**South Fork Research, Inc.**

12

**Bio-Energetics Model User’s Guide**

**In support of the ISEMP Monitoring Program**

**Matt Nahorniak, South Fork Research**

Table of Contents

[**1.** **Introduction** 3](#_Toc468800443)

[**2.** **System Requirements** 3](#_Toc468800444)

[**3.** **Files included in the BioEnergetics Model** 4](#_Toc468800445)

[3.1. **BioE.R**: 4](#_Toc468800446)

[3.2. **BioE\_Functions.R**: 4](#_Toc468800447)

[**3.3.** **Input\_BioELookup\_Tables.csv:** 4](#_Toc468800448)

[3.4. **Example Input Files: Input\_BioE\_Header.csv, Temp\_P\_Prey\_Input.csv, and Weights\_Consumption\_Input.csv:** 4](#_Toc468800449)

[**4.** **Entering Data into the Input Files** 5](#_Toc468800450)

[**5.** **Running the R Code** 11](#_Toc468800451)

[**6.** **Output** 13](#_Toc468800452)

[**6.1.** **Default Output Files** 13](#_Toc468800453)

[**6.2.** **Output Plots** 14](#_Toc468800454)

[**6.3.** **Accessing results for manipulation in R** 14](#_Toc468800455)

[**7.** **Tips and Troubleshooting:** 15](#_Toc468800456)

[**8.** **References** 16](#_Toc468800457)

[**9.** **Contact Information** 16](#_Toc468800458)

[Appendix A. Data Frames used by BioE.r program 17](#_Toc468800459)

[Appendix B: Iterative Solution Algorithm used for Solution Method = 2 or 3 20](#_Toc468800460)

[Appendix C: List of Functions in “BioE\_Function.r” 21](#_Toc468800461)

1. **Introduction**

The Bioenergetics model, as described in [2, 3], has been coded into R, a free open source programming language, for use within the ISEMP (Integrated Status and Monitoring Program) program [1]. This document is intended as a user’s guide for this R based Bioenergetics program [2].

The Bioenergetics model calculates fish growth by solving the energy balance equation:

**Consumption = Respiration + Egestion + Excretion + Growth**

The model inputs initial fish weights and information about the habitat (temperature, prey information), from which it calculates growth over time, based on species specific equations for consumption, respiration, egestion, and excretion. For a given species, one of three consumption equations will be specified. Similarly, one of two respiration equations and one of three egestion/ excretion) equation sets are used for a given species. This implementation of the model does not include weight losses due to reproduction.

The Bioenergetics model can be run in three different simulation modes. The first mode takes inputs as described above and outputs fish growth. For the 2nd mode, total fish growth is instead used as an input, and average p-values are calculated. For the 3rd mode, total consumption is input by the user and average p-values are calculated.

A single species of fish can be modeled with each run of the model. However, multiple sites can be modeled simultaneously, for as many time steps as the user requires. “Site” is defined, for this program, as a set of temperature, p-value, and prey inputs over time of the simulation. All sites simulated simultaneously must be run the same number of time steps (days). For each site modeled, the user may model more than one fish (as defined be start weight, and, if applicable, end weight or consumption).

This document does not serve as a technical description or justification for the science behind the model implemented. Scientific justifications and descriptions of the model are in [2] and [3].

1. **System Requirements**

The BioEnergetics model has been written in the R programming language. R is a free, open source programming language designed specifically for statistical computing and graphics. The software and documentation are available free at: <http://www.r-project.org/>.

1. **Files included in the BioEnergetics Model**

Four files are needed to run the BioEnergetics model. To run the model, these four files must be copied to a common directory. The four files are as follows:

* 1. **BioE.R**:

This is a short R script that serves as the main R program that calls all other functions and runs the BioEnergetics Program.

* 1. **BioE\_Functions.R**:

This R file contains a series of functions that make up the engine of the Bioenergetics model. These include functions for all equations in the BioEnergetics model; functions to read and write data, input, and output files; functions to iteratively solve for p-values; and a function to plot results.

For users who wish to understand in detail how the code works, or are interested in making code modifications, details on each function, and the data structures used by the program, are included as appendices to this document.

