Spline Adaptation in Extended Linear Modeling

by

Mark H. Hansen

August 10, 1997

Spline Adaptation in Extended Linear Modeling

Joint work with

Charles Kooperberg (University of Washington)

Charles Stone (UC Berkeley)

Jianhua Huang (University of Pennsylvania)

Young Truong (UNC Chapel Hill)

Robert Kohn (University of New South Wales)

Various documents are available online, including a sizable discussion paper to appear in the *Annals*. Old fashioned PostScript is available at

http://cm.bell-labs.com/who/cocteau/papers

For a Java-enabled treatment of this material, complete with fancy interactive graphics, visit

http://cm.bell-labs.com/who/cocteau/java/trioApplet/

Outline

- Motivation

  Multivariate function estimation

  Adaptively chosen polynomial spline spaces

  Theoretical properties in the class of extended linear models

- Approaches to adaptation

  Stepwise addition and deletion

  Bayesian formulations

  MDL

- Example: Triogram models

Function Estimation

Familiar statistical problems requiring function estimation:

- Regression: From observations (Y_i, X_i), we want to estimate the conditional expectation $E(Y|x) = f(x)$, where $x = (x_1, \ldots, x_k)$
- Density Estimation: From observations Y_i we try to uncover the important features in the density $f(y)$

Function Estimation

Classical Parametric Framework

$$\widehat{f}(x_1, \ldots, x_k) \;=\; \widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \cdots + \widehat{\beta}_k x_k$$

A nonparametric approach removes such severe restrictions, but modeling $\widehat{f} = \widehat{f}(x_1, \ldots, x_k)$ can be difficult. This motivated the use of additive models

$$\widehat{f}(x_1, \ldots, x_k) \;=\; \widehat{f}_0 + \widehat{f}_1(x_1) + \cdots + \widehat{f}_k(x_k)$$

…possibly allowing for interactions

$$\widehat{f}(x_1, x_2) \;=\; \widehat{f}_0 + \widehat{f}_1(x_1) + \widehat{f}_2(x_2) + \widehat{f}_{12}(x_1, x_2)$$

Function Estimation

There are many procedures based on polynomial splines and their tensor products

- Regression

  TURBO (Friedman and Silverman)

  MARS (Friedman)

  Pi (Breiman)

  br() (Smith and Kohn)

  Bayesian CART/MARS (Denison et al.)

- Density estimation

  Logspline (Kooperberg and Stone)

  Salsa (Hansen and Kooperberg)

Function Estimation

In each of these cases, $\widehat{f}$ and its components are expanded in a basis. For example,

$$\widehat{f_1}(x_1) \;=\; \sum_{i=1}^{J_1} \beta_{1i} B_{1i}(x_1)$$

where the basis functions B_{1i} are determined adaptively. Think of this as selecting from the set

$$1, x_1, \ldots, x_1^k,$$

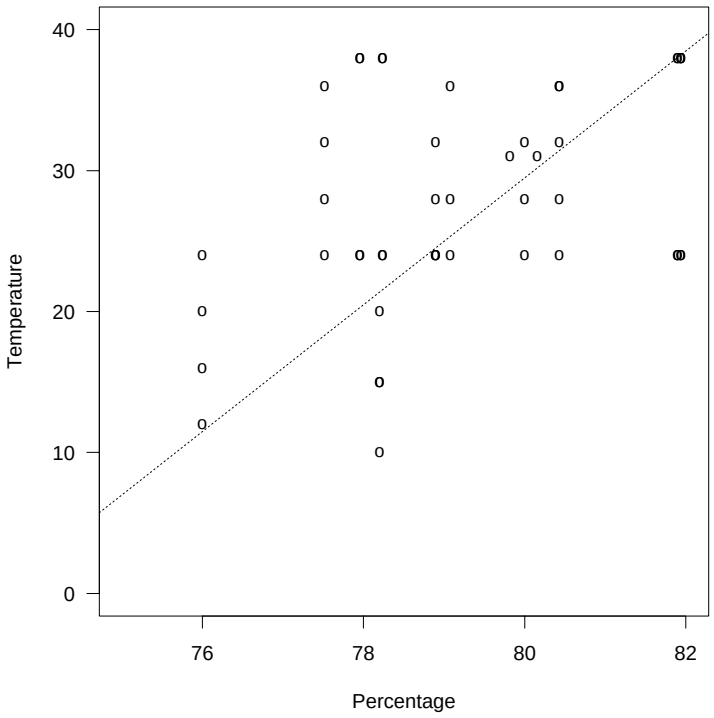
$$(x_1 - t_{11})_+^k, \ldots, (x_1 - t_{1m})_+^k$$

...but it can be much more interesting!

A Regression Surface with a Ridge

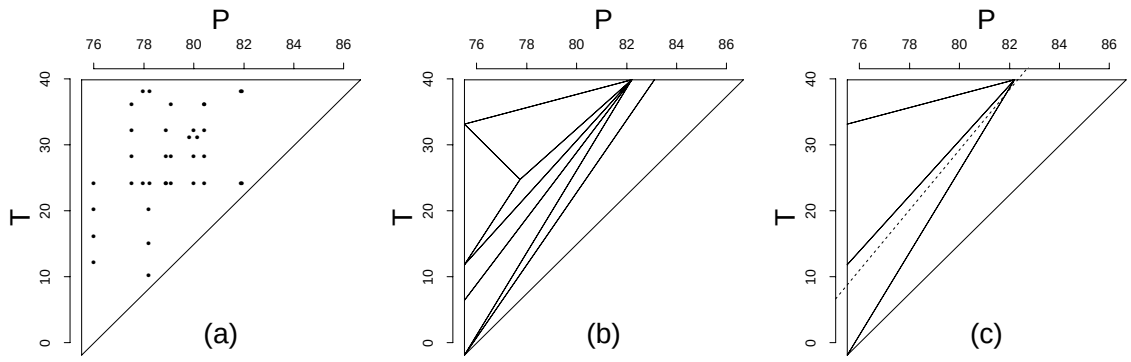
Experimental data: Cleveland and Fuentes (1996) consider an experiment aimed at improving the processing of liquid crystal mixtures.

P: percentage of liquid crystal in mixture
T: temperature (degrees Celsius)
V: voltage needed to make material clear



A Regression Surface with a Ridge

- Starting triangulation: smallest triangle containing the data, enlarged by 15%.
- Adaptation: Stepwise addition and deletion of vertices



Function Estimation

- Stone (1994) treats the case where each component is the tensor product of univariate spline functions

$$1, x_1, \ldots, x_1^k,$$

$$(x_1 - t_{11})_+^k, \ldots, (x_1 - t_{1m})_+^k$$

- Hansen (1994) and Stone et al. (1997) extend this to allow $x_i \in \mathbb{R}^{p_i}$, $1 \leq i \leq k$. In this case, the effects in the previous expansion are *multivariate splines* and their tensor products.

Finite elements

multivariate B-splines

⋮

Function Estimation

In general, to estimate a function of d variables in this way, the L_2 rate of convergence is $n^{-p/(2p+d)}$ where p is related to the smoothness of f.

By considering an unsaturated approximation, we can improve the rate $n^{-p/(2p+d^*)}$ where d^* is the largest number of terms in any single interaction component.

Functional ANOVA

An additive model

$$\widehat{f}(x_1, x_2) \;=\; \widehat{f}_0 + \widehat{f}_1(x_1) + \widehat{f}_2(x_2)$$

The unsaturated approximation

$$f^*(x_1, x_2) \;=\; f^*{}_0 + f^*{}_1(x_1) + f^*{}_2(x_2)$$

minimizes $E[Y - f^*(X_1, X_2)]^2$ over the space H of additive functions in x_1 and x_2.

