A Synthetic Case

Data Worth

A GMDSI tutorial

by Rui Hugman and John Doherty

gmdsi
Groundwater Modelling Decision Support Initiative

BHP Flinders University NATIONAL CENTRE FOR GROUNDWATER RESEARCH AND TRAINING RioTinto

PUBLISHED BY

The National Centre for Groundwater Research and Training
C/O Flinders University
GPO Box 2100
Adelaide SA 5001
+61 8 8201 2193

PREFACE

The Groundwater Modelling Decision Support Initiative (GMDSI) is an industry-funded and industry-aligned project focused on improving the role that groundwater modelling plays in supporting environmental management and decision-making.

Over the life of the project, GMDSI will produce a suite of tutorials. These are intended to assist modellers in setting up and using model-partner software in ways that support the decision-support imperatives of data assimilation and uncertainty quantification. Not only will they focus on software usage details. They will also suggest ways in which the ideas behind the software which they demonstrate can be put into practice in everyday, real-world modelling.

GMDSI tutorials are designed to be modular and independent of each other. Each tutorial addresses its own specific modelling topic. Hence there is no need to work through them in a pre-ordained sequence. That being said, they also complement each other. Many employ variations of the same synthetic case and are based on the same simulator (MODFLOW 6). Utility software from the PEST suite is used extensively to assist in model parameterization, objective function definition and general PEST/PEST++ setup. Some tutorials focus on the use of PEST and PEST++, while others focus on ancillary issues such as introducing transient recharge to a groundwater model and visualization of a model's grid, parameterization, and calculated states.

The authors of GMDSI tutorials do not claim that the workflows and methodologies that are described in these tutorials comprise the best approach to decision-support modelling. Their desire is to introduce modellers, and those who are interested in modelling, to concepts and tools that can improve the role that simulation plays in decision-support. Meanwhile, the workflows attempt to demonstrate the innovative and practical use of widely available, public domain and commonly used software in ways that do not require extensive modelling experience nor an extensive modelling skillset. However, users who are adept at programming can readily extend the workflows for more creative deployment in their own modelling contexts.

We thank and acknowledge our collaborators, and GMDSI project funders, for making these tutorials possible.

Rui Hugman

John Doherty

CONTENTS

1. INTRODUCTION

This document is one of series of tutorials which demonstrates workflows for parameter estimation and uncertainty analysis with PEST/PEST++. The workflows which they demonstrate are not the only (nor necessarily the best) options; the purpose of the tutorials is to take the reader through the basics of how to accomplish common tasks.

The present tutorial addresses the ability (or otherwise) of yet-ungathered data to reduce the uncertainties of decision-critical predictions using linear analysis utilities from the PEST suite. Data worth analysis is applied to a model which was calibrated in another GMDSI tutorial; this model was subjected to a suite of common linear analysis tasks in yet another tutorial in this series.

The PEST roadmaps are a useful complement to this series of tutorials. Perusal of **Roadmap 12: Linear Analysis** prior to commencing this tutorial is recommended. This roadmap provides a brief discussion of concepts and theory; it also provides an outline of steps required to undertake linear analysis. The present tutorial focuses on practical implementation of concepts discussed in the Roadmap. The abovementioned GMDSI tutorial on model calibration provides details of the model that is employed in the present tutorial, and of the PEST configuration on which the current tutorial is based. A brief summary is provided in Chapter 1.1.

This tutorial assumes that you are at least partially familiar with PEST and the PEST software suite. Although completion of the previous tutorials is not a requirement, this is recommended if you are not familiar with PEST.

Completed versions of all files generated during this tutorial can be found in a folder named *completed*. These are useful if you need to troubleshoot your own files, or if you wish to jump into the tutorial at a specific point. That being said, it is recommended that you at least read through the parts of this tutorial that you do not complete yourself.

Software Executables

A number of programs are used throughout this tutorial. To make completion of this tutorial easier, executable versions of these programs have been placed in relevant folders; these are **.exe* files. This is generally not recommended practice, for data files and executable files should be kept separate! Preferably, executable files (i.e. programs) should be located in a folder that is cited in your computer's PATH environment variable. Doing this allows them to be run from a command prompt that is open to any other folder without including the full path to these executables in the command to run them.

1.1 Background

The case being modelled is entirely synthetic. It is the authors' experience that building synthetic models that look like "the real thing" never really works well. However the synthetic nature of a didactic model does not detract from the lessons that it teaches. Some design decisions have been taken to highlight the application of specific tools or techniques at the expense of sometimes making the case "un-realistic". So please bear with us, dear reader, if sometimes things look a bit weird!