* 1. **Input\_BioELookup\_Tables.csv:**

This file contains physiological parameters for each fish species which can be modeled with the BioEnergetics model. Initially, these values were taken from Appendix A of ***Fish BioEnergetics* [3]**. The user may add columns for any additional fish for which parameters are available (up to 200 species of fish may be included). In addition to required parameters, this dataset includes rows to specify which equations are to be used for consumption, respiration, and egestion/excretion, for each fish species included in the lookup table.

Please note that this data file is read using one of the R functions in “BioE\_Functions.r”. The function requires that each parameter be listed on a specific row in the column under the species name. Unlike working within spreadsheets alone, links to specific cells will not update if cells are moved, rows or columns are added, etc. Adding or moving rows, re-arranging the order of parameters, etc. are changes that will NOT be recognized by the Bioenergetics program, and the code will not run properly if such changes are made. However, in the file “BioELookupTale.csv” file, new columns CAN be created, in order to add species to the lookup table; but all new columns must be entered in the same parameter order, by row, as existing columns.

* 1. **Example Input Files: Input\_BioE\_Header.csv, Temp\_P\_Prey\_Input.csv, and Weights\_Consumption\_Input.csv:**

Three input files are used with the program: a header file (with information regarding the number of sites, number of time steps, and type of simulation); a file containing information on the daily temperatures, prey availability, and p-values, for one or more locations to be modeled; and a file containing start weights, end weights, and total consumption for each site modeled. Note that depending on the simulation method used, not all input information is necessary; although the input files must contain rows and columns for all information, whether use or not for the chosen simulation method. Examples of both of these files are included. These may be used as templates for use when setting up simulations. Details are described below.

1. **Entering Data into the Input Files**
   1. **Header File “Input\_BioE\_Header.csv”**

The first file must be called “Input\_BioE\_Header.csv”, as the name of this file is hard-coded in the R-code. An example or template file is provided with the code, and the user will simply need to modify the template and save under the name “Input\_BioE\_Header.csv” (rather than fill out the file from scratch). Note, this must be saved in comma separated value (.csv) format. The following table illustrates the information that must be entered into this file.

Figure 1: Inputs Required in “Input\_BioE\_Header.csv”



Note that formatting is shown in Figure 1; however, this is for illustration purposes only. Comma separated values include no cell formatting. Details on each input are described by table 1:

**Table 1: Inputs Required in Header File “Input\_BioE\_Header.csv”**

|  |  |
| --- | --- |
| **Input** | **Description** |
| **Species Name** | The name of the species being modeled. This **must** be spelled exactly as in the file Input\_BioELookup\_Tables.csv, including case. A list of names is also provided in the header file for reference. |
| **Simulation Method** | Specifies which input / solution method is to be used:  1: Enter p-values, solve for weights  2: Enter end weights, solve for average p-values  3: Enter total consumption, solve for average p-values |
|  |
| **Number of Sites & Fish to Model** | The number of sites & fish combinations being modeled simultaneously. This can be 1 sites modeled with multiple fish, multiple sites modeled with 1 fish each, or any combination thereof. This number must equal the total number of data entry rows on the **Start weight, end weight, and/or consumption Input filename** file. |
| **Number of Time Steps** | The number of days to be modeled |
| **Pred** | Specific energy, in Joules per gram of fish, of the fish being modeled. |
| **Prey** | Specific energy, in Joules per gram, of the prey consumed by the fish being modeled. |
| **Oxygen** | Constant used in Respiration equations to convert mass lost to respiration (in grams per gram of predator mass per day) to energy lost to respiration (in Joules per gram of predator mass per day). Default value is 13560. |
| **PFF** | Percent indigestible prey, as described in section 2.4 of Ref. 3. |
| **Stability Factor** | Factor used in numerical algorithm for simulation methods 2 and 3. Increasing this factor increases the speed at which the algorithm will converge to a solution. However, if this number is set too high, the solution algorithm becomes unstable will fail to converge. If the algorithm converges, the Stability Factor will not affect the results, only the speed at which the program converges to the solution. See “Useful Tips” for more information. |
| **Error Lower Limit** | For simulation methods 2 and 3, an iterative algorithm is used to approximate a solution. The iterative solution continues until the error is less than the specified error lower limit. For solution method 2, the error is calculated end weight - specified end weight, in grams. For solution method 3, the error is calculated consumption - specified consumption, in grams. |
| **P-val and temperature filename** | Name of file, including “.csv” suffix, containing temperatures, p-values, and prey information for each site over time. |
| **Start weight, end weight, and/or consumption Input filename** | Name of file, including “.csv” suffix, containing start weights, end weights, and/or total consumption, for each site. |

