Geo372
Vertiefung GIScience

Processing errors

Herbstsemester

Ross Purves

Last week

- We looked at error sources (**blunders**, **systematic errors** and **random errors**)
- We explored some models to **quantify errors** focussing on **interval/ ratio** and **nominal data** for both **positional and attribute values**
- I introduced **OpenStreetMap** as an example of Volunteered Geographic Information
- We looked at all **five elements** of the SDTS (positional accuracy, attribute accuracy, completeness, logical consistency and lineage) using **OSM** and the **literature**

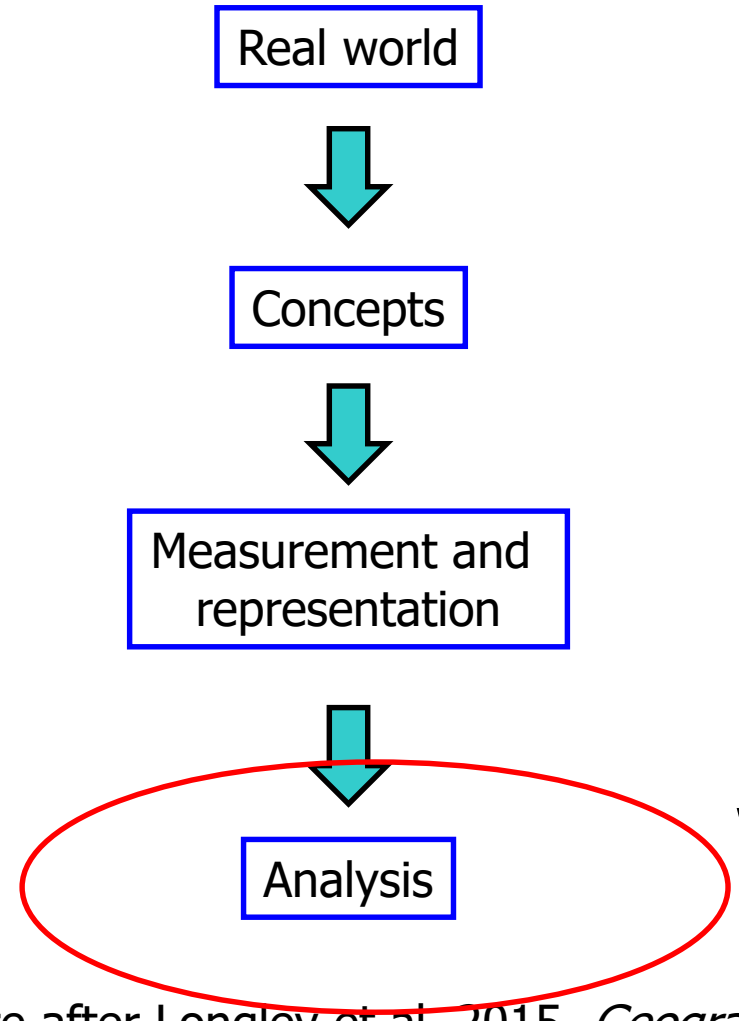
Learning objectives

- You understand how we can **estimate errors** for uncorrelated, differentiable functions for linear models using **simple error analysis**
- You know what **Monte Carlo simulation** is and can give examples of its use
- You know why **spatial autocorrelation** is important in producing **error realisations** for Monte Carlo simulation
- You understand the **basic concepts** behind **fuzzy set theory** and can show how they can be used to **overlay two uncertain values**

Outline

- Today's lecture will explore **propagation of error**
- We will first look at **simple error theory**, which is **aspatial** and can be applied to **local functions** through **map algebra**
- Then we will explore **Monte Carlo simulation** and its use to model **uncertainty** in a range of **DEM outputs** and a **complex process model**
- Finally, we will briefly look at an example of the use of **fuzzy set theory** to understand how we can model **vague** and **uncertain concepts** in GIS

Today's focus: Recall conceptual view of uncertainty sources



At each of these stages **uncertainty** can be introduced which **should** be reported with respect to the **data quality** and **may** have an influence on the data's **fitness for use**

What happens when we combine data?

Figure after Longley et al. 2015. *Geographic Information Systems and Science*.

Simple Error Propagation theory

For random, independent (no correlation) errors

where U is a function of variables $a_1, a_2, \ldots a_j$:

$$U = f(a_1, a_2, \ldots a_j)$$

then the standard deviation in U (SU) is **approximated** by

$$SU = \left[\sum_{i=1}^{j} \left(\frac{\partial U}{\partial a_i} \right)^2 Sa_i{}^2 \right]^{\frac{1}{2}}$$

where Sa_i is the standard deviation in a_i

Simple Error Propagation theory (2)

- We can easily apply this to **map algebra**, IF we know the **error** in the **individual parameters**
- We are assuming that the variables are **uncorrelated** (see Burrough and McDonnell for correlated variables) and that the system is linear
- The method described here uses **Taylor series** to **approximate** the error(you should have heard about this in maths)

If $U = a_1 + a_2 + a_3$

or $U = a_1 - a_2 - a_3$

then

$$SU = \left[\sum_{i=1}^{3}\left(\frac{\partial U}{\partial a_i}\right)^2 Sa_i^{\,2}\right]^{\frac{1}{2}}$$

$$= \left[\begin{array}{l}\left(\frac{\partial(a_1+a_2+a_3)}{\partial a_1}\right)^2 Sa_1^{\,2} + \\ \left(\frac{\partial(a_1+a_2+a_3)}{\partial a_2}\right)^2 Sa_2^{\,2} + \\ \left(\frac{\partial(a_1+a_2+a_3)}{\partial a_3}\right)^2 Sa_3^{\,2}\end{array}\right]^{\frac{1}{2}}$$

$$= \left[1.Sa_1^{\,2} + 1.Sa_2^{\,2} + 1.Sa_3^{\,2}\right]^{\frac{1}{2}}$$

$$= \left[Sa_1^{\,2} + Sa_2^{\,2} + Sa_3^{\,2}\right]^{\frac{1}{2}}$$

If $U = a_1 a_2 a_3$

then

$$SU = \left[\sum_{i=1}^{3}\left(\frac{\partial U}{\partial a_i}\right)^2 Sa_i^{\,2}\right]^{\frac{1}{2}}$$

$$= \left[\begin{array}{l}\left(\frac{\partial(a_1a_2a_3)}{\partial a_1}\right)^2 Sa_1^{\,2} + \\ \left(\frac{\partial(a_1a_2a_3)}{\partial a_2}\right)^2 Sa_2^{\,2} + \\ \left(\frac{\partial(a_1a_2a_3)}{\partial a_3}\right)^2 Sa_3^{\,2}\end{array}\right]^{\frac{1}{2}}$$

$$= \left[\begin{array}{l}(a_2a_3)^2 Sa_1^{\,2} + (a_1a_3)^2 Sa_2^{\,2} \\ + (a_1a_2)^2 Sa_3^{\,2}\end{array}\right]^{\frac{1}{2}}$$

Error propagation example

A farmer wants to calculate net income per field using the following formula

$N = Y \text{ (yield)} \times P \text{ (price)} - C \text{ (costs)}$

$Y = 6 \pm 2 \text{ tha}^{-1}$

$P = 100 \pm 10 \text{ CHFt}^{-1}$

$C = 40 \pm 20 \text{ CHFha}^{-1}$

$N = (6 \times 100) - 40 =$
$560 \text{ CHFha}^{-1} \pm ??$

We calculate the uncertainty using the formulas we have seen. The uncertainty in the gross price is simply

$$S_G = \sqrt{(P^2 S_Y^2 + Y^2 S_P^2)}$$
$$= \sqrt{(100^2 2^2 + 6^2 10^2)}$$
$$= 116.62 \text{ CHFha}^{-1}$$

$$S_N = \sqrt{(S_G^2 + S_C^2)}$$
$$= \sqrt{(116.62^2 + 20^2)}$$
$$= 118.32 \text{ CHFha}^{-1}$$

So

$N = 560 \pm 118.32 \text{ CHFha}^{-1}$

Implications

- If the variables are **correlated**, the errors are **larger**
- **Adding** leads to smaller combined errors than **multiplication/ division,** subtraction can lead to very **large relative errors**
- We use this theory to estimate errors for any **differentiable equation**
- This **analytical method** makes it easy to assess where we need to reduce errors…

But, for typical, non-linear, complex, models simple error theory is not valid

Estimating errors in more complex models

- In more **complex models**, where relationships are non-linear or not local, then **Taylor series** are **not appropriate**
- A common method is to estimate error propagation by **brute force** using **Monte Carlo simulation (MCS)**
- **MCS** is typically used for **interval or ratio data**, and is computationally intensive (we have to do large numbers of calculations)

Monte Carlo simulation

- Basic idea, for every location, do the following N times:
  - a) Generate a set of **realisations** of variables whose uncertainty we want to model (a_1, a_2... a_j)
  - b) Calculate the **resulting value** of U=(a_1, a_2... a_j) for **every realisation**
  - Calculate **useful statistics** for our N realisations (typically mean and standard deviation)
- How **big** do you think N needs to be?

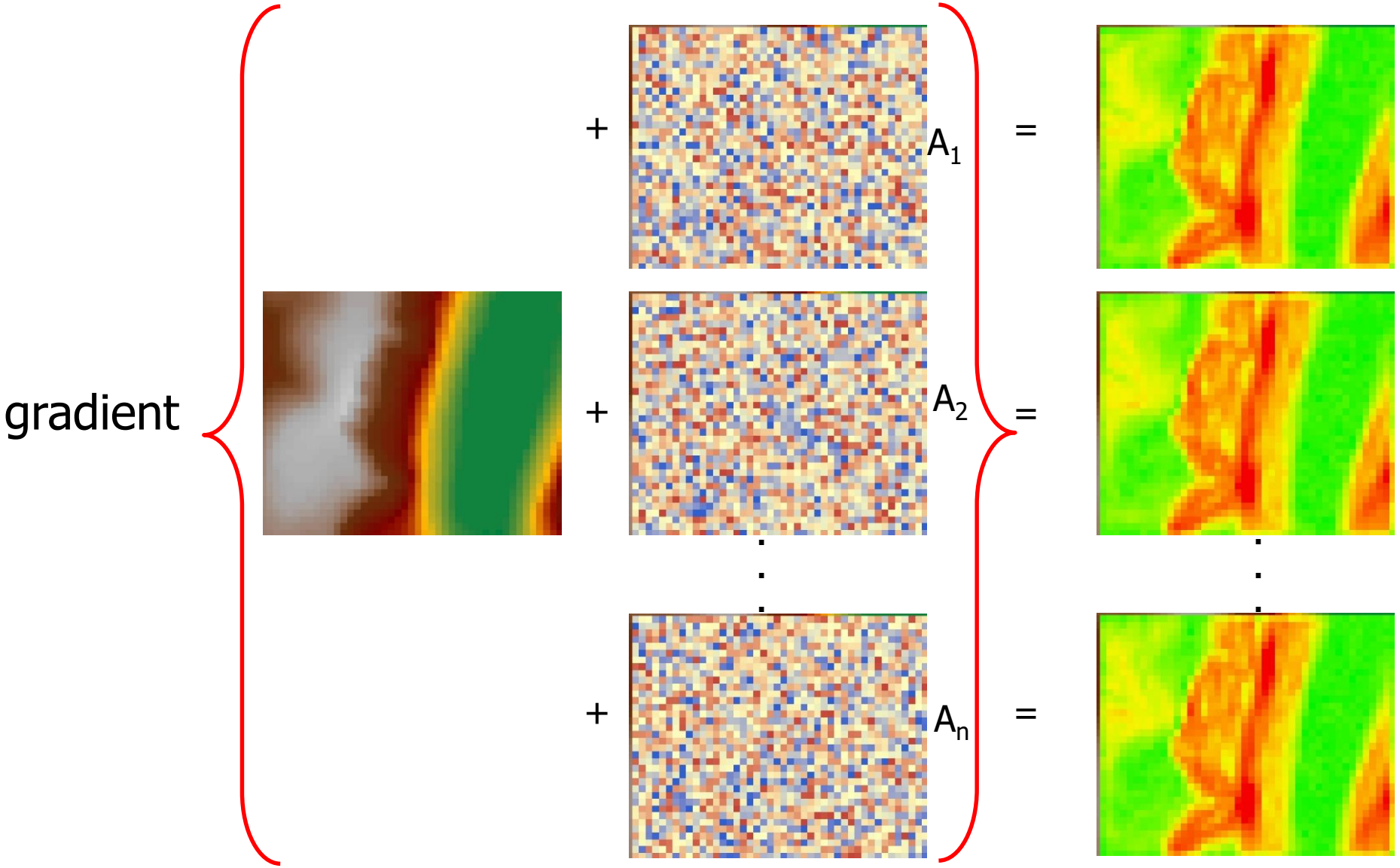
Monte Carlo example

- We are going to explore Monte Carlo simulation by looking some examples **using the DEM** from the hydrology lecture
- The DEM has a **resolution** of **50m** – metadata state that this DEM has an **RMSE of 3m**
- We will start by assuming that the error is **normally distributed** and that there is **no spatial autocorrelation** in error
- This is a common (but probably wrong!) assumption

