Spatial Statistics - An Introduction

Upmanu Lall[1]
David K. Stevens[2]
Utah Water Research Laboratory
Utah State University
Logan UT 84322-8200

COLUMBIA UNIV.
ULA2@COLUMBIA.EDU

[1] ulall@pub.uwrl.usu.edu, 801-797-3184, FAX: 801-797-3663
[2] stevens@quito.eee.usu.edu, 801-797-3229, FAX: 801-797-1185

Outline

- ***Problem Considered***
  - Given scattered data on water levels, contaminant concentrations, rainfall, temperature, water levels, hydraulic conductivity etc. Estimate/map the variable at unsampled locations.

- ***Methods***
  - "Trend Surface" Estimation/**Regression**
    - Parametric Methods
    - Non-Parametric Methods
  - "Geostatistics" aka Kriging

- ***Related Issues***
  - Subsurface Characterization
  - Conditional Simulation
  - Space-Time Data - Principal Components + forecasts

Some Ideas Common to All Methods:

Let the sampling location i be identified as (x_i, y_i) or $\mathbf{x}_i$ and the observation at that location by Z_i

1) The estimation process is defined through

$$Z_i = f(\mathbf{x}_i) + e_i \qquad i = 1...n$$

where e_i is the error or difference between the "fit" $f(\mathbf{x}_i)$ at $\mathbf{x}_i$ and the observation Z_i.

For an exact interpolation scheme, $e_i = 0$; $e_i \neq 0$ may be due to measurement or approximation error or to "local effects"

2) The estimate at any location $\mathbf{x}$ can be written as a weighted average of the data

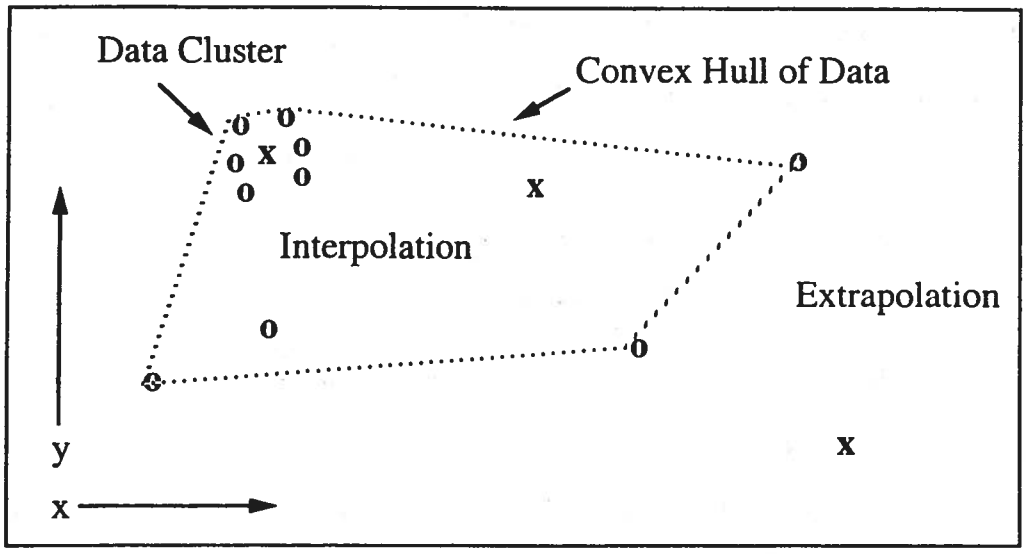
$$f(\mathbf{x}) = \sum_{i=1}^{n} w(\mathbf{x}, \mathbf{x}_i)\, Z_i$$

where the weights $w(\mathbf{x}, \mathbf{x}_i)$ depend ONLY on the location of the estimate, $\mathbf{x}$, the sample locations $\mathbf{x}_i$ and NOT on the variable observed, Z_i.

---

*Methods **differ** in how the weights are specified !*
*We'll look at the attributes of and "philosophy" behind some of the methods and seek a **uniform** way to compare them.*

Data Attributes



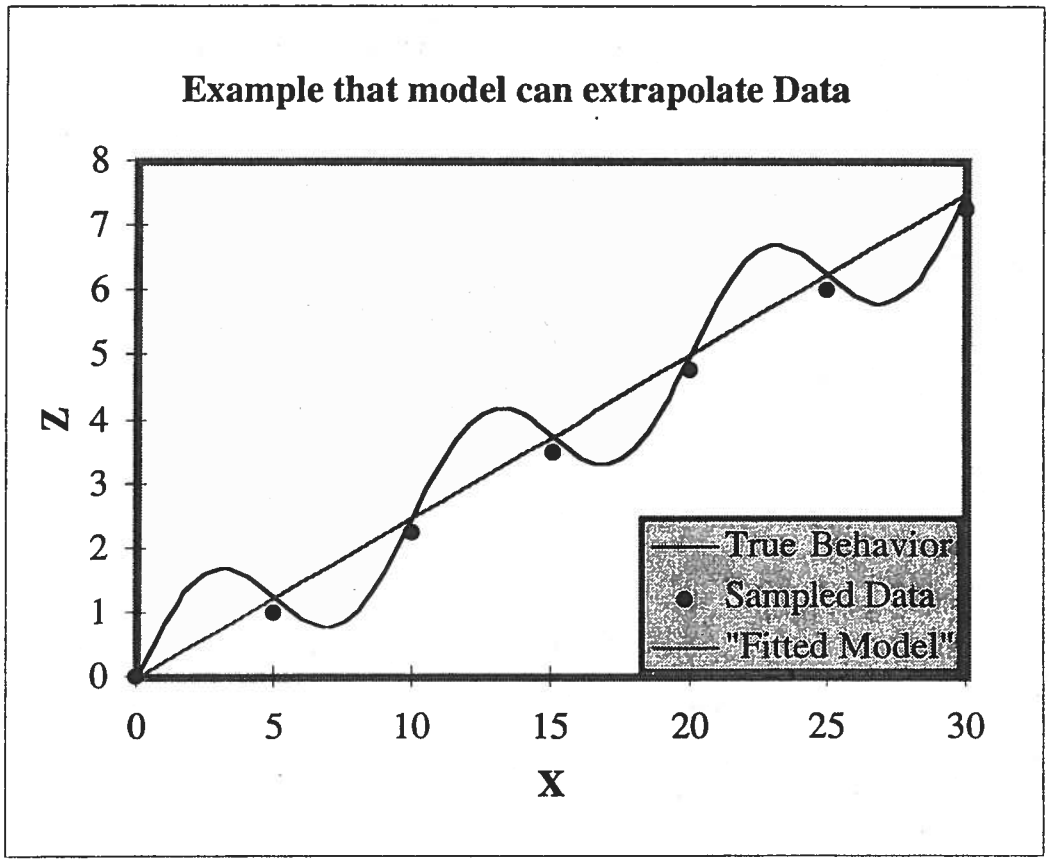
- Sampling may be clustered and sparse
- Only positive values of Z may be meaningful (-ve weights ?)
- Values of Z may vary over several orders of magnitude
- Z may represent a physical variable (e.g., rain in mm) or an indicator (e.g., Z=1 if rain is >5 mm ; 0 else)
- We may be interested in the point values of Z at various locations or its average over a region

