Applied Spatiotemporal Analysis and Modeling

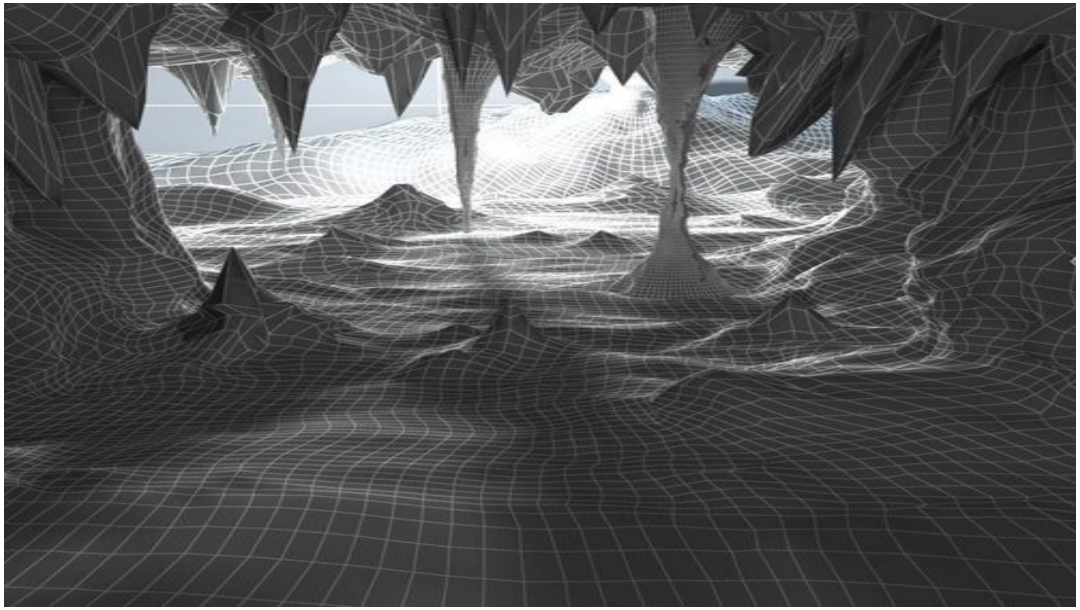
Guofeng Cao
Department of Geosciences
Texas Tech University

Spatial Fields

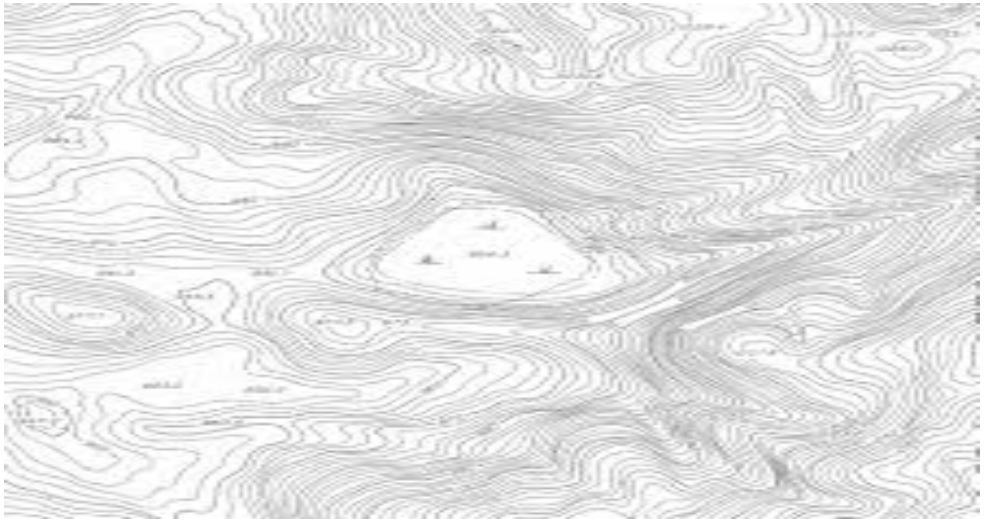
- Scalar versus vector fields:
  - scalar: quantity characterized only by its magnitude
    - *scalar fields* have a single value associated with each location
    - examples: temperature, elevation, precipitation
  - vector: quantity characterized by its magnitude and orientation (e.g., wind speed and direction)
    - *vector fields* have multiple values associated with each location
    - examples: http://hint.fm/wind/

Spatial Fields

- The following discussion will focus on scalar fields with the characteristics:
  - *continuity:* every location can be associated with a value
  - *uniqueness:* any location has one and only one value
  - 2.5 dimensions
- Compared with 3D dimensional cases:

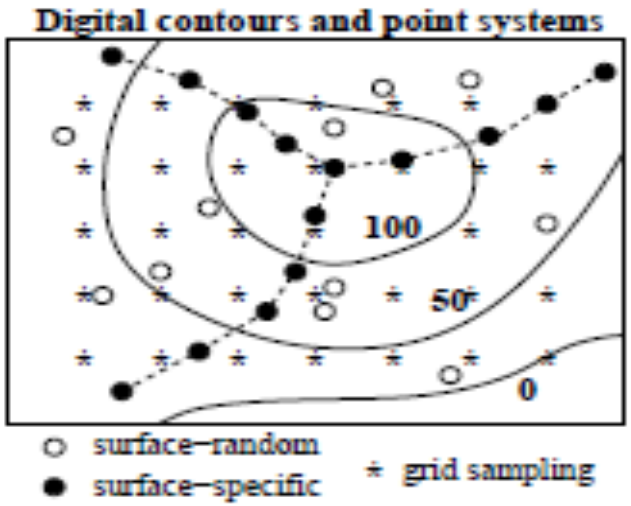


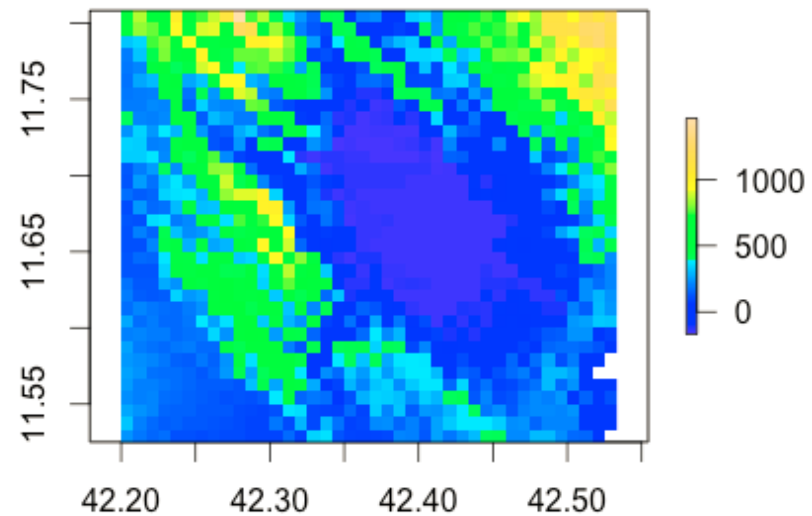
Surface Representation: Contours



- accuracy of digital sample depends on scale and accuracy of source analog map
- details falling between contour lines are lost
- oversampling of steep slopes (many contours) relative to gentle ones (few contours)
- many surface processing operations (e.g., slope calculation or point value determination) are extremely difficult to automate

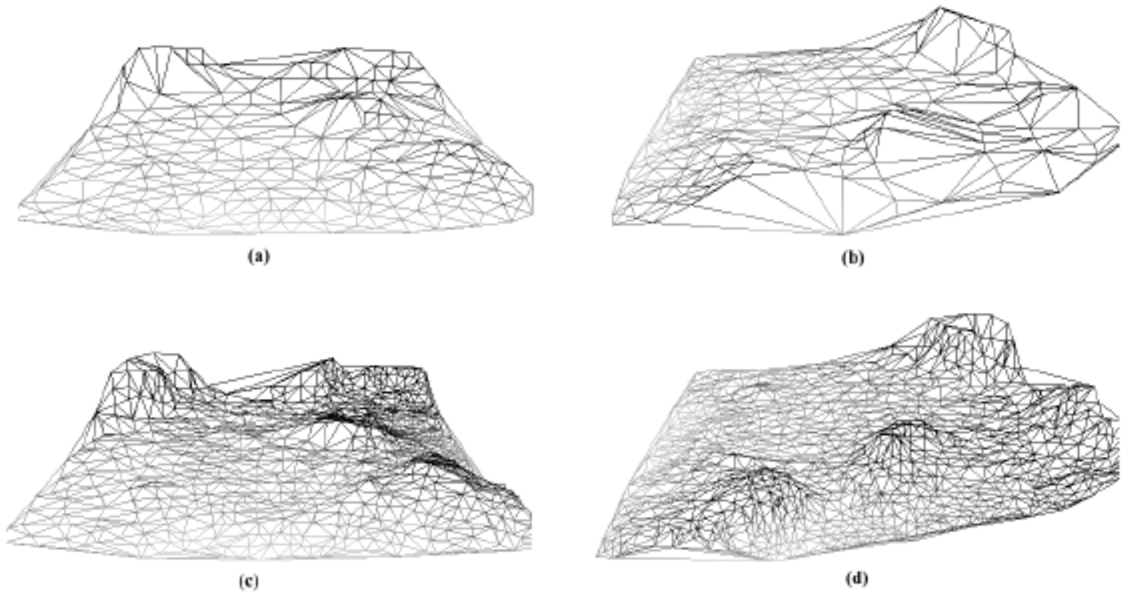
Surface Representation: Point Systems





- uniform data density enables display and surface processing
- no need to store spatial coordinates, just a single point and the grid spacing and orientation
- much larger sample size is required to enhance details (spatial resolution)
- Details/accuracy is controlled by the cell size/resolution
- Value of each cell is homogeneous represented by one single value