Function Estimation

These results can be derived from the somewhat more informative expression (Hansen, 1994)

$$\|\widehat{f} - f^*\|^2 \;=\; O_P(\,\rho^2(\overline{\delta}) + \underline{\delta}^{-d}/n\,)$$

where $\overline{\delta}$ and $\underline{\delta}$ represent the size of the largest and smallest cells in the partition, respectively, and ρ is the approximation rate of the underlying spline space.

Huang (1997) extends these results to include general approximating spaces like trigonometric polynomials

General Rates for ELMs

Rates of convergence for the components in these functional ANOVA models can be established in the context of an *extended linear model*

- Regression
- Generalized Regression
- Hazard Regression
- Polychotomous Regression
- Censored Regression
- Density Estimation
- Conditional Density Estimation

General Adaptation in ELMs

Adaptively placing *knots* in a spline expansion can be treated (roughly) like a problem in subset selection

- Stepwise Addition: Maximize Rao Statistic
- Stepwise Deletion: Minimize Wald Statistic
- Model Selection:

  Minimize $\text{AIC}_{\alpha,\nu} = -2\widehat{l}_{\nu} + \alpha\, p_{\nu}$

Stepwise Addition: Maximize Rao Statistic

Let Q be the quadratic approximation to the log-likelihood function about a point $\beta_0 \in \mathcal{B}$

$$Q(\beta) = \ell(\beta_0) + [\nabla\ell(\widehat{\beta}_0)]^t(\beta - \beta_0) + \frac{1}{2}(\beta - \beta_0)^t H(\beta_0)(\beta - \beta_0)$$

If $H(\beta_0)$ is negative definite, the Q is uniquely maximized at the point

$$\beta_1 = \beta_0 - H^{-1}(\beta_0)\nabla\ell(\beta_0)$$

From these two expressions, we find that

$$2[Q(\beta_0) - Q(\beta_1)] = [\nabla\ell(\beta_0)]^t H^{-1}(\beta_0)\nabla\ell(\beta_0)$$

If β_0 is the MLE in a subspace $\mathcal{B}_0$, then this is the Rao statistic for testing that $\beta \in \mathcal{B}_0$.

Stepwise Deletion: Minimize Wald Statistic

Let Q be the quadratic approximation to the log-likelihood function at the MLE $\widehat{\beta} \in \mathcal{B}$ and let $\mathcal{B}_0$ be a subspace of $\mathcal{B}$ consisting of those β satisfying $A\beta = 0$, A having full rank. Then, the maximum of Q over $\mathcal{B}_0$ occurs at

$$\widehat{\beta}_0 \;=\; \widehat{\beta} - H^{-1}(\,\widehat{\beta}\,)\,A^t\,[\,A\,H^{-1}(\,\widehat{\beta}\,)\,A^t\,]^{-1}\,A\,\widehat{\beta}$$

and hence

$$2[\,Q(\,\widehat{\beta}_0\,) \;-\; Q(\,\widehat{\beta}\,)\,] \;=$$

$$(\,A\,\widehat{\beta}\,)^t\,[\,AH^{-1}(\,\widehat{\beta}\,)\,A^t\,]\,A\,\widehat{\beta}$$

which is the familiar Wald statistic for testing that $\beta \in \mathcal{B}_0$ under the assumption that $\beta \in \mathcal{B}$.

A Bayesian Alternative

Regression: $Y = f(X) + \epsilon$, $\epsilon \sim N(0, \sigma^2)$

Fix a knot sequence and consider the basis

$$1, x, \ldots, x^k, (x - t_1)_+^k, \ldots, (x - t_m)_+^k$$

Introduce a vector $\gamma = (\gamma_1, \ldots, \gamma_{m+4})$ indexing the columns of the resulting design matrix: If $\gamma_i = 0$ the coefficient β_i on the ith basis function is zero.

Smith and Kohn (1995) make computationally efficient choices for $\beta = (\beta_1, \ldots, \beta_{m+4})|\gamma, \sigma^2$ and $\sigma^2|\gamma$, and use the Gibbs sampler to simulate from the posterior distribution of γ.

A Bayesian Alternative

Regression: $Y = f(X) + \epsilon$, $\epsilon \sim N(0, \sigma^2)$

Foster and George (1996) have found that in this framework, the mode of the posterior distribution for γ corresponds to a model that minimizes an expression of the form

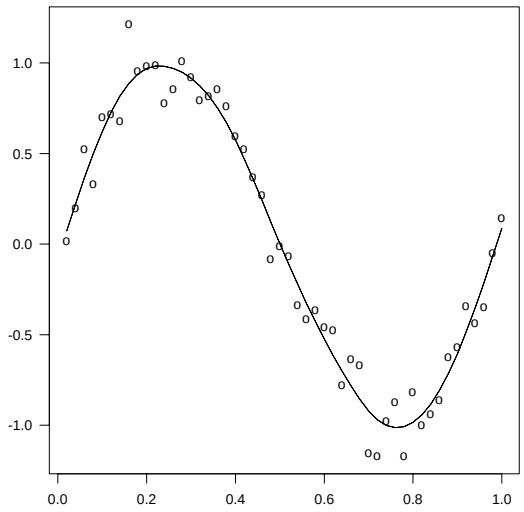
$$RSS(\gamma) + \frac{1+c}{c} \log(c+1) p(\gamma) \hat{\sigma}^2$$

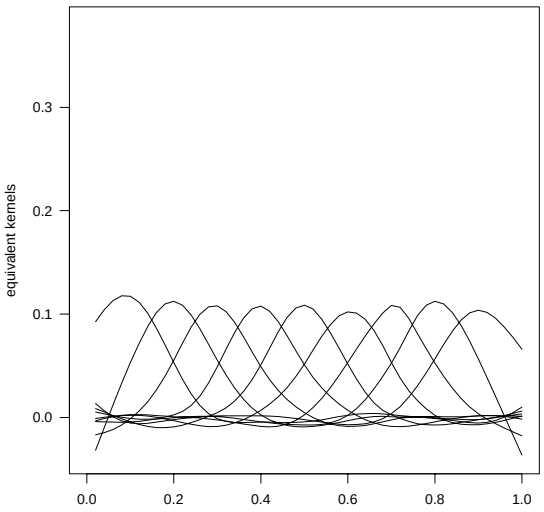
where c is a hyperparameter from the prior on β.

This means that the Gibbs sampler can be thought of as a stochastic search procedure to find the best model according to BIC

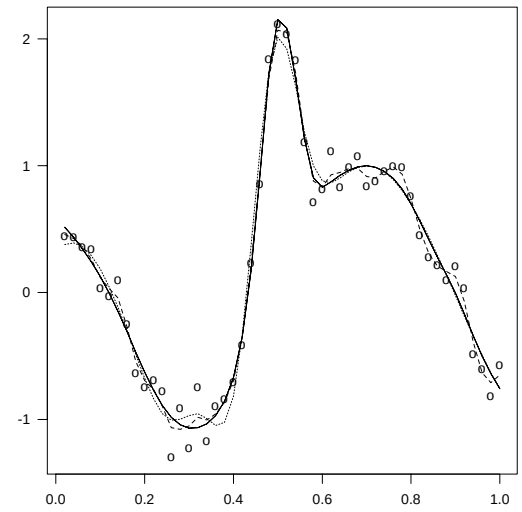
A Bayesian Alternative

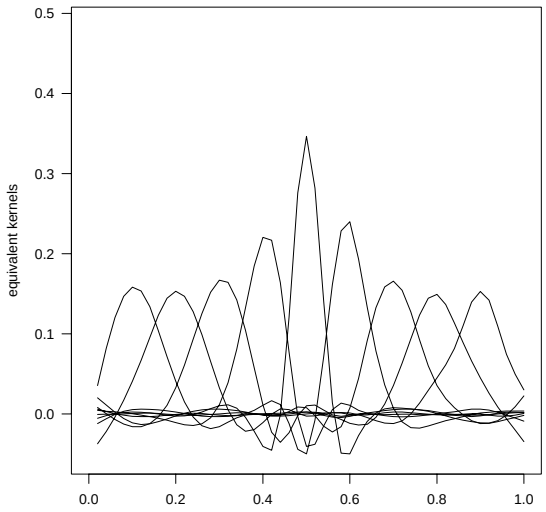
We gain insight into the behavior of the posterior mean of β by considering the *equivalent kernel* of the smoother

