calibration and uncertainty analysis: how well does a model accomplish its purpose

Jeremy T. White

U.S. Geological Survey
Texas Water Science Center

August 17, 2016

Outline

Theoretical aspects

why do we model?

because we are uncertain about unmeasured quantities

- springflow under different conditions
- brackish water-freshwater interaction
- potential subsidence in unstressed areas
- drawdown from different pumping

we need a **learning** framework to reduce uncertainty

What is calibration?
(and is calibration a **learning** framework?)

historical aspects of calibration

- computational resources were limited
- understanding of model usage was limited

∴ the best we could do **at that time**

what is calibration (thought to be)?

- “train” the model against reality
- improve the model’s representation of reality
- make the model more like reality

∴ the model simulates reality

dangers of calibration

"all models are **wrong** but some are **useful**"

Bayesian framework for uncertainty analysis
(a **learning** framework)

Bayes rule for uncertainty analysis

$$P(\theta|\mathbf{D}) \propto \mathcal{L}(\theta|\mathbf{D})P(\theta)$$

Bayes rule for uncertainty analysis

$$\underbrace{P(\theta|\mathbf{D})}_{\substack{\text{what we}\\\text{know now}}} \propto \underbrace{\mathcal{L}(\theta|\mathbf{D})}_{\substack{\text{what we}\\\text{learned}}} \underbrace{P(\theta)}_{\substack{\text{what we}\\\text{knew}}}$$

$\mathbf{D}$	data
θ	model parameters
$P(\theta)$	prior parameter probability distribution
$\mathcal{L}(\theta\|\mathbf{D})$	likelihood of the model parameters given the data
$P(\theta\|\mathbf{D})$	posterior parameter probability distribution

the Prior

$$\underbrace{P(\theta|\mathbf{D})}_{\substack{\text{what we}\\ \text{know now}}} \propto \underbrace{\mathcal{L}(\theta|\mathbf{D})}_{\substack{\text{what we}\\ \text{learned}}} \underbrace{P(\theta)}_{\substack{\text{what we}\\ \text{knew}}}$$

- a useful modeler
- what we know **and don't know** about parameters
- initial ("best guess") ("mean value") parameters
- the spread around the "best guess"

likelihood

$$\underbrace{P(\theta|\mathbf{D})}_{\substack{\text{what we}\\\text{know now}}} \propto \underbrace{\mathcal{L}(\theta|\mathbf{D})}_{\substack{\text{what we}\\\text{learned}}} \underbrace{P(\theta)}_{\substack{\text{what we}\\\text{knew}}}$$

- “model-based learning”
- information transfer observations → parameters
- smaller residuals→higher likelihood
- ugly
  - non-parametric
  - high dimensional

the Posterior

$$\underbrace{\color{red}{P(\theta|\mathbf{D})}}_{\substack{\text{what we}\\ \text{know now}}} \propto \underbrace{\mathcal{L}(\theta|\mathbf{D})}_{\substack{\text{what we}\\ \text{learned}}} \underbrace{P(\theta)}_{\substack{\text{what we}\\ \text{knew}}}$$

- combination of expert knowledge and information in data
- non-parametric
- high dimensional
- expense to fully characterize

calibration in context

$$\underbrace{P(\theta|\mathbf{D})}_{\text{what we know now}} \propto \underbrace{\mathcal{L}(\theta|\mathbf{D})}_{\text{what we learned}} \underbrace{P(\theta)}_{\text{what we knew}}$$

- calibration→maximum likelihood estimation
- a single point in parameter space
- what about other points that fit the data? (non-uniqueness)
- ignores the Prior
- ignores us, the modelers

Synthetic example

why use synthetics?

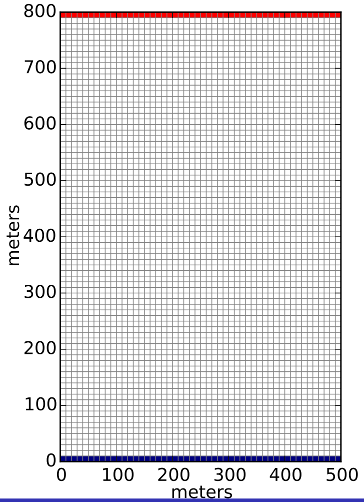
- demonstration
- runs quickly
- “cheating” (answers in the back of the book)

tools

- PEST++ V3 (USGS)
  - object-oriented version of PEST
  - integrated parallel run manager
  - global sensitivity analyses
  - builds on Windows,Linux, and Mac OS
  - **integrated linear-based uncertainty analyses**
- pyEMU (about to be accepted at EMS - hopefully)
  - python framework for linear-based uncertainty analyses
  - exploratory and data worth analyses
  - easy to use (says the guy who made it)

