Spline Functions – An Elegant View of Interpolation

Bruce Cohen
bic@cgl.ucsf.edu
http://www.cgl.ucsf.edu/home/bic

David Sklar
dsklar46@yahoo.com

Start with xv intensity controls

Some Goals

To present a concrete introduction to a widely used class of methods for creating continuous functions that interpolate discrete data

To apply straightforward geometric and algebraic operations with functions and their graphs that may revolutionize your views of interpolation and approximation

To present a concrete example of a general method for extending a 1-dimensional scheme into higher dimensions

To stimulate some curricular or student project ideas

To see how elementary themes can lead to some beautiful pictures and a lingering vision

Game Plan

Connecting the dots – continuous piecewise linear interpolation

- using the data to compute a linear equation for each subinterval
- using a linear combination of “Spline basis functions”

Connecting the dashes – smooth, piecewise cubic, interpolation

- using slope and function data to compute a cubic equation for each subinterval
- using a linear combination of “Cubic spline basis functions”

As time allows:

Interpolation using 2-D splines

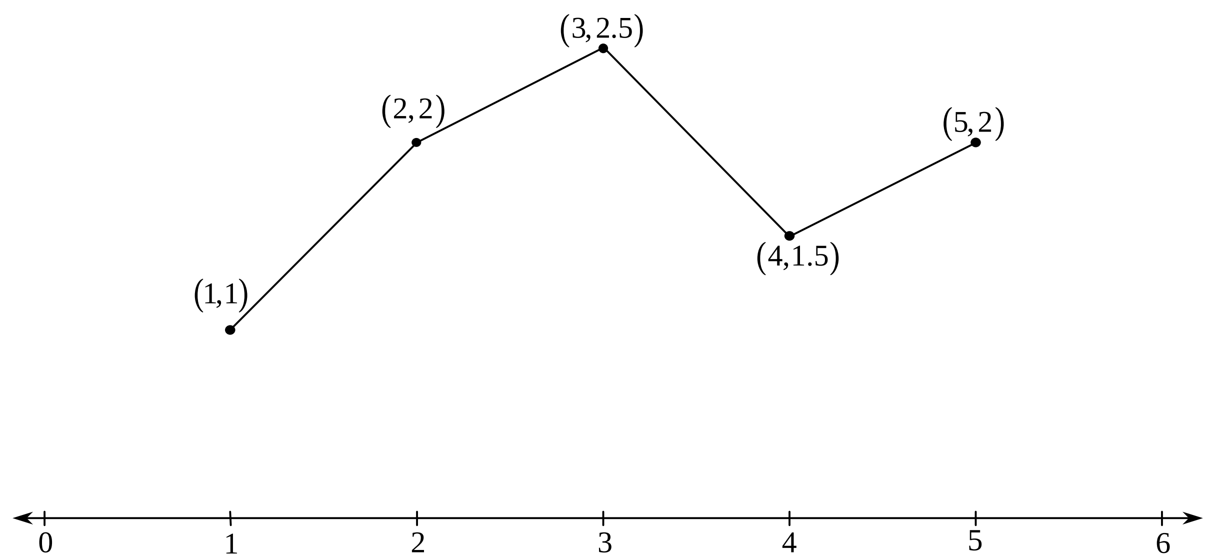
- Bilinear spline basis functions
- Bicubic spline basis functions

Overview – “elementary mathematics from an advanced standpoint”

- linear algebra, finite dimensional function spaces, inner products, small support, almost orthogonal bases, tensor products, finite element methods, wavelets, …

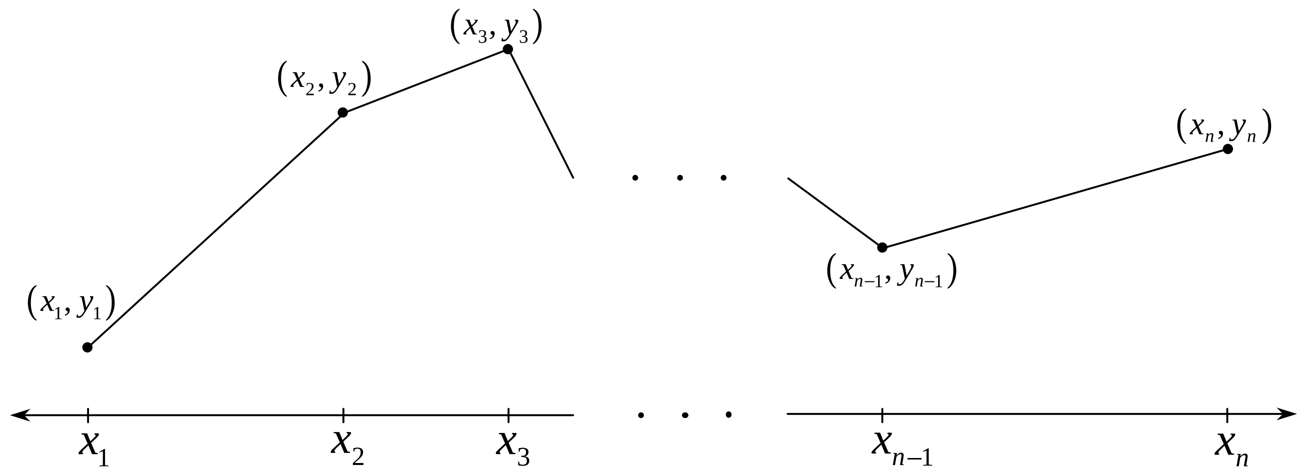
Connecting the Dots

Write a formula for a piecewise linear function that interpolates five given data points



$$f(x) = \begin{cases} x & \text{if } 1 \le x \le 2 \\ \frac{1}{2}x + 1 & \text{if } 2 \le x \le 3 \\ -x + \frac{11}{2} & \text{if } 3 \le x \le 4 \\ \frac{1}{2}x - \frac{1}{2} & \text{if } 4 \le x \le 5 \end{cases}$$

Write a formula for a piecewise linear function that interpolates *n* given data points