Three DEM examples

- Calculating **gradient** using **finite differences** with ArcGIS
- Calculating **stream power**, which is a function of **gradient** (focal) and **specific accumulation area** (global)
- Calculating **viewshed** (global) from a single point

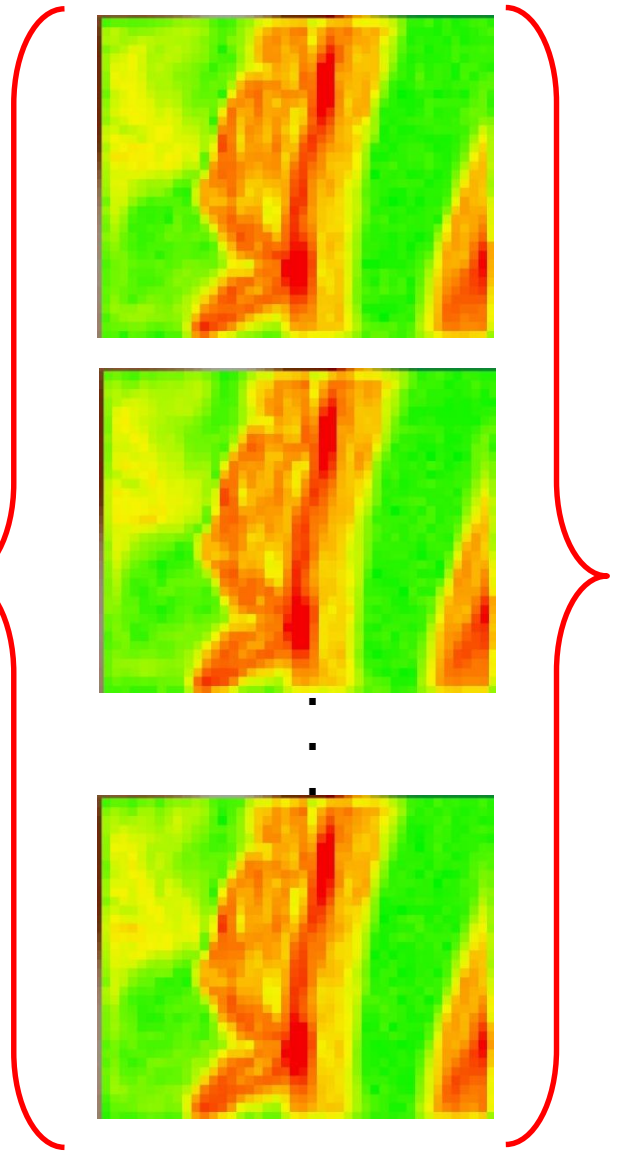
Example procedure - gradient

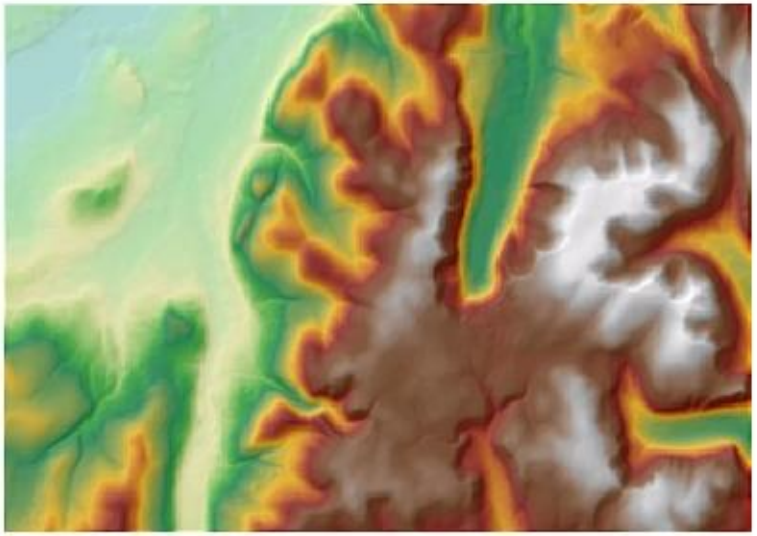


Summarising MC results

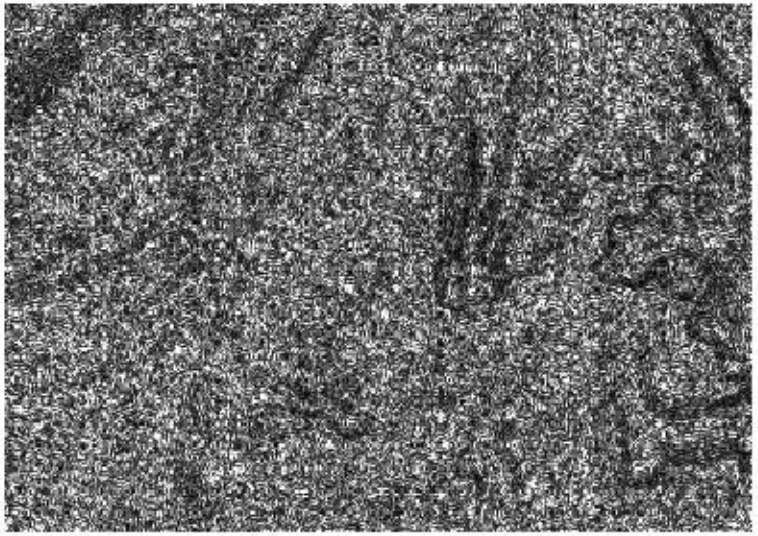
- When we have completed our N realisations, we can calculate summary statistics
- Most commonly we look at **mean** and **standard deviation**
- Dividing the standard deviation by the mean gives the **relative deviation** -> where are we least sure about our answer?

Mean
Standard deviation
Min
Max
etc…

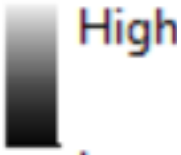




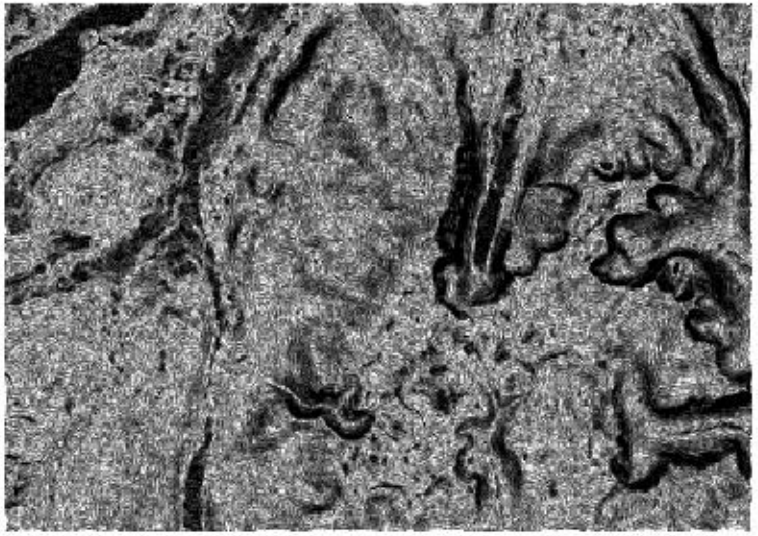
Original data



Gradient std.dev. (N=5)

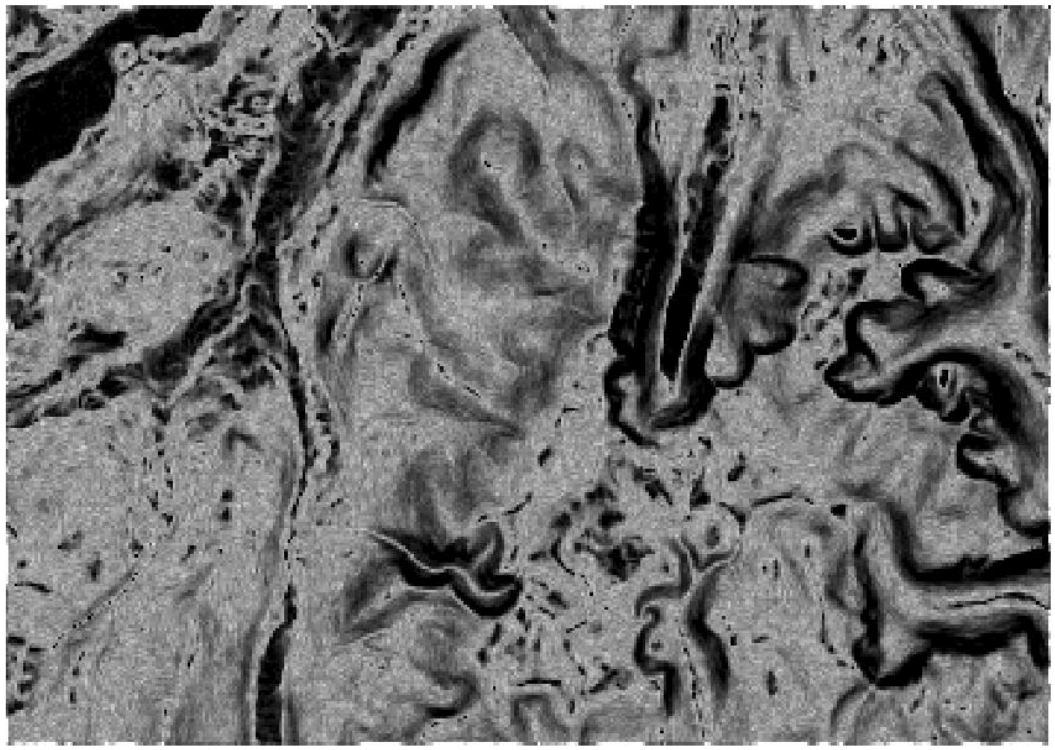


Gradient std.dev. (N=25)

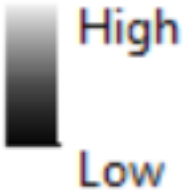


Gradient std.dev. (N=100)

