UNIVERSITETET I OSLO

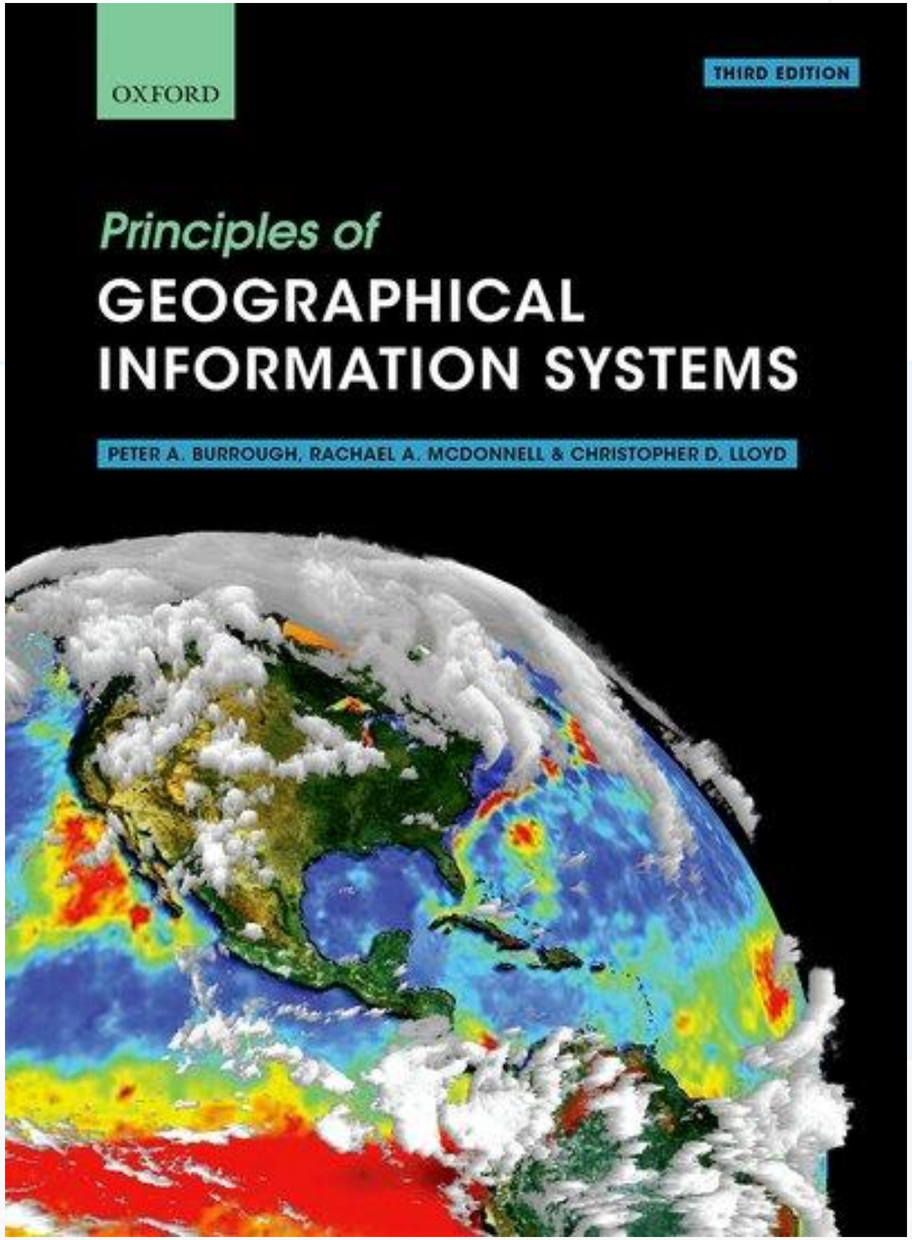
GEO3460 – Landmåling, fotogrammetri og romlig analyse – vår 2024

GEO3460 - Surveying, photogrammetry and spatial analysis - spring 2024

Interpolation

Martin Lund (e.m.lund@geo.uio.no)

Based on the lectures by Dr. Livia Piermattei



- Reference text book:

"*Geographical information systems*" (not available online)

  - Chapter 8

Learning Objectives

1

GIS interpolation
- What?
- When?
- Data sources for interpolation

2

- Spatial Interpolation Methods
- Interpolation algorithms

3

Interpolation examples and application

Today's topics

Learning Objectives

1

GIS interpolation
- What?
- When?
- Data sources for interpolation

2

- Spatial Interpolation Methods
- Interpolation algorithms

3

Interpolation examples and application

Today's topics

Interpolation – What and When

- Have you ever heard the word “interpolation”?
- Can you define it?
- and come up with some examples?

Interpolation – What and When

- **What**
  - The prediction of data values at locations where we have no measurements
- **Point interpolation**
  - Convert data from point observations to continuous fields
  - Predicting the value of attributes at unsampled sites from measurements made at point locations within the same area
- **When**
  - Changing grid cell size
  - Converting grids/images between coordinate systems
  - Conversion between different data models
  - Generating continuous surfaces from discrete data

What is a spatial interpolation?

- **The basis assumption of interpolation:**
  - On average, the values of points close to sampled points are more likely to be similar than those farther away.

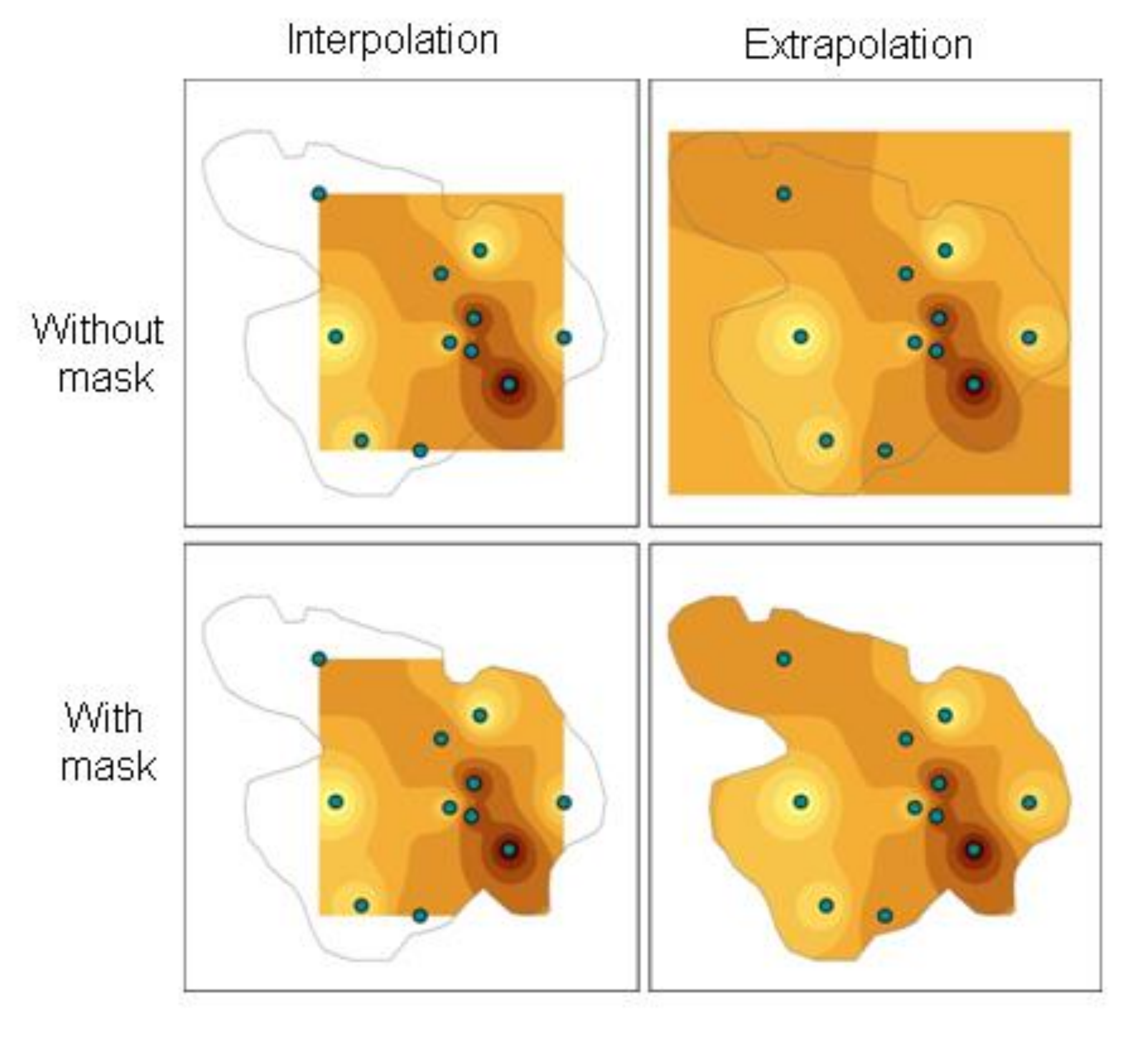
> *"Everything in the universe is related to everything else, but closer things are more related."* – Tobler's First Law of Geography

What is a spatial interpolation?

We can predict conditions at unsampled locations on the basis of information from the nearest available measured points.

- ☐ ***Spatial interpolation*** when the predictions are made within the spatial extent of the measured point locations

- ☐ ***Spatial extrapolation:*** predictions are made outside the spatial extent of the measured points



What is a spatial interpolation?

- **Examples of spatial interpolation (or extrapolation)**

  1. elevation between survey benchmark locations to create a digital elevation model (DEM);

  2. meteorological conditions such as precipitation or temperature at locations other than weather stations;

  3. a continuous air pollution surface from a network of regulatory monitoring stations (e.g. measuring NO2, ozone, etc.)

  4. indicators of water quality (e.g. salinity, temperature, dissolved oxygen) in surface water bodies based on field measurements at a set of sampling locations.

  – Do you have other examples?

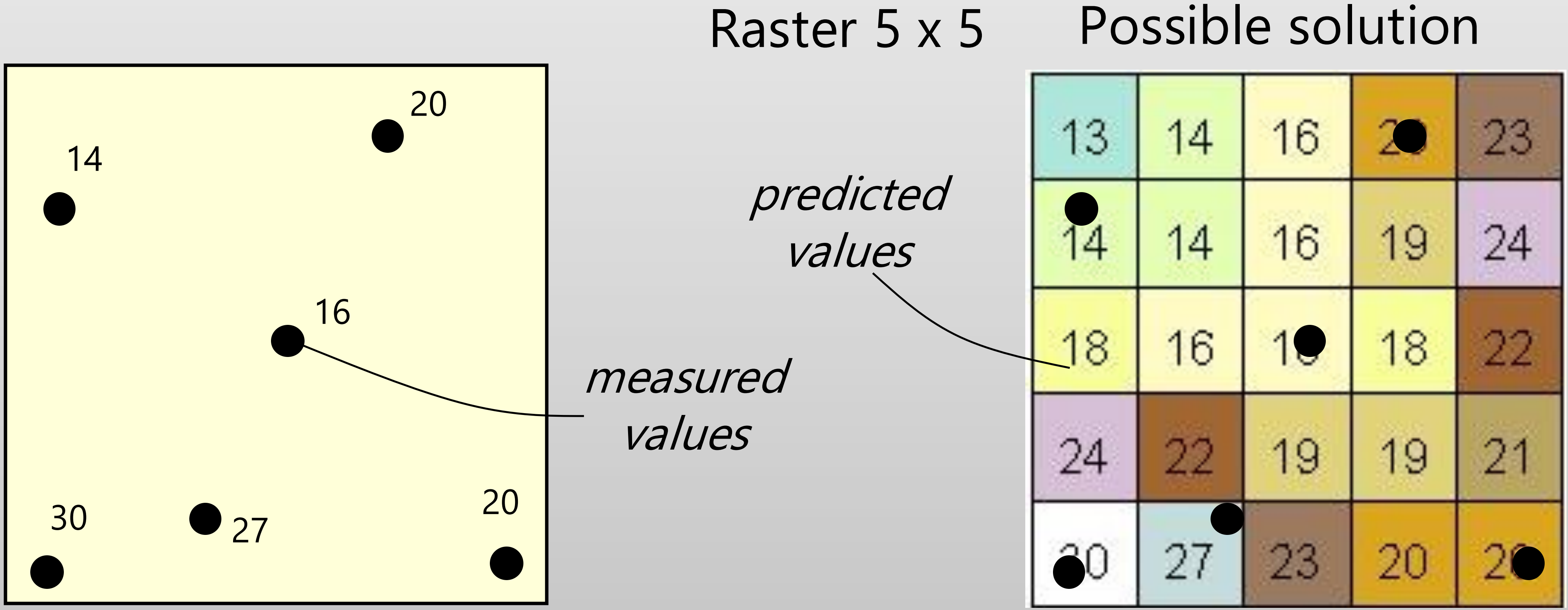
What is a spatial interpolation?

For a given sample of measurements $\{z_1, z_2, \ldots, z_n\}$
at locations $\{x_1, x_2, \ldots, x_n\}$

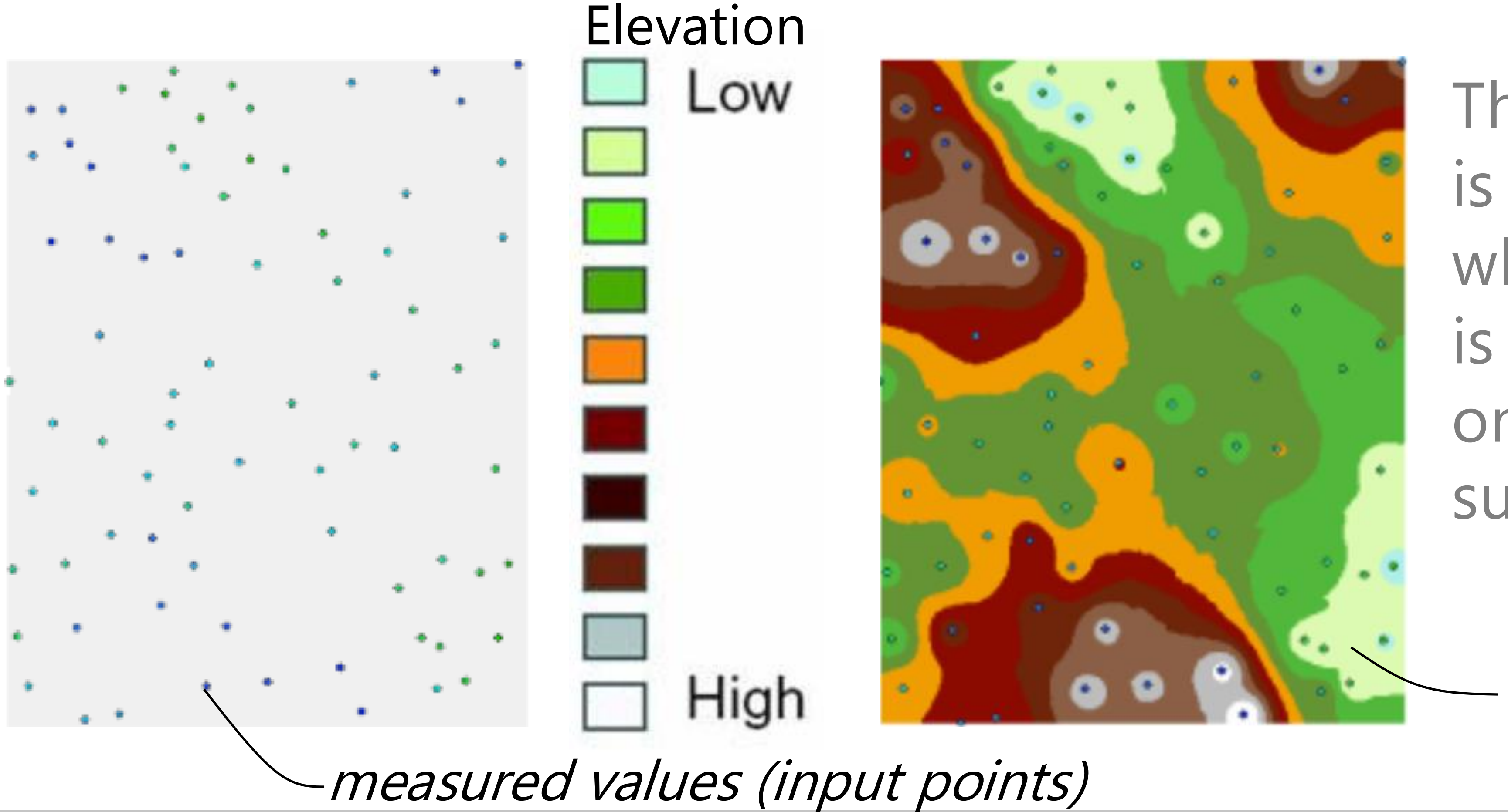
…the main goal is
to **estimate** the **value z**
at some new point x

When: Generating continuous surfaces from discrete data (samples)

When: Generating continuous surfaces from discrete data (samples)

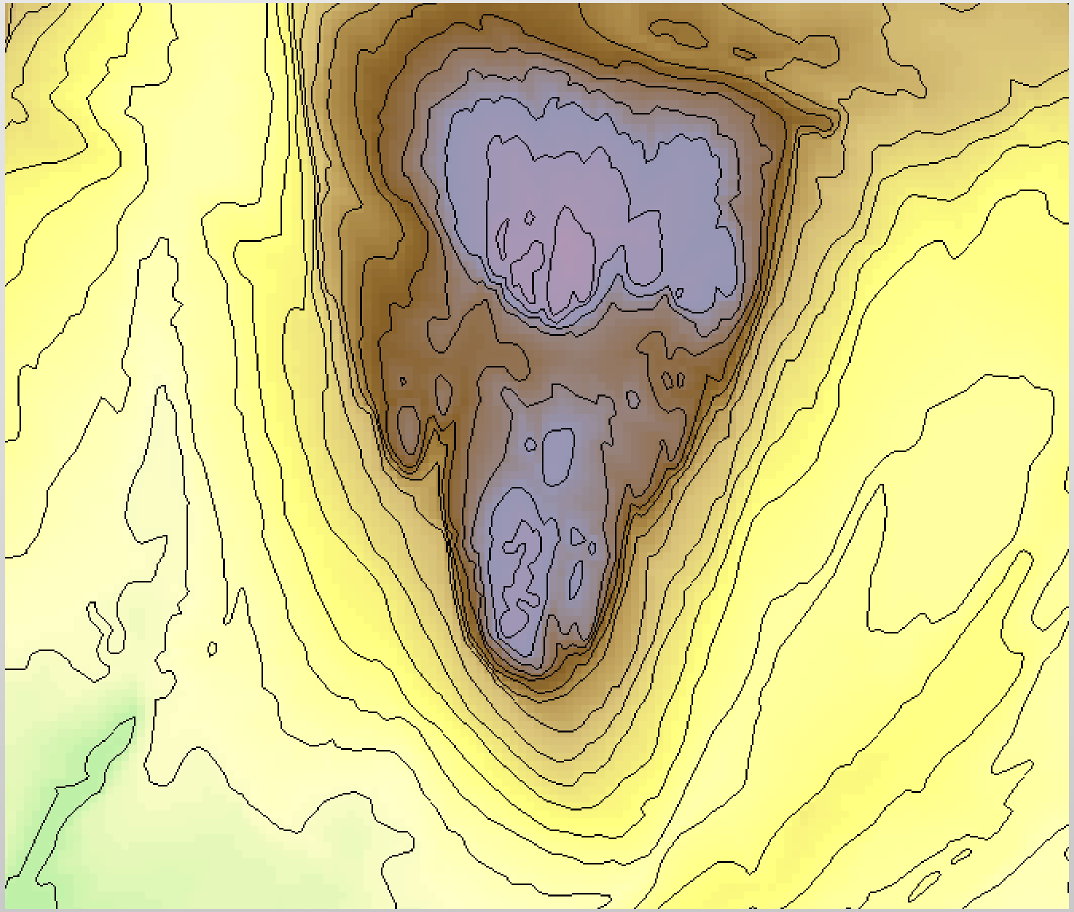


When: Generating continuous surfaces from discrete data (samples)



The resulting grid is a prediction of what the elevation is at any location on the actual surface.

When: Conversion between data models



Vector

↓

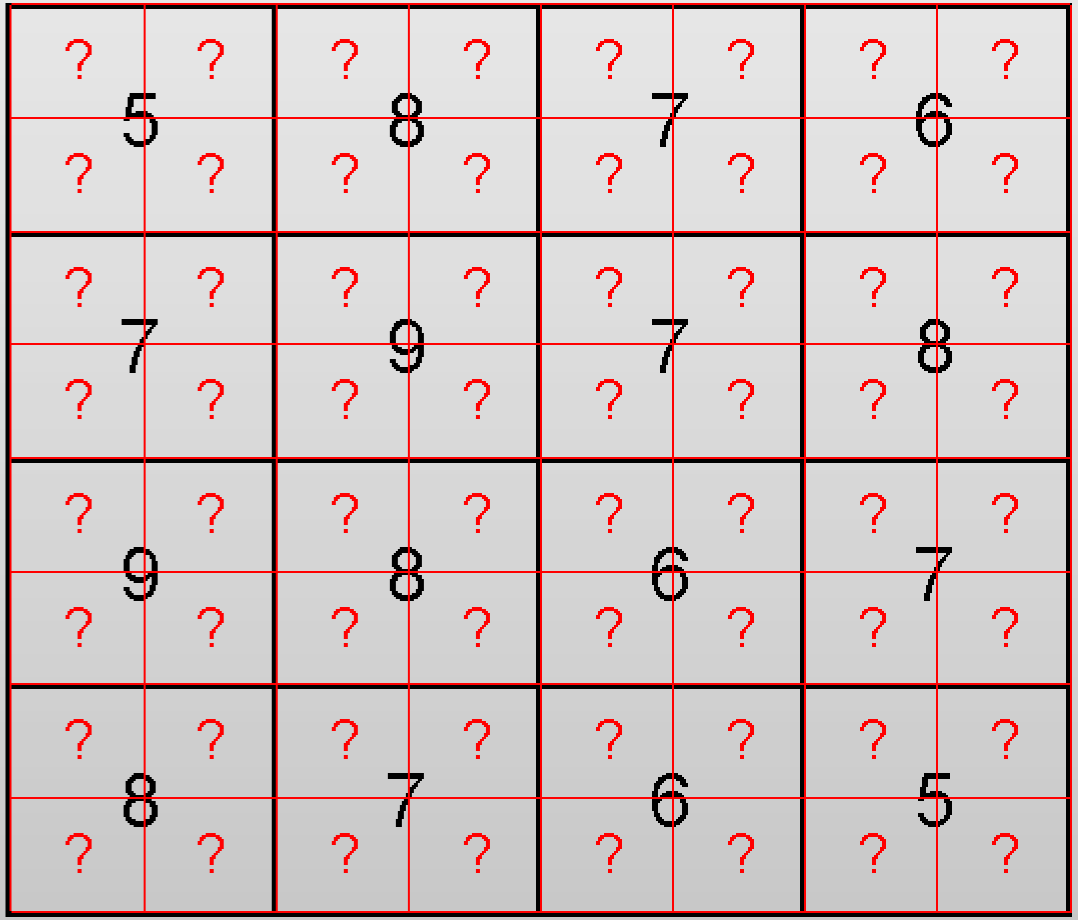
TIN

↓

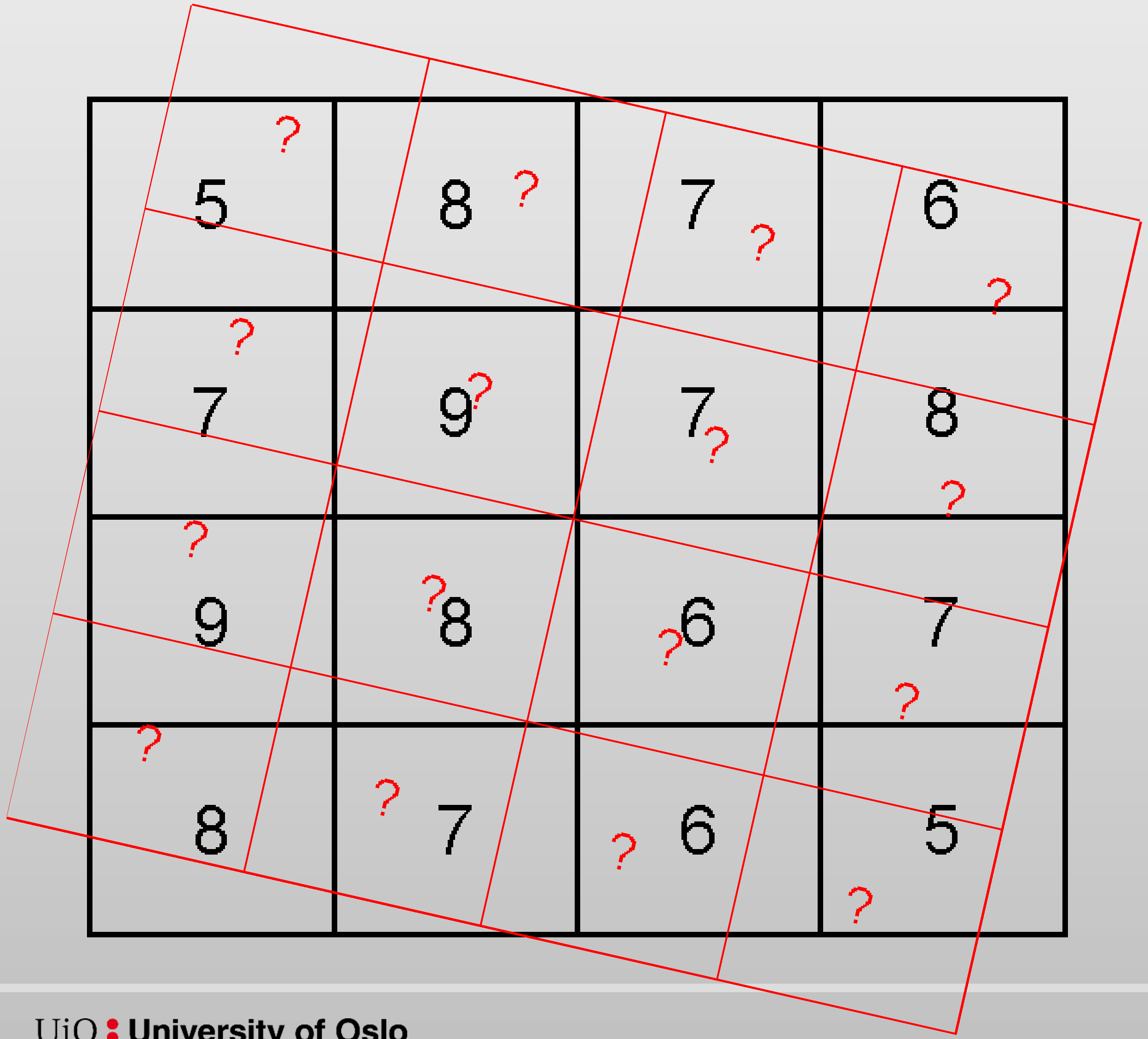
Grid

When: Changing cell size

When: Changing cell size



When: Transforming to another coord. sys.