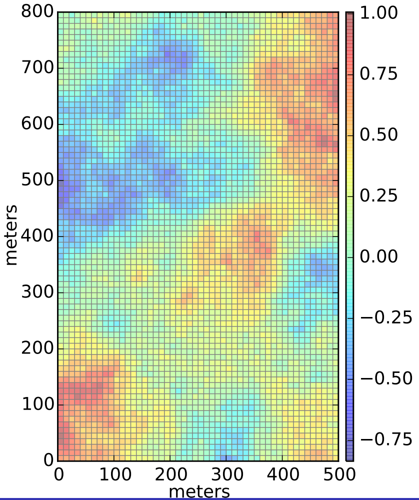
model details

- MODFLOW-2005
- MODPATH v6
- 1 stress period
- 1 layer
- 80 rows
- 50 columns



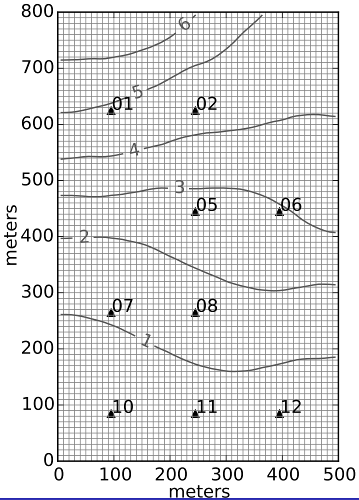
the “truth”

- sequential Gaussian simulation
- exponential variogram
- range 600.0m
- sill 0.2 $log_{10}\frac{m}{d}$



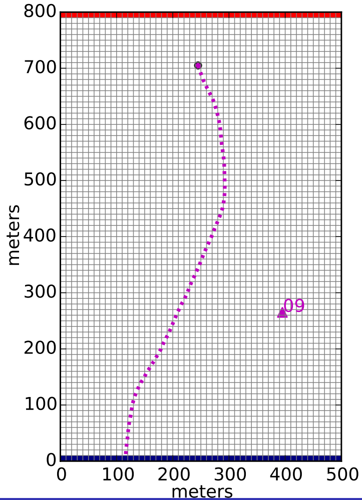
“observations”

- “observed” from the “true” model run
- heads at 9 locations
- Gaussian noise
- $\mathcal{N}(0, 0.1)$