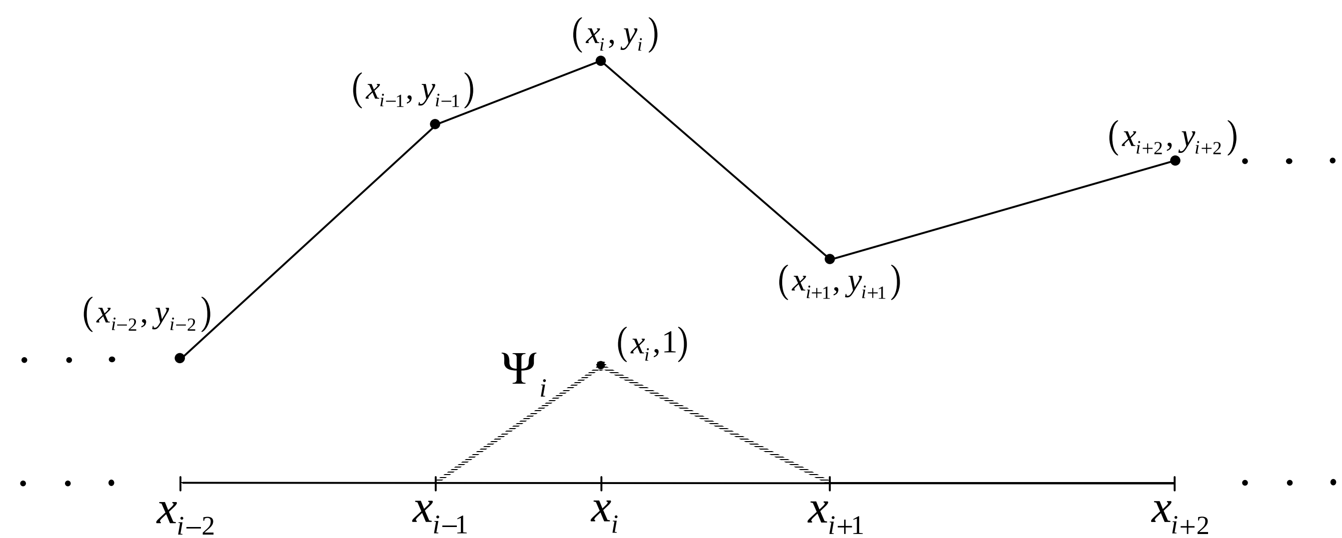


$$f(x) = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} x + \dfrac{y_1 x_2 - y_2 x_1}{x_2 - x_1} & \text{if } x_1 \le x \le x_2 \\ \dfrac{y_3 - y_2}{x_3 - x_2} x + \dfrac{y_2 x_3 - y_3 x_2}{x_3 - x_2} & \text{if } x_2 \le x \le x_3 \\ \vdots & \vdots \\ \dfrac{y_n - y_{n-1}}{x_n - x_{n-1}} x + \dfrac{y_{n-1} x_n - y_n x_{n-1}}{x_n - x_{n-1}} & \text{if } x_{n-1} \le x \le x_n \end{cases}$$

Interpolation using a Linear Spline Basis

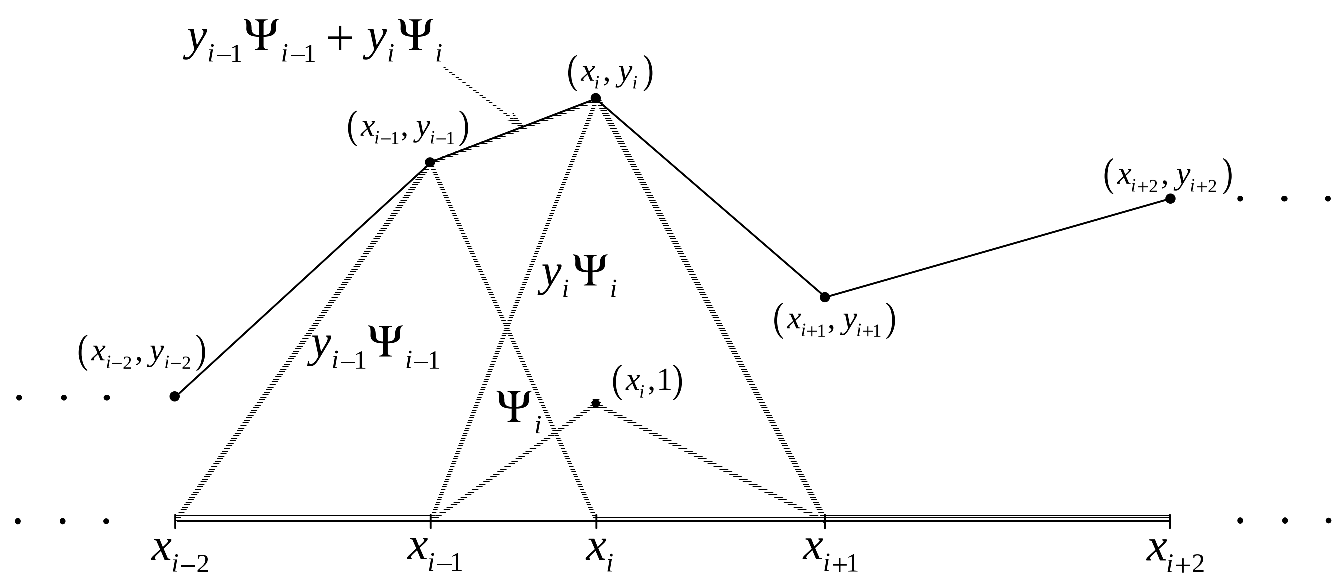
The "linear spline function" approach involves carefully choosing a set of "basis functions" $\Psi_1, \Psi_2, \ldots, \Psi_n$ such that the interpolating function f can be written as a simple linear combination:

$$f(x) = y_1\Psi_1(x) + y_2\Psi_2(x) + \cdots + y_n\Psi_n(x) = \sum_{i=1}^{n} y_i\Psi_i(x) \quad \text{, for all } x_1 \le x \le x_n$$