Surface Representation: Triangulated Irregular Network



- extremely compact way of storing fields, and their properties (e.g., slope, aspect)
- can capture important surface characteristics
- accuracy depends on accuracy of underlying field (assumed known)

Sampling Spatial Fields

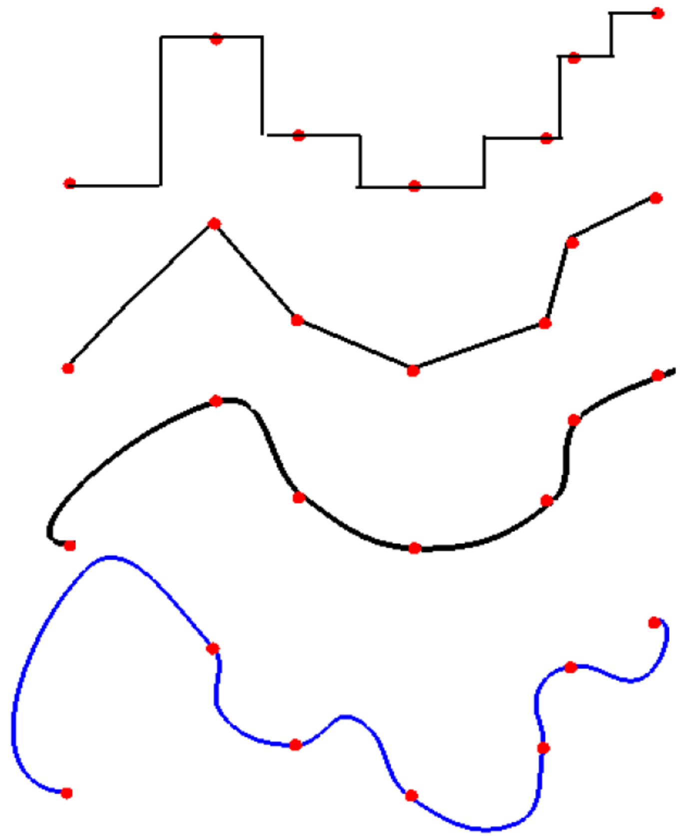
- **Sampling schemes:**
  - collection of measurements at a set of locations(e.g., precipitation at rain gauges, elevation spot heights)
  - regular grids obtained from aerial and/or satellite remote sensing (such measurements are area integrals)
  - digitized contour maps = points from digitized contours derived from analog topographic maps

Sampling Spatial Fields

- **Issues to consider:**
  - data constitute a *sample* of the underlying continuous field (exhaustive sampling is almost always impossible)
  - measurements might have both spatial *and temporal* components
  - often, data are not collected at random $\Rightarrow$ biased and non-random sampling
  - sometime, contours maps are derived from spot heights $\Rightarrow$ digitized contour maps should be treated with caution
  - All measurements are subjective to **<u>uncertainty</u>** (spatial uncertainty, more in the next lecture)

Spatial Interpolation

- Why spatial interpolation:
  - Observations/samples are sparse
- Interpolation: discrete->continuous
- Underline Rationale
  - Again, TFL
- It is difficult



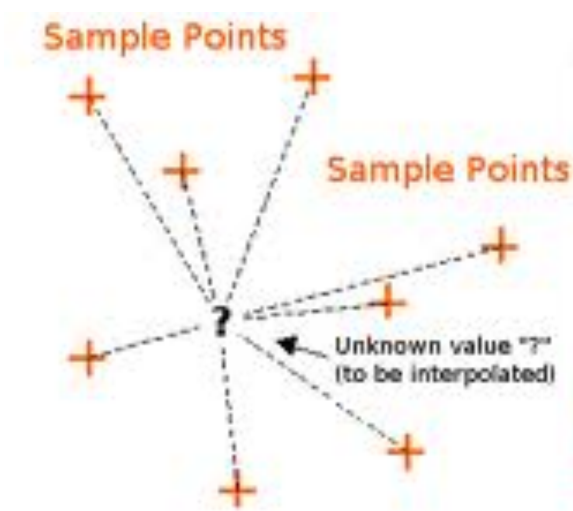
Spatial Interpolation

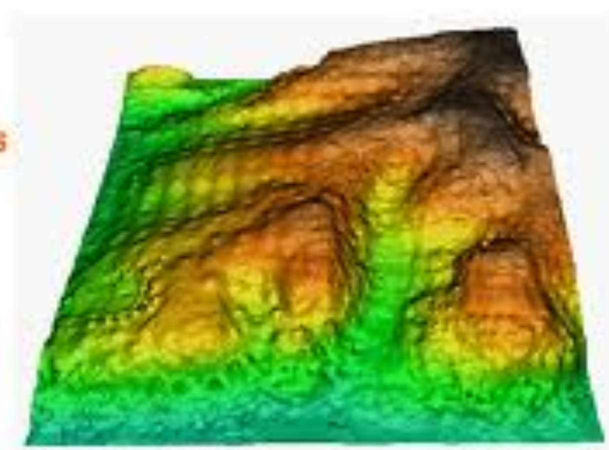
- General formulation of spatial interpolation

