# ‘epower’: Statistical Power Analysis for BACI designs; Guide to Software and Statistical Methods

Version 1.3
Oct/2018

# ‘epower’: Statistical Power Analysis for BACI designs; Guide to Software and Statistical Methods

*Prepared by*

**BMT , the Australian Institue of Marine Science, and Queensland Unviersity of Technology**

**Oct/2018**

*Version 1.3*

## Status

This manual is 'Draft' until approved for final release, as indicated below by inclusion of signatures from: (i) the author and (ii) a Director of BMT Pty Ltd or their authorised delegate. A Draft manual may be issued for review with intent to generate a 'Final' version, but must not be used for any other purpose.

**Approved for final release:**

| **Authors** | **Director (or delegate)** |
|---|---|
| Date: | Date: |

**Cover**

Main image: Irvine Island, Bucaneer Archipelago of the Kimberley, Western Australia Replace and describe image as relevant to (BMT Pty Ltd)

## Disclaimer

This manual and associated toolbox have been prepared jointly by BMT, the Australian Institute of Marine Science and Queensland University of Technology. Use is subject to the agreed terms and scope between both the Australian Institute of Marine Science and Queensland University of Technology and BMT Pty Ltd. BMT, the Australian Institute of Marine Science and Queensland University of Technology accept no liability or responsibility for in respect of any use of or reliance upon this manual and associated toolbox.

# Overview

‘epower’: statistical Power Analysis for BACI designs is a toolbox package that has been developed for examining statistical power specifically for the Before/After Control/Impact interaction term in BACI, m-BACI and beyond BACI monitoring designs for detecting environmental impacts.

The toolbox is based on modern statistical methods for testing for statistical significance of a BACI interaction term in hierarchical nested designs, including mixed effect modelling as implemented by INLA in R. The ‘epower’ (v1.3) package is freely available to users under license, subject to BMT’s Terms of use. ‘epower’ can be obtained at: https://[insert.url].

The toolbox package contains two parts, an excel workbook file where pilot data is entered (if available) and the particular analysis to be undertaken is specified, and an R package with two main functions: fitData, which reads in the analysis specification information and pilot data, fits the appropriate mixed model and extracts a posterior sample of the parameters required to build the monte-carlo simulation; and assessPower, which carries out a monte-carlo simulation and summarises the results. The tool box features:

- The ability to determine power for BACI designs with up to two spatial and two temporal hierarchical groupings
- Analyse designs with either fixed or random replicate units (transects/quadrats)
- Examine both Type 1 and Type 2 error
- Conduct cost benefit analysis by changing levels of replication at any hierarchical level of the design within a single analysis
- Explore power for a range of effect sizes for your existing design, or with increased/decreased levels of replication
- Effect sizes applied as percentage (multiplicative) or fixed effects

## Contents

# List of Figures

# 1. Introduction

## 1.1 Background

The ‘epower’ package was developed during the Western Australian economic boom of 2010 to 2015, when increased capital expenditure in the resources sector led to several infrastructure development projects in Australia’s north-west. Given the sensitivities of the developments, the projects were subjected to intense regulatory scrutiny and approved only under the strictest of environmental conditions.

Conditions included the need to demonstrate that the proponent’s (and their consultants’) approaches to environmental management were sound and of appropriate statistical rigour. Notably, some conditions required proponents to demonstrate that the environmental thresholds chosen as management triggers, were reasonably detectable given the proposed sampling effort and known levels of background variability.

This posed challenges given the complexity of the monitoring programs, which typically comprised mixed model (Before-After-Control-Impact) BACI designs with fixed and random factors, capturing multiple sources of variation across space and time.

Statistical power assessment for mixed model designs is not straightforward (Searle 1971) and is best undertaken using simulations (Underwood & Chapman 2003). The ‘epower’ package was thus born as a practical tool for environmental managers and consultants to assess the statistical power of BACI designs. The package employs a simple user interface, which when correctly populated, calls on the background computer code to run the simulations automatically. Several BACI designs are catered for, with up to two spatial and two temporal hierarchical levels allowed.

It is not the intent of this manual to act as a statistical text, explain the intricacies of statistical power in the context of BACI designs, or the underlying mixed model framework; there are plenty of good texts on these subjects (Bates 2010, Zuur et al. 2009). A more detailed account of the statistical details of the ‘epower’ package can be found in Fisher et al. (in review). The purpose of the manual is to outline the application of ‘epower’, its underlying assumptions and briefly its underlying mechanics. Use of ‘epower’ does not require background knowledge of computer coding, but users will benefit from having a basic understanding of experimental design for detection of environmental impacts and basic familiarity with MS excel.

# 2. Real-world environmental monitoring

As environmental consultants, we work with a range of monitoring program designs from non-parametric analyses involving comparison of medians and other percentiles (as per ANZECC & ARMCANZ 2000), to simple parametric hypothesis-based assessments of two populations (t-tests)), to more complicated multi-factorial assessments involving analysis of variance (ANOVA). ANOVA in particular has long been considered a powerful and versatile tool in experimental design, which is suitable for planned environmental monitoring in which predictive hypotheses are tested by collecting data at various spatial and temporal scales (Underwood 1991). Most recently, ANOVA methods are being superseded by generalised mixed model analyses that can cope with non-Gaussian response variables (rather than falsely assuming normality of response data) and more efficiently handle the non-independence associated with many hierarchal sampling designs (e.g. sites nested within locations), that are typical in environmental monitoring.

## 2.1 Testing for environmental impacts

In any attempt to analyse monitoring data to assess environmental impact, the analyst must firstly define the hypothesis to be examined, determine which factors (or which interaction of factors) should be examined in order to draw reasonable conclusions about whether to accept or reject the null hypothesis (typically of 'no effect"), and make sure that any underlying dependencies are accounted for in the statistical method used. Underwood and Chapman (2003) provide a comprehensive overview of the types of responses one might expect following an impact, including 'press' and 'pulse' type effects. In BACI designs (generally intended to detect press disturbances, Fig 1.1), the Before v After × Control v Impact (BACI) term is the factor of interest, because it is this term that identifies that a change has occurred at the impact site that is beyond any natural change that occurred at the control(s). As described in detail in Fisher et al. (in review), for a BACI analysis the hypothesis being tested is that the BACI interaction term is 'significant' beyond any naturally occurring spatial and temporal variability.

Monitoring programs designed to detect changes in environmental systems pose a range of statistical challenges, including: high levels of natural temporal and spatial variability, missing data and unbalanced designs, and complex hierarchical structure (in time and space) that must be accommodated to avoid problems with pseudoreplication. The statistical methods to be employed for detecting change must be capable of coping with these challenges. Further, the associated power to detect change must be assessed in the context of these challenges, and such should be carried out within the same statistical framework. The ''epower'' package uses a Bayesian mixed modelling approach (Rue et al. 2009) to fit the appropriate hierarchical sampling design to the data, and assess relative probabilities for a model with, and without, this BACI interaction term (see details in Fisher et al. in review).

## 2.2 Statistical power

Secondly, as well as assessing the hypothesis of a 'significant' BACI interaction term, it is important for the analyst to examine the statistical power of their design, to determine if the design would have been capable of detecting an effect if an effect does in fact occur. Such an analysis can also be done (and perhaps preferably so) using just 'before' data (or other pilot) data prior to the impact being examined, in order to ensure that the survey design is sufficient to detect the relevant impact the study intends to examine. Power (1-β) is defined as the probability of detecting a change if the change exists, and depends on a number of factors including alpha (and/or the hypothesis testing method adopted), sampling effort and background variability. The doctrine is for users to achieve power of 0.8 with an alpha of 0.05, or respectively 'an 80% chance of detecting the specified change, if the change exists' and 'a 5% chance of concluding that the specified change occurred, when in fact it didn't'. With increasing computing power and the rise in generalised mixed models allowing the relaxation of the assumption of normality along with the ability to model response variables of interest on their natural scale (for example, Poisson for count data), so too have methods for testing hypotheses also evolved. Recent advances include likelihood ratio tests (Zuur et al. 2009), weight of evidence approaches based on AICc (Burnham and Anderson 2002), and Bayesian methods for assessing relative model probabilities (the hypothesis testing implemented in 'epower', see Fisher et al, in review). While some of these alternative hypothesis testing procedures do not have a specified 'alpha' level (as is the case with the method implemented in 'epower'), the corresponding alpha associated with the design can easily be assessed through 'epower' by including scenarios with 0 applied effect. Assessing the proportion of 'significant' outcomes associated with 0applied effect provides a direct estimate of alpha, the type one error associated with the design for the implemented hypothesis testing method.