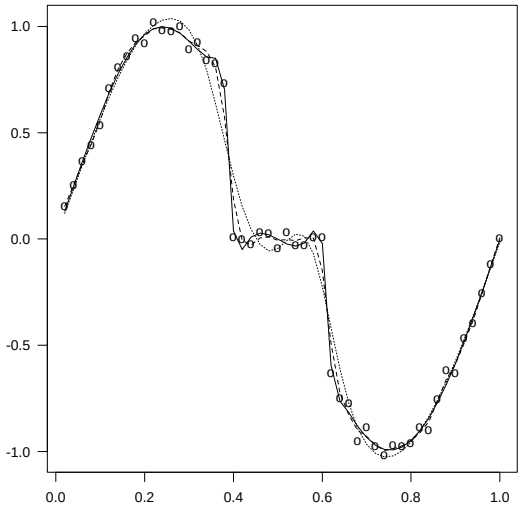


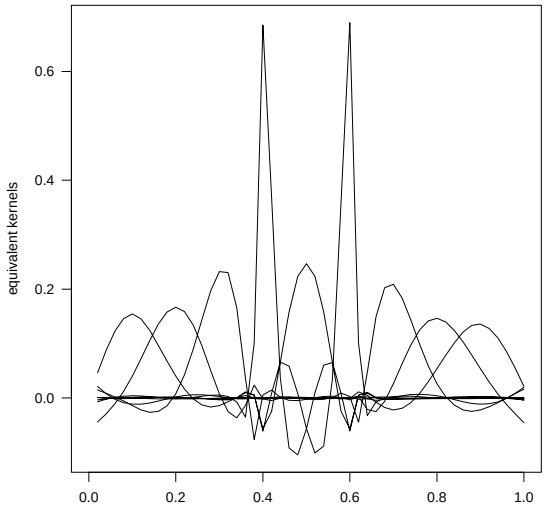


A Bayesian Alternative









An Alternative Bayesian Framework

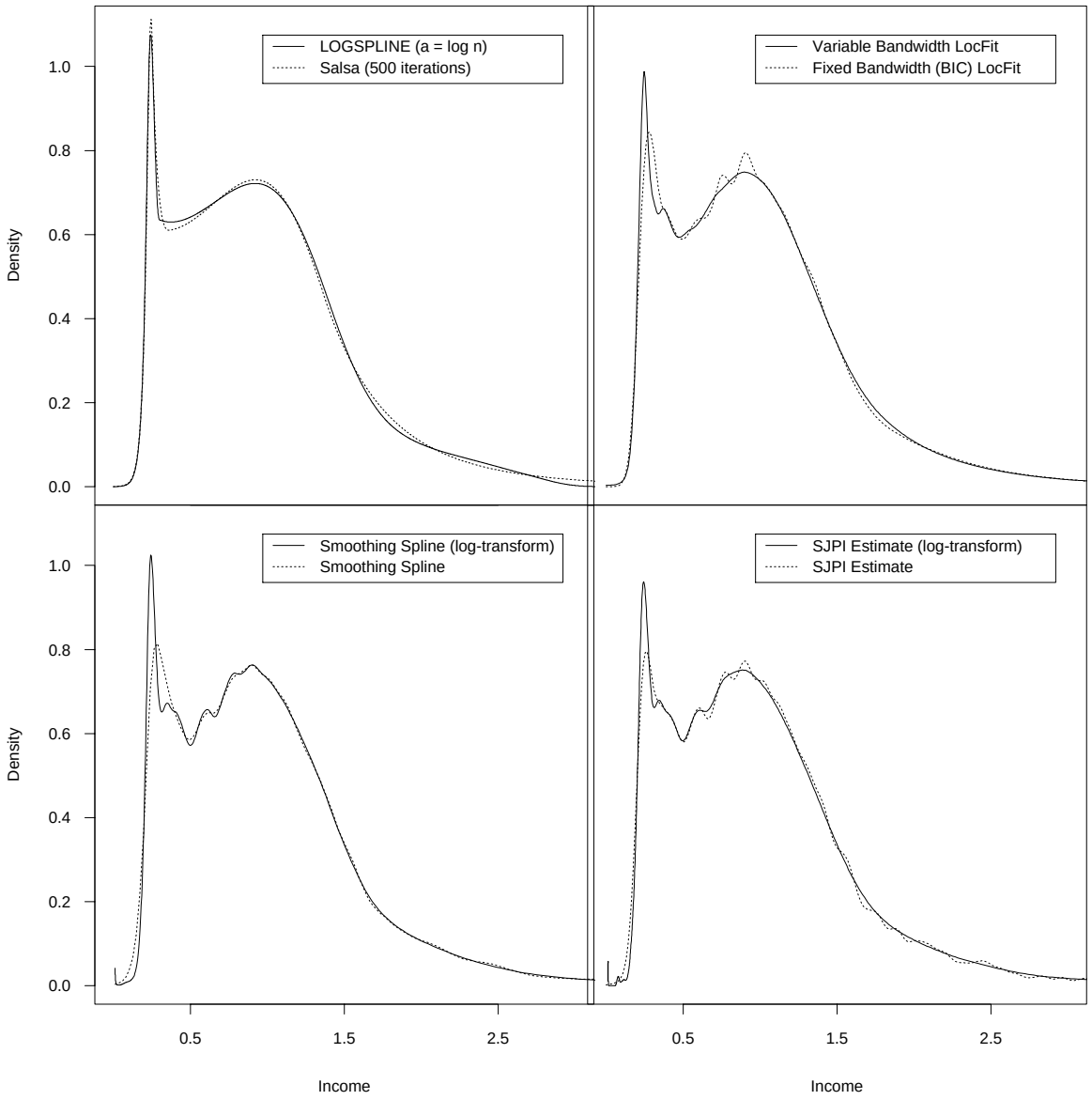
Borrowing heavily from Green (1995), Denison et al. used reversible jump MCMC to average across competing spline models.

- For computational reasons, a full Bayesian approach is not taken
- A Poisson prior with mean λ is assigned to the number of knots