What do we believe if we wish to interpolate ?

- Z(**x**) is a continuous and smooth function of **x.**
- The model we impose on the data represents this variation.
- The estimation point **x** lies in the convex hull of the sample.
- All estimates ***extrapolate*** data. Assumed model is correct ?

Sampling and Extrapolation

An extreme example



The True Function in the above example is

$$Z(x) = x/4 + \sin(2\pi\, x/10)$$

The Data was collected at x = 0, 5, 10, 15, 20, 25, 30.

At these points $\sin(2\pi\, x/10)$ is zero.

The fitted model (Z=x/4) is perfect for the data collected.

It "extrapolates" the data to the unsampled locations.

Predictive performance of the fit is poor.

Solution: Irregular sampling design + additional data

What is Usually Done:

- Given Z_i values at a set of locations $\mathbf{x}_i$; i=1...n.
- Construct a grid with spacing ∂x, ∂y over the domain.
- Apply a scheme to estimate Z_j (or **really** $f(\mathbf{x}_j)$) at the grid nodes $\mathbf{x}_j$; j=1...np.
- Linearly interpolate across grid nodes to develop contours or a surface plot of $Z(\mathbf{x})$.

So What is Going On ?

- The process of estimating the $f(\mathbf{x}_j)$ "*smooths*" the original data, since weighted averages are being formed.
  - Most people appreciate this aspect of contouring.

- The contour or surface drawing process further *smooths* the original data by interpolating across the evaluation grid.
  - This aspect is often ignored.
  - The statistical lore focuses on ways to develop the $f(\mathbf{x}_j)$ estimates and their accuracy.
  - The choice of the grid spacing, ∂x, ∂y may also be critical for proper perception.

Issues in Spatial Estimation

1. How much data do I need ?
   - How accurate do the estimates need to be ?
   - Cost of data collection.
   - Complexity of process.

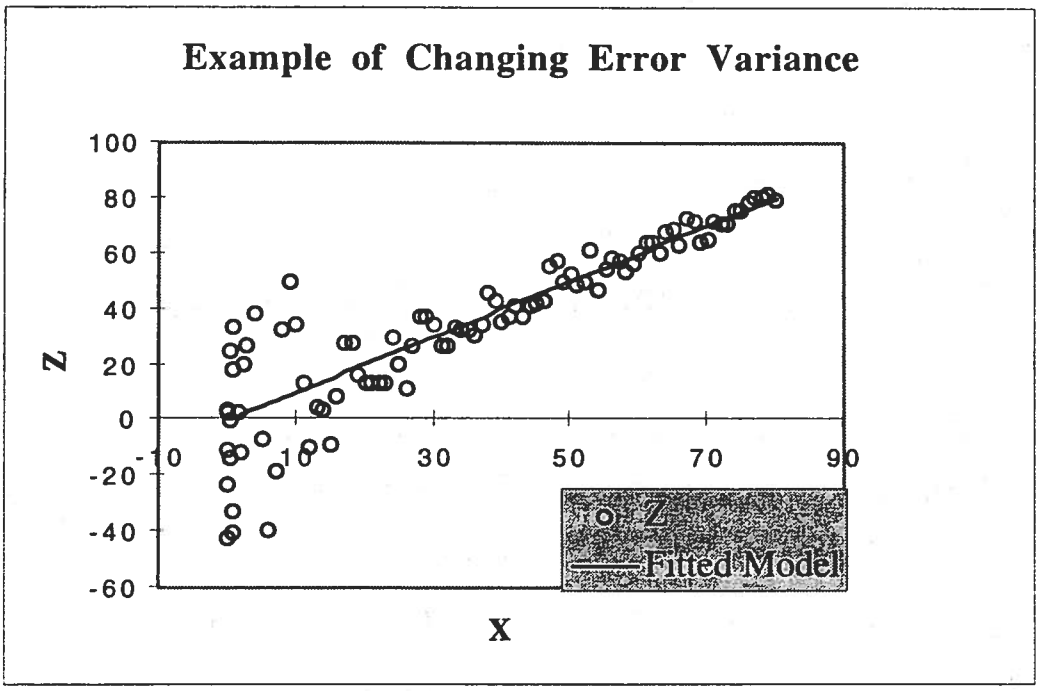
2. What is the average error of estimation for my scheme ?
   - Error in fitting data ?
   - Error when predicting at an unsampled location.
     - Cross Validation or leave one out and predict ?
   - Hope the average error is 0, i.e., scheme is ***unbiased***.

