Machine learning techniques in predicting uncertainty of environmental models

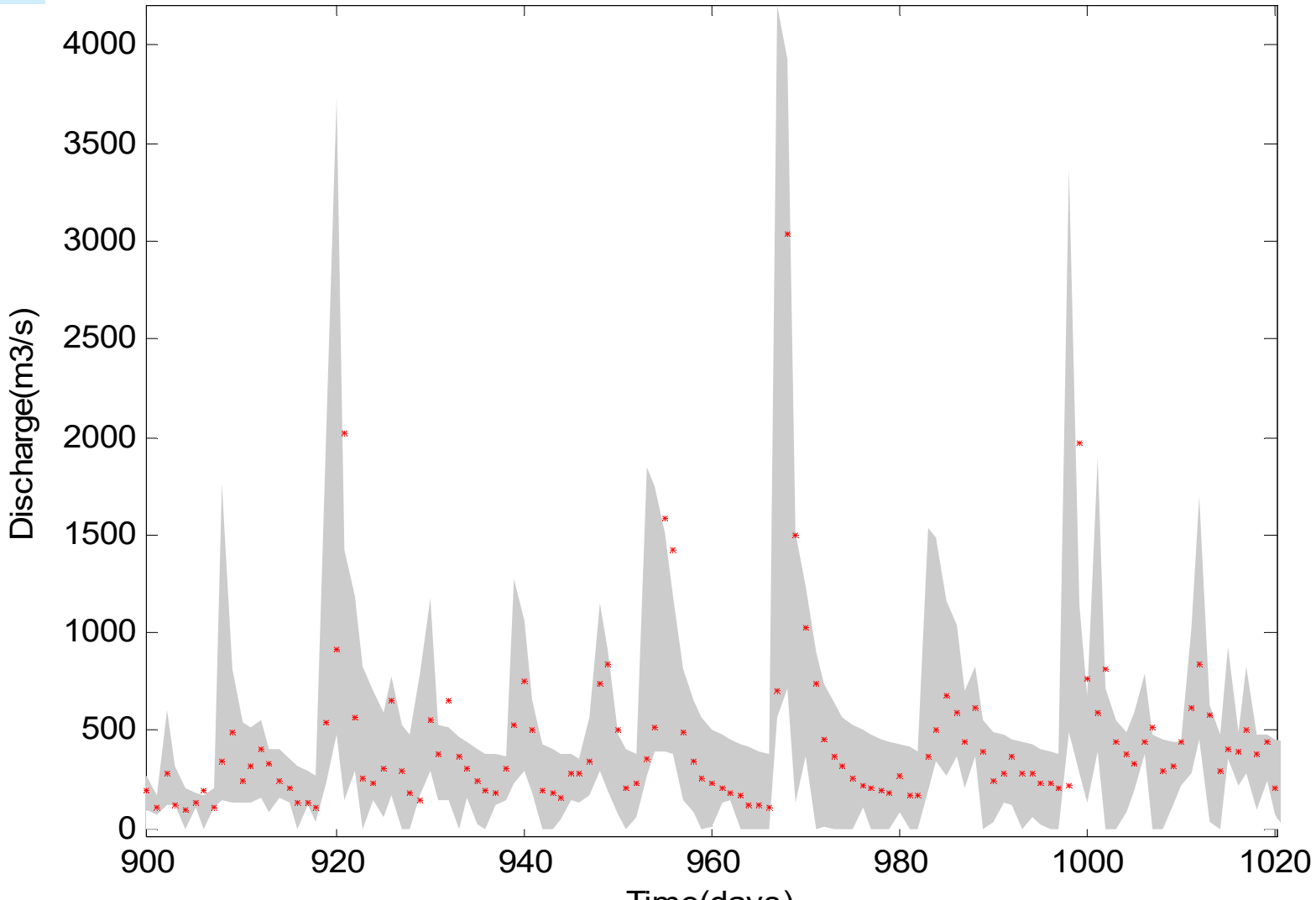
Dimitri Solomatine

Professor of Hydroinformatics,
IHE Delft Institute for Water Education
Delft, The Netherlands

Outline

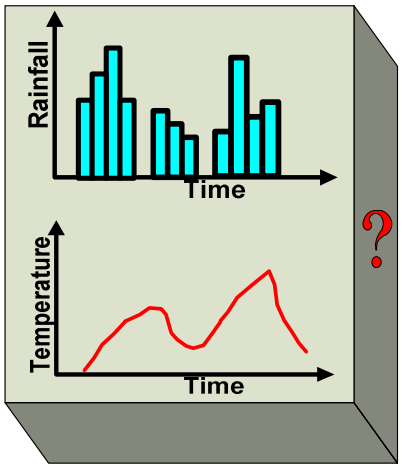
- Introduction: what are analysisng?
- Machine learning methods to (a) analyse and (b) predict the model uncertainty
- Suggested approach: “escalation” of uncertainty
- Examples

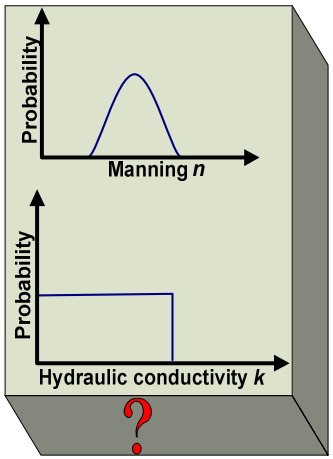
Example for a quick start: deterministic forecasts and 90% uncertainty bounds

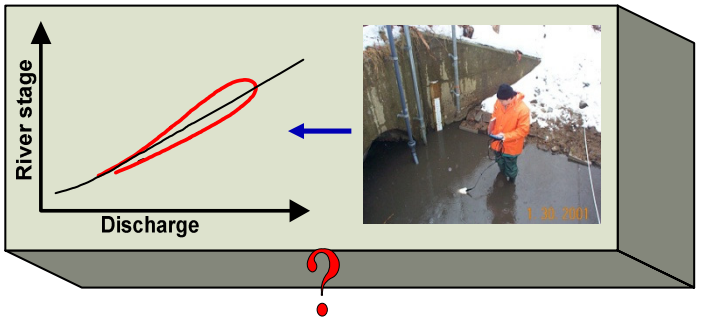


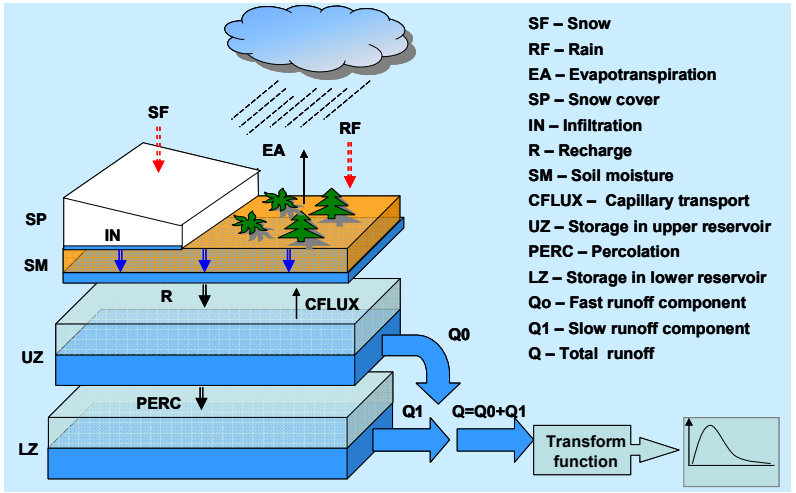
Sources of model uncertainty: perceptual, structure, parameters, data

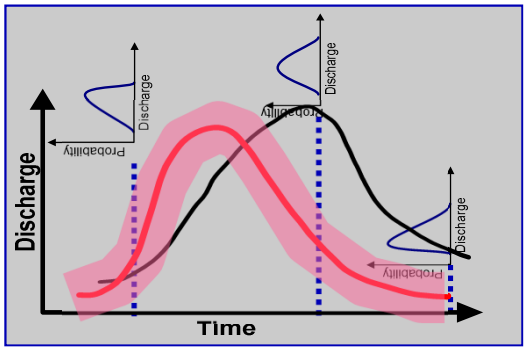
$$y = M(x, p) + \varepsilon_s + \varepsilon_\theta + \varepsilon_x + \varepsilon_y$$











Traditional steps in uncertainty analysis of a calibrated model

- Identification of sources of uncertainty (input, parameter, model structure)
- Quantification of uncertainty (e.g. as distribution)
- Studying *propagation of uncertainty* through the model (e.g. by Monte Carlo simulation)
- Quantification of uncertainty in the model outputs (i.e. identification of output distribution (pdf) or its characteristics – mean, st.dev., quantiles)
- If possible, reduction of uncertainty (e.g. model improvement, more accurate measurements, etc.)
- Application of the uncertain information in decision making process

Data uncertainty (input, parameters): propagation of uncertainty through the model

- $y\hat{} = M(x, p)$
- x = input, p = parameters
  - Uncertainty in X and p *propagates* to output y
  - pdf of parameters → pdf of output $pdf_p \rightarrow pdf_y$
  - pdf of inputs pdf_x → pdf of output $pdf_x \rightarrow pdf_y$

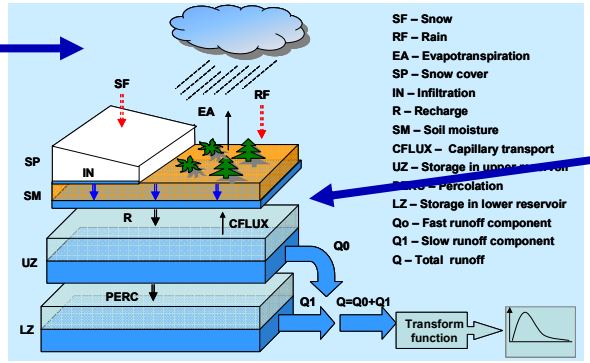
Monte Carlo Simulation

Mote Carlo casino: roulette wheel

- It is a random number generator – uses uniform distribution with the range of [0, 36]

Single model run (no uncertainty)

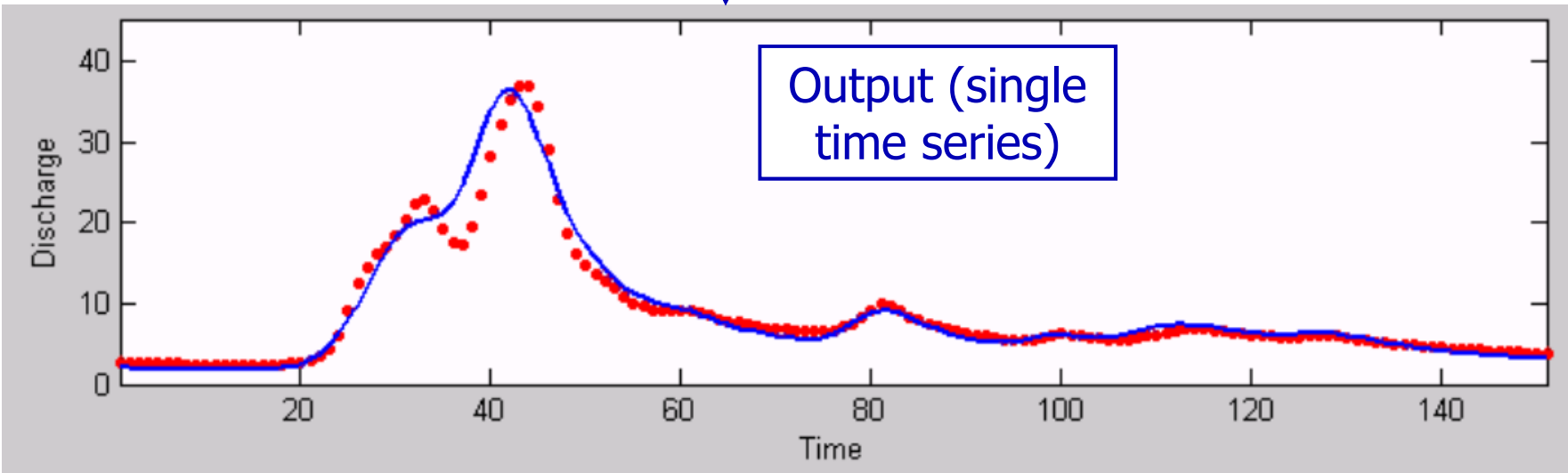
Input (single time series)



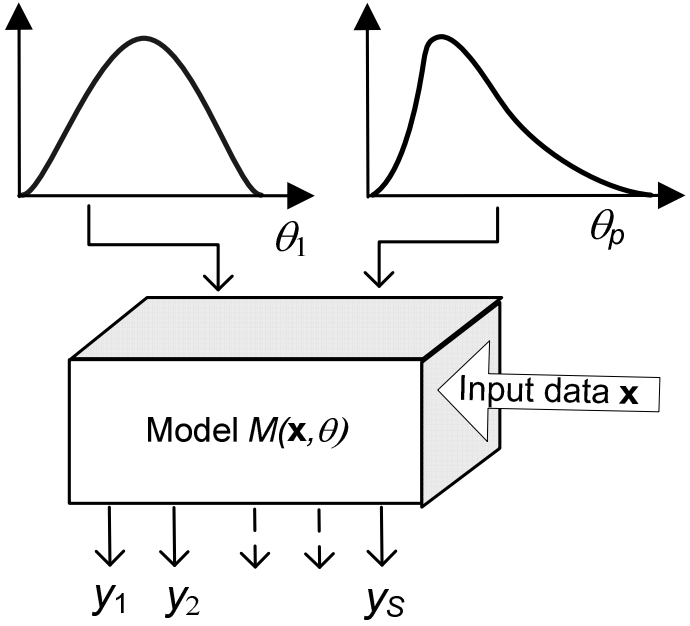
Single parameter vector P:
FC, ALPHA, K, MAXBAS, etc.