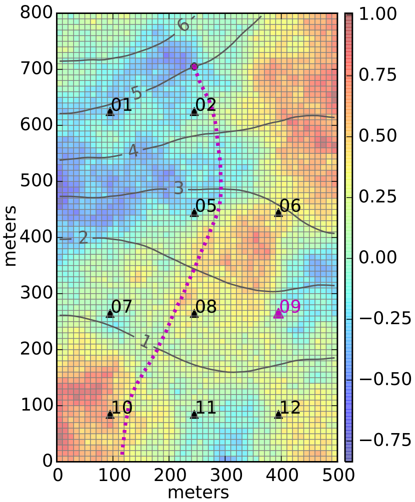


forecasts

- 1.)head at location #9
- 2.)particle exit point
- 3.)particle travel time



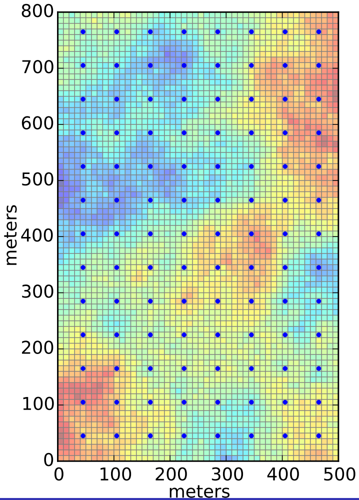
tying it all together



a tale of two models

two parameterizations

- only K is different
- 104 pilot points
- a single zone

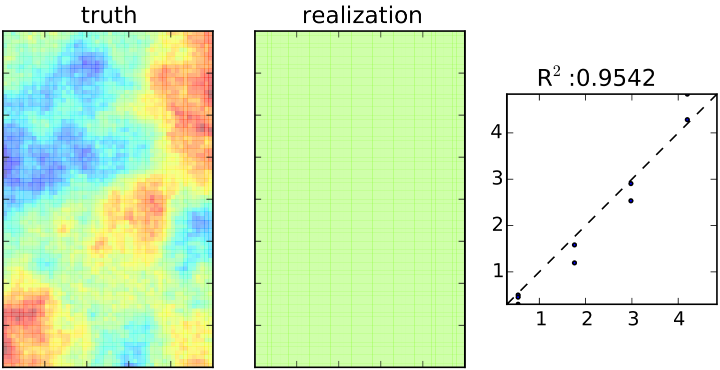


Calibration attempts

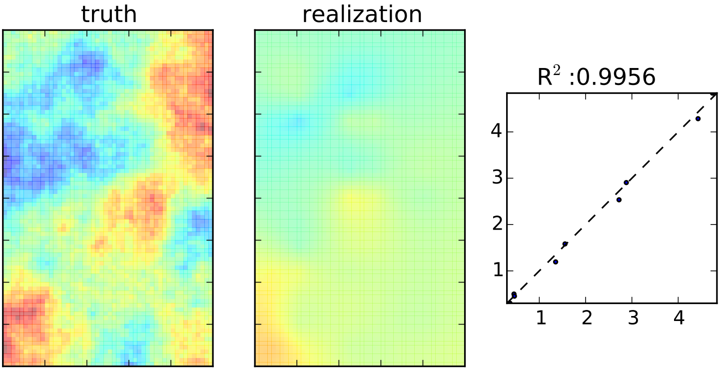
how

- PEST++ V3
- subspace and Tikhonov regularization

single parameter results

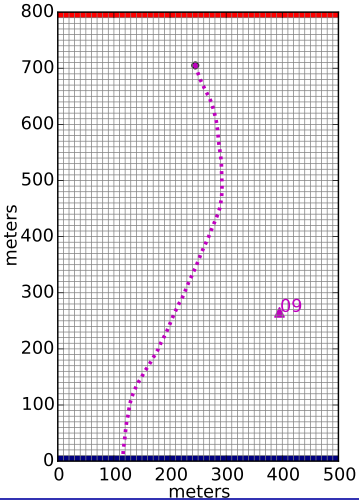


pilot point results

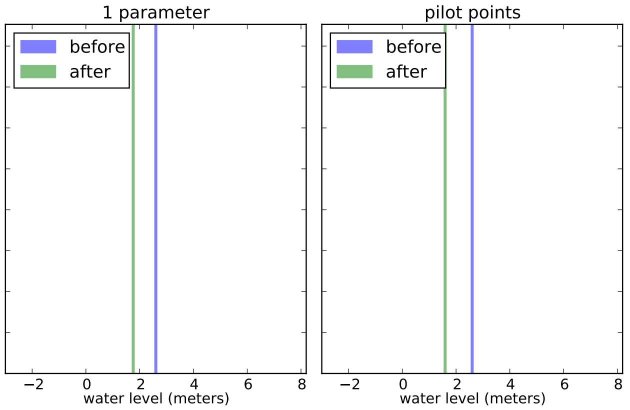


head prediction

- 1.)head at location #9
- 2.)particle exit point
- 3.)particle travel time

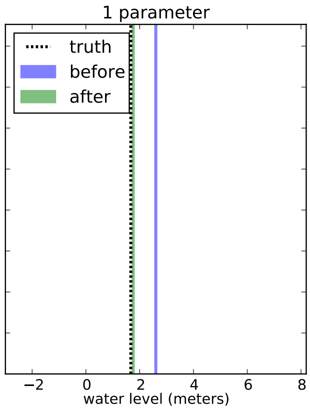


head prediction



- calibration has changed the value of the prediction from ≈2.2m to ≈ 2m

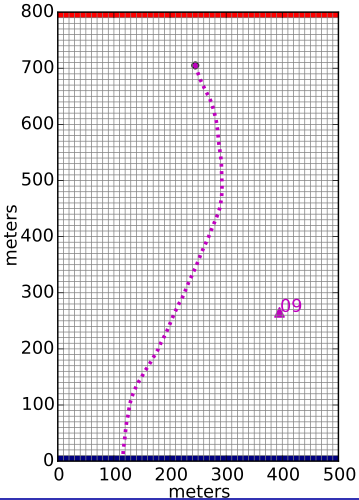
head prediction (with truth)



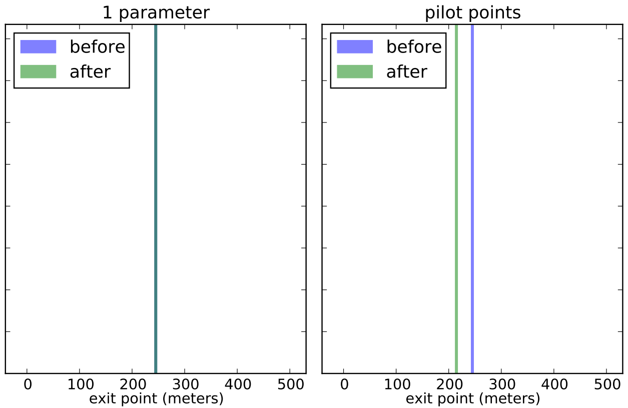
- both parameterizations calibrate to ≈ the true value

exit point prediction

- 1.)head at location #9
- 2.)particle exit point
- 3.)particle travel time

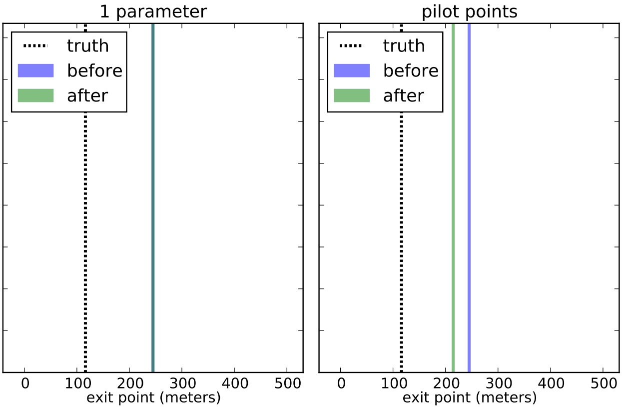


exit point prediction



- calibration of the single parameter model does not effect the prediction

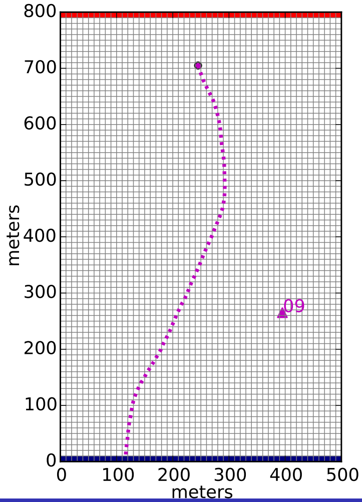
exit point prediction (with truth)