unknown value $z(s_i)$ at any non-sampled location s_i expressed as weighted average of n sample data $\{z(s_\alpha),\ \alpha = 1,\ldots \mathrm{n}\}$:

$$z(s_i) = \sum_{\alpha=1}^{n} w_{i\alpha} z(\mathrm{s}_\alpha)$$

$w_{i\alpha}$ denotes weight given to datum $z(s_\alpha)$ for prediction at location s_i



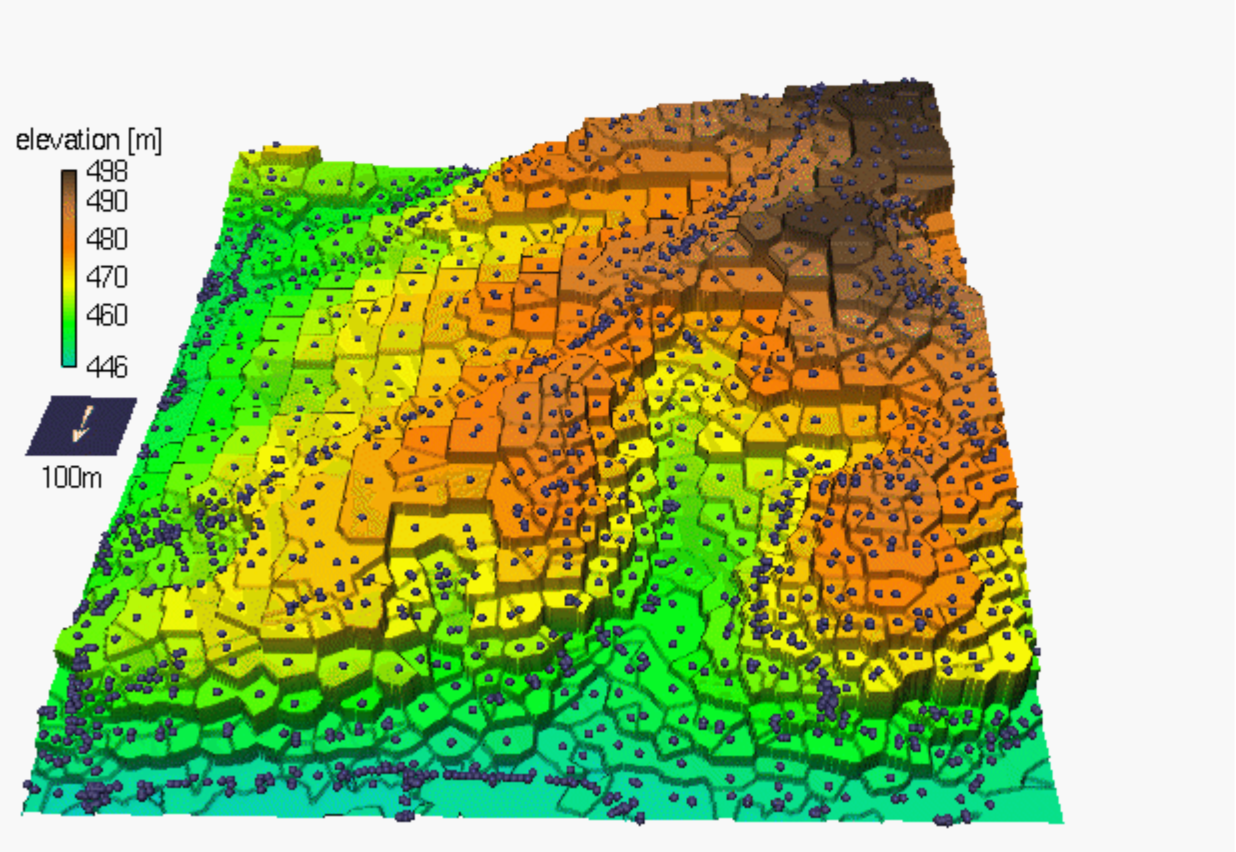


Spatial Interpolation Methods

- Deterministic Interpolators
  - Nearest Neighbor/Natural neighbor
  - Trend Surface
  - Inverse distance weighted method
  - Spatial spline
  - Triangulation
- Stochastic Interpolators
  - Kriging
  - Outcome the credibility information compared to the deterministic interpolators