Statistical power of generalised mixed models must be carried out using simulation methods (Underwood and Chapman 2003). The 'epower' package uses a monte-carlo simulation method to assess power. The package first fits the appropriate generalised mixed model to the supplied pilot data using the Bayesian package INLA (Rue et al. 2009). A posterior sample of the resulting fitted mixed model (including estimates for variance at each level of the spatial and temporal random hierarchy) is obtained and used the construct the monte-carlo simulation (see Fisher et al. for technical details). Using this posterior sample the monte-carlo simulation generates *n* simulated datasets (with *n* being the number of requested simulations) based on the design to be assessed for statistical power within the toolbox excel workbook interface. Each simulated dataset will have a user defined 'effect' imposed in the 'after' treatment of the 'impact' location(s) only. Effects can be applied as multiplicative effects (a proportion of the existing value) or as fixed effects (see details in Fisher et al. in review). The hypothesis testing procedure is then applied to the simulated datasets, and the proportion yielding a 'significant' outcome recorded, and calculated as a percentage of the number of simulated tests.

## 2.3 Functionality of 'epower'

The 'epower' package is designed to assess the power of BACI designs to detect 'press disturbances'. Under a press disturbance, affected organisms increase or reduce their numbers to a new and stable running average (Figure 2.1). As many ecological systems are inherently naturally temporally and spatially variable, we concur with Underwood (1991, 1993) that replicate control locations are required to be certain that an impact actually occurred at the 'impact' location and is not a result of 'random chance'. As such, 'epower' requires at least three 'sites' (one impact and at least two controls), and at least three 'before' impact surveys, and this is simplest BACI design allowed (see Figures 2.1-A and 2.2-A). An underlying assumption of the BACI analysis is that surveys through time capture replicate 'before' assessments of the system that represent natural temporal variability. In this context temporal patterns should appear 'random' (see Fig 2.1) rather than as a distinct 'time series' exhibiting strong and consistent temporal trends. If trends exist in the data, they may need to be removed (de-trended) or other statistical approaches used.

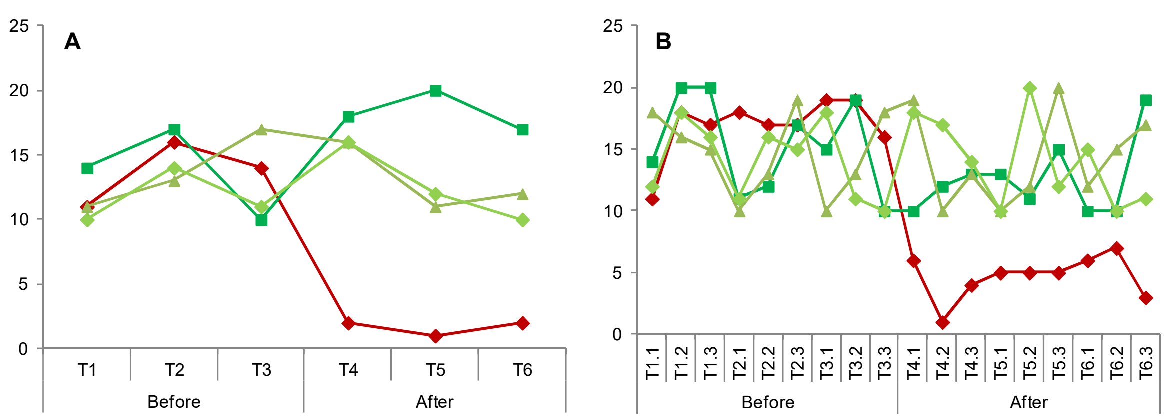



**Figure 2.1** **Examples of press disturbances.** The survey design in (A) is an example of the simplest BACI analysis accommodated in 'epower'. This includes an impact location (red), three control locations (green), surveyed three times 'before', and three times 'after' a disturbance. BACI designs can involve other levels of sampling hierarchy, such as that shown in (B) with the same three control locations surveys three times before and after, but with three sub-times surveyed during each time period.

If the supplied data contains both 'before' and 'after' data, 'epower' will fit the appropriate random structure as well as the fixed effects and provided the summary statistics required to assess the 'significance' of the BACI interaction term. If no 'after' data is available, the 'before' data is fit based on the appropriate random structure, and the Control Impact fixed factor only.

The present package (V1.3) allows users to assess the power of BACI designs incorporating up to two temporal factors and two spatial factors (Figure 2.2-D) and any combination thereof (Figure 2.2-B&C), providing there is at least one spatial and one temporal level of the hierarchy (Figure 2.2-A). Within these factors, users are provided opportunity to manipulate the number of levels and replicate surveys at will, with the only limit being the computational time required to run the simulations. If the design includes fixed replicates (i.e. the same quadrat or transect is visited through time) this must be included as the second spatial hierarchical level (i.e. 'Spatial 2' or Site), with only a single replicate. This is necessary to properly model the non-independence associated with repeated sampling of fixed replicates.

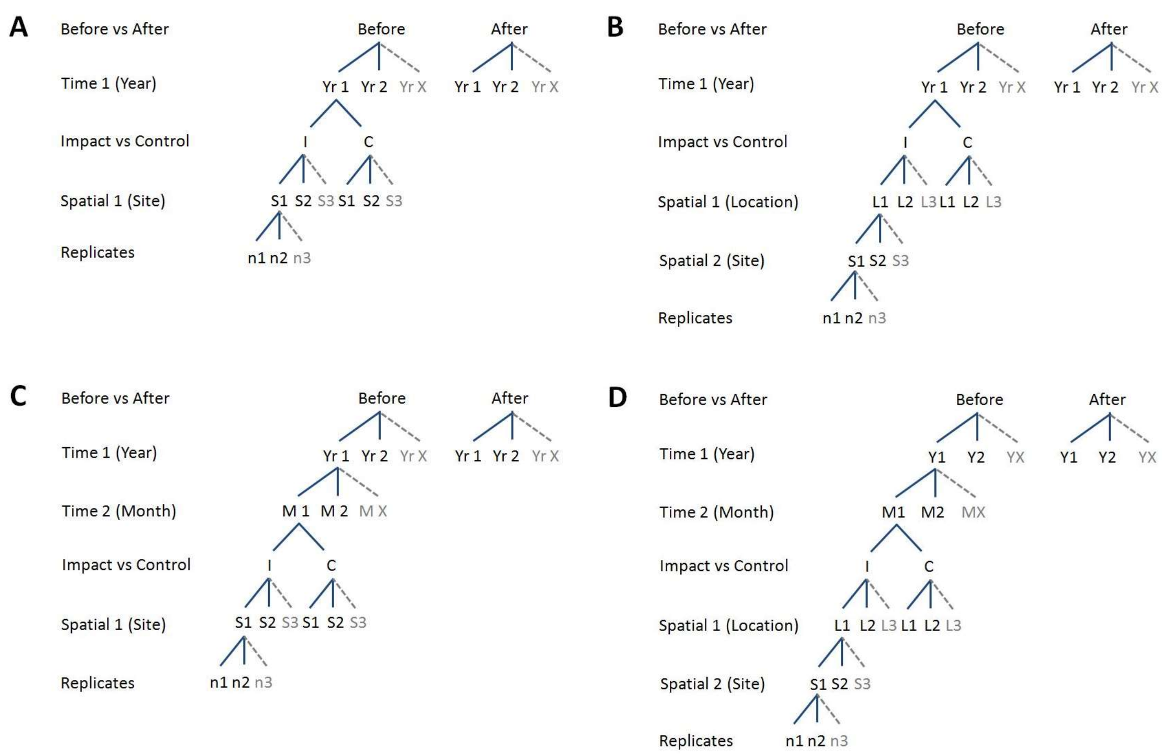



**Figure 2.2 Hierarchical designs available under 'epower' V1.3.**

Note that only designs similar to the pilot data can be examined (i.e. the design must have the same structure with respect to spatial and temporal hierarchical random effects). While different levels within each hierarchy can be explore (3 versus 6 sites, nested within locations, for example), the underlying design must remain the same. For example, if there is no sub-time hierarchical level in the supplied pilot data, there is no estimate of the sub-time variance components and data with sub-time structure cannot be effectively simulated.