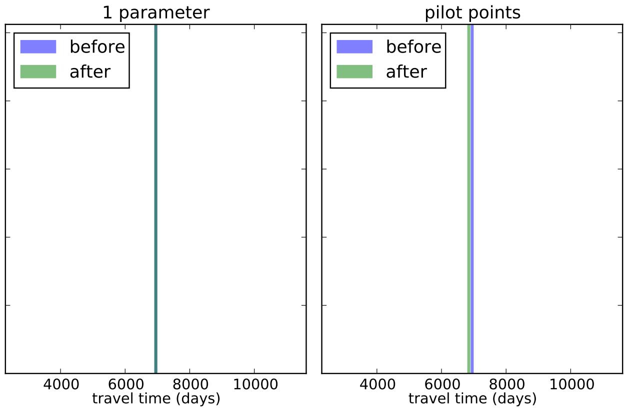
- both calibrated models are wrong

travel time prediction

- 1.)head at location #9
- 2.)particle exit point
- 3.)particle travel time

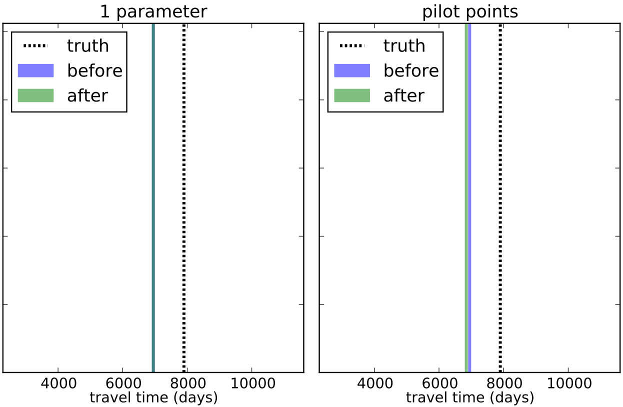


travel time prediction



- calibration does not effect the value of the prediction for either model

travel time prediction (with truth)



- both calibrated models are wrong

what did we learn about the model's purpose?

(Calibration as an analysis tool)

- before calibration, each prediction had a value
- after calibration, each prediction has a (new) value
- increasing the number of parameters doesn't have any effect on our understanding of the model's purpose

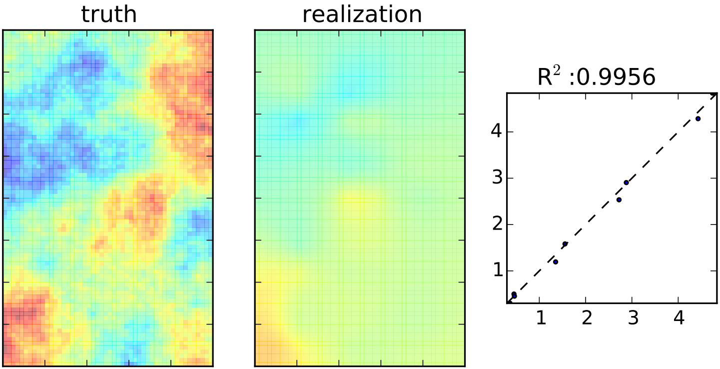
$\therefore$ we don't know what we learned about purpose

Check your work

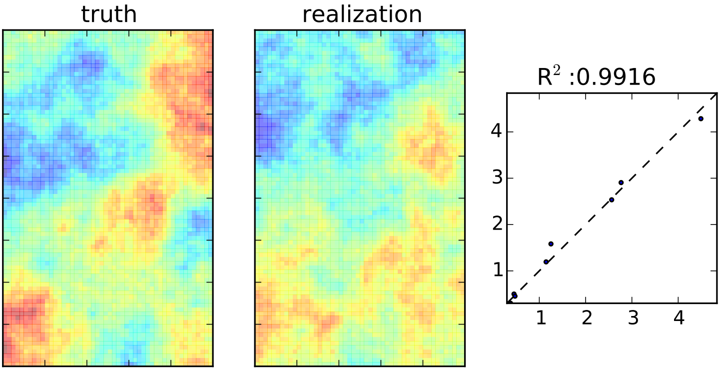
Since we know the "truth"...

Why did we not get the correct forecast?