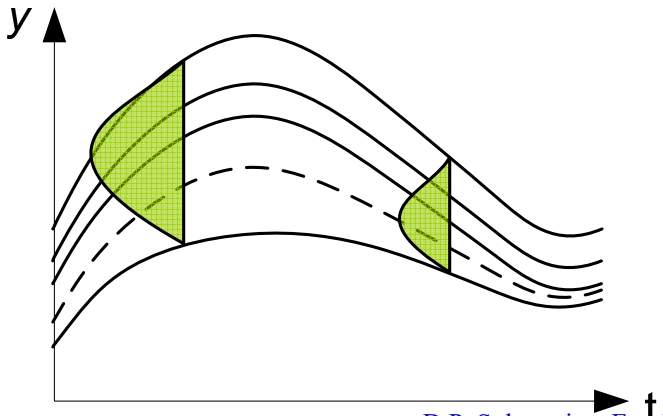
Run the model

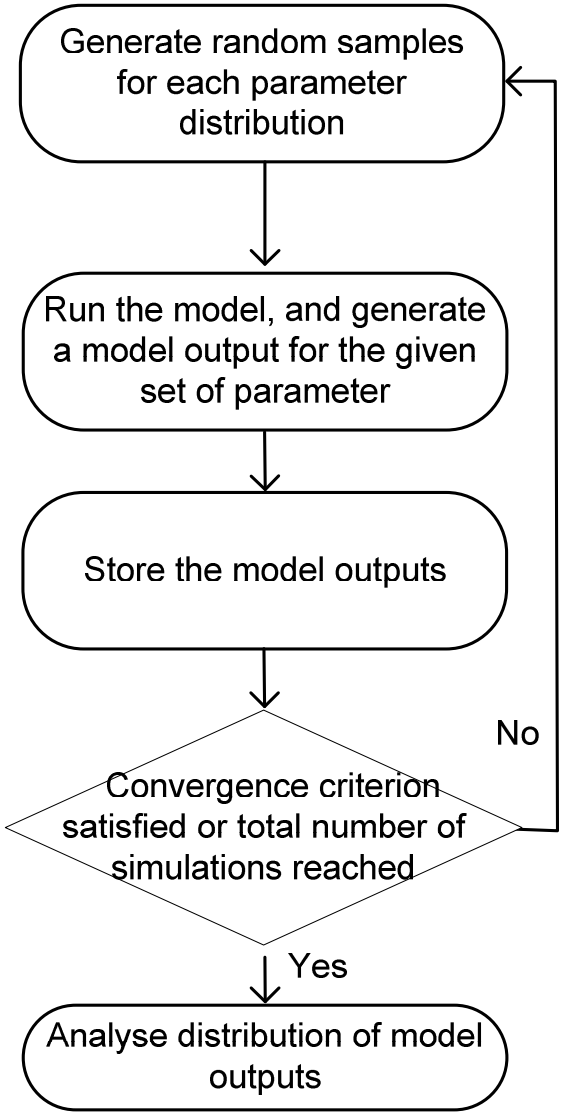
Output (single time series)



Monte Carlo simulation in analysing parametric uncertainty







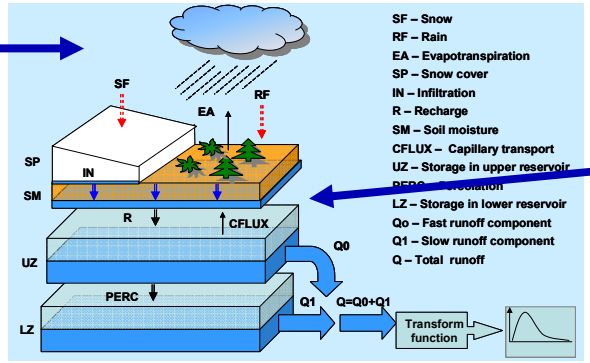
Sampling parameters and multiple model runs

Do this muliple times

Sample one parameter vector from distributions

P

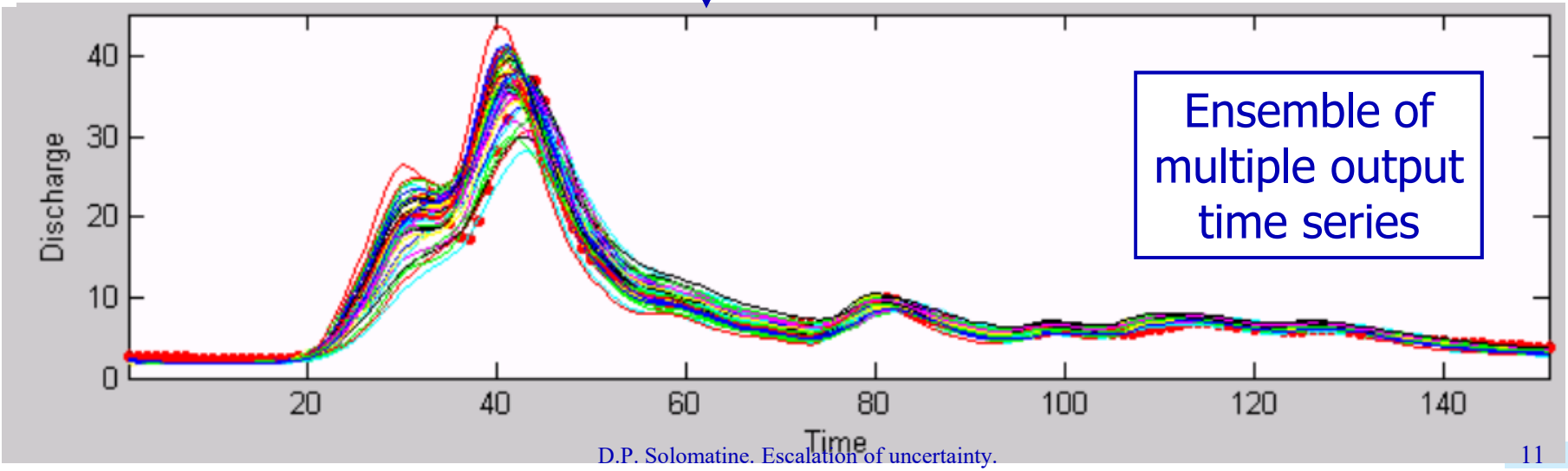
Input (single time series)



Single parameter vector P: FC, ALPHA, K, MAXBAS, etc.

Run the model

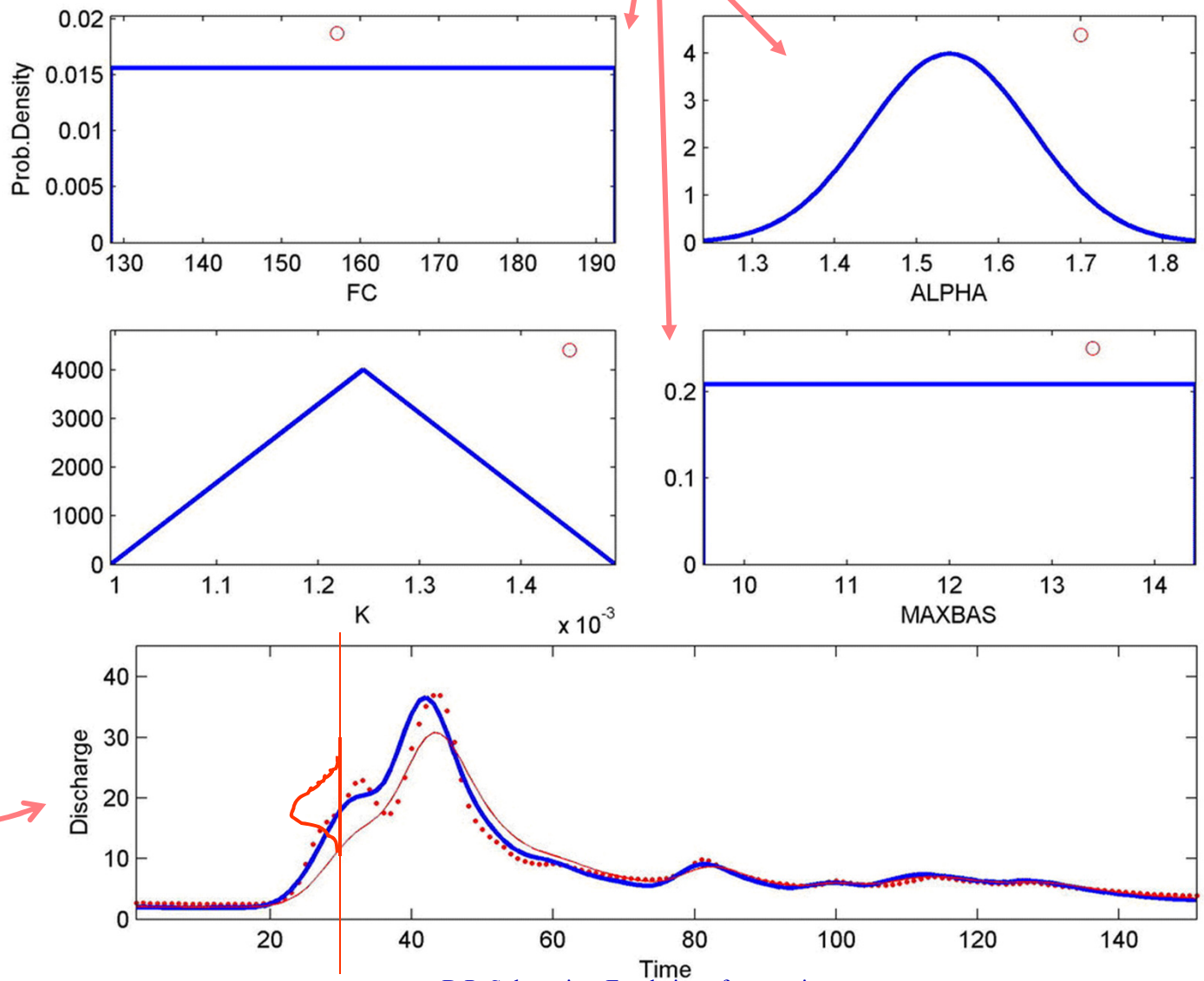
Single output



Ensemble of multiple output time series

Monte Carlo sampling: illustration

$$y = M(x, s, \theta) + \varepsilon_s + \varepsilon_\theta + \varepsilon_x + \varepsilon_y$$

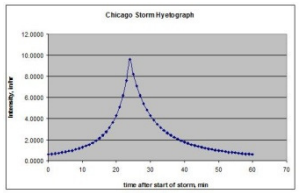


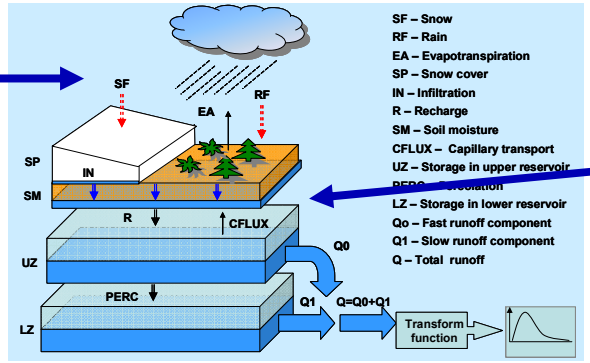
Sampling rainfall and multiple model runs

Sample one input time series from distributions

Do this muliple times

Input (single time series)

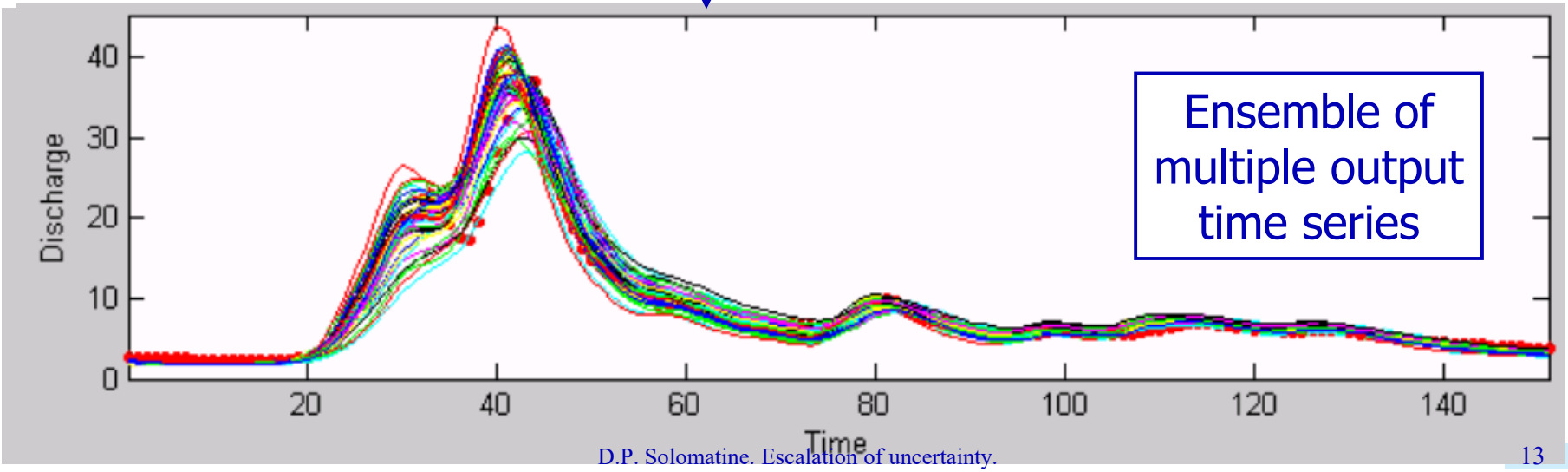




Single parameter vector P: FC, ALPHA, K, MAXBAS, etc.

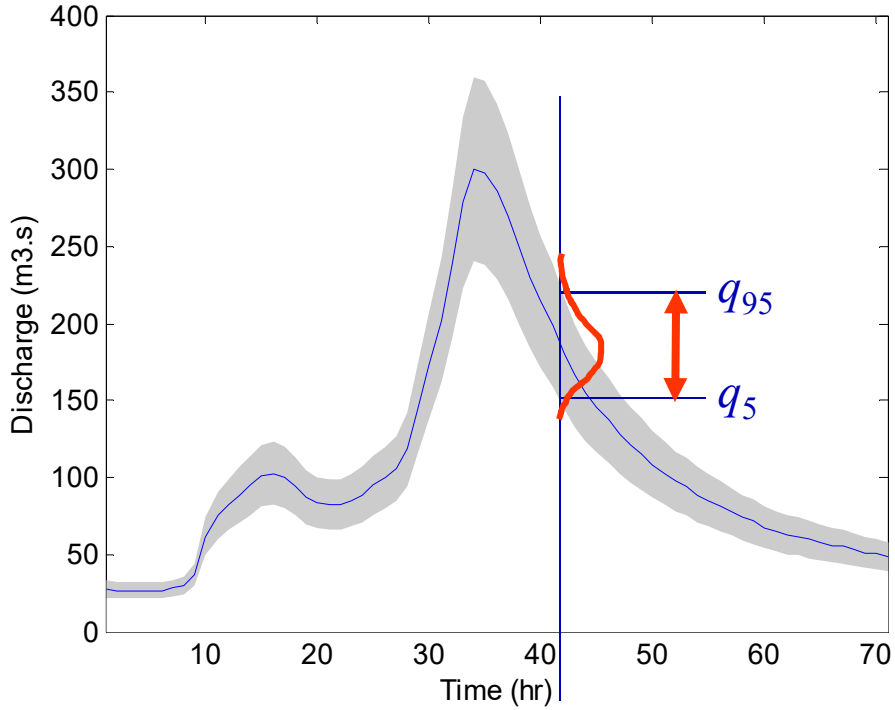
Run the model

Single output



Ensemble of multiple output time series

Representing uncertainty of model output by the confidence bounds



Instead of fitting a theoretical distribution, we can use mean, standard deviation, quantiles.

E.g., 5% and 95% form the **90% confidence bounds**

Propagation of parameters/data uncertainty
by Monte Carlo simulation
is a typical practical approach.

But is it the only one?