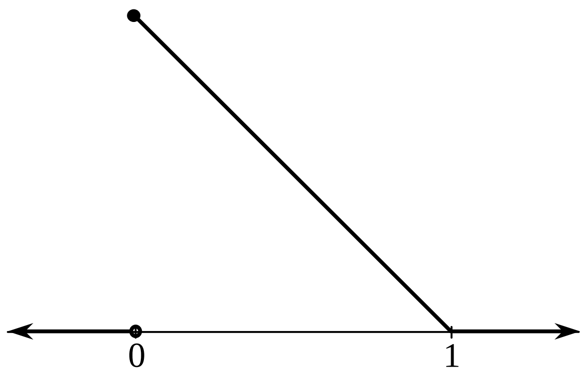


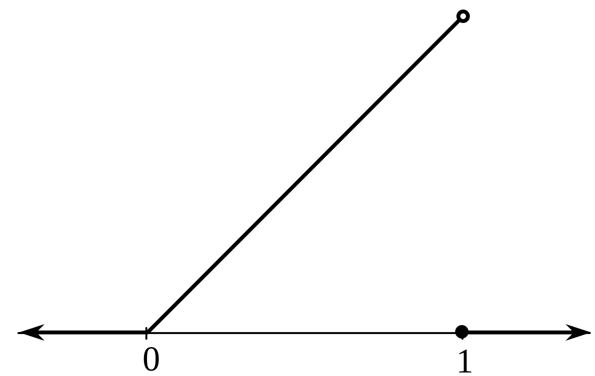
A closer look at a linear combination of basis functions

$$f(x) = y_1\Psi_1(x) + y_2\Psi_2(x) + \cdots + y_n\Psi_n(x) = \sum_{i=1}^{n} y_i\Psi_i(x)$$



The linear spline basis functions can be constructed as sums of translations and horizontal scalings of two "elementary basis functions"

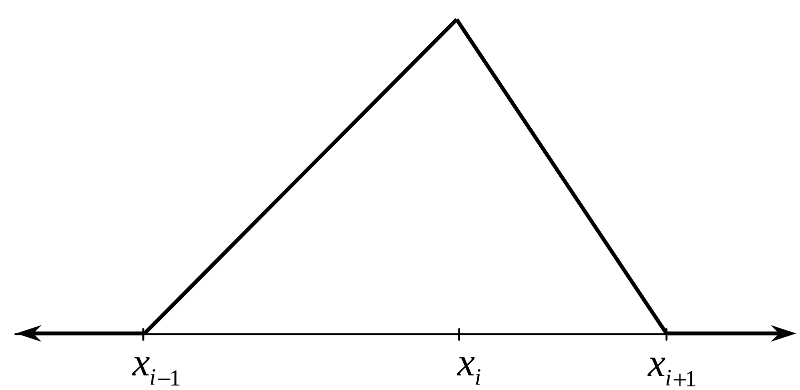




$$\theta_1(x) = \begin{cases} 1 - x & \text{if } x \in [0,1) \\ 0 & \text{if } x \notin [0,1) \end{cases}$$

$$\theta_2(x) = \begin{cases} x & \text{if } x \in [0,1) \\ 0 & \text{if } x \notin [0,1) \end{cases}$$

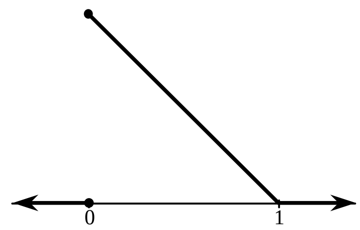
$$\Psi_i(x) = \theta_2\left(\frac{x - x_{i-1}}{x_i - x_{i-1}}\right) + \theta_1\left(\frac{x - x_i}{x_{i+1} - x_i}\right)$$



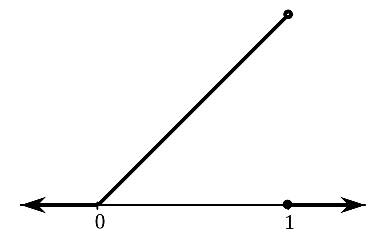
A summary description of the linear spline basis

1. Elementary basis functions – basically constructed on the unit interval

$$\theta_1(x) = \begin{cases} 1 - x & \text{if } x \in [0,1) \\ 0 & \text{if } x \notin [0,1) \end{cases}$$



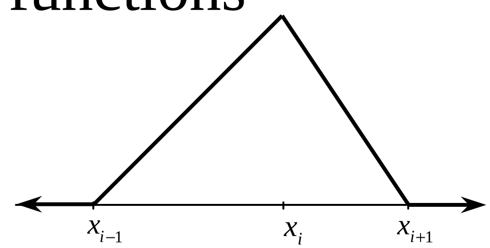
$$\theta_2(x) = \begin{cases} x & \text{if } x \in [0,1) \\ 0 & \text{if } x \notin [0,1) \end{cases}$$



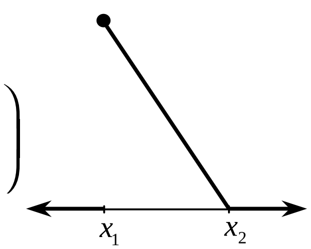
2. A set of nodes -- $x_1 < x_2 < \cdots < x_n$

3. Spline basis functions – sums of (usually) two translated and scaled elementary basis functions

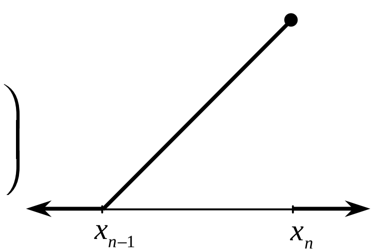
interior: $i = 2, \ldots, n-1$ $\quad \Psi_i(x) = \theta_2\left(\dfrac{x - x_{i-1}}{x_i - x_{i-1}}\right) + \theta_1\left(\dfrac{x - x_i}{x_{i+1} - x_i}\right)$



