2009` data frame is assigned to the `siteID` variable (or variables in one case) in the data frames. The six data frames that will be created are named as follows: `sites`, `repeats`, `subpop`, `design`, `data.cat`, and `data.cont`. In order to obtain change estimates for the continuous variables using both the mean and median, the `test` argument of the `change.analysis` function is assigned the following value: `c("mean", "median")`. Note that the default value for the `test` argument is "mean".

The `sites` data frame identifies sites to use in the analysis and contains three variables: (1) `siteID` - site ID values, (2) `Survey1` - a logical vector identifying sites for survey one, and (3) `Survey2` - a logical vector identifying sites for survey two. The `Survey1` variable is created by assigning the value `TRUE` to every site for which the `Survey` variable in the `NRSA_2009` data frame equals the value "WSA". Similarly, the `Survey2` variable is created by assigning the value `TRUE` to every site for which the `Survey` variable in the `NRSA_2009` data frame equals the value "NRSA".

The `repeats` data frame identifies repeat visit sites and contains two variables: (1) `siteID_1` - the site ID value for survey one and (2) `siteID_2` - the site ID value for survey two. The `siteID_1` variable is created by selecting values of the `siteID` variable in the `NRSA_2009` data frame for which the `Survey` variable in the `NRSA_2009` data frame equals the value "WSA" and the `Revisit_Site` variable in the `NRSA_2009` data frame equals "Y". The `siteID_2` variable is created using an analogous process. For each row of the `repeats` data frame, the two site ID values must correspond to the same site. Note that the `NRSA_2009` data frame has been organized so that repeat visit sites for WSA occur in the same order as repeat visit sites for NRSA.

The `subpop` data frame defines populations and, optionally, subpopulations for which estimates are desired. Unlike the `sites` and `design` data frames, the `subpop` data frame can contain an arbitrary number of columns. The first variable in the `subpop` data frame identifies site ID values and each subsequent variable identifies a type of population, where the variable name is used to identify type. A type variable identifies each site with a character value. If the number of unique values for a type variable is greater than one, then the set of values represent subpopulations of that type. When a type variable consists of a single unique

value, then the type does not contain subpopulations. For this analysis, the subpop data frame contains three variables: (1) siteID - site ID values, (2) Western_Mountains - which will be used to calculate estimates for all of the sample sites combined, and (3) Stream_Size - which will be used to calculate estimates for each of the two classes of stream size (large and small). The rep (repeat) function is used to assign values to the Western_Mountains variable in the subpop data frame, and the Stream_Size variable in the NRSA_2009 data frame is assigned to the Stream_Size variable in the subpop data frame. Recall that nr, which is included in the call to the rep function, is an object containing the number of rows in the NRSA_2009 data frame.

The design data frame consists of survey design variables. For the analysis under consideration, the design data frame contains the following variables: (1) siteID - site ID values; (2) wgt - survey design weights; (3) xcoord - x-coordinates for location; and (4) ycoord - y-coordinates for location. The wgt, xcoord, and ycoord variables in the design data frame are assigned values using variables with the same names in the NRSA_2009 data frame.

The final two data frames, data.cat and data.cont, provide values of categorical response variables and continuous response variables, respectively. Like the subpop data frame, the data.cat and data.cont data frames can contain an arbitrary number of columns. The first variable in the data.cat data frame identifies site ID values and each subsequent variable identifies a categorical response response variable. For this analysis, the categorical response variables are Nitrogen_Condition, Phosphorus_Condition, and Benthic_MMI_Condition, which are assigned, respectively, variables NTL_cond, PTL_cond, and Benthic_MMI_cond in the NRSA_2009 data frame. For benthic MMI score, there are four sites (three in WSA and one in NRSA) for which the MMI score could not be calculated. Those sites are coded as NA for the Benthic_MMI variable and as category "Not Assessed" for the Benthic_MMI_cond variable.