QUESTION 1.
On assumptions

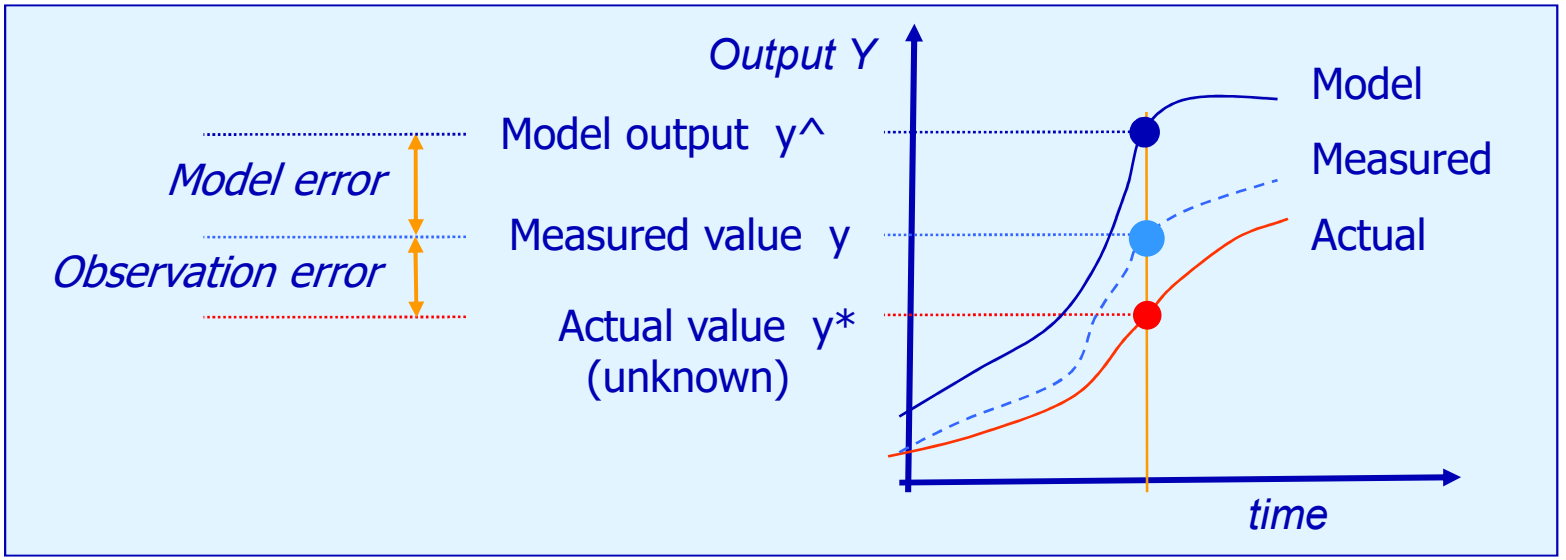
- We are assuming some known distributions of parameters or inputs. How safe is this?
- Could we take a safer route and assume less?

- Let's make a step backwards and pose the

  QUESTION 1:

  what is the uncertainty of the calibrated model itself?

Residual uncertainty: uncertainty of a calibrated ("optimal") model

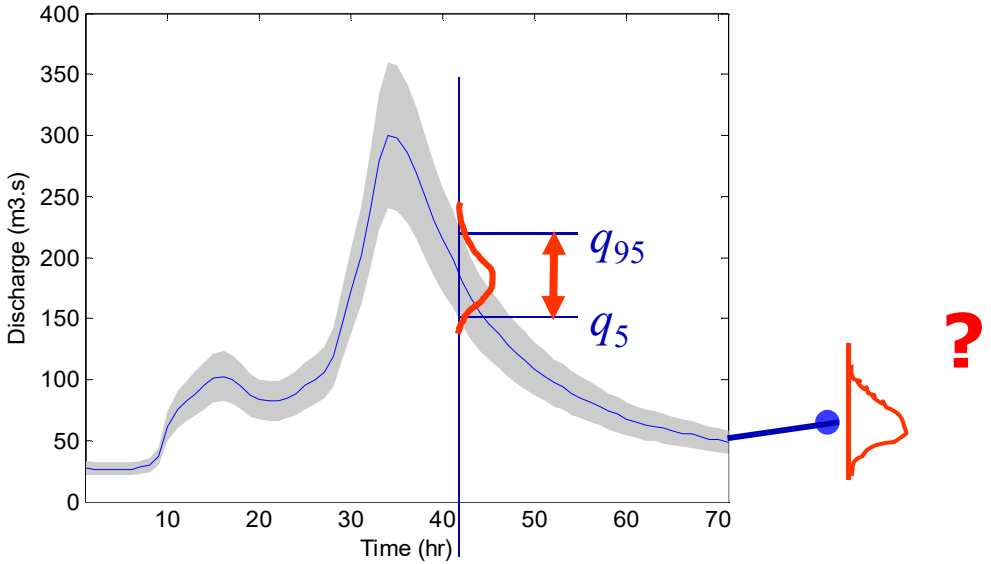


- Uncertainty of an *optimal model* M (x, θ)
  - Model M is calibrated on measured data y
  - We say the model M uncertainty is manifested in the residual model error ε = y^ – y
  - This error incorporates all uncertainties due to: observational errors, inaccurately estimated parameters, inadequate model structure

ESCALATION ("build up") of model uncertainty [message 1]

- 1. Study the (residual) uncertainty of an *optimal model* M (p*)
- 2. Add and study (typically, by MC simulation)
  - A) uncertainty of M (p*) due to DATA uncertainty
  - B) uncertainty of M (p) due to PARAMETERS uncertainty
- 3. Add and study uncertainty of M (p) due to STRUCTURAL uncertainty
- 4. Study uncertainty of a *model class* M (p), given the probabilistic properties of parameters and data

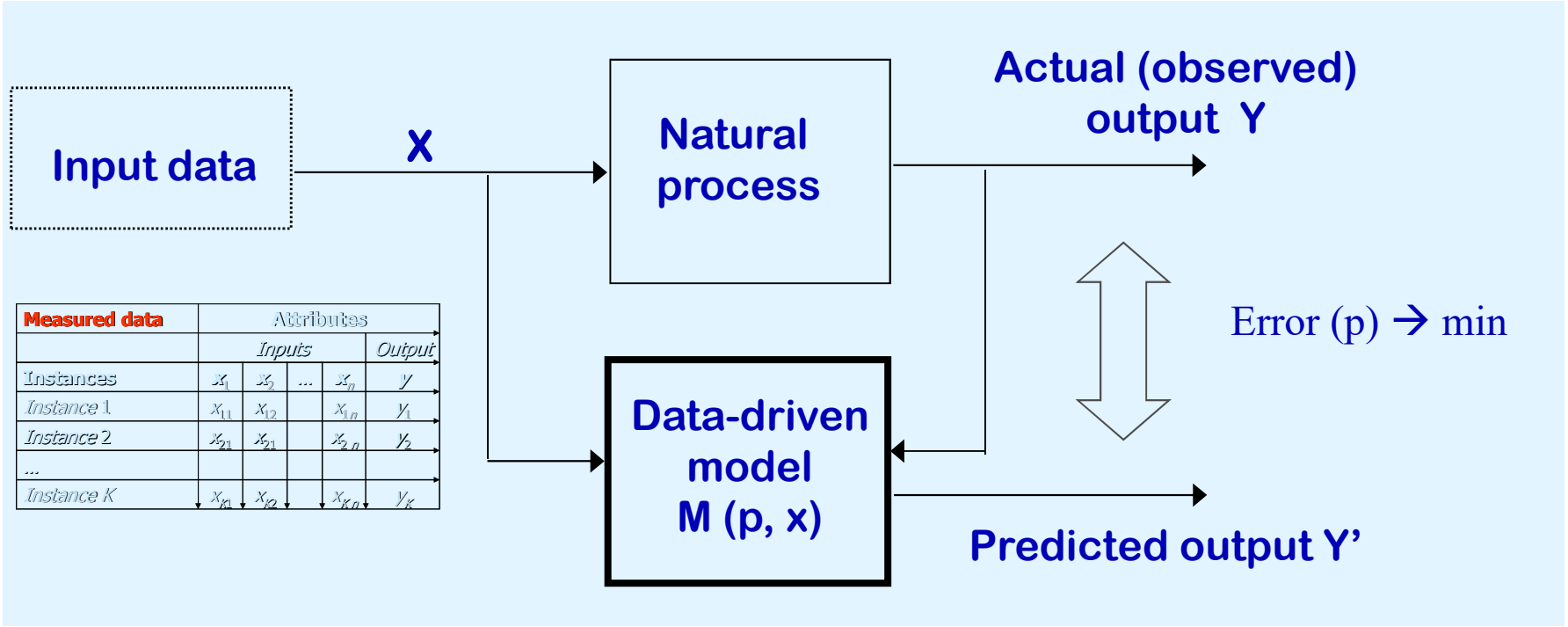
QUESTION 2.
On what is analysed



- In UA we always use the past data, so Estimates of uncertainty are about the PAST.
- QUESTION 2:
  how can we assess the model uncertainty for new inputs, i.e. for the future?
  - and this question we pose for all sources of uncertainty (and not only residual)

Models of Residual Uncertainty : Using Methods of Computational Intelligence

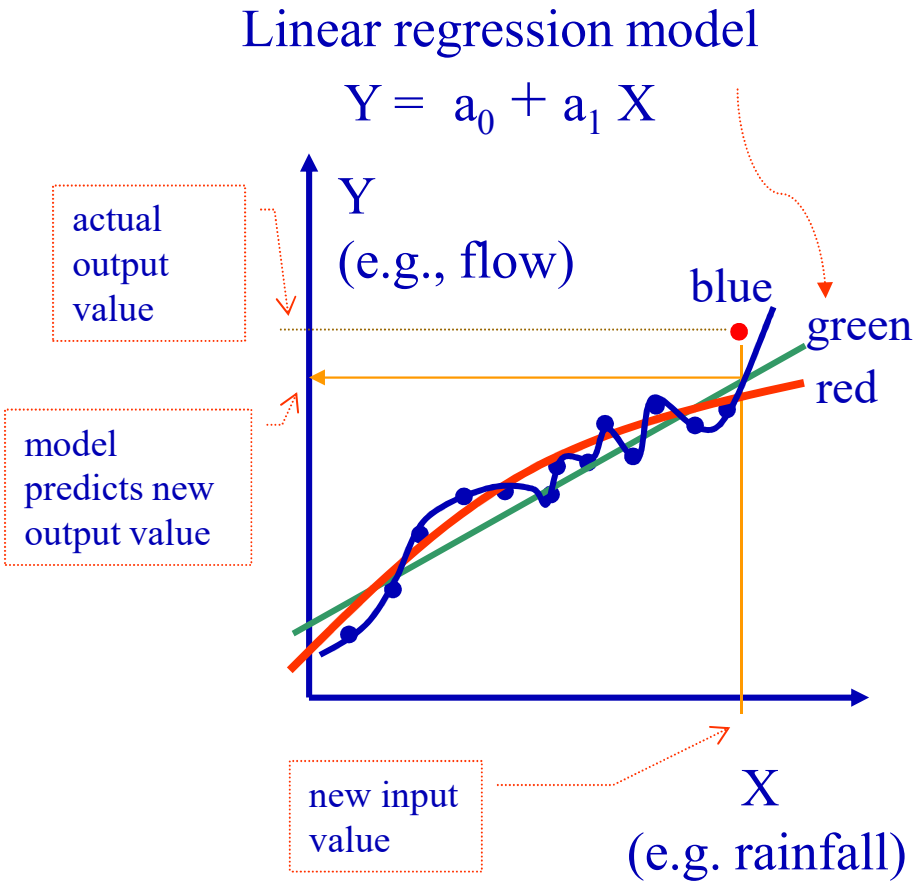
CI in building models of natural processes - why not build a model of uncertainty?



- CI provides methods to build *Data-driven* models
- Ideally, such models are "ultimate models" since they are not polluted by theories

Example of a data-driven (statistical, CI) model

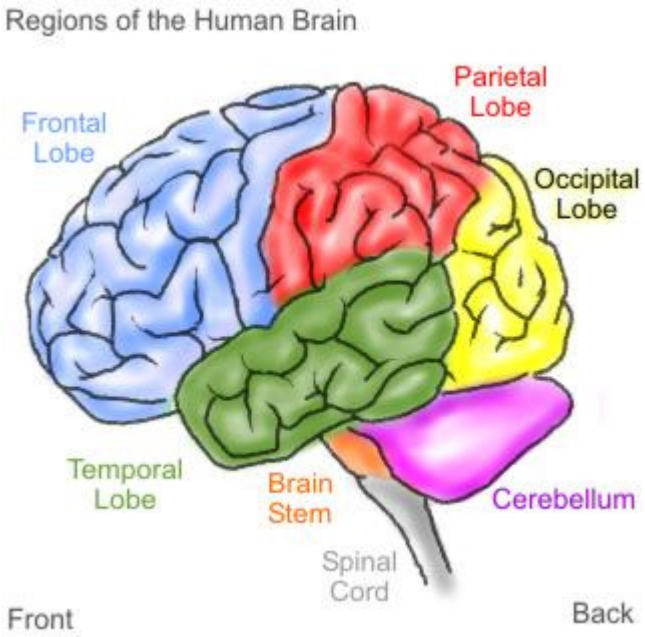
- observed data characterises the input-output relationship
  X → Y
- model parameters are found by optimization
- the model then predicts output for the new input without actual knowledge of *what* drives Y

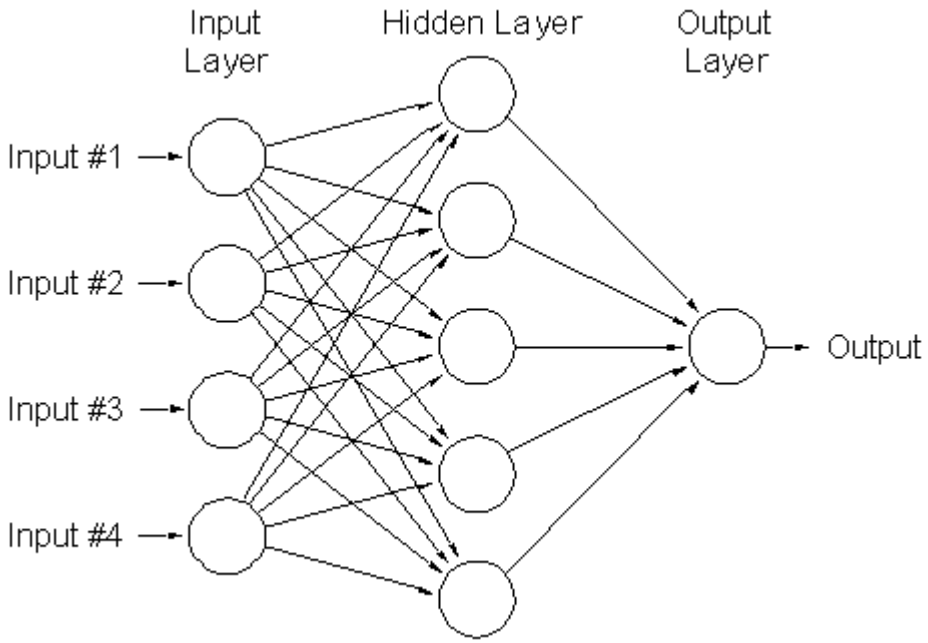


Which model is "better":
green, red or blue?

CI models: are they indeed intelligent?