Data sources for interpolation

- Point samples of attributes measured in the field
- Remote sensing (airborne and spaceborne)
- Stereo images (for elevation data)

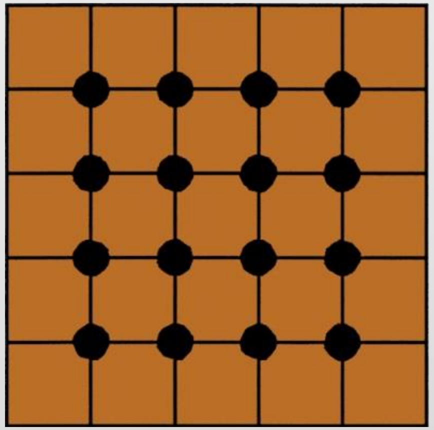
Spatial sampling (data/points)

Spatial sampling (data/points)

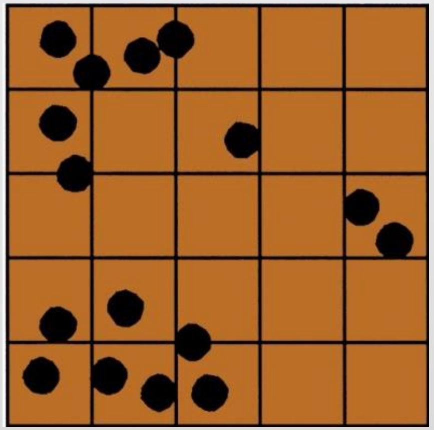
- **Two characteristics of sampling**
  - Location of samples
  - Number of samples (point density)

- **Sometimes we can not control sampling**
  - Cost of sampling
  - Available resources
  - You may be limited to occurrences of an event

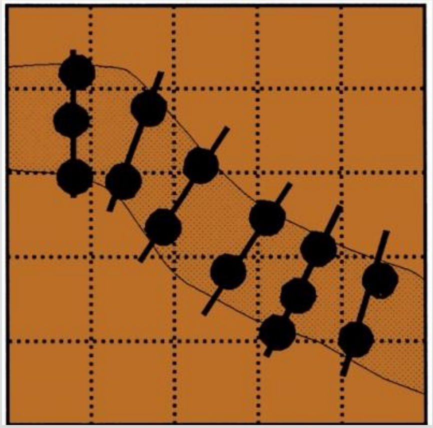
Spatial sampling: Common sampling pattern



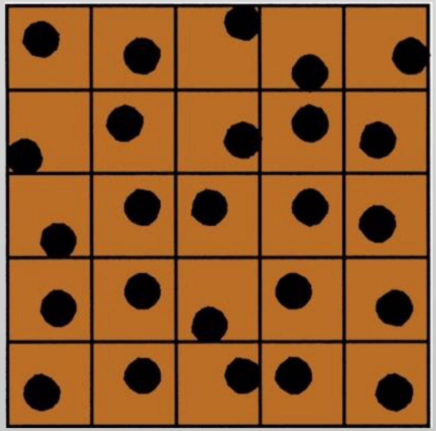
regular sampling



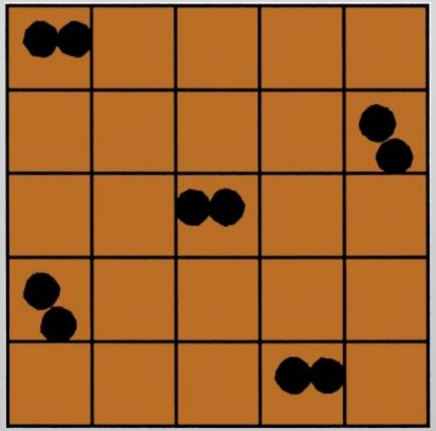
random sampling



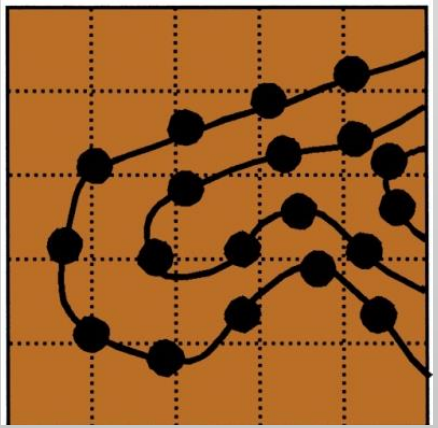
transect sampling



Stratified random sampling



cluster sampling



contour sampling

Burrough et al. 2015

Learning Objectives

1

GIS interpolation

- What?
- When?
- Data sources for interpolation

2

- Spatial Interpolation Methods
- Interpolation algorithms

3

Interpolation examples and application

Today's topics

Spatial Interpolation Methods?

- **No interpolation method is superior for all datasets**
- **Method choice depends on:**
  - Characteristics of the sample data (variable) to be measured
    - Sampling pattern
  - Accuracy requirements of the users
- **Methods differ in:**
  - the mathematical functions used to weight each observation
  - the number of observations used

How interpolate?

- Deterministic vs. stochastic (statistical)
- Global vs. local
- Exact vs. inexact/approximation
- Error estimate vs. no error estimate

How interpolate

- **Deterministic methods:**
  - Deterministic Models use a mathematical function to predict unknown values and result in hard classification of the value of features.
  - Example: splines, IDW, Natural Neighbour, Trend

- **Statistical (geostatistical) methods:**
  - Statistical techniques are based on statistical models that include autocorrelation: the statistical relationship among the measured points.
  - Provide some measure of the accuracy of the predictions.
  - Example: kriging

How interpolate

- **Global interpolators**
  - Use all available data to provide predictions
  - Used for examining trends
- **Local interpolators**
  - Operate within a small zone around the point being interpolated

How interpolate

- **Inexact interpolation:**
  - The measured value is not reproduced
  - In these methods the differences between measured and predicted values can be used as an indicator of the quality
- **Exact interpolation:**
  - The predicted value at a sample point is identical to the measured value
  - In these methods we may use an independent dataset to test the quality

Interpolation algorithms

- ☐ **Trend surfaces** (global polynomial interpolation, GPI)
- ☐ **Nearest neighbour (Thiessen)**
- ☐ **Inverse distance weighting (IDW)** (moving average)
- ☐ **Radial basis functions or Spline**
- ☐ **Kriging** (ordinary, simple, universal, etc....analyses of spatial variation)

next lecture

Interpolation algorithms

- ☐ **Trend surfaces** (global polynomial interpolation, GPI)
- ☐ **Nearest neighbour (Thiessen)**
- ☐ **Inverse distance weighting (IDW)** (moving average)
- ☐ **Radial basis functions or Spline**
- ☐ **Kriging** (ordinary, simple, universal, etc….analyses of spatial variation)

Interpolation algorithms: Trend surfaces

- Fit a polynomial line f(x) or surface f(x,y) by least squares through the sampled points.
  - Trend is based on regression. Regression analysis is the optimized fitting of a line or curve to derive a mathematical function that best fits the raw data.

Linear example:

$z(x) = b0 + b1x + \varepsilon$

- where b0 is the intercept, b1 is the slope and ε is the residual (noise)
- By increasing the number of terms it is possible to fit any set of points by a complicated curve, thereby reducing ε to zero

Surface example (two dimensions):

linear: $z(x, y) = b0 + b1x + b2y$

quadratic (polynomial regression): $z(x, y) = b0 + b1x + b2y + b3x2 + b4xy + b5y2$

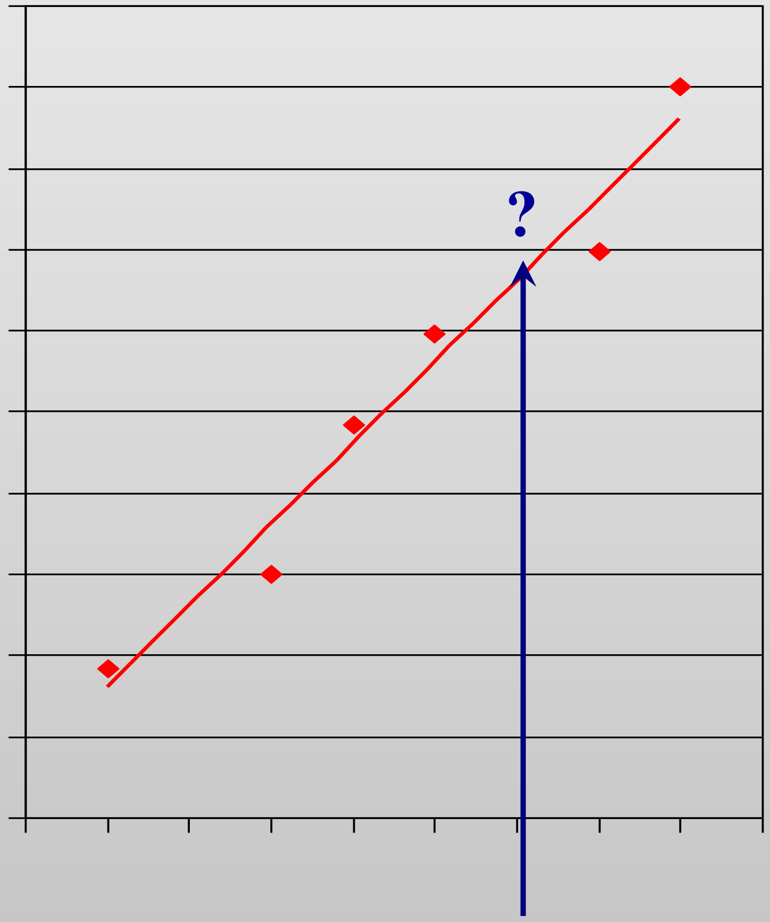
Interpolation algorithms: Trend surfaces

- **Linear interpolation**

  <u>Least square fit</u>
  the sum of the squared distances between the predicted and the measured points is minimised

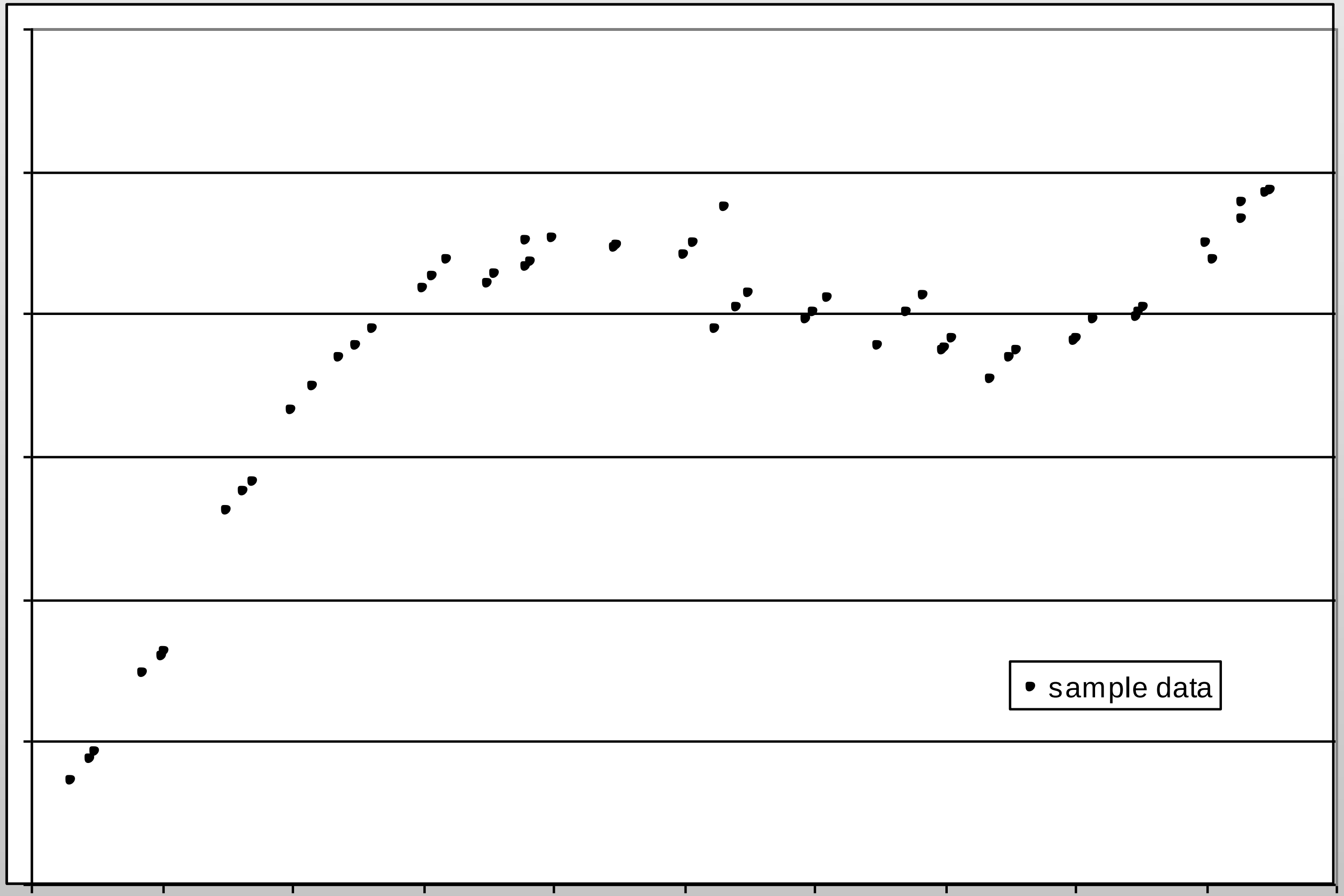
  $z(x) = b0 + b1x + \varepsilon$

  - where b0 is the intercept, b1 is the slope and ε is the residual (noise)



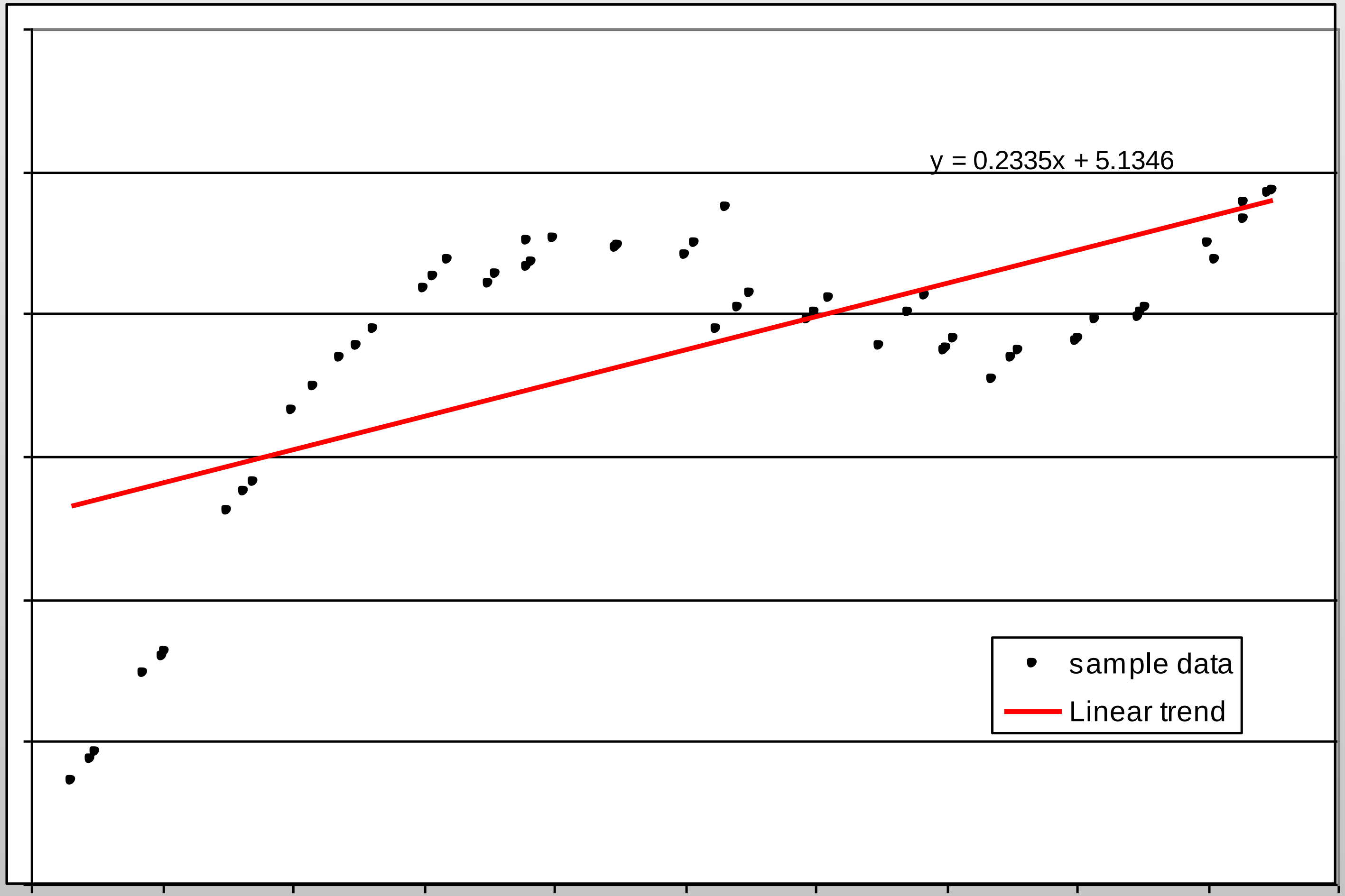
Interpolation algorithms: Trend surfaces

- Linear interpolation



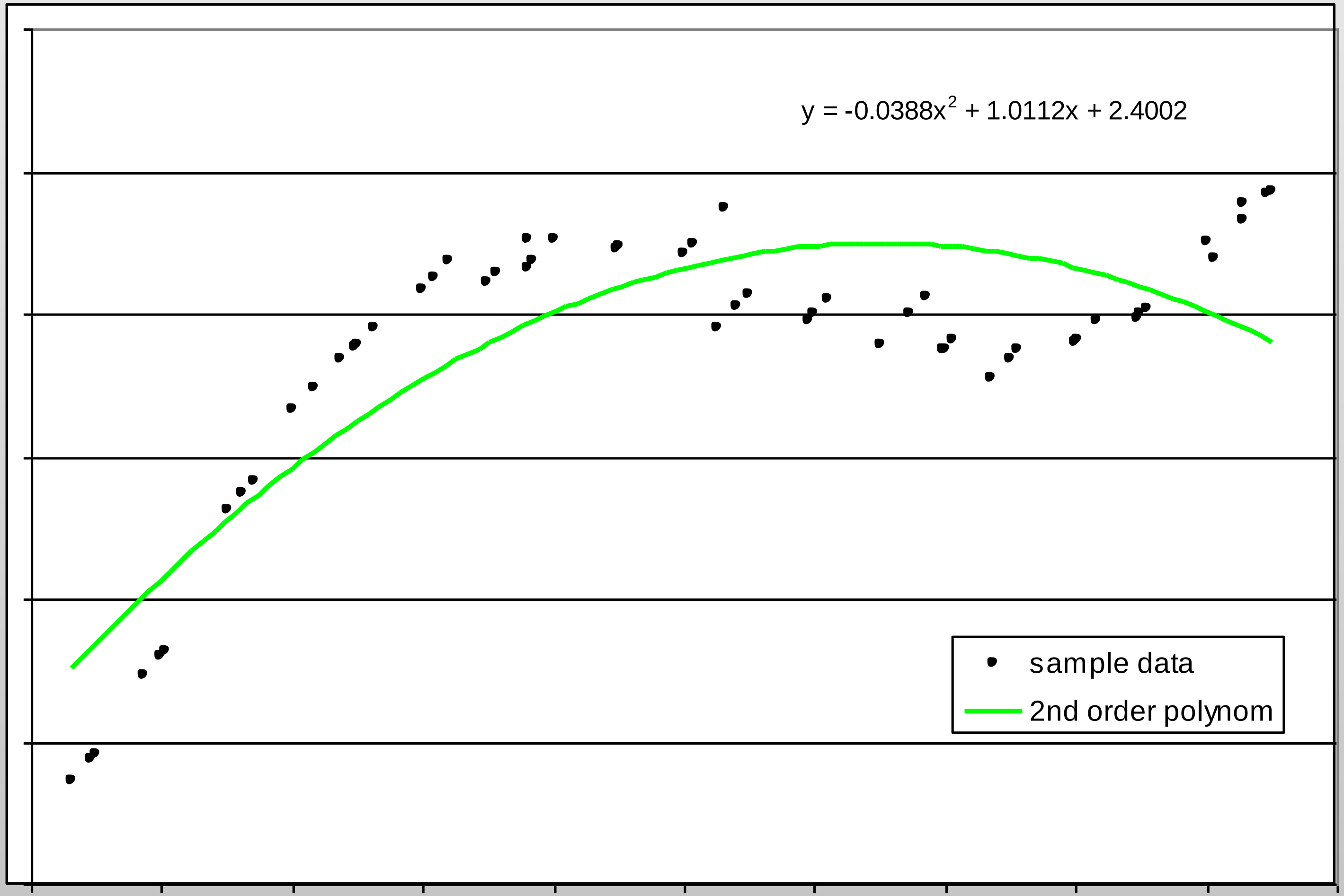
Interpolation algorithms: Trend surfaces

- Linear interpolation



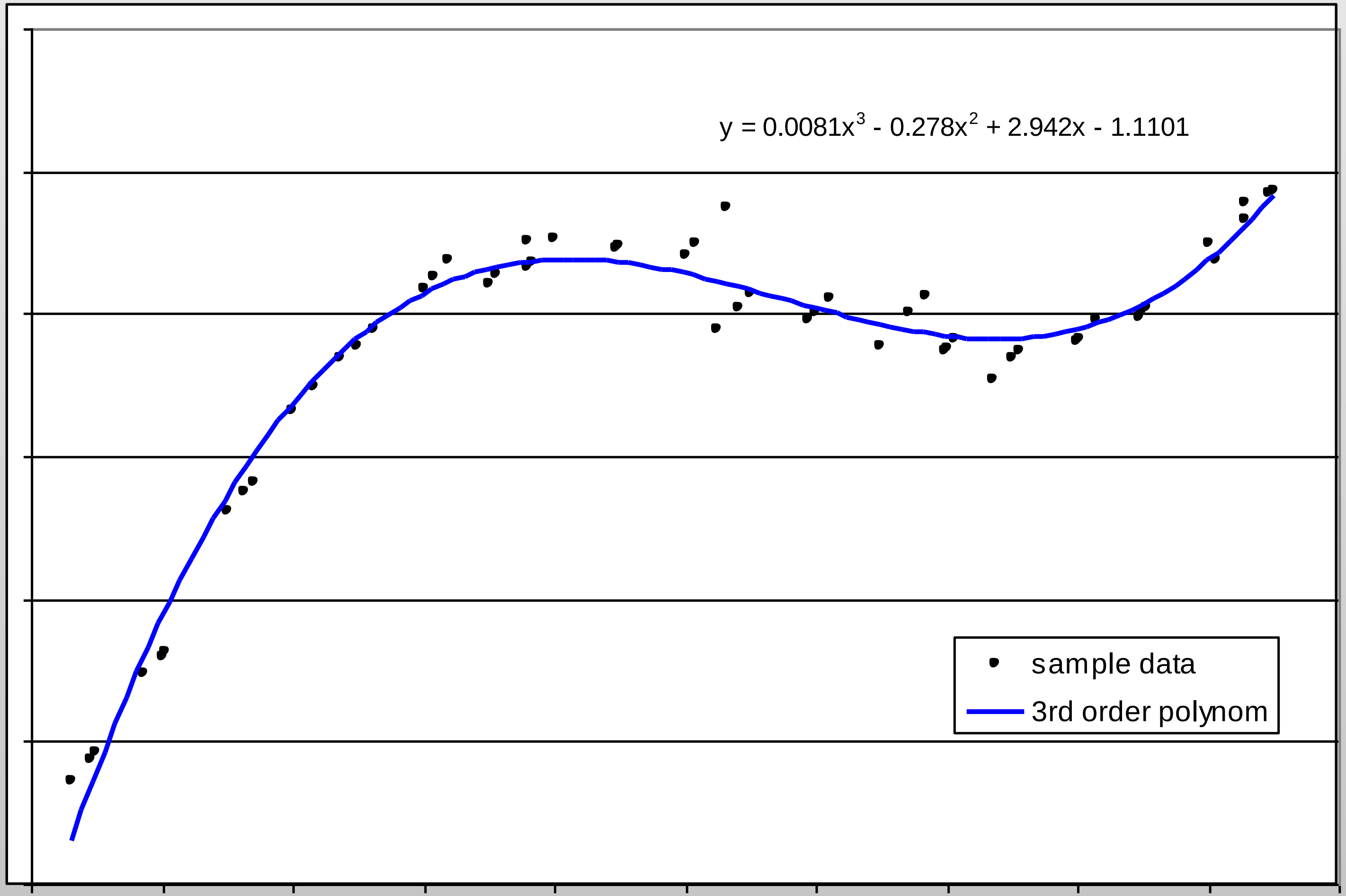
Interpolation algorithms: Trend surfaces