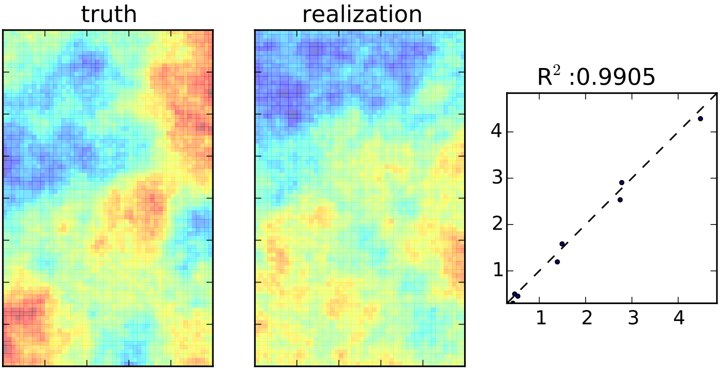
pilot point results



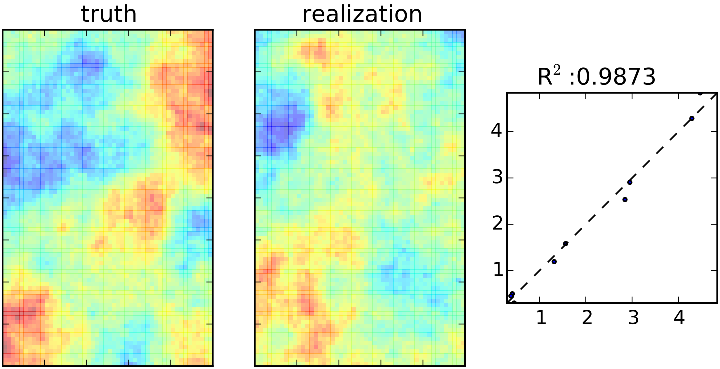
what did we learn?



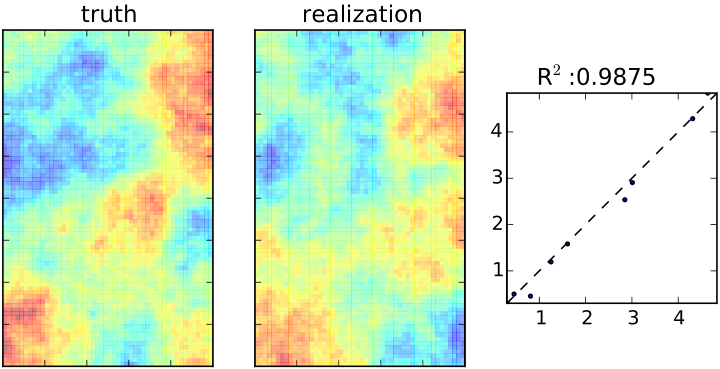
what did we learn?



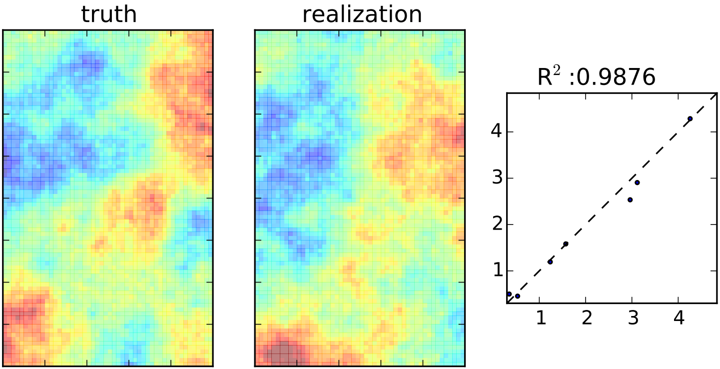
what did we learn?



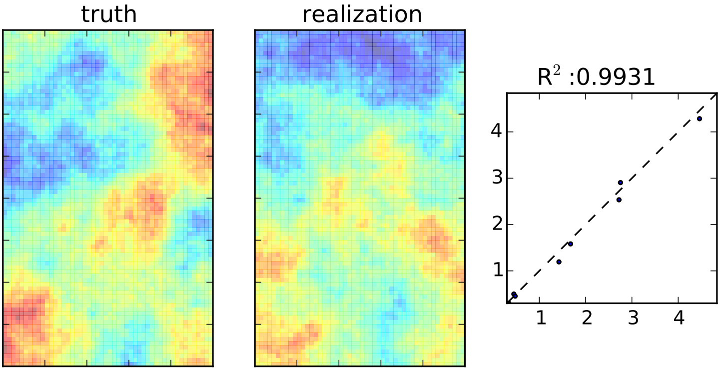
what did we learn?



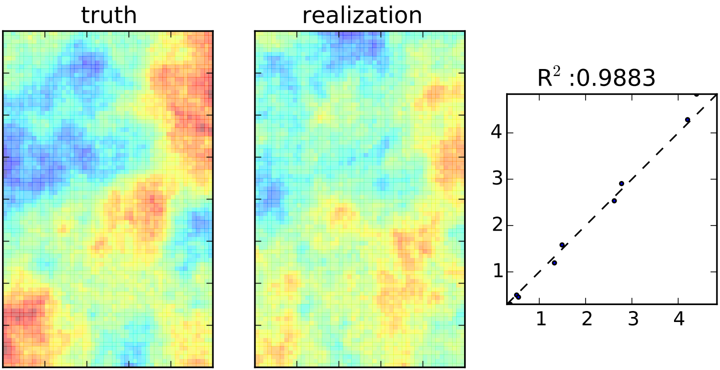
what did we learn?