The data.cont data frame is organized analogous to the data.cat data frame. The first variable in the data frame identifies site ID values and each subsequent variable identifies a continuous response response variable. For this analysis, the continuous response variables are Log_Total_Nitrogen, Log_Total_Phosphorus, and Benthic_MMI, which are assigned, respectively, variables NTL, PTL, and Benthic_MMI in the NRSA_2009 data frame. Note that total nitrogen and total phosphorus are analyzed using the base ten log scale, which are created by use of the log10 function.

Create the sites data frame.

```
> sites <- data.frame(siteID=NRSA_2009$siteID,  
+                     Survey1=NRSA_2009$Survey == "WSA",  
+                     Survey2=NRSA_2009$Survey == "NRSA")
```

Create the repeats data frame.

```
> repeats <- data.frame(siteID_1=NRSA_2009$siteID[NRSA_2009$Survey == "WSA" &  
+                     NRSA_2009$Revisit_Site == "Y"],
```

```
+           siteID_2=NRSA_2009$siteID[NRSA_2009$Survey == "NRSA" &
+           NRSA_2009$Revisit_Site == "Y"])
```

Create the subpop data frame.

```
> subpop <- data.frame(siteID=NRSA_2009$siteID,
+           Western_Mountains=rep("Western Mountains", nr),
+           Stream_Size=NRSA_2009$Stream_Size)
```

Create the design data frame.

```
> design <- data.frame(siteID=NRSA_2009$siteID,
+           wgt=NRSA_2009$wgt,
+           xcoord=NRSA_2009$xcoord,
+           ycoord=NRSA_2009$ycoord)
```

Create the data.cat data frame.

```
> data.cat <- data.frame(siteID=NRSA_2009$siteID,
+           Nitrogen_Condition=NRSA_2009$NTL_Cond,
+           Phosphorus_Condition=NRSA_2009$PTL_Cond,
+           Benthic_MMI_Condition=NRSA_2009$Benthic_MMI_Cond)
```

Create the data.cont data frame.

```
> data.cont <- data.frame(siteID=NRSA_2009$siteID,
+           Log_Total_Phosphorus=log10(NRSA_2009$PTL+1),
+           Log_Total_Nitrogen=log10(NRSA_2009$NTL+1),
+           Benthic_MMI=NRSA_2009$Benthic_MMI)
```

Calculate change estimates.

```
> Change_Estimates <- change.analysis(sites, repeats, subpop, design, data.cat,
+   data.cont, test=c("mean", "median"))
```

During execution of the program, 37 warning messages were generated. The warning messages are stored in a data frame named 'warn.df'. Enter the following command to view the warning messages: warnprnt()
To view a subset of the warning messages (say, messages number 1, 3, and 5), enter the following command: warnprnt(m=c(1,3,5))

Like other functions in the spsurvey package, the change.analysis function generates warning messages when certain situations are encountered in the data. When warning messages are

generated, the functions print a message to the R console window stating the number of warning messages and explaining the procedure for recovering the messages. The call to the `change.analysis` function generated thirty-seven warning messages. These messages fall into two categories: (1) cases where the number of repeat visit sites was less than two and (2) cases where a category level was not present among the repeat visit sites in one of the surveys. For both cases, covariance among the revisited sites was not included in calculation of the standard error estimate. The `warnprnt` function is used to display two of the warning messages.

Display warning messages 1 and 3.

```
> warnprnt(m = c(1, 3))
```

Warning Message 1

Function: `change.est`

Population Type: `Stream_Size`

Subpopulation: `Large`

Indicator: `Nitrogen_Condition`

Warning: The number of nonmissing repeat visit sites was less than two in one of the surveys.

Action: Covariance among the revisited sites was not included in calculation of the standard error estimate.

Warning Message 3

Function: `changevar.prop`

Population Type: `Western_Mountains`

Subpopulation: `Western Mountains`

Indicator: `Benthic_MMI_Condition`

Warning: Category level "Not Assessed" was not present among the repeat visit sites in one of the surveys.