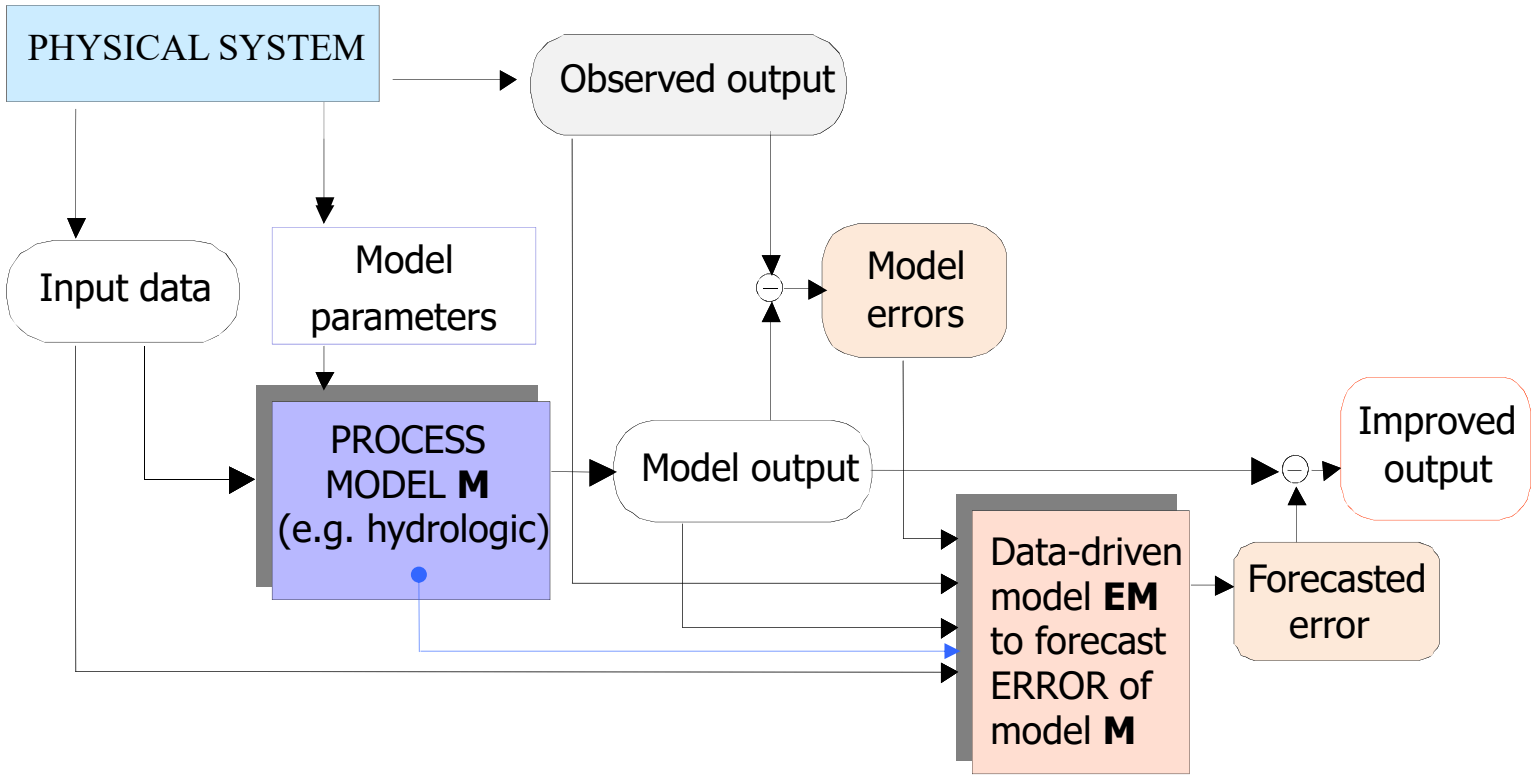
Artificial neural network



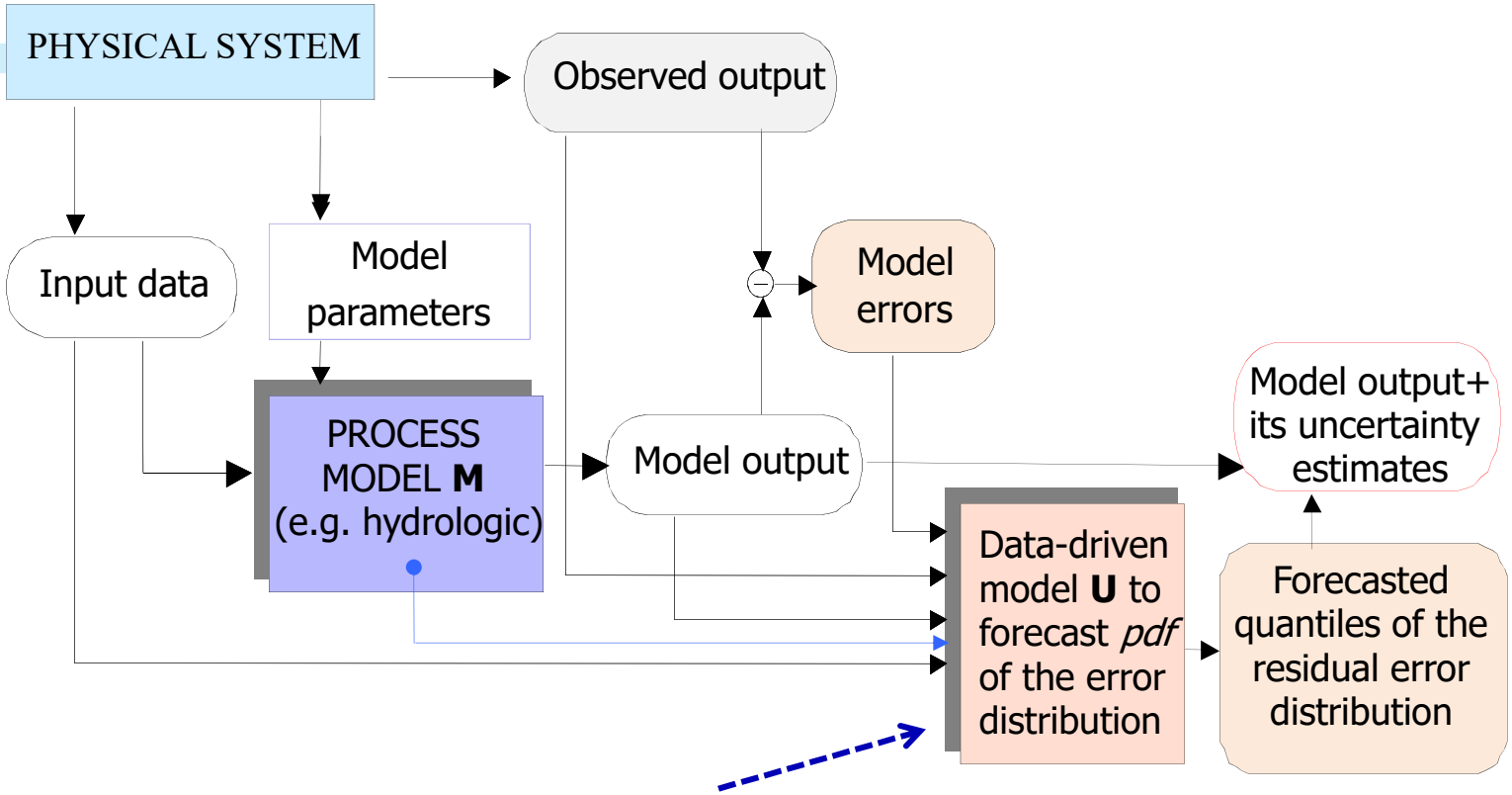


$$Y = g^{out}(b_{0k} + \sum_j b_{jk}\ g^{hid}(a_{0j} + \sum_i a_{ij} x_i^{(t)}))^2 \qquad where \quad g(u) = \frac{1}{1 + e^{-\alpha u}}$$

Data-driven model as an *error corrector* for a process (physically-based) model



Data-driven model to predict the residual error distribution



Train data-driven model (e.g. Neural Network) to forecast residual error *pdf* (*i.e. the model **M** output uncertainty*)

Some of the models of residual uncertainty

- **QR** (1978) (quantile regression): autoregressive linear model of model residuals predicts the distribution quantiles [Koenker & Basset]
- **DUMBRAE** (2012) (Dynamic Uncertainty Model By Regression on Absolute Error) [Pianosi & Raso]: autoregressive model of model residuals (it corrects the model residual first and then carries out the uncertainty prediction by an autoregressive model)
- **UNNEC** (2006, 2009) (UNcertainty Estimation based on local Errors and Clustering) [Shrestha & Solomatine]: it takes into account all variables influencing such uncertainty and uses machine learning (non-linear) methods (neural networks, model trees, instance-based learning etc.)

UNEEC method

UNcertainty Estimation based on local Errors and Clustering

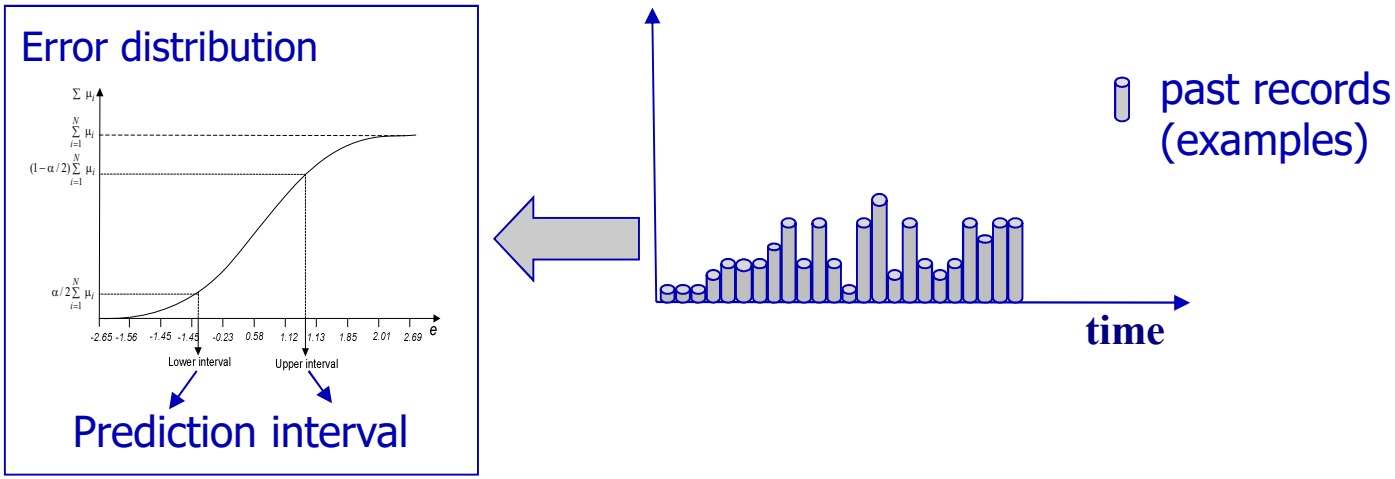
- machine learning model of the *past residual errors of the optimal process model* is built

D.P. Solomatine, D.L. Shrestha (2009). A novel method to estimate model uncertainty using machine learning techniques. *Water Resources Res*. 45, W00B11.

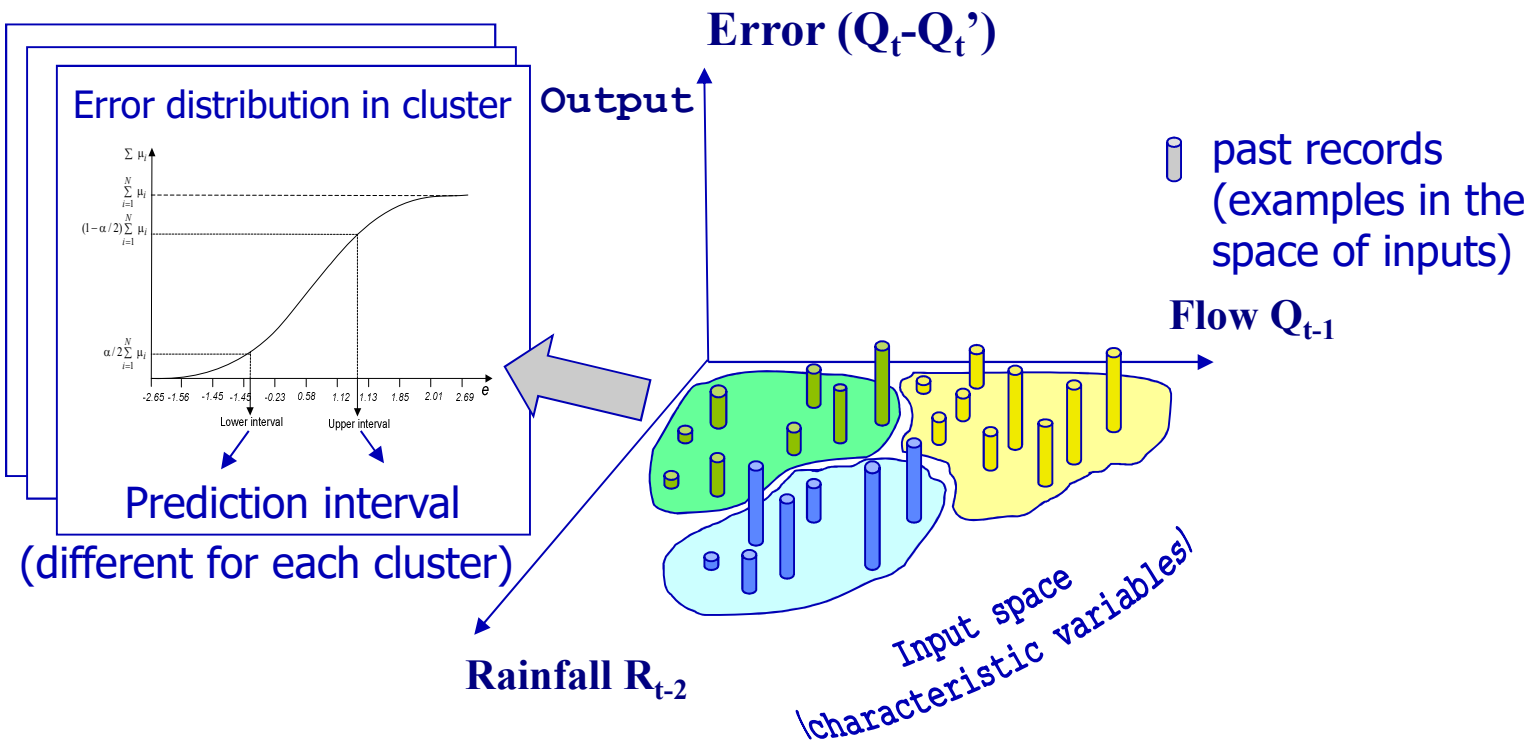
UNEEC: assumptions, constraints

- Assumptions
  - Model error is an indicator of the model uncertainty
  - Model error depends on the current condition of a natural system and can be predicted
  - Model errors are similar for similar conditions
- Constraints
  - Model structure and parameters are fixed
  - Need to re-train the error model with the changes in the catchment characteristics (e.g. land use change)
  - Data hungry, more data are needed for reliable results

Idea 1: assess CDF from all historical data about errors



Idea 2: local modelling of errors (link CDF to "characteristis variables")