Increasing N

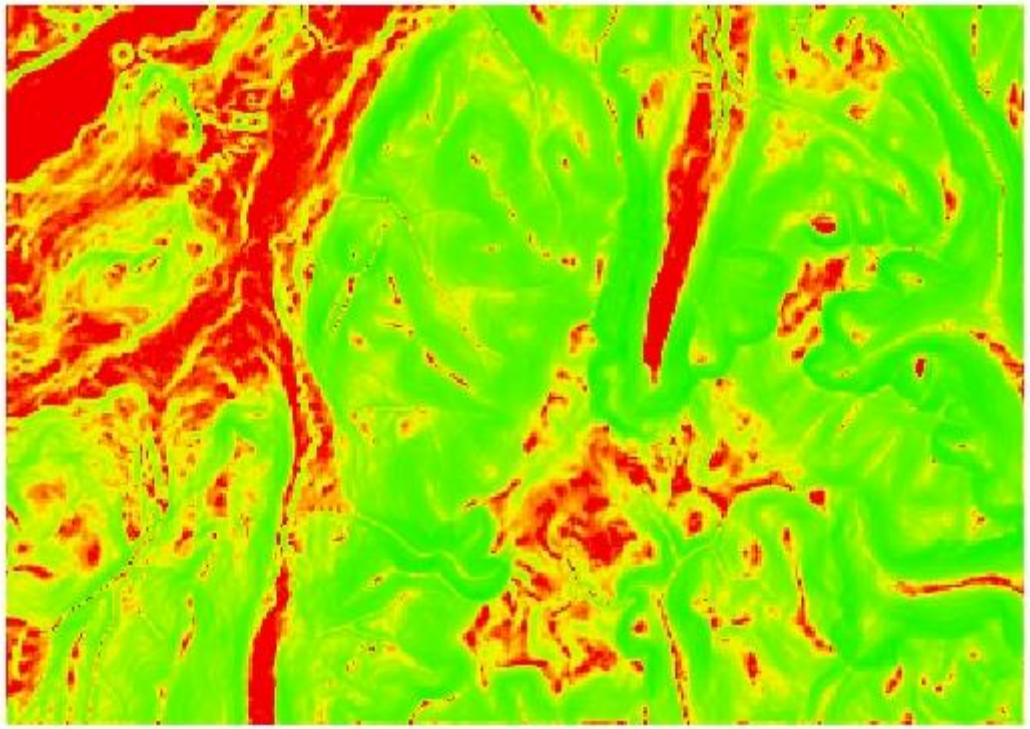


Gradient std. dev. (N=1001)



- If we increase N then the standard deviation becomes **more spatially autocorrelated**
- Does the pattern **make sense** – where would you expect uncertainty to be least/ greatest?

Relative standard deviation



Relative dev. (Std. dev. / Mean) (N=1001)

- Absolute standard deviation doesn't tell us the whole story
- Here we see areas of **high relative standard deviation** (red) and low relative standard deviation (green)
- These are the **opposite** of absolute standard deviation – **why?**

Comments – uncertainty in gradient

- The **relative uncertainty** is greatest at **flat locations**
- We looked at standard deviation and not mean – mean values at this resolution and for a large grid look very similar
- Even a very small RMSE (3.0m) at a 50m resolution made a real difference to slope values
- After **100 iterations** the standard deviation surface was still **very rough** – typical that we need to do **1000s of iterations**

Uncertainty in stream power

- **Recall: Stream power** can be defined as

$$\Omega = \rho g q \tan \beta$$

where ρ is the density of water;

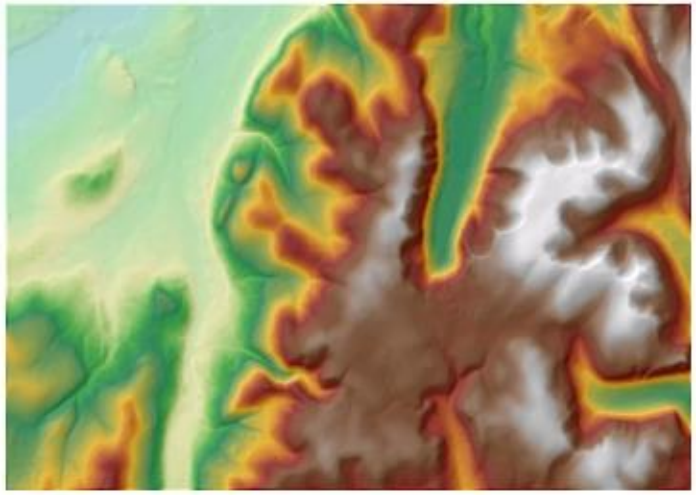
g is the acceleration due to gravity; and

q the discharge per unit width.

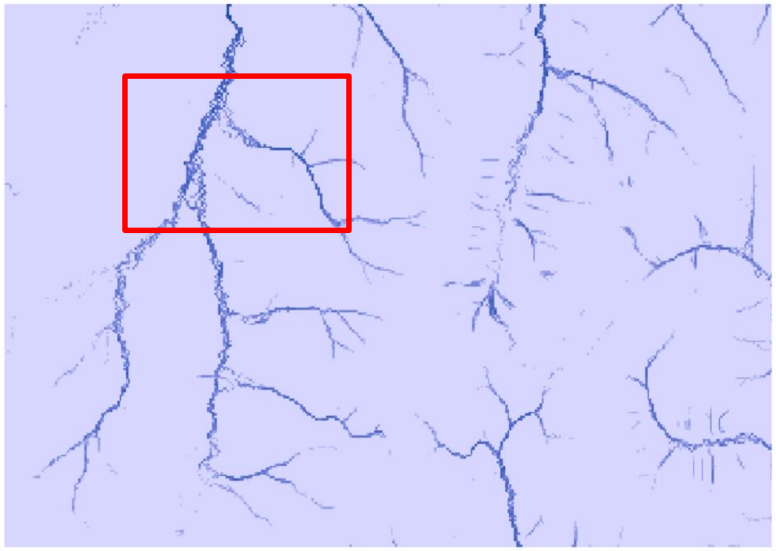
- A_s (*specific catchment area*) is often considered to be **proportional to discharge**, so a **compound index** for stream power is

$$A_s \tan \beta$$

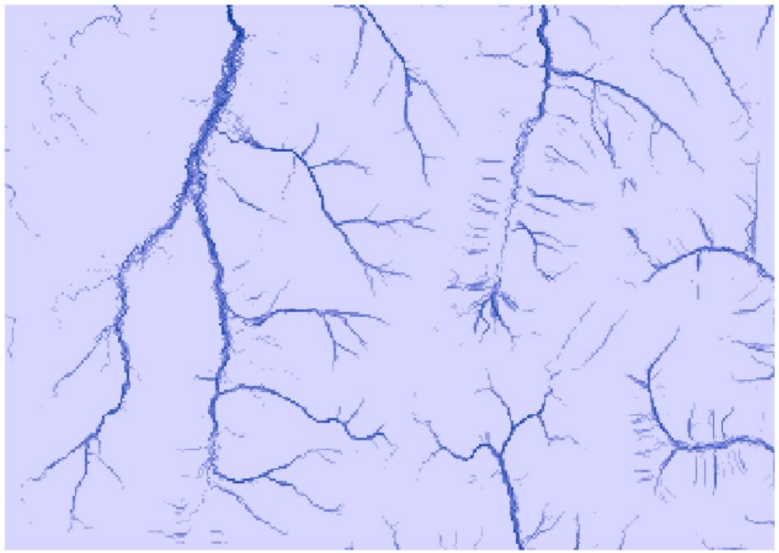
- To calculate stream power we need gradient (β) (as before) and A_S – **we must** (for **every realisation**)
  - **fill** the DEM
  - calculate **flow direction**
  - calculate **flow accumulation**



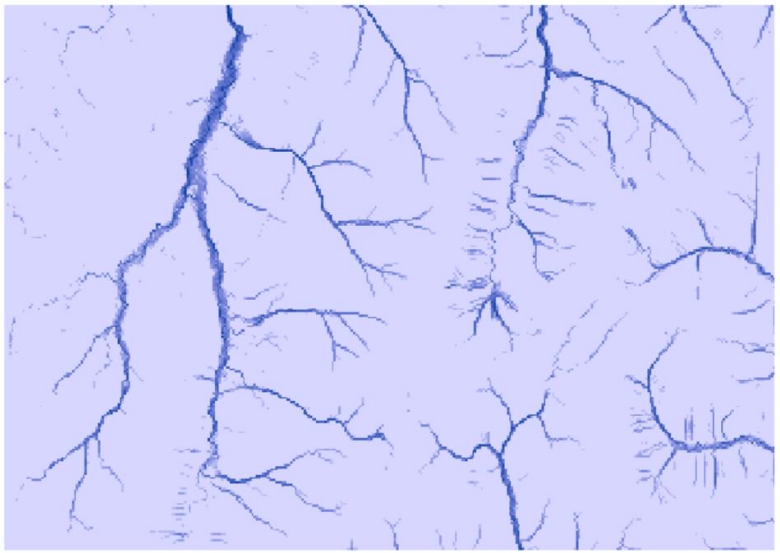
Original data



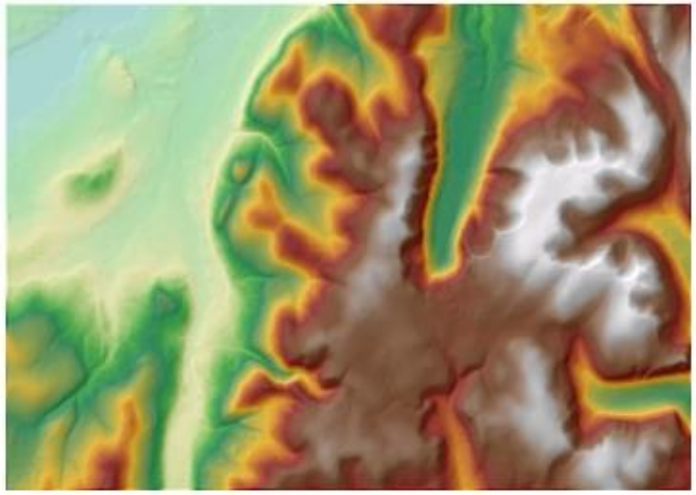
Stream power mean (N=5)



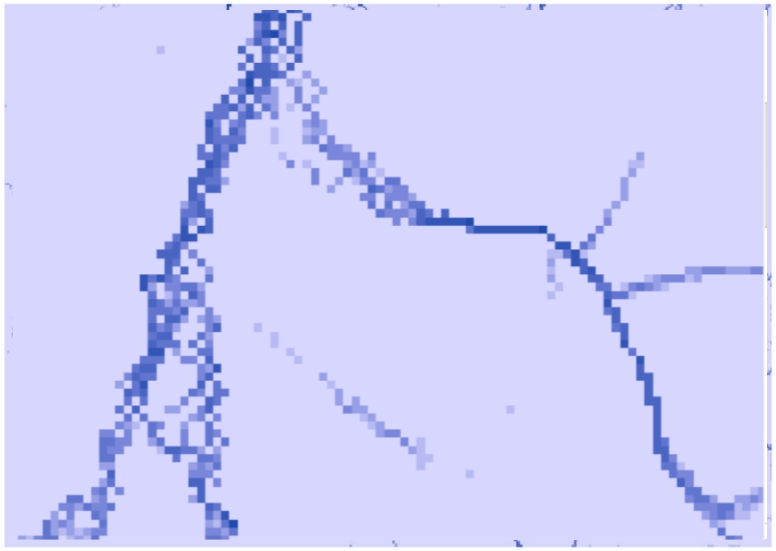
Stream power mean (N=25)



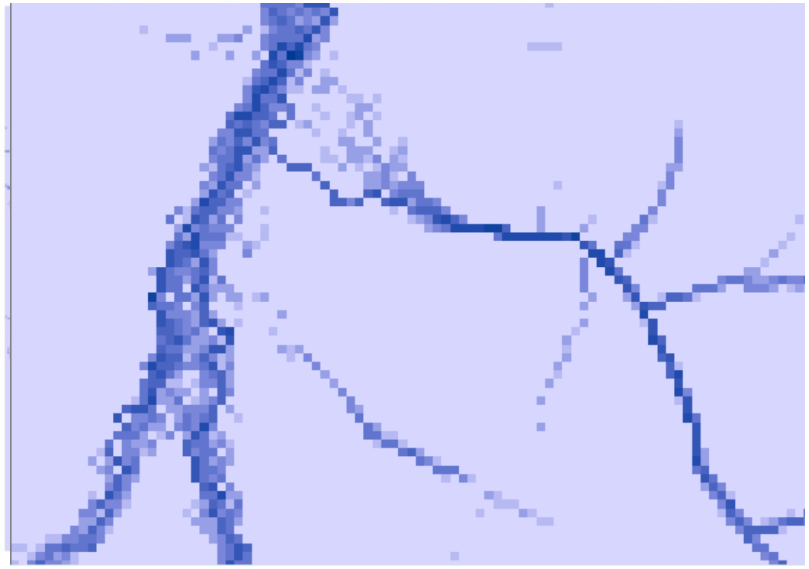
Stream power mean (N=100)