**P-val, temperature, and prey Input File**

Figure 2 shows an example of the input file that must be generated for each site for information containing temperature (TEMP) by day, P-value by day, and percent of percent of each prey by day along with the corresponding pretty energy density values corresponding to those percentages. All cells must be filled out by the user. All cells must contain a value, even if the value is zero. (The code is not sophisticated enough to handle blanks). As in figure 1, formatting is included here for illustrative purposes only. Actual input files include, in .csv format, do not enable cell formatting. Note that each site modeled need only be listed once. Multiple fish may be modeled per site, as specified in file **Start weight, end weight, and/or consumption** (see below). Note that sites must be numbered in the leftmost input column, and numbers should begin with 1. These site numbers must match site numbers used in the input file **Start weight, end weight, and/or consumption**, and all site numbers must be listed in integer values in increasing order. Descriptive site names (used in plots and output files) are not listed in this input file, but will be input in the file **Start weight, end weight, and/or consumption.**

Figure 2: Input File Example for Site-Specific Data Entry



Cells in row 3, starting under “species 1” and up to “species 20”, are used to enter prey energy density for up to 20 prey species. The species names may be changed from “species 1”, “species 2”, etc., to the actual species name or an appropriate abbreviation.

Data for temperature, p-values, and percent of each prey type, are entered by day starting in row 7. Data for each day, for each site, does not need to be entered. The first day (day 1) requires a value, and the last day (as specified in the header file) requires a value. Additional days may be entered. For any days not listed, all values for un-specified days will be calculated via linear interpolation between the two nearest days for which values are specified. Note that days specified must be listed in increasing order (i.e. day 1, 15, and 40). Specifying constant inputs still requires at least two rows per site. Information for all sites must be specified. All cells must contain numeric values, even if the solution method doesn’t require a value (for example, if p-values are being estimated, as specified by entering a “2” in cell B2 of the header file), dummy values must be present. If you do not want values interpolated, you simply ensure there is one row of input data for each day at each site to be modeled.

The leftmost column contains “Site”. Sites must be numbered (Site Names for plots and results are entered into a different input file); and site numbers must be consistent across input files. The next column contains “Day”. For each site, the first day must be 1, and the last day must be the same as the number of days specified in the header file. Additional days may optionally be specified at values between day 1 and the maximum day. For the remainder of the columns, temperature values (in “Temp”), p-values (“P-Val”), and the percent of each prey type are entered.

Up to 20 different prey species may be included in the simulation. The upper right box, in figure 2, shows the format for data entry. Row 2, columns D through W, are used to label prey (for user reference only; these string values are not read or used by the program). In row 3, the prey energy density of each prey type is entered. In rows 7 on, the fraction of the diet containing each prey species is specified, on a daily basis. Each row should add up to exactly 1.0. As in all other inputs, blanks cells are not allowed; values of zero should be input where prey is not included, for all columns D through W, and all rows from 7 through (7 + Number of Timesteps-1).

**Start weight, end weight, and/or consumption Input file**

A separate input file is used to specify site names, start weights (beginning fish weight), end weights, and total consumption. Note that all cells must contain a value, even if they are not applicable for the solution method (specified in the header file cell B2). In this case, zeros or “-99” values, or any “dummy” value, as long as it is numeric, is acceptable.

Figure 3 shows an example for four sites. Note that the site numbers listed here must match the sites numbers specified in the P-val, temperature, and preyinput file. Also note that the total number of rows of input (not included the header row) must equal the value in the **Total Number of Sites & Fish Combinations to Model** cell in the header file.

