Assignment # 6 Steady State Model Calibration: Calibrate your model. If you want to conduct a transient calibration, talk with me first. Perform **calibration using UCODE.** **Be sure your report addresses global, graphical, and spatial measures of error as well as common sense.** Consider more than one conceptual model and compare the results. **Remember to make a prediction with your calibrated models and evaluate confidence in your prediction.** Be sure to save your files because you will want to use them later in the semester.

Suggested Calibration Report Outline

- **Title**
- **Introduction**
  - **describe the system to be calibrated (use portions of your previous report as appropriate)**
- **Observations to be matched in calibration**
  - **type of observations**
  - **locations of observations**
  - **observed values**
  - **uncertainty associated with observations**
  - **explain specifically what the observation will be matched to in the model**
- **Calibration Procedure**
- **Evaluation of calibration**
  - **residuals**
  - **parameter values**
  - **quality of calibrated model**
- **Calibrated model results**
  - **predictions**
  - **uncertainty associated with predictions**
  - **problems encountered, if any**
- **Comparison with uncalibrated model results**
- **Assessment of future work needed, if appropriate**
- **Summary/Conclusions**
- **References**

submit the paper as hard copy and include it in your zip file of model input and output
submit the model files (input and output for both simulations) in a zip file labeled: ASSGN6_LASTNAME.ZIP

Overview of UCODE & Associated Codes

Modes that can be accomplished:

Forward Process Model run with Residuals

Conduct Sensitivity Analysis

Estimate Optimal Parameter values and associated linear uncertainty

Evaluate quality of the model

Estimate values of Predictions and associated linear uncertainty

Evaluate model linearity

Evaluate NonLinear uncertainty associated with estimates of parameter values and predicted values

Auxiliary: Investigate Objective Function

See UCODE Manual
Chapter 1 for overview and description of manual contents
Chapter 2 for overview and program control
Also Chapter 4 Table 3

Overview of UCODE in Parameter Estimation Mode

1) manipulates application model input files and reads simulated values from application model output files

2) compares user-provided observations with equivalent simulated values derived from the values read from the application model output files using a weighted least-squares objective function

3) uses a modified Gauss-Newton method to adjust the value of user selected input parameters in an iterative procedure to minimize the value of the weighted least-squares objective function

4) reports the estimated parameter values

5) calculates and prints statistics to be used to
  - diagnose inadequate data
  - identify parameters that probably cannot be estimated
  - evaluate estimated parameter values
  - evaluate accuracy of model representation of processes
  - quantify the uncertainty of model simulated values

See UCODE Manual Chapter 2 for Flow Chart

See UCODE Manual Chapter 3 for general issues related to using UCODE for Nonlinear Regression

In parameter estimation mode, UCODE performs inverse modeling, using nonlinear regression. Sensitivities are calculated by perturbation (many significant figures are needed) or read from a process model output file.

Any application/process model or set of models (pre/post processors)

Must have numerical (ASCII text) input and output files, with numbers of sufficient significant digits

Estimated parameter can be a quantity in the input files, or one used in conjunction with functions to calculate that quantity

Observations can be any quantity for which a simulated equivalent value can be read or calculated from values in model output files using additive/multiplicative functions

Prior information on estimated parameters also can be included

KEEP UP-TO-DATE on

New Developments

And

UCODE Software updates

USGS Software Web Site
http://water.usgs.gov/nrp/gwsoftware/
Also on the USGS general software page

BOOKMARK THAT PAGE!
DOWNLOAD AND INSTALL THE LATEST version of UCODE

MODFLOW is our process model to be calibrated

MODFLOW-2000/2005 include:
GLO Global Process that controls overall program flow
GWF Ground-water Flow Process
OBS Observation Process

MODFLOW-2000 also includes:
SEN Sensitivity Process (slated for MF2005)
PES Parameter Estimation Process (being discontinued)

UCODE INPUT

Once we have a working forward model that runs in batch mode

Set up UCODE files and run the forward model through UCODE to be sure everything is functioning properly.

NEVER bypass this step.

Follow input directions in the UCODE MANUAL provided (review Chapters 1-3 for an overview, Chapter 4 for how to execute UCODE, and Chapter 5 for the general block format input that is used throughout the UCODE input file.

OR USE THE NEW USGS code

MODELMATE TO SET UP UCODE FILES

UCODE INPUT

We'll use a beta version of the soon-to-be-released ModelMate and a Tutorial tailored to our class example to generate ucode files from MF2005 files

(** first!! download software, tutorial for our example, MFfiles-PreMM today's class web page)

It is easier to learn the input after you have a better idea of the process

So approach this with the idea of absorbing the gestalt vs the details

UCODE Manual Chapter 5, Table 4, OVERVIEW of INPUT BLOCKS

Follow tutorial and see:

Ucode_main.in

Major items of note General Input Format (note these in main ucode input file on web page link):

- a # in column 1 specifies a comment and the line is not read by the code
- Input is specified in blocks, these are designated with a BEGIN and an END
- Notice the Block Names, their order and whether they are required or optional.
- Begin is followed by the block name and the format option for how the input will be entered (keywords or table, or file format: the default is keywords), and END is followed by the block name
- Blocks have keywords to identify the start of a new group of information
- Some input data are matrices which may be input as Standard or Compressed format (see an example of the compressed format in the Matrix Input Format section)

BLOCK NAMES

Follow tutorial and see Ucode_main.in

Original

Chapter	Purpose	Blocklabel	Default column order[2]
[4]6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[5]	Yes
		Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction _Data[6]	Yes
		Derived_ Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

Additional

Purpose	Blocklabel	Default column order[2]
Define UCODE_2005 operation	Options	No
	Merge_Files	No
	UCODE_Control_Data	No
	Reg_GN_Controls	No
	Reg_GN_NonLinInt	No
	Model_Command_Lines	No
Define parameters	Parameter_Groups	No
	Parameter_Groups_For_Prediction	No
	Parameter_Data	Yes
	Parameter_Data_For_Prediction	Yes
	Parameter_Values	Yes
	Derived_Parameters	Yes
	Derived_Parameters_For_Prediction	Yes
Define observations	Observation_Groups	No
	Observation_Data	Yes
	Derived_Observations	Yes
Define predictions	Prediction_Groups	No
	Prediction _Data	Yes
	Derived_ Predictions	Yes
Define prior information	Prior_Information_Groups	No
	Prior_Information_Groups_For_Prediction	No
	Linear_Prior_Information	Yes
	Linear_Prior_Information_For_Prediction	Yes
Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
Interact with process-model input and output files.	**Model_Input_Files**	Yes
	Model_Output_Files	Yes
Run process model(s) using multiple processors	Parallel_Control	No
	Parallel_Runners	Yes

BLOCK FORMATS

Follow tutorial and see Ucode_main.in

Table 5. Blockformat options.