3. What is the spread of errors ?
   - What is the variability of the scheme/data ?
     - Error ***Variance***.
   - Can I develop pointwise confidence limits ?

4. How do I measure process complexity ?
   - Separation of signal (f(**x**)) from noise (e).
   - Formal Definition of Smoothness or continuity of Z(**x**)
     - Covariance based or Curvature based.

$\text{cov}(Z(\mathbf{x}), Z(\mathbf{x}+\mathbf{h}))$ $\int f''(\mathbf{x})^2 d\mathbf{x}$

5. These properties can be considered **globally** (i.e. averaged over the site) or **locally** (using only data near **x**).
   - Consider the extrapolation example. The average global error of the scheme is 0; i.e., it is **globally unbiased**. But the local error is non-zero at all the prediction points (except the original data locations), and so the scheme is **locally biased** unless $x \pm i*5$, for some integer i.
   - If the model was "correct", i.e. it is globally and locally unbiased, global and local error **variances** may not be the same as shown below.



6. Often, a statistical criteria such as the "**Mean Square Error**" (MSE) is used to judge the "goodness of fit" of a model and to choose the weights.

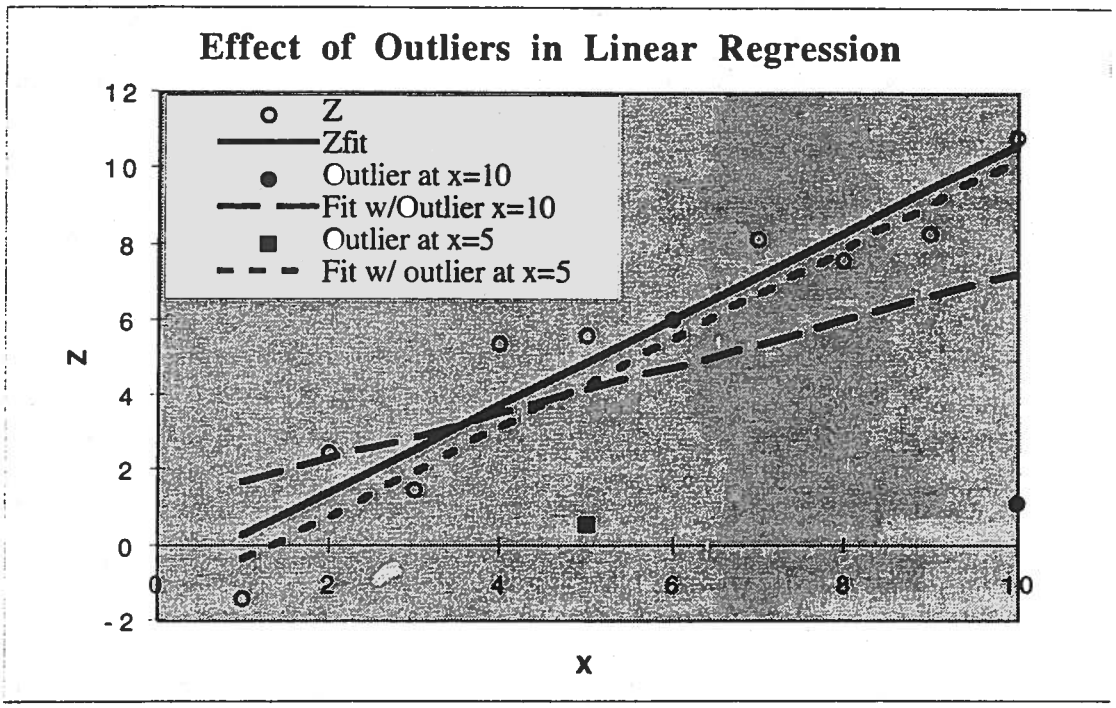
$$\text{MSE} = \text{Bias}^2\,(\text{B}^2) + \text{Variance (V)}$$

$$\sum_{i=1}^{n} e_i^2/\text{n} \;=\; \bar{e}^2 \;+\; \sum_{i=1}^{n} (e_i - \bar{e})^2 / n$$

$$\bar{e} = \sum_{i=1}^{n} e_i / n$$

- If the method is designed to be unbiased, then the model fitting procedure seeks to minimize the error spread or variance.
- Since we never know the true model, we may not know the real bias or the variance of the "noise".
- Including or dropping a data point may affect the fitted model - change the weights and the MSE.
- The "influence" a data point has on the fit can depend on the relative location of the sample point.

7. A good fit to the data does not imply good predictive performance.
   - One can drive the MSE to 0, but have no predictive ability.
   - Cross validated (leave one out and predict) mean squared error is a better measure of predictive skill.



Data is generated from $Z = x+e$ where $e \sim N(0,1)$.

Regressions for the original data and for 2 cases with outliers (original value at that point/10) are done.

Case	Intercept	Slope	MSE	R^2
Original Data	-0.90	1.15	1.14	0.90
Outlier @ x=10	1.05	0.62	6.9	0.23
Outlier @ x=5	-1.57	1.18	2.65	0.79

An outlier at the edge of the data affects both intercept and slope while in the middle only the intercept is affected. MSE ?

- "Influence" is related to relative sampling location ?
- Outlier or Wrong Model ?