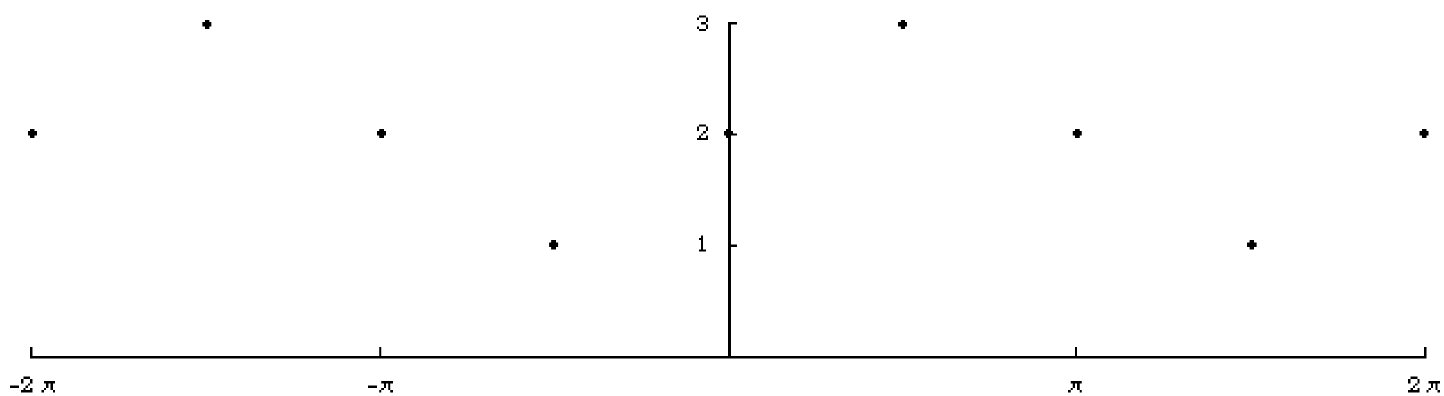
endpoints: $i = 1$ and n $\quad \Psi_1(x) = \theta_1\left(\dfrac{x - x_1}{x_2 - x_1}\right)$



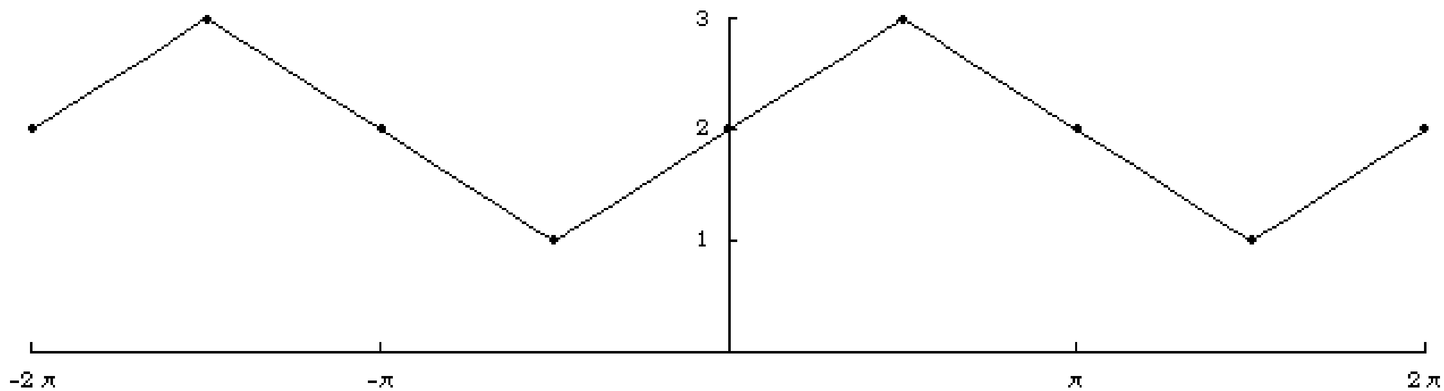
$$\Psi_n(x) = \bar{\theta}_2\left(\frac{x - x_{n-1}}{x_n - x_{n-1}}\right)$$