Blockformat	Prescribed input format
KEYWORDS	Blockbody consists of a series of lines of the form: Keyword=value Under some circumstances there are restrictions on how the lines are ordered; see the input block instructions. If no blockformat is specified, KEYWORDS is assumed, but it is advisable to explicitly identify the block format to reduce errors. Comments are allowed.[1,2]
TABLE	Blockbody consists of a table of data that may have labels on the columns and may be read from the main input file or from another input file. See the text for additional information. Comments are allowed right after the BEGIN statement but not in the rest of the input block.[1]
FILES	Blockbody consists of the pathname for one or more files. Comments are allowed.[1,2] To allow the format to be specified, the contents of each of the listed files needs to begin with a 'Begin Blocklabel [Blockformat]' line and end with an 'End Blocklabel' line. The Blocklabel needs to be the same as in the 'Begin Blocklabel FILES' block within which the files are listed. See the section "Observation_Data Input Block" for an example.

[1] Comments are separate lines starting with a # in the first column. No blank lines are allowed within any input blocks.
[2] Comments can be inserted anywhere within the input block.

OPTIONS BLOCK

Chapter 6 (p52)

Follow tutorial and see Ucode_main.in

In:
ucode1
FILE: ep_Ucode_main.in

```
BEGIN Options
  Verbose=0
END Options
```

Later, if UCODE is not running as you expect, use 5

Notice other possible input items

Chapter[1]	Purpose	Blocklabel	Default column order[2]
[4]6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[5]	Yes
		Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction_Data[6]	Yes
		Derived_Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

UCODE_CONTROL_DATA BLOCK

Chapter 6 (p55)

Follow tutorial and see Ucode_main.in

In:
ucode1
FILE: ep_Ucode_main.in

BEGIN UCODE_CONTROL_DATA KEYWORDS
ModelName=antfarm
ModelLengthUnits=centimeters
ModelTimeUnits=seconds
sensitivities=no
optimize=no
DataExchange=yes
END UCODE_CONTROL_DATA

Later, other options will be of interest, so notice other possible input items

Chapter[1]	Purpose	Blocklabel	Default column order[2]
[4]6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[5]	Yes
		Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction_Data[6]	Yes
		Derived_Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

Model_Command_Lines BLOCK
Chapter 6 (p65)

Follow tutorial and see Ucode_main.in

If a un-required block is skipped, the defaults are used

```
BEGIN MODEL_COMMAND_LINES
'Command=ep.bat'
 purpose=forward
 CommandId=modflow
END MODEL_COMMAND_LINES
```

The batch file should run all the codes you want to run for your calibration

ALL SHOULD RUN AND CLOSE WITHOUT HUMAN INTERVENTION **** REMOVE PAUSE FROM EP.BAT*******

CommandId not used by UCODE at this time

Chapter[1]	Purpose	Blocklabel	Default column order[2]
[4]6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[5]	Yes
		Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction_Data[6]	Yes
		Derived_Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

PARAMETER_GROUPS BLOCK

Follow tutorial/see Ucode_main.in

Chapter 7 (p68)

pval package

```
8
K1  0.01
K2  10.0
KVA 1.0
KVCB 1E-5
rch1 0.1
rch2 0.05
rch3 0.01
Kriv 1e-3
```

```
BEGIN PARAMETER_GROUPS KEYWORDS
groupname=Ks    transform=yes adjustable=yes
groupname=RCH   transform=no  adjustable=yes
groupname=FIXED transform=no  adjustable=no
END PARAMETER_GROUPS
```

Use groups to define common characteristics

NOTE Log Transforming Parameters:
* may help the problem converge more readily
* prevents parameter values from becoming negative
* allows specification of statistics on prior information to be in log space

Notice other possible input items

Chapter[1]	Purpose	Blocklabel	Default column order[2]
[4]6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[5]	Yes
		Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction_Data[6]	Yes
		Derived_Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

PARAMETER_DATA BLOCK Chapter 7 (p69)

Follow tutorial/see Ucode_main.in

```
BEGIN PARAMETER_DATA TABLE
nrow=8  ncol=7  columnlabels
paramname startvalue lowervalue uppervalue scalepval groupname
K1        0.01       0.001      0.1        0.000001     Ks
K2        10.0       1.0        100.0      0.0010       Ks
KVA        1.0       0.1        10.0       1e-4         FIXED
KVCB      1e-5       1e-6       1e-4       1e-9         Ks
rch1      0.1        0.01       1.0        1e-5         RCH
rch2      0.05       0.005      0.5        5e-7         RCH
rch3      0.01       0.001      0.1        1e-7         RCH
Kriv      1.E-1      1.E-2      1.E-0      1.E-5        Ks
END PARAMETER_DATA
```

NOTE: reasonable maximum & minimum does not affect the regression

It is best to NOT impose limits on estimated parameter values, but this can be done in UCODE with the Constrain variable

Not constraining may yield unreasonable values, that are valuable for identifying conceptual model errors, data errors, or insufficient info

Only use constrain if the model will crash beyond that value

Notice other possible input items

Chapter[1]	Purpose	Blocklabel	Default column order[2]
[4]6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[3]	Yes
		Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction_Data[5]	Yes
		Derived_Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