Action: Covariance among the repeat visit sites was not included in calculation of the standard error estimate.

The change estimates are displayed using the `print` function. For categorical response variables and continuous response variables using the median, change estimates are printed for the complete set of sites only. For continuous response variables using the mean, all change estimates are printed. The object produced by `change.analysis` is a list composed of three data frames. The first data frame, named `catsum`, contains estimates for categorical response variables. The second data frame, named `contsum_mean`, contains estimates for continuous response variables using the mean. The third data frame, named `contsum_median`, contains estimates for continuous response variables using the median. The `catsum` and `contsum_median` data frames will be described first. The initial four columns in those data frames identify the population (Type), subpopulation (Subpopulation), response variable (Indicator), and category of the response variable (Category). The next four columns provide results for change estimates using the percent scale: the change estimate (`DiffEst.P`),

standard error of the estimate (StdError.P), lower confidence bound (LCB95Pct.P), and upper confidence bound (UCB95Pct.P). Argument `conf` for `change.analysis` allows control of the confidence bound level. The default value for `conf` is 95, hence the column names for confidence bounds contain the value 95. Supplying a different value to the `conf` argument will be reflected in the confidence bound names. Confidence bounds are obtained using the standard error and the Normal distribution multiplier corresponding to the confidence level. The next four columns provide results for change estimates using the size (units scale): the change estimate (DiffEst.U), standard error of the estimate (StdError.U), lower confidence bound (LCB95Pct.U), and upper confidence bound (UCB95Pct.U). For this data, the units are kilometers of stream length. The next nine columns provide estimates for survey one: the first column is the number of response values for a category (NResp); the next four columns contain survey one estimates, standard errors, and confidence bounds in the percent scale; and the final four columns contain survey one estimates, standard errors, and confidence bounds in the units scale. The final nine columns of the `catsum` data frame provide estimates for survey two using the format described for survey one.

Description of the `contsum_mean` data frame follows. The initial four columns in that data frame identify the population (Type), subpopulation (Subpopulation), response variable (Indicator), and statistic employed for the change estimate (Statistic). The Statistic column contains the value "Mean" as a reminder that change estimates for continuous response variable use the mean. The next four columns provide results for the change estimates: the change estimate (DiffEst), standard error of the estimate (StdError), lower confidence bound (LCB95Pct), and upper confidence bound (UCB95Pct). The next five columns provide estimates for survey one: the first column is the number of response values for a category (NResp); the next four columns contain survey one estimates, standard errors, and confidence bounds. The final five columns of the `contsum_mean` data frame provide estimates for survey two using the format described for survey one.

```
> # Print Western Mountains change estimates for categorical variables
> print(subset(Change_Estimates$catsum, Type == "Western_Mountains"))
```

	Type	Subpopulation	Indicator	Category						
1	Western_Mountains	Western Mountains	Nitrogen_Condition	Fair						
2	Western_Mountains	Western Mountains	Nitrogen_Condition	Good						
3	Western_Mountains	Western Mountains	Nitrogen_Condition	Poor						
10	Western_Mountains	Western Mountains	Phosphorus_Condition	Fair						
11	Western_Mountains	Western Mountains	Phosphorus_Condition	Good						
12	Western_Mountains	Western Mountains	Phosphorus_Condition	Poor						
19	Western_Mountains	Western Mountains	Benthic_MMI_Condition	Fair						
20	Western_Mountains	Western Mountains	Benthic_MMI_Condition	Good						
21	Western_Mountains	Western Mountains	Benthic_MMI_Condition	Not Assessed						
22	Western_Mountains	Western Mountains	Benthic_MMI_Condition	Poor						
					DiffEst.P	StdError.P	LCB95Pct.P	UCB95Pct.P	DiffEst.U	StdError.U
1					-8.5432080	3.810749	-16.012138	-1.074278	-17946.688	8395.978
2					8.0514935	4.619531	-1.002621	17.105608	21684.775	16484.007