Our imaginary site is illustrated in Figure 1. Several streams traverse the area, flowing from west to east. These streams are perennial. Stream flow is maintained by groundwater. Let us imagine that these streams are one of the last remaining habitats of a particular species of frog which needs an aquatic environment all year round. Let us accept that it is in humanity's best interest to keep this species of frog alive.

The Prediction

The nearby town of Makebelievesville needs to expand its water supply through groundwater abstraction. Land access and existing infrastructure limit possible sites for a wellfield. Of these, the preferred site is close to one of the streams; see the red star in Figure 1. Proposed abstraction rates are constant at 2,000 m^3/d. Proximity of groundwater extraction to the stream raises concerns that streamflow may be depleted. After extensive research, ecologists have determined that the frog species requires a minimum stream flow rate of 6,500 m^3/d to survive. If proposed abstraction reduces groundwater stream discharge to less than this threshold "a bad thing" happens (and humanity will suffer).

A numerical model was developed to assess whether proposed abstraction will reduce groundwater discharge to the streams below the critical threshold of 6,500 m^3/d under steady state pumping conditions. Two steady state stress periods are simulated. The first represents historical conditions. The second simulates proposed additional abstraction.

History-Matching

Model parameters were calibrated against hydraulic head and stream flow measurements. Values of recharge rate, hydraulic conductivities in each model layer, and conductances of boundary conditions were parameterized using pilot points. Parameter uniqueness was achieved through Tikhonov regularisation. Covariance matrices were used instead of weights in prior information equations that specified preferred parameter values, in order to "spread out" emergent hydraulic property heterogeneity.

The model prediction of interest is simulated groundwater discharge into the upper stream reach during the second stress period. This observation is named *p-trib_1* in the PEST control file. The calibrated parameter set results in a predicted discharge of 6,899 m^3/d to the stream once additional abstraction is in place.

Linear Uncertainty Analysis

Outcomes of linear analysis indicate that the posterior uncertainty of the prediction is still too large to exclude the possibility of the bad-thing happening. Additional data collection is being considered to further constrain the uncertainty of the prediction. Before proceeding with expensive drilling and data-acquisition, the worth of several possible new observation wells will be assessed.

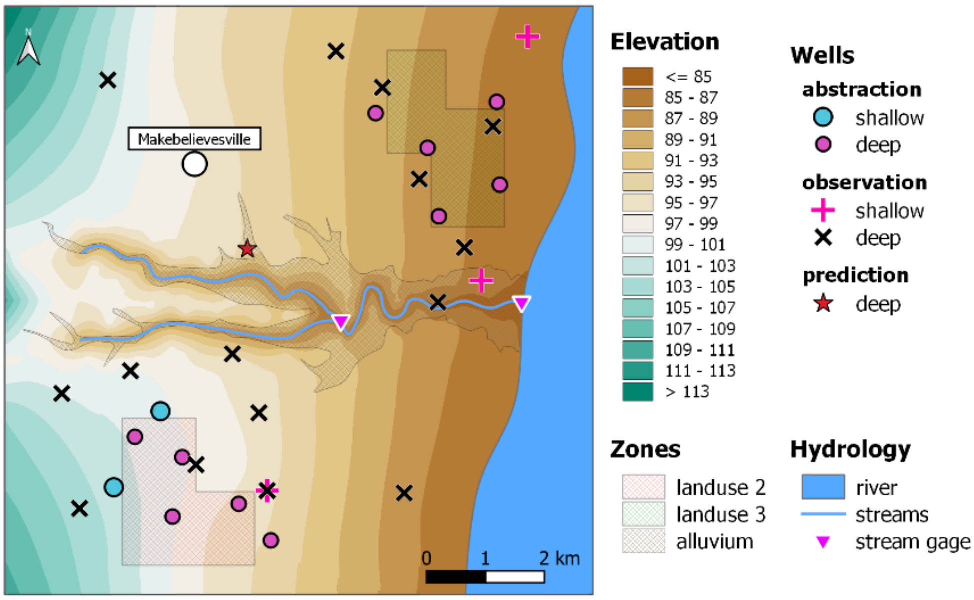

Figure 1 – Layout of the synthetic case and main features of interest.

2. LINEAR ANALYSIS (A SHORT RECAP)

The current Chapter serves as a brief reminder of topics covered in the GMDSI linear analysis tutorial. The current tutorial on assessing data worth continues where the abovementioned tutorial left off.