OBSERVATIONS Chapter 8

Follow tutorial/see Ucode_main.in

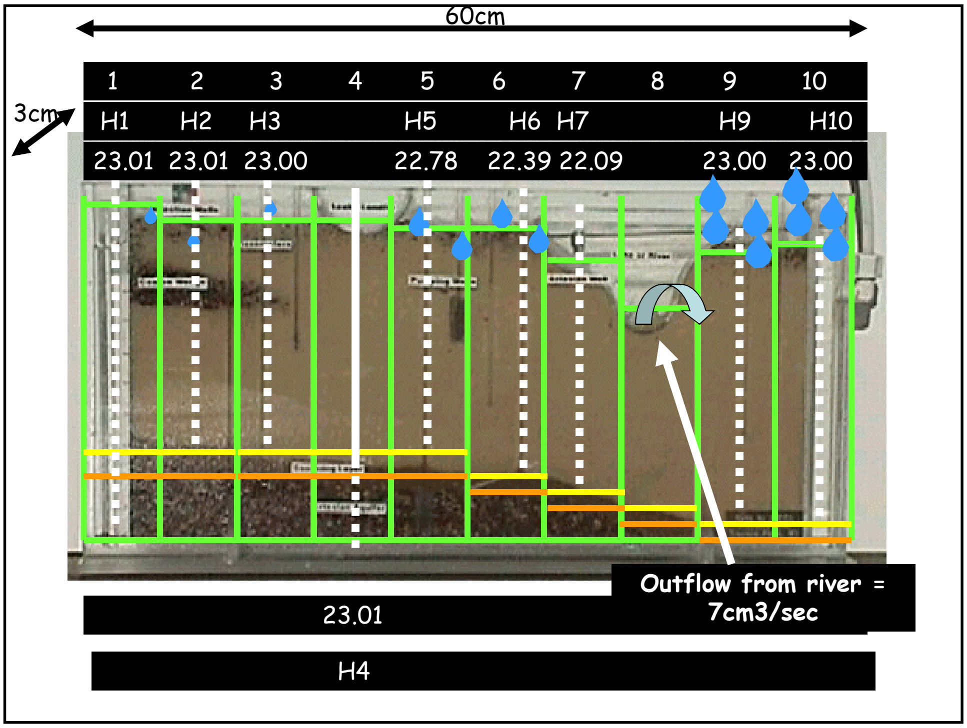
We use the simple approach of reading the observation in order from the MODFLOW _os file

You can do very complex extractions, involving searching for character strings and counting lines and sapces such that values can be extracted from ANY format of text file

Also you can make calculations from the items you extract to obtain simulated equivalents

Chapter[1]	Purpose	Blocklabel	Default column order[2]
[4]6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[3]	Yes
		Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction_Data[5]	Yes
		Derived_Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

One or the other



Head Observations 95% confidence is 0.1 cm

				HEAD		
1	x= 1.5	y=1.5	multilayer	23.01 cm	column 1	coff 0
2	x= 7.5	y=1.5	layer 1	23.01 cm	column 2	coff 0
3	x=13.5	y=1.5	layer 1	23.00 cm	column 3	coff 0
4	x=19.5	y=1.5	layer 2	23.00 cm	column 4	coff 0
5	x=25.5	y=1.5	layer 1	22.78 cm	column 5	coff 0
6	x=33.0	y=1.5	layer 1	22.39 cm	column 6	coff 0.25
7	x=37.5	y=1.75	layer 1	22.08 cm	column 7	coff -0.25
8	x=49.5	y=1.5	layer 1	22.99 cm	column 9	coff 0
9	x=55.5	y=1.5	layer 1	23.01 cm	column 10	coff 0

Flow Observation 95% confidence +/-5%

Outflow from river = 7.0cm^3/sec

add to name file: data 22 ep._os

hob package

```
 9 1 2 22 888.
 1.0
H1     -2    2   1      1      1       0.0  0.0 23.01
 1  0.5  2 0.5
H2      1    2   2      1      1       0.0  0.0  23.00
H3      1    2   3      1      1       0.0  0.0  23.01
H4      2    2   4      1      1       0.0  0.0  23.00
H5      1    2   5      1      1       0.0  0.0  22.78
H6      1    2   6      1      1       0.0  0.25 22.39
H7      1    2   7      1      1      -0.25 0.0  22.08
H8      1    2   9      1      1       0.0  0.0  22.99
H9      1    2  10      1      1       0.0  0.0  23.00
```

rvob package

```
 1 3 1 22
 1.0
 1 -3
 Q1 1 1.0  -7.0
 1 1 8 1.0
 1 2 8 1.0
 1 3 8 1.0
```

RUN MODFLOW AND OBSERVE _os file

OBSERVATION_GROUPS BLOCK
Chapter 8 (p81)

Follow tutorial/see Ucode_main.in

```
BEGIN OBSERVATION_GROUPS
  groupname=Heads
  plotsymbol=1
  groupname=RIV_flows
  plotsymbol=2
END OBSERVATION_GROUPS
```

Use groups to define common characteristics

We could extract some values but not "use" them as simulated equivalents, rather we might calculate simulated equivalents with those values

Notice other possible input items

Chapter[1]	Purpose	Blocklabel	Default column order[2]
[4]6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[5]	Yes
	One or the other	Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction_Data[6]	Yes
		Derived_Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

Follow tutorial/see Ucode_main.in

OBSERVATION_DATA BLOCK - Chapter 8 (p83)

```
BEGIN OBSERVATION_DATA TABLE
NROW=10  NCOL=5  COLUMNLABELS
obsname  obsvalue statistic  statflag  groupname
h1       23.01     0.0025      var     heads
h2       23.00     0.0025      var     heads
h3       23.01     0.0025      var     heads
h4       23.00     0.0025      var     heads
h5       22.78     0.0025      var     heads
h6       22.39     0.0025      var     heads
h7       22.08     0.0025      var     heads
h8       22.99     0.0025      var     heads
h9       23.00     0.0025      var     heads
Q1       -7.0      0.05        cv      RIV_flows
END OBSERVATION_DATA TABLE
```


Chapter[1]	Purpose	Blocklabel	Default column order[2]
4/6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[5]	Yes
		Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction_Data[6]	Yes
		Derived_Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

One or the other

Follow tutorial/see Ucode_main.in

MODEL_INPUT_FILES BLOCK - Chapter 11 (p108)

```
BEGIN MODEL_INPUT_FILES
MODINFILE=ep.pval  TEMPLATEFILE=ep_pval.jtf
END MODEL_INPUT_FILES
```

Any file names are OK

I use jtf or tpl to help myself identify files

The modinfile must be what the process model expects to use

Template files start with jtf and a symbol. They are copies of process model input files with values where estimated parameters should go replaced by parameter names from the PARAMETER_DATA block bounded by the symbol. Leave room for lots of significant figures