Methods for Spatial Estimation

- Trend Surface Estimation
  - Polynomial Regression**
  - Weighted Moving Average**
    - Equal Weights in Region of Influence**
    - Inverse Distance Weights**
    - Other Distance Weights**
    - Natural Neighbor Interpolation
    - Locally Weighted Polynomial Regression**
  - Splines
    - Thin Plate Smoothing Splines
    - Regression Splines
  - Neural Networks
    - Radial Basis Functions, Kernels or Splines

- Geostatistical Framework
  - Ordinary Kriging**
  - Universal Kriging
  - Moving Neighborhood Kriging
  - Indicator Kriging*
  - Block Kriging
  - Co-Kriging

Polynomial Regression

A polynomial of order p is used to regress the observations Z_i on the sampling location coordinates $\mathbf{x}_i$ - (x_i, y_i).

$$Z(\mathbf{x}) = a_o + a_1x + a_2y + a_3x^2 + a_4 y^2 + a_5xy + ...a_q y^p + e$$

p=0 ; $f(x) = a_o$; The average of Z is the estimate
All weights $w(x,x_i) = 1/n$

p = 1 ; $f(x) = a_o + a_1x + a_2y$; A linear regression.
Weights need to be derived and shown

p=2 and greater; Really a linear regression on "new"coordinates (e.g., new variable b = xy; $c = x^2$)

Claimed Assumptions/Justifications:

- A "deterministic" process drives the system. The observed quantity is a direct measure of the state of this system. It may be observed with error.
- A polynomial representation of some order can recover this "signal" over the entire site. Where Z varies over many magnitudes, log(Z) rather than Z may be usable (Z=0 ?).
- If confidence limits are of interest, the errors e(**x**) are assumed to be spatially uncorrelated.

Linear Regression: Weighted Average?

Initially, let us just consider linear regression in one variable to simplify the math and convey the basic ideas.

$$Z_i = a_o + a_1 x_i + e_i$$

To solve for the two coefficients a_o and a_1

$$\underset{a_o, a_1}{Min} \sum_{i=1}^{n} (Z_i - a_o - a_1 x_i)^2 = \sum_{i=1}^{n} e_i^2$$

The solution turns out to be:

$$a_o = \bar{Z} - a_1 \bar{x} \qquad a_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(Z_i - \bar{Z})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

or

$$f(x) = \sum_{i=1}^{n} w(x, x_i) Z_i \qquad w(x, x_i) = \frac{1}{n} + \frac{(x - \bar{x})(x_i - \bar{x})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

So we see that linear regression can be written as a weighted average of the data, where the weights depend only on the location of the point of estimate x, and the sample locations x_i.

What use is this ?

If multiple predictors are considered, it is useful to adopt a matrix notation.

$$\mathbf{Z} = \mathbf{X}\beta + \mathbf{e}$$

Here **Z** is a (n x 1) vector of observations, **e** is a (n x 1) error vector, **X** is a (n x m) matrix of data on m predictors (e.g., x, y, x^2, xy) and β is an (m x 1) vector of regression coefficients (the a_0, a_1, etc.).

$$\beta = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Z}$$

$$f(\mathbf{X}) = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Z} = \mathbf{HZ}$$

H is called the hat or influence matrix since it records the entries $w(x_i, x_j)$.

Let us look at a numerical example that will help us understand the "outlier" example presented earlier.

The data used in that example was:

x	1	2	3	4	5	6	7	8	9	10
Z	-1.40	2.45	1.45	5.38	5.60	5.99	8.10	7.54	8.24	10.8

Note that the x values are equally spaced. We will treat each one of these points as a point of estimate and compute the weights associated with each data point to provide this estimate. This is the matrix **H**.

Weight Matrix H for the example data

x	1	2	3	4	5	6	7	8	9	10
1	.35	.29	.24	.18	.13	.07	.02	-.04	-.09	-.15
2	.29	.25	.21	.16	.12	.08	.04	-.01	-.05	-.09
3	.24	.21	.18	.15	.12	.08	.05	.02	-.01	-.04
4	.18	.16	.15	.13	.11	.09	.07	.05	.04	.02
5	.13	.12	.12	.11	.10	.10	.09	.08	.08	.07
6	.07	.08	.08	.09	.10	.10	.11	.12	.12	.13
7	.02	.04	.05	.07	.09	.11	.13	.15	.16	.18
8	-.04	-.01	.02	.05	.08	.12	.15	.18	.21	.24
9	-.09	-.05	-.01	.04	.08	.12	.16	.21	.25	.29
10	-.15	-.09	-.04	.02	.07	.13	.18	.24	.29	.35

• The weights applied to the Z corresponding to estimate at x=1 are shown in row 1; for the estimate at x=10 in row 10.

• The weights in any row add to 1.

• The estimate at x=1 is given as

f(x=1) = -1.4*.35 + .29*2.45 + .24*1.45 + .18*5.38 + .13*5.6 + .07*5.99 + .02*8.1 -.04*7.54 -.09*8.24 -.15*10.8 = 0.183

• If we were to take a simple average of all the data the weights would be 1/10. This is close to the weights for estimating near the middle (x=5 or 6). Effect of an outlier in the middle ?

• For the end points (x=1 or 10), the max weight is at the point itself, and then there is a large (>for intermediate points) negative weight at the other end. Effect of an outlier at ends ?

• The end points influence the fit more than the middle points. The method is ***global*** - the weights don't go to 0 away from x.

• The diagonal weights give the fraction of the information at each point coming from that point. So, if $w(x_i,x_i)$ is 1, the weight at all other points will be zero, and the estimate at that point will just be the original data value of Z_i. This will give us a great fit to the data, but no predictive ability. Why ?