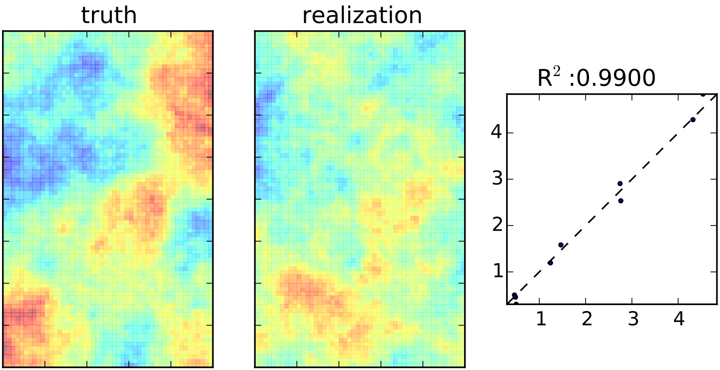
what did we learn?



what did we learn?



what did we learn?



Uncertainty analysis

Bayes for GW models

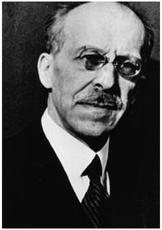
$$P(\theta|\mathbf{D}) \propto \mathcal{L}(\theta|\mathbf{D})P(\theta)$$

- in its purist form, Bayes is very difficult to apply to GW models
  - lots of parameters
  - long model run times
- so...we make some simplifying assumptions:
  - linear-based uncertainty analysis

background on linear theory (FOSM)

- **F**irst **O**rder **S**econd **M**oment
- **FO**→linearity assumption
- **SM**→multivariate Gaussian posterior assumption
- $P(\theta|\mathbf{D}) \approx \mathcal{N}(\overline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}, \overline{\boldsymbol{\Sigma}}_{\boldsymbol{\theta}})$
- requires only one run per parameter!





Schur's complement

$$\underbrace{\overline{\boldsymbol{\Sigma}}_{\boldsymbol{\theta}}}_{\text{what we still don't know}} = \underbrace{\boldsymbol{\Sigma}_{\boldsymbol{\theta}}}_{\text{what we didn't know}} - \underbrace{\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\mathbf{J}^{\mathbf{T}}\left[\mathbf{J}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\mathbf{J}^{\mathbf{T}} + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}\right]^{-1}\mathbf{J}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}}_{\text{what we learned}}$$

$\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$	prior parameter covariance matrix
$\mathbf{J}$	Jacobian matrix
$\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$	measurement noise covariance matrix
$\overline{\boldsymbol{\Sigma}}_{\boldsymbol{\theta}}$	posterior parameter covariance matrix

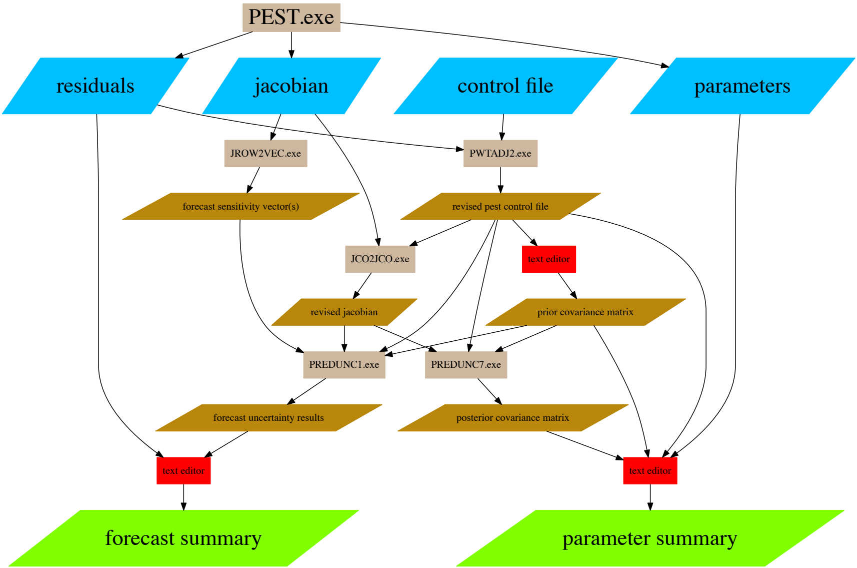
parameters → forecasts

$$\sigma_s^2 = \mathbf{y}^\mathbf{T} \mathbf{\Sigma}_{\boldsymbol{\theta}} \mathbf{y}$$

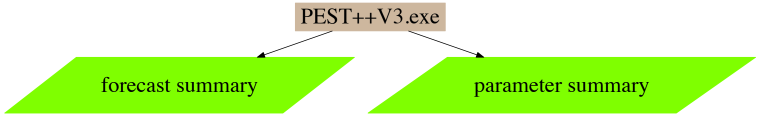
$$\overline{\sigma}_s^2 = \mathbf{y}^\mathbf{T} \overline{\mathbf{\Sigma}}_{\boldsymbol{\theta}} \mathbf{y}$$