Chapter[1]	Purpose	Blocklabel	Default column order[2]
4/6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[5]	Yes
		Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction_Data[6]	Yes
		Derived_Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

Follow tutorial/see ep_pval.jtp

Make your template files - Chapter 11 (p109)

Template files start with jtf and a symbol. They are copies of process model input files with values where estimated parameters should go replaced by parameter names from the PARAMETER_DATA block bounded by the symbol.
Leave room for lots of significant figures

It is convenient to use the pval file because all of the parameters we might want to estimate are there.

pval package

```
8
K1   0.01
K2   10.0
KVA 1.0
KVCB 1E-5
rch1 0.1
rch2 0.05
rch3 0.01
Kriv 1e-3
```

```
jtf @
8
K1 @K1                         @
K2 @K2                         @
KVA @KVA                        @
KVCB @KVCB                       @
rch1 @rch1                       @
rch2 @rch2                       @
rch3 @rch3                       @
Kriv @Kriv                       @
```

Follow tutorial/see Ucode_main.in

MODEL_OUTPUT_FILES BLOCK - Chapter 11 (p111)

```
BEGIN MODEL_OUTPUT_FILES
  MODOUTFILE=ep._os
  INSTRUCTIONFILE=ep._os.jif
  category=obs
END MODEL_OUTPUT_FILES
```

Any file names are OK

The modoutfile must be what the file the process model produces with the results you want to extract

Chapter[1]	Purpose	Blocklabel	Default column order[2]
[4]6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[5]	Yes
		Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction_Data[8]	Yes
		Derived_Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

Follow tutorial/see ep._os.jif

INSTRUCTION FILE - Chapter 11 (p 113 115)

The way we have MODFLOW set up now we can use a standard instruction file. Far more complicated extractions can be made.

```
jif @
StandardFile  1  1  10
H1
H2
H3
H4
H5
H6
H7
H8
H9
Q1
```

There is great flexibility to address most any situation you encounter. Chapter 11 has much detail on this.

Batch file to run UCODE EXECUTE UCODE in a FORWARD RUN

Follow tutorial/see RunUcode_epMM.bat

```
Path-to-ucode-executable ucode-input-file root-calib-output root-pred-output
pause
```

See UCODE Manual Chapter 4 (p29)
for executing UCODE & associated codes

See Ucode_main_out.#uout and _files

What new files were created?
Drag them into the text editor
Read through their contents

VERY IMPORTANT USE YOUR COMMON SENSE

Have expectations for the results, question all aspects of the situation when calculations do not match expectations

Fix Problems

If you do not have a pause in the batch file,
you may be confused about what might have gone wrong
Go to the command window
Type “cd” space and drag the model folder in, enter
Now drag the batch file in, enter
Note what is on the screen for clues to any problems

REMEMBER

When you run a code, you should expect that there will be errors and be pleasantly surprised if there are not. When you see an error:

1) look closely at the error message, try to understand it, use any clue that may be provided (paths, directories, file names, numbers) to explore it
2) check the directory to see what files were created and view their contents, look at the dates and times on files to determine what was created recently
3) delete outputs and try it again and look at the new outputs
4) as Winston Churchill once said, "never, never, give up". If you do not find the error, keep thinking and experimenting to decipher the situation. Utilize “show me” skills.

Follow tutorial/see Ucode_main_out.#uout and _files

EVALUATING OUTPUT

Notice any errors in the command window and read the file to confirm everything is what you expected

The most common error is related to paths and file names
Next common error is improper substitution or extraction

Check that the UCODE input items are echoed correctly.
View the output (see Chapters 14 and 16)
fn.#uout & DataExchange files: fn._*

Note GWChart works for ucode _ files

Follow tutorial and use GWChart

Follow tutorial and use GWChart

EVALUATING OUTPUT

fn.**#uout includes statistics, top portion of Fit Statistics Table 28**

These reflect model fit
given the initial model configuration and starting values
USE GWChart for convenient viewing of files

Exceptionally large discrepancies between simulated and observed values may indicate that there is a conceptual error either in the model configuration or in the calculation of the simulated values

Fixing these now can eliminate many hours of frustration.

Data exchange files include residual informations at starting values Table 31

It is essential for UCODE to perform correctly in the forward mode.
Proceeding with errors will result in an invalid regression and wasted time.

Resolve any problems and continue

Due this week:

UCODE files set up using modelmate for your project and preliminary analysis of output

Include the associated mtc file and the modflow files

Follow tutorial for sensitivity run / see ep_Ucode_main.in & Ucode_main_out.#uout

SENSITIVITY of the SIMULATED EQUIVALENTS to the PARAMETER VALUES

Sensitivities serve two functions:

indicators of

- importance of the observations to the estimation of the different parameters

- importance of each parameter to the simulated values

needed by the modified Gauss-Newton method to

- determine parameter values that produce the best fit

The sensitivities are determined as

[simulated(current b values)-simulated(perturbed b values)]/[(current b's) - (perturbed b's)]

i.e.

$$\frac{\text{simulated}(b_o) - \text{simulated}(b')}{b_o - b'}$$

They form a 2D array (ND,NP) i.e. (#observations x #parameters)

Notation:
1D array, single underscore
2D array, double underscore

$$\underline{\underline{X}} = \begin{bmatrix} \frac{\partial y'_1}{\partial b_1} & \frac{\partial y'_1}{\partial b_2} & \cdots & \frac{\partial y'_1}{\partial b_{NP}} \\ \frac{\partial y'_2}{\partial b_1} & \frac{\partial y'_2}{\partial b_2} & \cdots & \frac{\partial y'_2}{\partial b_{NP}} \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ \frac{\partial y'_{ND}}{\partial b_1} & \frac{\partial y'_{ND}}{\partial b_2} & \cdots & \frac{\partial y'_{ND}}{\partial b_{NP}} \end{bmatrix}$$

Follow tutorial for sensitivity run / see Ucode_main.in & Ucode_main_out.#uout

Evaluate SENSITIVITY of the
SIMULATED EQUIVALENTS to the PARAMETER VALUES
ΔSimulatedEquivalent/ΔParameterValue
with perturbation sensitivities
ucode1