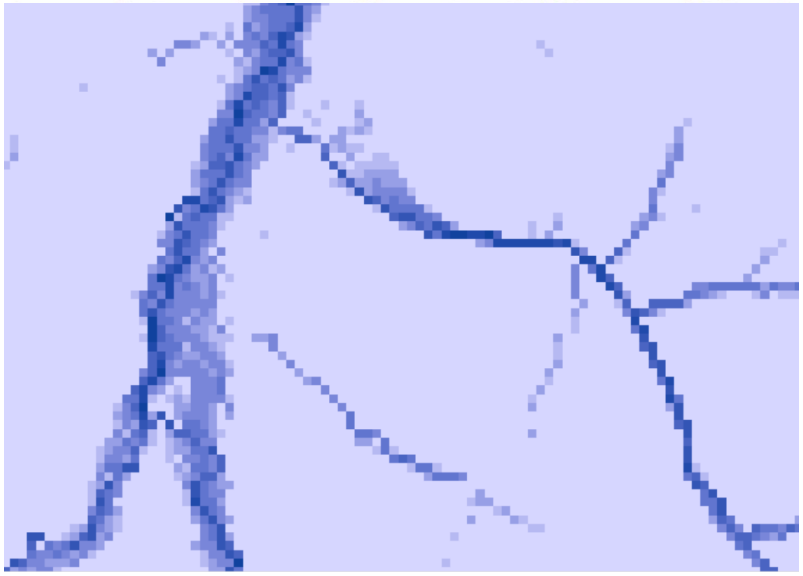
Original data



Stream power mean (N=5)



Stream power mean (N=25)



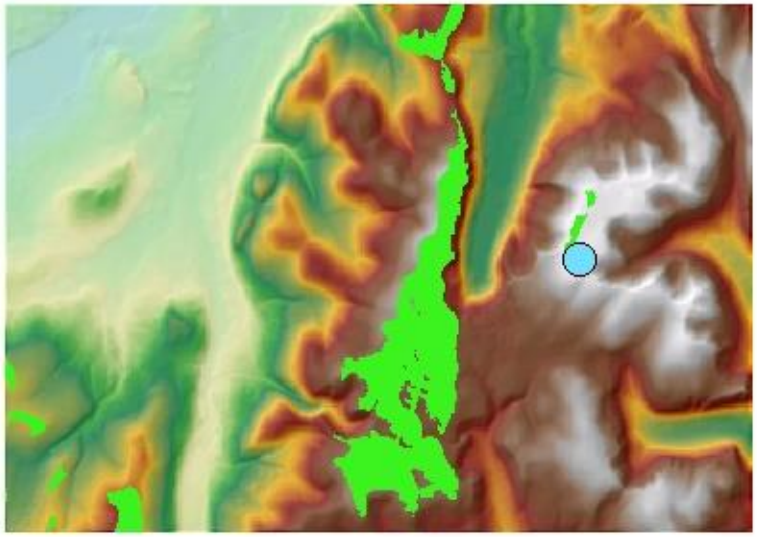
Stream power mean (N=100)

Comments on uncertainty in stream power

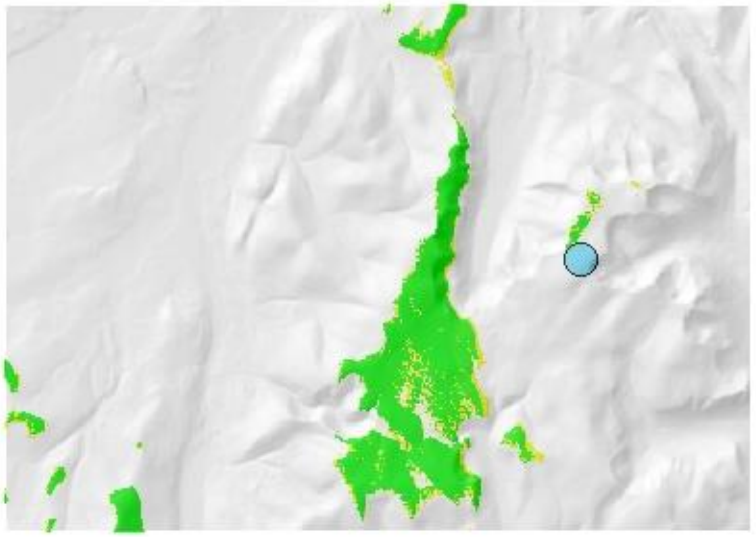
- Here we look at **mean stream power**
- Channels **concentrate** and **uncertainty in position reduces** with more iterations
- Again, **greatest uncertainty** in **flat areas** – but note how information increases with more iterations (e.g. buffer of likely locations)

Viewsheds

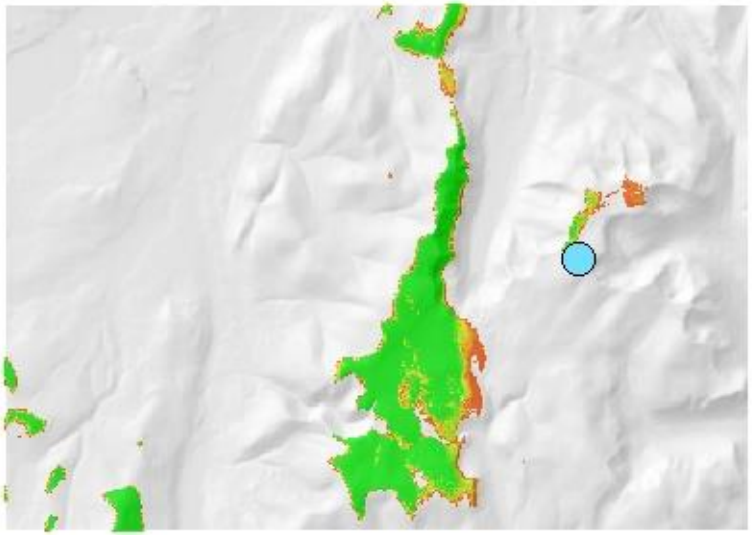
- Two weeks ago I reported a result from Fisher who said that **uncertainty decreased** the **size of viewsheds**
- I ran Monte Carlo simulations for the same area, again with an RMSE of 3.0m from a single view point
- The results illustrate how often a location is visible for different values of N (we could consider this equivalent to a **probability viewshed (red is rarely visible, green often)**



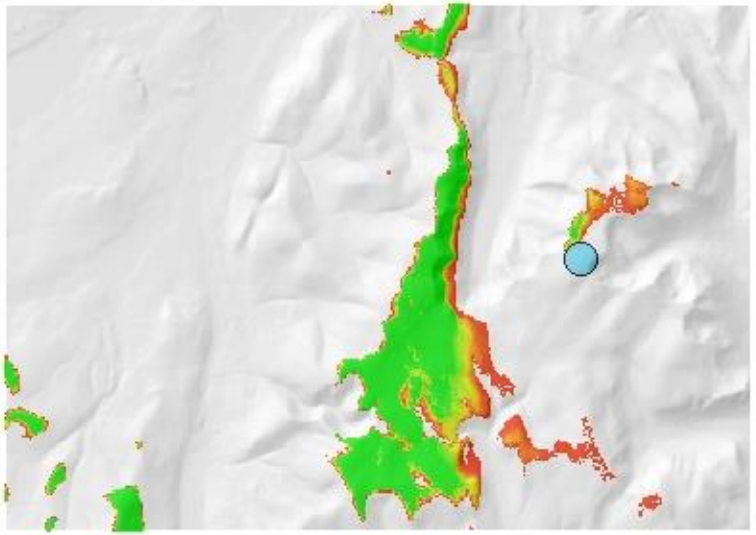
Original data
5606 cells visible



Viewshed mean (N=5)
Mean 4287 cells visible



Viewshed mean (N=25)
Mean 3957 cells visible



Viewshed mean (N=100)
Mean 3805 cells visible

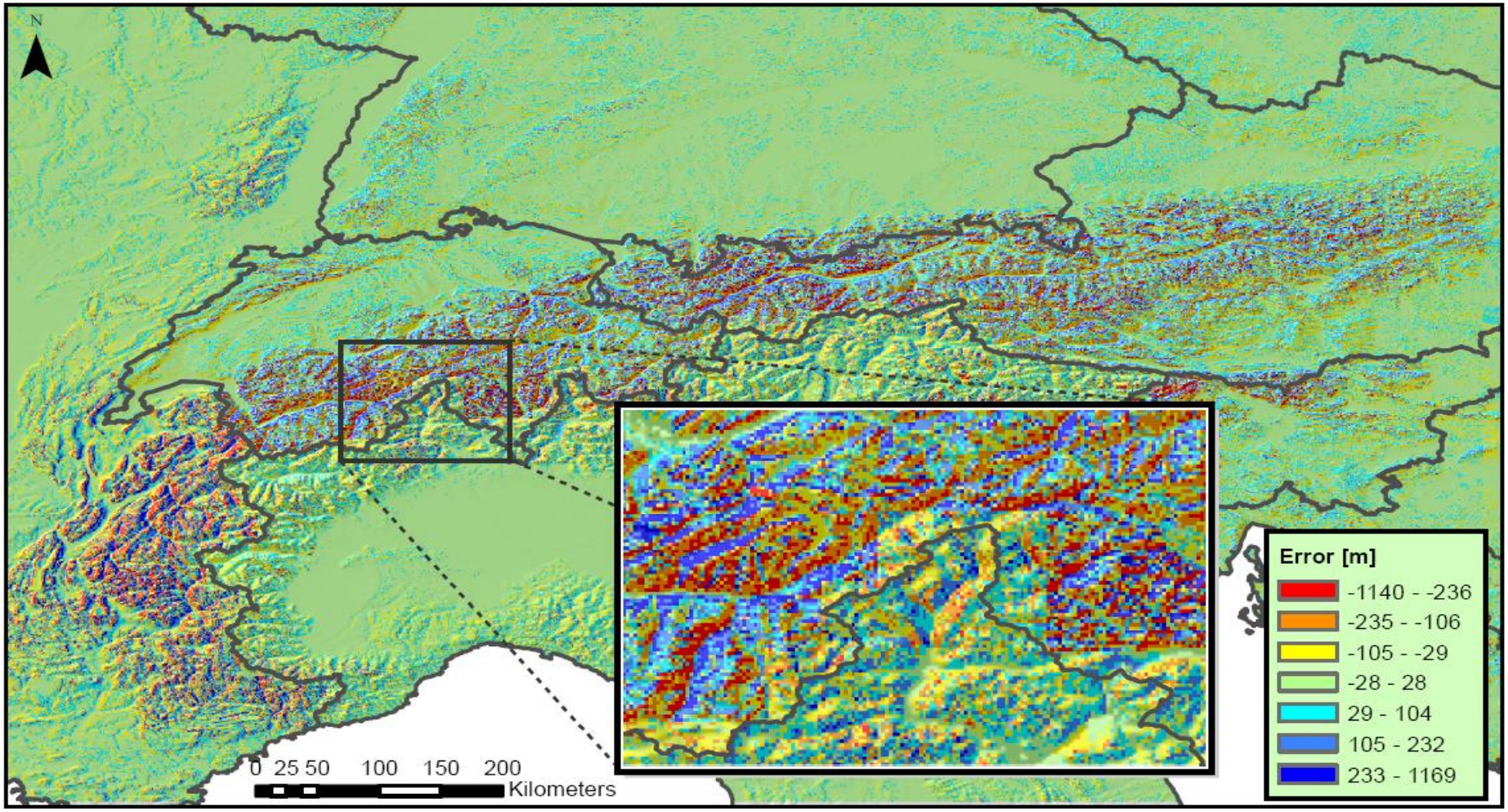
What's wrong with this approach?