• A predictive MSE is estimated by dropping each observation in turn, predicting at that point, and computing the average of the squares of the predictive errors. The predictive MSE can be estimated without going through this tedium by

$$\text{PMSE}= \frac{1}{n}\sum_{i=1}^{n}\left(\frac{e_i}{1-w(x_i,x_i)}\right)^2$$

• If we add the diagonal weights (the weight ascribed to the point for estimating at the point), we get 2 for this example. Recall that we fit 2 coefficients (a_0 and a_1). In general we will get this sum to equal m, the total number of coefficients fit. Then one could approximate the PMSE by the Generalized Cross Validation (GCV) Score

$$GCV=\sum_{i=1}^{n}\frac{e_i^2/n}{\left(1-\sum_{i=1}^{n} w(x_i,x_i)/n\right)^2}=\frac{\sum_{i=1}^{n} e_i^2/n}{(1-m/n)^2}$$

• ***We can use GCV to compare different methods.***

Some Comments on Polynomial Regression

• A low order polynomial (p=0 or 1) will invariably miss a lot of the local structure (e.g., "extrapolation" example, even with properly sampled data) in the field observations. The linear fit presumes that the underlying function f(**x**) has 0 curvature and hence is inadequate for representing situations where the data attributes change rapidly and non-monotonically.

• For clustered data, the sparse sampling locations far from the cluster will have a dominating influence on the *entire* fitted surface including within the cluster. A bad entry at such a location will have a significant impact. On the other hand, the linear regression approach deals with clustered information in a natural way - appropriate weight to the cluster.

• For Z data with a large range, a log transform is useful. However, this can lead to a biased estimate on back-transforming and is difficult to do if Z=0 at some places.

• Constraints that Z>0 are difficult to enforce directly.

• Higher order (p=2,3,4) polynomial fits may fit the data better, but with reduced degrees of freedom, greater susceptibility to outliers and with spurious oscillations introduced.

• Local (e.g., moving window), rather than global approximations to f(**x**) may overcome some of the problems faced by higher order fitting, and allow a more flexible representation than low order polynomial fits.

Weighted Moving Average Methods

Concept:

Recover the estimate as a weighted average of data within some *local* neighborhood of the point of estimate.

General Form:

$$f(\mathbf{x}) = \sum_{i=1}^{n} \frac{K((\mathbf{x}-\mathbf{x}_i)/h)}{\sum_{i=1}^{n} K((\mathbf{x}-\mathbf{x}_i)/h)} Z_i$$

where

K(u) is a kernel or weight function and h is a bandwidth or radius of "influence", and $u = (\mathbf{x}-\mathbf{x}_i)/h$ is a fractional distance from the point of estimate to its neighborhood limit.

• The denominator ensures that the resulting weights sum to 1.

Typical kernels:

Uniform: K(u) = 1 if |u|<1; 0 else Simple moving Average

Normal: $K(u) = e^{-u^2/2}/\sqrt{2\pi}$

Quadratic: $K(u) = 0.75\ (1-u^2)$

Nearest Neighbor: $h=d_k$, the distance to the k^{th} nearest neighbor of **x**. K(u) = Uniform, Normal etc.

- Local neighborhood size is larger where data are sparse

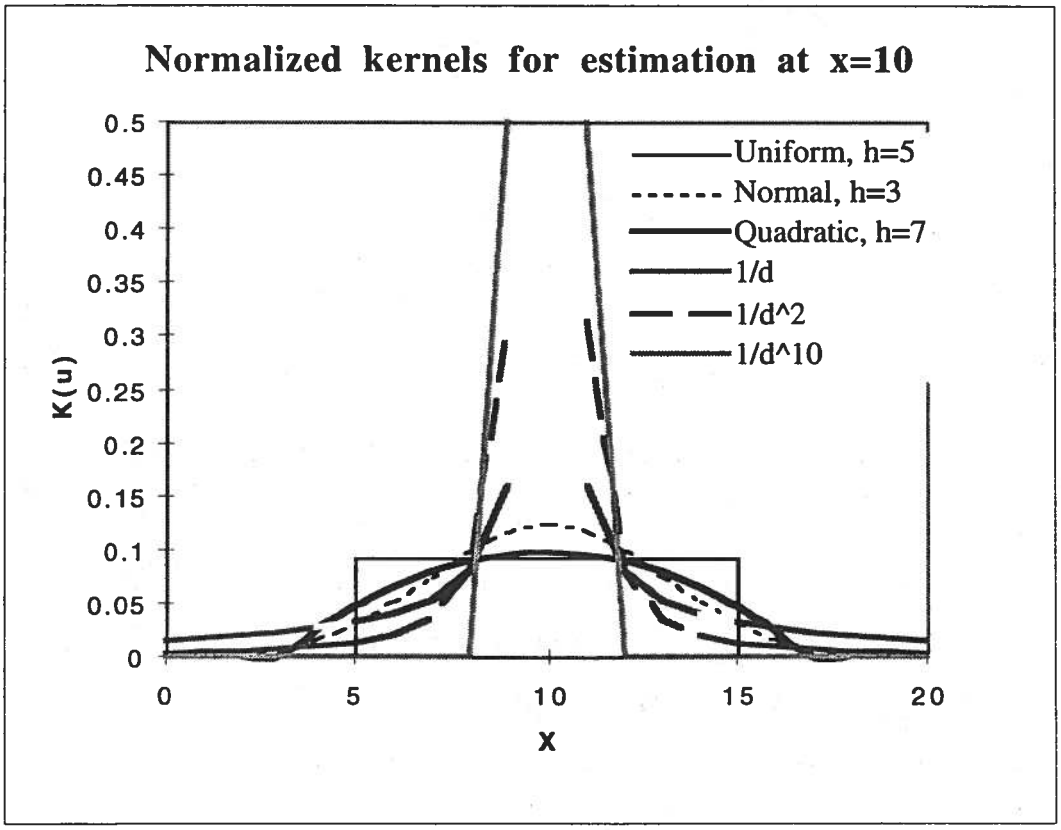
Inverse distance : $K(u) = 1/d^{\alpha}$; h=1, α=0,1,2, ...or ∞

- Region of influence controlled by α rather than h.