- Linear interpolation



Interpolation algorithms: Trend surfaces

- Linear interpolation

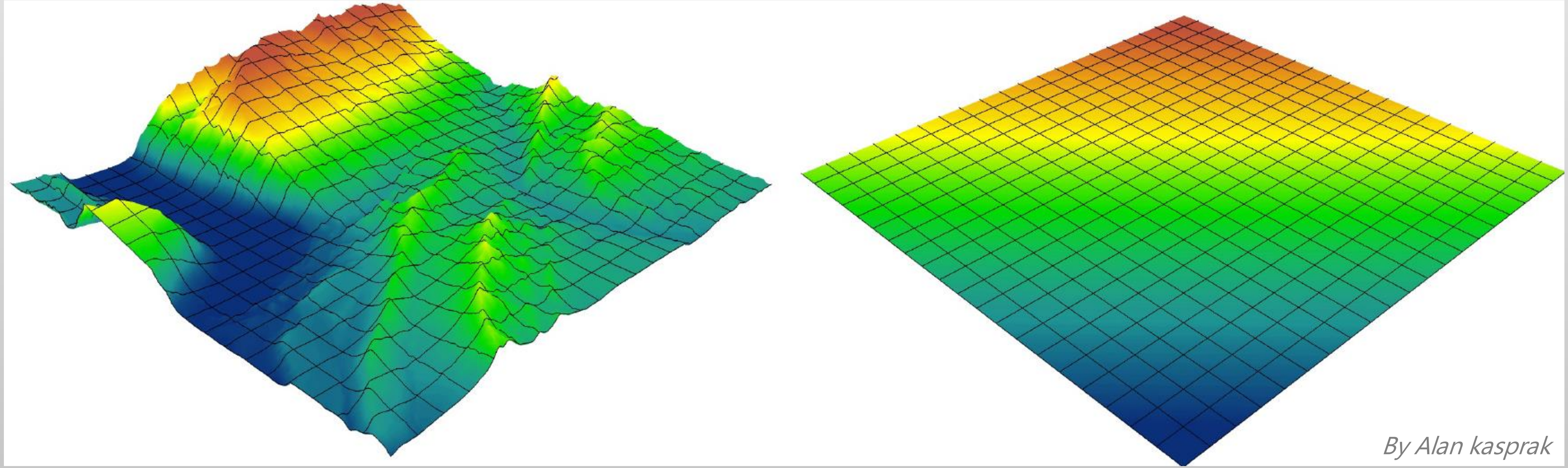


Interpolation algorithms: Trend surfaces

- **Surface interpolation:** Trend spatial interpolation is just regression in two dimensions

Actual landscape

First-order (y = x); some constant value

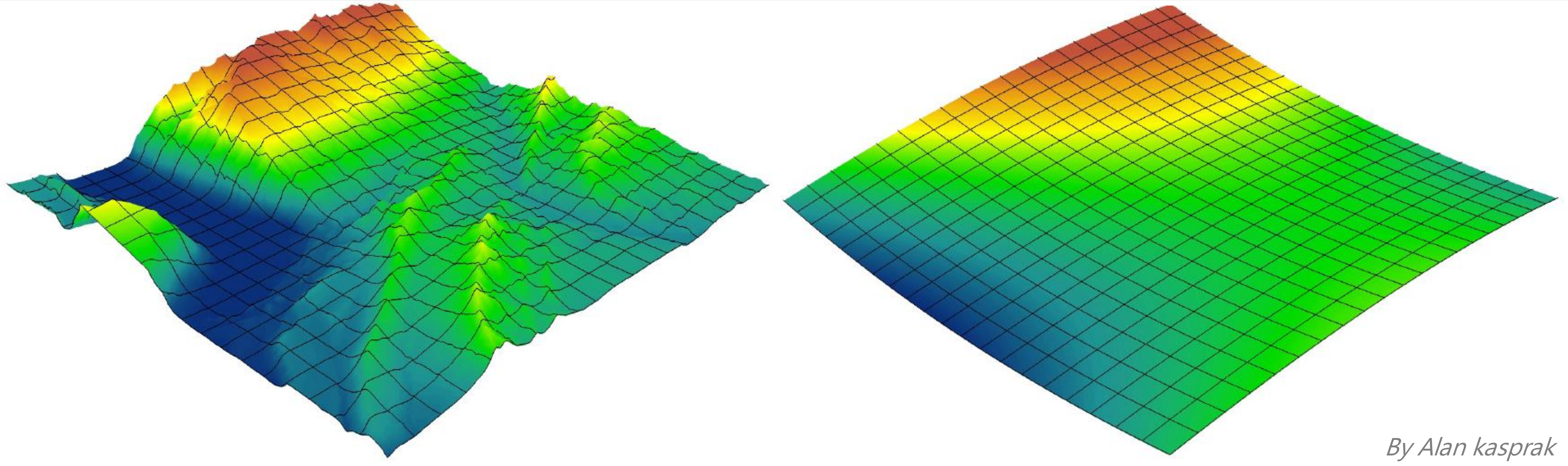


The higher-order polynomial, the more complex terrain you can reproduce (but it takes more time)

Interpolation algorithms: Trend surfaces

- **Surface interpolation:** Trend spatial interpolation is just regression in two dimensions

Actual landscape second-order ($y = x^2 + x$)



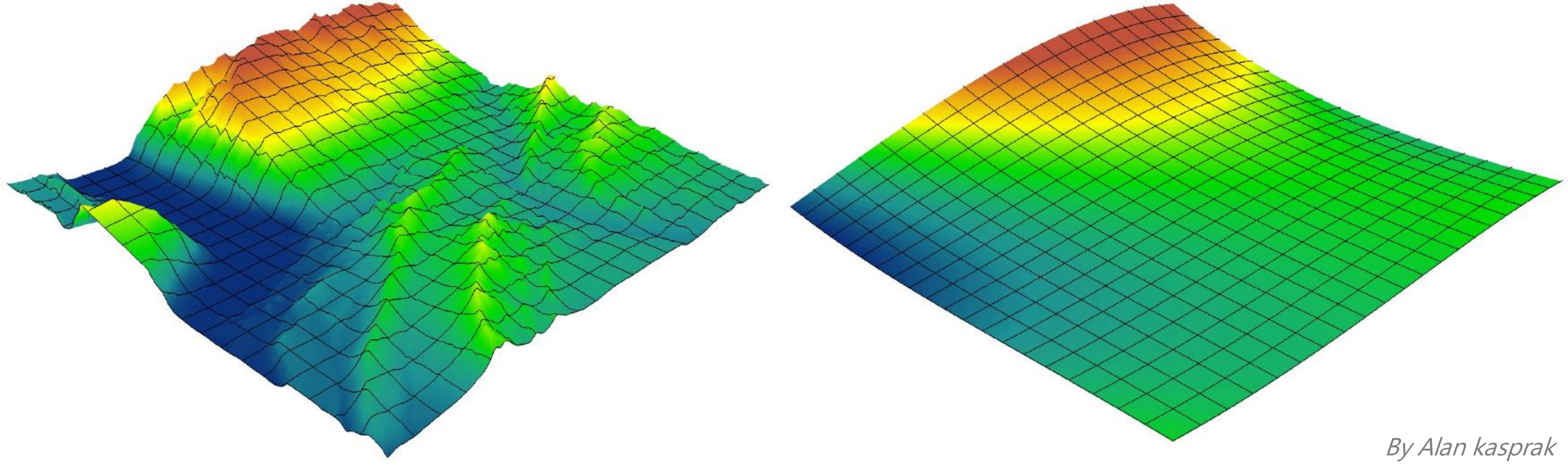
The higher-order polynomial, the more complex terrain you can reproduce (but it takes more time)

Interpolation algorithms: Trend surfaces

- **Surface interpolation:** Trend spatial interpolation is just regression in two dimensions

Actual landscape

third-order ($y = x^3 + x^2 + x$)



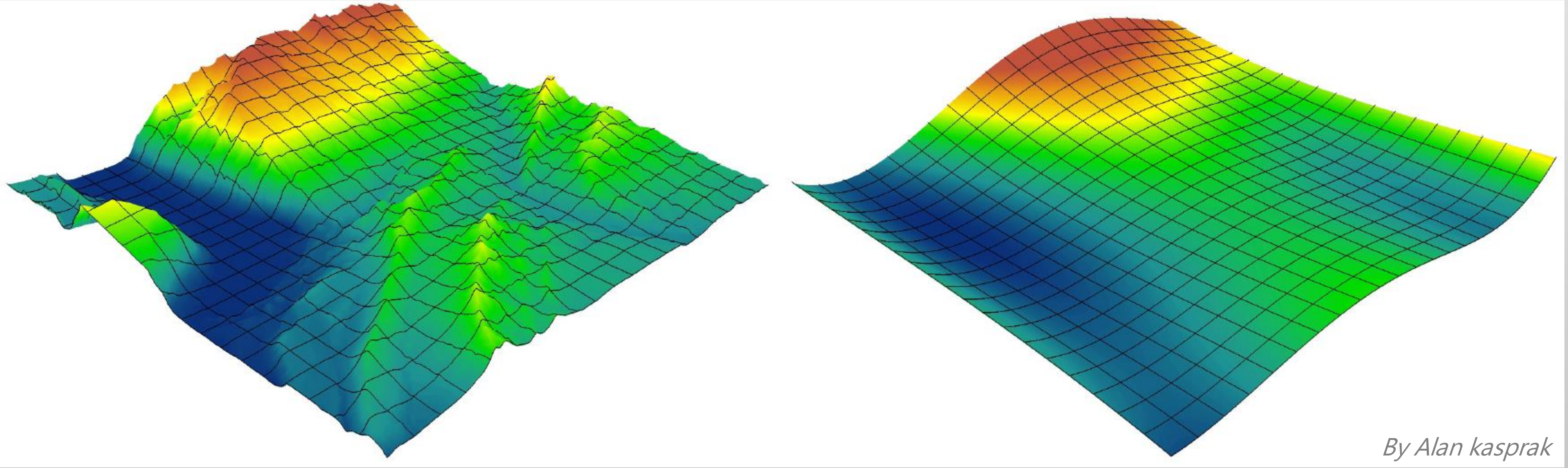
The higher-order polynomial, the more complex terrain you can reproduce (but it takes more time)

Interpolation algorithms: Trend surfaces

- **Surface interpolation:** Trend spatial interpolation is just regression in two dimensions

Actual landscape

fourth-order ($y = x^4 + x^3 + x^2 + x$)



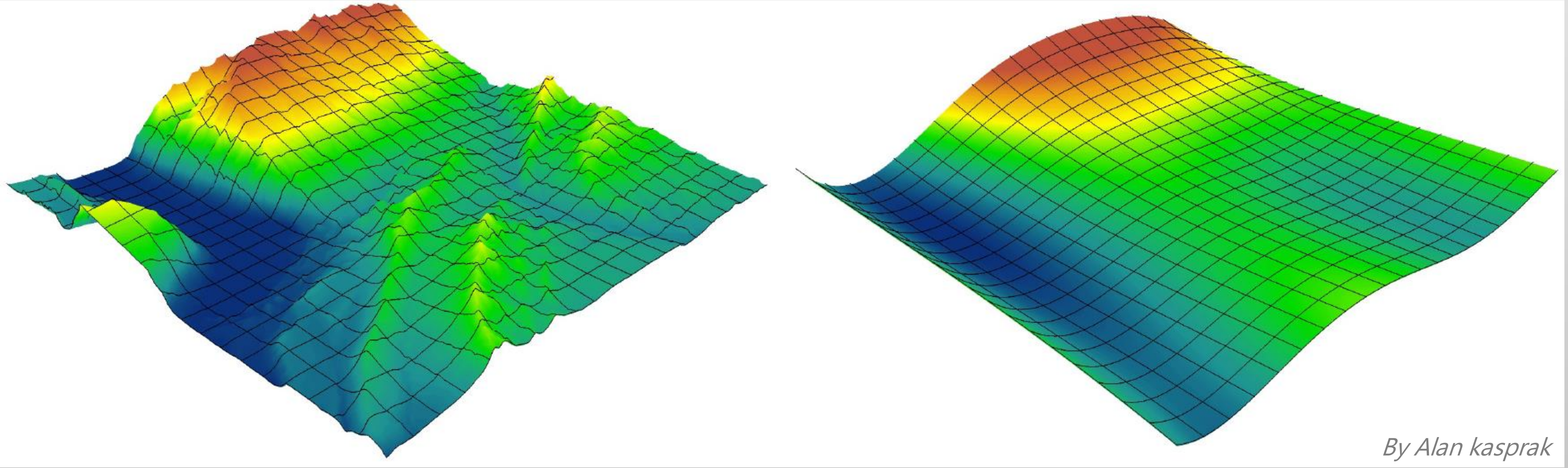
The higher-order polynomial, the more complex terrain you can reproduce (but it takes more time)

Interpolation algorithms: Trend surfaces

- **Surface interpolation:** Trend spatial interpolation is just regression in two dimensions

Actual landscape

fifth-order ($y = x^5 + x^4 + x^3 + x^2 + x$)



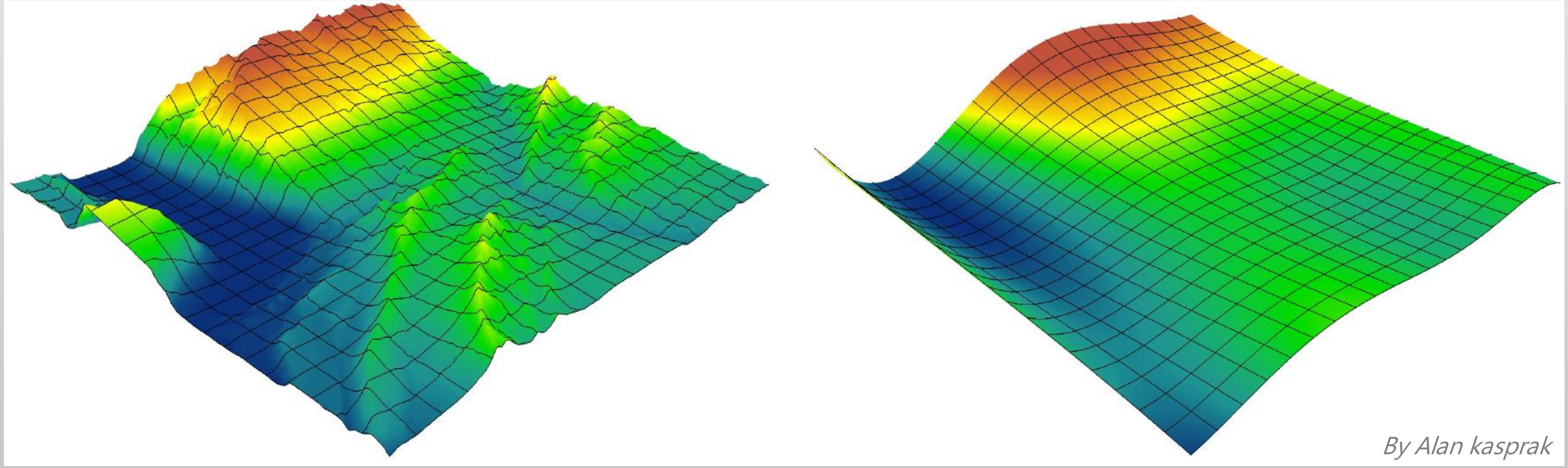
The higher-order polynomial, the more complex terrain you can reproduce (but it takes more time)

Interpolation algorithms: Trend surfaces

- **Surface interpolation:** Trend spatial interpolation is just regression in two dimensions

Actual landscape

sixth-order ($y = x^6 + x^5 + x^4 + x^3 + x^2 + x$)

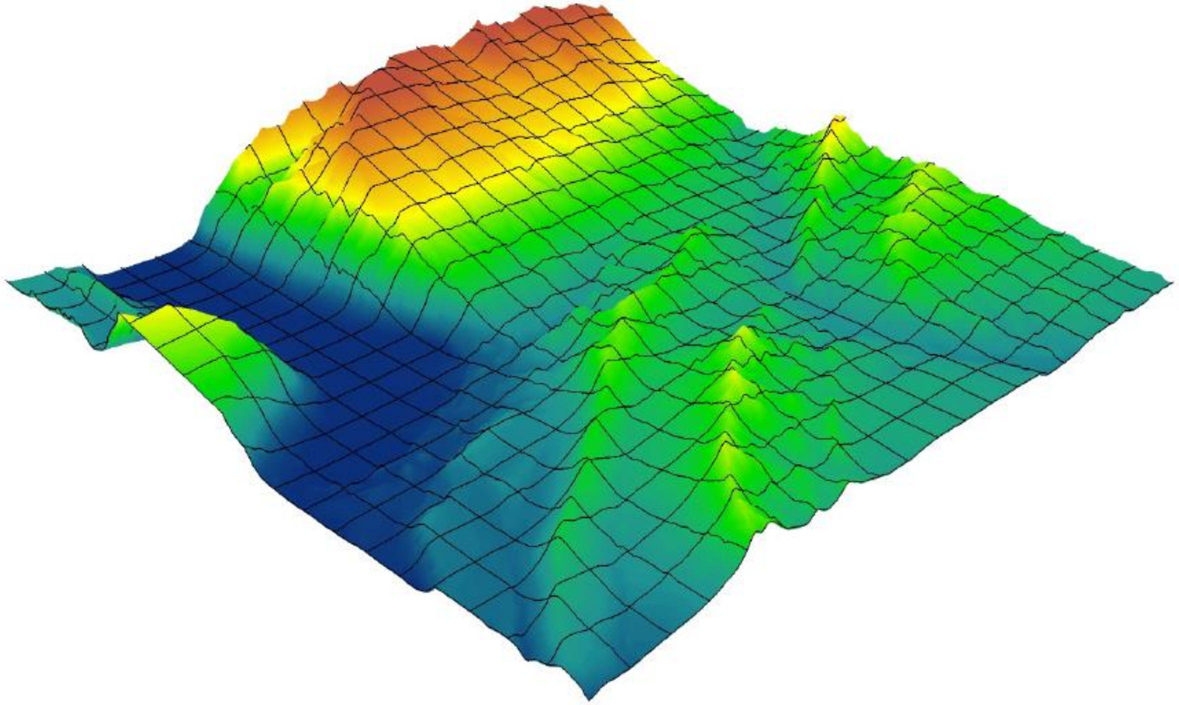


The higher-order polynomial, the more complex terrain you can reproduce (but it takes more time)

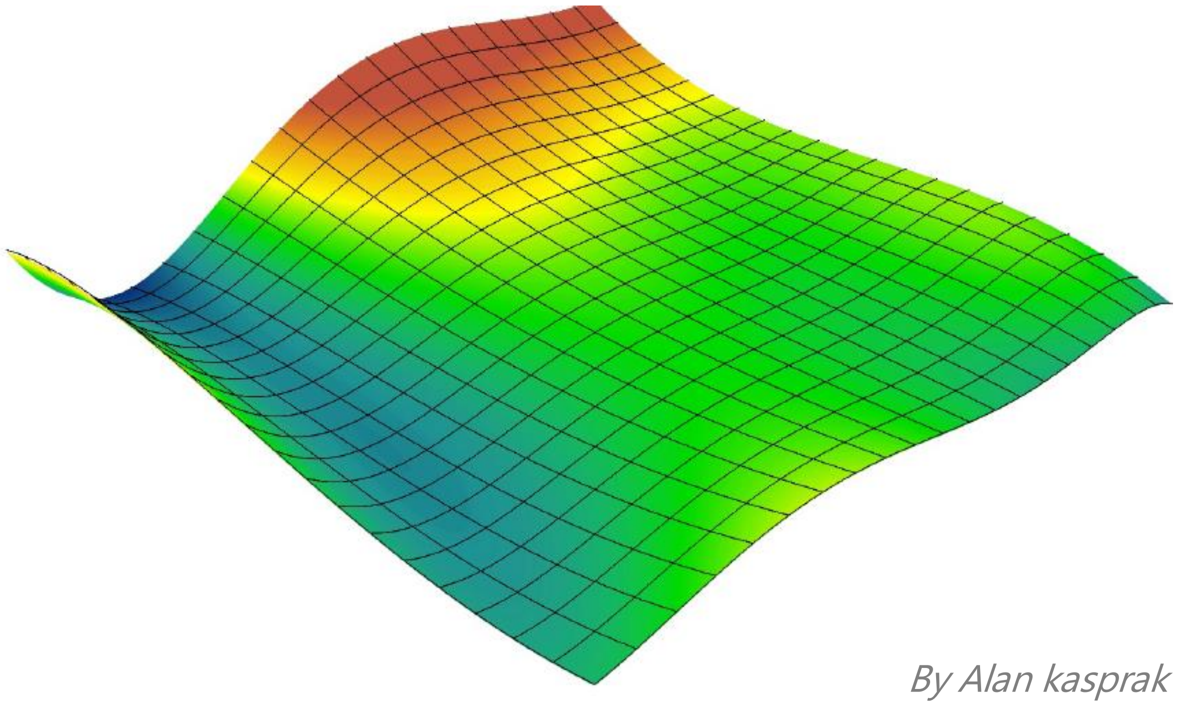
Interpolation algorithms: Trend surfaces

- **Surface interpolation:** Trend spatial interpolation is just regression in two dimensions

Actual landscape



seventh-order ($y = x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x$)

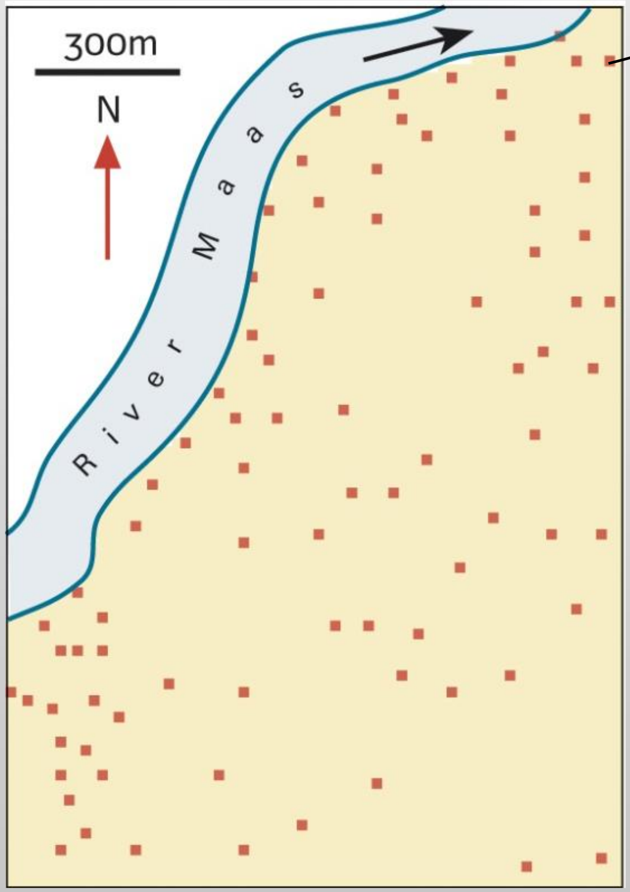


By Alan kasprak

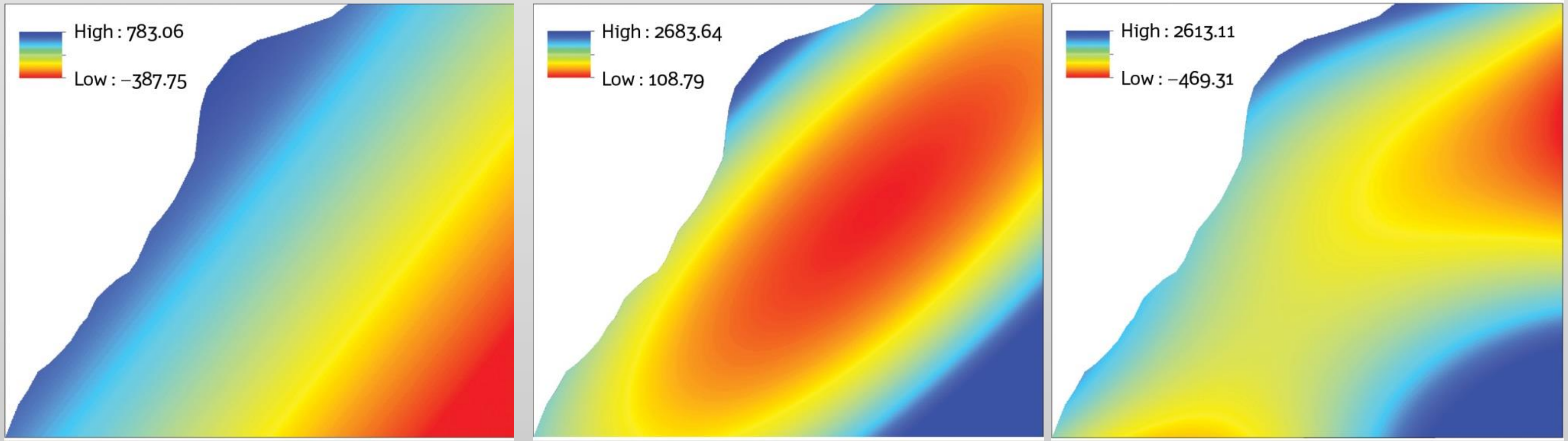
The higher-order polynomial, the more complex terrain you can reproduce (but it takes more time)