- Remember, Regionalised Variable Theory from the interpolation lecture
- We said that for a variable which varies in space we could describe 3 components:
  - A **structural component** (e.g. a constant mean or trend m(x)).
  - A **regionalised variable**, a random, but spatially autocorrelated component (that is the variation which we model with a local interpolator (ε´(x)))
  - A spatially uncorrelated **random noise component** (variability which is not dependent at all on location ε")
- In the examples so far we have **treated error** as if it is only a function of the **last, random part ε"**

Considering spatial autocorrelation in MC simulation

- To consider **spatial autocorrelation**, we need to include a **regionalised variable**
- We need to **choose a value** for spatial autocorrelation – we need to know something about error structure
- We could explore this with a **more accurate dataset** and then **calculate the semivariogram**
- The error surface should have the same global properties (same RMSE, normally distributed) as random error

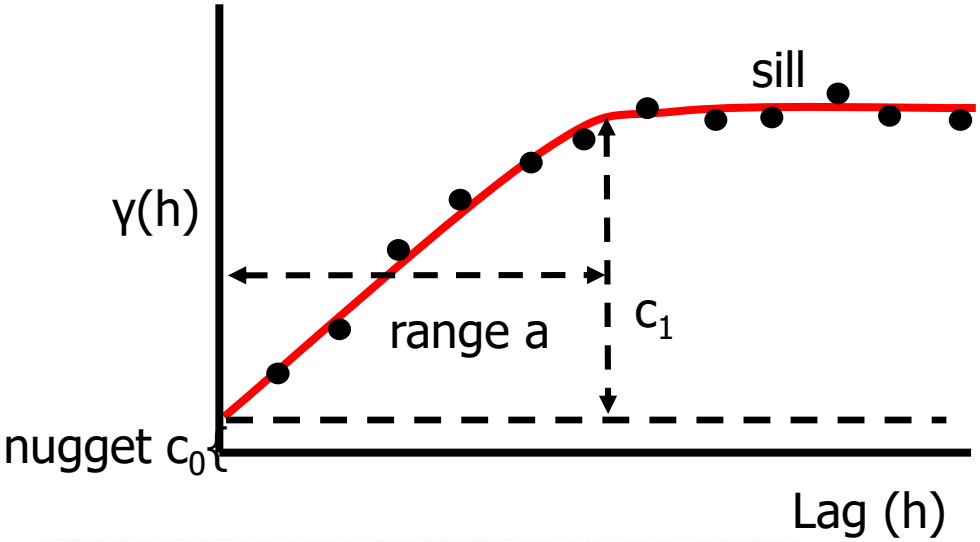
Autocorrelated error surface

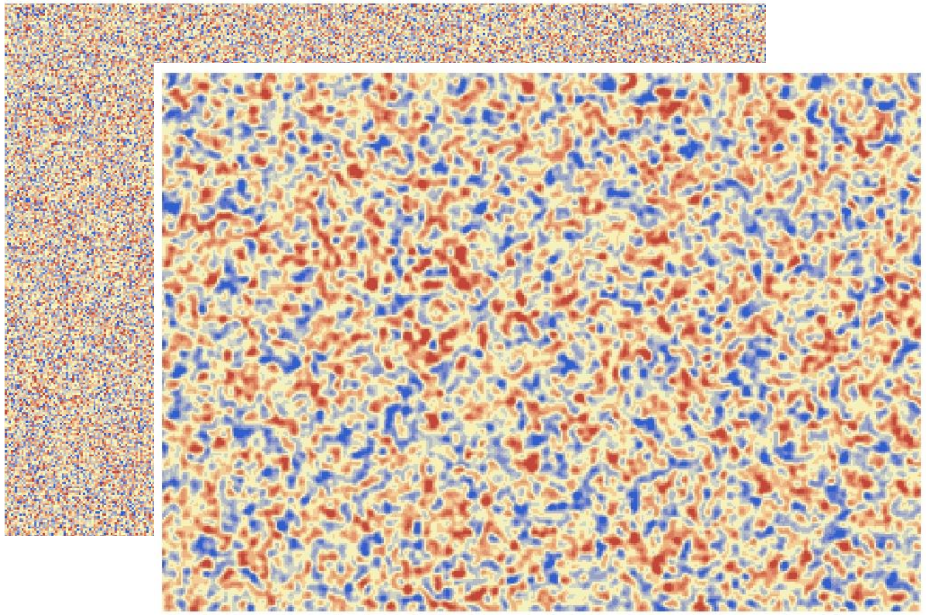


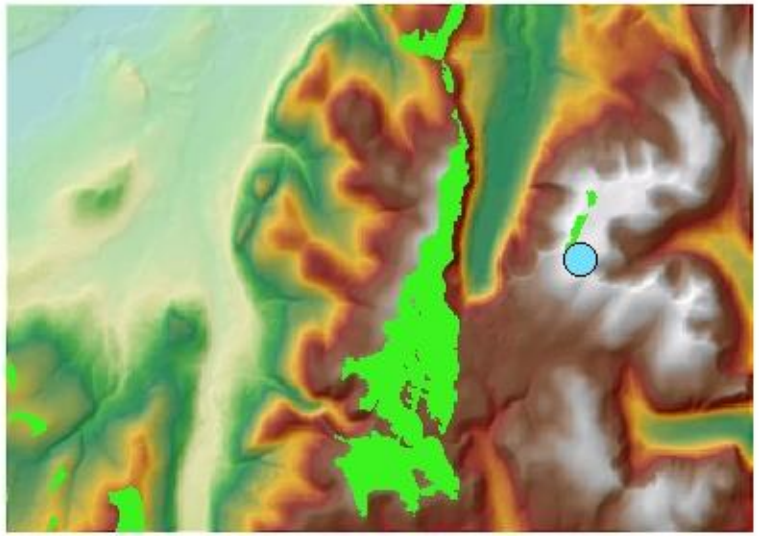
Error surface compares two datasets (GLOBE (~1km) with SRTM (~90m)) where we assume SRTM is "truth"

Spatially autocorrelated MC simulation

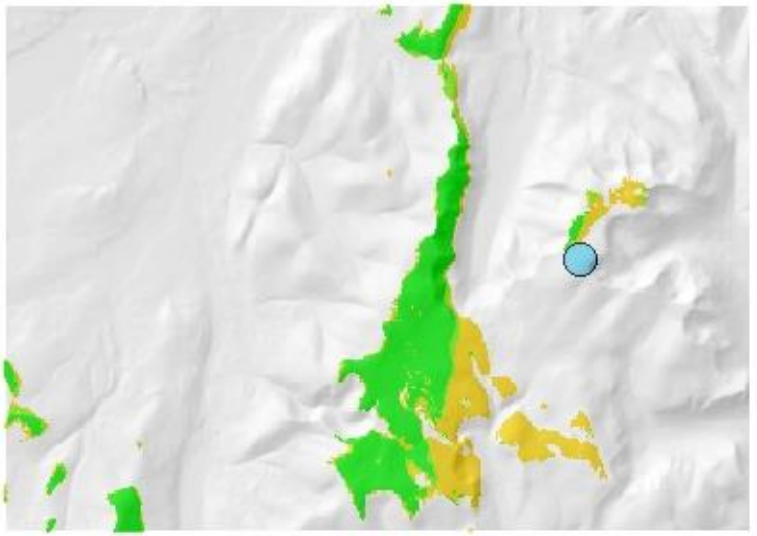
- The **range** of the **semivariogram** tells us about the **distance** over which error is **spatially autocorrelated**
- Often we have no "better" data to estimate this with, and we **guess** a **sensible value**
- In the following examples we created DEMs where the **error** was **spatially autocorrelated** with a **range** of about **250m** (i.e. 5x5 pixels)