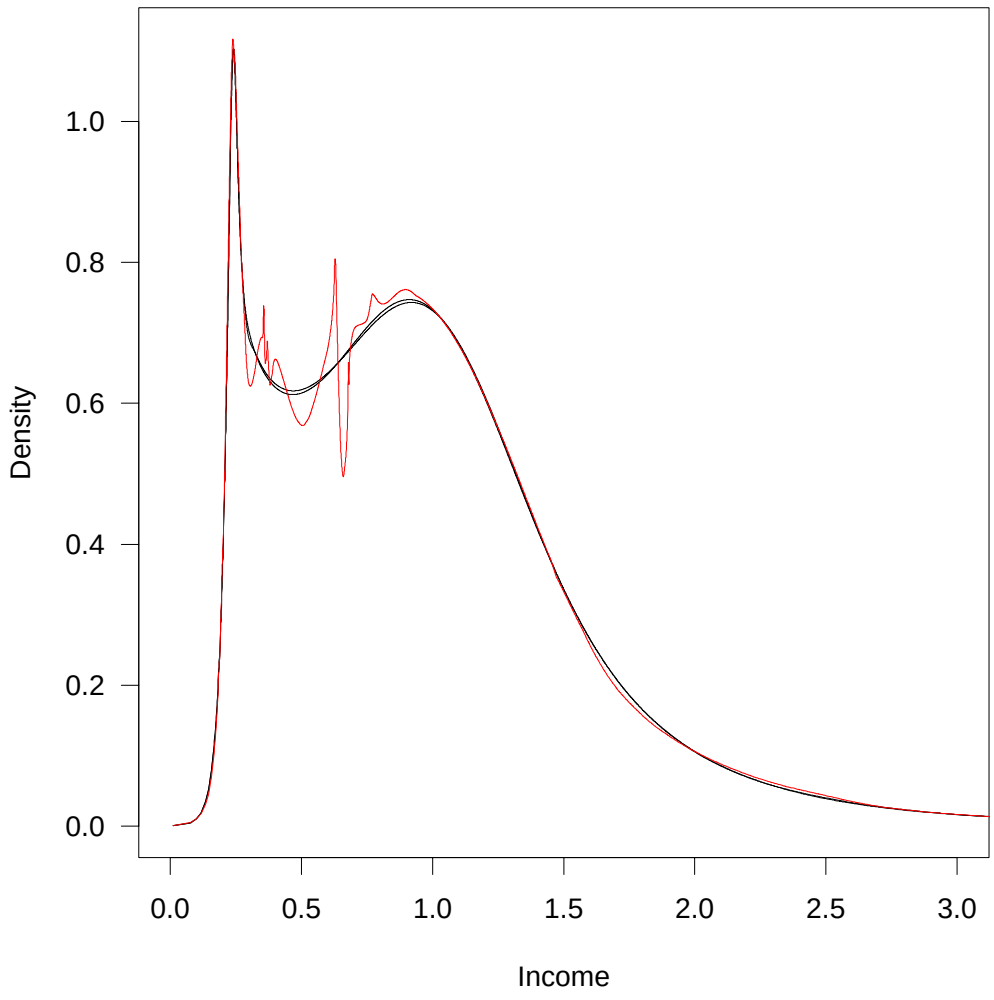
Hansen and Kooperberg (1997) apply these ideas to density estimation, replacing the Poisson prior with a geometric distribution with $p = 1 - 1/\sqrt{n}$.

With this choice, posterior model probabilities again scale like BIC

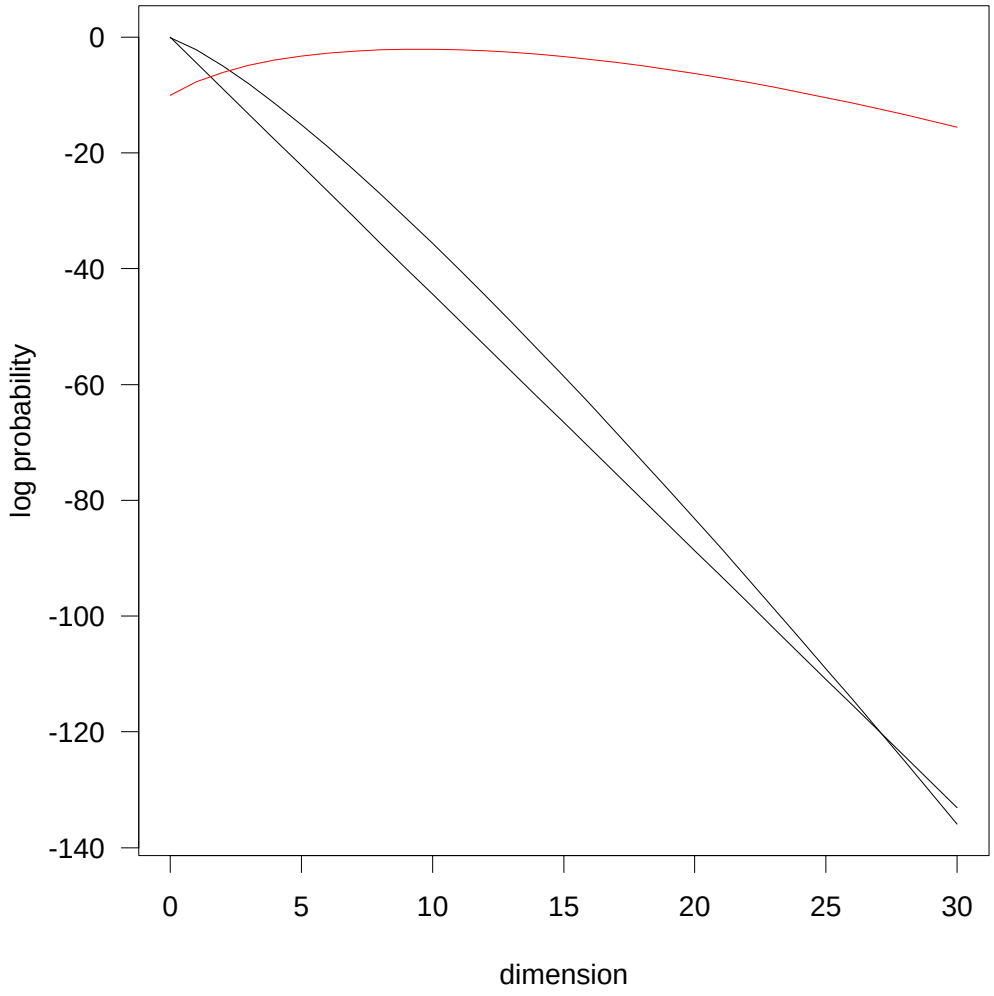
Salsa and bLogspline



Salsa and bLogspline



Salsa and bLogspline



A Hybrid Smoother

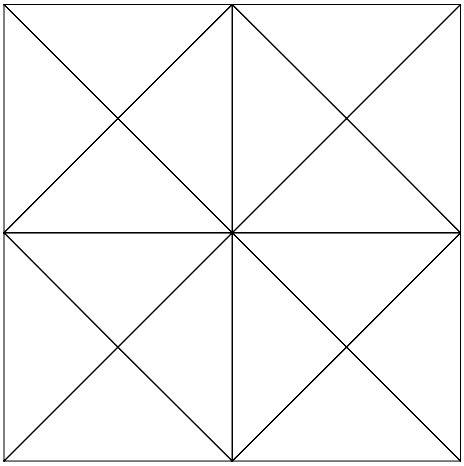
Recently, Luo and Wahba (1996) proposed a hybrid between the greedy ELM procedure and smoothing splines. The idea is to follow a stepwise addition phase (this time with RKs) by a penalized fit.

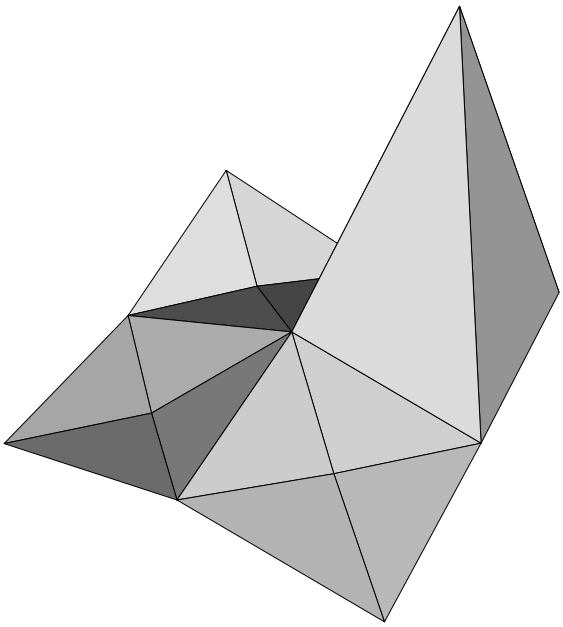
Triogram Models

- Bivariate functions
  (spatial data, two-factor interactions, …)