α=0 global average; α=2 Euclidean distance; $\alpha \to \infty$ very local

Design Issues:

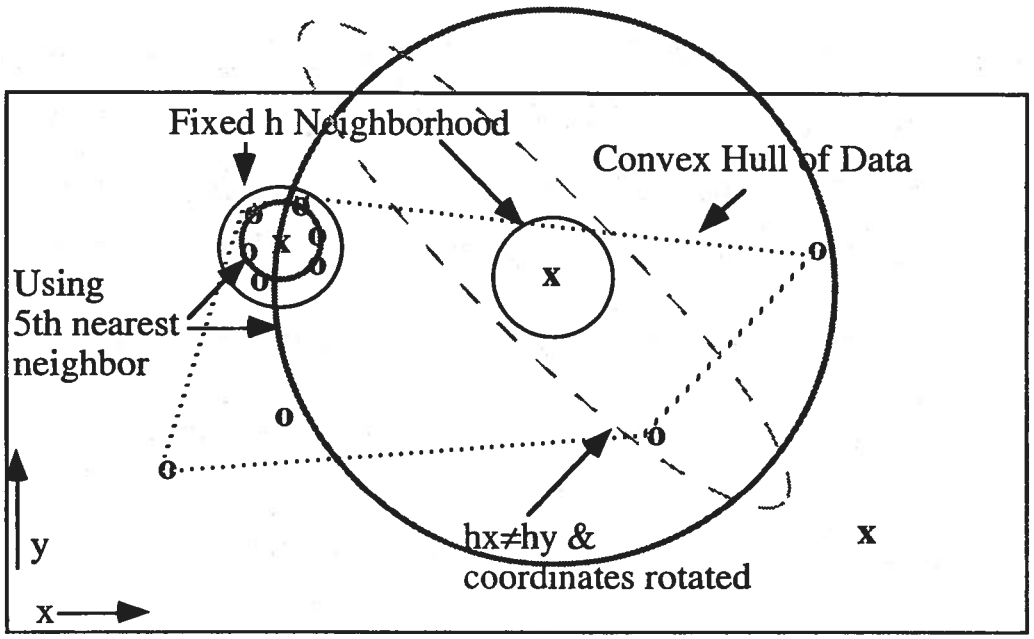
- **Choose Kernel:**
  - Typical kernels all give strictly positive weights that decrease monotonically from the point of estimate.
  - In terms of MSE performance, the choice of Kernel is secondary to the choice of size (h) of the neighborhood.



- Weight at x=10 (a data point) with inverse distance methods is ∞ - so the data point is reproduced exactly. α is equivalent to h.

- Adjust h with a new kernel to get about the same answer.

- **Choose Bandwidth or Averaging Neighborhood**
  - Fixed or Variable h ?
  - Symmetric or asymmetric neighborhood



o = Sampling Location **x = Point of Estimate**

- Using a fixed h neighborhood may fail to reach any data at some estimation points.Using k nearest neighbors to define the neighborhood always gets k points , but the associated h may be too big. Use a different kernel ? Is a simple weighted average adequate ?
- Asymmetric neighborhoods with rotation are possible.
- For a given K(u), as h increases, bias (Average error) can increase, but variability (var(error)) may decrease.
- GCV can be used to choose K(u) and h.
- Typical: Specify K(u), optimize h or k or α.

Let's continue our example with 10 data points to compare linear regression to moving averages with the uniform kernel. Consider 3, 5 and 7 point moving averages, i.e., h=1, 2 and 3 respectively. We'll compare everything only where all the estimators are readily defined (i.e. x=4,5,6,7).

3 point moving average at x=4

$$f(x=4) = 1/3(Z_3+Z_4+Z_5)$$

x	4	5	6	7	MSE	GCV
Linear reg.	3.69	4.84	5.99	7.13		
Linear reg error	1.69	0.76	0.01	0.97	1.09	1.35
3pt. Est.	4.14	5.66	6.57	7.21		
3pt error	1.24	-0.06	-0.57	0.89	0.67	1.50
5pt. Est.	4.18	5.31	6.53	7.10		
5pt. error	1.21	0.30	-0.53	1.01	0.71	1.11
7pt. Est.	3.94	5.22	6.05	7.38		
7pt. error	1.44	0.39	-0.05	0.72	0.69	0.94

So based on MSE we pick the 3 point moving average and based on GCV we pick the 7 point moving average.

Actual Error^2: (computed using true f(x) rather than Z)

Model	x=4	5	6	7	TMSE
Linear Reg.	0.10	0.03	0.00	0.02	0.14
3pt MA	0.02	0.44	0.32	0.05	0.21
5pt MA	0.03	0.09	0.28	0.01	0.10
7pt MA	0.00	0.05	0.00	0.15	0.05

Comments on Weighted Moving Average Methods

• Desirable where underlying function is complex (mean and variance of Z change over the domain).

• Not meaningful outside the convex hull of the data.