σ_s^2	prior variance of forecast s (what we didn't know)
$\overline{\sigma}_s^2$	posterior variance of forecast s (what we still don't know)
$\mathbf{y}$	forecast sensitivity vector

implementation - Legacy PEST utilities



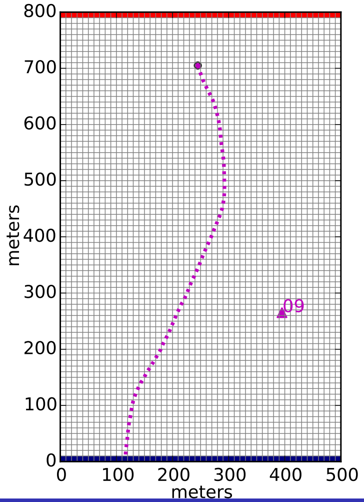
implementation - PEST++ V3



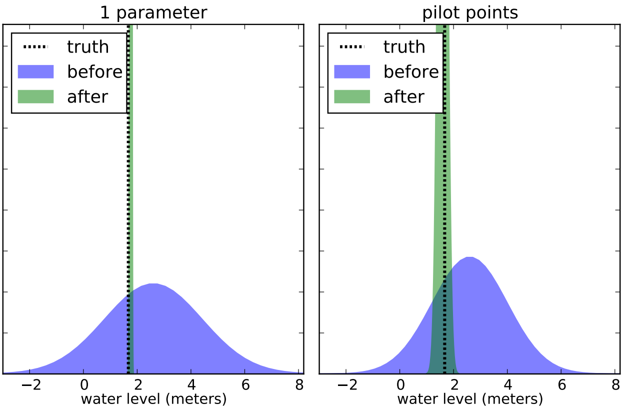
- no additional model runs required!
- no reason not to use this valuable information

head forecast

- 1.)head at location #9
- 2.)particle travel time
- 3.)particle exit point



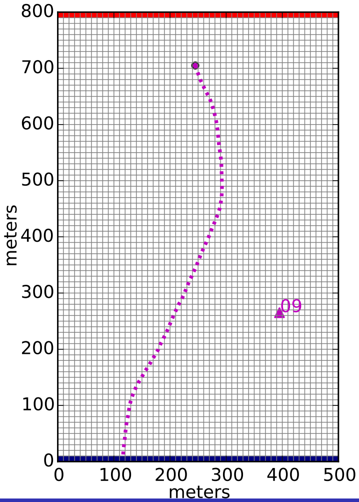
uncertainty: head forecast



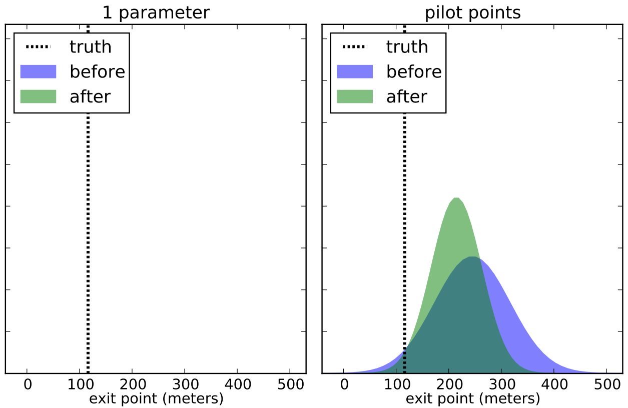
- calibration of both models reduces uncertainty
- pilot points model bracket the ”true” value

exit point forecast

- 1.)head at location #9
- 2.)particle exit point
- 3.)particle travel time



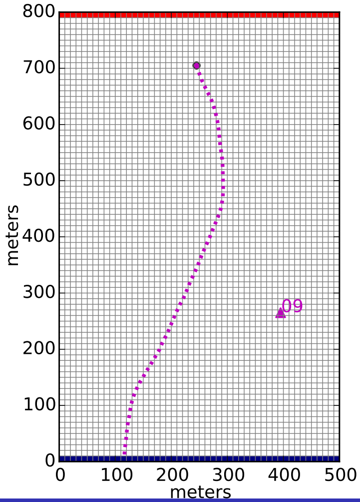
uncertainty: exit point forecast



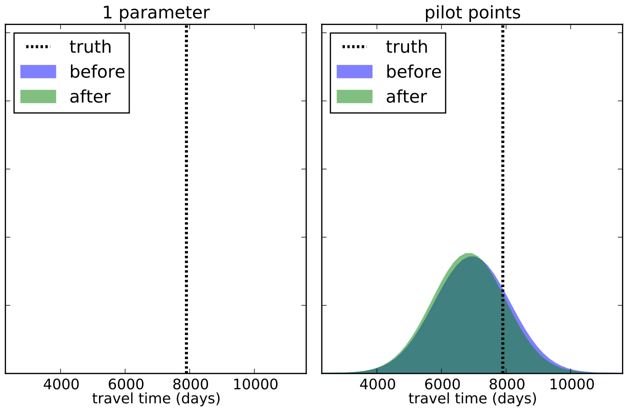
- 1 parameter model grossly underestimates uncertainty
- calibration only slightly reduces uncertainty

travel time forecast

- 1.)head at location #9
- 2.)particle exit point
- 3.)particle travel time



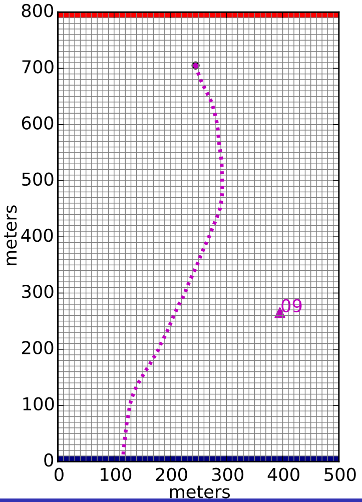
uncertainty: travel time forecast



- 1 parameter model grossly underestimates uncertainty
- calibration does not reduce uncertainty