Spatial Interpolation: Nearest Neighbor

- Assign value of nearest sample point
- Thiessen Polygons/Voronoi diagram



Spatial Interpolation: Nearest Neighbor

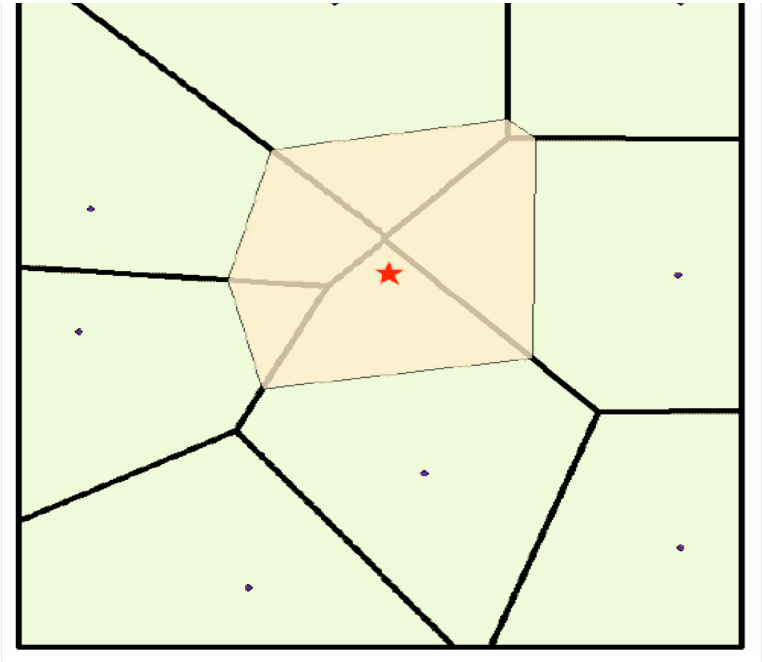
- Datum closest to the prediction location receives all weights

$$z(s_i) = \sum_{\alpha=1}^{n} w_{i\alpha} z(\mathrm{s}_\alpha) = z(\mathrm{s}_\alpha) + \sum_{\alpha=1}^{n-1} 0 * z(\mathrm{s}_\alpha)$$

- Unbiased estimation $\sum_{\alpha=1}^{n} w_{i\alpha} = 1$
- set of predicted values form discontinuous (patchy) surface

Natural Neighbor Interpolation

- Finds the closest subset of input samples to a query point and applies weights to them based on proportionate areas in order to interpolate a value
- “Area-stealing”
- Local interpolation: using only a subset of samples that surround a query point



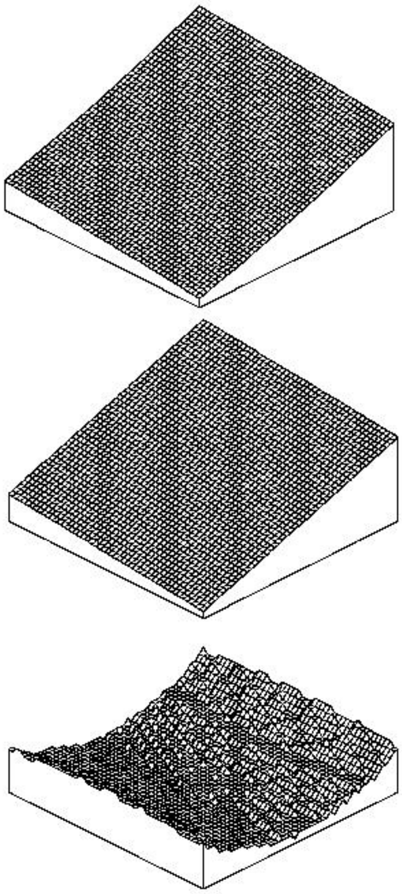
Spatial Interpolation: Trend Surface

- explicit mathematical function(s) of coordinates that interpolates or approximates (smooths) the surface. For example:

$$z(s_i) = a_0 + a_1 * x + a_2 * y$$

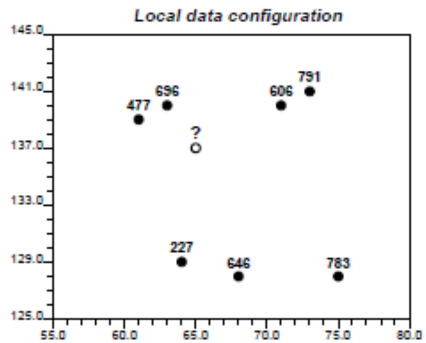
$$or$$

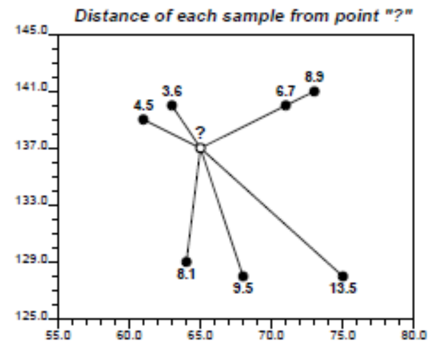
$$z(s_i) = a_0 + a_1 * \mathrm{x}^2 + a_2 * y^2 + a_3 xy$$