3	0.4917145	3.251915	-5.881922	6.865351	2026.684	7167.314
10	-8.0018563	3.842896	-15.533795	-0.469918	-16614.900	8735.346
11	-12.0570460	5.499675	-22.836211	-1.277881	-23754.653	14805.495
12	20.0589023	5.143497	9.977833	30.139972	46134.325	12423.393
19	0.4641174	5.209334	-9.745990	10.674225	2633.944	13136.374
20	3.1916027	5.386772	-7.366276	13.749482	10016.622	13816.453
21	-0.6911747	1.273905	-3.187982	1.805633	-1470.634	2840.759
22	-2.9645454	3.397343	-9.623216	3.694125	-5415.160	7693.388
	LCB95Pct.U	UCB95Pct.U	NResp_1	Estimate.P_1	StdError.P_1	LCB95Pct.P_1
1	-34402.502	-1490.8736	118	23.563946	2.610170	18.44811
2	-10623.285	53992.8360	303	60.554135	2.706990	55.24853
3	-12020.993	16074.3613	101	15.881919	1.646566	12.65471
10	-33735.863	506.0627	119	25.445973	2.796803	19.96434
11	-52772.889	5263.5834	315	60.551675	2.842622	54.98024
12	21784.922	70483.7282	88	14.002353	1.934169	10.21145
19	-23112.877	28380.7644	112	27.497661	2.781613	22.04580
20	-17063.127	37096.3719	262	48.649879	2.801831	43.15839
21	-7038.419	4097.1511	3	1.582266	1.012512	0.00000
22	-20493.924	9663.6033	145	22.270194	2.334555	17.69455
	UCB95Pct.P_1	Estimate.U_1	StdError.U_1	LCB95Pct.U_1	UCB95Pct.U_1	NResp_2
1	28.679786	51889.065	5996.625	40135.90	63642.234	30
2	65.859737	133343.431	8129.499	117409.91	149276.957	76
3	19.109130	34972.831	3513.873	28085.77	41859.896	40
10	30.927605	56033.388	6798.708	42708.17	69358.609	34
11	66.123112	133338.013	6904.461	119805.52	146870.509	51
12	17.793254	30833.927	4439.806	22132.07	39535.786	61
19	32.949522	60551.314	6778.412	47265.87	73836.757	29
20	54.141366	107129.625	6320.259	94742.14	119517.106	66
21	3.566754	3484.235	2255.171	0.00	7904.288	1
22	26.845839	49040.154	5477.911	38303.65	59776.661	50
	Estimate.P_2	StdError.P_2	LCB95Pct.P_2	UCB95Pct.P_2	Estimate.U_2	
1	15.0207384	2.9277034	9.282545	20.75893	33942.38	
2	68.6056284	3.9698939	60.824779	76.38648	155028.21	
3	16.3736332	2.9533868	10.585102	22.16216	36999.52	
10	17.4441164	2.7560875	12.042284	22.84595	39418.49	
11	48.4946287	5.0086922	38.677772	58.31148	109583.36	
12	34.0612550	4.9066246	24.444447	43.67806	76968.25	
19	27.9617781	4.6008866	18.944206	36.97935	63185.26	
20	51.8414817	4.7868394	42.459449	61.22351	117146.25	
21	0.8910915	0.7730795	0.000000	2.40630	2013.60	
22	19.3056487	2.6390704	14.133166	24.47813	43624.99	
	StdError.U_2	LCB95Pct.U_2	UCB95Pct.U_2			
1	6203.537	21783.67	46101.087			
2	15236.562	125165.09	184891.320			
3	6561.709	24138.80	49860.229			

```

10    5753.495    28141.84    50695.131
11    14240.877    81671.75    137494.966
12    11908.016    53628.97    100307.534
19    11717.757    40218.88    86151.639
20    12850.603    91959.53    142332.968
21    1727.460         0.00    5399.359
22    5751.028    32353.19    54896.801