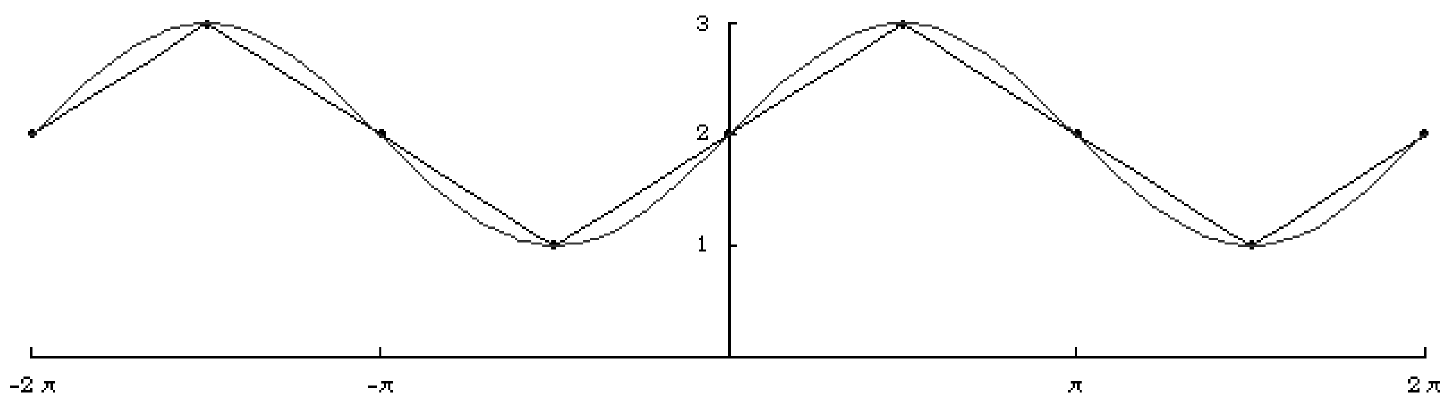
Points from sin(x)
Points



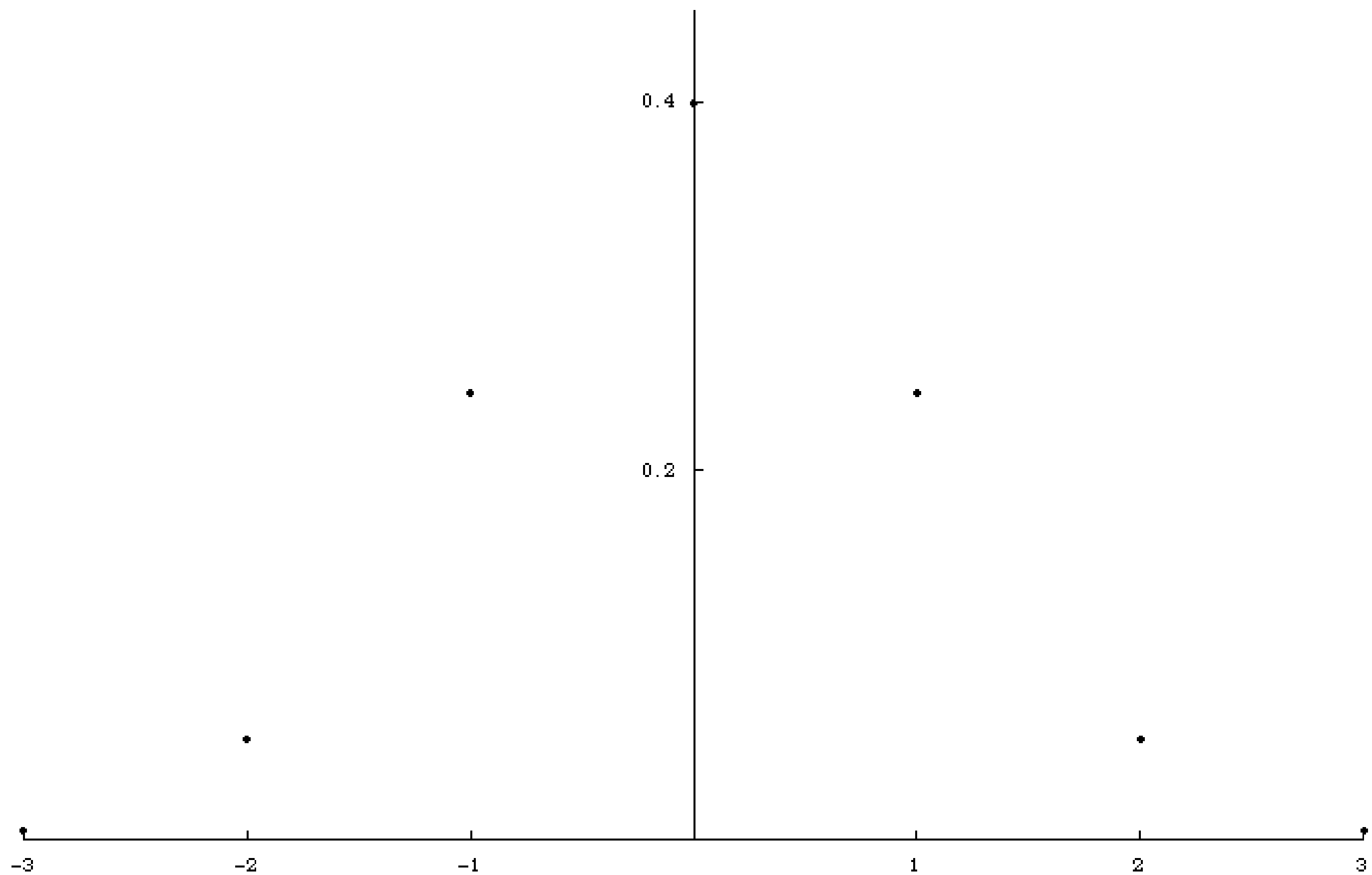
Points from sin(x)
Linear Interpolation



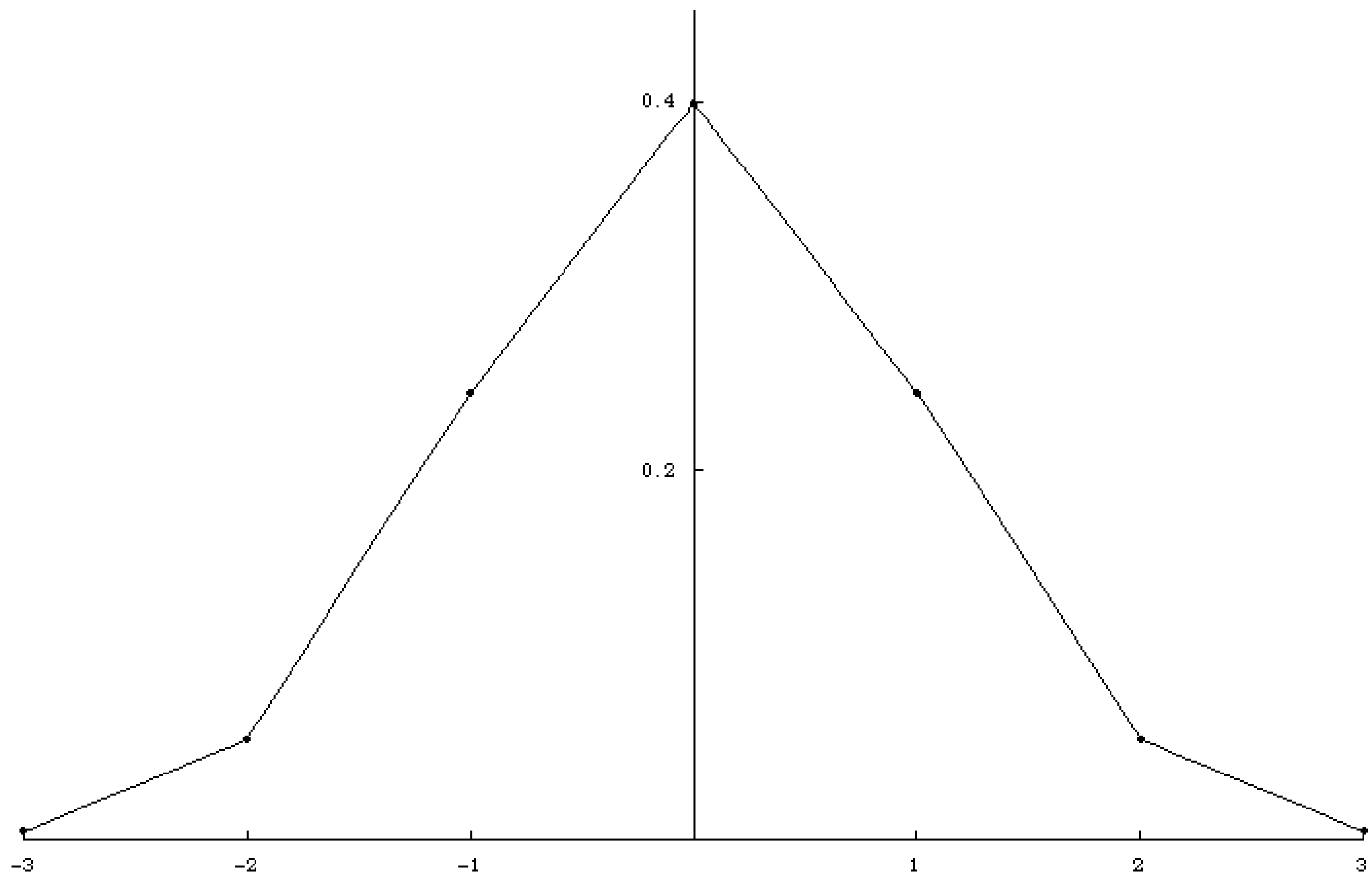
Points from sin(x) Linear Interpolation & “retail”