• Will do a better job of handling sudden real jumps in the data than other methods. Are susceptible to outliers only in the neighborhood of the outlier. Performance depends on sampling density and location of estimation point. Global performance may be worse than a global method, but local is usually better.

• Fixed radius (h) methods lead to problems with sparse and clustered data, since no data may be found within distance h of the estimation point, or there may be no good h value to use.

• Nearest neighbor methods overcome the above problem, but the estimates may not be meaningful if all or most of the k nearest neighbors are far away.

• Data within a cluster may have little independent information, but get counted as k neighbors or dominate the estimate in the neighborhood defined by h. This gives a false sense of the actual degrees of freedom, and hence of GCV or error variance.

• GCV or related measures can be used to choose the h, k, the weight function and compare with a global model such as polynomial regression.

• Correspond to locally fitting a polynomial of order 0. Can we do better ?

Locally Weighted Polynomial Regression

Motivation:

- Seek better approximation to f(**x**) than provided by moving averages.
- Adapt linear regression machinery for extrapolation and treatment of clusters, and for confidence intervals.

Approach:

- Choose k nearest neighbors of the point of estimate, **x**
  - distance to neighbor i, $d_i = |x-x_i|+|y-y_i|$ or $(x-x_i)^2+(y-y_i)^2$
- Specify a distance based weight function
  - $K(u_i) = 1/k$ (Uniform) or $\frac{(1-u_i^2)}{\sum_{i=1}^{k}(1-u_i^2)}$ (Bisquare)

  where $u_i = d_i/d_k$ = relative distance in the neighborhood

- Fit a Low order (p=1 or 2) polynomial

$$\underset{\beta}{Min}\ (\mathbf{Z}-\mathbf{X}\beta)^T\mathbf{K}(\mathbf{Z}-\mathbf{X}\beta) = \sum_{i=1}^{k} K(u_i)e_i^2$$

$$\beta = (\mathbf{X}^T\mathbf{K}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{K}\mathbf{Z}$$

- Evaluate the locally fitted polynomial at the point of estimate.

$$f(\mathbf{x}) = \mathbf{x}\beta$$

- The order of the polynomial and the number of neighbors k, for a specified weight function K, are chosen by GCV.

- Robust regression techniques are used to reduce outlier effects.

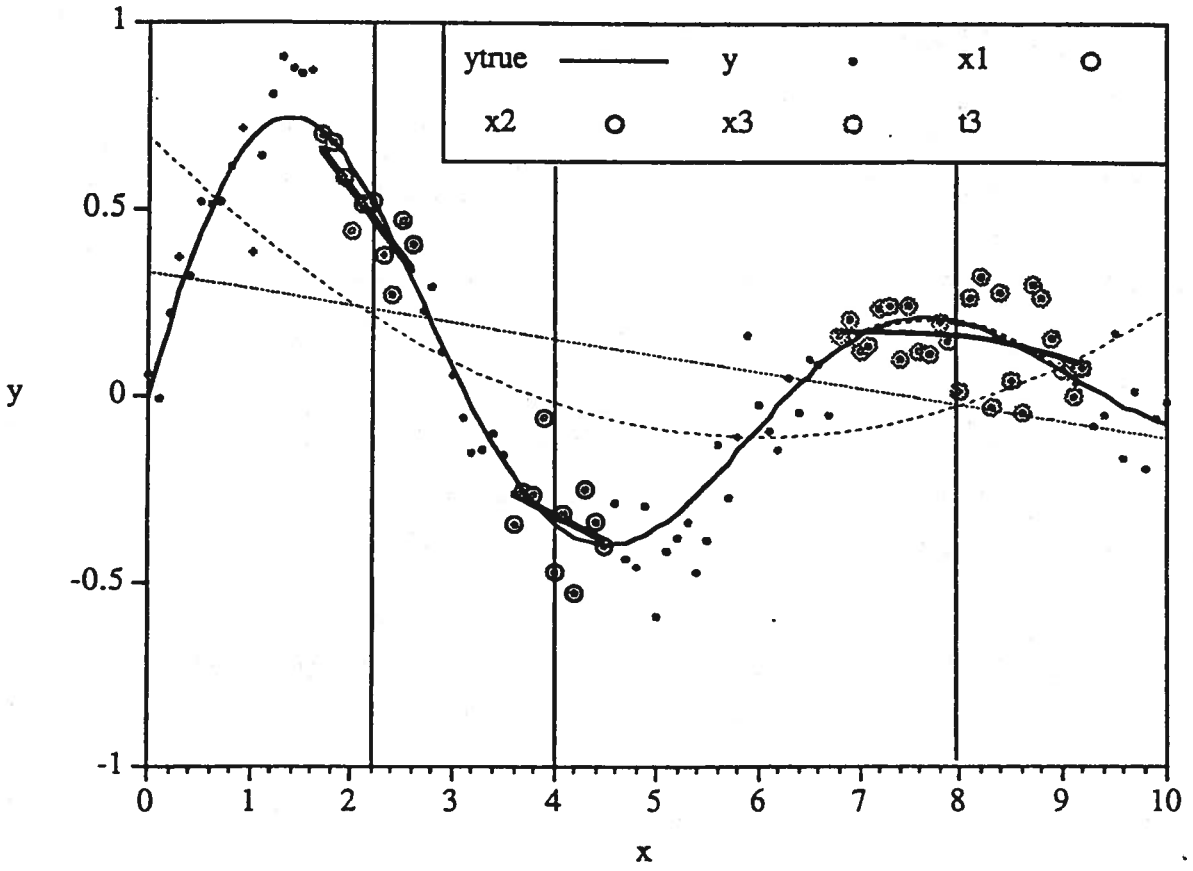


Illustration of local linear and local quadratic regression, with the weights $w_{ii} = 1/k$.