Linear uncertainty analysis is also known as "first order second moment" (or "FOSM") analysis. It provides approximate mathematical characterisation of prior predictive probability distributions, and of posterior parameter and predictive probability distributions. It has other uses as well. It can be used to demonstrate how the history-matching process bestows worth on data. It can also be deployed to track the flow of information from field measurements of system state to parameters, and ultimately from parameters to model predictions. It does all of these things by implementing Bayes equation under the following assumptions.

- The prior probability distribution of parameters is multiGaussian.
- "Measurement noise" (including structural noise) is also characterized by a Gaussian probability distribution.
- The relationships between model outputs that correspond to measurements of system state and parameters employed by a model can be approximated by the action of a matrix on a vector.
- Model outputs that correspond to predictions of management interest can be calculated using another matrix that acts on model parameters.

Ideally, linear analysis is undertaken after a model has been calibrated. However, if a model is truly linear (which it never is), the outcomes of linear analysis are independent of parameter values; they can therefore, in theory, be applied with a model that is endowed with user-supplied prior mean parameter values.

If a model has undergone calibration, then minimum-error variance (i.e. calibrated) parameter values should be assigned to parameters as their initial values in the "parameter data" section of the PEST control file on which linear analysis is based. The Jacobian matrix should be calculated using these parameters. If the uncertainty of a prediction is being examined, then the model output that pertains to this prediction must be included as a (zero-weighted) "observation" in the PEST input dataset; sensitivities of this model output to model parameters will therefore appear in the Jacobian matrix. Alternatively, a separate prediction-only Jacobian matrix can be built based on a PEST control that is dedicated to this purpose.

2.1 Getting Ready for Linear Analysis

Three files are needed prior to undertaking linear analysis with the PEST suite. These are as follows:

1) A PEST control file in which:
   a) the initial values of parameter are their calibrated values (this is easily achieved using the PARREP utility);
   b) all regularisation is removed (this can be done using the SUBREG1 utility); and
   c) observation weights are equal to the inverse of the standard deviation of measurement noise (this can be accomplished using PWTADJ2).
2) A Jacobian matrix (JCO file) (obtained by running PEST with NOPTMAX set to -2 in the PEST control file, or by using the JCO2JCO utility to modify an existing JCO file);
3) A "parameter uncertainty file" in which prior parameter uncertainties are recorded (prepared by the user).

See the abovementioned linear analysis tutorial for step-by-step instructions on how to build these files.

2.2 Running Linear Analysis Utilities

A suite of utility programs whose names begin with "PREDUNC" are included in the PEST suite. ("PREDUNC" stands for "PREDictive UNCertainty".) A complementary suite of utilities have names that begin with "PREDVAR". These explore parameter and predictive *error* rather than *uncertainty.* In general, programs from the PREDVAR suite are not as useful as those from the PREDUNC suite. Programs belonging to both these suites are described in detail in Part 2 of the PEST manual.

GENLINPRED and GENLINPRED_ABBREV run utilities from the PREDNUC and PREDVAR suites in the background, automating file-preparation and keyboard response tasks that normally accompany solitary use of these programs. In the linear analysis tutorial the GENLINPRED_ABBREV utility was used to undertake a default set of tasks. In the current tutorial we will be using the PREDUNC5 utility directly.

3. DATA WORTH

The worth of data is measured by their ability to reduce the uncertainties of model predictions that we care about. Linear analysis is particularly useful for exploring data worth. This is because the equations that it uses to calculate predictive uncertainty do not include terms that represent the actual values of observations or of parameters; only sensitivities of model outputs to parameters are required. Therefore, linear analysis can be used to assess the ability (or otherwise) of yet-ungathered data to reduce the uncertainties of decision-critical predictions.

This means that potential field measurements that correspond to one or many outputs of a model can be assessed for their worth. For example, it is possible to assess the worth of observations of head in every single model cell at every time step of a model run with a relatively small computational burden. This makes linear analysis a useful tool for designing and comparing strategies for data-collection, when data acquisition seeks to reduce the uncertainties of one or a number of decision-critical predictions.

PREDUNC5, a utility from the PEST suite, is used for analysing the effect of individual observations, or combinations of observations, on the uncertainty of a prediction. It has two modes of operation; these are as follows:

1) It allows ranking of the relative worth of existing observations by calculating predictive uncertainty with selected individual or combined observations removed from a calibration dataset.
2) It allows ranking of the relative worth of new observations by calculating predictive uncertainty with selected individual or combined observations added to an existing calibration dataset.