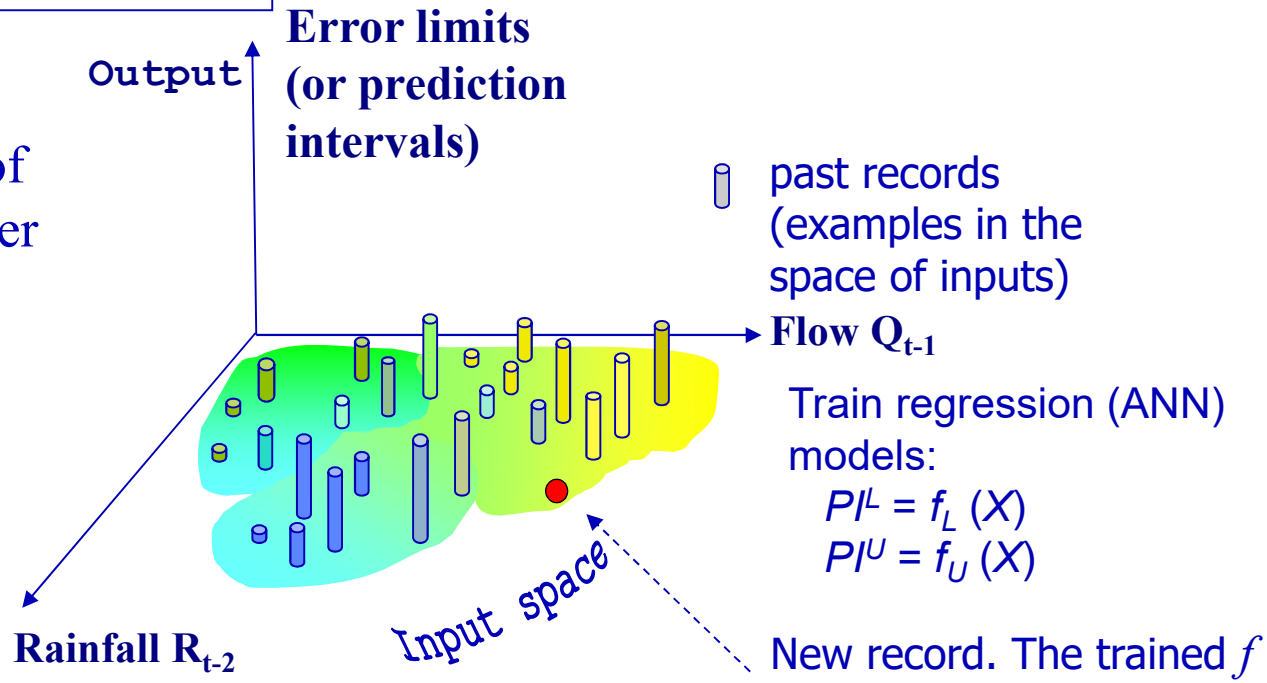


Idea 3: Use fuzzy clustering of examples to generate training data sets

$$PI^{L}_{example} = \sum_{clus=1}^{Nclus} \mu_{clus,\,example}\; PIC^{L}_{clus}$$

- $\mu_{clus,,example}$ is the membership grade of the *example* to cluster *clus*

Eager learning (ANN or M5 model tree)

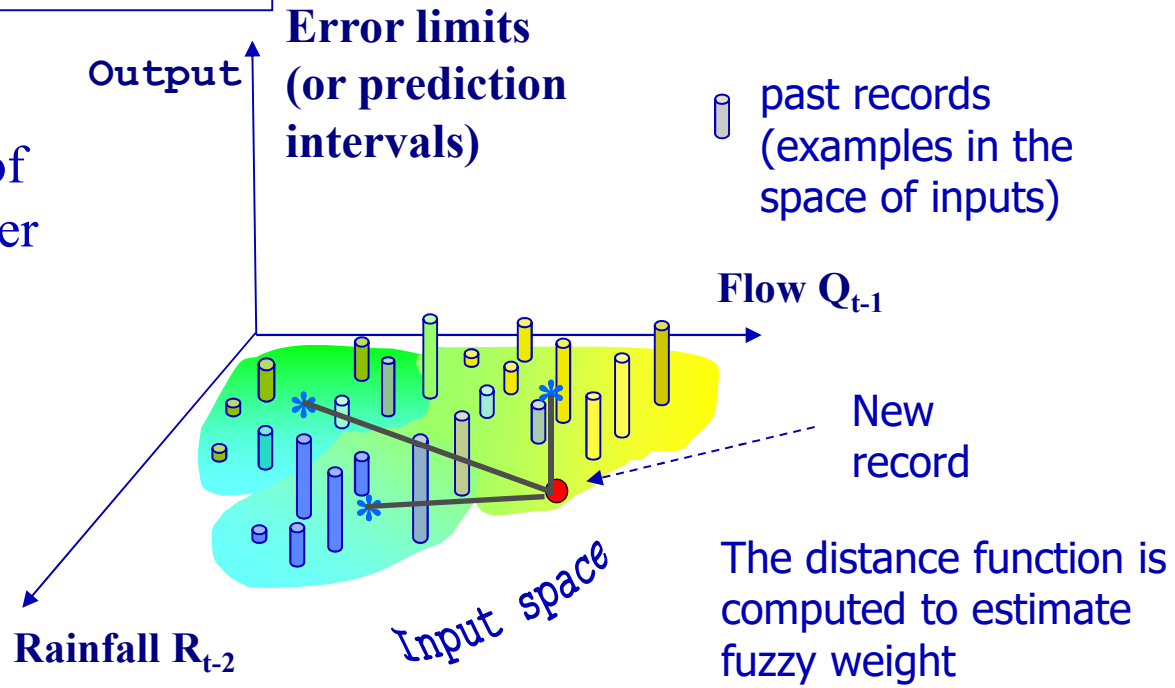


Using instance-based learning

$$PI^{L}_{example} = \sum_{clus=1}^{Nclus} \mu_{clus,\,example}\; PIC^{L}_{clus}$$

- $\mu_{clus,,example}$ is the membership grade of the *example* to cluster *clus*

Instance based learning



UNEEC details. Step 1: clustering

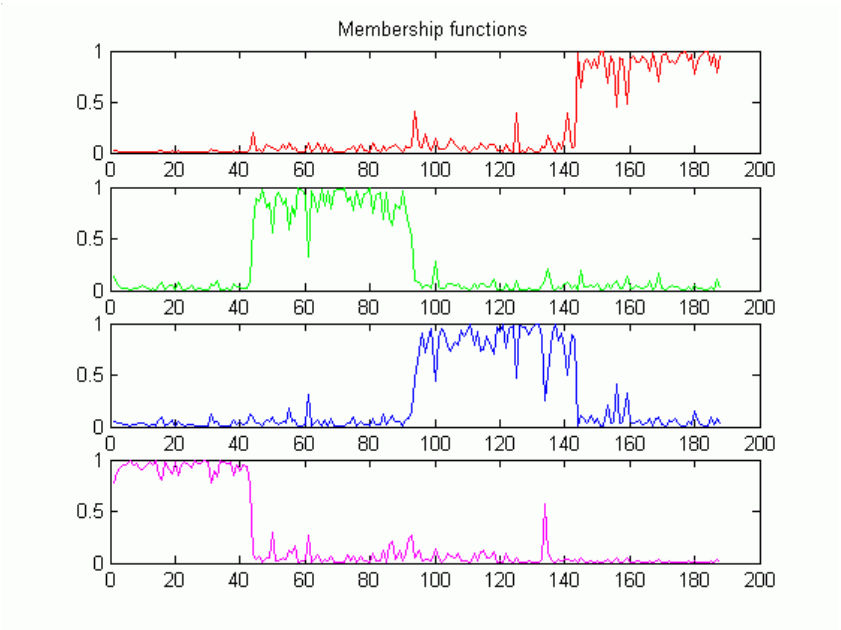
- Clustering (finding groups of data in the space characterising hydro-meteo condition): K-means clustering, fuzzy C-means clustering

Obj. function $$\min_{(U,V)}\left\{J_m(U,V)=\sum_{j=1}^{c}\sum_{i=1}^{N}\mu_{i,j}^{m}D_{i,j}^{2}\right\}$$

Distance $$D_{i,j}^{2}=\left\|x_i-v_j\right\|_A^2$$

Degree of Fuzzification $$m\geq 1$$

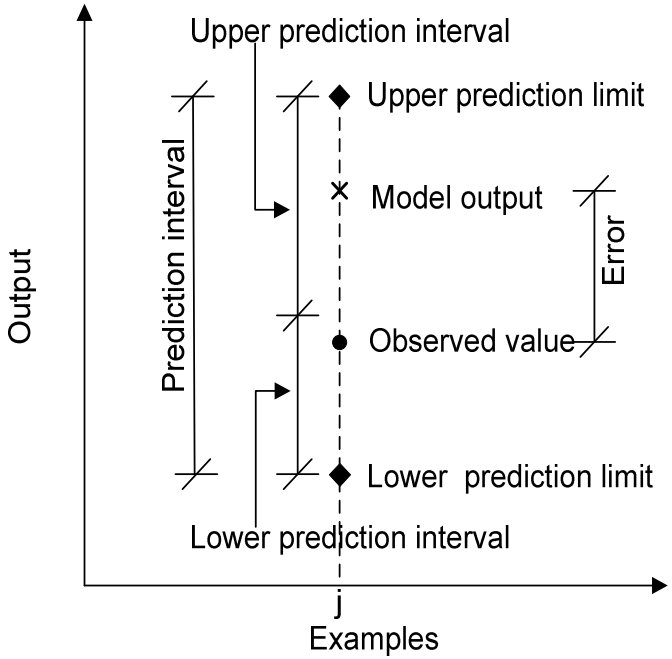
Constraint $$\sum_{j=1}^{c}\mu_{i,j}=1,\forall i$$

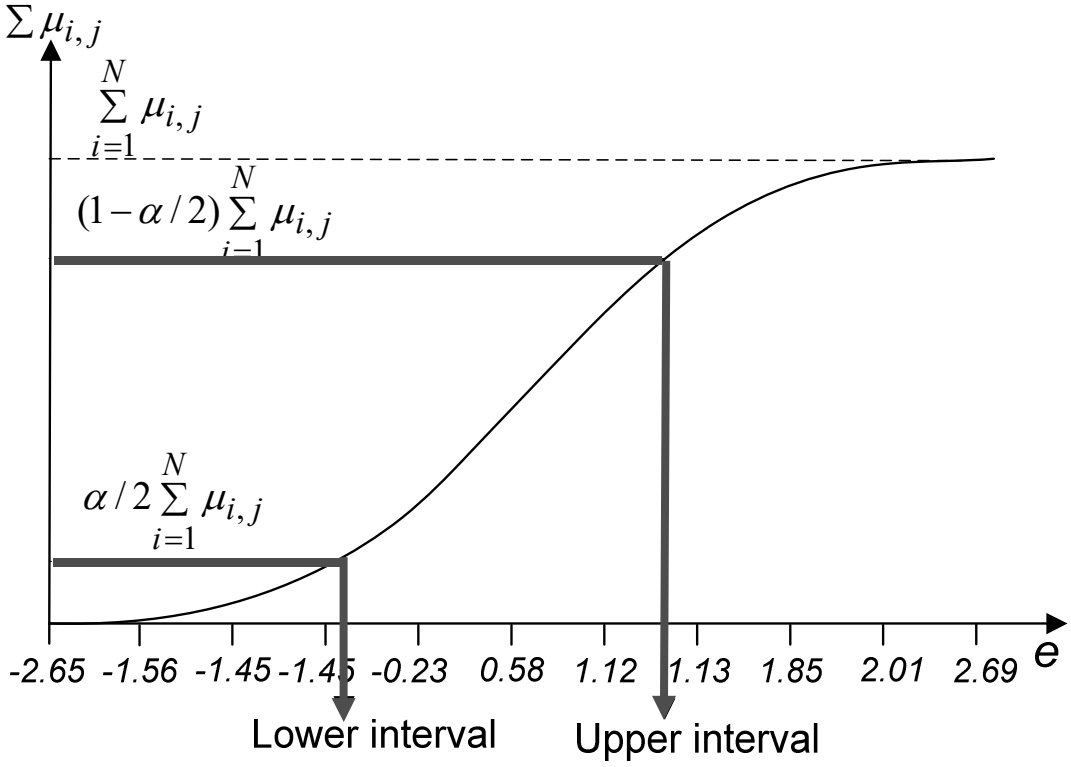


UNEEC details. Step 2: Determining Prediction Interval (PI) for each cluster

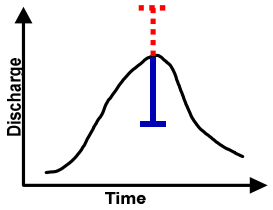
$$PIC_j^L = e_i \Rightarrow i : \sum_{k=1}^{i} \mu_{i,j} < \alpha/2 \sum_{i=1}^{N} \mu_{i,j}$$

$$PIC_j^U = e_i \Rightarrow i : \sum_{k=1}^{i} \mu_{i,j} < (1-\alpha/2) \sum_{i=1}^{N} \mu_{i,j}$$





UNEEC details.
Step 3, 4, 5: Building and using the model



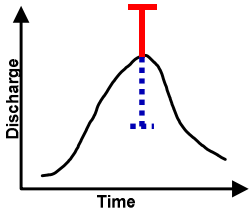
$$PI_i^L = \sum_{j=1}^{c} \mu_{i,j}\, PIC_j^L$$

$$PI^L = f_u^L(\mathrm{X_u})$$

$$PI^L = f_u^L(\mathrm{X_v})$$

$$PL_i^L = \underbrace{\hat{y}_i + PI_i^L}$$

Independent Computation



$$PI_i^U = \sum_{j=1}^{c} \mu_{i,j}\, PIC_j^U$$

$$PI^U = f_u^U(\mathrm{X})$$

$$PI^U = f_u^U(\mathrm{X_v})$$

$$PL_i^U = \underbrace{\hat{y}_i + PI_i^U}$$

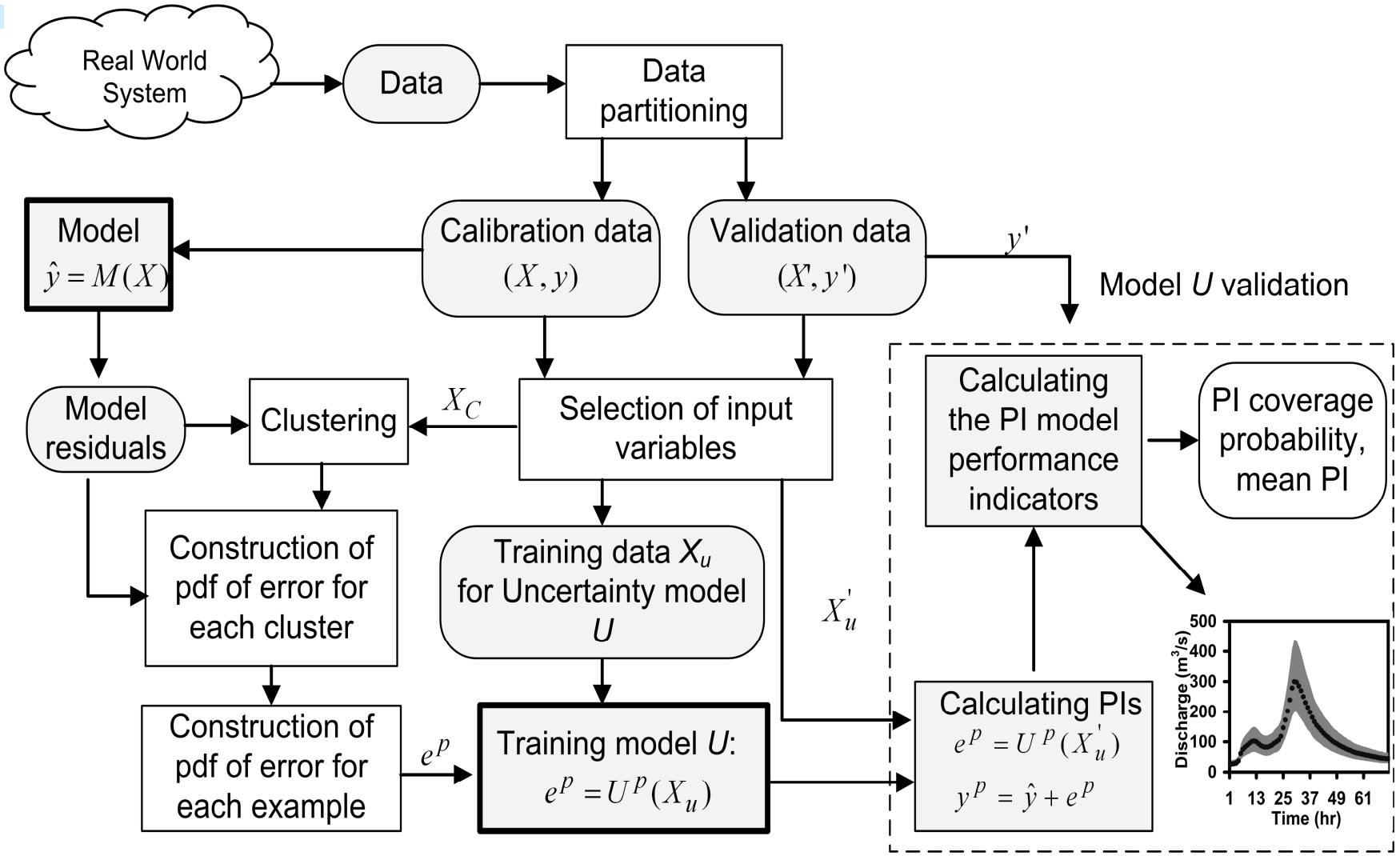
Step 3: Generation of Prediction intervals for each example