$y = \text{Sin}(x)e^{-0.2x} + N(0,0.1)$.

The true function is the solid line.

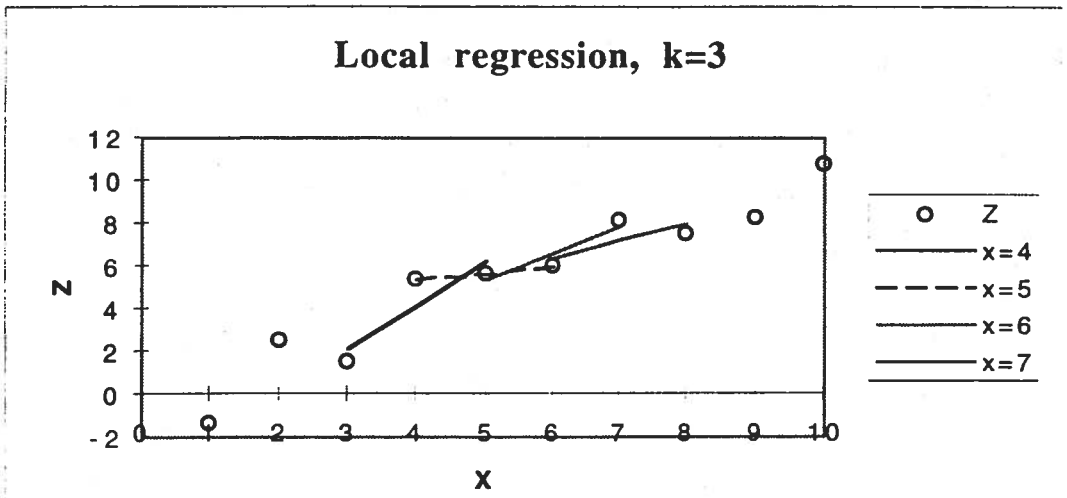
The thin dotted line is a linear regression through the full data.

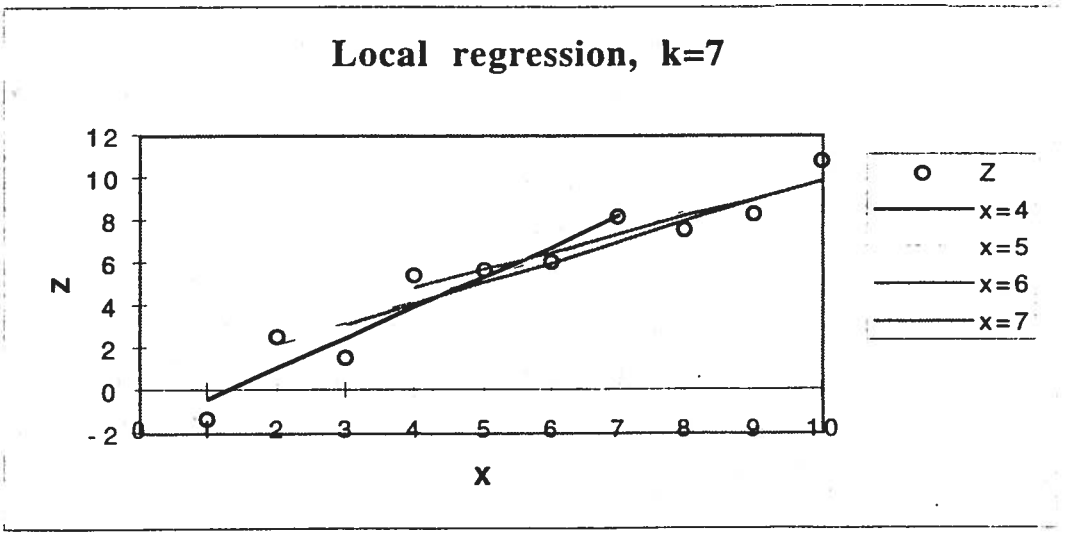
The thin dashed line is a quadratic fit through the data.

Estimates are considered at 3 points, x=2.2, 4 and 8.

Local linear fits with 10 neighbors are used at the first two points, and a locally quadratic fit with 25 neighbors is used at x=8.

Revisit the running example using k=3, 7 and a Uniform kernel





	MSE	GCV	TMSE
3 point reg	0.66	1.48	0.20
7 point reg	0.69	0.94	0.05

So GCV indicates that we pick the 7 pt linear regression or 7 pt Moving Average.
They also have the same GCV and True MSE.
Why does this work out so ?

Comments on Trend Surface Estimation methods

• The Goal is to recover a surface from the raw data. Different degrees of smoothing are used to recover this surface.

- A global average or linear regression smooths the most.
- Higher order polynomial regressors can reproduce more features, but may give rise to spurious oscillations due to high sensitivity to outliers.

• Local techniques - weighted moving averages or moving regressions adapt better to local variations and features.

- The underlying data can be quite heterogeneous. As f"(**x**) increases, reduce k and increase p to control bias.
- Variability increases as k <n and the number of parameters increases with p. So there is a bias-variance trade-off in choosing k,p. Global MSE may be worse but local MSE may be better in most places.
- What about the "extrapolation example", if we had the data to track the sin wave component ?

• GCV like measures provide a way to compare across different weighting schemes and parametric (global) or nonparametric (local) estimation schemes. It is important to look at predictive MSE rather than fitting MSE measures.

• Suppose you want to estimate the average contaminant concentration over an area where data are clustered and sparse. Simple Average vs. Average of Estimates on a grid ?

• Geostatistical methods were developed in response to the inability of global models to deal with local features and of the deficiency of moving averages in dealing with clustered data.