```

```

> # Print change estimates for continuous variables using the mean
> print(Change_Estimates$contsum_mean)

```

	Type	Subpopulation	Indicator	Statistic
1	Western_Mountains	Western Mountains	Log_Total_Phosphorus	Mean
2	Stream_Size	Large	Log_Total_Phosphorus	Mean
3	Stream_Size	Small	Log_Total_Phosphorus	Mean
4	Western_Mountains	Western Mountains	Log_Total_Nitrogen	Mean
5	Stream_Size	Large	Log_Total_Nitrogen	Mean
6	Stream_Size	Small	Log_Total_Nitrogen	Mean
7	Western_Mountains	Western Mountains	Benthic_MMI	Mean
8	Stream_Size	Large	Benthic_MMI	Mean
9	Stream_Size	Small	Benthic_MMI	Mean

	DiffEst	StdError	LCB95Pct	UCB95Pct	NResp_1	Estimate_1
1	0.28048779	0.05511323	0.1724678	0.38850773	522	1.077315
2	0.06992848	0.16844500	-0.2602176	0.40007461	8	1.592550
3	0.28496217	0.05607767	0.1750520	0.39487238	514	1.067896
4	-0.13898630	0.04683131	-0.2307740	-0.04719862	522	2.095254
5	0.18594596	0.15987119	-0.1273958	0.49928775	8	2.291269
6	-0.14385707	0.04793736	-0.2378126	-0.04990158	514	2.091671
7	0.66121946	1.53852011	-2.3542245	3.67666346	519	49.014131
8	-13.22855538	10.14257203	-33.1076313	6.65052052	8	44.325181
9	0.87835322	1.54413067	-2.1480873	3.90479371	511	49.101251

	StdError_1	LCB95Pct_1	UCB95Pct_1	NResp_2	Estimate_2	StdError_2	LCB95Pct_2
1	0.02708917	1.024221	1.130409	146	1.357803	0.05122909	1.257396
2	0.15404633	1.290625	1.894475	24	1.662479	0.06814282	1.528921
3	0.02724145	1.014504	1.121289	122	1.352859	0.05216151	1.250624
4	0.01536527	2.065139	2.125369	146	1.956268	0.04603932	1.866032
5	0.14757094	2.002035	2.580503	24	2.477215	0.06149485	2.356687
6	0.01552844	2.061236	2.122106	122	1.947814	0.04702752	1.855641
7	1.03521685	46.985144	51.043119	145	49.675351	1.25700084	47.211675
8	8.86140780	26.957140	61.693221	24	31.096625	4.93429015	21.425594
9	1.04319033	47.056636	51.145866	121	49.979604	1.25059598	47.528481

	UCB95Pct_2
1	1.458210
2	1.796036

```

3  1.455093
4  2.046503
5  2.597743
6  2.039986
7  52.139027
8  40.767656
9  52.430727

```

```

> # Print change estimates for continuous variables using the median
> print(subset(Change_Estimates$contsum_median, Type == "Western_Mountains"))