# 3. The 'epower' Interface and usage

The 'epower' package was specifically developed for use by a wide range of environmental practitioners that may not have the specialist skills required for statistical programming in R, despite having a robust understanding of BACI sampling design and related theory. To facilitate use by such practitioners we built an MS excel workbook interface (epower_interface_V1.3.xlsx) to allow specification of the design, the scenarios to be examined, and the input of the data to be used in the analysis. This excel workbook interface is the only input that must be supplied by the user (via an interactive dialogue box) to run the toolbox in the underlying statistical programming language “R”. The workbook has two worksheets. The first is the ‘design_specification’ sheet, which is where all the information regarding the survey design and scenarios to be explored are supplied by the user. The second sheet is the ‘pilot_data’ sheet, which is where the data to be used in the analyses are supplied. The design_specification sheet includes five sections: Response Type (where the type of data to be analysed is specified); Design Specification (the details of the specific survey design being examined, including the names (headers) of the columns corresponding to the response variable, the hierarchical structuring variables, such as Location and Time, the Before v After and Control v Impact columns, and the actual values corresponding to each state or “Before”, “After”, “Control” and “Impact”); Scenario Specification (which outlines the sampling designs to be examined); Effect Type (multiplicative versus fixed); and Resource Requirements (should the analysis be run in parallel). For more details of each section of the design_specification spreadsheet, please see section 3.2 of Fisher et al. (in review) as this contains a screen shot and associated explanation of the parameters set within each of these five sections, and has not been repeated here.

Once the workbook interface is fully populated, the initial analysis of the pilot data can be run in R by first installing and loading the ‘epower’ package (and any associated dependencies), and typing fitData() into the R console. This will prompt the user to select the workbook file to be analysed, and will also set the working directory to the folder containing that file. This folder is where all outputs generated by the toolbox will be saved, and can be later accessed by the user. Calling fitData() unpacks and processes all of the information contained in the workbook being analysed, sets up the design scenarios to examine, reads in the pilot data, and fits the appropriate generalised mixed model to taht pilot data. Relevant statistical outputs are extracted and formatted such that they can be written to file for the user’s perusal. The statistical information is saved as “…Model_fit_stats.csv”. Also generated is a pdf plot of the PIT values (Dawid 1984), which are plotted as a histogram and saved as “…Pit_histograme.pdf” so the user can evaluate the quality of the model fit. A good model fit is indicated by an even frequency distribution of values (i.e., the PIT histogram should be largely flat).

If the pilot data contains “Before”, “After”, “Control” and “Impact” factor levels, ‘epower’ will fit mixed models with and without the BA*CI interaction term, calculate model probabilities, and return model fit statistics for both models, along with the calculated model probabilities in the file “…Model_fit_stats.csv”. This functionality can be used to test for a significant impact in the supplied data. The model probability represents the probability that the “null” model is preferred (the model without the BACI interaction term, bit instead just the main effects of BA and CI. Thus a value of <0.5 indicates the null model is not preferred (less than 50% probable) and we can infer that the BACI interaction term is ‘significant’ and there has been an “impact”.

A power analysis is carried out by calling assessPower(), which will launch the monte-carlo based power assessment, based on the number of simulations specified, for each the scenarios specified in the “Scenario Specification” section of the design_specification sheet. This can only be done after the fitData() function call is complete, because it requires the posterior sample of the Bayesian model fit to the pilot data to construct the necessary monte-carlo simulations. After the monte-carlo simulations are complete, the proportion of successful detections is combined with the generated scenario matrix, and output as a csv file “…scenario_power_summary.csv” into the working directory folder.

# 4. Step-by-step Instructions

## 4.1 Installing the toolbox and required dependencies

To run the ‘epower’ toolbox, the user must first install R, and then install the ‘epower’ package and associated directories. R can be installed from the CRAN website at: https://cran.r-project.org/. Use the “Precompiled binary distribution” under “Download and Install R” for the appropriate operating system (for example, just click “Download R for Windows”) and follow the prompts, selecting default settings as required. While there are many packages available on the CRAN website, ‘epower’ is not available through CRAN, and must be installed from a local “.gz” file obtained from hppt:\\website.to.be.confirmed.

Before you can install ‘epower’, you must first ensure that dependencies INLA, doParallel and XLConnect are installed in R and functioning correctly, along with any of their dependencies. Packages doParallel and XLConnect are available from the CRAN website (https://cran.r-project.org/web/packages/), and can also be installed by selecting “Install package(s)…” from the “Packages” drop down menu of the RGui. INLA is not available on CRAN and must be installed via the instructions on the INLA website (http://www.r-inla.org/download). The stable version (recommended) can be installed by typing install.packages("INLA", repos=c(getOption("repos"), INLA="https://inla.r-inla-download.org/R/stable"), dep=TRUE) into the R console. You will need to select a CRAN mirror to download the INLA package dependencies. Check that the dependencies are working properly by loading them in R by selecting “Load package…” from the “Packages” drop down menu of the RGui, or typing library(INLA); library(doParallel); library(XLConnect) straight into the R console. Check the R console carefully for any error messages suggesting the packages might not be loaded properly (see 4.4 Troubleshooting) below for known errors sometimes associated with installing XLConnect).

Once all the dependencies are installed and working properly, it is now time to install the ‘epower’ package. To install ‘epower’ copy the “epower_1.3.tar.gz" file to a directory. Open R and select “Install package(s) from local zip files…” from the “Packages” drop down menu of the RGui (Fig 4.1). Use the GUI interface to select the “epower_1.3.tar.gz" file from the directory where it was saved, and follow the prompts. As for the other dependencies, you can check that it is installed properly by typing library(epower) into the R console, or loading it manually using the drop down “Packages” menu. If the package loads correctly no messages will be returned. Once the ‘epower’ package and dependencies are installed and working in R, the user can do a power analysis using their own data by following steps 1 through to 10 (described below in 4.2 Starting a new project and 4.3 Execution in R).