UCODE_Control_Data Block (p55)
sensitivities = yes

Choose printing options or default

Notice other possible input items

Chapter[1]	Purpose	Blocklabel	Default column order[2]
[4]6	Define UCODE_2005 operation	Options	No
		Merge_Files[3]	No
		UCODE_Control_Data[3]	No
		Reg_GN_Controls[3]	No
		Reg_GN_NonLinInt[3,4]	No
		Model_Command_Lines	No
7	Define parameters	Parameter_Groups[3]	No
		Parameter_Data[3]	Yes
		Parameter_Values[3]	Yes
		Derived_Parameters[3]	Yes
8	Define observations	Observation_Groups	No
		Observation_Data[5]	Yes
		Derived_Observations	Yes
	Define predictions	Prediction_Groups	No
		Prediction_Data[6]	Yes
		Derived_ Predictions	Yes
9	Define prior information	Prior_Information_Groups	No
		Linear_Prior_Information	Yes
10	Define variance-covariance matrices to weight groups of observations or prior information with correlated errors.	Matrix_Files	No
11	Interact with process-model input and output files.	**Model_Input_Files**	Yes
		Model_Output_Files	Yes
12	Run process model(s) using multiple processors	Parallel_Control	No
		Parallel_Runners	Yes

Follow tutorial for sensitivity run / see Ucode_main.in & Ucode_main_out.#uout

EXECUTE UCODE in the SENSITIVITY MODE

Look for the differences in the #uout file
What are the sensitivities?

Are there some parameters that will be difficult to estimate?
Dimensionless scaled sensitivity – 1 for each obs and parameter
dss = unscaledsens * (PARAMETER_VALUE * (wt**.5))
Composite Scaled Sensitivities – 1 for each parameter
css = ((SUM OF THE SQUARED DSS) / ND)**.5
Generally should be >1 AND
within ~ 2 orders of magnitude of the most sensitive parameter

Notice statistics are calculated for the starting parameter values as if they were optimal
This can be useful if you want to regenerate the statistics for an optimal parameter set

Follow tutorial for sensitivity run / see Ucode_main.in & Ucode_main_out.#uout

See "Perturbation Sensitivities" starting on p15

Accuracy of Sensitivities Depends on:

number of accurate significant figures in extracted simulated values (print many significant figures and extract them all)

magnitude of the simulated values

magnitude of the substituted parameter values

size of the parameter perturbations, for nonlinear parameters

What if Sensitivities are zero?

If more than a few sensitivities equal zero, it may indicate extracted perturbed & unperturbed values are identical (given the significant figures) or perhaps the model did not execute

See "What to Do When Sensitivities Equal Zero" (p37) of the UCODE manual.

If sensitivities are zero for a Parameter:

- If many other sensitivities are nonzero, observation is not very important, NO corrective action needed

- If all sensitivities are zero, corrective action is needed (if there is a hydraulic reason for lack of sensitivity, do not estimate the parameter)

- If many sensitivities are zero, corrective action MAY OR MAY NOT be needed

What if Sensitivities are zero?

Five possible corrective actions:
1) smaller solver convergence criteria can be specified in the application codes;
2) the extracted values can be printed with more significant figures in the application model output file if the values are calculated with sufficient accuracy;
3) the datum of the problem can be changed or a normalization can be applied;
4) the perturbation for the parameter can be changed; too small perturbations may result in negligible differences in extracted values, or differences that are obscured by round-off error; too large may yield inaccurate sensitivities for nonlinear parameters
5) the methods for coping with insensitive parameters discussed later can be employed.
- Reconsider the model construction
- Modify the defined parameters
- Eliminate observations or prior information, if biased
- Adjust weights either for groups of, or individual, observations

Sensitivities calculated for the values of the parameters just prior to failure can be investigated by substituting these parameter values as starting values in the prepare file and executing UCODE with sensitivities=yes, optimize=no. (add SenMethod=2 to also evaluate correlation)

Sensitivities for all intermediate sets of parameter values can be investigated by setting IntermedPrint=sensitivities in the input file and executing UCODE again with optimize=yes.

Follow tutorial for parameter estimation run/see Ucode_main.in & Ucode_main_out.#uout

Estimate Optimal Parameter Values & Associated Uncertainty

UCODE_Control_Data Block (p55)

optimize = yes

EXECUTE UCODE in the OPTIMIZE MODE

Have a look at the output to see the troubles

This is what you will need to do to get your projects going in the future at work

The regression control defaults work OK in this case

In other cases use a Reg_GN_Controls Block (p60)

To override defaults, most commonly

```
BEGIN REG_GN_CONTROLS KEYWORDS
tolpar=0.01
maxiter=10
maxchange=2.0
stats_on_nonconverge=yes
END REG_GN_CONTROLS
```

Notice other possible input items

EVALUATING PARAMETER ESTIMATION OUTPUT

View fn.#uout

Consider Progress of the Regression Table 29 (p 178)

Parameter Statistics Table 30 (p179)

Follow tutorial for parameter estimation run/see Ucode_main.in & Ucode_main_out.#uout

Read through the resulting files

VERY IMPORTANT: USE YOUR COMMON SENSE

Most common trouble is lack of convergence, or progress toward it. Consider how to tackle that.

Have expectations for the results, question all aspects of the situation when calculations do not match expectations

Fix Problems

Evaluate Results

What do you make of the estimated parameter values? What of the confidence intervals?

EVALUATING PARAMETER ESTIMATION OUTPUT

OVERALL FIT, SUM OF SQUARED ERRORS

$$S(\underline{b}) = \sum_{i=1}^{ND} \omega_i \left[y_i - y_i^{'}(\underline{b}) \right]^2$$

CALCULATED ERROR VARIANCE (cev)

$$\mathbf{cev} = \mathbf{s}^2 = \frac{\mathbf{S(b)}}{\mathbf{ND - NP}}$$

STANDARD ERROR sqrt(cev)

$$\mathbf{s} = \sqrt{\mathbf{s}^2}$$

Model Selection Criteria

MLOF / AIC / AICc / BIC / KIC

The regression is not extremely sensitive to the weights, thus the casual approach to their definition is not a problem

The weighting can be evaluated at the end of the regression by considering the cev (calculated error variance)

smaller values of s^2 and s indicate a better fit

values close to 1.0 indicate the fit is consistent with the data accuracy as described by the weighting

cev > 1 (eg 95% confidence intervals on cev completely above 1) indicates the modeler globally underestimated the variances (i.e. the model does not fit the observations as well as the variances assigned by the modeler would reflect)

cev < 1 (eg 95% confidence intervals on cev completely below 1) indicates the modeler globally overestimated the variances (i.e. the model fits better than expected)

The 95% confidence intervals on cev are calculated using the ChiSq distribution. Deviations from 1.0 are significant if 1.0 falls outside of the confidence limits.