- surface operations (e.g., curvature) and values can be analytically computed
- Fit polynomial equation to sample points
- Goal is to minimize deviations between sample points and surface
- arbitrary choice of number and type of functions
- local versus global fitting

Spatial Interpolation: Inverse Distance





Procedure:

- predict unknown value $z(\mathbf{s}_i)$ at any non-sampled location $\mathbf{s}_i$ as weighted linear combination of $n(\mathbf{s}_i)$ nearby data $z(\mathbf{s}_\alpha)$:

$$\hat{z}(\mathbf{s}_i) = \sum_{\alpha=1}^{n(\mathbf{s}_i)} w_{i\alpha} z(\mathbf{s}_\alpha)$$

  where $w_{i\alpha}$ denotes weight received by sample $z(\mathbf{s}_\alpha)$ for prediction at location $\mathbf{s}_i$

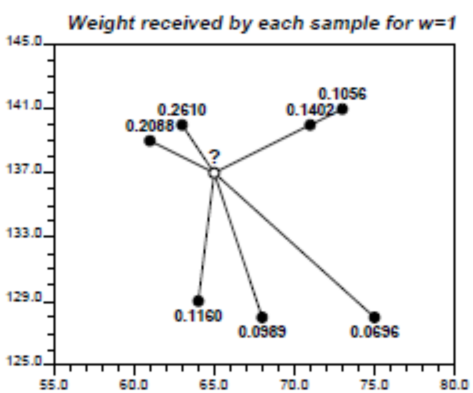
- make weight $w_{i\alpha}$ inversely proportional to power k of distance $h_{i\alpha} = ||\mathbf{s}_i - \mathbf{s}_\alpha||$:

$$w_{i\alpha} = \frac{h_{i\alpha}^{-k}}{\sum_{\alpha=1}^{n(\mathbf{s}_i)} h_{i\alpha}^{-k}} = \frac{1/h_{i\alpha}^{k}}{\sum_{\alpha=1}^{n(\mathbf{s}_i)} 1/h_{i\alpha}^{k}}$$

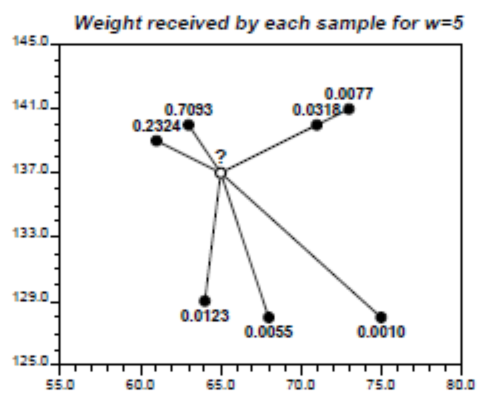
Spatial Interpolation: Inverse Distance

Characteristics:

- unbiased interpolation procedure, since $\sum_{\alpha=1}^{n(\mathbf{s}_i)} w_{i\alpha} = 1$
- "exact" interpolator: $\hat{z}(\mathbf{s}_\alpha) = z(\mathbf{s}_\alpha), \ \forall\alpha$
- exponent k controls importance of data closer to $\mathbf{s}_i$;
  e.g., $k = 2$: inverse distance squared interpolation