Figure 3. Example Start Weight, End Weight, and Consumption Input File.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Site | Run Name | Start Weight | End Weight (if Input$B6=2) | Total Consumption (if Input$B6=3): grams |
| 1 | Site 1 Fish A | 40 | 42 | 125 |
| 2 | Site 2 Fish A | 40 | 45 | 125 |
| 2 | Site 2 Fish B | 45 | 50 | 130 |
| 2 | Site 2 Fisch C | 50 | 55 | 135 |
| 3 | Site 3 Size 1 | 40 | 45 | 125 |
| 3 | Site 3 Fish B | 45 | 50 | 130 |
| 3 | Site 3 Fish C | 50 | 55 | 135 |
| 3 | Site 3 Fish D | 55 | 60 | 140 |
| 4 | Site 4 Fish A | 40 | 45 | 125 |
| 4 | Site 4 Fish B | 45 | 50 | 130 |

The user may model multiple fish per site, as is shown in example three. For this example, there are four sites defined: site 1, site 2, site 3, and site 4. For site 1, there is just one fish modeled. For site 2, there are three fish modeled; for site 3, there are four fish modeled, and for site 4, there are 2 fish modeled. Any combination of sites and fish per site is allowable; however, the order that sites appear in the **P-val and temperature** input file **must match exactly** the order that sites appear in this input file, regardless of how many fish per site are modeled.

Run name, on this file, can be used to describe the site and fish combination being modeled. This input will be used as an identifier on output plots and output files.

**Entering Fish Parameters into “BioE\_Lookup.csv”**

The file “BioE\_Lookup.csv” contains physiological parameters used by the BioE model, and specifies which set of consumption, respiration, and excretion / egestion equations are to be used for a given species of fish. Initial parameters were taken from the appendix of “Fish Bioenergetics” [3]. Parameters may be modified, and/or parameters for additional fish species may be added to this file.

Figure 5.1 displays the layout of the BioE\_Lookup.xlsx file, including data for several species. All data for this file is on a single worksheet.

To add an additional species, simply add a new column on the right of the list, or insert a new column anywhere within the existing file structure. The R-program matches the name of the species being modeled to the column header; therefore, the order of columns does not matter. However, they order of the rows must be as seen in Figure 5.1. Columns A, B, and C must not contain parameters for any species (since the R-program begins searching for a species match in column D, and continues searching columns to the right). In addition, there should not be any blank columns within the table, as the R-code may interpret this as the end of the table. Blank cells within columns are acceptable, where constants are present that are not required for the specified equation.

**Figure 5.1: Left-most columns of “BioE\_Lookup.xlsx” file**



Note: care should be taken not to include any characters or formatting of cells to the right of the right-most column. The read.csv function in R is not robust enough to recognize formatted, but blank columns as “empty”. Unpredictable errors may result if unused characters, formatting, or hidden characters are present in columns to the right of the rightmost column.

After editing, the file must be saved under the same filename, “Input\_BioELookup\_Tables.csv”. It is recommended that a backup copy be saved under a different filename prior to editing.

1. **Running the R Code**

Once the input files for each site, and the header file “Input\_BioE\_Header.csv” has been filled out, the user can run the actual R script. The steps listed below must be completed to execute the script. Note than since R is a command line programming environment, there cannot be a pre-compiled executable file. Instead, in order to run the program, the user must open R and run the main script manually. However, the user does NOT need to open the accompanying file containing additional R-functions. These are called automatically by the main script.

**Step 1: Open R and set the working directory**

Launch R. Change the working directory to the folder where the R scripts and input files are located. All input files and R scripts must be in the same directory. To change the working directory in R, with the R console as the active window, click “File”, “Change Dir”, and navigate to the folder where the files are stored.

**Step 2: Open R-script “BioE.R”**

Open the R.script “BioE.R” by selecting “File”, “Open Script”, and selecting “Bioe.R”. This is the main R script that calls all other functions required to run the Bioenergetics analysis.

**Step 3: Run the Script**

To run the entire program, from the “edit” menu, select “run all”. (A shortcut method is to use <ctrl-a> to select all, followed by <ctrl-r> to run the entire selection.) Depending on the number of sites, number of days modeled, and the simulation method specified, the program will take anywhere from a few seconds to several minutes to run. If simulation method 1 is specified (p-values are entered, weights are calculated), the program should take less than a minute or two to run (except, of course, if many sites or many hundreds of time steps are included). If Simulation Method =2 or Simulation Method =3 is input, the program may take many minutes or more to find a solution, due to the iterative procedure required. Details on the iterative solution procedure are found in the appendix.