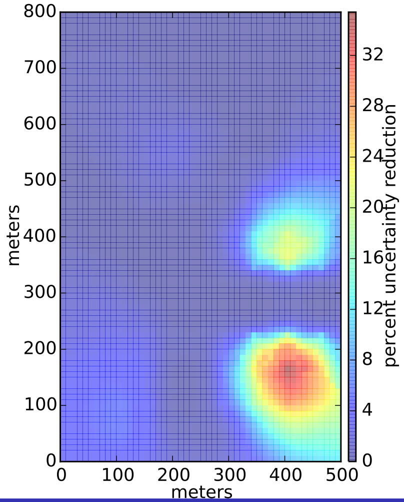
data worth analyses

- ”what if...” scenarios
- **data that we don’t have!!!**
- guide data acquisition
- where to ”learn” about K
  - most reduce uncertainty
  - **model purpose**
- forecast specific
- no additional run required!
  - **save $**



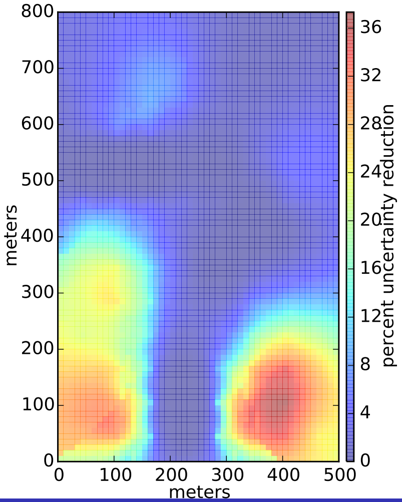
exploratory analysis: head forecast

- knowing K in the right place can reduce forecast uncertainty by 35%
- down gradient of forecast



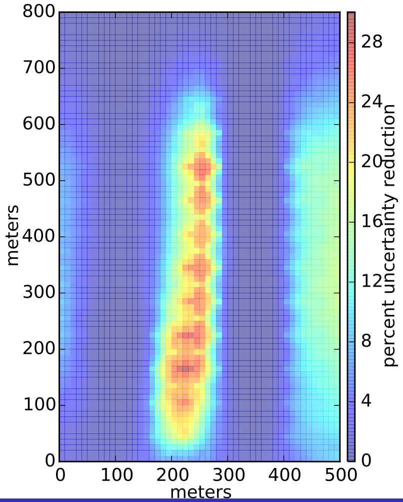
exploratory analysis: exit point forecast

- knowing K in the right place can reduce forecast uncertainty by 37%
- up gradient of forecast

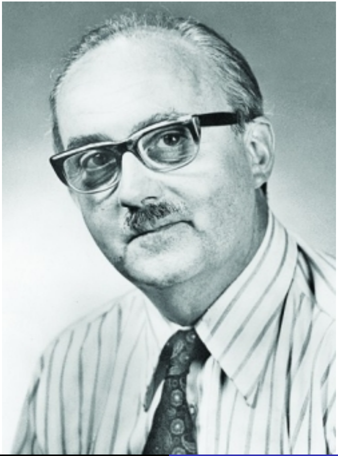


exploratory analysis: travel time forecast

- knowing K in the right place can reduce forecast uncertainty by 30%
- centerline of travel path



"all models are **wrong** but some are **useful**"



Why calibration is dangerous

- The **goal** of calibration: get the “right” answer
  - yikes!

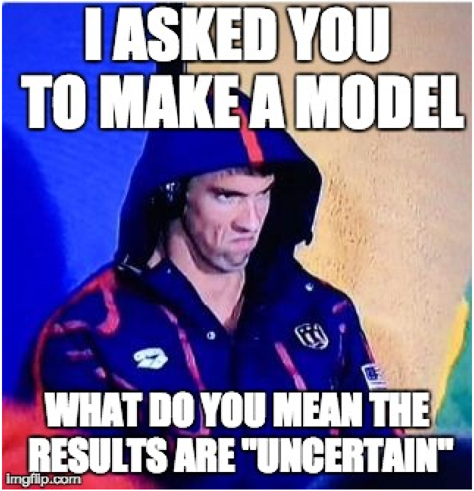
Calibration doesn’t improve understanding (purpose)

why uncertainty analysis is better

- The **goal** of uncertainty analysis: bracket the "right" answer
  - a much better chance of being "right"
- **uncertainty analysis directly addresses learning**

data worth analyses

- great complement to uncertainty analysis
- super powerful tool for modeling analyses
- maximize value of expensive data collection
- very easy to apply
  - no additional model runs
  - pyEMU
- directly answers questions related to improved understanding of model purpose


I ASKED YOU
TO MAKE A MODEL
WHAT DO YOU MEAN THE
RESULTS ARE "UNCERTAIN"
imgflip.com