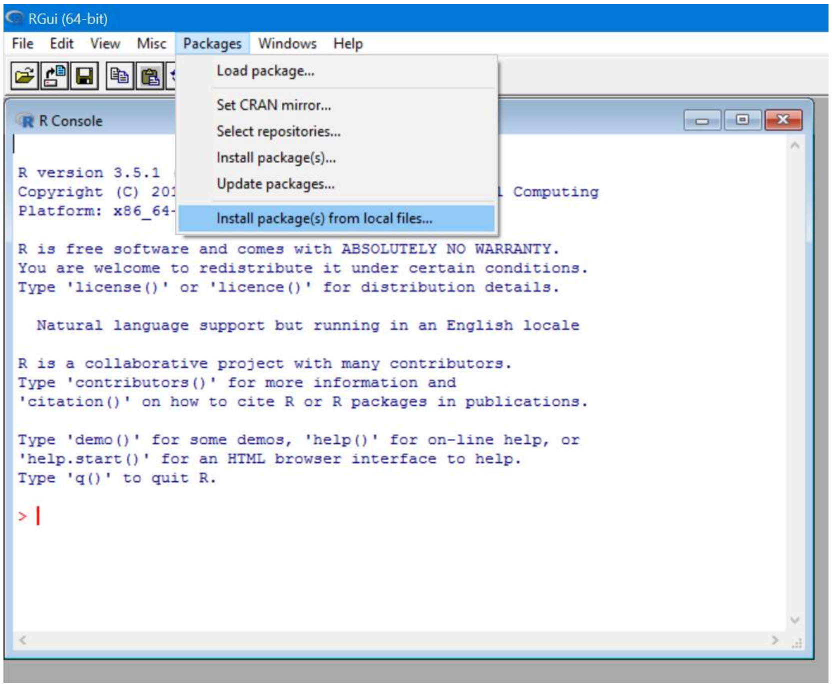



**Figure 4.1 Selecting the "Install packages(s) from local zip files…" option from the "Packages" drop down menu of the RGui.**

## 4.2 Starting a new project

***Step 1:*** To run a new power analysis copy the "epower_interface_V1.3.xlsx" excel workbook interface file into the desired working directory (this will be where all outputs from running 'epower' will also be saved). It is good practice to rename the excel workbook with your project information, as the workbook name will be included in the names of all output files (Fig 4.2). This was done to ensure that, in the event that more than one workbook is to be analysed, the output files can be easily associated with the right input workbook file.

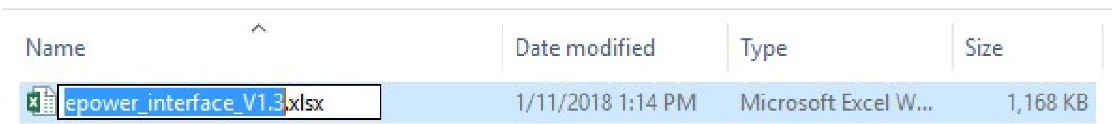
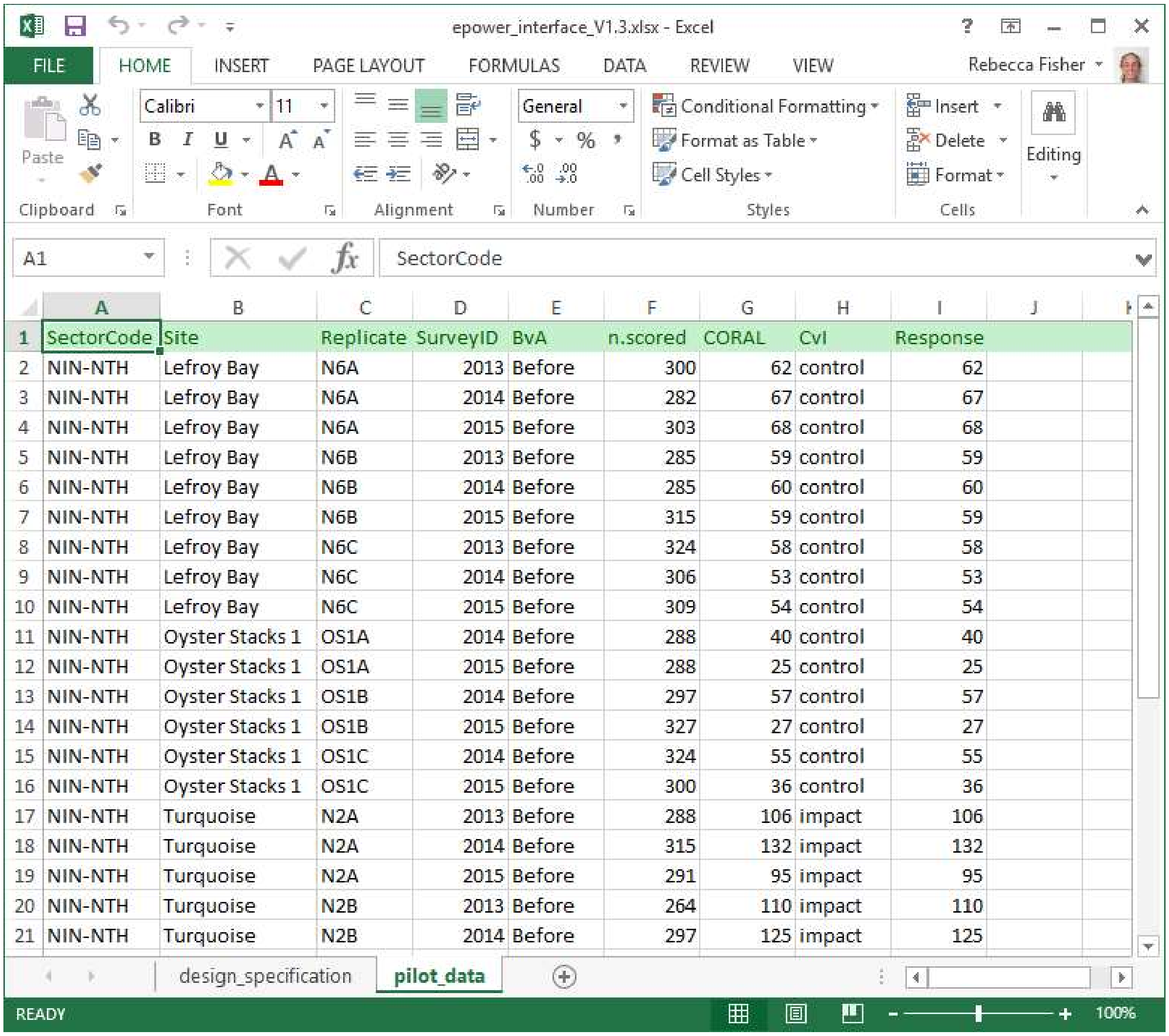



**Figure 4.2 Renaming the "epower_interface_V1.3.xlsx" template file.**

***Step 2:*** Open the (renamed) excel workbook interface and delete all data contained the pilot_data worksheet (Figure 4.3). Do not delete the worksheet itself, as this will disable drop down menu links on the design_specification worksheet. Paste the pilot data to be analysed onto the now empty pilot_data worksheet. Make sure that row 1 of the worksheet contains the column headings of the pilot data (i.e. do not leave any blank rows at the top of the data, and only 1 row of column heading information is allowed, see the existing pilot data for an example). Note that the pilot data must have a column representing the "before" versus "after" status of each row. Often only before data will be available (as is the case in this example), however this pilot_data worksheet will still need a column designating this factor, even though every row may be listed as "before". Note that any column heading is allowed ("BvA", "before.versus.after", "BA"), as well as any indicator of before versus after status ("b" & "a", "before" and "after", 0 & 1). These values are specified on the design_specification worksheet in Step 3, as explained below. A column indicating "control" and "impact" status for each row must also be included on the pilot_data worksheet, and this must have at least one spatial level identified as "impact". In our example we are exploring power to detect an "impact" at the Site called "Turquoise", with sites "Lefroy Bay", "Oyster Stacks 1" and "Winderbandi" used as controls. In this example, it is "Turquiose" for which power will be assessed to detect a decline in hard coral cover associated with a hypothetical impact event. As for the "before" versus “after” any column heading or level indicator can be used, this must just be specified correctly on the design_specification page in Step 3. Note that, in general, when working with R it is better to avoid spaces and other symbols both in header rows as well as cell values, because for various reasons these can sometimes cause errors.