The modeler could adjust weights to obtain 1, but it is not necessary as long as the cev is discussed along with the input variances

CONSIDER HOW THE PARAMETER UNCERTAINTY IS CALCULATED

Variance Optimal Parameters:

$$V(\underline{b}) = \frac{\text{Sum of Squared Weighted Residuals}}{\text{\#Observations - \#Parameters}} \left[\underline{X}^T \underline{w} \underline{X} \right]^{-1}$$

$$V(\underline{b}) = \text{cev} \left[\underline{X}^T \underline{w} \underline{X} \right]^{-1}$$

$\underline{b}$ vector of optimal parameters (e.g. K,S,R,H,Q)

$\underline{X}$ sensitivity matrix

$\underline{w}$ weight matrix for observations

Results in NPxNP matrix, with variances on the diagonal

$$V(\underline{b}) = \begin{pmatrix} K & KS & KR & KH & KQ \\ SK & S & SR & SH & SQ \\ RK & RS & R & RH & RQ \\ HK & HS & HR & H & HQ \\ QK & QS & QR & QH & Q \end{pmatrix}$$

$$V(\underline{b}) = cev\ [\underline{X}^T \underline{w}\, \underline{X}]^{-1} = V(\underline{b}) = \begin{pmatrix} var1 & cov12 & cov13 \\ cov21 & var2 & cov23 \\ cov31 & cov32 & var3 \end{pmatrix}$$

[NP x NP] = constant [NP x ND] [ND x ND] [ND x NP]

e.g. If NP=3 and ND=12 then V(b) is a 3x3:

$$V(\underline{b}) = \left(cev \begin{bmatrix} X_{1,1} & X_{2,1} & X_{3,1} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & X_{ND,1} \\ X_{1,2} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ X_{1,NP} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & X_{ND,NP} \end{bmatrix} \begin{bmatrix} w_1 & w_{1,2} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ w_{2,1} & w_2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & w_3 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & w_4 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & w_5 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & w_6 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & w_7 & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & w_8 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & w_9 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & w_{10} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & w_{11} & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & w_{ND} \end{bmatrix} \begin{bmatrix} X_{1,1} & X_{1,2} & X_{1,NP} \\ X_{2,1} & \cdot & \cdot \\ X_{3,1} & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ X_{ND,1} & \cdot & X_{ND,NP} \end{bmatrix} \right)^{-1}$$

VARIANCE (K)

$$VAR(K) = \left(\underline{\underline{X}}^T \underline{\underline{\omega}}\, \underline{\underline{X}}\right)^{-1}_{1,1} (EVAR)$$

$$Std\ \ Dev = \sqrt{VAR\ \ (K)} \qquad 95\ \%\ Confid = K +/- 2 * StdDev$$

VARIANCE (H)

$$VAR(H) = \left(\underline{\underline{X}}^T \underline{\underline{\omega}}\, \underline{\underline{X}}\right)^{-1}_{4,4} (EVAR)$$

$$Std\ \ Dev = \sqrt{VAR\ \ (H)} \qquad 95\ \%\ Confid = H +/- 2 * StdDev$$

CORRELATION (normalized variance)

$$CORR(i,j) = \frac{COV(i,j)}{\sqrt{VAR(i)} * \sqrt{VAR(j)}}$$

$$\begin{array}{c|cccc|} & j=1 & \bullet & \bullet & j=NP \\ i=1 & 1,1 & 1,2 & \bullet & 1,NP \\ \bullet & 2,1 & 2,2 & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet \\ i=NP & NP,1 & 4,2 & NP,3 & NP,NP \end{array}$$

$$Corr = \begin{pmatrix} 1 & CorKS & CorKR & CorKH & CorKQ \\ CorSK & 1 & CorSR & CorSH & CorSQ \\ CorRK & CorRS & 1 & CorRH & CorRQ \\ CorHK & CorHS & CorHR & 1 & CorHQ \\ CorQK & CorQS & CorQR & CorQH & 1 \end{pmatrix}$$

Follow tutorial for parameter estimation run/see Ucode_main.in Ucode_main_out.#uout and _files

As before view residual statistics / sensitivities

Using GWChart also

See previous items and more from

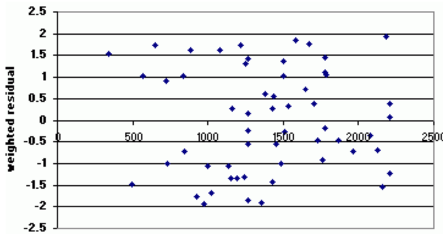
Tables 28 (p 176) and 31 (p 180)

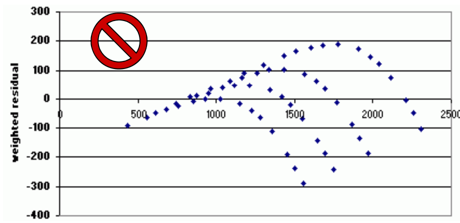
EVALUATING PARAMETER ESTIMATION OUTPUT

RESIDUALS

ws (weighted residuals vs simulated equivalents)