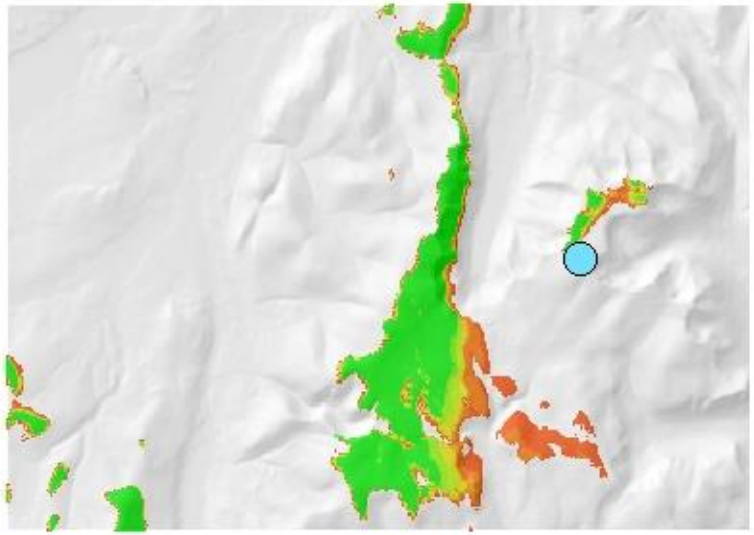




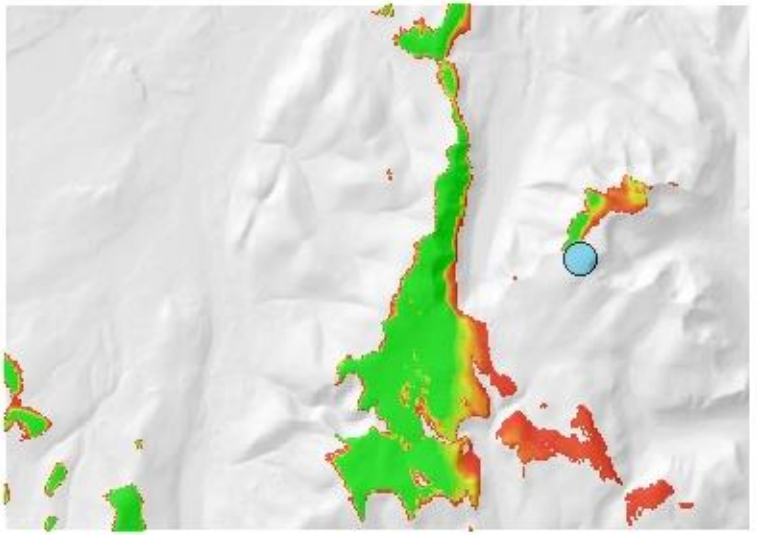
Original data
5606 cells visible



Viewshed mean (N=5)
Mean 5229 cells visible



Viewshed mean (N=25)
Mean 3957 cells visible



Viewshed mean (N=100)
Mean 4922 cells visible

Comments on uncertainty in viewshed

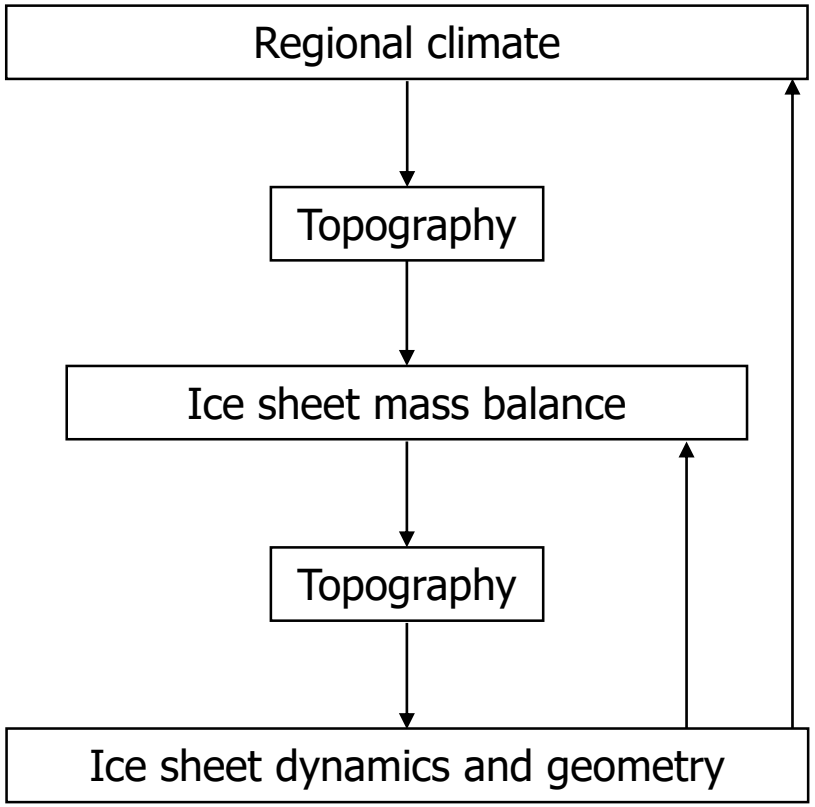
- Using uncorrelated errors reduced the viewshed size considerably
- This is unrealistic – the error surfaces **are rougher** than we would expect
- By introducing **spatial autocorrelation** (which we would expect in errors) we found the difference with the original viewshed to be less
- Note, that the viewshed still decreases in size though
- In general MCS without spatial autocorrelation gives **"worst case"** results

Applying MC simulation to a complex process model

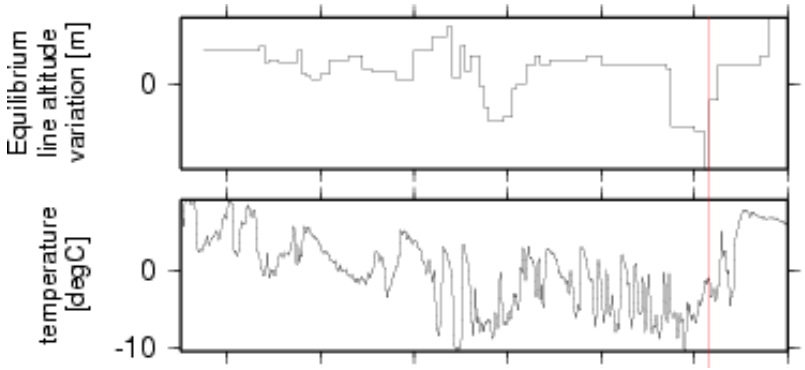
- So far I have showed examples using MC simulation for simple GIS functions – these can be calculated relatively quickly (which will take the longest?)
- Process models **model dynamic processes** through **time**
- These models are **much more complex** and a **single model run** may take **hours or days**
- We can apply MCS to these types of models too – often we need to calculate **uncertainty** for **many parameters**
- Going to look briefly at one example – ice sheet modelling

Modelling ice sheets

Modelling these,
not these

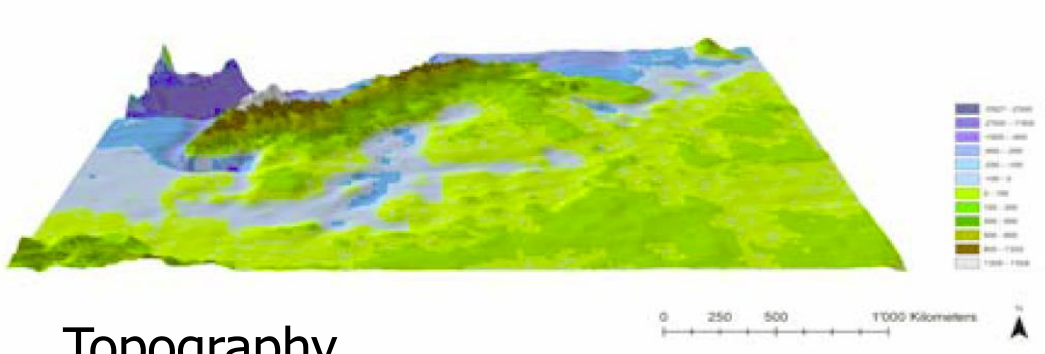


Role of topography in ice sheet modelling



Climate

+



Topography

+

Various parameters…

ISM

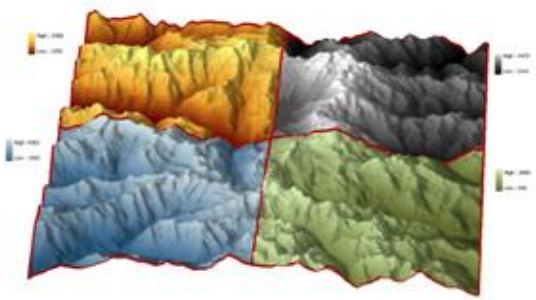
This example was produced using the GLIMMER ISM by Magnus Hagdorn.

(NB this model is no longer state of the art – not important for our discussion today)

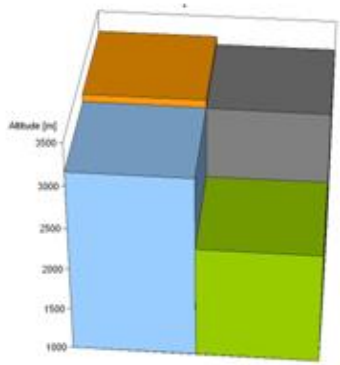
Research questions

- How can uncertainty in topography be incorporated in ice sheet modelling – one obvious way is using MCS
- What influence does uncertainty in topography have on the modelled extent and volume of ice sheets?

Zermat area

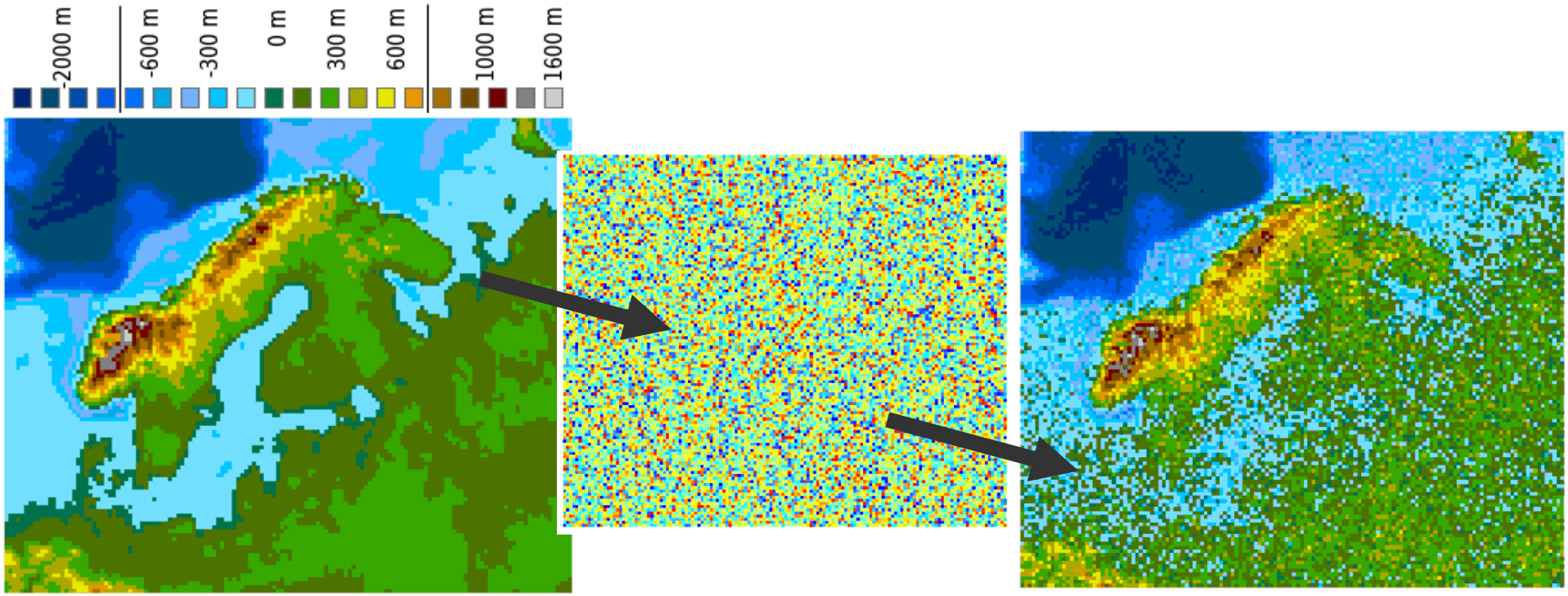


resolution ~90m



typical ISM resolution
~ 5km

Randomly distributed error

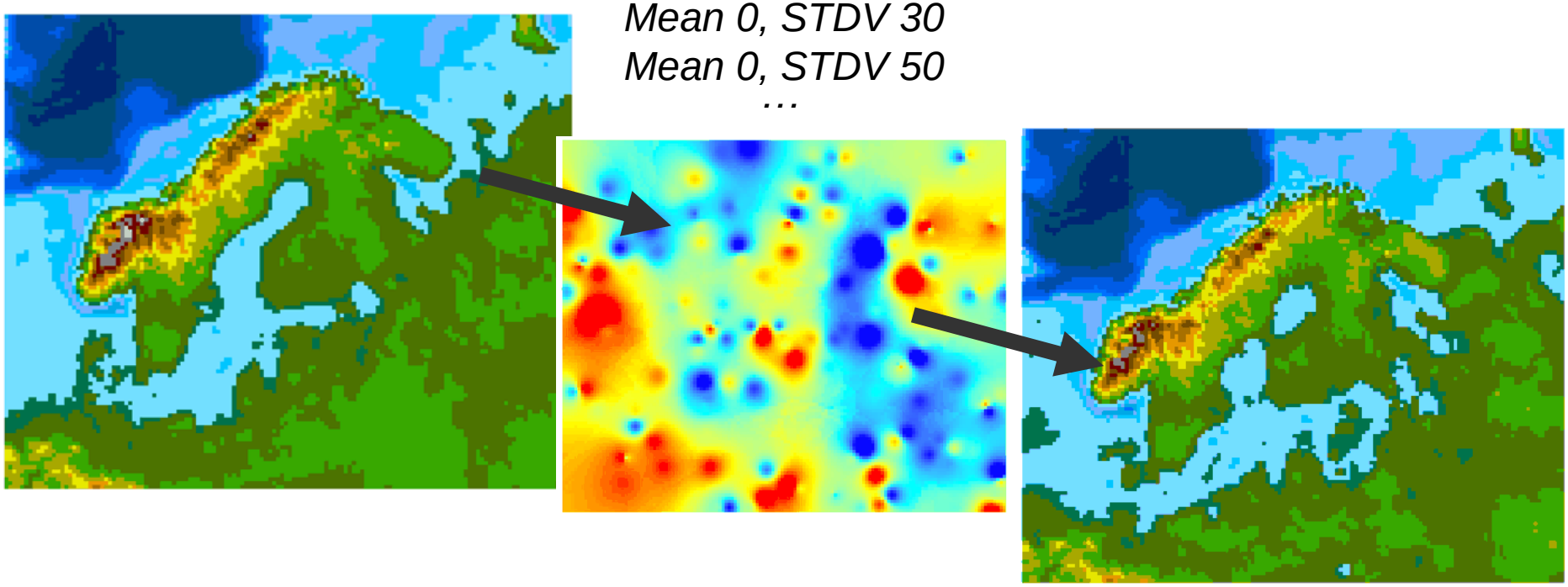


- Produced 150 surfaces as input for MCS, based on RMSE from GLOBE
- Surfaces produced are **noisy** (and unphysical – very large, abrupt changes in elevation) – **ice sheet model unstable**