| SectorCode | Site | Replicate | SurveyID | BvA | n.scored | CORAL | CvI | Response |
|---|---|---|---|---|---|---|---|---|
| NIN-NTH | Lefroy Bay | N6A | 2013 | Before | 300 | 62 | control | 62 |
| NIN-NTH | Lefroy Bay | N6A | 2014 | Before | 282 | 67 | control | 67 |
| NIN-NTH | Lefroy Bay | N6A | 2015 | Before | 303 | 68 | control | 68 |
| NIN-NTH | Lefroy Bay | N6B | 2013 | Before | 285 | 59 | control | 59 |
| NIN-NTH | Lefroy Bay | N6B | 2014 | Before | 285 | 60 | control | 60 |
| NIN-NTH | Lefroy Bay | N6B | 2015 | Before | 315 | 59 | control | 59 |
| NIN-NTH | Lefroy Bay | N6C | 2013 | Before | 324 | 58 | control | 58 |
| NIN-NTH | Lefroy Bay | N6C | 2014 | Before | 306 | 53 | control | 53 |
| NIN-NTH | Lefroy Bay | N6C | 2015 | Before | 309 | 54 | control | 54 |
| NIN-NTH | Oyster Stacks 1 | OS1A | 2014 | Before | 288 | 40 | control | 40 |
| NIN-NTH | Oyster Stacks 1 | OS1A | 2015 | Before | 288 | 25 | control | 25 |
| NIN-NTH | Oyster Stacks 1 | OS1B | 2014 | Before | 297 | 57 | control | 57 |
| NIN-NTH | Oyster Stacks 1 | OS1B | 2015 | Before | 327 | 27 | control | 27 |
| NIN-NTH | Oyster Stacks 1 | OS1C | 2014 | Before | 324 | 55 | control | 55 |
| NIN-NTH | Oyster Stacks 1 | OS1C | 2015 | Before | 300 | 36 | control | 36 |
| NIN-NTH | Turquoise | N2A | 2013 | Before | 288 | 106 | impact | 106 |
| NIN-NTH | Turquoise | N2A | 2014 | Before | 315 | 132 | impact | 132 |
| NIN-NTH | Turquoise | N2A | 2015 | Before | 291 | 95 | impact | 95 |
| NIN-NTH | Turquoise | N2B | 2013 | Before | 264 | 110 | impact | 110 |
| NIN-NTH | Turquoise | N2B | 2014 | Before | 297 | 125 | impact | 125 |

**Figure 4.3** Example of pilot data

***Step 3:*** Fill out all user specified cells of the “design_specification” worksheet as instructed. Please read the additional explanatory information contained in section 3.2. “Description, capabilities and usage” of Fisher et al. (in review) for more specific details relevant to each cell. Only cells with fill colours light grey or blue are to be modified (column B, Figure 4.4). Grey cells must contain a single value, whereas light blue cells may contain multiple values (separate by ";" if desired, see Figure 4.4, row 29 for an example). To avoid typing mistakes causing errors (R is very picky in this regard) we have generated drop down menus for many cells based on the column headings contained in the pilot_data worksheet. If the pilot_data worksheet is deleted from the template (rather than just deleting the data itself) these drop down menus will no longer work properly (see Step 2 above).

In the example contained in the supplied template, we select a binomial “Response Distribution” under “Response Type” (Figure 4.4, row 4). A binomial distribution is appropriate as the data are point scores of live coral cover on images collected on replicate transects. Other distributions are of course available (see the Fisher et al. (in review) for more details. As the “Response” under “Design Specification” we select “CORAL” from the drop down menu (Figure 4.4, row 7), as we know that this is the column that contains the value for the number of points scored as live hard coral cover, which is the response or indicator variable being assessed for statistical power in this example. Under “Trials” we select “n.scored” (Figure 4.4, row 8), as this is the total number of points scored on each replicate transect. Only the binomial distribution requires a value here and this can be left blank for other statistical distributions. Our “Location” variable for the example is “Site” (Figure 4.4, row 9) and our sublocation is “Replicate” (Figure 4.4, row 10). This is because we have fixed replicates that are visited repeatedly through time and therefore cannot be considered random (the non-independence of repeated surveys at the same replicate transect must be accounted for in the statistical modelling). In our example the column “SurveyID” represents the “Time” factor in our BACI (Figure 4.4, row 11), and there is no subtime element in this data so this is left blank (Figure 4.4, row 12). The columns representing “BvA” and “CvI” are labelled as “BvA” and “CvI” in our example (Figure 4.4, row 13 and 14). The following four rows (Figure 4.4, rows 17-20) indicate the “levels” or actual cell values of the BvA and CvI indicator columns. Unfortunately these are not available through the drop-down menu (which is only linked to column headings in the pilot_data worksheet and not the column values), and these must be typed in manually. Note that this is case sensitive, and must match exactly the values contained in the corresponding pilot_data worksheet columns. In our example we have “Before” (note the capital “B”), “control” (lower case “c”), and “impact” (lower case “i”, Figure 4.4, rows 17, 18 & 20). Any value can be supplied for “After” in this example, because there are no “After” data anyway. Typos within the design_specification worksheet and/or the pilot_data itself are a common source of error when attempting to run ‘epower’ so please check this carefully before any attempt to run an analysis.

The remaining rows of the “design_specifcation” worksheet relate to the details of the scenario to be examined in the power analysis. The number of simulations to run for the monte-carlo is set on row 25 (“Number of Iterations”, Figure 4.4, row 25). This should be a minimum of 500, but we recommend setting this very low initially (say ~5) to check that everything is working as intended, as the simulations can take some time to run, depending on the size of the dataset being analysed, and the complexity of the scenarios to be examined. The *filename* cell allows the user to specify a name which will be appended to all output files and is optional (Figure 4.4, row 26). By default the ‘epower’ package also includes the name of the excel workbook interface being analysed, thus filename can be left blank unless the user wishes to run different scenarios off a single workbook.

The remaining light blue cells in the “Scenario Specification” section (Figure 4.4, rows 27-34) allow the user to indicate the number of levels to be evaluated for power (or type 1 error if an effect of 0 is defined, see below) for each of the random grouping variables of the sampling design. Different scenarios can be created by assigning multiple values at any level, separated by “;”, which in this example we have done for the “Number of sublocations within Location” (Figure 4.4, row 29). The ‘epower’ will generate all combinations of these to make a complete matrix of possible scenarios. In the current implementation the number of sublocations is assumed to be the same for every Location, and likewise the number of subtimes is assumed the same for every Time. If either sublocation (row 29) or subtime (row 32) are not included in the design, they may be left blank or given a value of "1". As the design in this example is for fixed replicates, the number of replicates (row 34) is assigned a 1. The column heading containing the identity of individual replicates (as measured over time) was already specified in the sublocation cell of the “Design Specification” section (“Replicate”, row 10, described above), and in our example we have indicated we want to assess power for designs with 3 and 4 fixed replicates at this sublocation level (Figure 4.4, row 29).