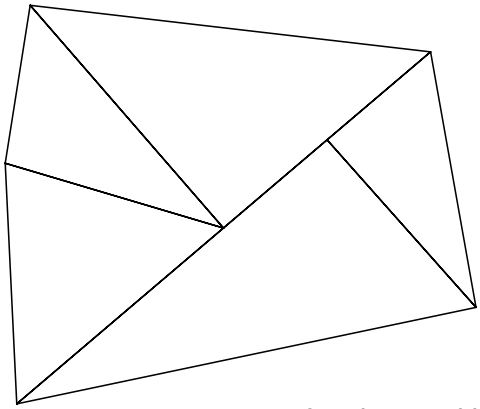
- Continuous, piecewise linear functions defined over arbitrary triangulated regions in the plane
- Adaptive, local refinements to the space based on "classical" procedures for stepwise model building

Piecewise Linear Basis Functions

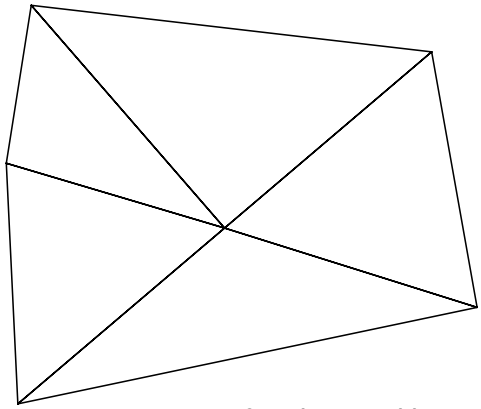




Conforming Triangulations

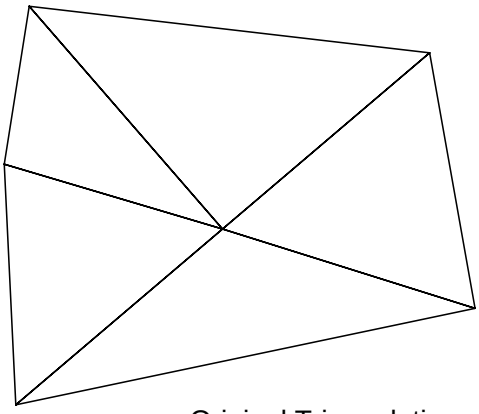


Non-Conforming Partition

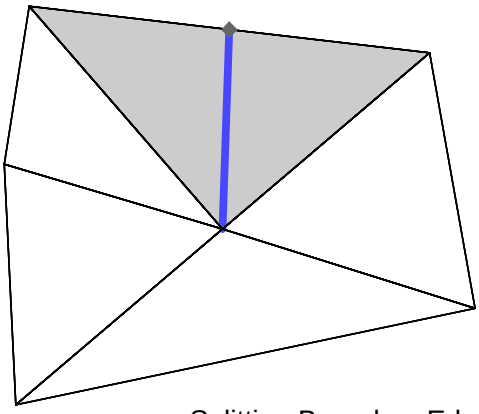


Conforming Partition

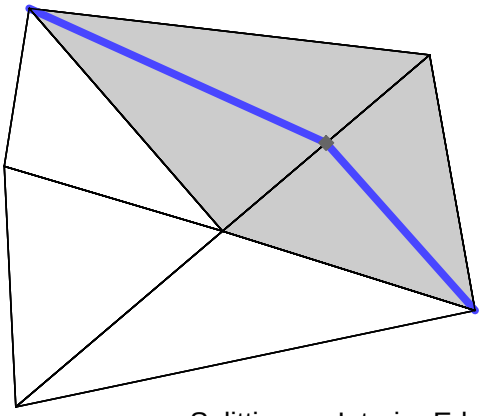
Stepwise Addition



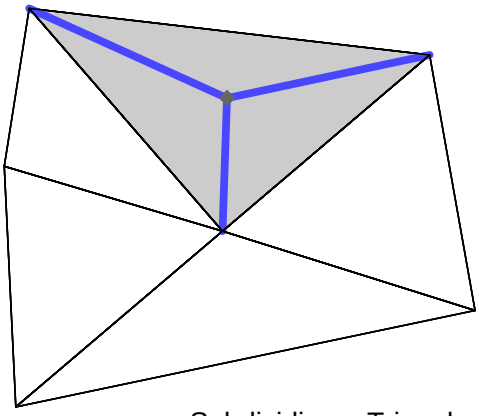
Original Triangulation



Splitting Boundary Edge

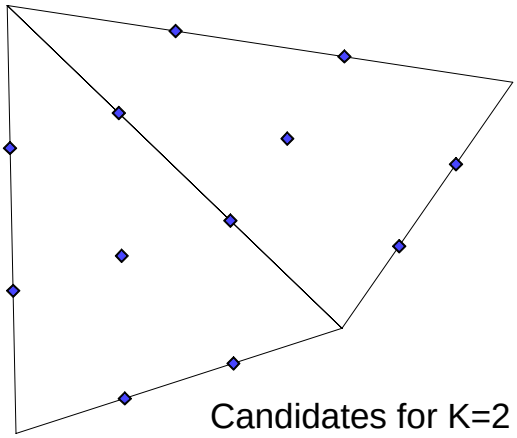


Splitting an Interior Edge

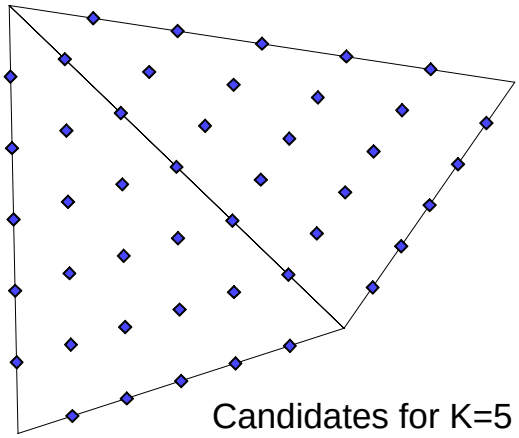


Subdividing a Triangle

Candidate Vertices



Candidates for K=2



Candidates for K=5

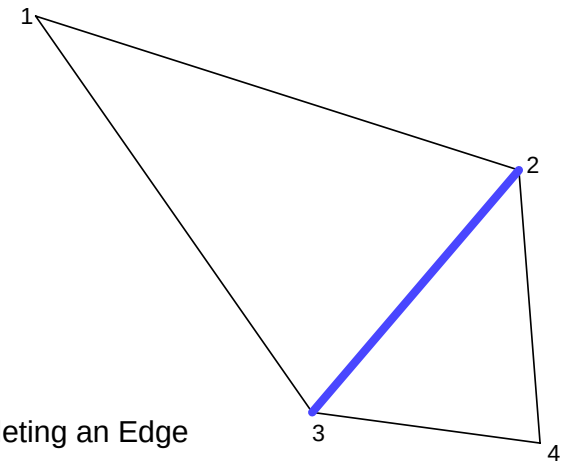
Candidate Vertices

Candidate vertices are located at the barycentric coordinates

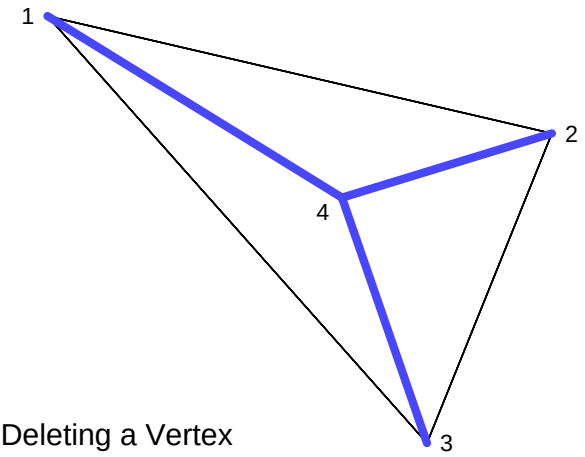
$$\left(\frac{k_1}{K+1}, \frac{k_2}{K+1}, \frac{K+1-k_1-k_2}{K+1} \right),$$

for $k_1, k_2, K \in \mathbb{N}$ and $k_1 + k_2 \leq K + 1$.