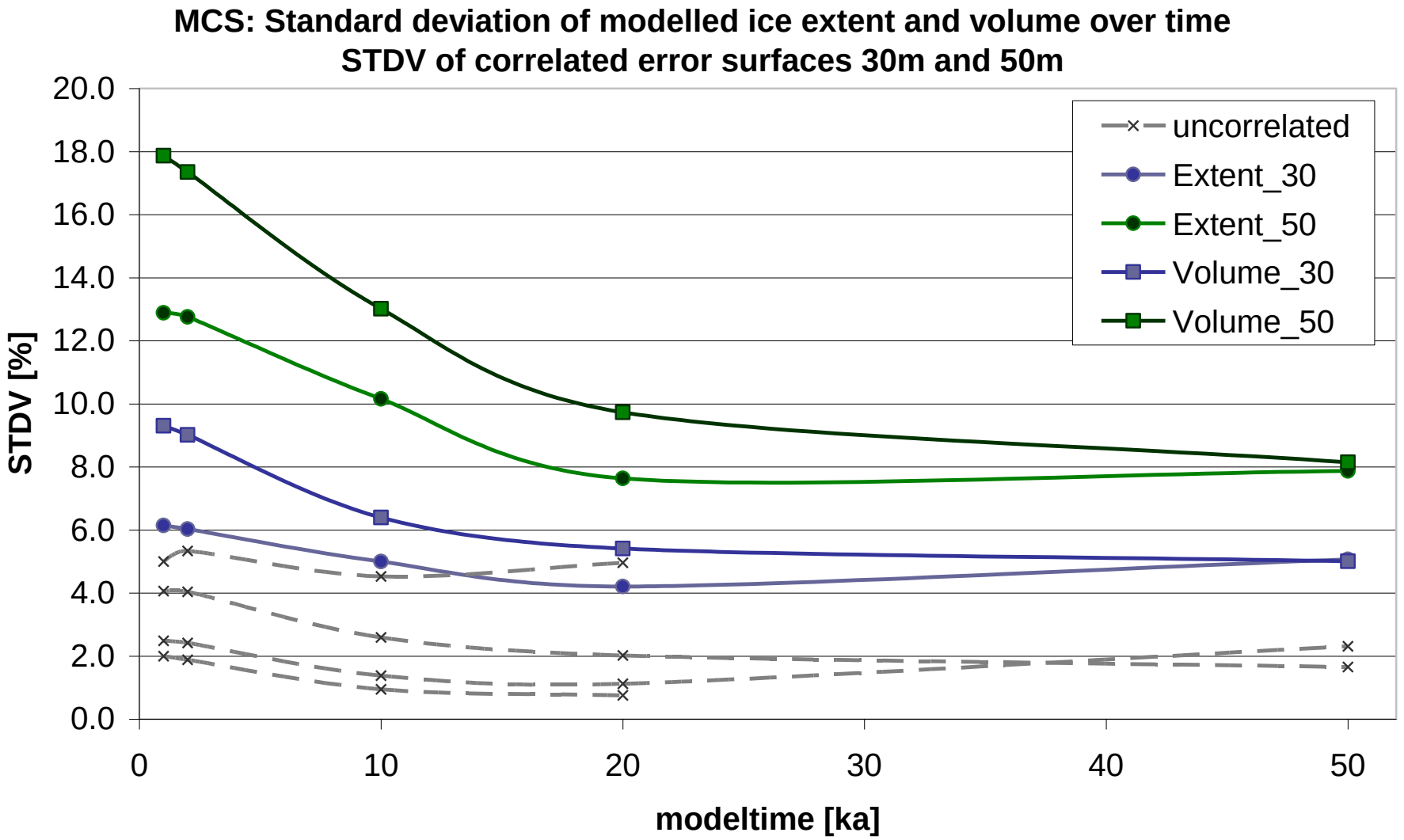
Spatially autocorelated error

- 150 **random points** selected within DEM extent
- Points **assigned values** with normally distributed z values (mean 0, STDV of 50 & 100m)
- Error surface **interpolated** using inverse distance weighting (IDW)

Mean 0, STDV 30
Mean 0, STDV 50
…



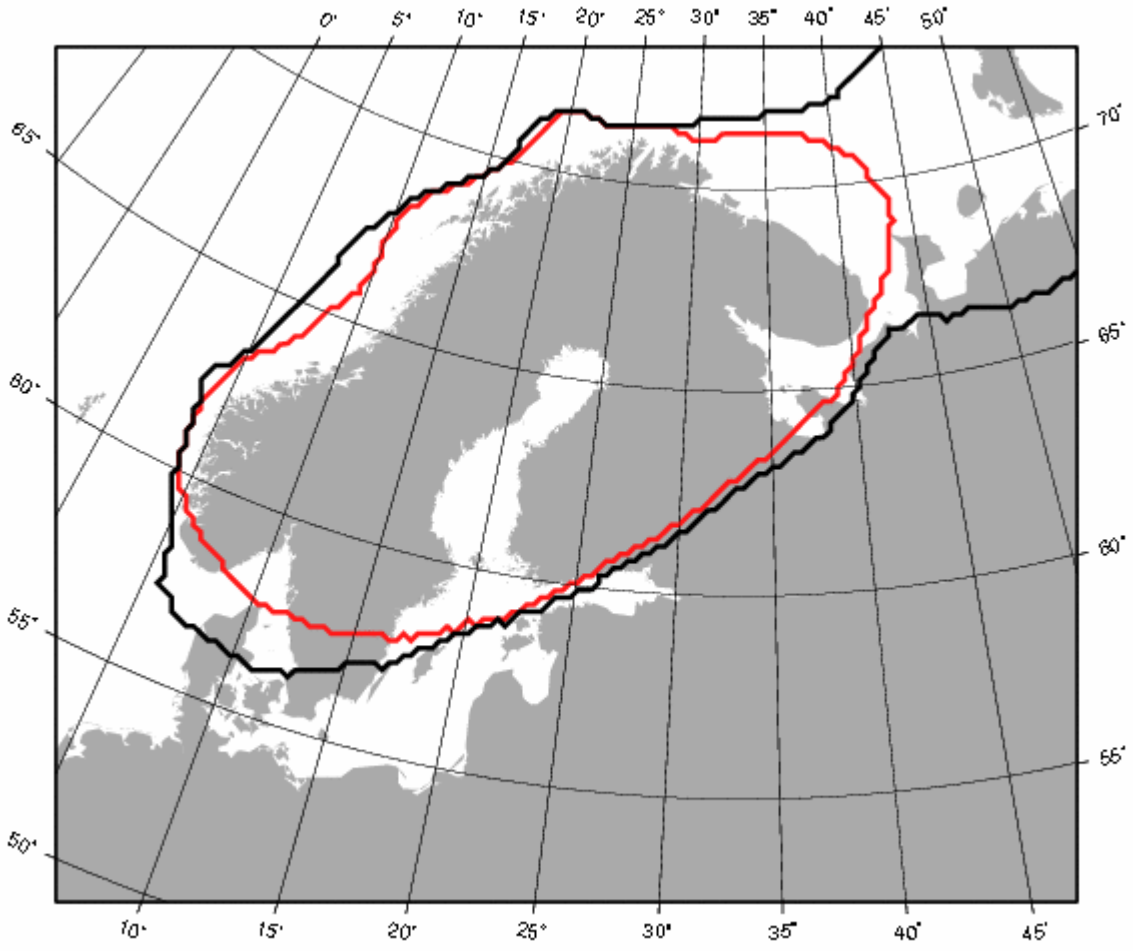
Results – spatially autocorrelated (1)



Results – spatially autocorrelated (2)

MCS using spatially autocorrelated correlated error ~50m STDV

Ice extents of minimum and maximum runs plotted on Scandinavian topography in steps of 10ka



Ice extent

minimum

maximum

Comments on MCS

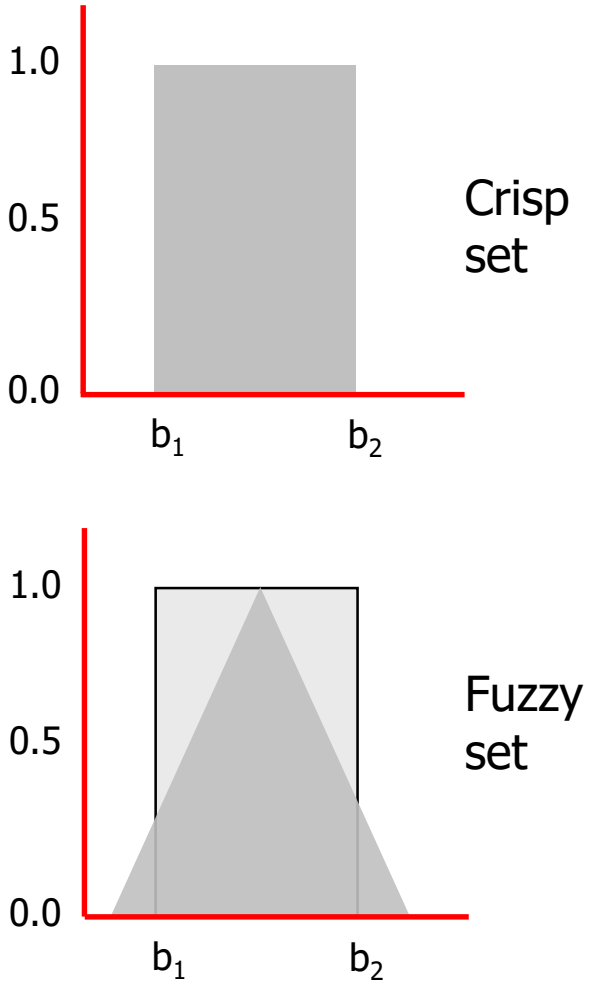
- MCS is a very powerful tool in exploring uncertainty
- To apply it properly we need to answer a number of questions:
  - **Which inputs** have **uncertainty?**
  - **How large** are these **uncertainties?**
  - Are they **normally distributed?**
  - Are they **spatially autocorrelated?**
- **How many iterations** are necessary to achieve a stable result?
- What **summary statistics** do we need to explore our results?
- In **process modelling**, surfaces should be **physically sensible**
- Because MCS is **computationally expensive** we should plan carefully

Fuzzy set methods

- So far, we have **classified geographic phenomena** as either **entities or field**
- We generally assume that we can query objects according to **crisp borders** or some sharp threshold (e.g. **tram stops** in **Zurich**; **slopes steeper** than **30°**)
- This way of thinking is based on Aristotelian notions
  - The law of **identity** (everything is what it is – a house is a house)
  - The law of **non-contradiction** (something cannot be a house and not a house)
  - The principle of the **excluded middle** (this slope is steep or not steep)
- *There are other ways of thinking – in Scotland the result of a legal process can be "guilty", "not guilty" or "not proven"*