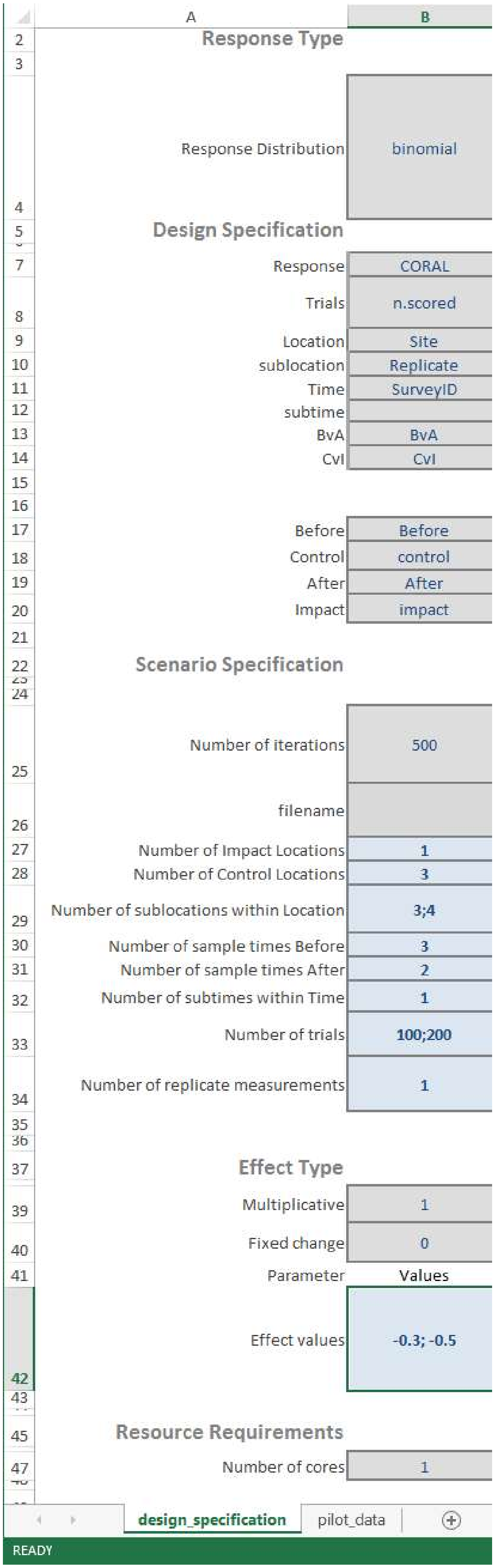



**Figure 4.4** Populating the design_specification worksheet.

Only three more cells remain to be populated (Figure 4.4, rows 39-47). Under “Effect Type” a “1” needs to be indicated next to the type of effect to be examined, multiplicative or a fixed change (see Fisher et al. in review for details). Values for the effect to be examined are supplied on row 42 (Figure 4.4). In our example we ask for a power assessment for a “Multiplicative” effect type of 30% and 50% coral loss respectively (-0.3; -0.5, Figure 4.4, row 42). Finally, we indicate that we only want to use one computer core (Figure 4.4, row 47), so in this example the power analysis will not run in parallel.

It can be quite easy to generate a very large number of scenarios, which may not be realistically run within a sensible time frame, so care must be taken to ensure that the scenario set remains realistic. For example, specifying only two separate values for each light blue cell would yield $2^{18}$ (=262,144) possible scenarios! In our own use of ‘epower’ we tend to run an analysis that represents the existing design first, across a range of effect sizes, and based on that output undertake more detailed analysis varying relevant parts of the design (For example see the case study in Fisher et al, in review).

***Step 4:*** Once all cells have been entered, save and the excel workbook to ensure that the most up-to-date information will be read into R.

## 4.3 Execution in R

***Step 5****:* Open R.

***Step 6:*** Load the ‘epower’ package by selecting “Load package …” from the “Packages” drop down menu in the RGui (Figure 4.1) or by typing library(epower) into the R console (Figure 4.5).

***Step 7***: Call the function fitData, but typing fitData() [including the brackets ()] into the R console (Figure 4.5).. This will open a dialogue box that can be used to select the interface file being used for the analysis (Figure 4.5), and will also set the R working directory to the same directory as that file. Here we have used the excel workbook template “epower_interface_V1.3.xlsx” that has been provided with the toolbox, but the user may change the name of the workbook to be analysed in that session, as suggested in Step 1 under “Starting a new project” above. The name of the excel workbook interface file can also be supplied manually to fitData() if this is preferred for some reason. In this case the working directory must already set in R to that directory containing the workbook to be analysed, or this must also be supplied to fitData() as a second argument “dir.name”.

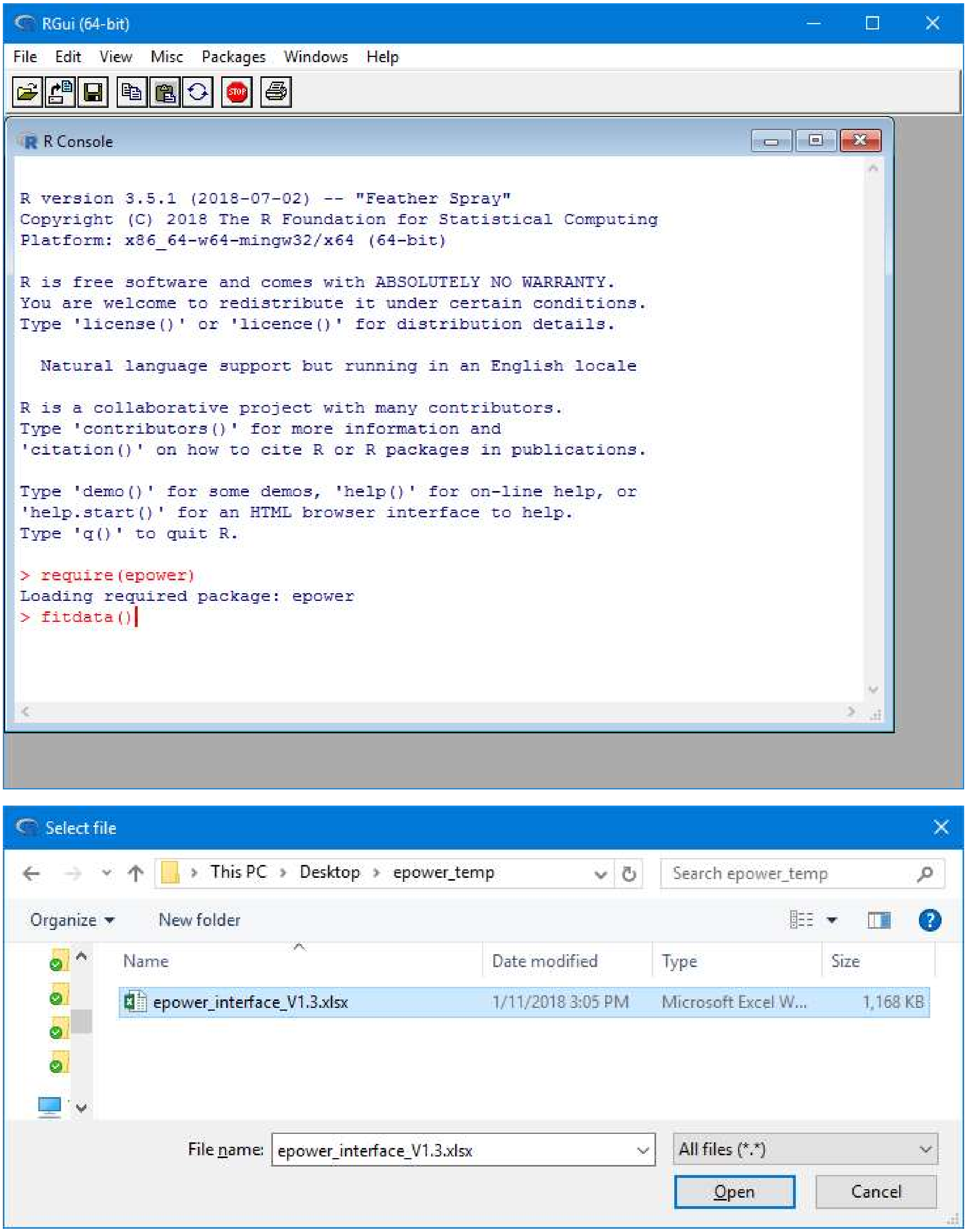



**Figure 4.5 Loading 'epower' and running fitData() from R.** Upper image shows the default R console. Lower image shows the R choose.file interface that allows the user to select the workbook for analysis.