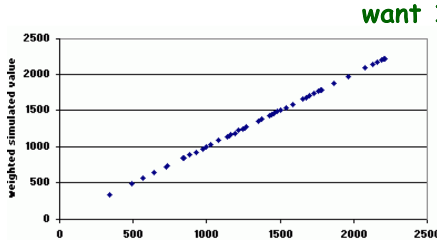
want narrow band around 0

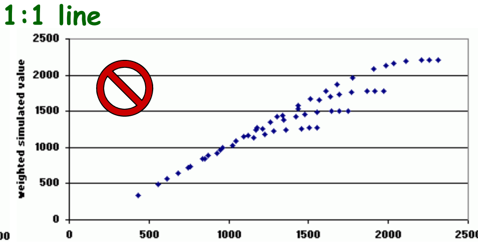




ww (weighted observed versus simulated)

want 1:1 line



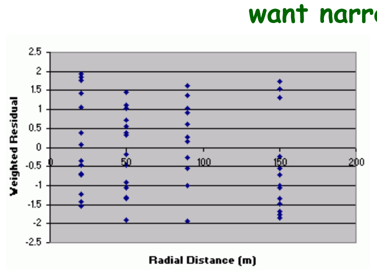


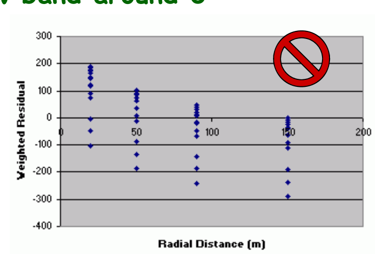
EVALUATING PARAMETER ESTIMATION OUTPUT

RESIDUALS (if you include a root.xyzt file)

wxyzt (weighted residuals vs space [1D distance in this illustration])

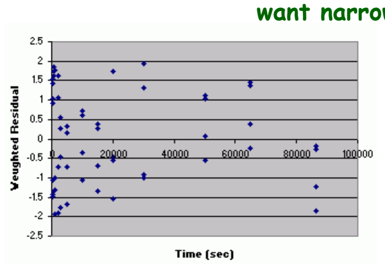
want narrow band around 0

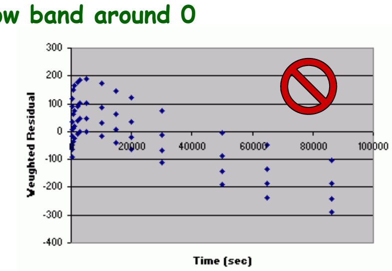




wxyzt (weighted observed versus time)

want narrow band around 0



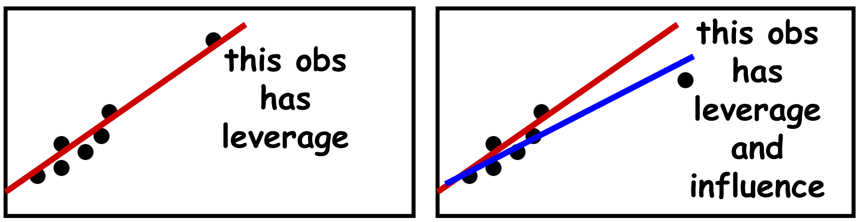


Extensive model analysis and development work can be accomplished by analyzing residuals

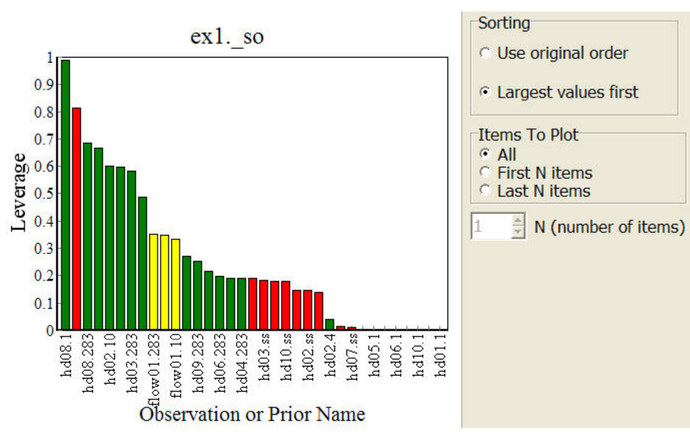
Explore the rest of the data exchange files

Various sensitivity representations (sc sd s1 so su)

Parameter Information (paopt pc pasub)



A typical _so



If the PARAMETER ESTIMATION is successful:
Further EVALUATE RESULTS
with UCODE's Residual_Analysis (p159 and on)
It only needs the data exchange files, but there is optional input described in the ucode manual

Create batch file for residual_analysis OR run in ModelMate

Additional Residual Analyses can be obtained running

residual_analysis.exe >>> fn.#resan

VIEW RESULTS WITH GW_CHART
_nm - want normally distributed residuals
If not a straight line compare to realizations of residuals:
Uncorrelated _rd – if these look like your nonlinear nm plot the cause is too few residuals
Correlated _rg – if these look like your nonlinear nm plot it is OK, due to correlation in the regression

ALSO see rdadv of residual_analysis_adv.exe on next slide

Create a batch file to run residual_analysis_adv or run in ModelMate

View _rdadv in GW_Chart

to see the theoretical confidence limits on the weighted residuals

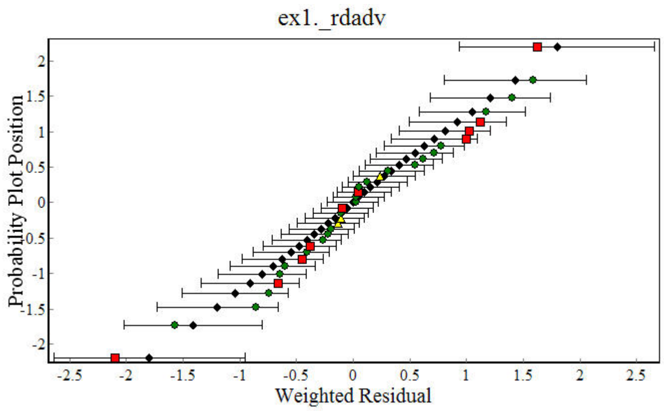
#resanadv

Mean Weighted Residual should be ~ 0
Slope should be ~ 0

INTRINSIC NONLINEARITY << Sum of Squared Residuals
If large Corfac_plus correction factors may not be accurate

CED correlation of weighted residuals and means of synthetic residuals
PROB – probability that a correlation would be <= CED if the residuals were normally distributed