Interpolation algorithms: Trend surfaces

Zinc sample data



Trend surface



first order second order third order

Burrough et al. 2015

Interpolation algorithms: Trend surfaces

- Trend interpolation is useful when conditions vary gradually over relatively broad areas (when the surface varies gradually from region to region over the area of interest).
  - Elevation is a bad example
  - Atmospheric conditions (temperature, humidity, pollution, etc.) and aquatic conditions (temperature, pH, salinity) are good examples

- Broad features can be modelled with low-order surfaces
- Examining or removing the effects of global trends
- Difficult to describe a physical meaning to complex higher order polynomials.
  - Higher-order polynomials will always increase the model fit, but at the risk of over-fitting your model and takes more time to interpolate
- Susceptible to edge effects (do not extrapolate!) and to outliers (extremely high and low values)

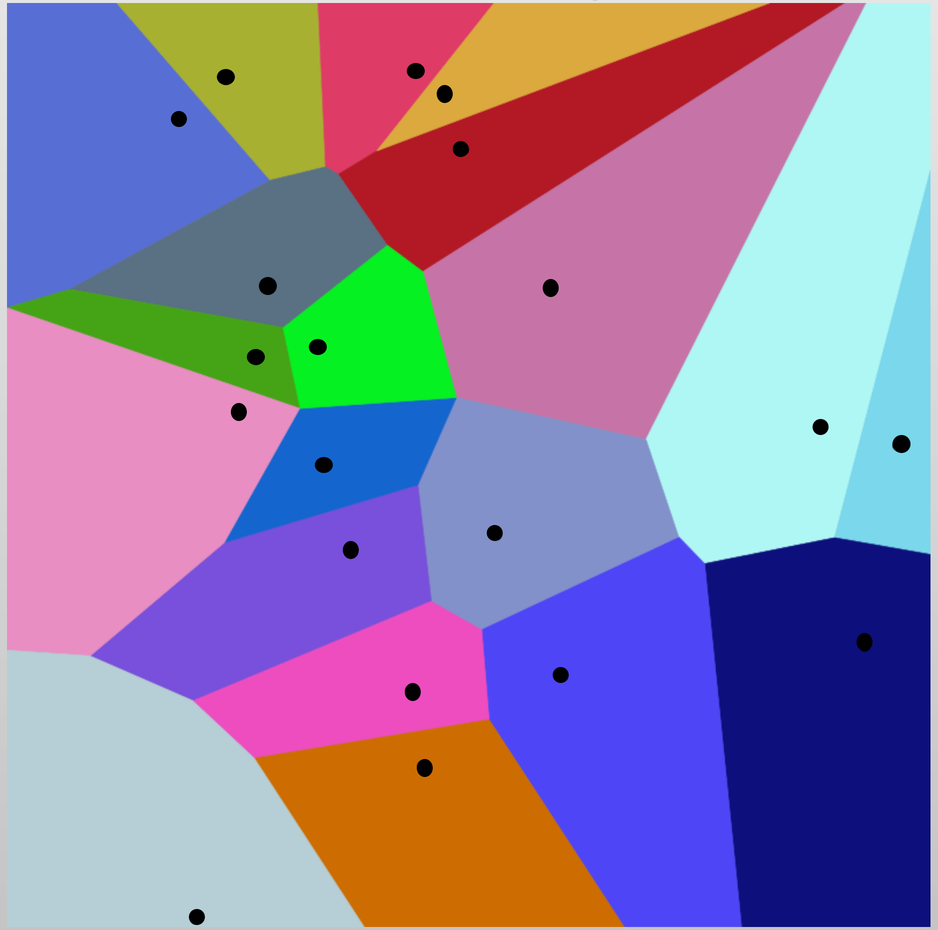
Interpolation algorithms

- **Trend surfaces** (global polynomial interpolation, GPI)
- **Nearest neighbour (Thiessen)**
- **Inverse distance weighting (IDW)** (moving average)
- **Radial basis functions or Spline**
- **Kriging** (ordinary, simple, universal, etc….analyses of spatial variation)

Interpolation algorithms: Nearest neighbour

- **Voronoi diagram (Thiessen polygons)**
  - Voronoi diagram is a partition of a plane into regions close to each of a given set of point $\{p_1, \ldots, p_n\}$
    For each point there is a corresponding region consisting of all points of the plane closer to that point than to any other.
  - The regions *(Voronoi cells)* are known as *Thiessen polygons* (R_k)

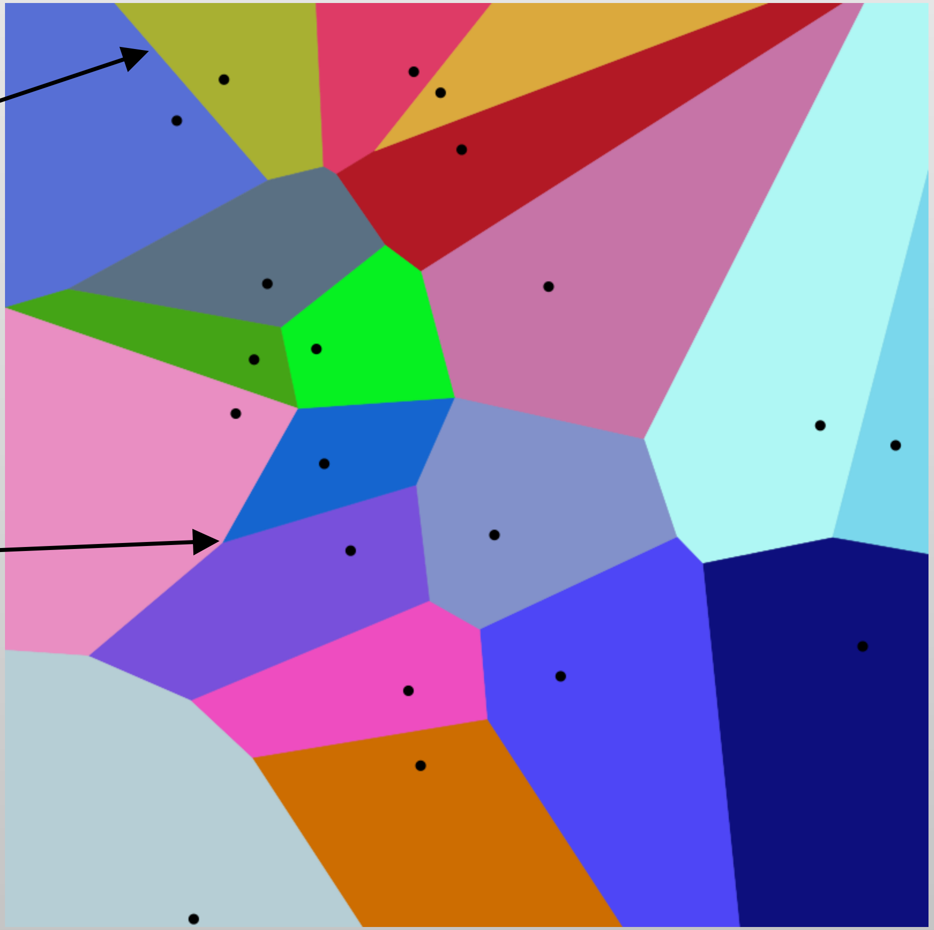
Assign to all unsampled locations the value of the closest sampled location



Black dots are random set of points in 2D

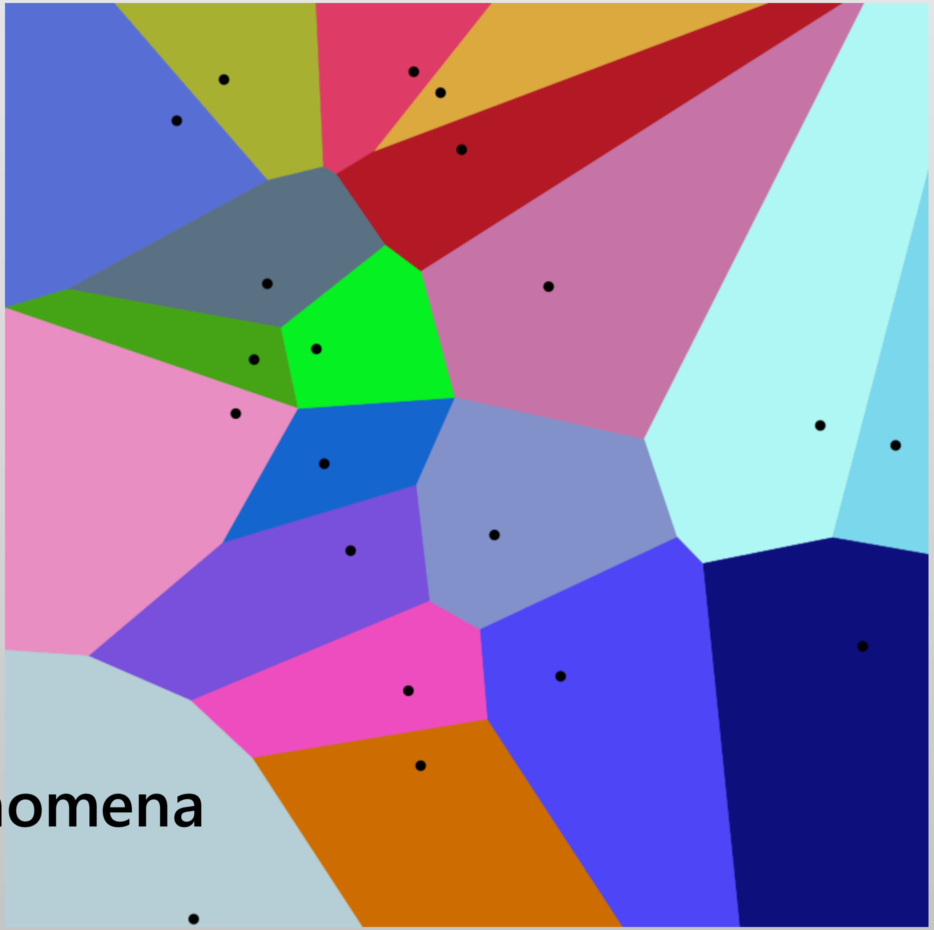
Interpolation algorithms: Nearest neighbour

- Voronoi diagram (Thiessen polygons)
  - The line segments of the Voronoi diagram are all the points in the plane that are equidistant to the two nearest sites.
  - The Voronoi vertices (nodes) are the points equidistant to three (or more) sites.



Interpolation algorithms: Nearest neighbour

- **Voronoi diagram (Thiessen polygons)**
  - Assigns a value to an unsampled location that is equal to the value found at the nearest sample location
    - The predicted value is equal to the value of the nearest sample point
    - Exact interpolator: Value at each sample point is preserved.
  - No variation within polygons
  - **Not appropriate for gradually varying phenomena**
  - **Suitable for qualitative data**



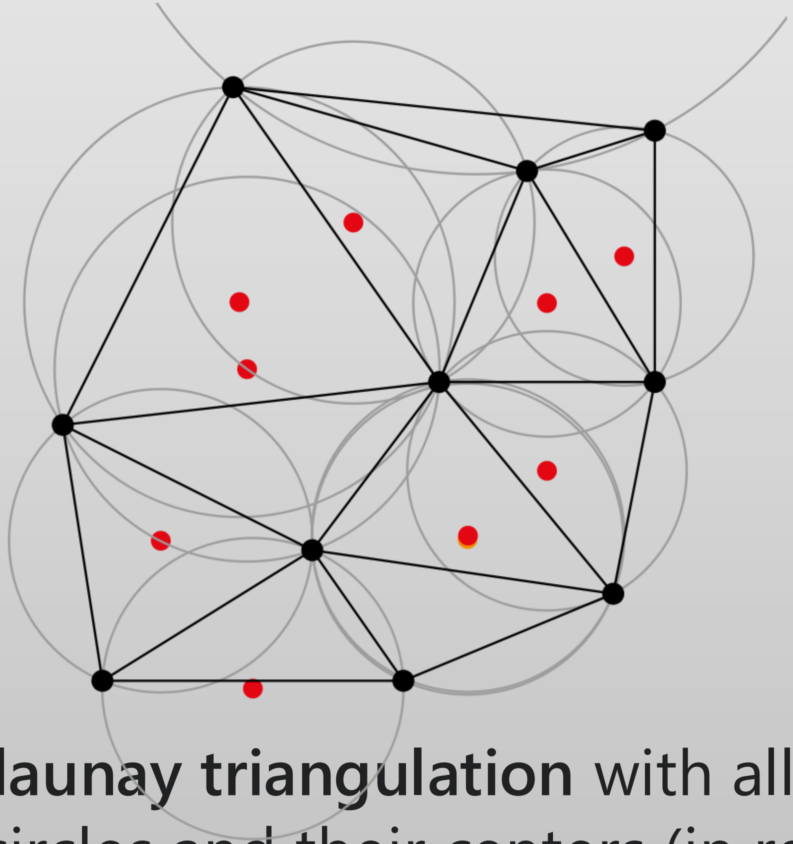
Interpolation algorithms: Nearest neighbour

- **Relationship of the Voronoi diagram with Delaunay triangulation**
  - The vertices of the Voronoi diagram are the circumcenters of Delaunay triangles
  - The triangles satisfies the "Delaunay condition", i.e., that the circumcircles of all triangles have empty interiors

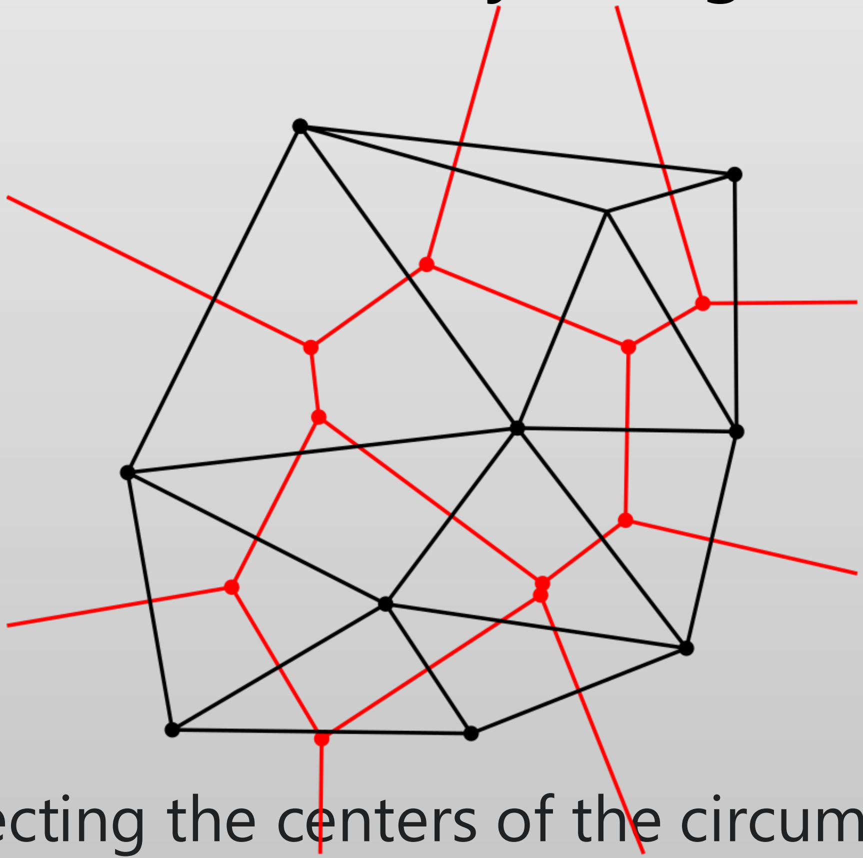
A Delaunay triangulation in the plane with circumcircles shown

Interpolation algorithms: Nearest neighbour

- Relationship of the Voronoi diagram with Delaunay triangulation



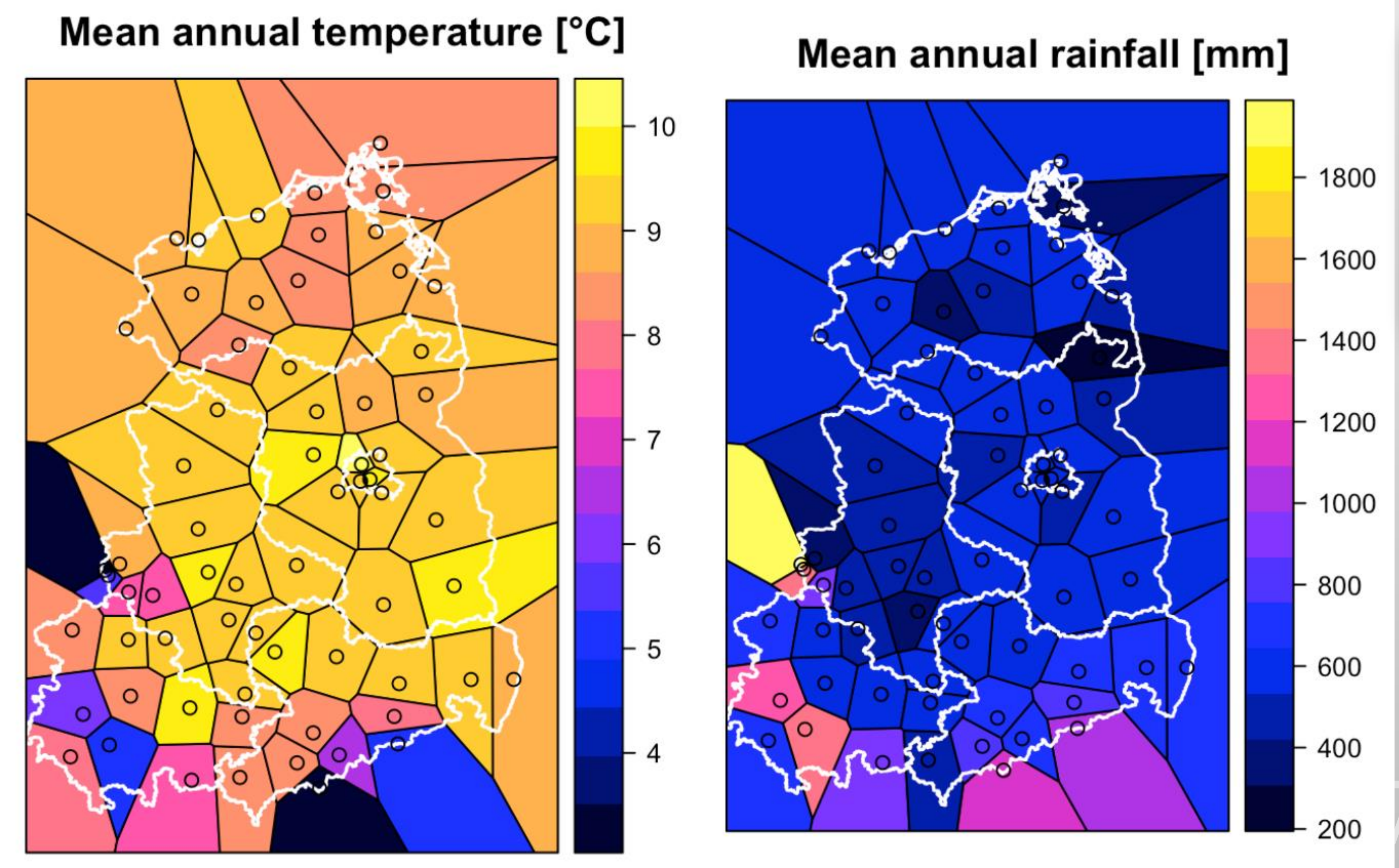
The **Delaunay triangulation** with all the circumcircles and their centers (in red)



Connecting the centers of the circumcircles produces the **Voronoi diagram (in red).**

Interpolation algorithms: Nearest neighbour

Voronoi polygons for the weather station data set in Germany



What is the problematic aspect of this approach ?