Stepwise Deletion

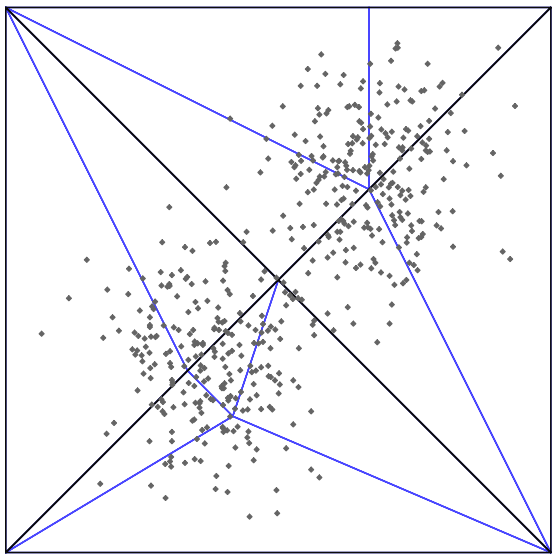


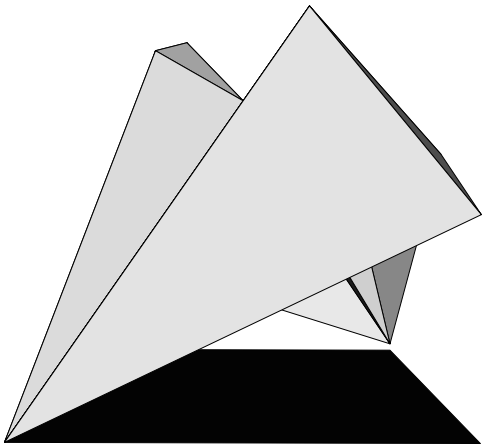
Deleting an Edge

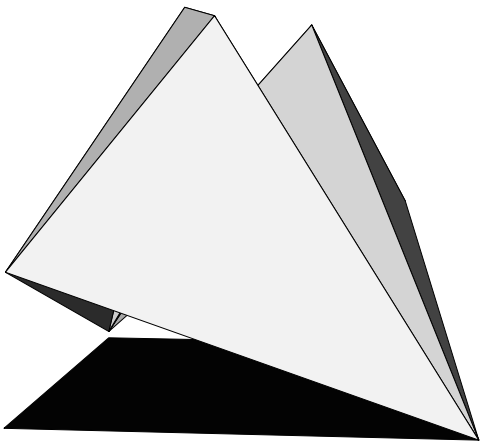


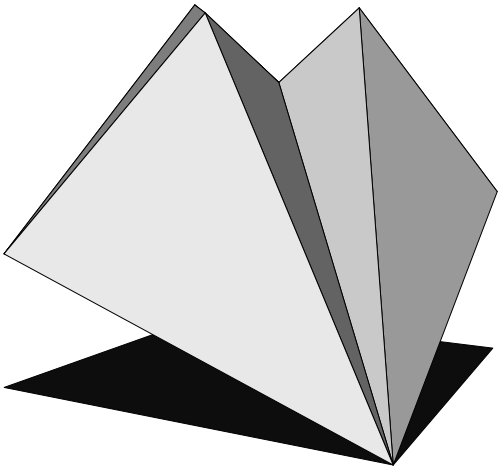
Deleting a Vertex

Simulated Example

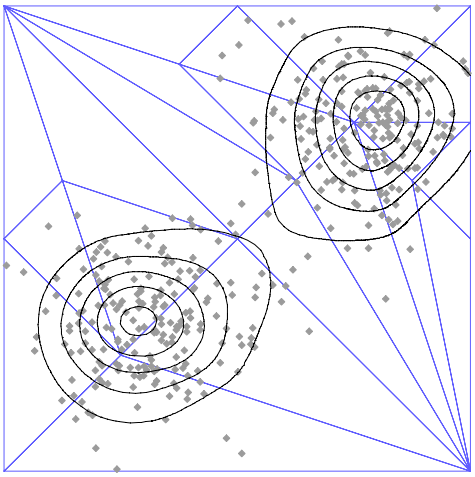


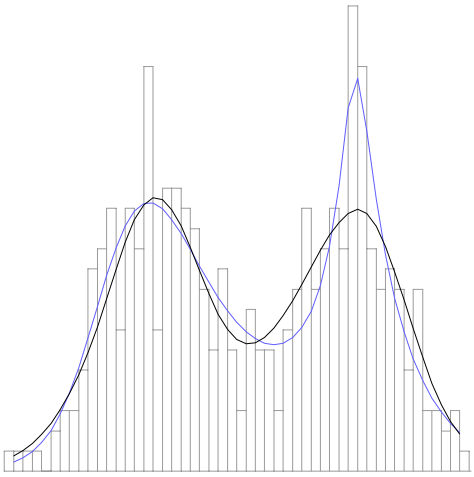


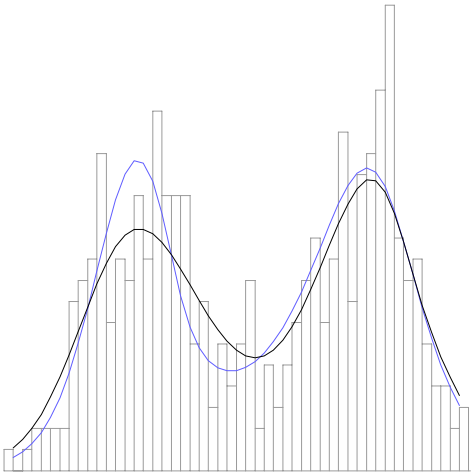


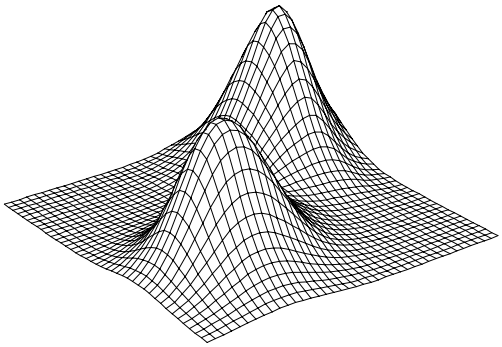


Simulated Example









Data Restricted to a Triangle

100 protein structures from the Brookhaven Protein Data Bank

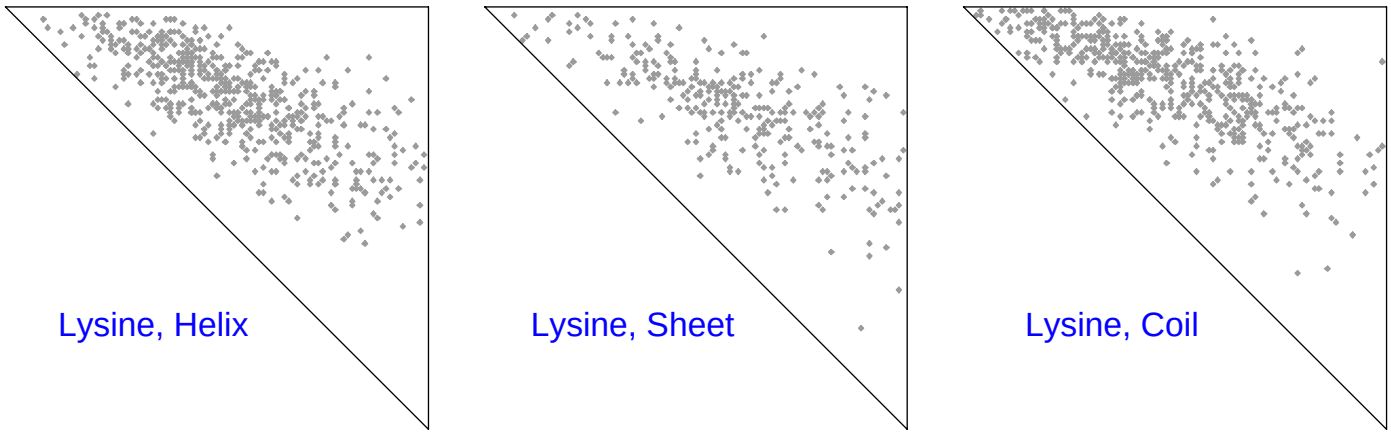
Each amino acid in each protein was assigned to one of three secondary structure classes:

helix, sheet, coil

Two measurements taken for each amino acid

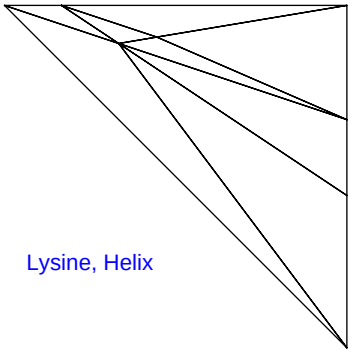
- Fraction of the amino acid side-chain area buried in the protein structure
- Fraction of the amino acid side-chain area covered by polar atoms

Protein Example: Lysine

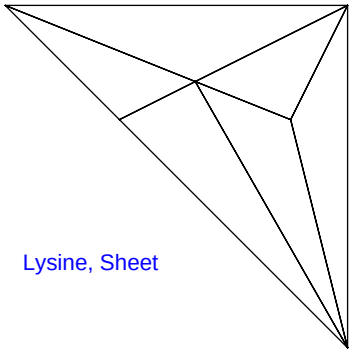


- Each set of measurements is restricted to a triangle
- We will fit separate densities to each collection of measurements

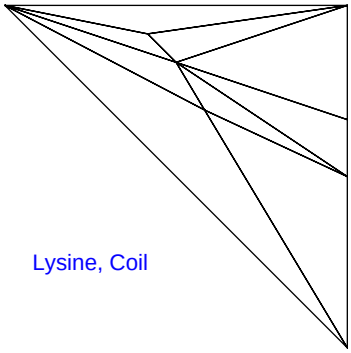
Protein Example: Lysine



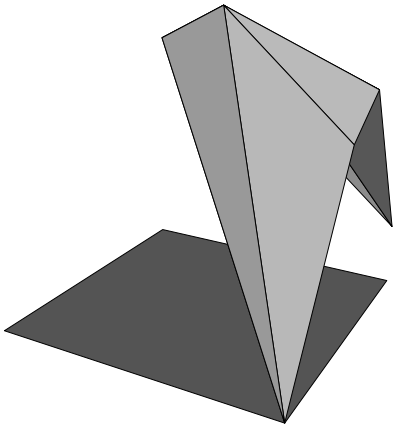
Lysine, Helix

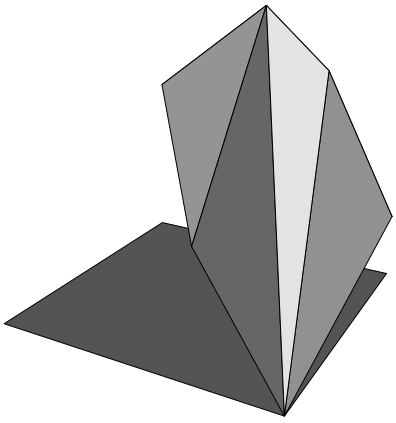


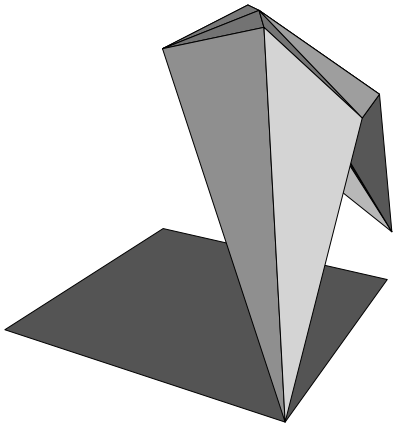
Lysine, Sheet



Lysine, Coil

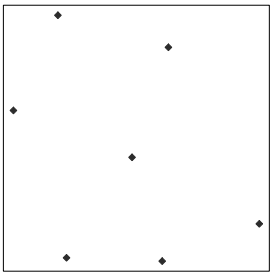




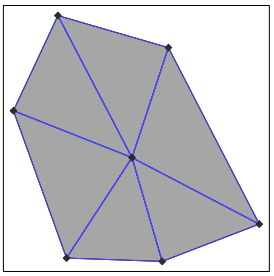


Implementation in S

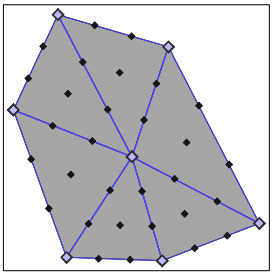
plot(x,y)



tri <– triangulate(x,y,“del”)
lines(tri)

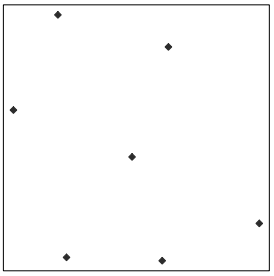


net <– bnet(tri,d=3)
points(net)

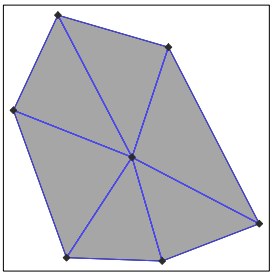


Implementation in S

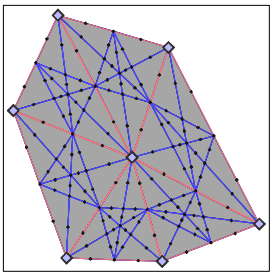
plot(x,y)



tri <– triangulate(x,y,"del")
lines(tri)



vs <– gvs(tri)
net <– as.bnet(vs)
lines(net)



Implementation in S

Given bivariate data X and a response variable Z, we have the following fitting routines

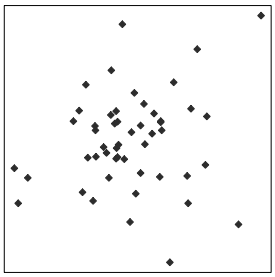
$$\text{fit} <- \text{elm}(\ Z \sim \text{triogram}(\ X, \text{obj}\ ),\ \text{method},\ \text{model},\ \text{family}\ )$$

$$\text{fit} <- \text{elm}(\ \sim \text{triogram}(\ X, \text{obj}\ ),\ \text{method}\ )$$

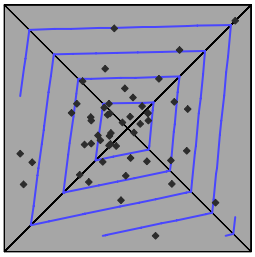
where *obj* contains starting values, and *method* specifies the type of model adaptation.