**Monitoring Progress of Iterative Solution**

**Figure 6.1. Real Time Program Updates on Progress of Iterative Solution**

If Simulation Method =2 or Simulation Method =3 is used, after each iteration the program will write a “progress update” to the R console window. Included in this update are the P-value for each site, assumed by the most recent iteration; and the error (difference between calculated results and user specified endpoints) for the most recent iteration. The user can confirm that error for each site is getting smaller in magnitude as the number of iterations increases. The iterative procedure will conclude when the error term for every site is less than 0.1 gram. An example of the progress update is shown by Figure 6.1.

Each set of iteration updates lists one row for each site being modeled. On each row, the value “Pnew” is the latest estimate for the p-value, and the “Error” is the difference between the target value (end weight for solution method = 2, total consumption for solution method = 3), and the value achieved using the Pnew as the p-value and all other model inputs from the input file. When the “Error” for all sites modeled is less than 0.1 gram, the iterations will cease and the values of “Pnew” at the last iteration as taken as the solutions for the p-values

1. **Output**

Functions are called by the main R-script to create a file of output data, as well a set of default plots. For users requiring output parameters that are not included in the output file, consult Appendix A for details on the data structure. Users with basic familiarity with R will be able to access the results within R, and perform any additional analysis and/or plotting by writing their own R code that can run within the workspace present upon completion of the BioE.R script.

* 1. **Default Output Files**

Output files are also in .csv format. 5 output files are generated from this code: “Results\_Respiration.csv”, “Results\_Consumption.csv”, “Results\_Excretion.csv”, “Results.Egestion.csv”, and “Results\_Growth.csv”. In each file, results are listed by day for each site included in the analysis. Results will be written to the current working directory in R, and previous results will be over-written.

Figure 7.1 shows example data for growth, from an output file “Results\_Growth.csv”. All output files will follow this format, with one column for each output metric and site combination. Some rows and columns have been truncated for this example.

Figure 7.1. Example Output in “Results\_Growth.csv”



For the files: “Results\_Respiration.csv”, “Results\_Consumption.csv”, “Results\_Excretion.csv”, and “Results.Egestion.csv”, results are output in four formats for each site. The first two sets of outputs are for specific mass and specific energy: the units are grams per gram of fish per day and joules per gram of fish per day. The second set of outputs includes absolute mass and energy: the units are grams per day and joules per day.

**Figure 7.2.1. Example Output Plot: Growth vs Time, by Site **

* 1. **Output Plots**

Plots of fish growth vs. time, by site; and fish weight vs. time, by site, and generated by default. The <Page Up> and <Page Down> keys can be used to toggle back and forth between the two plots. Figures 7.2.1 and 7.2.2 show example output plots.

* 1. **Accessing results for manipulation in R**

In addition to the output file and plots, data can be accessed directly from within R. The data are stored in three global data frames: Input, Constants, and Results. Descriptions of each of these data frames are included in Appendix A. Users with some familiarity with R programming will be able to manipulate these data as needed to perform additional analysis, plot data, and/or modify the existing R scripts. Appendix A also provides a brief example of how to access data from these data frames within R.

**Figure 7.2.2. Example Output Plot: Fish Weight vs Time, by Site**



1. **Tips and Troubleshooting:**

While efforts have been made to make the Bioenergetics code reasonably robust, there are many instances where set-up errors, data input problems, etc., will cause the code to crash or produce erroneous results. Common problems, the issue likely causing the observed symptom, and user solutions, are listed by table 8.1. See the Contact Information (section 10) for further support, if needed.