This tutorial focuses on the latter, “observation addition” mode of PREDUNC5 usage. The “observation addition” option is useful for assessing the relative worth of data acquisition strategies that are presently under consideration. All potential additional observations must be included in a PEST control file, along with the existing calibration dataset; the latter comprise the “base” observation type, as will be described shortly. PREDUNC5 then evaluates the uncertainty of a user-specified prediction with different observation sets in turn added to the set of “base” observations. It is then possible to compare the relative efficacy of these observation sets in lowering the uncertainty of the prediction of interest. Note that a given observation can appear in more than one observation set; hence it is possible to compare different combinations of individual observations in the same analysis.

3.1 Getting Ready to run PREDUNC5

If PREDUNC5 is used to assess the worth of new observations, these must be listed in the PEST control file on which linear analysis is to be based. The Jacobian matrix that is calculated using this PEST control file will thereby include the sensitivities of these new observations to all parameters.

Apart from the prerequisites listed in Chapter 2.1, PREDUNC5 requires two further inputs. These are as follows:

1) A prediction sensitivity file; this can be obtained from a Jacobian matrix using the JROW2VEC utility.
2) An “observation type list” file; this is prepared by the user.

Updated PEST Dataset and Jacobian Matrix

When undertaking history-matching, observations are “cheap”, as the inclusion of additional observations incurs little additional computational cost. In principle, there is nothing stopping a modeller from “observing” the field measured counterpart of every conceivable output generated by a model in a PEST calibration dataset (apart from computer memory and the effort involved in building such a PEST control file). If modelling is undertaken with data worth analysis as among its goals, then it makes sense to include observations whose worth are to be assessed from the get-go; these can be assigned arbitrary values and weights of zero (which can be re-assigned later for linear analysis) so that they do not influence history-matching. In this manner, sensitivities of these observations are calculated as part of the normal history-matching process; this obviates the task of calculating their sensitivities later on.

Alternatively, if observations were not included in the PEST control file to begin with, then the Jacobian matrix must be recalculated with the new observations added to the PEST input dataset. The values assigned to these observations can be arbitrary. However, their weights should be the inverse of the standard deviation of noise that is likely to be associated with corresponding field measurements.

In our current case, potential new observations were not included in the PEST control file when it was used as a basis for history-matching. After history-matching (and after preliminary linear analysis), potential new observation sites were identified and added to the PEST input dataset. These are comprised of nine new boreholes disposed on a 250x250m grid around the abstraction well. Each of these posited wells can measure heads separately in both layer 1 and layer 3. (Ideally, head difference observations should be included in the potentially updated calibration dataset; however this was not done for reasons of author laziness). A weight of 10 was associated with each observation. This value is the inverse of an assumed standard deviation of measurement noise equal to 0.1 m. The updated PEST dataset and model files have been provided for you.

1 If you inspect the tutorial directory you will see two folders; these are named *pest-dataworth* and *runmodel*. (The directory structure of these folders is described in the GMDSI Calibration tutorial.) The *runmodel* folder contains all model files, as well as those required by model pre- and postprocessing utilities. The *pest-dataworth* folder contains files that pertain to the PEST dataset. (If you completed the linear analysis tutorial you might recognise that these files correspond to the completed version of that tutorial).

2 In the *pest-dataworth* folder you will find a PEST control file named *dataworth.pst* along with several PEST output files. See, in particular, the corresponding Jacobian matrix file named *dataworth.jco*. The latter file was obtained by running PEST based on *dataworth.pst* with the NOPTMAX control variable set to -2.

3 Open *dataworth.pst* in a text editor to inspect it. In the “observation data” section of this file you will see 18 observations which are associated to an observation group named *newobs.* (These are shown in the text box below). The observation name prefix “h1” or “h3” indicates whether they are located in layer 1 or in layer 3.

```
* observation data
(..)
h1-2045   0     10    newobs
h1-1712   0     10    newobs
h1-1502   0     10    newobs
h1-1291   0     10    newobs
h1-1137   0     10    newobs
h1-962    0     10    newobs
h1-888    0     10    newobs
h1-821    0     10    newobs
h1-760    0     10    newobs
h3-2045   0     10    newobs
h3-1712   0     10    newobs
h3-1502   0     10    newobs
h3-1291   0     10    newobs
h3-1137   0     10    newobs
h3-962    0     10    newobs
h3-888    0     10    newobs
h3-821    0     10    newobs
h3-760    0     10    newobs
```

Prediction Sensitivity Matrix

Once the Jacobian matrix file has been calculated, use of the JROW2VEC utility makes acquisition of a prediction sensitivity matrix a simple matter.