The surface values change abruptly across the tessellated boundaries

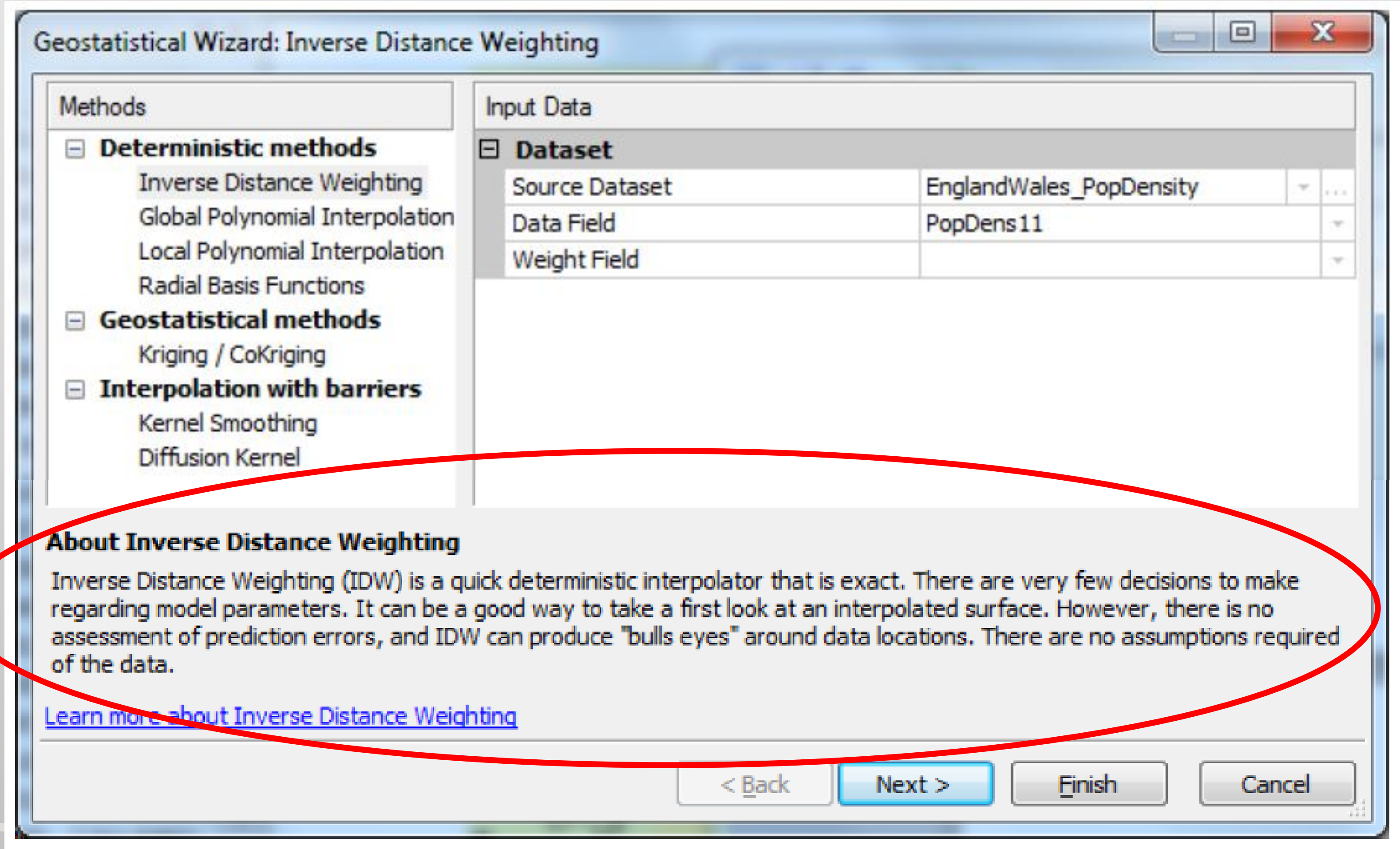
Brundson and Comber, 2015

Interpolation algorithms

- **Trend surfaces** (global polynomial interpolation, GPI)
- **Nearest neighbour (Thiessen)**
- **Inverse distance weighted (IDW)** (moving average)
- **Radial basis functions or Spline**
- **Kriging** (ordinary, simple, universal, etc….analyses of spatial variation)

Interpolation algorithms: IDW

- Inverse distance weighting (IDW) or weighted moving average



Interpolation algorithms: IDW

- Inverse distance weighting (IDW) or weighted moving average
  - The predicted value at any point is the weighted average of the surrounding points

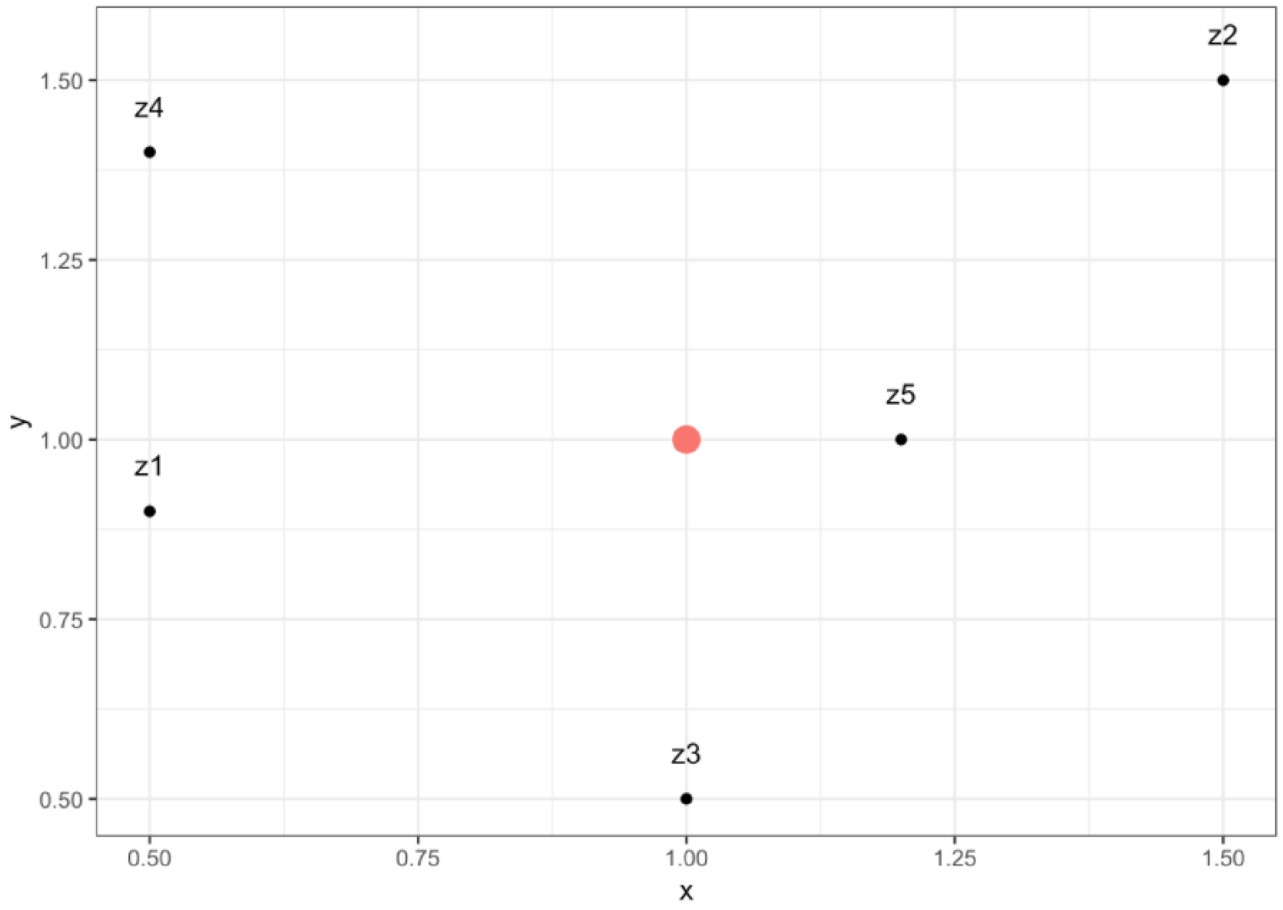
$$\hat{z}(\mathbf{x}) = \frac{\sum_i^n w_i z_i}{\sum_i^n w_i} \quad \text{where} \quad w_i = \left|\mathbf{x} - \mathbf{x}_i\right|^{-\beta}$$

where:
- $\beta \geq 0$
- β is the inverse distance power (or power coefficient)
- $|x - x_i|$ is the euclidean distance
- n is the number of surrounding points to be included

Interpolation algorithms: IDW

- Example: Inverse distance weighting (IDW)

ID	x	y	z
z0	1.0	1.0	?
z1	0.5	0.9	1
z2	1.5	1.5	3
z3	1.0	0.5	5
z4	0.5	1.4	7
z5	1.2	1.0	7



Interpolation algorithms: IDW

- Example: Inverse distance weighting (IDW)

ID	x	y	z
z0	1.0	1.0	?
z1	0.5	0.9	1
z2	1.5	1.5	3
z3	1.0	0.5	5
z4	0.5	1.4	7
z5	1.2	1.0	7

The euclidean distance, d, between two points (u,v) in a plane (R2) is given by

$$d(\mathbf{u},\mathbf{v}) = \sqrt{(u_1 - v_1)^2 + (u_2 - v_2)^2}$$

You can compute w_i as follow. Consider β=2 and n=5

$$\begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_5 \end{bmatrix} = \begin{bmatrix} |\mathbf{z} - \mathbf{z}_1|^{-\beta} \\ |\mathbf{z} - \mathbf{z}_2|^{-\beta} \\ \dots \\ |\mathbf{z} - \mathbf{z}_5|^{-\beta} \end{bmatrix}$$

Interpolation algorithms: IDW

- Example: Inverse distance weighting (IDW)

ID	x	y	z
z0	1.0	1.0	?
z1	0.5	0.9	1
z2	1.5	1.5	3
z3	1.0	0.5	5
z4	0.5	1.4	7
z5	1.2	1.0	7

$$\begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \end{bmatrix} = \begin{bmatrix} |\mathbf{z}-\mathbf{z}_1|^{-\beta} \\ |\mathbf{z}-\mathbf{z}_2|^{-\beta} \\ |\mathbf{z}-\mathbf{z}_3|^{-\beta} \\ |\mathbf{z}-\mathbf{z}_4|^{-\beta} \\ |\mathbf{z}-\mathbf{z}_5|^{-\beta} \end{bmatrix} = \begin{bmatrix} \sqrt{(1-0.5)^2+(1-0.9)^2}^{\,-2} \\ \sqrt{(1-1.5)^2+(1-1.5)^2}^{\,-2} \\ \sqrt{(1-1)^2+(1-0.5)^2}^{\,-2} \\ \sqrt{(1-0.5)^2+(1-1.4)^2}^{\,-2} \\ \sqrt{(1-1.2)^2+(1-1)^2}^{\,-2} \end{bmatrix} \approx \begin{bmatrix} 3.846 \\ 2 \\ 4 \\ 2.439 \\ 25 \end{bmatrix}$$

Hence:

$$\sum_i^n w_i \approx 37.285 \qquad w_i z_i = \begin{bmatrix} 3.846 \\ 2 \\ 4 \\ 2.439 \\ 25 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \\ 5 \\ 7 \\ 7 \end{bmatrix} \qquad \sum_i^n w_i z_i = 221.919$$

Interpolation algorithms: IDW

- Example: Inverse distance weighting (IDW)

ID	x	y	z
z0	1.0	1.0	?
z1	0.5	0.9	1
z2	1.5	1.5	3
z3	1.0	0.5	5
z4	0.5	1.4	7
z5	1.2	1.0	7

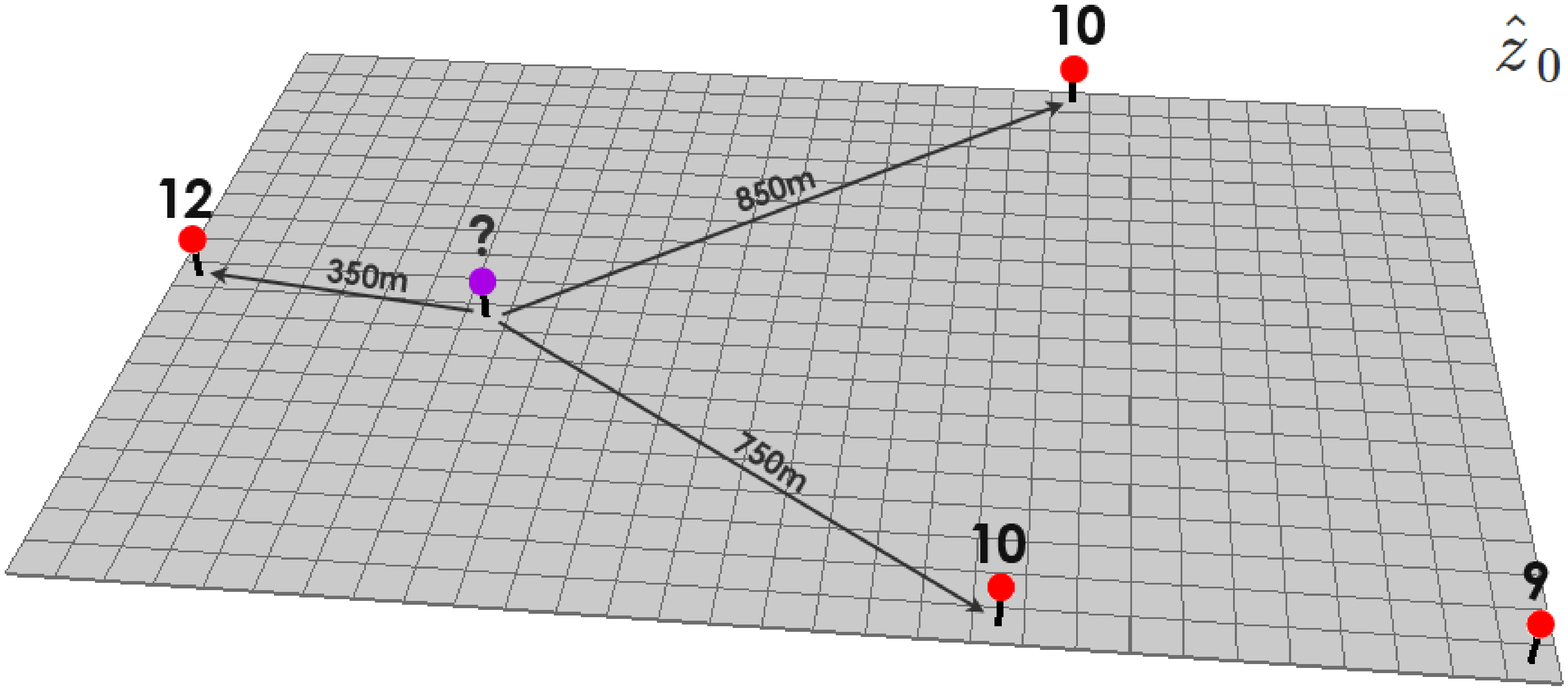
Plugging the numerator and the denominator into the equation from above yields

$$\hat{z}_0 = \frac{\sum_i^n w_i z_i}{\sum_i^n w_i} \approx \frac{221.919}{37.285} = 5.952$$

ID	x	y	z
z0	1.0	1.0	5.952

Interpolation algorithms: IDW

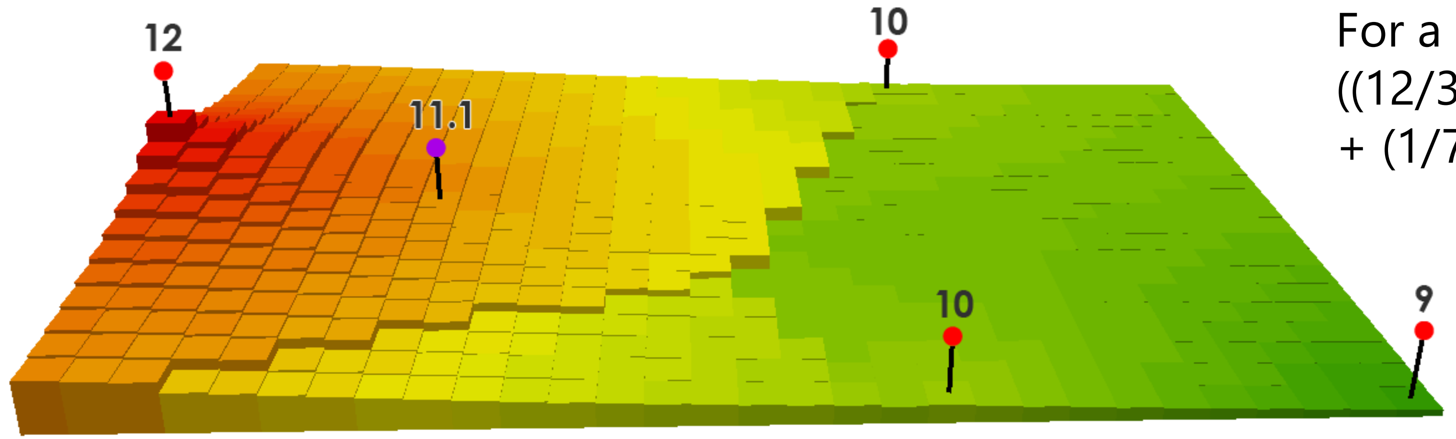
- Example: Inverse distance weighting (IDW)



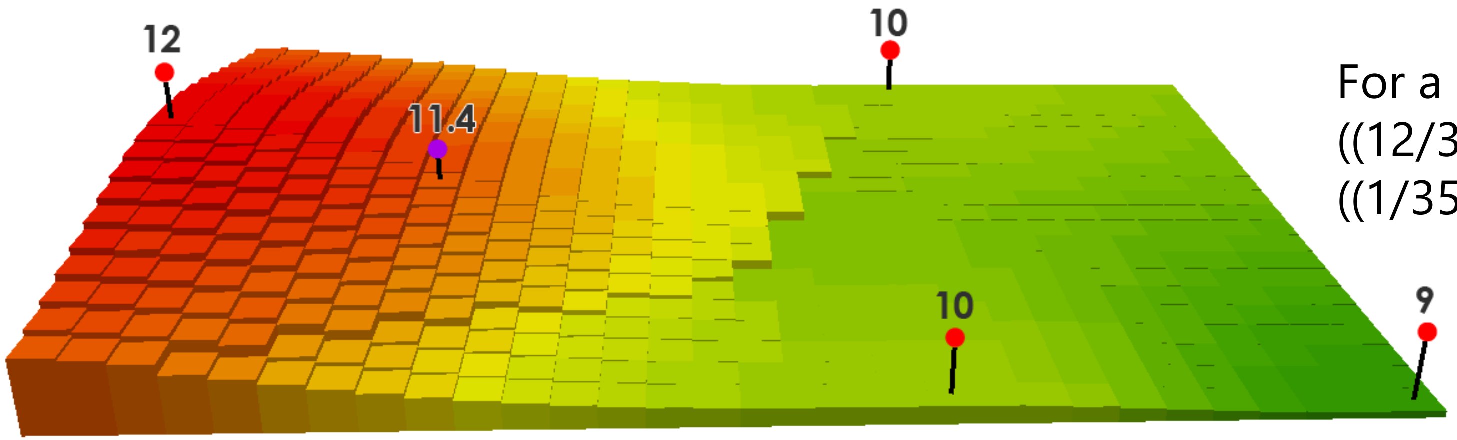
$$\hat{z}_0 = \frac{\sum_i^n w_i z_i}{\sum_i^n w_i}$$

GISGeography.com

Interpolation algorithms: IDW



For a **power of 1**, that cell value is equal to: ((12/350) + (10/750) + (10/850)) / ((1/350) + (1/750) + (1/850)) = 11.1

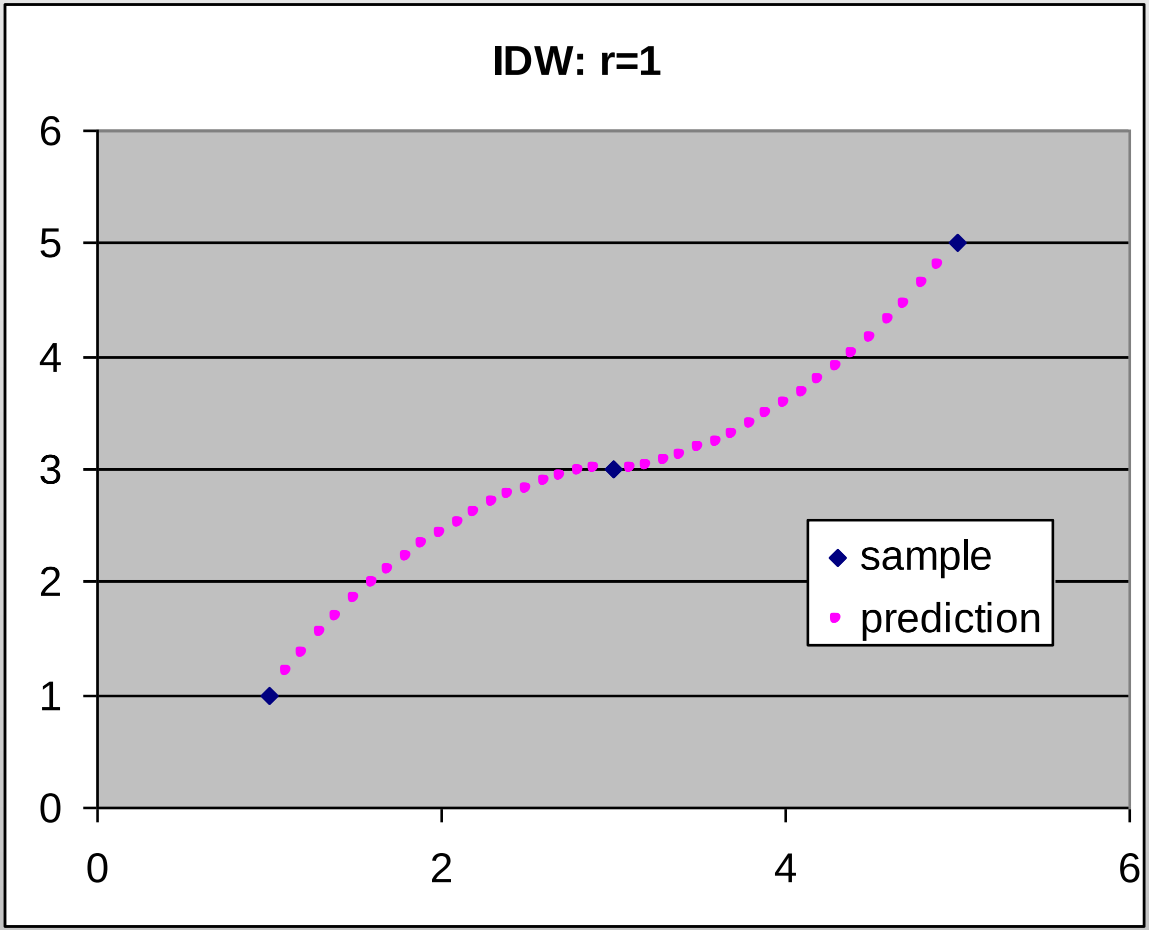


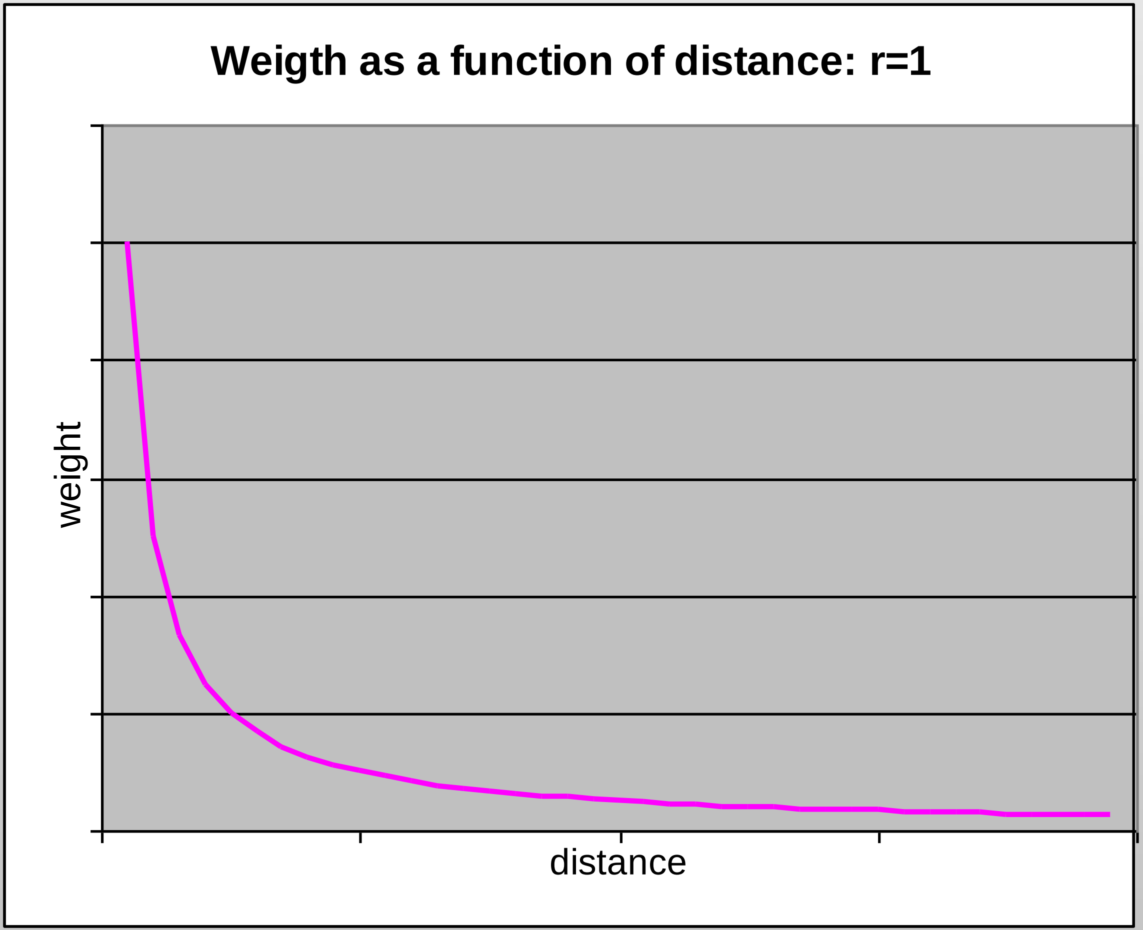
For a **power of 2**, that cell value is equal to: $((12/350^2) + (10/750^2) + (10/850^2))$ / $((1/350^2) + (1/750^2) + (1/850^2))$ = 11.4

GISGeography.com

Interpolation algorithms: IDW

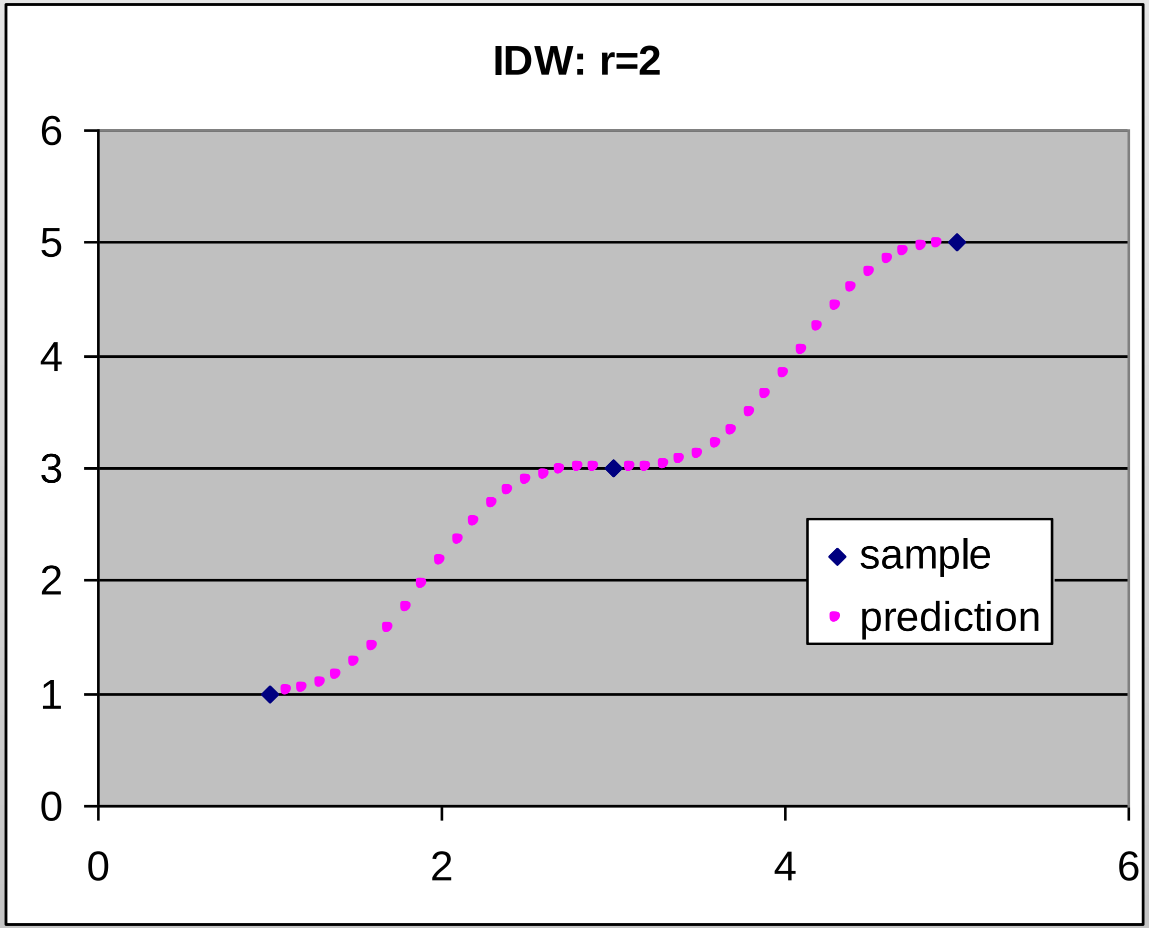
- The r value controls the level of relative spatial dependency (low r value)