A typical _rdadv



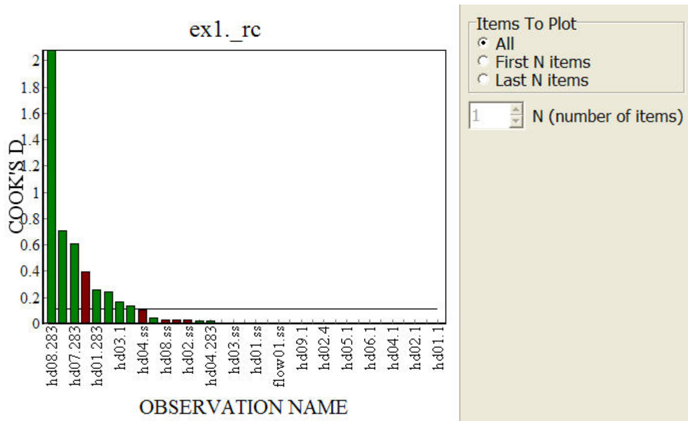
Back to:
EVALUATING PARAMETER ESTIMATION OUTPUT from
Residual_analysis fn.#resan _rc _rb

Cook'sD large values indicate observations that most influence all estimated parameter values

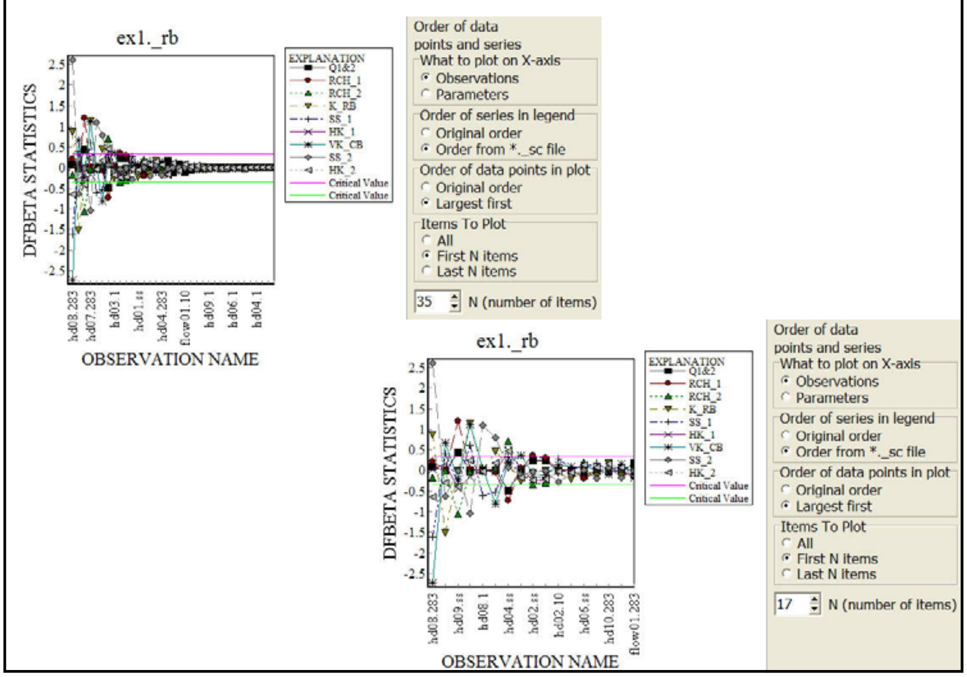
DFBetaS large values indicate observations important to individual parameters

Do you understand why the flow observation is so important? What would you be able to say about the parameter values without that observation?

A typical _rc



A typical _rb



Typical #resan results

```
*****************************************************************************
ANALYSIS OF COOKS D
FOR PLOTTING, COOKS D STATISTICS ARE LISTED IN THE _RC OUTPUT FILE

INFLUENTIAL OBSERVATIONS WITH COOKS D > CRITICAL VALUE (4/(NOBS+MPR)) = 0.114

OBS# OBSERVATION      PLOT-SYMBOL      COOK'S D
   3 hd01.283                   2        0.19920774E+00
  10 hd03.1                     2        0.16229212E+00
  11 hd03.283                   2        0.28741326E+00
  23 hd07.283                   2        0.69218923E+00
  25 hd08.1                     2        0.77273656E+00
  26 hd08.283                   2        0.19342235E+01
  27 hd09.ss                    1        0.41209141E+00
  29 hd09.283                   2        0.15028790E+00

NUMBER OF INFLUENTIAL OBSERVATIONS IDENTIFIED:    8
```

```
*****************************************************************************
ANALYSIS USING DFBETAS
FOR PLOTTING, DFBETA STATISTICS ARE LISTED IN THE _RB OUTPUT FILE

PARAMETER NUMBERS AND NAMES:

            1            2            3            4            5
            Q1&2         RCH_1        RCH_2        K_RB         SS_1
            6            7            8            9
            HK_1         VK_CB        SS_2         HK_2

INFLUENTIAL OBSERVATIONS WITH DFBETA >
                                  CRITICAL VALUE (2/(NOBS+MPR)**0.5) = 0.338

PARAMETERS INFLUENCED IDENTIFIED BY #

                                 PARAMETER NUMBER

OBS# ID               PLOT-SYMBOL    1  2  3  4  5  6  7  8  9
   3 hd01.283                   2    -  -  -  #  #  -  #  #  -
   4 hd02.ss                    1    -  #  #  -  -  -  -  -  -
  10 hd03.1                     2    -  -  -  -  -  -  #  -  -
  11 hd03.283                   2    -  -  -  #  #  -  #  #  -
  12 hd04.ss                    1    #  #  #  -  -  #  -  -  #
  23 hd07.283                   2    -  -  -  #  #  -  #  #  -
  25 hd08.1                     2    -  -  -  -  #  -  -  #  -
  26 hd08.283                   2    -  -  -  #  #  -  #  #  #
  27 hd09.ss                    1    #  #  #  -  -  #  -  -  #

NUMBER OF INFLUENTIAL OBSERVATIONS IDENTIFIED:    9
```

EVALUATING PARAMETERIZATION

High parameter correlations calls for either
Additional data that will break the correlations
Or
Reparameterization

Barring the availability of additional data, consider reparameterization e.g. USING DERIVED_PARAMETERS Block
As an example you could define
rch2=0.5*rch1 and rch3=0.1*rch1
However, notice that the true values do not have those ratios

To evaluate if correlations are too high
try starting from different values
USE PARAMETER_VALUES Block
If results are the same (parameter values fall within one standard deviation of those determined with different starting values) **correlation is not an issue**
Thus parameters are being independently estimated