***Step 8***: When R has finished execution of fitData() it is important to examine the output generated. This output will be written to the active working directory, which also contains the excel workbook template that was selected in the analysis (Figure 4.6A). Fit data produces two additional files, both of which will have file names that start with the name of the excel workbook template analysed. The csv generated ("…_Model_fit_statistics.csv") contains the estimated parameter values and other fit statistics for the model, and can easily be viewed in excel (Figure 4.6B). Estimates for the random effects associated with each level present in the hierarchical BACI design are provided as a "precision" (1/variance) following convention in Bayesian analysis. The can be converted to the more familiar estimates of variable by Variance=1/Precision. These variance estimates indicate where the largest variation in the design occurs (smallest values or Precision) and in the example this is for the among Location variance (the column Site in the pilot_data) as well as the Time × Location interaction (There is large variation among Sites at any given SurveyID, and/or large variation among SurveyID's at any given Site). The parameter labels in this model fit output will always match the headings contained in the workbook template, regardless of what actual column headings are used as indicators for these in the pilot_data. Where data are available for "Before" and "After" as well as "Impact" and "Control" locations, this csv file contain model fit statistics for a model with, and another model without, the BA×CI interaction term, as well as corresponding model probabilities. The pdf ("…_Pit_histogram.pdf") generated contains a histogram of the PIT values for the fitted model. A model that fits the data well should have an even frequency distribution of pit values (the histogram should be flat, as is the case for the example shown, Figure 4.6B).

***Step 9:*** Once fitData() has been run, and the user is satisfied that the model epower has fit to the data (based on the information specified in the excel workbook template) is appropriate and adequately fits the observed data (evenly distributed PIT histogram, see above), the user may then run the function assessPower(), simply by typing assessPower() [including the brackets ()], into the R console. No arguments need be passed to this function, as the required inputs are already generated globally in the R workspace by the initial call to fitData(). The assessPower() function can therefore only by run following the completing of a call the fitData().

The function assessPower() will evaluate power for all possible combinations of the parameters provided under "Scenario specification" in the workbook being analysed (Step 3, above). The function returns a single .csv file ("…_scenario_power_summary.csv", see Figure 4.7A), which contains a row for every scenario evaluated (in the example 8 scenarios), the details of the sampling design for that scenario, the effect applied (effect) and the proportion of significant outcomes (sig.outcomes) (Figure 4.7B). The sig.outcomes (column k, Figure 4.7) is simply the number of "significant" outcomes in the simulation as a proportion to the number of iterations. Where no effect is applied for a given scenario (effect = 0), "sig.outcomes" represents type 1 error (also known as alpha, α), whereas if an effect was applied, "sig.outcomes" indicates power (1 - β). As for the model fit statistics file, the column headings in this power summary file correspond to the headings contained in the workbook template, regardless of what actual column headings are used as indicators for these in the supplied pilot_data. The user can refer to their populated workbook to remind themselves what each column (Location, sublocation, Times, subtimes, replicates and trials, Figure 4.7B) represents in the context of their own design.

The assessPower() function also automatically saves an R Workspace file into the working directory, which includes the R objects "scen.out", "scenarioParams", "dataComponents", and "scenario.matrix" (see Fisher et al. in review). This was added so that users familiar with R can work with the analysis inputs and outputs directly. The simulated data for each scenario is saved in the scen.out list, which will have as many elements as there were scenarios (8 in the example) with each element containing the model probabilities for each simulation (scen.out[[1]]$model.probs) and the corresponding simulated dataset (scen.out[[1]]$ sim.factor.data). The R Workspace file can be quite large (>2 gb in the example Figure 4.7A) and it can be safely deleted if not required by the user.

A.

| Name | Date modified | Type | Size |
|---|---|---|---|
| epower_interface_V1.3.xlsx | 1/11/2018 3:05 PM | Microsoft Excel W... | 1,168 KB |
| epower_interface_V1.3__Model_fit_stats.csv | 1/11/2018 3:02 PM | Microsoft Excel C... | 2 KB |
| epower_interface_V1.3__Pit_histogram.pdf | 1/11/2018 3:02 PM | Adobe Acrobat D... | 5 KB |

B.

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | Model | Parameter | mean | sd | X0.025quant | X0.5quant | X0.975quant | mode |
| 2 | Fitted model | BvAxCvIBefore - Control | -1.3946 | 0.2291 | -1.8703 | -1.3906 | -0.9396 | -1.3839 |
| 3 | | BvAxCvIBefore - Impact | -0.4656 | 0.3919 | -1.2657 | -0.4656 | 0.3353 | -0.4656 |
| 4 | | Precision for Location | 15.6055 | 16.8459 | 1.9411 | 10.6016 | 59.7292 | 5.0068 |
| 5 | | Precision for Time x Location | 92.864 | 90.6477 | 12.751 | 66.5164 | 331.2582 | 33.197 |
| 6 | | Precision for sublocation | 136.4205 | 134.3898 | 19.9353 | 97.0918 | 491.3411 | 50.4791 |
| 7 | | Precision for Time by sublocation | 18546.5957 | 18377.739 | 1252.9841 | 13112.2261 | 67137.7753 | 3416.8565 |
| 8 | | Precision for Time | 18439.4255 | 18300.9864 | 1233.4954 | 13020.7848 | 66887.2301 | 3354.4268 |
| 9 | | Precision for repID | 18555.9802 | 18358.6143 | 1240.806 | 13125.99 | 67142.3704 | 3375.0524 |
| 10 | | | | | | | | |
| 11 | dic | p.eff | mean.deviance | deviance.mean | family.dic | family.p.eff | waic | waic.p.eff |
| 12 | 248.7355 | 16.6192 | 232.1163 | 215.4971 | 248.7355 | 16.6192 | 254.2704 | 16.8298 |

C.

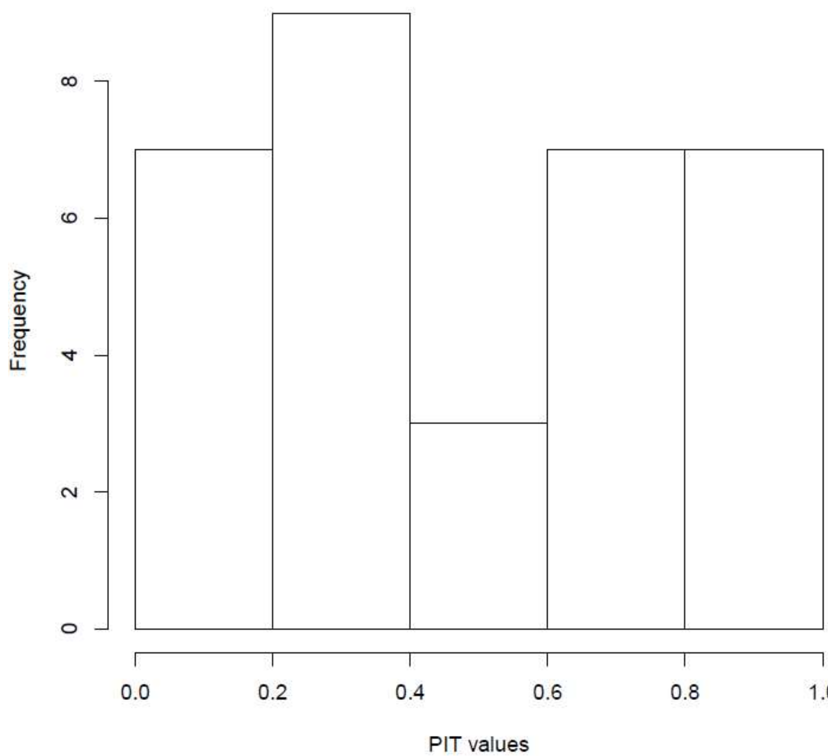



**Figure 4.6** Output files generated by function fitData(), including a screenshot showing the output files in the active working directory folder (A), an example of the model fit statistics that are returned (B), and an example PIT histogram for evaluating model fit (C).