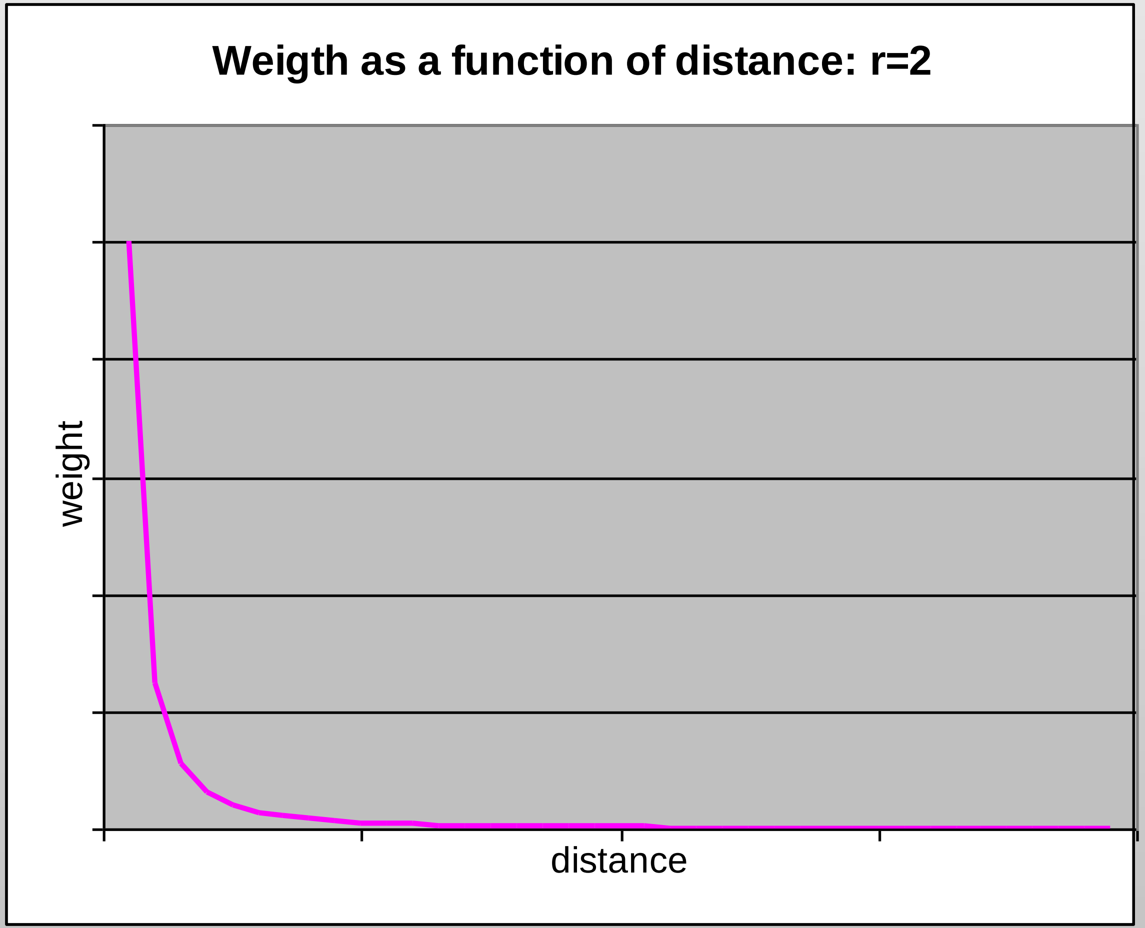




Interpolation algorithms: IDW

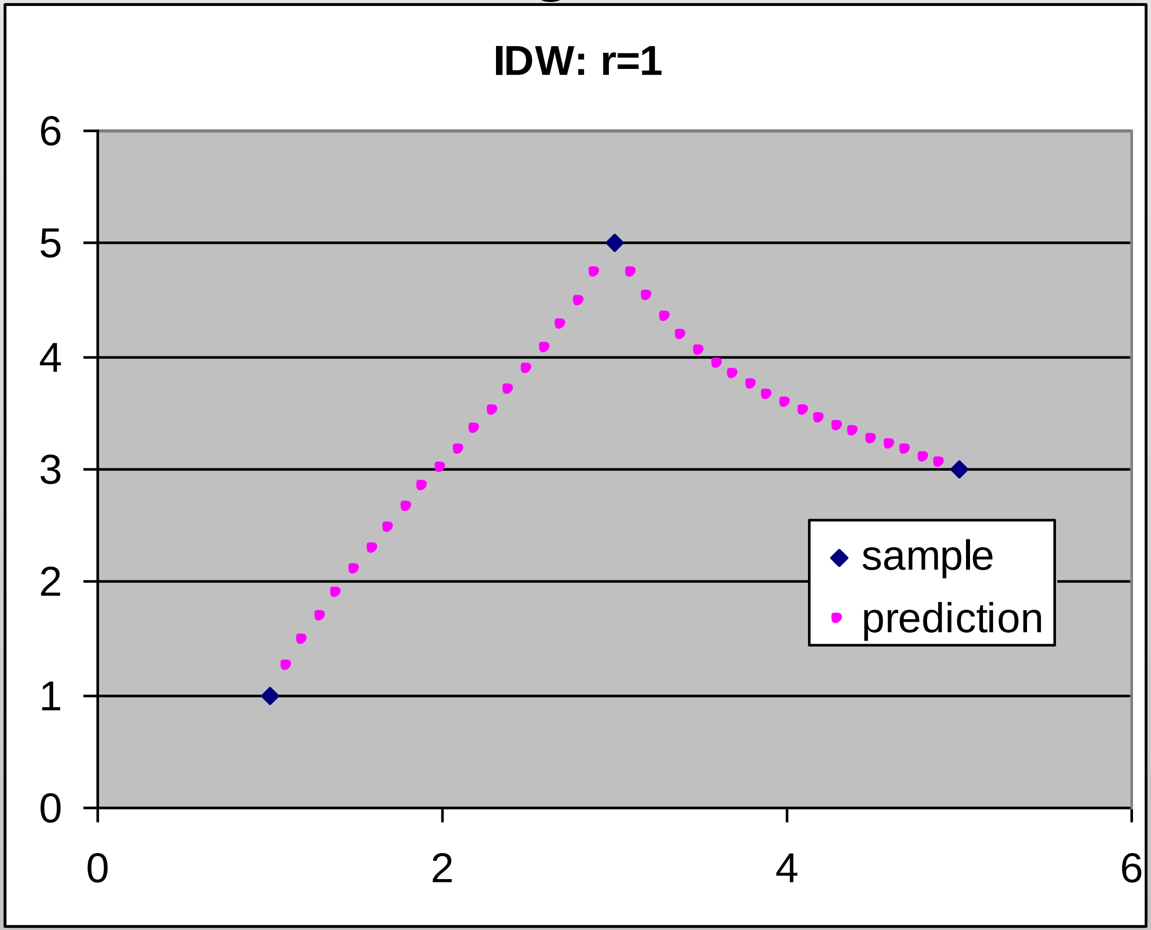
- The r value controls the level of relative spatial dependency (low r value)

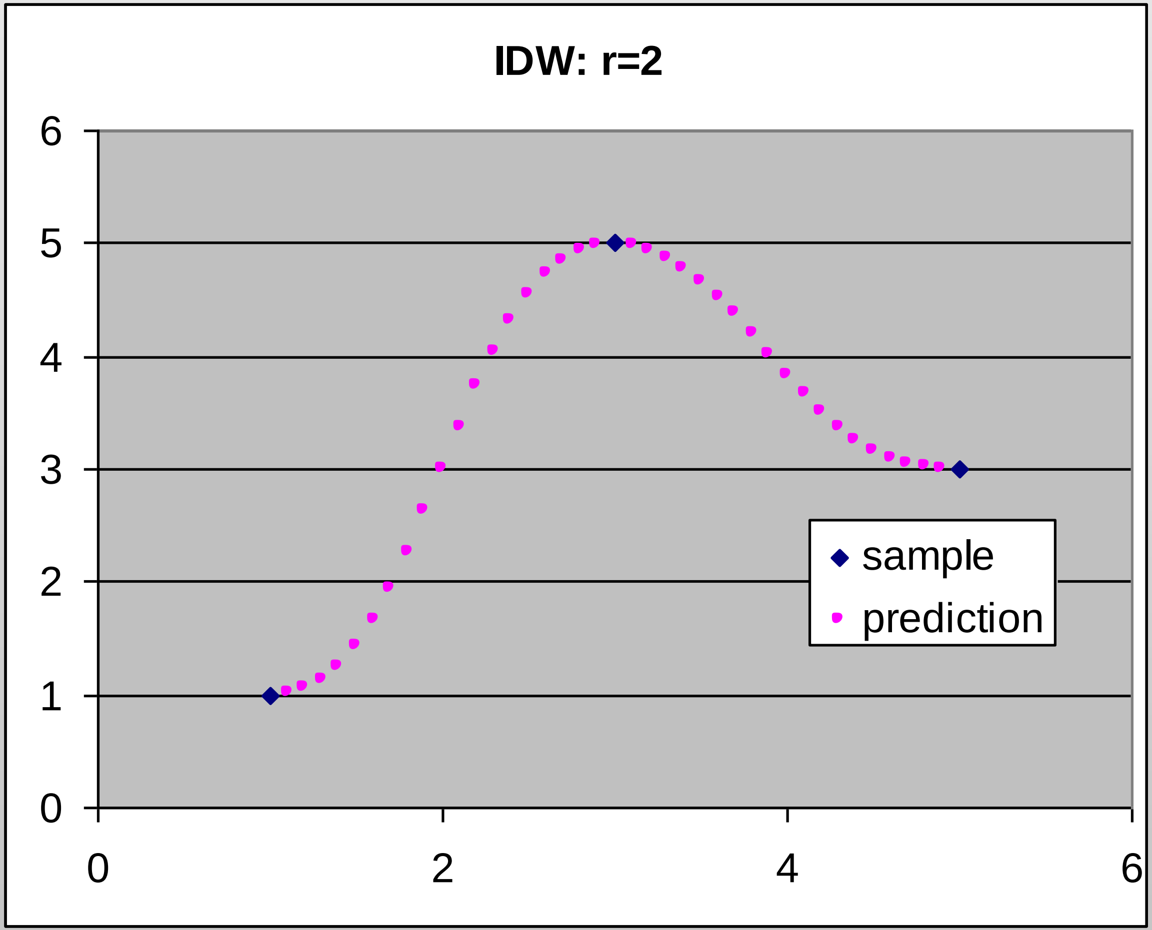




Interpolation algorithms: IDW

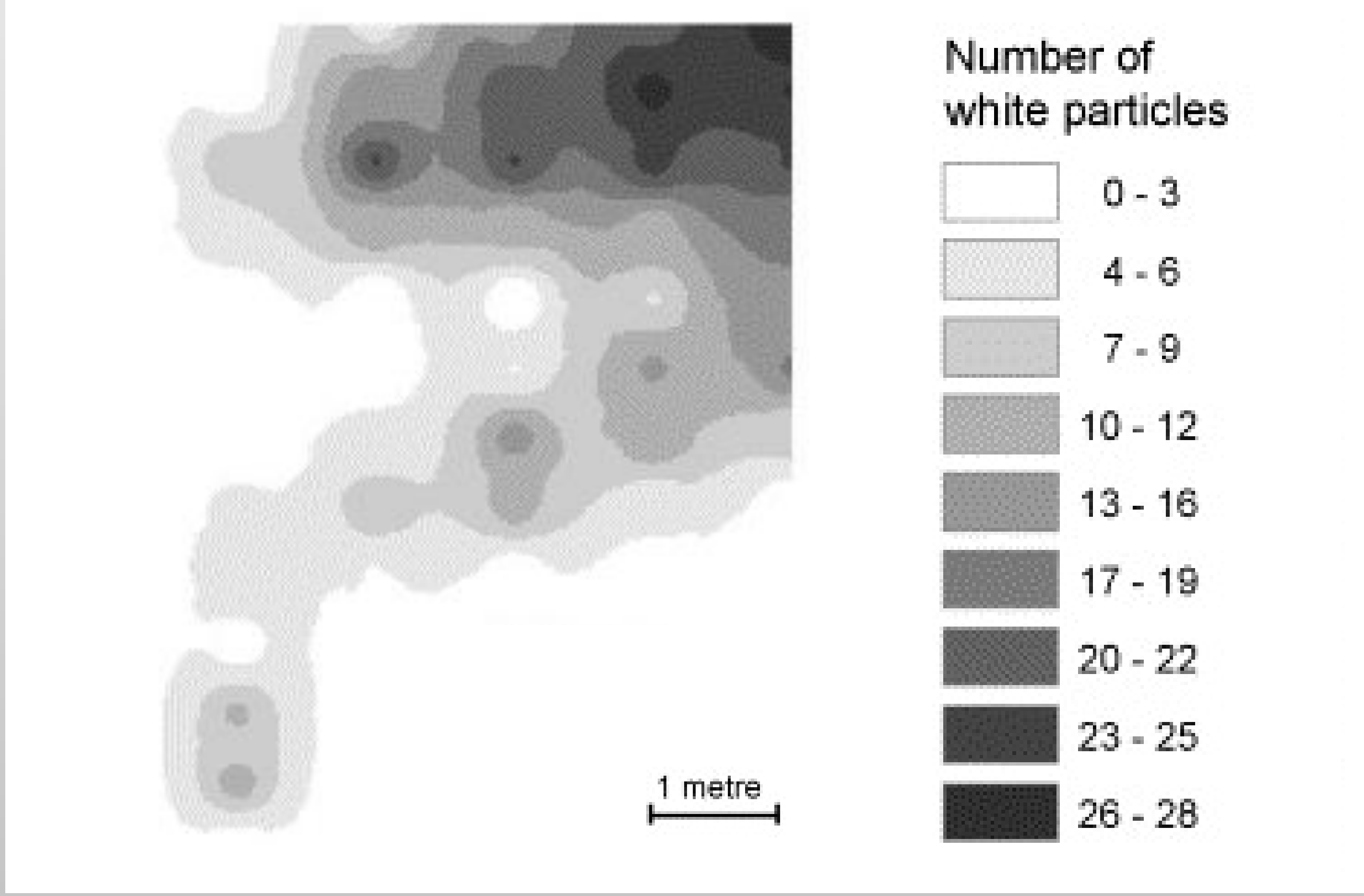
- “Duck-egg” pattern appear around solitary data points that differ greatly from their surroundings





Interpolation algorithms: IDW

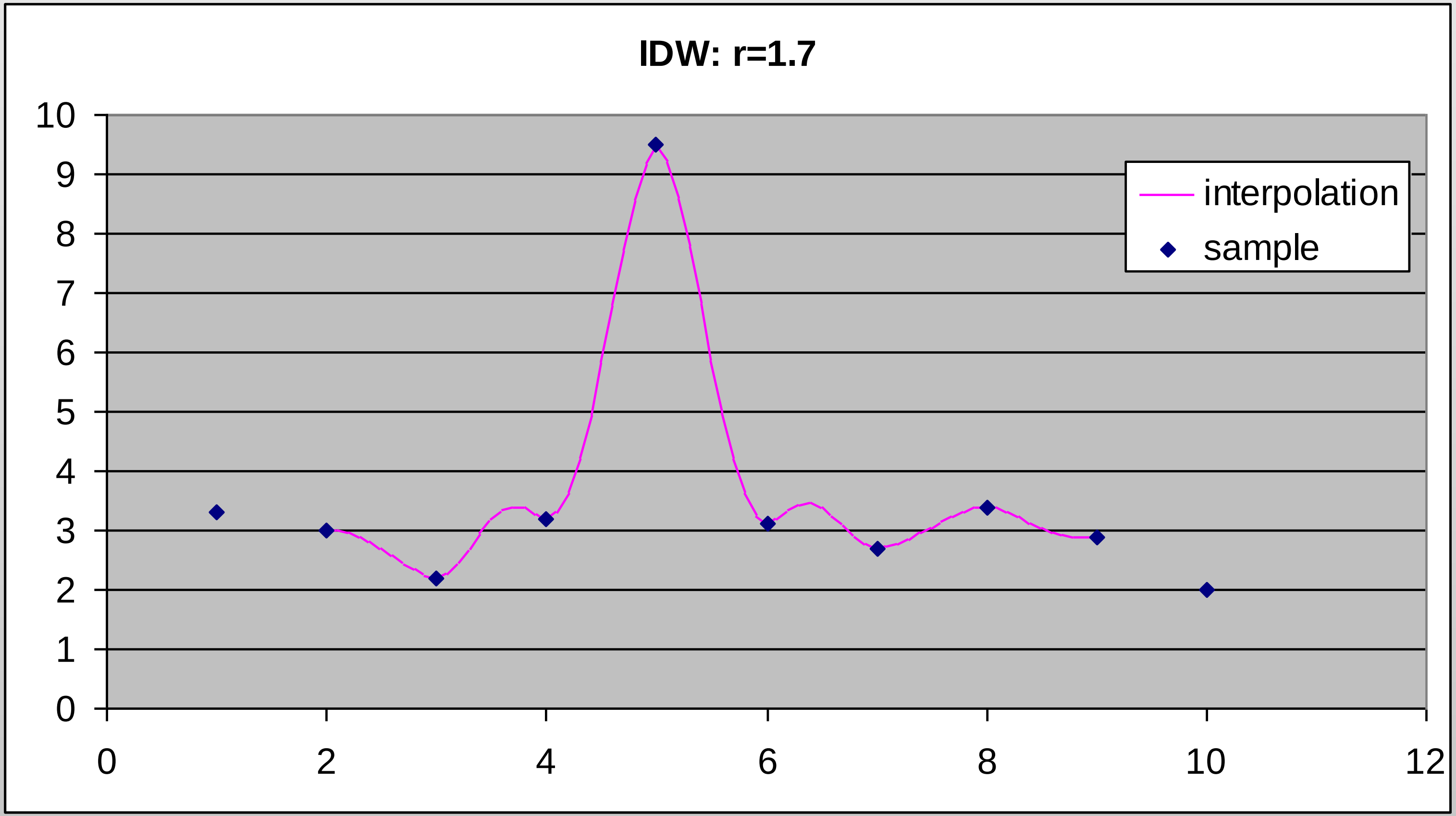
- “Duck-egg” pattern



(McKinley and Ruffell 2006)

Interpolation algorithms: IDW

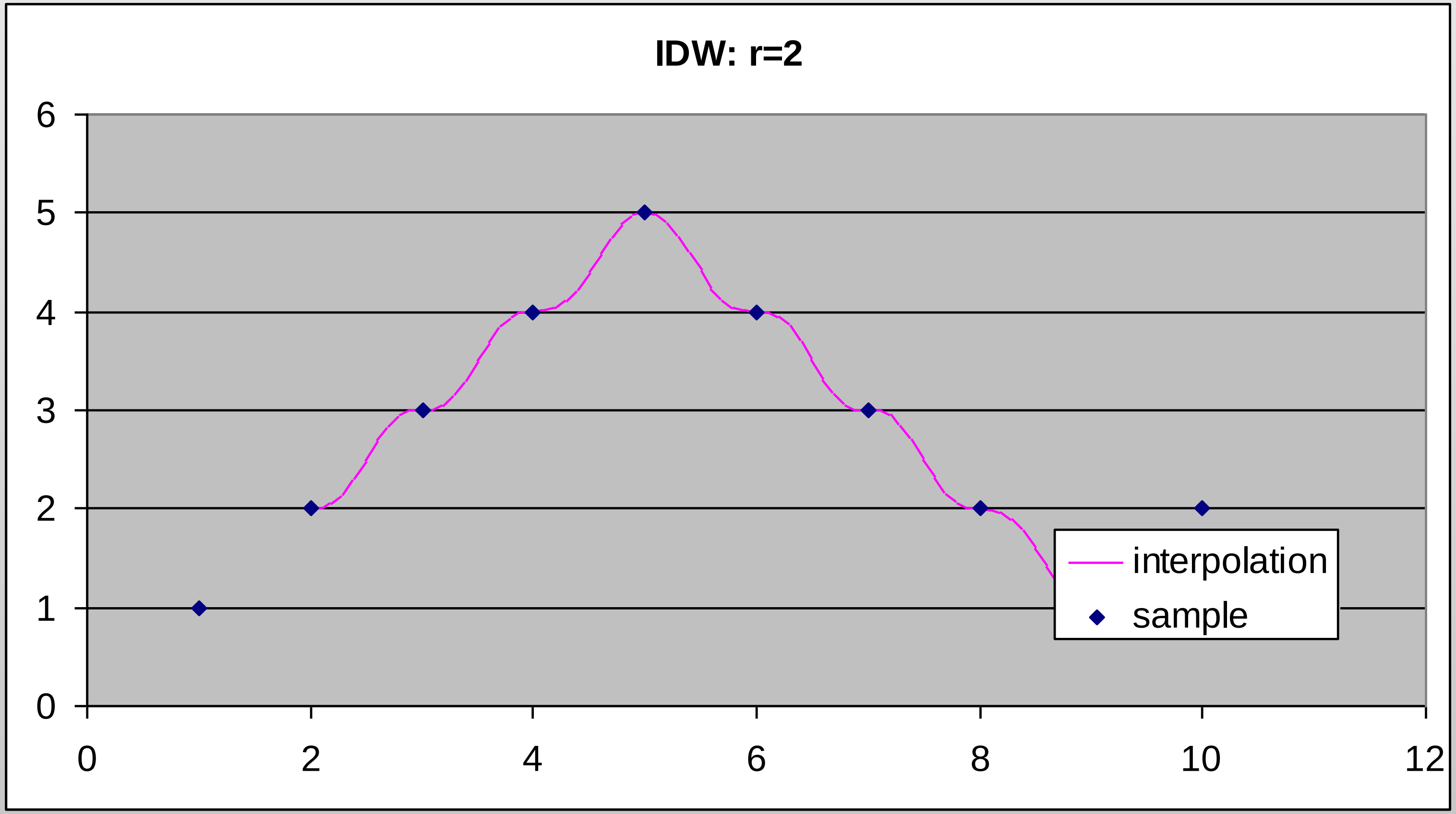
- Outlier effects



(McKinley and Ruffell 2006)

Interpolation algorithms: IDW

- Terracing effects may also occur (especially for high r values)



(McKinley and Ruffell 2006)

Interpolation algorithms

- **Trend surfaces** (global polynomial interpolation, GPI)
- **Nearest neighbour (Thiessen)**
- **Inverse distance weighting (IDW)** (moving average)
- **Radial basis functions or spline**
- **Kriging** (ordinary, simple, universal, etc….analyses of spatial variation)

Interpolation algor.: Radial basis functions

- Radial basis functions are a series of exact interpolation techniques i.e. the surface must pass through each measured sample value.

- There are five different basis functions:
  - Thin-plate spline
  - Spline with tension
  - Completely regularized spline
  - Multiquadric function
  - Inverse multiquadric function

- Radial basis functions methods are a special case of splines.