4 Open a command line window in the *pest-dataworth* folder. Then type in the following command to run JROW2VEC:

```
jrow2vec dataworth.jco p-trib_1 ptrib.vec
```

5 This command instructs JROW2VEC to (1) read the Jacobian matrix file named *dataworth.jco*, (2) extract the sensitivity vector that pertains to the observation named *p-trib_1* (this is our model prediction of interest), and (3) record the vector in a dedicated file named *ptrib.vec.* The latter is the prediction sensitivity matrix file which PREDUNC5 requires.

Observation Type List File

The last PREDUNC5 input that we need to prepare is the “observation type list” file. PREDUNC5 allows observations to be grouped into “types” for addition to, or subtraction from, an existing observation dataset (usually a history-matching dataset). Observations are grouped into “types” in (yes, you guessed it) the “observation type list” file.

An “observation type list” file is a text file comprised of a list of observation names. All observations listed in the PEST control file must be listed in the “observation type list” file, and vice versa. Observation “type” delimiters are inserted into this list to define sets of observations. Observations can be repeated in multiple observation “types”. In this manner it is possible to assess the influence of different combinations of observations on the uncertainty of a prediction.

6 Create a new blank text file in the *pest-dataworth* folder and name it *obstype.dat*.

7 We begin by defining the “base” observation type. This “type” is comprised of the set of observations against which the relative worth of new data will be compared. Usually these are the set of observations that were used for history-matching. Identify the beginning of the list of observations that will be included in the “*base*” type by typing in the following delimiter text. Any observation listed between this delimiter and the next delimiter will be included in the “*base*” type.

```
* observation type "base"
```

8 Now we can introduce the observation names into the "*base*" type. From the list of observation names in the *dataworth.pst* PEST control file, copy those which pertain to observation groups *head1, head3, streams* and *headiff*. (These observations groups comprise the history-matching dataset). Paste the list into *obstype.dat*. Your file should now look like this:

```
* observation type "base"
h1-3892_1
h1-1664_1
h1-162_1
h3-3521_1
h3-3988_1
h3-3417_1
h3-3829_1
h3-3357_1
h3-3594_1
h3-3892_1
h3-1152_1
h3-313_1
h3-209_1
h3-331_1
h3-707_1
h3-3956_1
h3-2270_1
h3-521_1
trib_1
total_1
dh3892_1
```

9 Next, let us create an observation type named "*forecast*". For the purposes of the current tutorial this is not actually relevant, however it keeps things tidy.

10 Add in a new delimiter to define the beginning of a new observation type named "*forecast*" as follows. (The new line is indicated in **bold** in the figure below; the first line of this figure "*(…)*" indicates that part of the file is not shown.)

```
(…)
h3-521_1
trib_1
total_1
dh3892_1
* observation type "forecast"
```

11 Copy and paste the observation names pertaining to model forecasts (observations from the observation group *pstreams*) from the PEST control file into *osbtype.dat.* The new lines in your file should look like this:

```
(…)
h3-521_1
trib_1
total_1
dh3892_1
* observation type "forecast"
p-trib_1
p-total_1
```

12 Finally, let us add a single observation type for all the potential observations named "*all*". The new lines in your file should look like this:

```
(…)
* observation type "forecast"
p-trib_1
p-total_1
* observation type "all"
h1-2045
h1-1712
h1-1502
h1-1291
h1-1137
h1-962
h1-888
h1-821
h1-760
h3-2045
h3-1712
h3-1502
h3-1291
h3-1137
h3-962
h3-888
h3-821
h3-760
```

Now we can run PREDUNC5. Let us give it a spin.