**Table 8.1. Common Problems, issues, and solutions**

|  |  |  |
| --- | --- | --- |
| **Symptom** | **Issue** | **Solution** |
| Convergence Not Occurring | P-value less than zero required to achieve end weight specified. | Check constants and input data. |
| Solution unstable - error going to +/- infinity | Decrease stability factor. |
| Iterative solution too slow - error not approaching zero quickly enough | Increase stability factor. |
| Within R, a message such as "The following object(s) are masked from 'InputData (position 4)':" appears. | R-script aborted during a prior run, prior to data management completion. Data and results may be compromised. | Close and re-start R, re-run script. |
| P-values greater than 1 found | P-value greater than 1 required to achieve end weight specified. | Check constants and input data. |
| R error "Cannot Open File…." | Working directory not set to directory containing R scripts and Excel files. | Set working directory to directory containing R scripts and Excel files. |
| Error reading “[filename.csv]” | Cells re-arranged from required format. | Check rows and in proper order. Check all numeric cells are numeric, and ensure no required data is missing. |
| Extra characters, formatting, or hidden characters to the right of the right-most column. | Select columns to the right of the right-most column of species constants, and select <edit>, <delete> to remove any hidden or unwanted characters or formatting. |
| Other | Number of Input Times for p-vals, temperatures, and prey not equal to number of time steps in header file | Ensure number of rows in input file tables is equal to number of time steps. |
| Other | Blanks in input files instead of zeros or numeric values. | Enter numeric values (even for zeros or for cells that aren’t applicable to a specific simulation) in all input file cells |

1. **References**
2. ISEMP (Integrated Status and Monitoring Program) home pages: http://www.nwfsc.noaa.gov/research/divisions/cbd/mathbio/isemp/index.cfm
3. Bouwes, Nicolaas. Weber, Nicholas. Eco Logical Research, Inc.

ISEMP (Integrated Status and Monitoring Program): Lesson Learned Synthesis Report, CHAPTER 8: Growth Potential Models (pages 139 – 152).

1. [Hanson, Paul C.](http://cfllibrary.uwcfl.org/biblio/author/586), [Johnson Timothy B.](http://cfllibrary.uwcfl.org/biblio/author/1895), [Schindler Daniel E.](http://cfllibrary.uwcfl.org/biblio/author/1039), and [Kitchell James F.](http://cfllibrary.uwcfl.org/biblio/author/29) [**Fish bioenergetics 3.0 for Windows**](http://cfllibrary.uwcfl.org/node/1146), Madison, Wisconsin, p.E-6, (1997)
2. **Contact Information**

For further information, contact:

Matt Nahorniak, [South Fork Research, Inc](http://www.southforkresearch.org/)

Phone: (541) 740-5487

E-mail: [matt@southforkresearch.org](mailto:matt@southforkresearch.org)

# Appendix A. Data Frames used by BioE.r program

Three global data frames are used with the BioE.R program. Data read from the input file is included in the data frame “Input”. Data read from the lookup table is stored in the data frame “Constants”. Results of the analysis are stored in the data frame “Results”.

Note that P values are stored in the “Input” data frame. If simulation method = 1 is selected, p-values are, indeed, input values. If simulation method =2 or 3, p-values are computed rather than inputs. Nevertheless, the results are still stored in the “Input” data frame.

**Table A.1. Elements within Global Data Frame**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data.Frame** | **Elements within Data.Frame** | **Description** | **Data Type** |
| **Input** | Species | Species being modeled | text variable |
| SimMethod | Simulatioin Method Selcted | Integer (1, 2, or 3) |
| Wstart | Starting Weights (grams) | Vector of length N.sites |
| Endweights | End Weights (grams) | Vector of length N.sites |
| TotalConsumption | Total Consumption (grams) | Vector of length N.sites |
| p | Daily p-values | N.steps x N.sites Array |
| Temps | Daily temperature values | N.steps x N.sites Array |
| N.sites | Number of sites being modeled | Integer |
| N.steps | Number of time steps (days) being modeled | Integer |
| sitenames | Names of sites veing modeled | Vector of text strings, of length N.sites |
| Pred | Joules per gram body weight of fish being modeled | Real Number |
| Prey | Joules per gram body weight of prey | Real Number |
| Oxygen | Constant used in Respiration equations to convert mass lost to respiration (in grams per gram of predator mass per day) to energy lost to respiration (in Joules per gram of predator mass per day). | Real Number |
| stab.factor | Stability Factor | Real Number |

|  |  |  |  |
| --- | --- | --- | --- |
| **Data.Frame** | **Elements within Data.Frame** | **Description** | **Data Type** |
| **Constants** | Consumption | List of constants for consumption equations | List of real numbers |
| Respiration | List of constants for Respiration equations | List of real numbers |
| Excretion | List of constants of Excretion equations | List of real numbers |