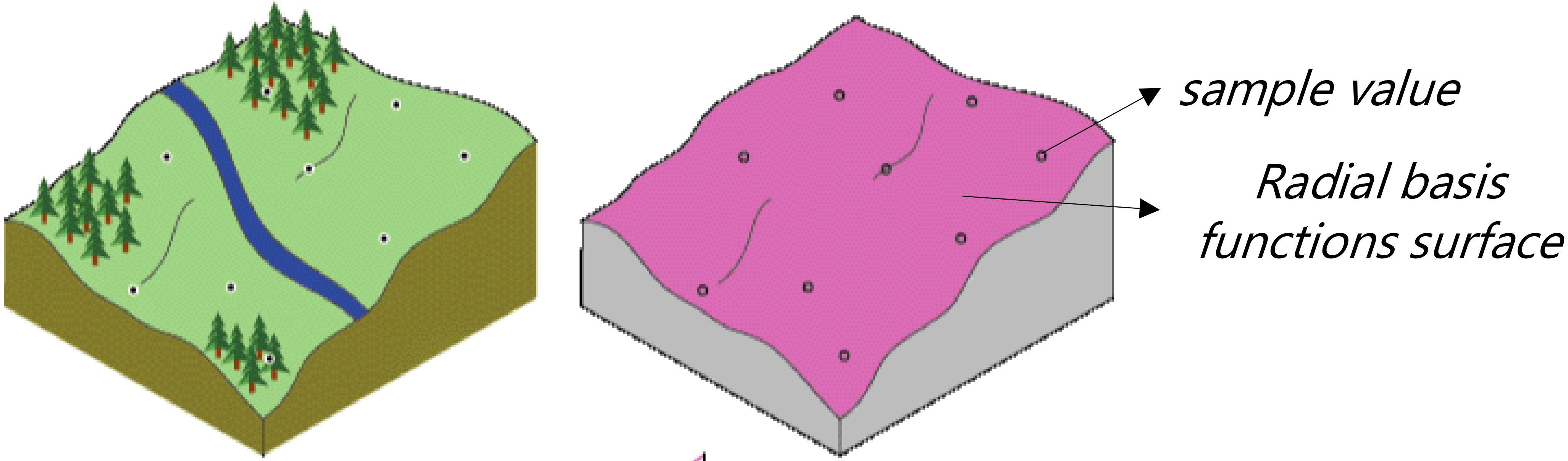
Interpolation algor.: Radial basis functions

Splines

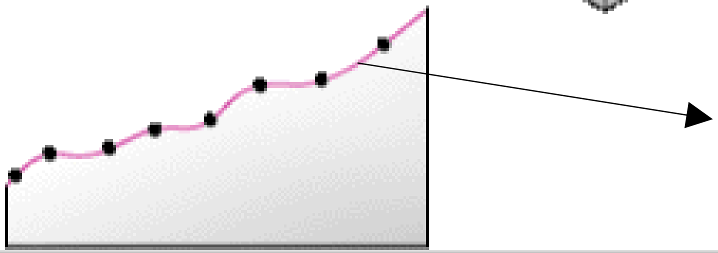
- Piece-wise functions
  - Fitted exactly to a small number of data points
  - Joins between one part of the curve and another are continuous
- Bicubic spline
  - Three dimensional (surface)
  - The resulting smooth surface passes exactly through the input points.
- Thin-plate spline
  - Exact spline surface is replaced by a locally smoothed average

Interpolation algor.: Radial basis functions

- Radial basis functions are a series of exact interpolation techniques i.e. the surface must pass through each measured sample value.

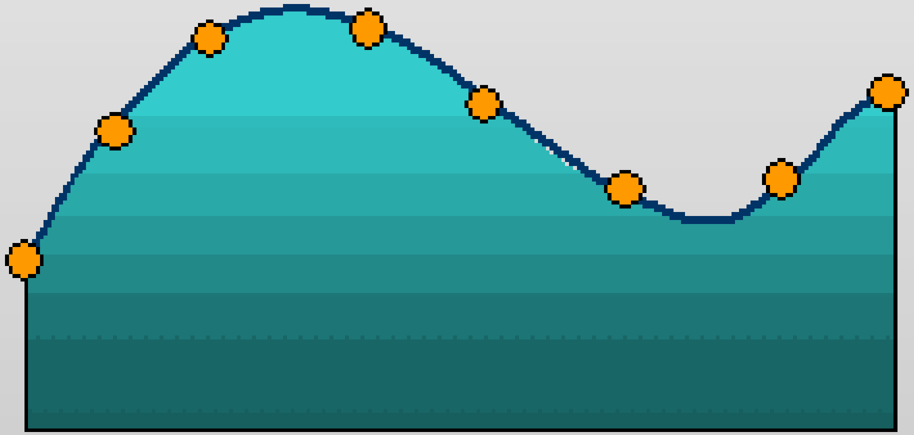


Cross section illustrating how RBF surface fits through the sample values

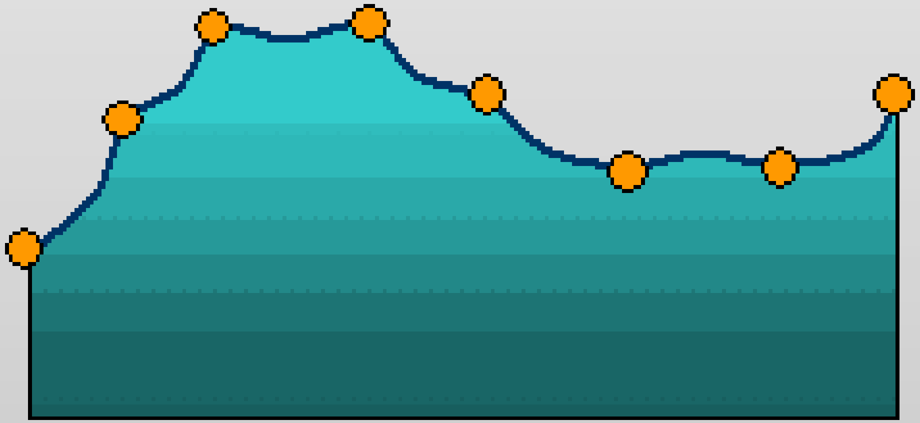


Interpolation algor.: Radial basis functions

- Difference between Radial basis functions (RBF) and Inverse Distance Weighted (IDW)



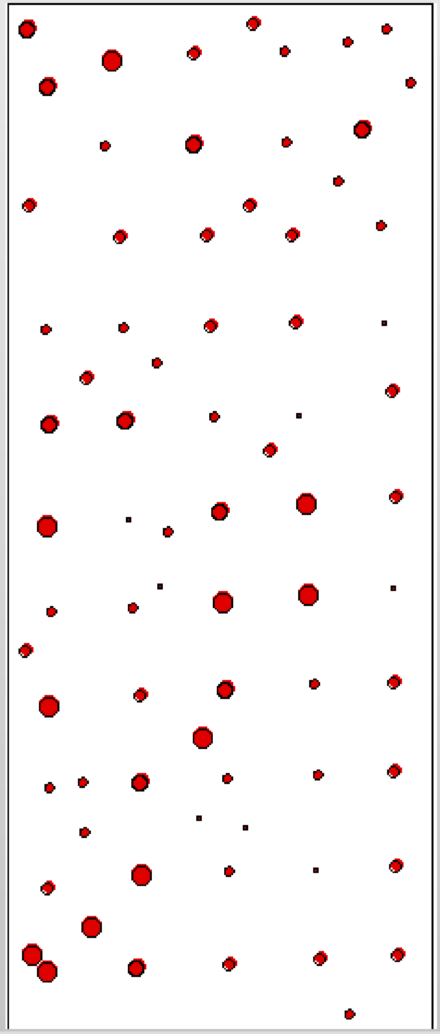
Example RBF profile



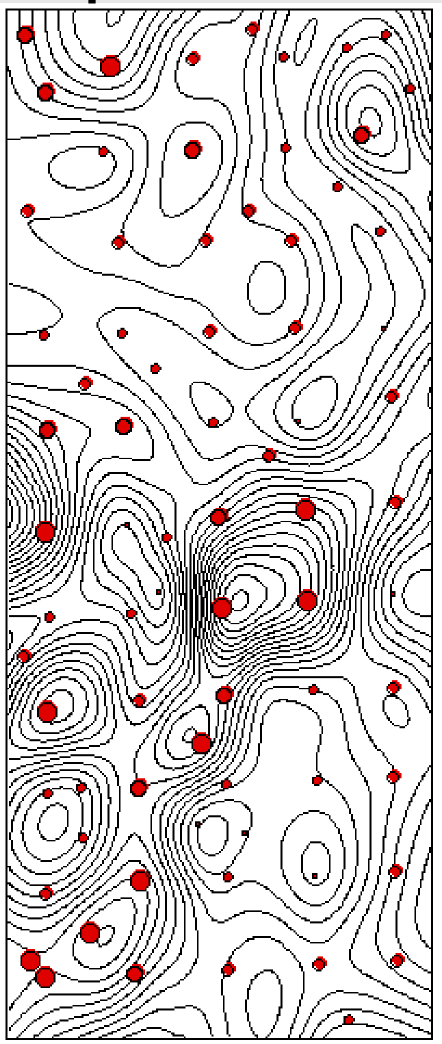
Example IDW profile

Interpolation algor.: Radial basis functions

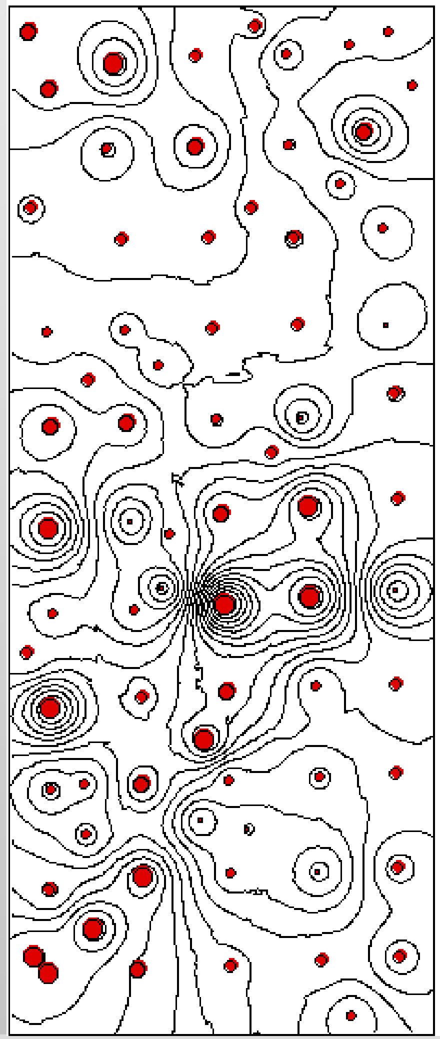
Point data



Spline result



IDW result



Interpolation algor.: Radial basis functions

- RBFs are used to produce smooth surfaces from a large number of data points.

- The functions produce good results for gently varying surfaces such as elevation.

- The techniques are inappropriate when
  – large changes in the surface values occur within short distances
  – the sample data is prone to measurement error or uncertainty

Interpolation algorithms

- Trend surfaces
- Nearest neighbour (Thiessen)
- Inverse distance interpolation / weighted moving average (IDW)
- Spline/local polynomials
- Kriging (analyses of spatial variation)

next lecture

Summary of interpolation tool

Tool	Description
Trend	Interpolates a raster surface from points using a trend technique.
Natural Neighbour	Interpolates a raster surface from points using a natural neighbour technique.
IDW	Interpolates a raster surface from points using an inverse distance weighted (IDW) technique.
Spline	Interpolates a raster surface from points using a two-dimensional minimum curvature spline technique. The resulting smooth surface passes exactly through the input points.
Spline with Barriers	Interpolates a raster surface, using barriers, from points using a minimum curvature spline technique. The barriers are entered as either polygon or polyline features.
Topography to Raster	Interpolates a raster surface from point, line, and polygon data. E.g. contour line
Kriging	Interpolates a raster surface from points using kriging.

Learning Objectives

1

GIS interpolation
- What?
- When?
- Data sources for interpolation

2

- Spatial Interpolation Methods
- Interpolation algorithms

3

Interpolation examples and application

Today's topics

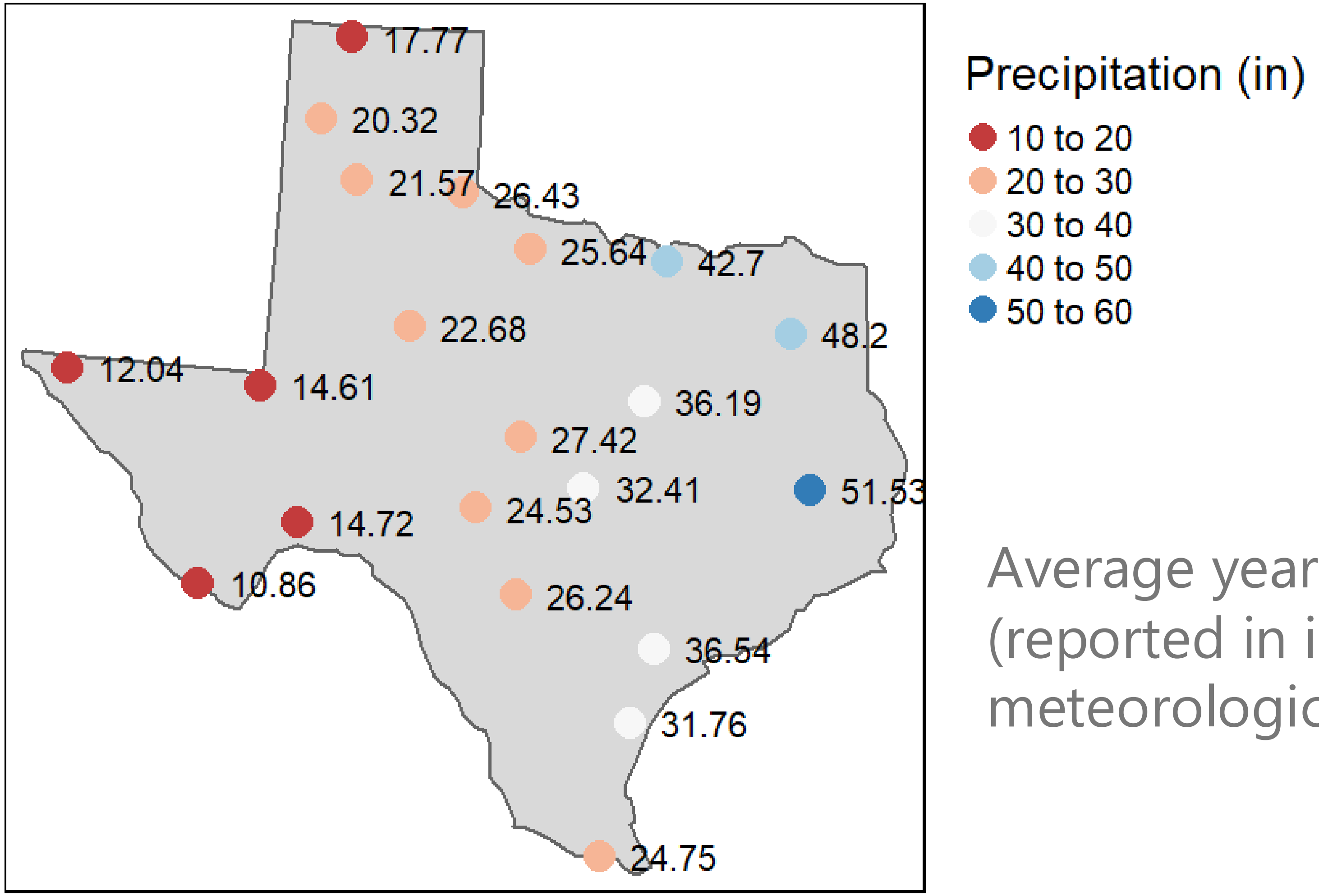
Interpolation tool: Summary & application

Given a distribution of point meteorological stations showing precipitation values,

How can I estimate precipitation values where data have not been observed?

Interpolation tool: Summary & application

Distribution of meteorological stations (points) showing precipitation values

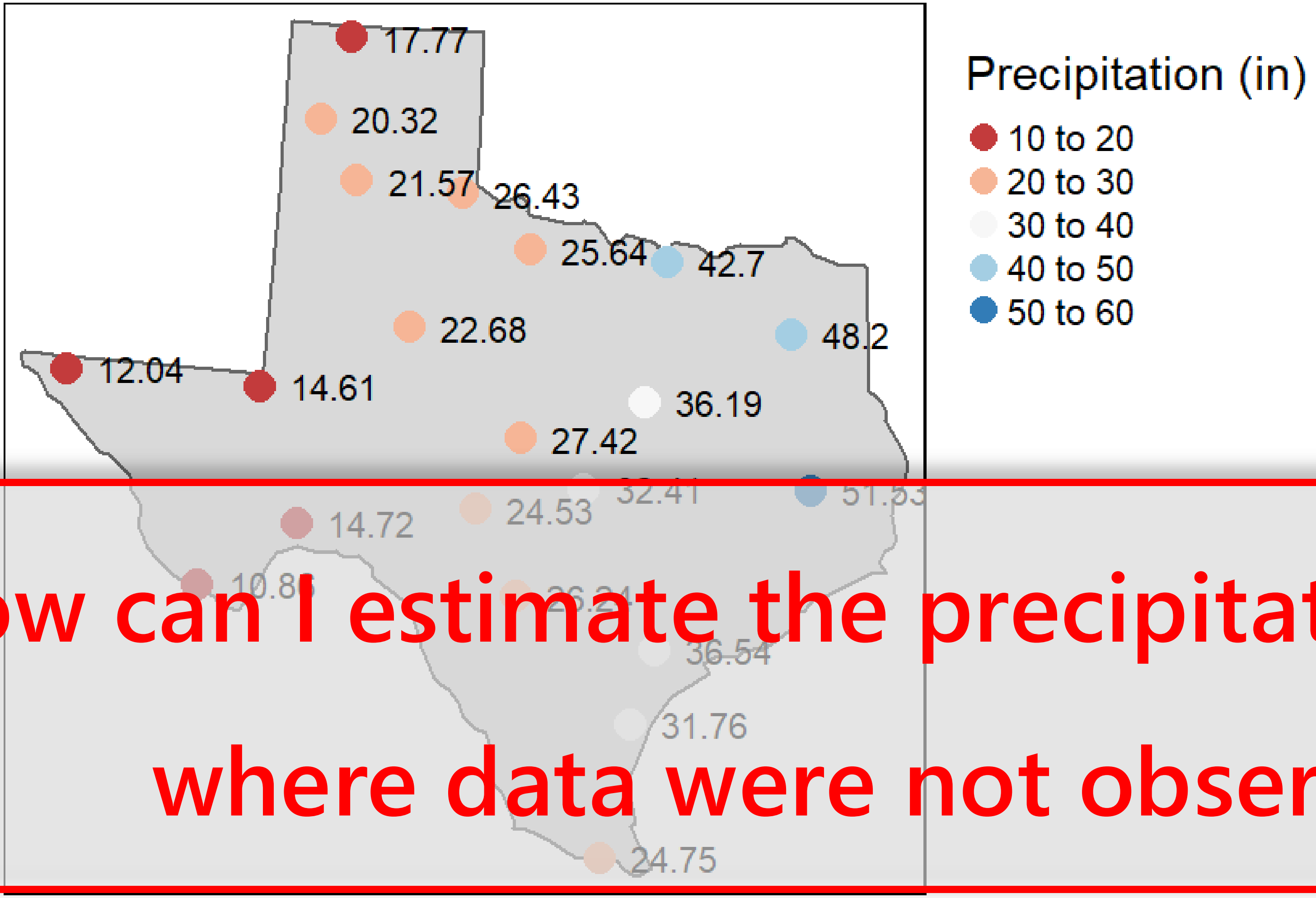


Average yearly precipitation (reported in inches) for several meteorological sites in Texas

https://mgimond.github.io/

Interpolation tool: Summary & application

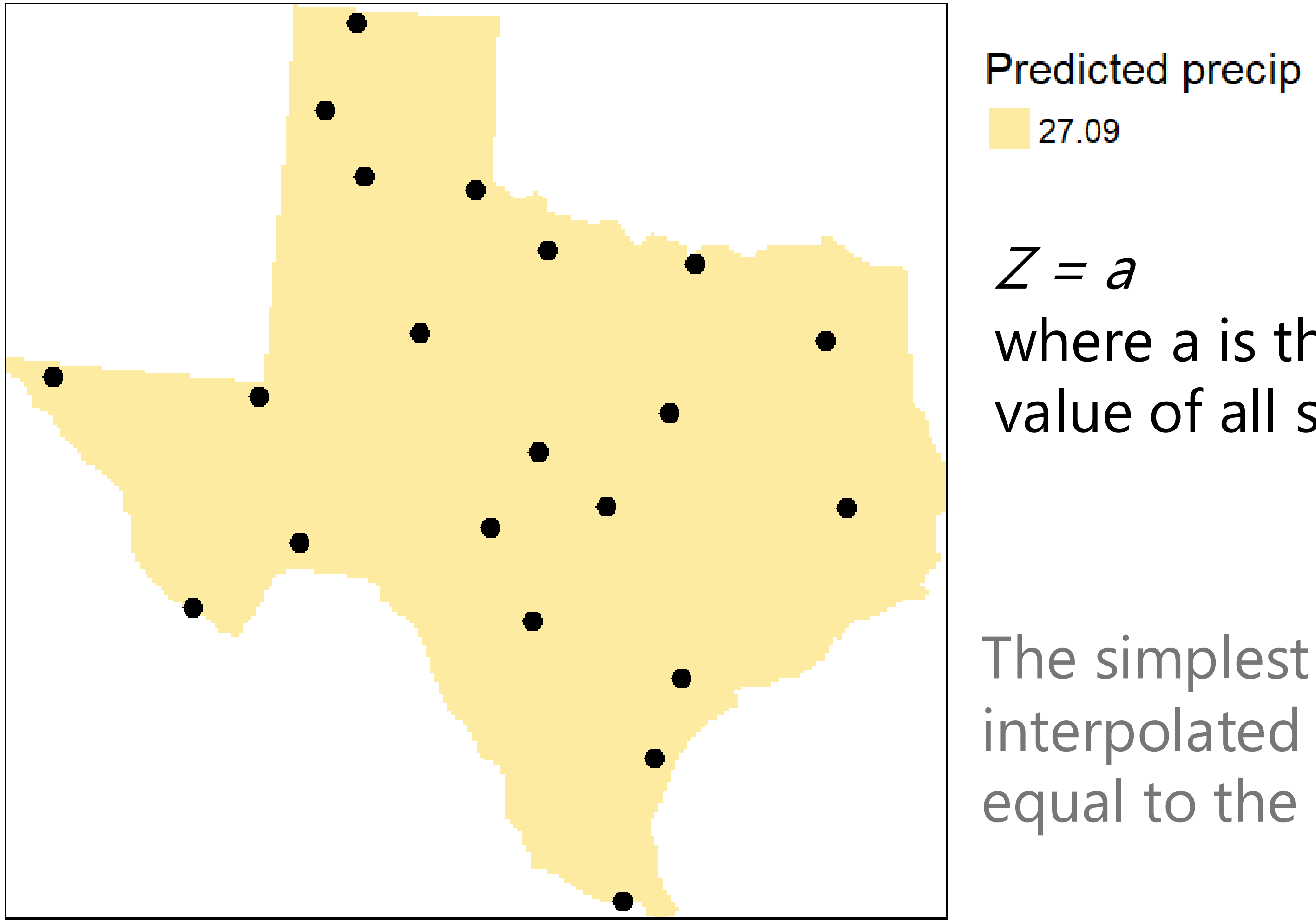
Distribution of meteorological stations (points) showing precipitation values



how can I estimate the precipitation values where data were not observed?