Implementation in S

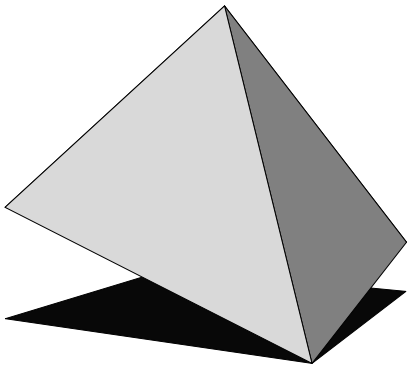
plot(x,y)



fit $<-$ elm($\sim$ triogram(x,y))
contour(fit,add=T)
lines(triangulate(fit))



geomview.triogram(fit)

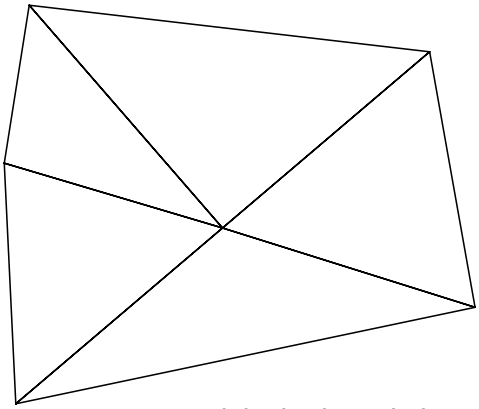


A Bayesian Alternative

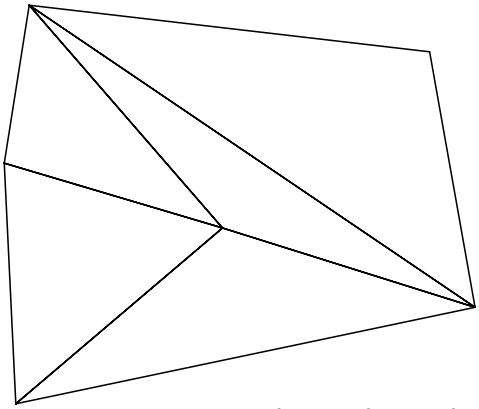
Building from the ideas of Green (1995), we have also implemented a full Bayesian Triogram procedure. Essentially, we specify a prior on triangulations and use rjMCMC to step through model space.

By averaging, our bTriograms are much smoother than their greedy counterparts.

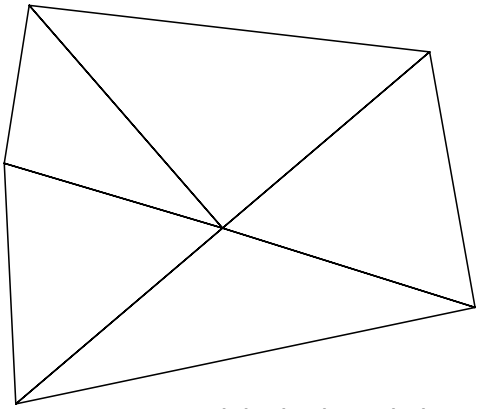
New Moves



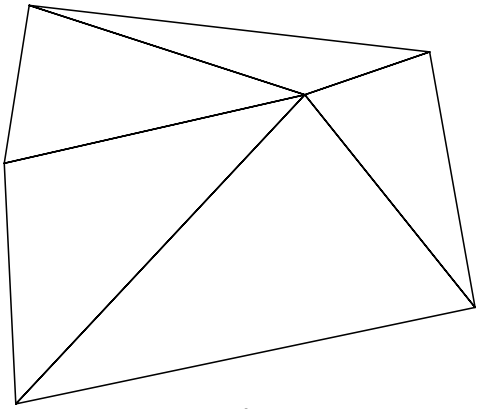
Original Triangulation



Swapping a Diagonal

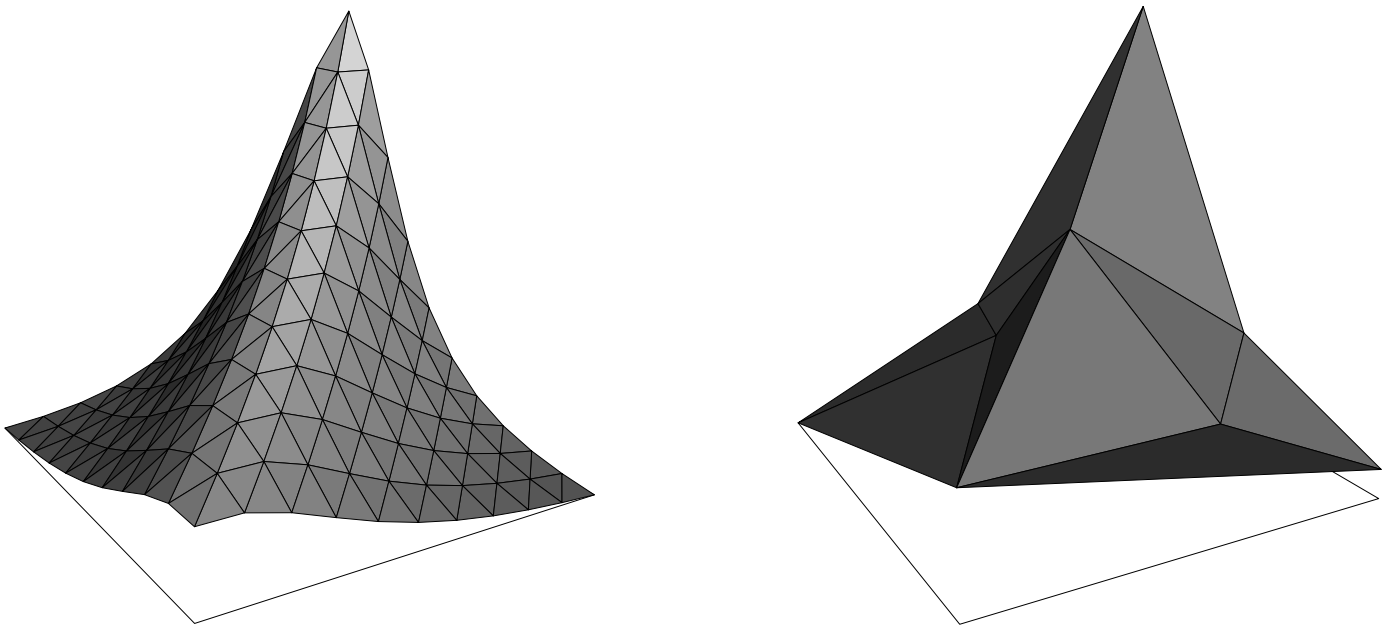


Original Triangulation



Moving a Vertex

Simulated Example

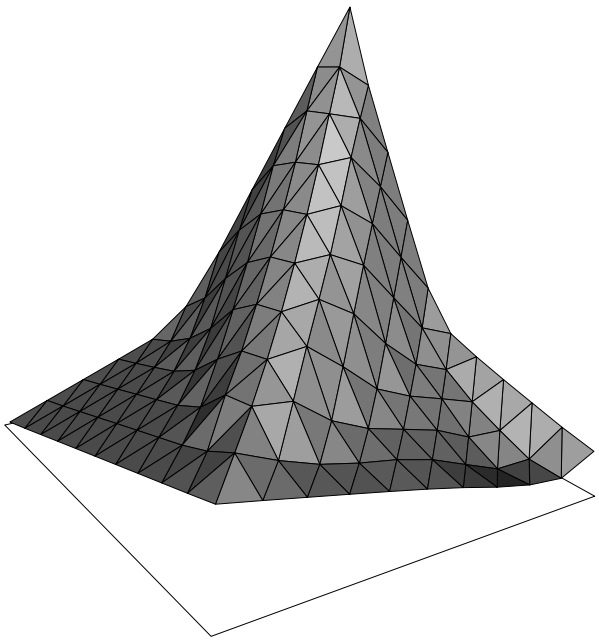


In this example, we generate 300 (essentially) random points in the unit square, evaluate the function

$$f(\mathbf{x}) = 40\exp\{8[(x_1-0.5)^2+(x_2-0.5)^2]\}/$$
$$(\exp\{8[(x_1-0.2)^2+(x_2-0.7)^2]\}+\exp\{8[(x_1-0.7)^2+(x_2-0.2)^2]\})$$

and add $N(0,1)$ noise.

Simulated Example



Ridge Example