Step 4: Building the uncertainty Model

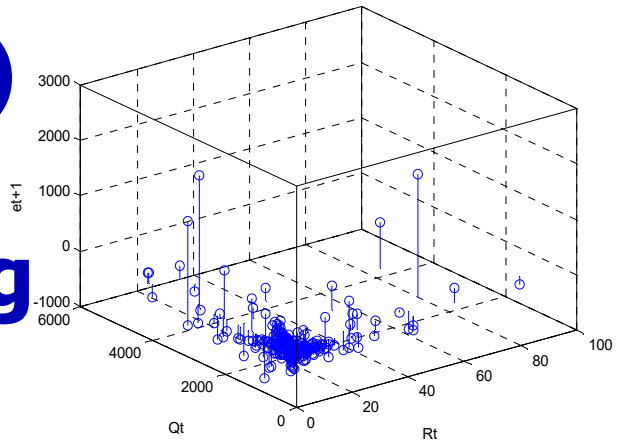
Step 5: Using the uncertainty Model

Model Outputs with uncertainty bounds

UNEEC methodology



Extensions (simplifications) of UNEEC: without clustering and using instance-based learning



- Based on Master study of Omar Wani (2015)
  - SKIBLUE (**S**treamflow-Centric **K** nearest neighbour Instance-**B**ased **L**earning and **U**ncertainty **E**stimation)
  - O. Wani, J. Beckers, A.H. Weerts, D.P. Solomatine. Non-parametric Predictive Uncertainty Estimation Using Instance Based Learning with Applications to Hydrologic Forecasting. HESS-D, 2016.
- Based on Master study of Ms. Jingyi Chen (2015)
  - UNEEC-IBL
  - Jingyi Chen (2015). Uncertainty Prediction in Hydrological Modelling: Case of Dapoling-Wangjiaba Catchment in Huai River Basin. UNESCO-IHE Master thesis

—— Escalating uncertainty ——

Assuming now uncertainty
in parameters and or data…

Running Monte Carlo simulations…

But how to estimate output uncertainty for the new model runs?

Models of Parametric Uncertainty **:**
… and again using Methods of Computational Intelligence

MLUE method

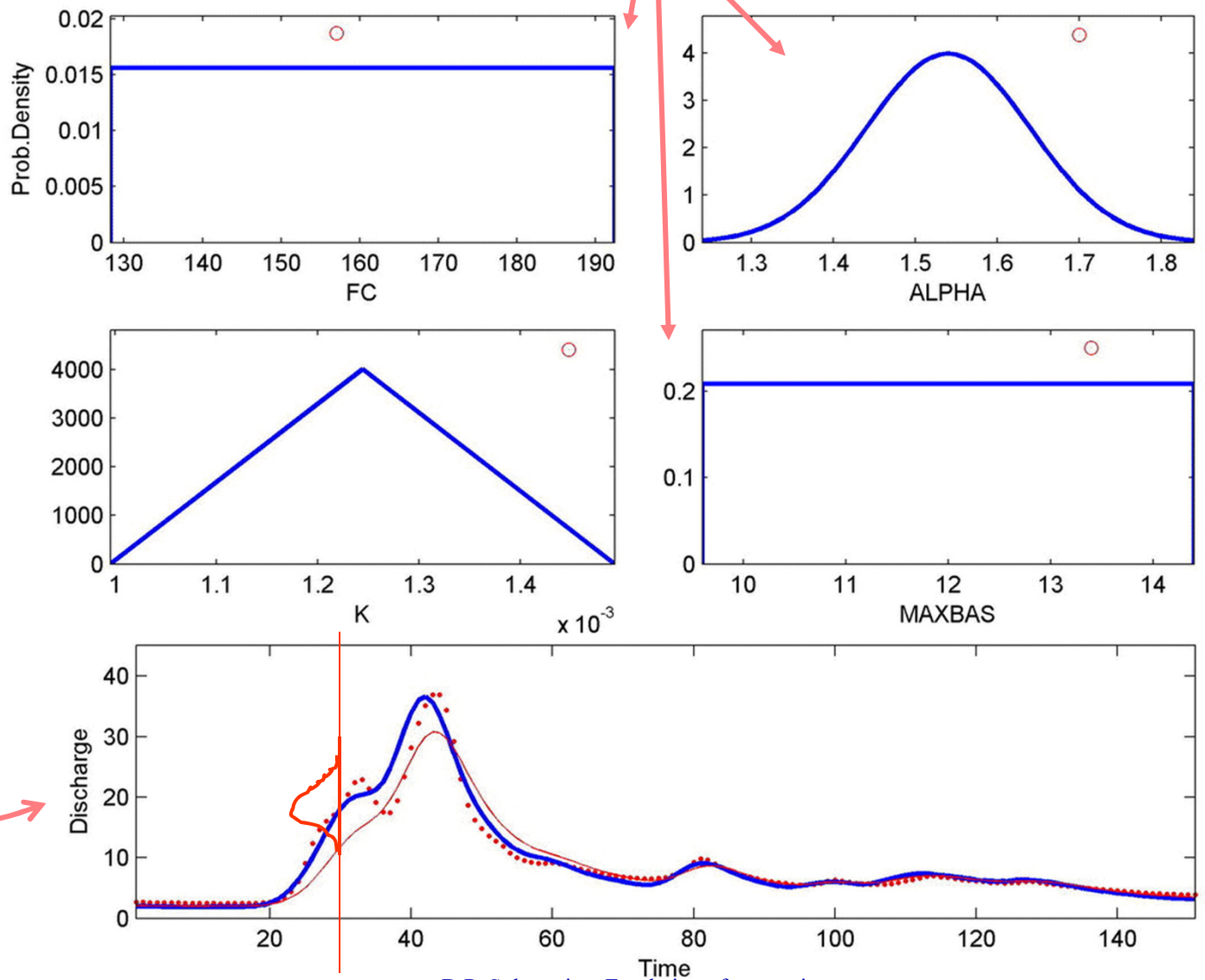
Machine Learning in Uncertainty Estimation

- machine learning model of the *process model's Monte Carlo simulation results* is built

D. L. Shrestha, N. Kayastha, and D. P. Solomatine (2009). A novel approach to parameter uncertainty analysis of hydrological models using neural networks. *HESS*, 13, 1235–1248.

Monte Carlo simulation of parametric uncertainty

$$y = M(x, s, \theta) + \varepsilon_s + \varepsilon_\theta + \varepsilon_x + \varepsilon_y$$



Issues with MC for new model runs in real-time

- Issues with re-running MC for new inputs:
  - 1) convergence of the Monte Carlo simulation is very slow (O(N^-0.5)) so larger number of runs needed to establish a reliable estimate of uncertainties
  - 2) number of simulation increases exponentially with the dimension of the parameter vector ((O(n^d)) to cover the entire parameter domain
- Idea:
  - encapsulate the results of MC simulation in a machine learning model

MLUE Methodology (1)

- Consider the sources of the uncertainty analysis to be conducted within the framework of Monte Carlo simulation
- Execute the MC simulations to generate the data

$$y_i(t) = M(\mathbf{X}(t), \mathbf{p}_i)$$

- Estimate the uncertainty measures of the MC realizations, e.g., mean, variance, prediction intervals, quantiles
  - to start with, estimate two quantiles (say, 5% and 95%), forming the prediction interval PI

MLUE Methodology (2)

- Analyze the dependency of the uncertainty measures (quantiles) on the input and state variables of the hydrological model
  - we used Correlation and Average mutual information analysis
- *Select the input variables* for machine learning model based on the dependency analysis
- *Train the machine learning model U* to predict the uncertainty measures of MC realizations $PI = U(\mathbf{X})$
- Validate machine learning model *U* by estimating the uncertainty measures with the "new" input data
- Use model *U*

Validation

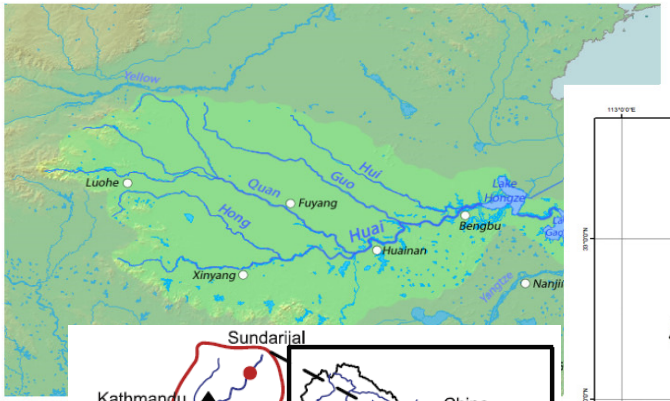
- Measuring predictive capability of uncertainty model *U* (measures the accuracy of uncertainty models in approximating the quantiles of the model outputs generated by MC simulations)
  - Coefficient of correlation (r) and root mean squared error (RMSE)
- Measuring the statistics of the uncertainty estimation (i.e. goodness of the model U as uncertainty estimator)
  - Prediction interval coverage probability (PICP) and mean prediction interval (MPI) (Shrestha & Solomatine 2006, 2008)

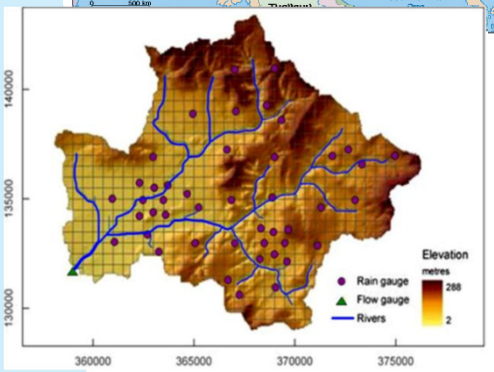
$$PICP = \frac{1}{n}\sum_{t=1}^{n} C$$

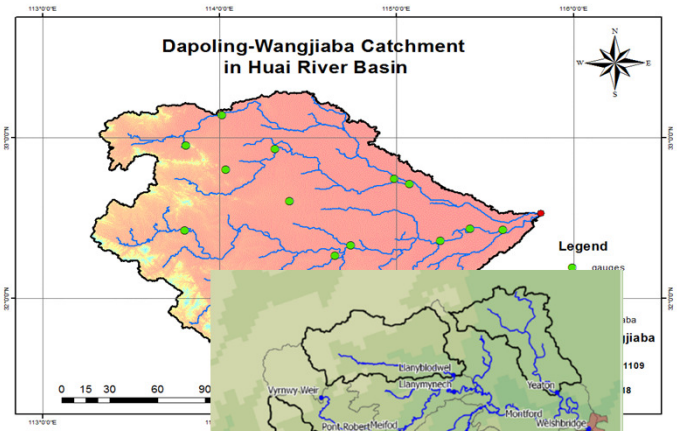
$$\text{with } C = \begin{cases} 1, & PL_t^L \le y_t \le PL_t^U \\ 0, & \text{otherwise} \end{cases}$$

$$MPI = \frac{1}{n}\sum_{t=1}^{n} (PL_t^U - PL_t^L)$$

- Visualizing such as scatter and time plot of the prediction intervals obtained from the MC simulation and their predicted values





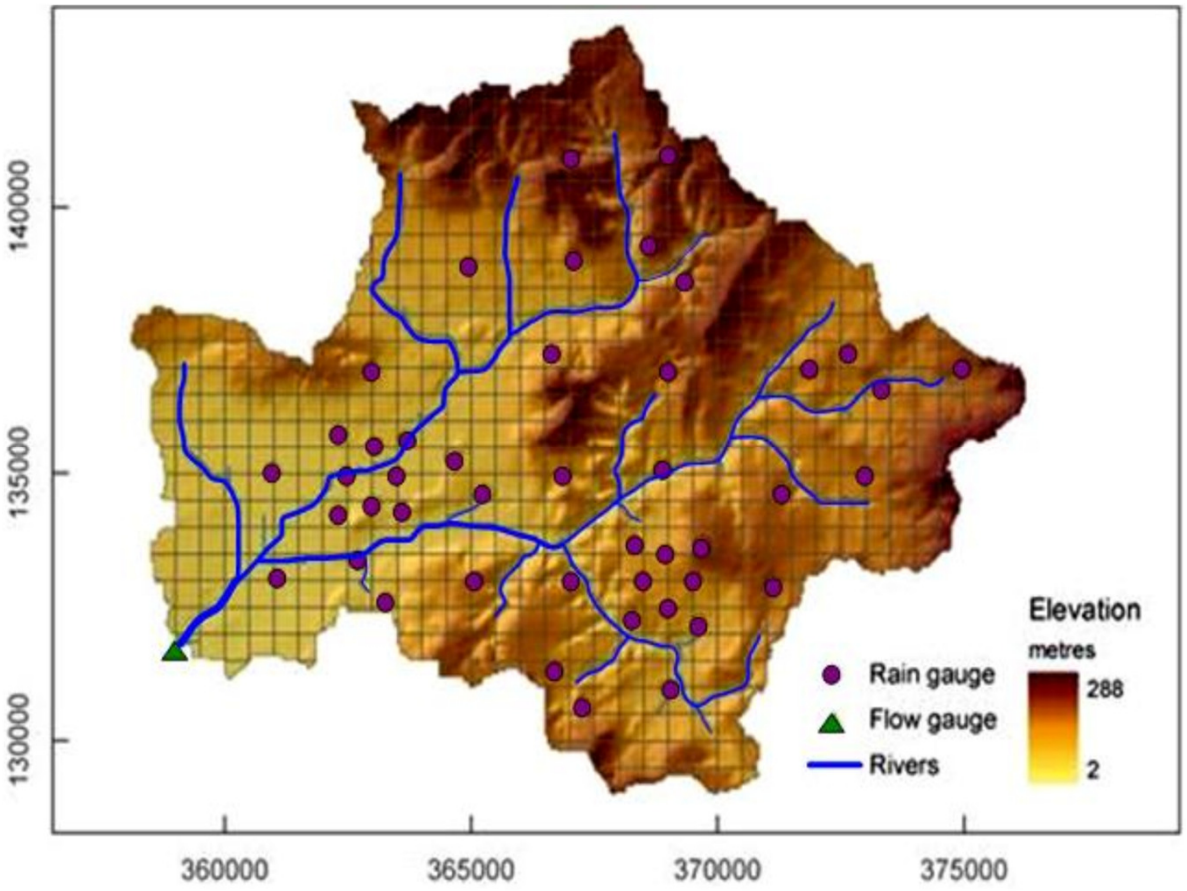


Applications

UNEEC and MLUE were tested and compared to other methods on 5 various cases:

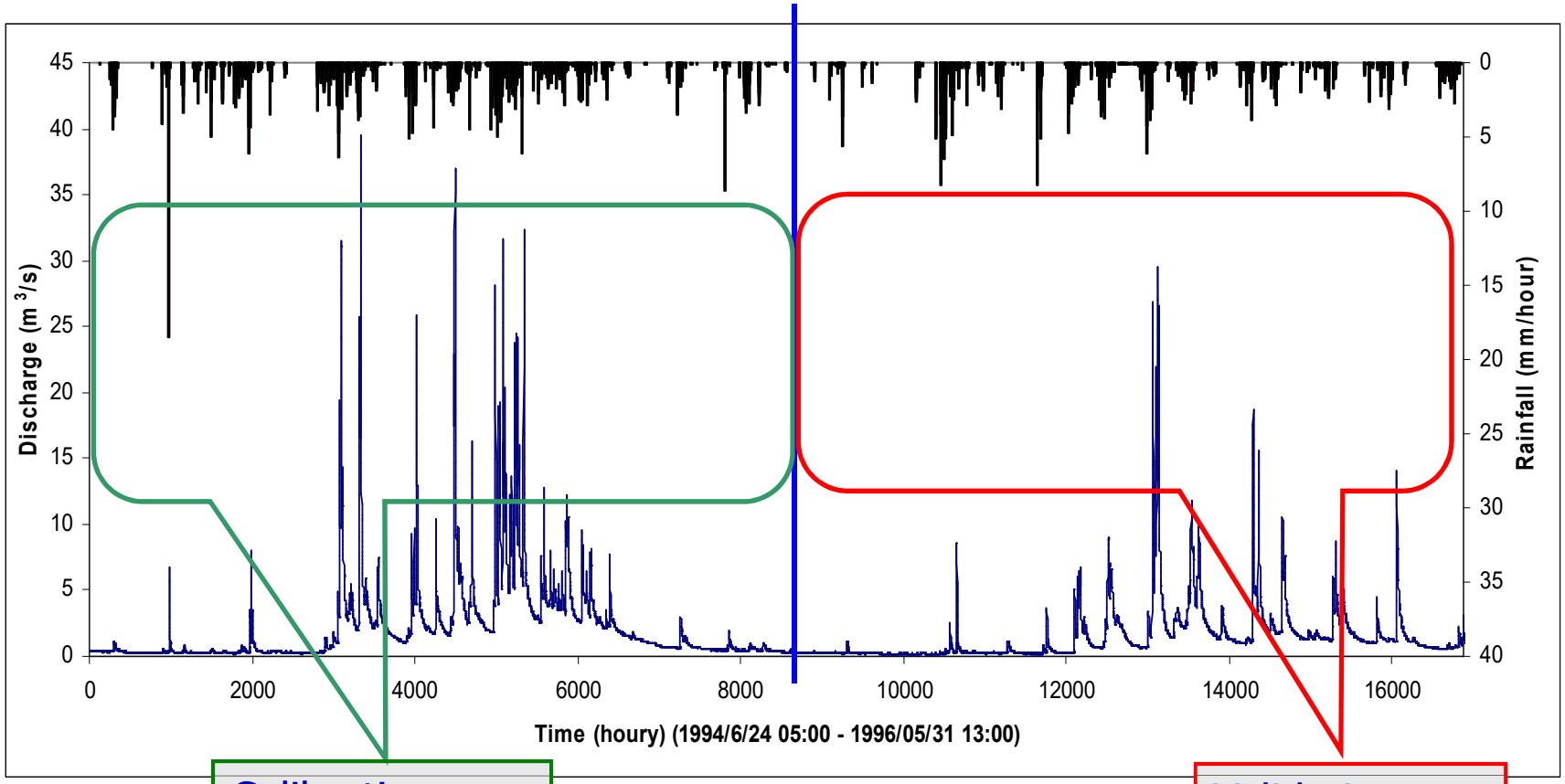
Brue, Bagmati, Sieve, Severn, Dapoling-Wanjiaba

Study area: Brue catchment, UK



- Catchment area: 135 km^2
- Location: south west of England
- Average annual rainfall: 867 mm
- Mean river flow: 1.92 m3/s
- Calibration data: 24/06/94-24/06/95
- Validation data: 24/06/95-31/05/96

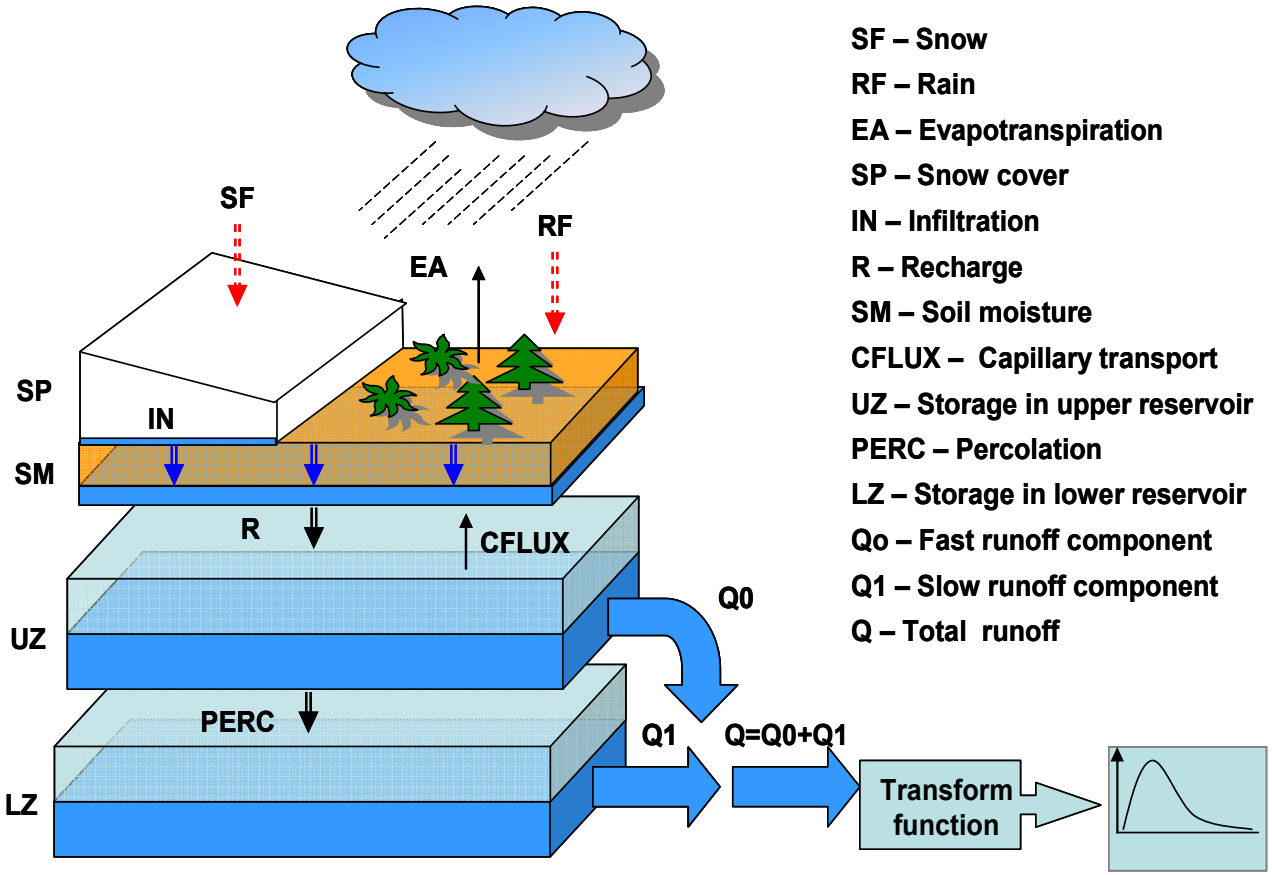
Study area: Brue catchment, UK



Calibration data (8760 points):

Validation data (8217 points):

Conceptual Hydrological model HBV



SF – Snow
RF – Rain
EA – Evapotranspiration
SP – Snow cover
IN – Infiltration
R – Recharge
SM – Soil moisture
CFLUX – Capillary transport
UZ – Storage in upper reservoir
PERC – Percolation
LZ – Storage in lower reservoir
Qo – Fast runoff component
Q1 – Slow runoff component
Q – Total runoff

Data Analysis

- Analysis of dependency btw various combinations of the input variables and the output
  - Correlation
  - Average mutual information (AMI) between *REt* and PIs,

    (optimal lag time is around 7-9 hours).
  - Additional analysis of the correlation and AMI between the PIs and observed discharge *Qt* are carried out. (i.e. with the lag of 0, 1, 2) have very high correlation with the PIs.

Experimental setup

- MC simulation
  - 9 Parameters of HBV model are sampled uniformly from the feasible ranges
  - Nash-Sutcliffe coefficient of efficiency (CE) is used as error measure
  - Convergence – stabilized after 10,000 (75,000 runs made)
  - Only 25,000 "good" models considered (rejection threshold is set to 0) to compute prediction quantiles

Experimental setup

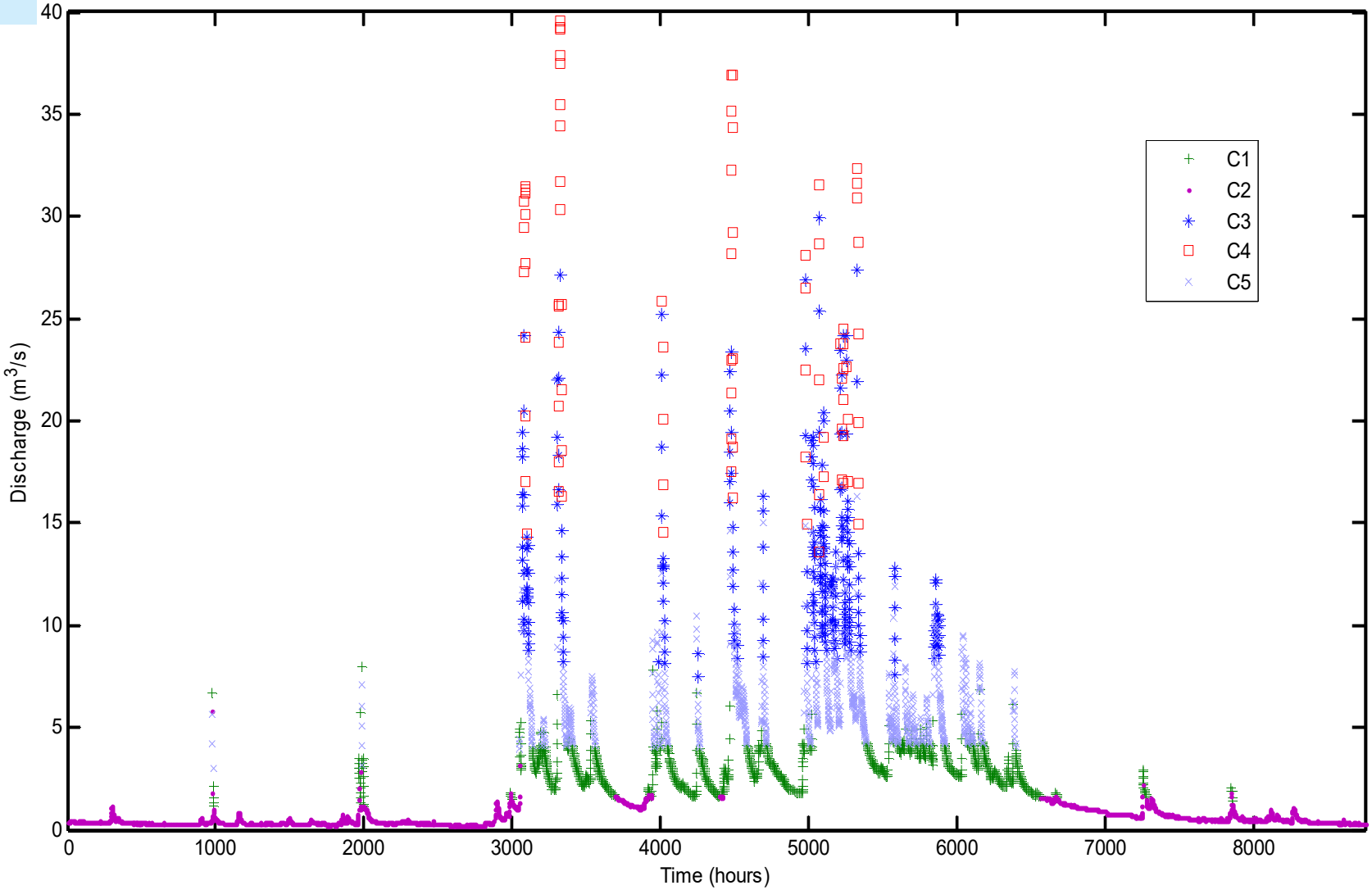
- Machine learning model *U*
  - $PI = U(RE_{t-5a}, Q_{t-1}, \Delta Q_{t-1})$
    - *PI* - lower or upper prediction intervals,
    - RE_{t-5a} - average of RE_{t-5}, RE_{t-6}, RE_{t-7}, RE_{t-8}, and RE_{t-9}
    - $\Delta Q_{t-1} - Q_{t-1} - Q_{t-2}$.
  - Input variables were selected based on the analysis of their relatedness to output error (average mutual information)

$$\mathrm{AMI} = \sum_{i,j} P_{XY}(x_i, y_j) \log_2 \left[\frac{P_{XY}(x_i, y_j)}{P_X(x_i) P_Y(y_j)} \right]$$

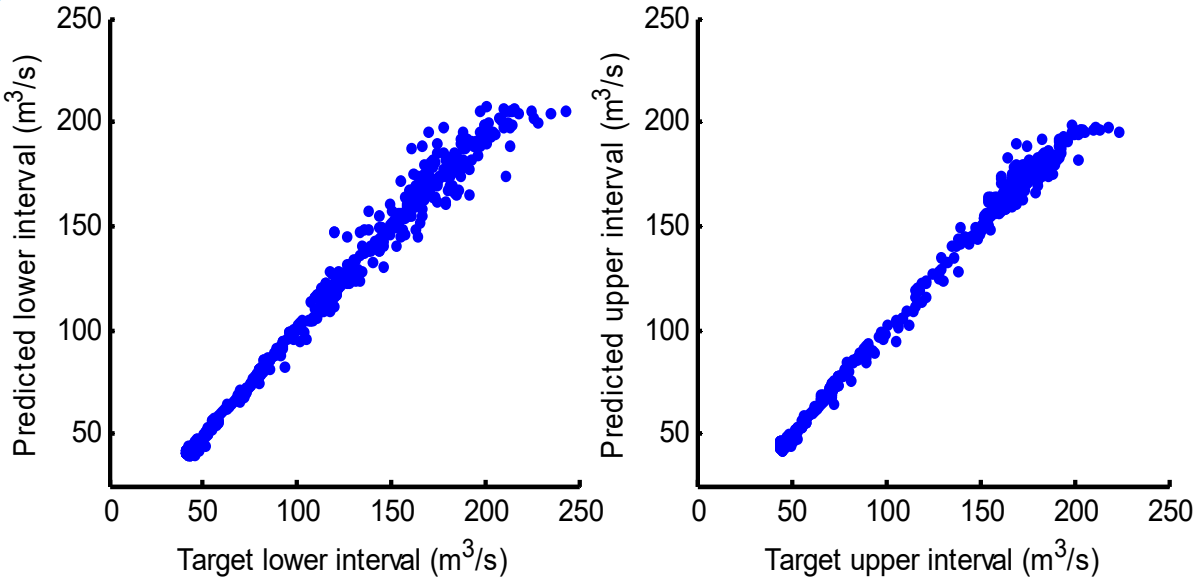
- Methods:
  - M5 model trees,
  - locally weighted regression
  - MLP neural networks