3.2 Running PREDUNC5

13 In the command line window (open a new one if you closed it), type the following, and then press <enter>:

```
i64predunc5
```

14 You will be prompted for the name of a PEST control file. Reply with *dataworth.pst* as shown below and then press <enter>:

```
Enter name of PEST control file: dataworth.pst
```

15 Next PREDUNC7 will ask you for the value of the observation reference variance. As observation weights in the PEST control file already represent the inverse of the standard deviation of measurement noise, respond to this prompt with **1**, and then press <enter>.

```
Enter observation reference variance: 1
```

16 At the next prompt, provide the name of the prior parameter uncertainty file (*param.unc*) which has been provided for you, and then press <enter>. (See the GMDSI linear analysis tutorial for construction details of this type of file.)

```
Enter name of parameter uncertainty file: param.unc
```

17 Next, provide the name of the sensitivity matrix file that we constructed in step 4, and then press <enter>.

```
Enter name of predictive sensitivity matrix file: ptrib.vec
```

18 This is followed by the name of the observation type list file that we constructed in steps 6 to 12; then press <enter>.:

```
Enter name of observation type list file: obstype.dat
```

19 You will be asked whether you wish to subtract or add observation types to the base type. For the purposes of this tutorial we are interested in adding. Respond accordingly by typing **a**, and then press <enter>.:

```
Subtract list members or add to base observations [s/a]: a
```

20 At the next prompt, you are asked to provide the name of the file to which PREDUNC5 should write results. Respond with *predunc5-add.out*, and then press <enter>**.**.

```
Enter name for predictive uncertainty output file: predunc5-add.out
```

21 Finally, you will be asked which equation you wish PREDUNC5 to use. We have a lot less observations than parameters, so respond with ***2***, and then press <enter>.

```
Use which version of linear predictive uncertainty equation:-
     if version optimized for small number of parameters   - enter 1
     if version optimized for small number of observations - enter 2
 Enter your choice: 2
```

22 At this point PREDUNC5 will get to work. As it does so, it will write output to the screen to inform you of what is going on. When it is finished (which should take less than a minute), you will see the following in the command line window if all went well:

```
- predictive uncertainty contribution file predunc5-add.out written ok.
```

23 Inspect the *pest-dataworth* folder. You should see a new file named *predunc5-add.out*. Open it in a text editor. You should see the following:

```
Name of prediction = "p-trib_1"

Added_obs_type        Precal_variance     Postcal_variance    Precal_uncertainty  Postcal_uncertainty
    base                  5304716.            28908.83            2303.197            170.0260
    forecast              5304716.            28908.83            2303.197            170.0260
    all                   5304716.            15252.72            2303.197            123.5019
```

PREDUNC5 has recorded the uncertainty variance and standard deviation of the *p-trib_1* prediction when each observation “type” listed in *obstype.dat* is added to the “*base*” set of observations. As the “*base*” type is comprised of observations used for history-matching, this shows the potential reduction in prediction uncertainty that is accrued if these new “types” were to be included in the history-matching dataset. As is to be expected, the “*forecast*” type does not reduce uncertainty (forecast observations were assigned a weight of zero). Fortunately the “*all*” type reduces uncertainty from 170.0 m^3/d to 123.5 m^3/d. This means that if all potential new observations are included, uncertainty will be reduced. Although that is good news, it does not answer the question of which of these measurements (or combinations of measurements) we should prioritize.

To address this issue, let us assess the worth of each potential new well one-at-a-time. This requires defining an individual observation “type” for each individual new well. Recall that each well is actually comprised of two observations; one in layer 1 and another in layer 3. So each individual well observation “type” must list two observation names.

24 Return to *obstype.dat* (or re-open it in a text editor). Do not remove anything that is already there. Add a new header line for each new well. It is useful to provide each new "type" with a distinct name to make it easy to distinguish it from other types. For the purposes of the tutorial, name each new "type" according to its pertinent well number. Then add the names of the respective observations beneath each "type" header. Your file should now look something like this (new lines are shown in **bold**):

```
(…)
* observation type "forecast"
p-trib_1
p-total_1
* observation type "760"
h1-760
h3-760
* observation type "821"
h1-821
h3-821
* observation type "888"
h1-888
h3-888
* observation type "962"
h1-962
h3-962
* observation type "1137"
h1-1137
h3-1137
* observation type "1291"
h1-1291
h3-1291
* observation type "1502"
h1-1502
h3-1502
* observation type "1712"
h1-1712
h3-1712
* observation type "2045"
h1-2045
h3-2045
```

25 Save *obstype.dat* and run PREDUNC5 again. Provide the same responses to prompts as before.

26 After PREDUNC5 has finished, inspect *predunc5-add.out* once more. The post-calibration predictive uncertainties that prevail when each of the new observation "types" (i.e. each well) is added to the existing calibration dataset are now listed.