Fuzzy set methods

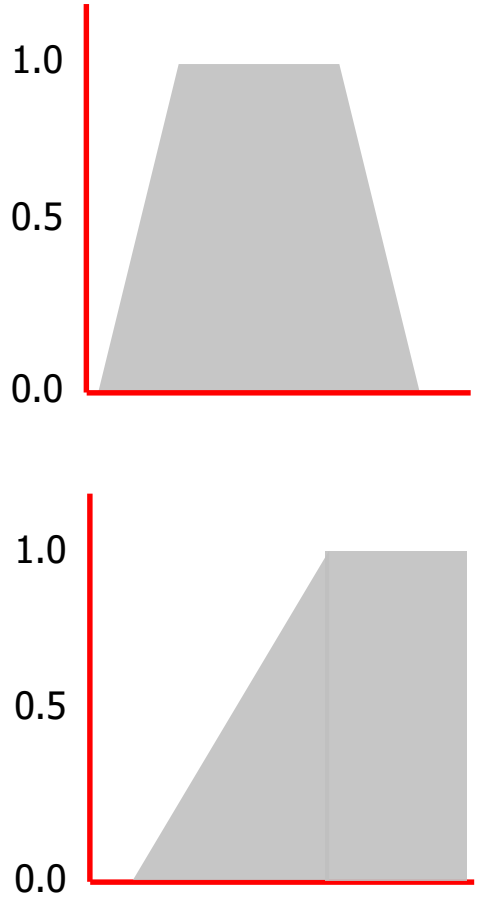
- Fuzzy sets are based around the idea that not all semantic classes or geometric objects have **sharp borders**
- Fuzziness is **not probability theory** – rather we are describing the **possibility** that some **statement is true**
- A **crisp set** can be described as follows:
  - $MF_B(z) = 1$ if $b_1 \leq z \leq b_2$
  - $MF_B(z) = 0$ if $z < b_1$ or $z > b_2$
  - For a crisp set the MFB is either 0 or 1
- A **fuzzy set** can be described as the set of pairs
  - $A = (z, MF_A(z))$ for all $z \varepsilon Z$
  - For a fuzzy set, the MF_A for any z is a value lying between [0,1]



Membership functions

More membership functions

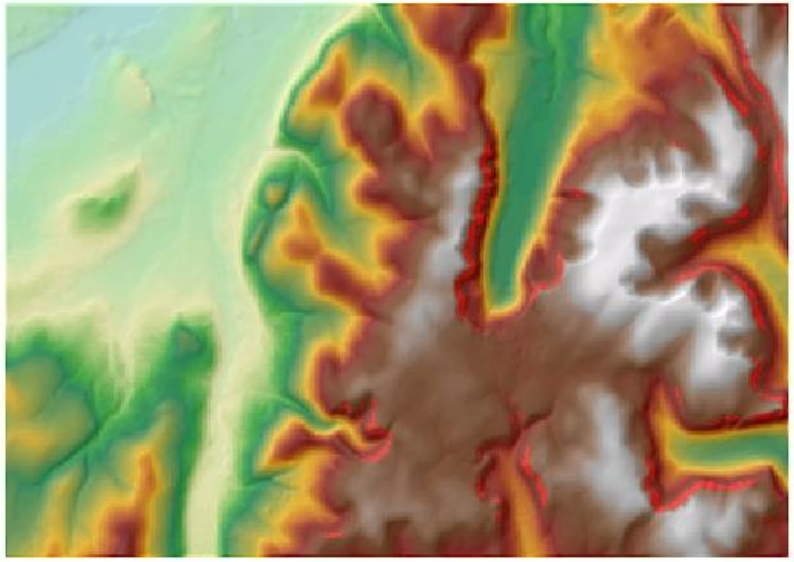
- The triangular membership function implied only one value is surely in our class
- More likely is a **trapezoid** – some region is surely in our class
- Function need not be symmetric – for instance the boundary between **steep** and **not steep** (or a linear function)



Example membership functions

Fuzzy sets example

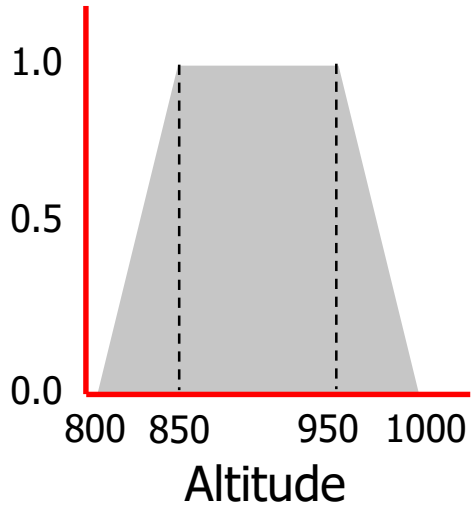
- We have been told that a **particular plant** is found on **steep slopes** of **quite high altitudes** in our test area
- We decide that steep slopes mean **>30°** and high altitudes between **850** and **950m**

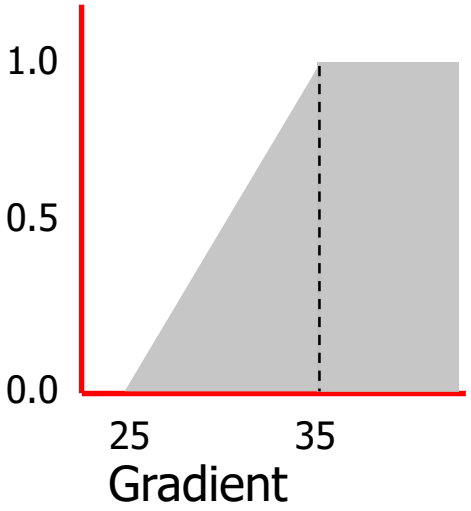


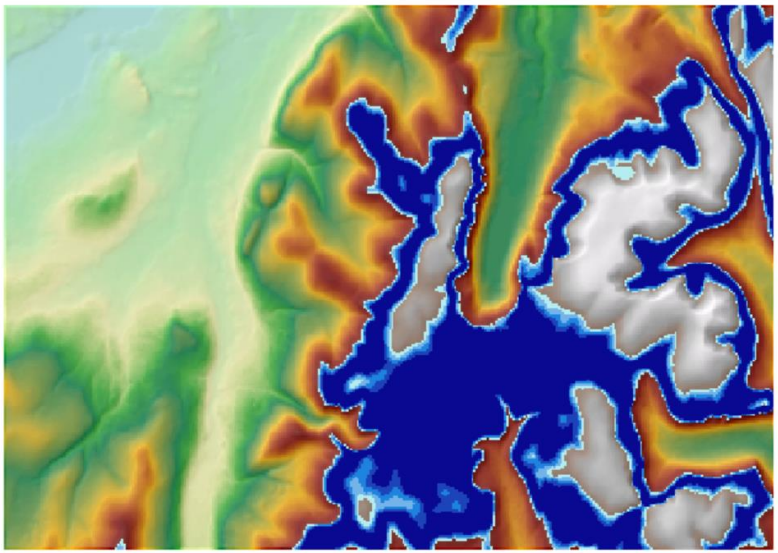
The **red areas** show the regions which belong to this **crisp set** (**1248 pixels**)

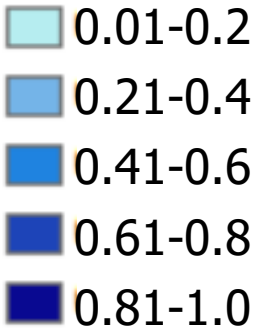
Fuzzy sets example

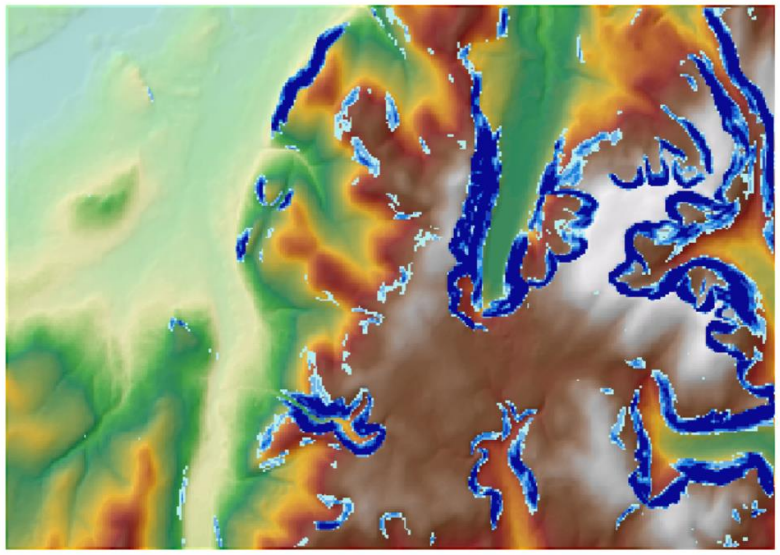
- We use the following membership functions to calculate fuzzy sets





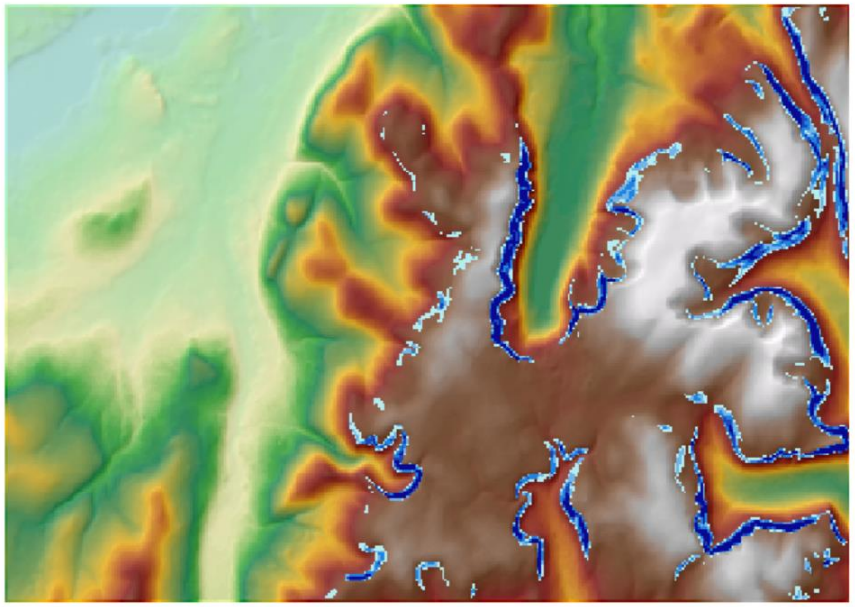






Fuzzy overlay

- We can combine fuzzy sets in many different ways
- One common approach is to say **U = min(a,b)** where a and b are the membership functions at a location



The shaded areas show the regions which are found in this fuzzy set to be ***possible locations where U>0*** (4748 pixels)

Fuzzy sets – strengths and weaknesses

- Main application of fuzzy sets in GIScience is in modelling with vague (steep, not steep) or uncertain (gradient is 30 ± 20°) data
- Fuzzy sets incorporate this vagueness into our calculations and so *better* represent a world which is **not crisp**
- Problems:
  - How do we **choose** the **membership function**? **Different functions** can lead to very **different results**
  - How do we **communicate** the **results to users** – **conceptually difficult** to understand and **practically difficult to visualise** (i.e. what does a membership function of 0.3 mean?)

Summary

- We looked at how we can use **simple error theory** to **estimate errors** and their **sensitivity** to individual factors
- We saw how **MCS** could be used to **model errors** in typical GIS outputs such as slope, stream power and viewshed
- We looked at how we can deal with problems where we **don't have crisp sets** and want to deal with **vague** or **unsharp boundaries** (either semantically or geometrically)

Next week

- We will look at multi-criteria analysis for problem solving
- How do we take account of a range of factors in assessing problems in space?
- How do we integrate the opinions of decision makers including citizens in the process?

References

Burrough et al. (2015): *Principles of Geographical Information Systems*. Second Edition. Oxford University Press.

Fisher, P.F. (1998): Improved Modeling of Elevation Error with Geostatistics. *GeoInformatica*, **2**(3): 215-233.

Hebeler, F., & Purves, R. S. (2009). The influence of elevation uncertainty on derivation of topographic indices. *Geomorphology*, *111*(1), 4-16.

Hebeler, F., Purves, R. S., & Jamieson, S. S. (2008). The impact of parametric uncertainty and topographic error in ice-sheet modelling. *Journal of Glaciology*, *54*(188), 899-919.

Heuvelink, G.B.M. (1998): *Error Propagation in Environmental Modelling with GIS*. London: Taylor & Francis.

GITTA materials on fuzzy sets at: http://gitta.info/Suitabilityi/en/html/index.html