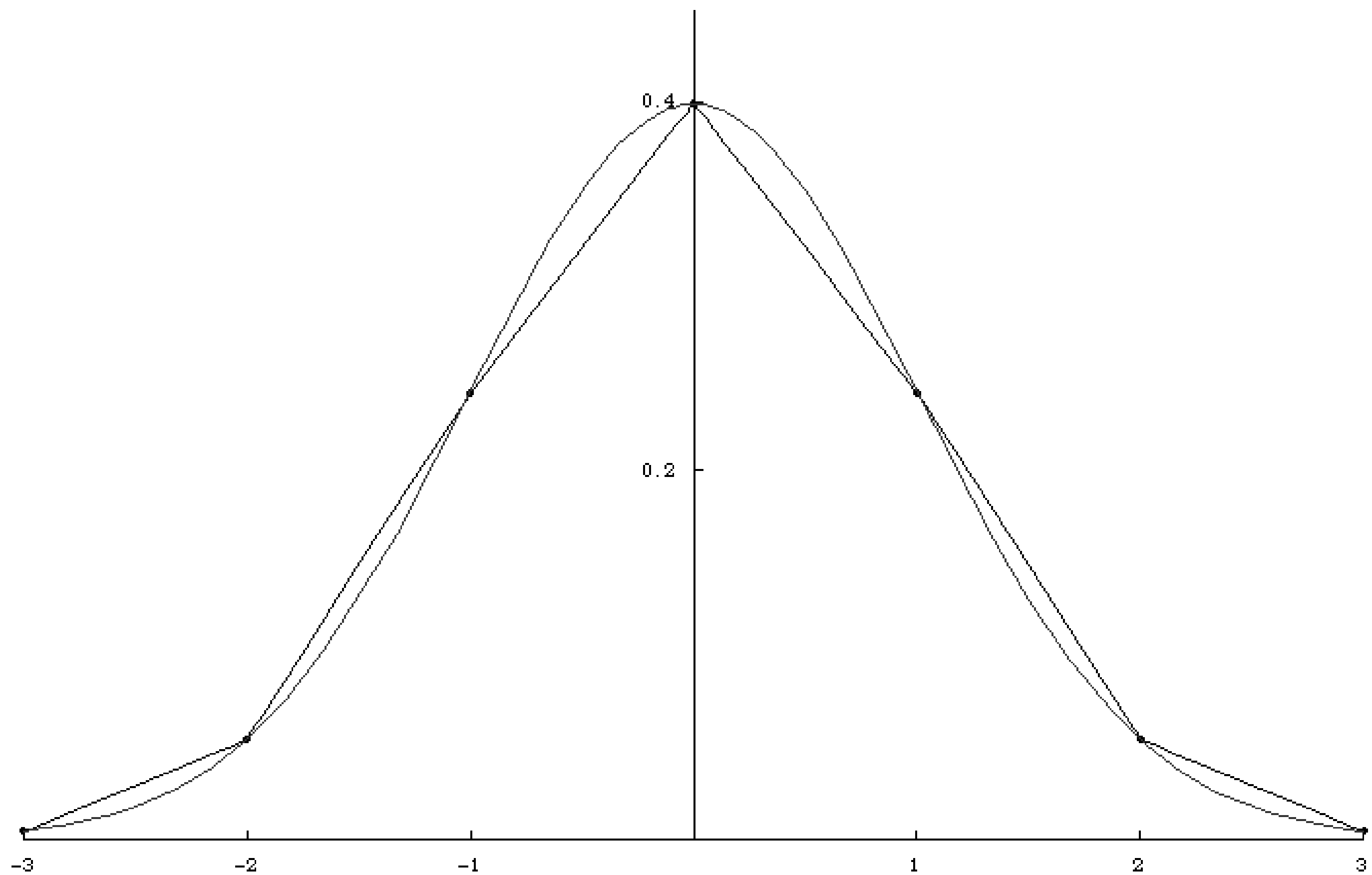
Points from $g(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$