Results

UNEEC: Clustering result example

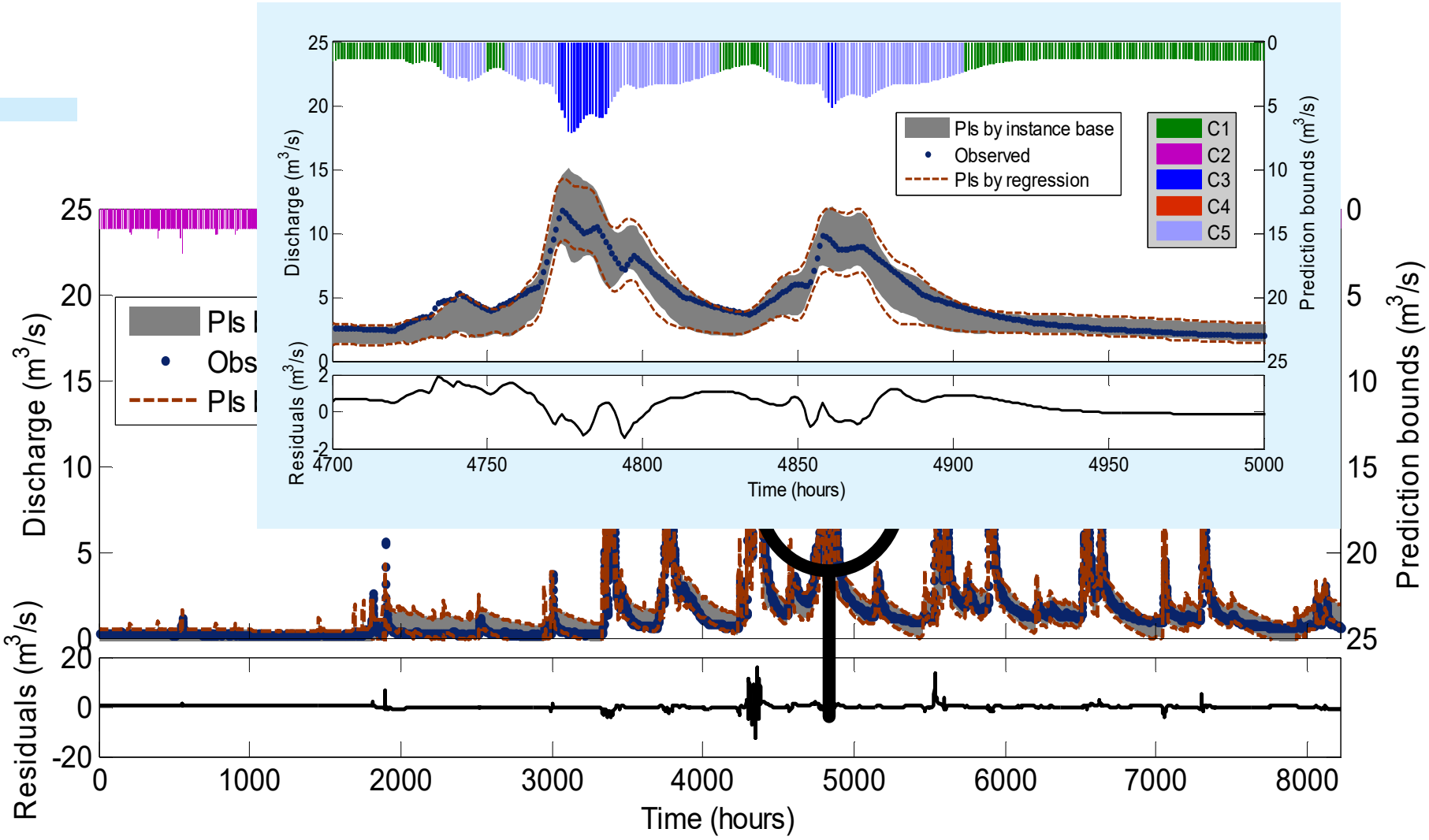


UNEEC: Performance (MLP ANN)

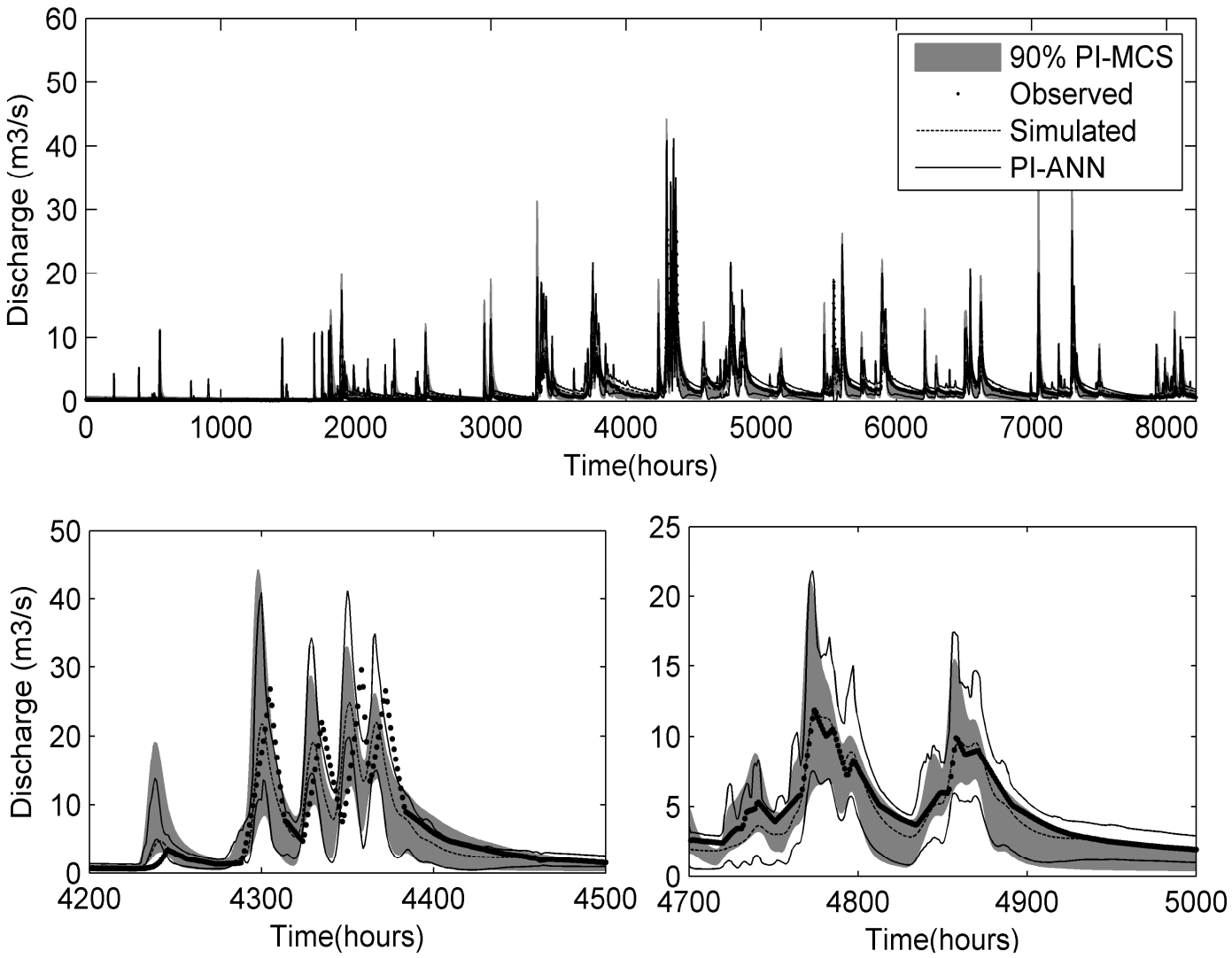


Prediction interval	Data set	Mean	Std. dev.	RMSE	Corr. coef.
lower	training	110.91	53.6	5.9582	0.9937
	CV	112.18	52.64	6.0852	0.9934
	training+CV	111.35	53.32	5.9582	0.9937
upper	training	115.16	55.11	3.9002	0.9975
	CV	116.69	54.18	3.9332	0.9974
	training+CV	115.66	54.79	3.9002	0.9975

UNEEC: Estimation of prediction intervals



MLUE: Estimation of prediction intervals



MLUE: Performances

- *Predictive capability*

	Corr C		RMSE	
	PI^L	PI^U	PI^L	PI^U
MT	**0.841**	**0.792**	**0.614**	**1.641**
LWR	**0.822**	**0.798**	**0.643**	**1.604**
ANN	**0.847**	**0.806**	**0.584**	**1.568**

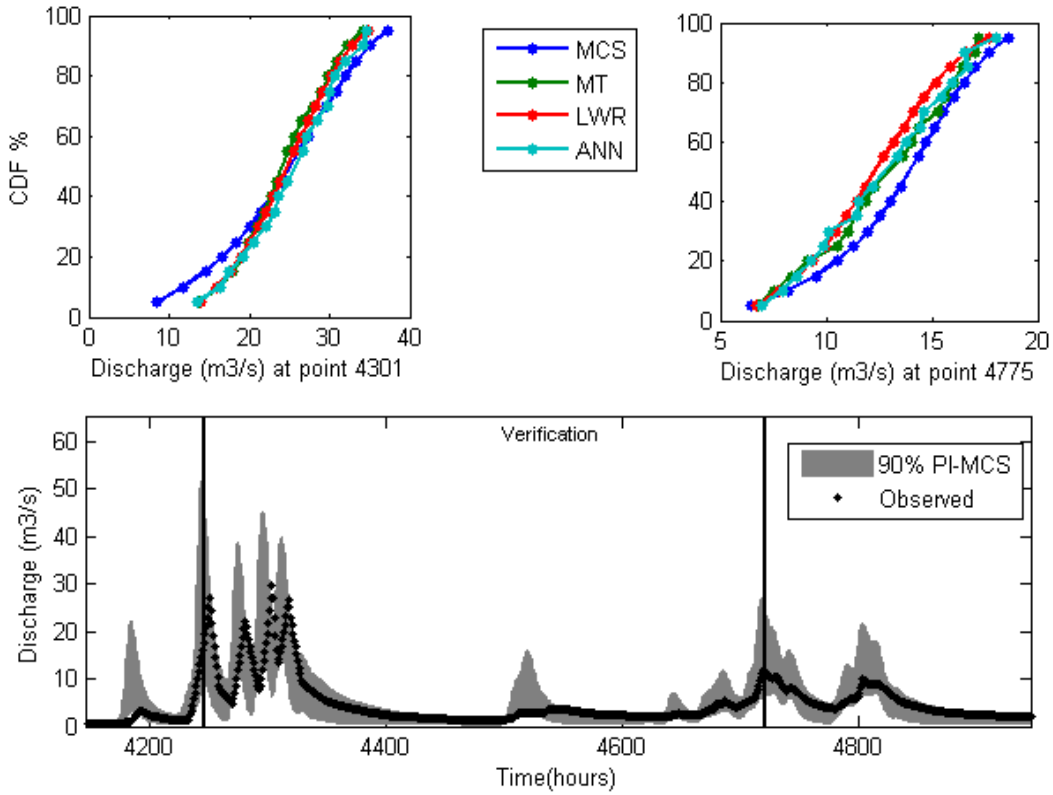
- *Goodness of uncertainty measures*

	MCS	MT	LWR	ANN
PICP %	77.24	66.97	**75.16**	65.54
MPI m^3/s	2.09	**2.03**	1.93	1.96

MCS = Monte Carlo
MT = M5 Model tree
LWR = local weighted regression
ANN =MLP neural network

Extensions

- Estimation of several quantiles 5%, 10%:10%:90%, 95%
  - i.e. estimating cdf of MC realizations by machine learning models



Use of Machine learning methods: conclusions

- Machine learning methods are able to replicate:
  - Past performance of a process model
  - Results of Monte-Carlo simulations
- The methods are computationally efficient and can be used in real time application
- They are to various kinds of models
- The results demonstrate that the interpretable uncertainty estimates are generated
- *Future work:*
  - Other ML methods are to be tested
  - The methods can be applied in the context of other sources of uncertainty - input, structure, or combined

References

- **UNEEC and extensions:**
  - D.L. Shrestha, D.P. Solomatine. Machine learning approaches for estimation of prediction interval for the model output. *Neural Networks* , 2006, 19(2), 225-235.
  - D.P. Solomatine, D.L. Shrestha. A novel method to estimate model uncertainty using machine learning techniques. *Water Resour Res*. 45, W00B11, 2009.
  - N. Dogulu, P. López López, D. P. Solomatine, A. H. Weerts, and D. L. Shrestha. Estimation of predictive hydrologic uncertainty using the quantile regression and UNEEC methods and their comparison on contrasting catchments, *Hydrol. Earth Syst. Sci.*, 19, 3181-3201, 2015.
  - O. Wani, J. Beckers, A.H. Weerts, D.P. Solomatine. Non-parametric Predictive Uncertainty Estimation Using Instance Based Learning with Applications to Hydrologic Forecasting. HESS-D, 2016
- **MLUE:**
  - D. L. Shrestha, N. Kayastha and D. P. Solomatine. A novel approach to parameter uncertainty analysis of hydrological models using neural networks. *Hydrol. Earth Syst. Sci.*, 13, 1235-1248, 2009.
  - Shrestha, D.L., Kayastha, N., Solomatine, D., Price, R. Encapsulation of parametric uncertainty statistics by various predictive machine learning models: MLUE method. *J Hydroinformatics*, 16 (1), 95-113, 2014.

Conclusions

- Uncertainty analysis should always contain explicit answers to two questions:
  - 1) what type of uncertainty is to be analysed: residual (which do not need MC), or parametric/data (which need MC)
  - 2) what is required: just analysis of the past, or also a model predicting the future uncertainty
- It is advisable:
  - to go explicitly through all stages of **uncertainty escalation**, starting from residual uncertainty
  - to try to build the **predictive models of uncertainty** at all stages
  - complement the deterministic models M with a *family of uncertainty models* U

What to know more?

- We teach Master courses:
  - Hydroinformatics
  - Flood Risk Management

UNESCO-IHE
Institute for Water Education

Erasmus Mundus
FLOODRisk Master
Global Change ▪ Hydroinformatics ▪ Planning

Thank you for your attention

UNESCO IHE DELFT

Erasmus Mundus FLOODRisk Master
Global Change ▪ Hydroinformatics ▪ Planning