|  |  |  |  |
| --- | --- | --- | --- |
| **Data.Frame** | **Elements within Data.Frame** | **Description** | **Data Type** |
| **Results** | TotalC | Total Consumption | Vector of length N.sites |
| W | Weight (grams) | N.steps x N.sites Array |
| Growth | Growth (grams) | N.steps x N.sites Array |
| Gg\_WinBioE |  | N.steps x N.sites Array |
| Gg\_ELR |  | N.steps x N.sites Array |
| Growth\_j | Growth (Joules) | N.steps x N.sites Array |
| Consumption | Daily Consumption (grams) | N.steps x N.sites Array |
| Consumption\_j | Daily Consumption (joules) | N.steps x N.sites Array |
| Excretion | Daily Excretion (grams) | N.steps x N.sites Array |
| Excretion\_j | Daily Excretion (joules) | N.steps x N.sites Array |
| Egestion | Daily Egestion (grams) | N.steps x N.sites Array |
| Egestion\_j | Daily Egestion (Joules) | N.steps x N.sites Array |
| Respiration | Daily Respiration (grams) | N.steps x N.sites Array |
| Respiration\_j | Daily Respiration (Joules) | N.steps x N.sites Array |
| S.resp | proportion of assimilated energy lost to SDA in g/g/d | N.steps x N.sites Array |
| Sj.resp | proportion of assimilated energy lost to SDA in J/g/d | N.steps x N.sites Array |

**Accessing Data from Data Frames**

Data within data frames in R is accessed by either attaching the data frame (making the data within that frame global), or by accessing a variable within the data frame using the “$” symbol, as in “data.frame.name$variable”.

For example, if the user wanted to plot growth2, for by site, versus respiration, the following script could be written after the BioE.R code was run. Start by selecting “file”, “new script”, from within the same R workspace.

**Example Script to Access, Manipulate, and Plot Results**

# Example of how to access and use Results within R

# Create a new variable to store Growth Squared

G.squared = Results$Growth^2

# Plot Growth Squared by Respiration

plot(Results$Respiration[,1], G.squared[,1],

xlim=c(min(Results$Respiration), max(Results$Respiration)),

ylim=c(min(G.squared), max(G.squared)),

main = "Respiration vs Growth Sqaured, by site")

for (i in 2:N.sites) {

points(Results$Respiration[,i], G.squared[,i], col=i)

}

# Make a legend

site.names=rep(" ", Input$N.sites)

for (i in 1:Input$N.sites) {site.names[i]=as.character(Input$sitenames[1,i])}

legendtext = site.names

legend("topleft",legendtext, col=seq(1:Input$N.sites), pch=16, cex=1)

The above script produces a plot such as shown below:



# Appendix B: Iterative Solution Algorithm used for Solution Method = 2 or 3

If solution method = 2 or solution method = 3, an iterative procedure is used to calculate p-values that, when used, result in either end weights (solution method = 2) or total consumption (solution method =3) that match those specified in the input file.

The iterative solution is performed in the “BioE” function, in the file “BioE\_Functions.R”. The program Using these p-values, the function next calculates consumption and weight gain for each site, as if the p-values used were the known, true p-values. These calculated results are then compared to the user input for either End Weights (if simulation method = 2) or Total Consumption (if simulation method = 3). If the difference between calculated results and user specified ending values is greater than the maximum allowable error (set as 0.1 grams), a new average p-value for each site is estimated based on the prior guess and the difference, and the algorithm is repeated.

The “stability factor” (specified on the Input worksheet of the file “BioEInput.xlsx”) is used to either speed convergence of the iterative algorithm, or increase the stability of the iterative solution. Increasing this factor increases the speed at which the algorithm will converge to a solution. However, if this number is set too high, the solution algorithm becomes unstable will fail to converge. If the algorithm converges, the Stability Factor will not affect the results, only the speed at which the program converges to the solution. See “Useful Tips” for more information.

# Appendix C: List of Functions in “BioE\_Function.r”

**Function “ReadInputFile”**

**Inputs:** N/A

**Outputs:** Species, SimMethod, Wstart, Endweights, TotalConsumption, p, Temps, N.sites, N.steps, sitenames, Pred, Prey, Oxygen, stab.factor

**Description:** This function utilizes the function “read.xlsx” from the package “xlsx” to read all the data from the excel file “BioEInput.xlsx”.