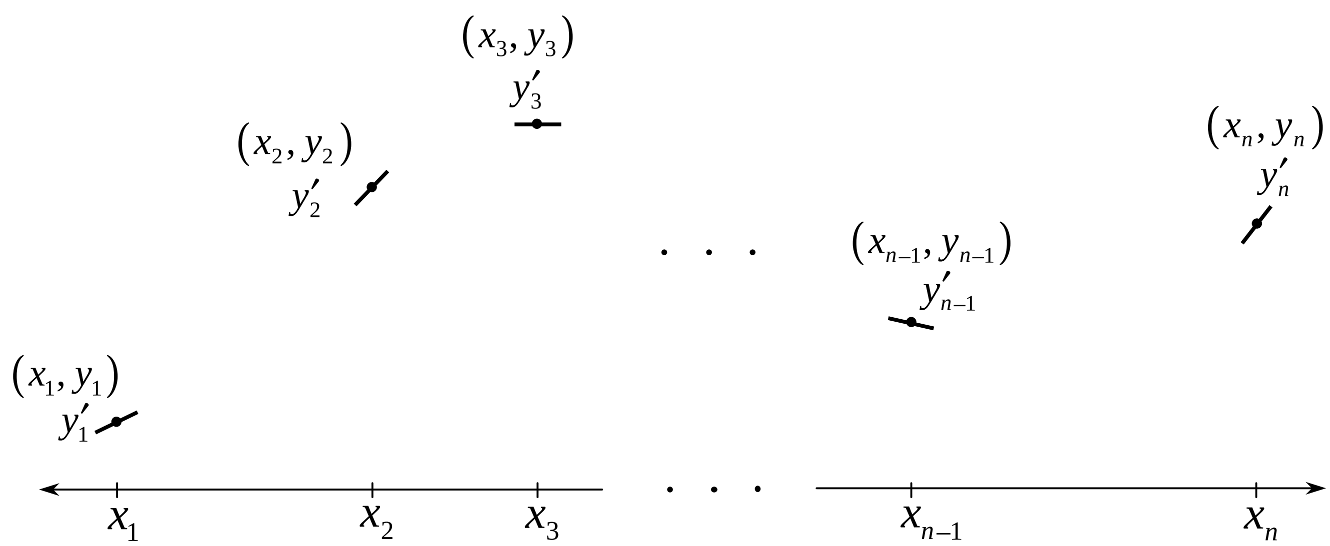
Points from g(x)
Linear Interpolation



Points from g(x)
Linear Interpolation & “retail”



Find a smooth piecewise cubic function that has specified function values and slopes at *n* given data points



where y_i' denotes the slope or first derivative at the data point (x_i, y_i)

Why cubic ?

Find a cubic function that has specified function values and slopes at 2 given data points

Let $p(x) = ax^3 + bx^2 + cx + d$ so $p'(x) = 3ax^2 + 2bx + c$

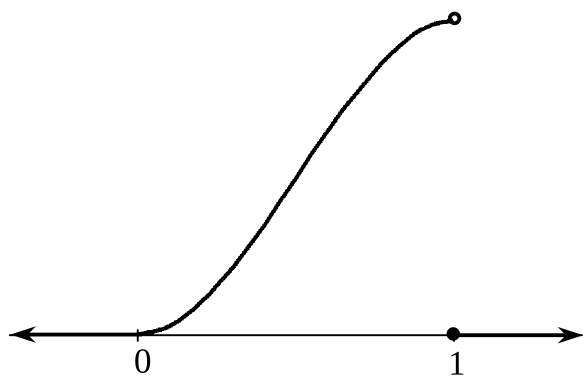
Using the data we have

$$p(x_1) = ax_1^3 + bx_1^2 + cx_1 + d = y_1$$
$$p(x_2) = ax_2^3 + bx_2^2 + cx_2 + d = y_2$$
$$p'(x_1) = 3ax_1^2 + 2bx_1 + c = y_1'$$
$$p'(x_2) = 3ax_2^2 + 2bx_2 + c = y_2'$$

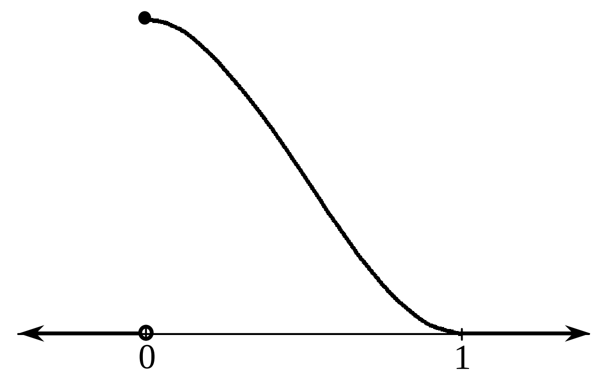
We have 4 equations in the four unknown coefficients. So we can solve for *a*, *b*, *c*, and *d*.

Geometer’s sketch pad demonstration

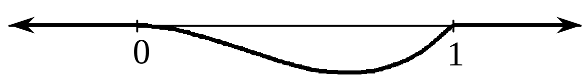
The cubic spline basis functions can be constructed as sums of translations and scalings of four "elementary basis functions"



$$\theta_2(0) = 0, \quad \theta_2(1) = 1$$
$$\theta_2'(0) = 0, \quad \theta_2'(1) = 0$$



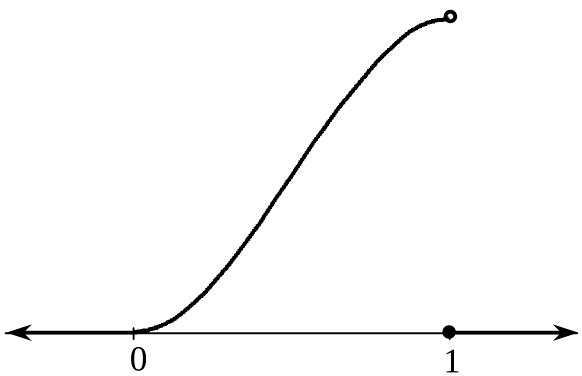
$$\theta_1(0) = 1, \quad \theta_1(1) = 0$$
$$\theta_1'(0) = 0, \quad \theta_1'(1) = 0$$



$$\varphi_2(0) = 0, \quad \varphi_2(1) = 0$$
$$\varphi_2'(0) = 0, \quad \varphi_2'(1) = 1$$

$$\varphi_1(0) = 0, \quad \varphi_1(1) = 0$$
$$\varphi_1'(0) = 1, \quad \varphi_1'(1) = 0$$

Construction of the cubic elementary basis functions



$$\theta_2(0)=0, \quad \theta_2(1)=1$$
$$\theta_2'(0)=0, \quad \theta_2'(1)=0$$

There are several reasonable ways to find the cubic function with the desired properties. A straightforward way is to write a general cubic, compute its derivative, substitute the desired conditions and solve the resulting system of four linear equations for the coefficients.