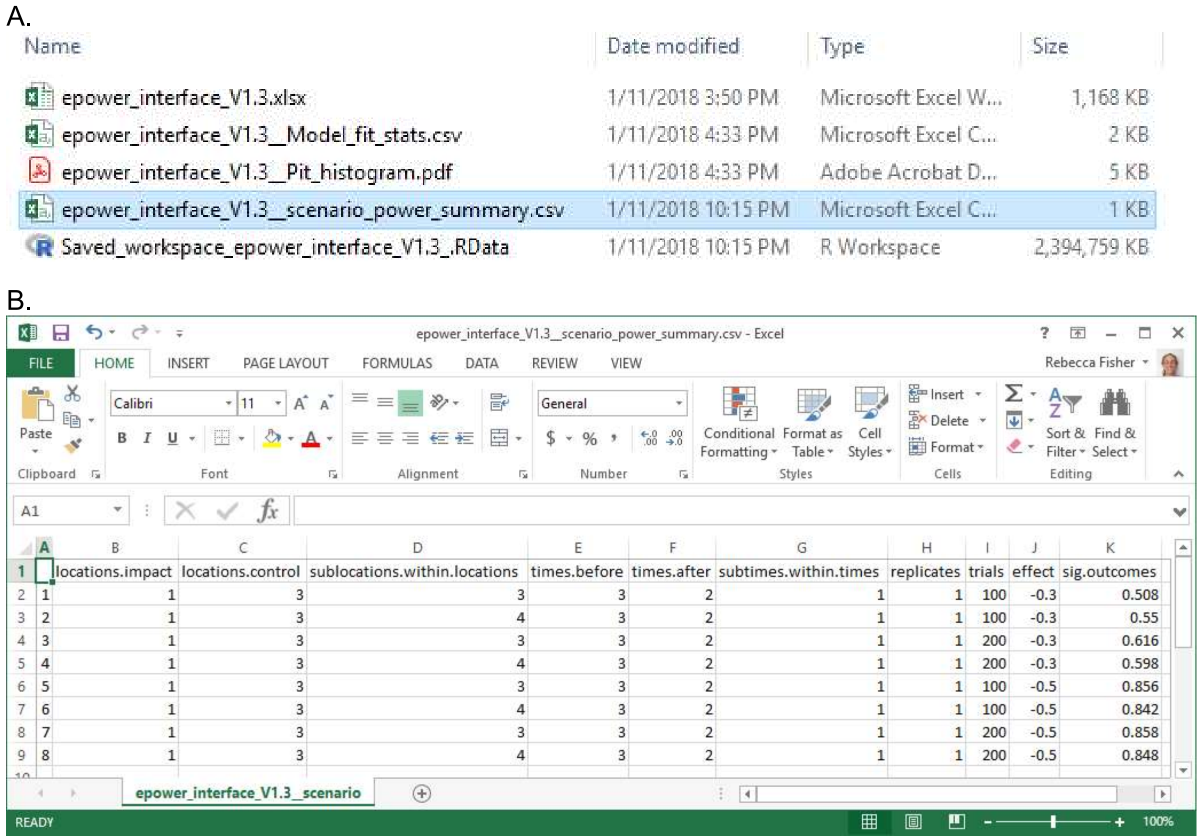



**Figure 4.7** Example output from the function assessPower(), showing the additional file as it appears in the current working directory (A) and the output .csv table generated (B).

***Step 10***: When R has finished running the simulations, close R. There is no need to save the workspace. Note that it can be a good idea to save your simulation runs to a separate folder, as the files will be over-written by R without a warning if you want to run another analysis again using a workbook template populated differently, but with the same workbook name, if it remains in the same location.

## 4.4 Troubleshooting

Errors loading the R packages

- There can be issues with installing XLConnect in R, as this requires rJava, which can sometimes have trouble interacting with the appropriate installed version of java on the machine. This is most common on machines that support both 32- and 64-bit versions. A helpful troubleshooting guide can be found at: https://www.r-statistics.com/2012/08/how-to-load-the-rjava-package-after-the-error-java_home-cannot-be-determined-from-the-registry/

If errors occur during the R execution step of ‘epower’, try the following things:

- Check all cells in the Excel workbook, in particular the “Design Specification” section of the design_specification worksheet. Ensure that the names of the factor variables in the analysis match those used in the pilot data supplied on the appropriate worksheet, and that the labels for the “Before”, “After”, “Impact” and Control” factor levels are correct, spelled exactly the same (R is case sensitive) and there are no other variations of the labels contained within the data supplied in pilot_data. It can be common for typographical errors to occur, and these can be hard to spot, for example, if there is a blank space on the leading or trailing edge of a value (ie "Impact" versus " Impact").

- Note that warning messages may sometimes appear in the R console, but these are generally not a concern providing the output files appear sensible.
- Occasionally INLA.exe may crash during execution of a simulation. If this happens an error message will appear and the user will have to close a dialogue window. Once this has been closed the analysis should continue without further input from the user, although it is possible an additional crash may occur requiring this step to be repeated more than once.

# 5. Availability, licensing and citing

The 'epower' (V1.3) package is a BMT product which has been developed in collaboration with the Australian Institute of Marine Science (AIMS), Queensland University of Technology (QUT) and Pink Lake Analytics. The method/code supporting 'epower' v1.3 is described in "Fisher R, Shiell GR, Sadler RJ, Inostroza K, Shedrawi G, Holmes TH, McGree JM (in review). 'epower': an R package for power analysis of Before-After-Control-Impact (BACI) designs. Environmental Modelling and Software.

'epower' is freely available to users under licence. Licence for use explicitly precludes the incorporation of the 'epower' code into other programs which are subsequently sold or used for commercial purposes.

Users are asked to please include the following acknowledgment: "Analyses were carried out using the software 'epower' V1.3 (BMT 2018) as described in Fisher et al (2018), (BMT 2018) and based on the statistical programming platform R (R-Core Team, 2018).

# 6. References


- Anderson MJ, Gorley RN, Clarke KR (2008). PERMANOVA+ for PRIMER: Guide to Software and Statistical Methods. PRIMER-E: Plymouyh, UK.
- ANZECC/ARMCANZ (2001) Australian and New Zealand guidelines for fresh and marine waters. Australian and New Zealand Environment and Conservation Council & Agriculture Resource Management Council of Australia and New Zealand, Canberra.
- Bates DM (2010) lme4: Mixed-Effects Modeling with R. Springer, New York
- Burnham KP, Anderson DR (2002) Model Selection and Multimodel Inference; A Practical Information-Theoretic Approach. Springer, New York
- Dawid AP (1984) Present Position and Potential Developments: Some Personal Views: Statistical Theory: The Prequential Approach. Journal of the Royal Statistical Society Series A (General) 147:278-292
- Fisher R, Shiell GR, Sadler RJ, Inostroza K, Shedrawi G, Holmes TH, McGree JM (in review) 'epower': an R package for power analysis of Before-After-Control-Impact (BACI) designs. Environmental Modelling and Software
- Rue H, Martino S, Chopin N (2009) Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 71:319-392
- Searle SR (1971) Linear Models. Wiley, New York
- Underwood A (1991) Beyond BACI: Experimental designs for detecting human environmental impacts on temporal variations in natural populations. Marine and Freshwater Research 42:569-587

- Underwood AJ (1992) Beyond BACI: the detection of environmental impacts on populations in the real, but variable, world. Journal of experimental marine biology and ecology 161:145-178
- Underwood AJ (1993) The mechanics of spatially replicated sampling programmes to detect environmental impacts in a variable world. Australian Journal of Ecology 18:99-116
- Underwood AJ, Chapman MG (2003) Power, precaution, Type II error and sampling design in assessment of environmental impacts. Journal of Experimental Marine Biology and Ecology 296:49–70
- Rue H., Martino S. and Chopin N.: Approximate Bayesian inference for Latent Gaussian Models Using Integrated Nested Laplace Approximations (with discussion). Journal of the Royal Statistical Society, Series B, 71, 319–392
- Zuur AE, Ieno EN, Walker NJ, Saveliev AA, Smith GM (2009) Mixed Effects Models and Extensions in Ecology with R. Springer-Verlag New York Inc., New York/US