**Function “ReadConstants”**

**Inputs:** Species

**Outputs:** Consumption, Respiration, Excretion

**Description:** This function utilizes the function “read.xlsx” from the package “xlsx” to read all the data from the excel file “BioELookup\_Tables.xlsx”. It uses the “species” parameter from the Input data frame to determine which column from which to pull data.

**Function “ConsumptionEQ1”:**

**Inputs**: W, TEMP, P, PREY, CA, CB, CQ

**Output**s: CMAX, CONS, CONSj

**Description:**  Exponential equation for consumption, used when the consumption equation =1.

**Function “ConsumptionEQ2”:**

**Inputs:** W, TEMP, P, PREY, CA, CB, CTM, CTO, CQ

**Output**s: CMAX, CONS, CONSj

**Description:** Temperature dependent equation consumption equation for warm-water species [3], used when the consumption equation =2.

**Function “ConsumptionEQ2”:**

**Inputs:** W,TEMP,P,PREY,CA, CB, CK1, CTO, CQ, CK4, CTL, CTM)

**Output**s: CMAX, CONS, CONSj

**Description:**  Temperature dependence for cool- and cold-water species [3] used when the consumption equation =3.

**Function ”ExcretionEQ1”:**

**Inputs:** CONS, CONSsj, TEMP,P,FA, UA

**Outputs**: EG, EGj, U, Uj

**Description:** Excretion / Egestion equation set one, used when waste losses are proportional to consumption [3]. Used when the excretion / egestion equation = 1.

**Function “ExcretionEQ2”:**

**Inputs:** CONS, CONSj, TEMP,P,FA, UA, FB, FG, UB, UG)

**Outputs**: EG, EGj, U, Uj

**Description:** Excretion / Egestion equation set 2, used when waste losses are dependent on mass, temperature, and ration [3]. Used when the excretion / egestion equation = 2.

**Function “ExcretionEQ3 :**

**Inputs:** CONS, CONSj, TEMP,P,FA, UA, FB, FG, UB, UG

**Outputs**: EG, EGj, U, Uj

**Description:** Excretion / Egestion equation set 3, similar to equation 2, with correction for indigestible prey [3]. Used when the excretion / egestion equation = 3.

**Function “RespirationEQ1”:**

**Inputs:** W, TEMP, CONS, EG, PREY, OXYGEN, RA,RB,ACT,SDA,RQ,RTO,RK1,RK4,RTL,BACT

**Outputs:** R, Rj, S, Sj

**Description:** Exponential with swimming speed respiration equation, used when respiration equation = 1.

**Function ”RespirationEQ2 “:**

**Inputs**: W, TEMP, CONS, EG, PREY, OXYGEN,RA,RB,ACT,SDA,RTM,RTO,RQ

**Outputs:** R, Rj, S, Sj

**Description:**  Temperature dependent with activity multiplier respiration equation, used when respiration equation = 2.

**Function “CalculateGrowth”:**

**Inputs:** Constants, Input, W

**Ouputs:** TotalC, W, Growth, Gg\_WinBioE, Gg\_ELR, Growth\_j, Consupmtion, Consutmption\_j, Excretion, Excretion\_j, Egestion, Egestion\_j, Respiration, Respiration\_j, S.resp, Sj.resp

**Description:** This function calculates growth and all other results by stepping through each time step, for each site being modeled, and calling the the appropriate equations for consumption, respiration, and Egestion/Excretion.

**Function “BioE”:**

**Inputs:** Input, Constants

**Output:** Results

**Description:** The main function that calls the functions other functions. For solution method=1, it simply calls the function “CalculateGrowth”. For simulation method=2 or simulation method = 3, this function performs the iterative procedure needed to solve for p-values producing results that match the inputs for either endweight (solution method =2) or total consumption (solution method = 3). Details on this are included as Appendix B.

**Function “plot.results “:**

**Inputs:** Input, Results

**Output:** N/A

**Description:** Creates plots of weight vs. day by region and growth per day vs site by region.

**Function “write.output”:**

**Inputs:** Input, Results

**Outputs:** N/A

**Description:**  write the output file “BioEOutput.xlsx”.