```

	Type	Subpopulation	Indicator	Category		
1	Western_Mountains	Western Mountains	Log_Total_Phosphorus	Greater_Than_Median		
2	Western_Mountains	Western Mountains	Log_Total_Phosphorus	Less_Than_Median		
7	Western_Mountains	Western Mountains	Log_Total_Nitrogen	Greater_Than_Median		
8	Western_Mountains	Western Mountains	Log_Total_Nitrogen	Less_Than_Median		
13	Western_Mountains	Western Mountains	Benthic_MMI	Greater_Than_Median		
14	Western_Mountains	Western Mountains	Benthic_MMI	Less_Than_Median		
	DiffEst.P	StdError.P	LCB95Pct.P	UCB95Pct.P	DiffEst.U	StdError.U
1	19.993073	5.329139	9.548152	30.437994	48065.0542	15968.94
2	-19.993073	5.329139	-30.437994	-9.548152	-42300.2824	12852.20
7	-9.172204	5.218530	-19.400334	1.055926	-17837.0098	11977.46
8	9.172204	5.218530	-1.055926	19.400334	23601.7816	16267.11
13	1.921212	5.488400	-8.835855	12.678278	7953.6486	14054.77
14	-1.921212	5.488400	-12.678278	8.835855	-718.2427	14693.78
	LCB95Pct.U	UCB95Pct.U	NResp_1	Estimate.P_1	StdError.P_1	LCB95Pct.P_1
1	16766.500	79363.608	270	50.07461	2.796763	44.59306
2	-67490.127	-17110.438	252	49.92539	2.796763	44.44383
7	-41312.402	5638.382	265	50.12217	2.831757	44.57202
8	-8281.173	55484.736	257	49.87783	2.831757	44.32769
13	-19593.202	35500.499	264	50.45979	2.809366	44.95353
14	-29517.523	28081.038	255	49.54021	2.809366	44.03396
	UCB95Pct.P_1	Estimate.U_1	StdError.U_1	LCB95Pct.U_1	UCB95Pct.U_1	NResp_2
1	55.55617	110267.0	7655.957	95261.56	125272.4	113
2	55.40694	109938.4	6795.463	96619.50	123257.2	33
7	55.67231	110371.7	6862.229	96921.96	123821.4	80
8	55.42798	109833.6	7717.698	94707.24	124960.1	66
13	55.96604	109357.0	6322.352	96965.42	121748.6	67
14	55.04647	107364.1	7836.067	92005.68	122722.5	78
	Estimate.P_2	StdError.P_2	LCB95Pct.P_2	UCB95Pct.P_2	Estimate.U_2	
1	70.06769	4.536281	61.17674	78.95863	158332.02	
2	29.93231	4.536281	21.04137	38.82326	67638.08	
7	40.94996	4.383401	32.35865	49.54127	92534.67	
8	59.05004	4.383401	50.45873	67.64135	133435.43	

13	52.38100	4.714870	43.14002	61.62197	117310.65
14	47.61900	4.714870	38.37803	56.85998	106645.85
	StdError.U_2	LCB95Pct.U_2	UCB95Pct.U_2		
1	14014.046	130864.99	185799.04		
2	10908.743	46257.34	89018.82		
7	9816.791	73294.11	111775.23		
8	14319.779	105369.18	161501.68		
13	12552.471	92708.26	141913.04		
14	12429.933	82283.63	131008.07		

>

The write.csv function is used to store the change estimates as comma-separated value (csv) files. Files in csv format can be read by programs such as Microsoft Excel. The three data frames created by the change.analysis function are stored in separate files.

```
> write.csv(Change_Estimates$catsum, file="Change_Estimates_Categorical.csv", row.names=FALSE)
> write.csv(Change_Estimates$contsum_mean, file="Change_Estimates_Continuous_Mean.csv", row.names=FALSE)
> write.csv(Change_Estimates$contsum_median, file="Change_Estimates_Continuous_Median.csv", row.names=FALSE)
```

References

- Omernik, J. M. (1987). Ecoregions of the conterminous united states. *Annals of the Association of American Geographers* 77, 118–125.
- U.S. Environmental Protection Agency (2006). Wadeable Streams Assessment: A collaborative survey of the nation's streams. Technical report, U.S. Environmental Protection Agency, Office of Research and Development and Office of Water. EPA 841-B-06-002.
- U.S. Environmental Protection Agency (2016). National Rivers and Streams Assessment 2008-2009: A collaborative survey. Technical report, U.S. Environmental Protection Agency, Office of Water and Office of Research and Development. EPA 841-R-16-007.