$\hat{z}(\mathbf{s}) = 598$



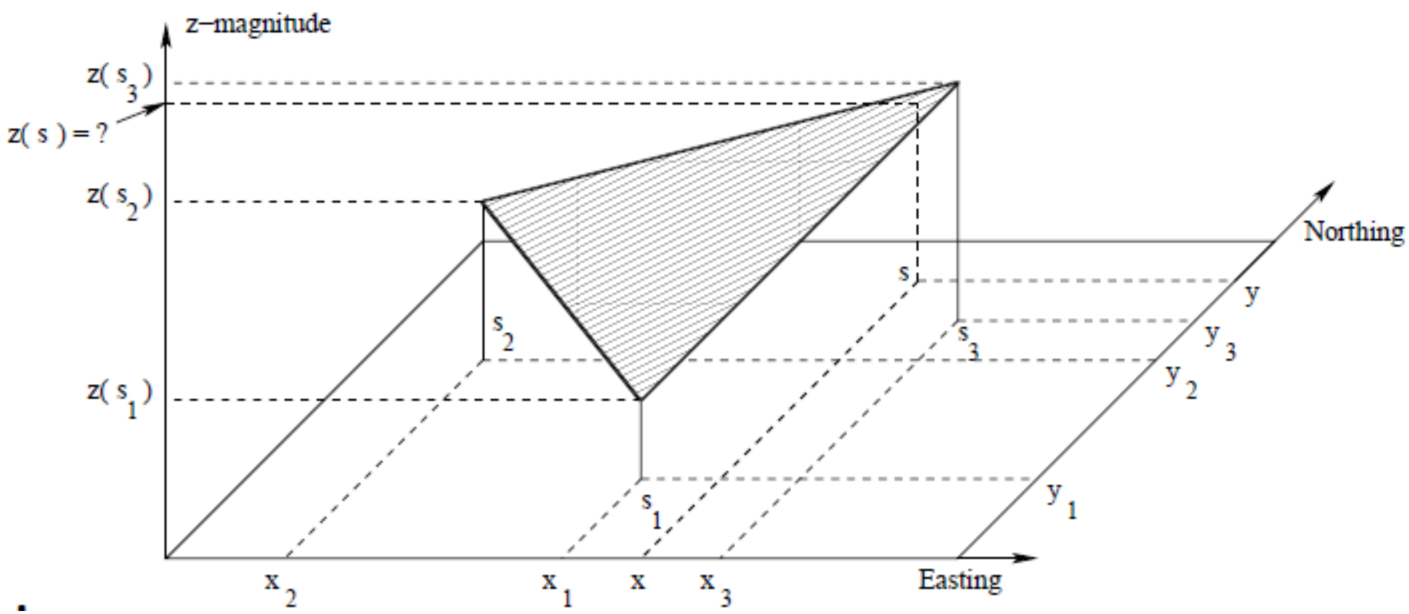
$\hat{z}(\mathbf{s}) = 637$

Spatial Spline

- Estimates values using a mathematical function that minimizes overall surface curvature
  - smooth surface
  - passes exactly through the input points

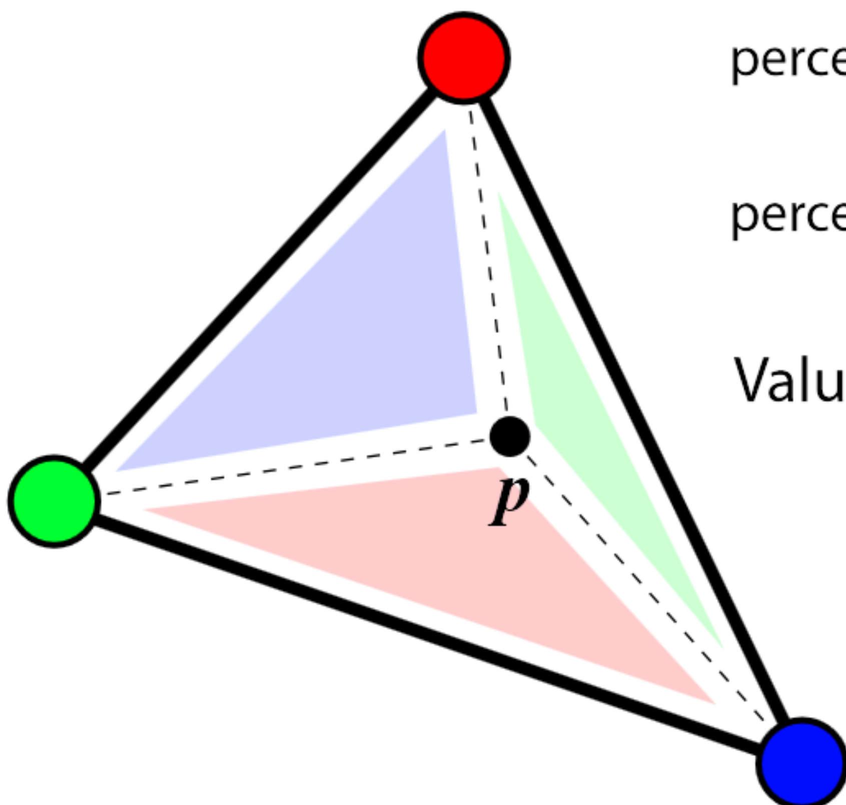
Spatial Interpolation: Triangulation

- Barycentric Interpolation



Spatial Interpolation: Triangulation

- Barycentric Interpolation



percent red = $\frac{\text{area of red triangle}}{\text{total area}}$

percent green = $\frac{\text{area of green triangle}}{\text{total area}}$

percent blue = $\frac{\text{area of blue triangle}}{\text{total area}}$

Value at p:

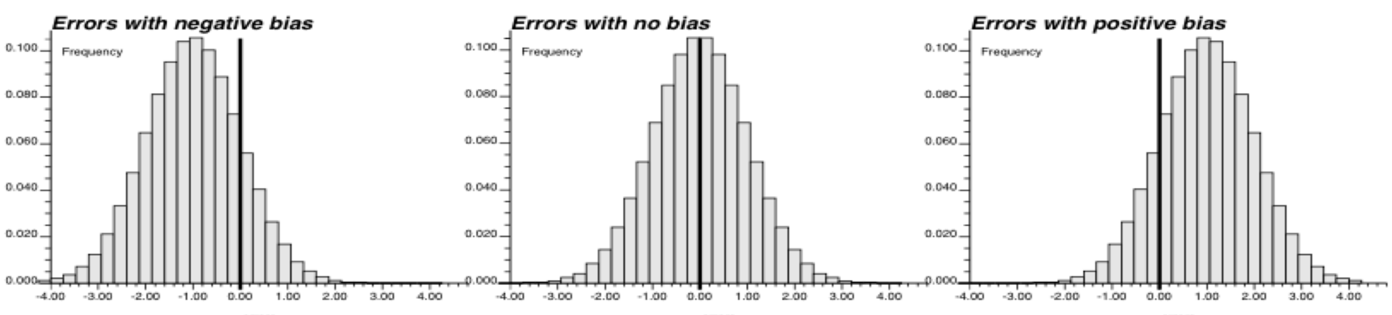
(% red)(value at red) +
(% green)(value at green) +
(% blue)(value at blue)

Evaluating Prediction Performance

- Cross-validation:
  - Loop over sample locations:
  - hide a sample datum
  - predict it from the remaining data using one of the spatial interpolation method
  - repeat until all sample locations are visited and cross-validation predictions are computed

Evaluating Prediction Performance

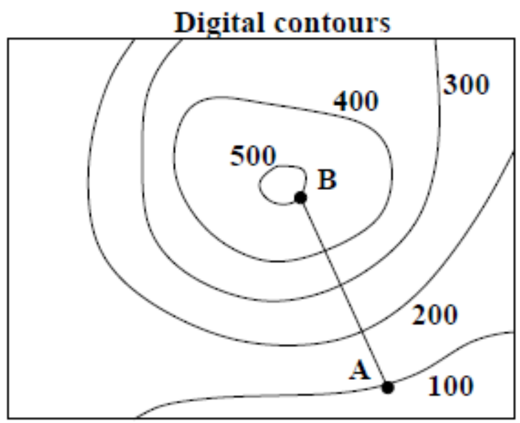
- Compare distribution of predicted values to that of true values for:
  - reproduction of mean (for possible bias), median, variance, and other summary statistics
  - reproduction of entire distribution of true values (QQ plot)

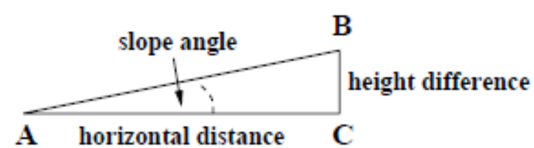


Surface Derivatives: Slope and Gradient

Gradient:

- vector quantity specified by (i) magnitude, and (ii) direction
- gradient magnitude = maximum rate of change of elevation at a point (slope)
- gradient direction = direction of steepest slope trough that point (aspect)





- calculating the tangent of the slope angle:

$$\tan(\theta) = \frac{\text{height difference}}{\text{horizontal distance}} = \frac{BC}{AC} \Rightarrow \theta = \arctan(\frac{BC}{AC})$$

in Matlab: $theta$=rad2deg(atan(BC/AC))

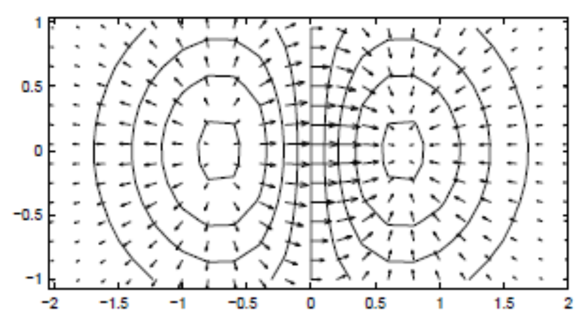
Surface Derivatives: Slope and Gradient

Gradient calculations:

- in TIN surface representation, gradient at s = gradient of containing Delaunay triangle
- in raster surface representation, gradient at s calculated using a square window (typically 9×9) centered at s.
  Slope θ and aspect α are calculated as:

  $$\theta = \sqrt{\theta_x^2 + \theta_y^2} \quad \text{and} \quad a = \arctan(\frac{\theta_x^2}{\theta_y^2})$$

  where θ_x and θ_y denote directional derivatives along x and y
  aspect α measured from vertical to direction of steepest slope;
  $\alpha = \alpha + 180$ if $\theta_y > 0$, and $\alpha = \alpha + 360$ if $\theta_x > 0$ and $\theta_y < 0$
- alternatively, a local mathematical surface is fitted within each window, and its derivative is analytically calculated



- End of this topic