Figure 2 plots the post-calibration variance of predictive uncertainty that results from the addition of each observation "type" listed in *predunc5-add.out* to the existing calibration dataset As you can see by comparing the variance of the "base" observation type (dark grey bar) to that obtained when this set is supplemented by each individual well "type", the ability of each well on its own to reduce the variance of the prediction is not equal. Some wells have almost no effect (i.e. 760, 821, 888), whilst others (1712 and 2045) have almost as much effect as all of the potential new wells combined (see the "*all*" type). In fact, the decrease in prediction uncertainty that is accrued by adding well 2045 is almost the same as that which is accrued by adding all of the new wells.

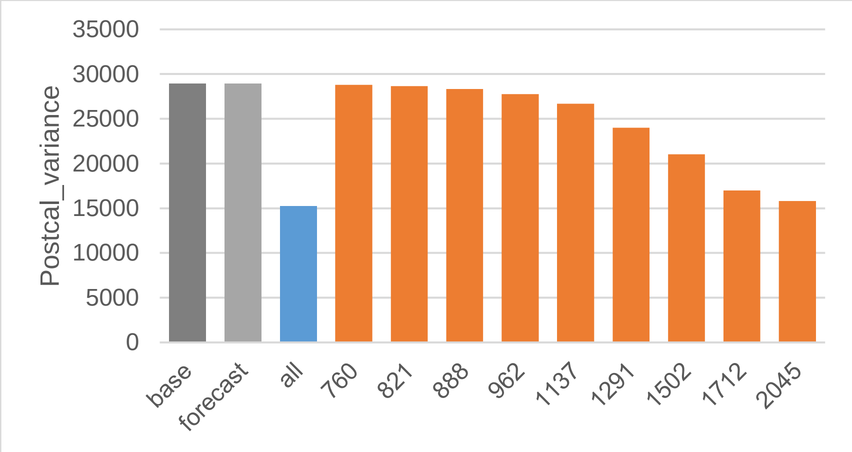

Figure 2 – PREDUNC5 calculated post-calibration variance of *p-trib_1* for each observation type added to the "base" calibration dataset.

Through the above analysis we have determined the ability of head observations taken in each individual well to reduce the uncertainty of our prediction. However, this tells us nothing about the uniqueness of information that each of these measurements carry. Because the sum of decreases in variance accrued through addition of each new well to the calibration dataset is less than the decrease in predictive uncertainty variance accrued through installation of all of them, some of the information that these potential head measurements carry must be duplicated So, for example, there is no guarantee that simultaneous installation of wells 1712 and well 2045 will result in the largest reduction in prediction uncertainty. It is entirely possible that information that is resident in head observations taken from well 1712 is repeated in head measurements made in well 2045.

3.1 Combinations of Observations

Say, for example, that we only have funds to install two new observation wells. If we wish to maximize the return on our investment, then we need to assess the decrease in prediction uncertainty this is accrued for all possible combinations of two new wells. In our case, that would be a total of 81 combinations; thus 81 observation "types" must be introduced to the observation type list file. Doing this programmatically is not too challenging. Doing it manually is possible, but painful (and we all have more important things to do with our lives). For the purposes of this tutorial we are going to cheat; we will only look at combinations of well 2045 (the one with the greatest uncertainty reduction) with each of the remaining eight potential wells. As well 2045 is so effective in reducing prediction uncertainty in comparison with the other wells, it is a good bet that a combination that includes well 2045 will be the one with the highest worth.

27 Return to *obstype.dat* (or re-open it in a text editor). Do not remove anything that is already there. Add new header lines for each new combination of well 2045 and each of the other eight wells. Name each new observation "type" as "*wellnumber + 2045*", replacing *wellnumber* with the appropriate well name. For example, the observation "type" that corresponds to the combination of well 760 and well 2045 should look like this. (Note that you can, if you wish, name these observation "types" whatever you like.)

```
(…)
* observation type "760 + 2045"
h1-2045
h3-2045
h1-760
h3-760
(…)
```

28 Add the eight combinations of wells as new observation “types” to the observation type list file. Then save *obstype.dat* and run PREDUNC5 again. Provide the same responses to its prompts as before.

29 After PREDUNC5 has finished, inspect *predunc5-add.out* once more. This file now lists post-calibration prediction uncertainties with each of the new observation “types” (i.e. the eight combinations of wells) included in the calibration dataset.