Interpolation tool: Summary & application

0th Order Trend Surface



$Z = a$

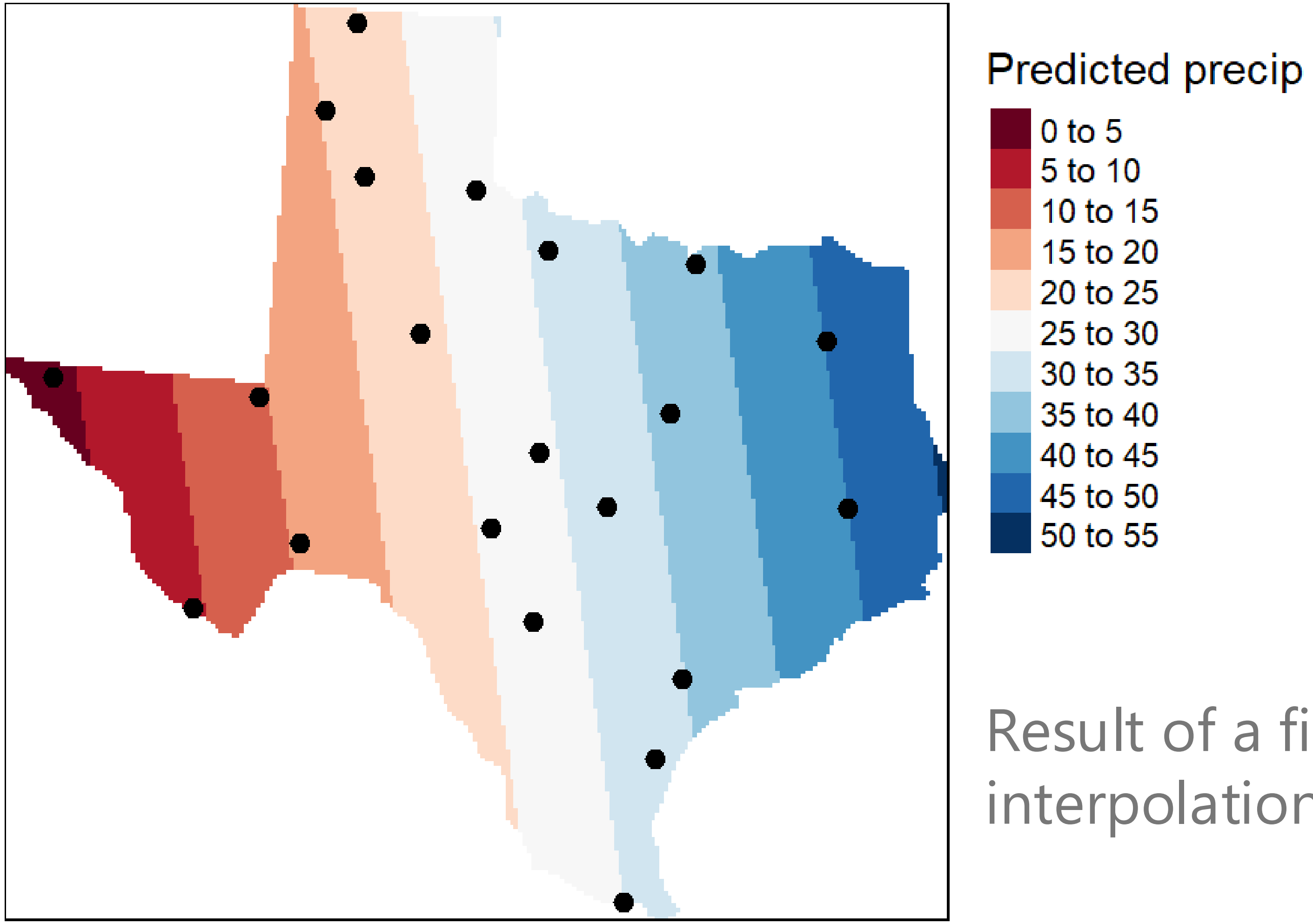
where a is the mean precipitation value of all sample points

The simplest model where all interpolated surface values are equal to the mean precipitation.

https://mgimond.github.io/

Interpolation tool: Summary & application

1st Order Trend Surface



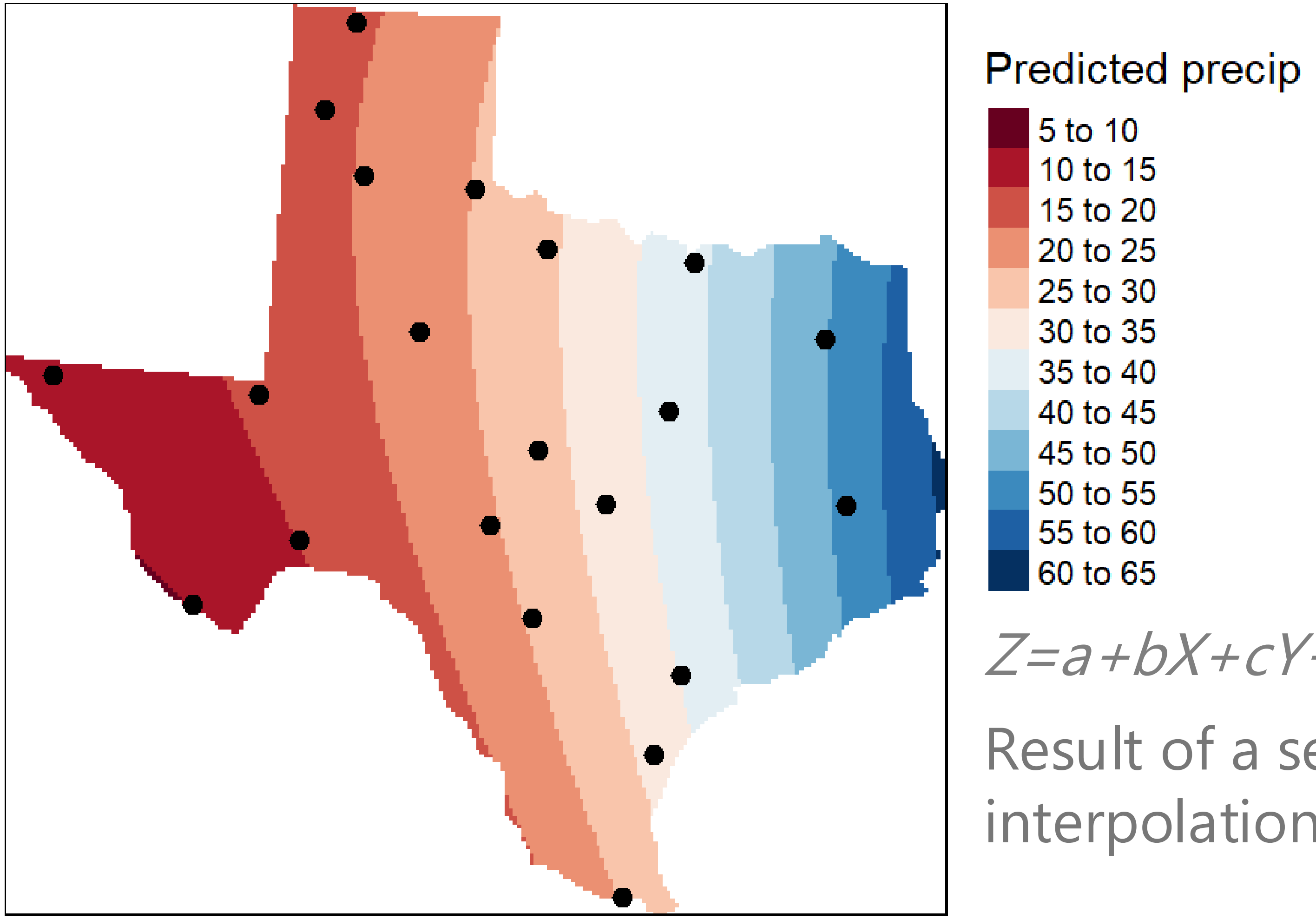
$Z = a + bX + cY$
where X and Y are the coordinate pairs

Result of a first order interpolation

https://mgimond.github.io/

Interpolation tool: Summary & application

2nd Order Trend Surface (quadratic polynomial)



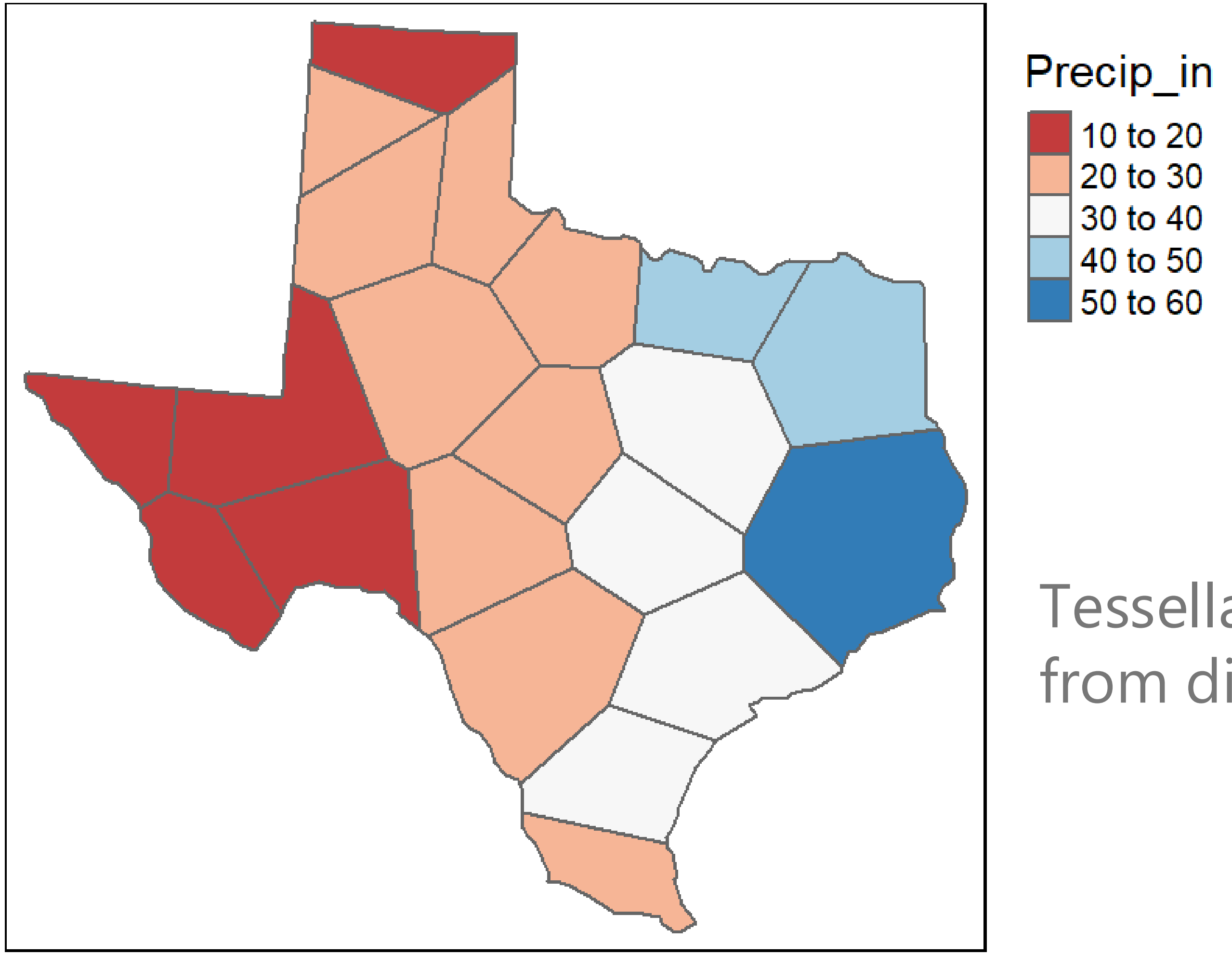
$Z=a+bX+cY+dX2+eY2+fXY$

Result of a second order interpolation

https://mgimond.github.io/

Interpolation tool: Summary & application

Thiessen interpolation

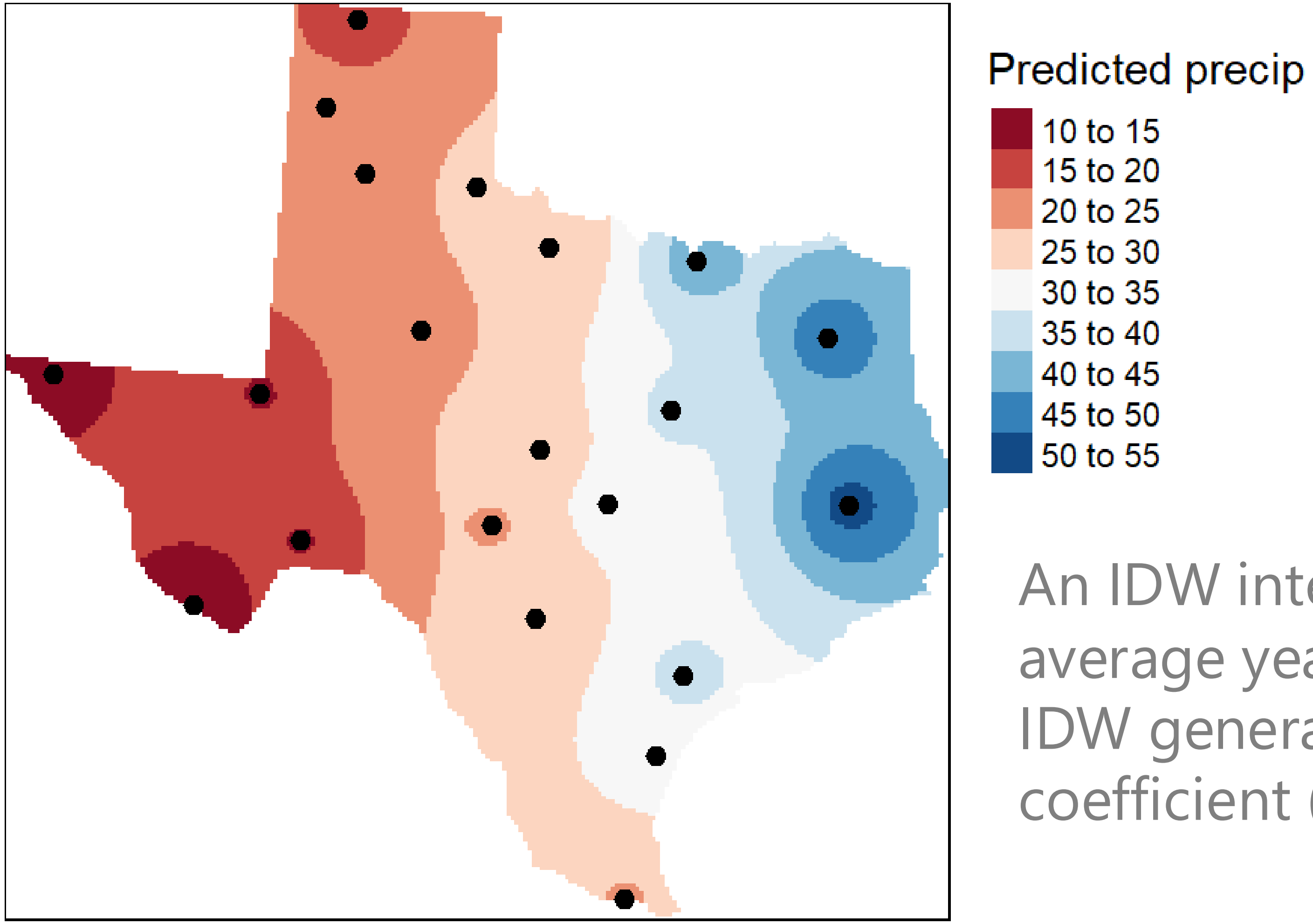


Tessellated surface generated from discrete point samples.

https://mgimond.github.io/

Interpolation tool: Summary & application

IDW interpolation, power of 2



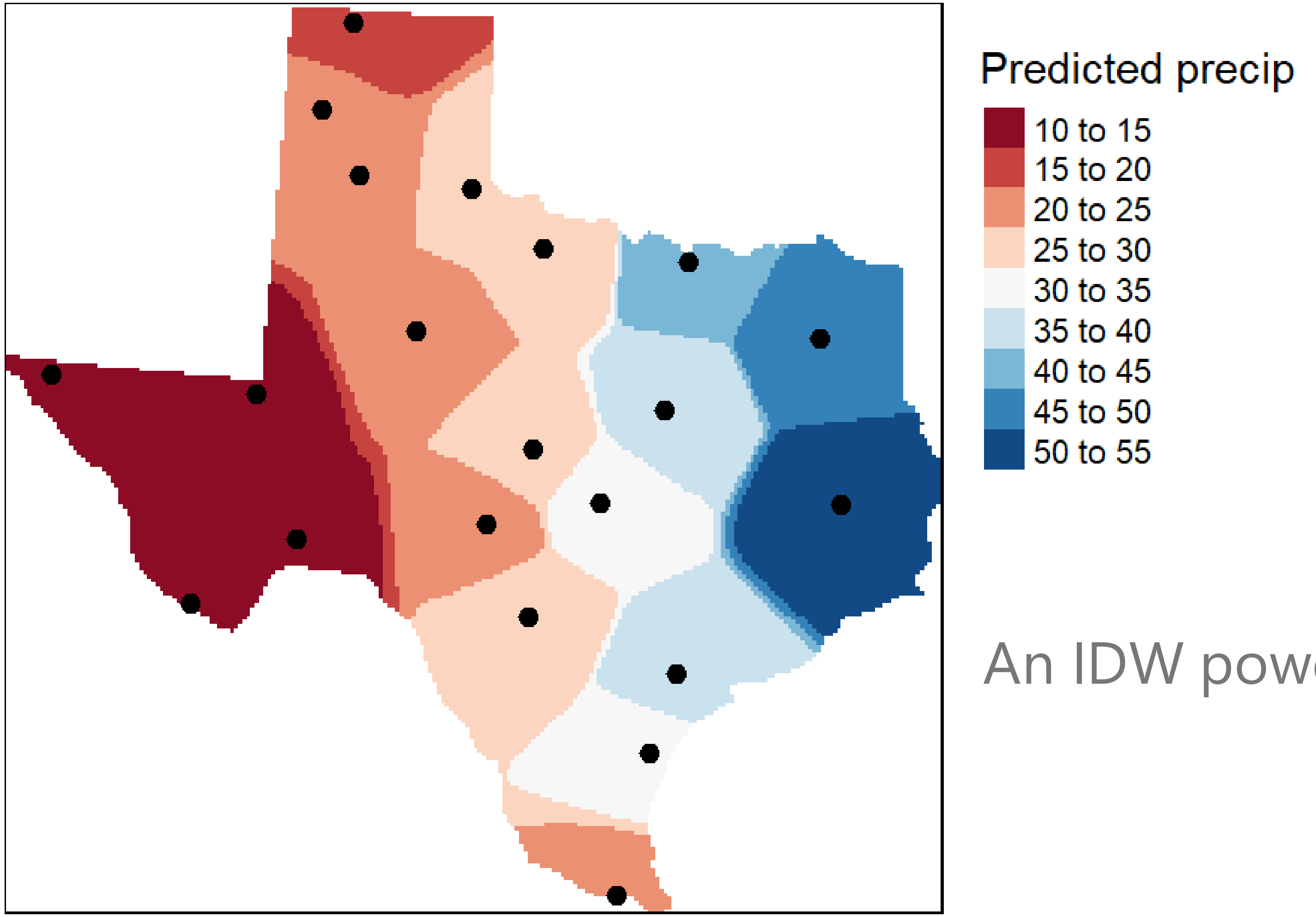
$$\hat{Z}_j = \frac{\sum_i Z_i / d_{ij}^n}{\sum_i 1/d_{ij}^n}$$

An IDW interpolation of the average yearly precipitation. An IDW generated with power coefficient (n) of 2

https://mgimond.github.io/

Interpolation tool: Summary & application

IDW interpolation, power of 15

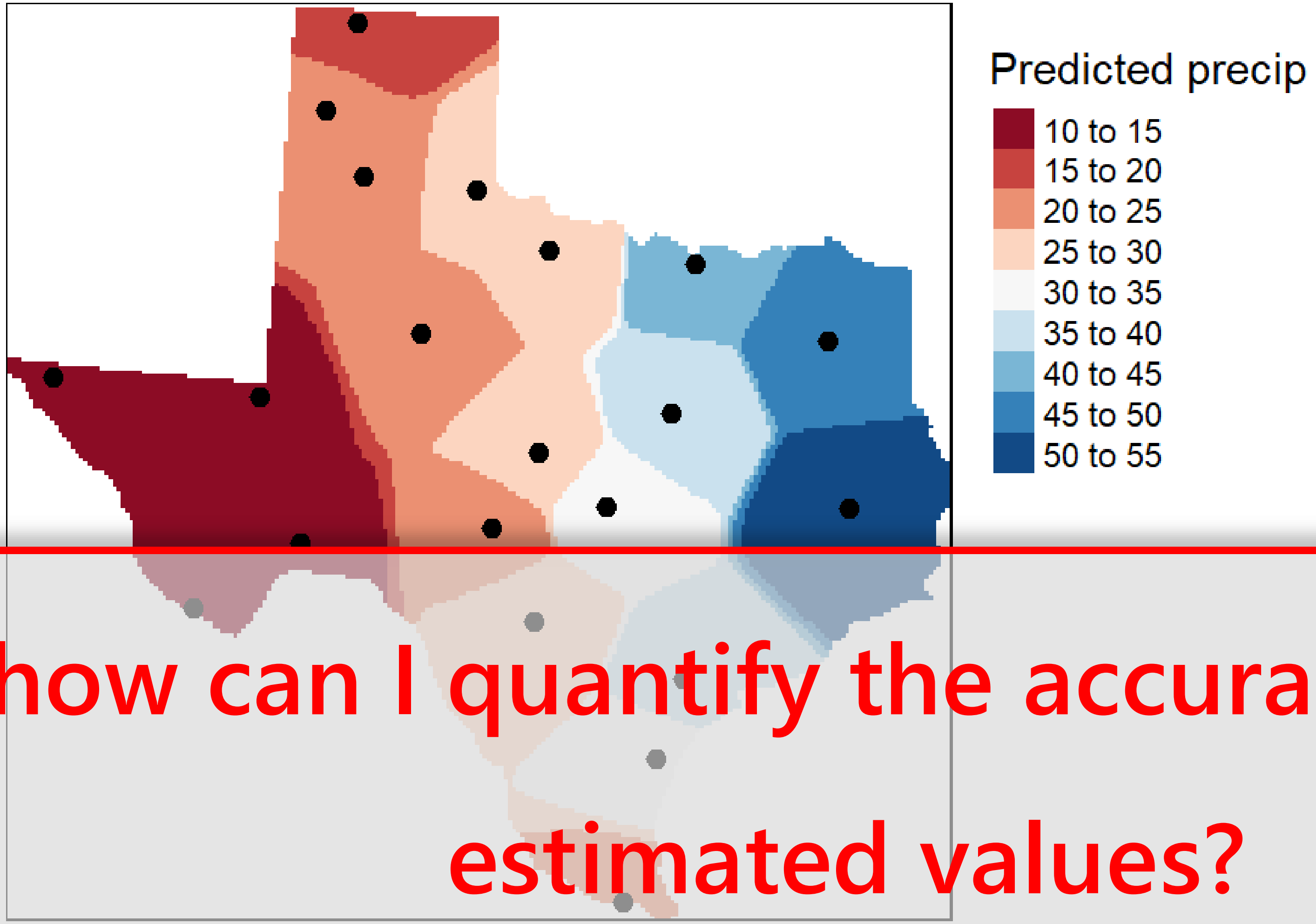


$$\hat{Z}_j = \frac{\sum_i Z_i / d_{ij}^n}{\sum_i 1/d_{ij}^n}$$

An IDW power coefficient n of 15

https://mgimond.github.io/

Interpolation tool: Summary & application



how can I quantify the accuracy of the estimated values?

Accuracy of the estimated values

- **Option 1: split the points into two sets:**
  - The points used in the interpolation operation and the points used to validate the results
- **Option 2: so called “leave-one-out cross validation analysis”**
  - Remove one data point from the dataset and interpolate its value using all other points in the dataset
  - Repeat this process for each point in that dataset (the interpolator parameters remain constant)
  - The interpolated values are then compared with the actual values from the omitted point
    - Statistics: example root-mean of squared residuals (RMSE)

Accuracy of the estimated values

- Option 2: so called "leave-one-out cross validation analysis"

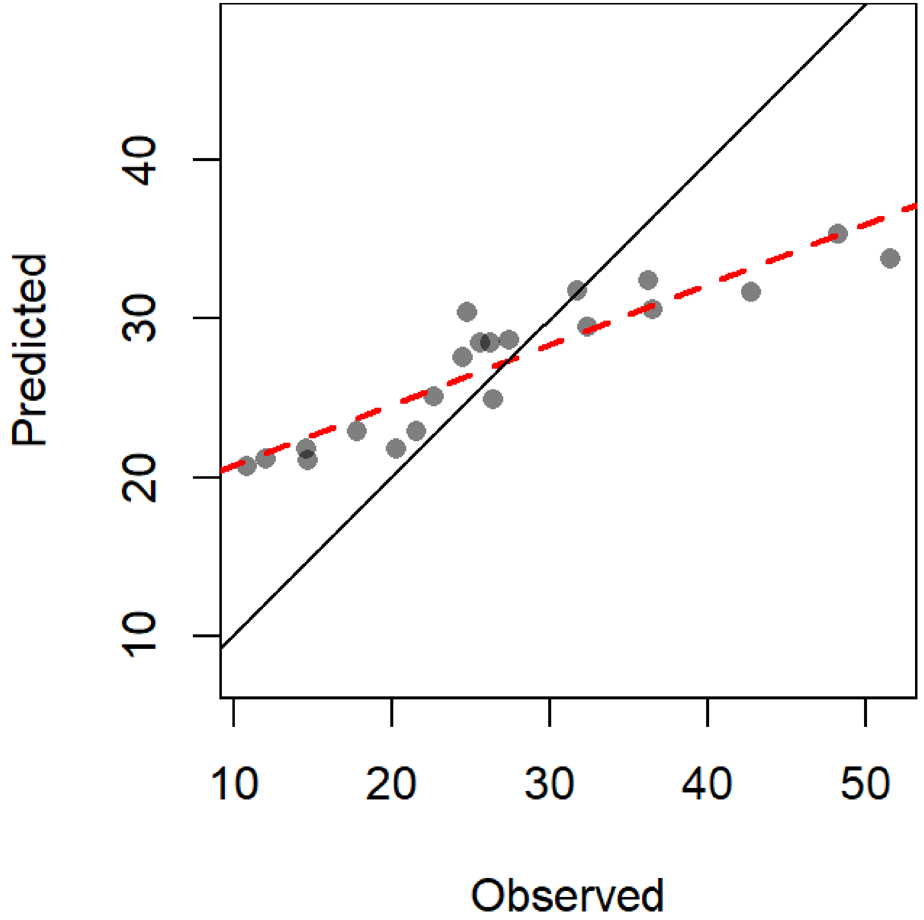
$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(\hat{Z}_i - Z_i)^2}{n}}$$

Where

- $\hat{Z}_i$ is the interpolated value at the unsampled location *i* where the sample point was removed
- Z_i is the true value at location *i*
- *n* is the number of points in the dataset

Accuracy of the estimated values

- Option 2: so called "leave-one-out cross validation analysis"



Scatter plot fitting predicted values vs. the observed values at each sampled location following a leave-one-out cross validation analysis

https://mgimond.github.io/

Thanks!

Feedback questions