Figure 3 (an extract from file *predunc5-add.out*) displays the updated post-calibration prediction variance resulting from augmentation of the existing calibration dataset with each new observation “type”. As you can see, combining observations from well 2045 with any of the other wells leads to a relatively small decrease in prediction uncertainty in comparison with that accrued from well 2045 alone. However the largest decrease comes from combining well 2045 and well 1712. (It turns out that they were indeed the best combination!). This combination of wells reduces prediction uncertainty almost as much as all of the new observation wells put together; (see observation type “*all*”). Therefore, if we can only afford to drill two new observation wells, the best bet is to drill wells 2045 and 1712.

That being said, there is very little gain in adding well 1712 to well 2045. So we could probably save some money and just drill well 2045 – for the time being at least. This is because we cannot know what our revised prediction will be once we measure heads in this well and use them to constrain parameters in a repeated calibration process. The revised prediction may be benign enough for extraction from the pumping well to be considered safe with the revised uncertainties taken into account. Alternatively, calibration with the new measurements may result in a pessimistic prediction whose uncertainty limits still include the “bad thing” that management of our groundwater system seeks to avoid (i.e. reduction of stream flow below 6,500 m^3/day). In this case, acquisition of further data becomes necessary.

The worth of linear analysis in guiding acquisition of decision-pertinent data should be readily apparent to you by now.

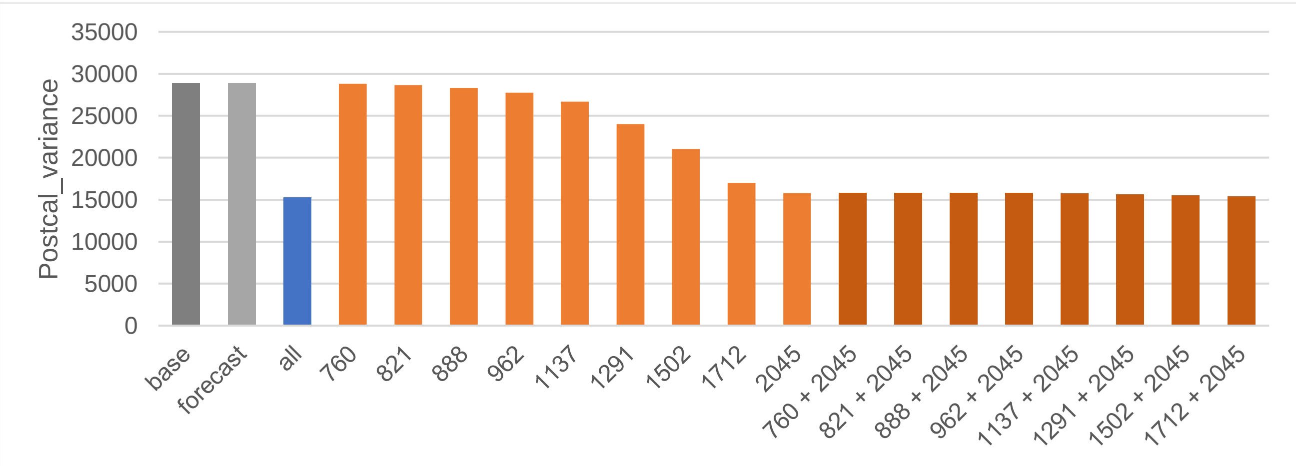

Figure 3 – Updated PREDUNC5 calculated post-calibration variance of *p-trib_1* for each observation type added to “base” calibration dataset.

3.2 Final Comments

Outcomes of data worth analysis such as the one demonstrated here can easily be plotted spatially (or over time) to provide didactic guidance to collection of further data. The example shown here is relatively simple. Its purpose is to introduce you to the basics of data worth analysis undertaken using PREDUNC5. Hopefully, this tutorial may encourage you to extend its use to real-world decision-support contexts.

As has been mentioned, it is entirely feasible to assess the worth of data that correspond to domain wide model outputs, and/or to model outputs at many simulation times, and/or any combination thereof. The complexity of your analysis will depend, of course, on the requirements of management that modelling is intended to support. More complex analyses will benefit from the use of self-written programs that automate construction and processing of PREDUNC5 input and output files. Alternatively, it is entirely feasible to prepare PREDUNC5 inputs, and postprocess PREDUNC5 outputs, using good old Excel. (Or an adventurous modeller may wish to write his/her own purpose-specific PREDUNC5; the equations on which it is based are not very complex.)

gmdsi

Groundwater Modelling Decision Support Initiative

gmdsi.org

BHP

Flinders UNIVERSITY

NATIONAL CENTRE FOR GROUNDWATER RESEARCH AND TRAINING

RioTinto