A somewhat more interesting way is to note that the most general cubic with a zero value and a zero derivative at 0 is

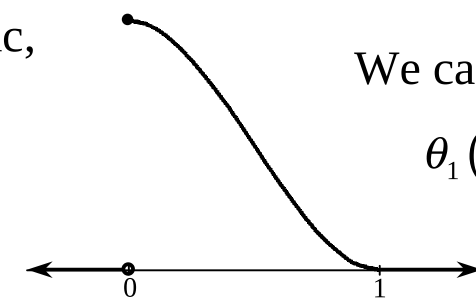
$$\theta_2(x)=x^2(ax+b) \quad \text{so} \quad \theta_2'(x)=3ax^2+2bx$$

Now $\theta_2(1)=1 \quad \Rightarrow \quad a+b=1$

and $\theta_2'(1)=0 \quad \Rightarrow \quad 3a+2b=0$

Hence $a=-2, \quad b=3$

and $\boxed{\theta_2(x)=x^2(3-2x)}$



$$\theta_1(0)=1, \quad \theta_1(1)=0$$
$$\theta_1'(0)=0, \quad \theta_1'(1)=0$$

We can see by symmetry that

$$\theta_1(x)=\theta_2(1-x)$$
$$=(1-x)^2\left[3-2(1-x)\right]$$
$$=(1-x)^2(2x+1)$$
$$=(x-1)^2(2x+1)$$

Construction of the cubic elementary basis functions

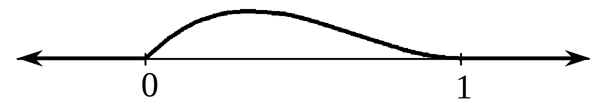
$$\varphi_2(0)=0, \quad \varphi_2(1)=0$$
$$\varphi_2'(0)=0, \quad \varphi_2'(1)=1$$

In this case we note that the most general cubic with a zero value and a zero derivative at 0 and a zero value at 1 is

$$\varphi_2(x)=ax^2(x-1) \text{ so } \varphi_2'(x)=a\left(3x^2-2x\right)$$

Now $\varphi_2'(1)=1 \Rightarrow a=1$

hence $\boxed{\varphi_2(x)=x^2(x-1)}$

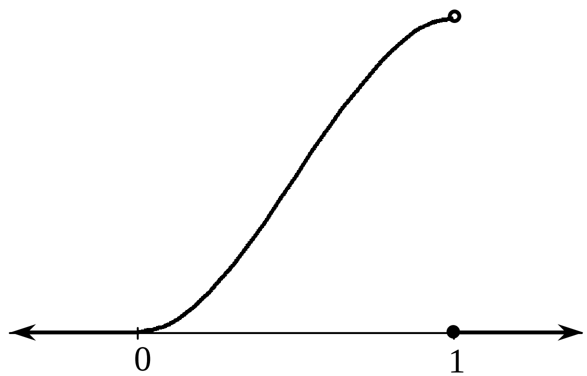


$$\varphi_1(0)=0, \quad \varphi_1(1)=0$$
$$\varphi_1'(0)=1, \quad \varphi_1'(1)=0$$

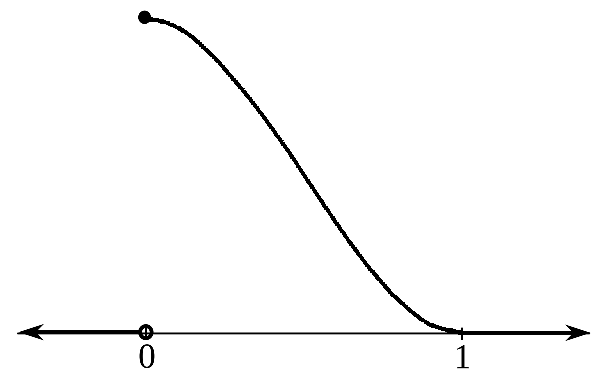
Again using symmetry we see that

$$\begin{aligned}\varphi_1(x)&=-\varphi_2(1-x)\\&=-(1-x)^2\left[(1-x)-1\right]\\&=(1-x)^2 x\\&=x(x-1)^2\end{aligned}$$

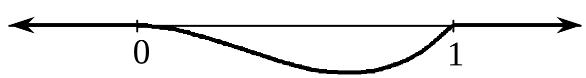
The Cubic Spline Elementary Basis Functions



$$\theta_2(x) = \begin{cases} x^2(3-2x) & \text{if } x \in [0,1) \\ 0 & \text{if } x \notin [0,1) \end{cases}$$



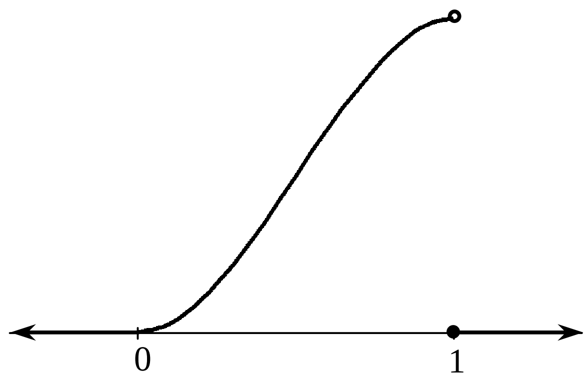
$$\theta_1(x) = \theta_2(1-x) = \begin{cases} (2x+1)(x-1)^2 & \text{if } x \in [0,1) \\ 0 & \text{if } x \notin [0,1) \end{cases}$$



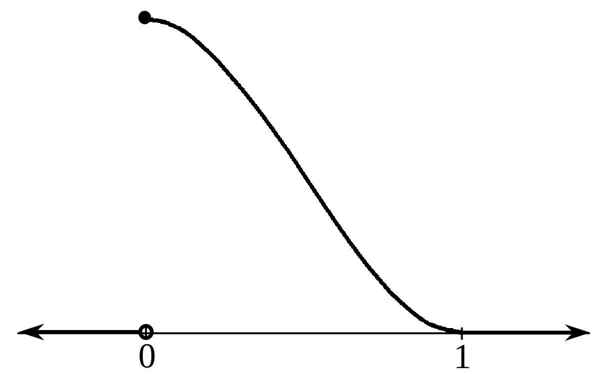
$$\varphi_2(x) = \begin{cases} x^2(x-1) & \text{if } x \in [0,1) \\ 0 & \text{if } x \notin [0,1) \end{cases}$$

$$\varphi_1(x) = -\varphi_2(1-x) = \begin{cases} x(x-1)^2 & \text{if } x \in [0,1) \\ 0 & \text{if } x \notin [0,1) \end{cases}$$

Construction of the cubic spline basis functions as translations and scalings of four “elementary basis functions”



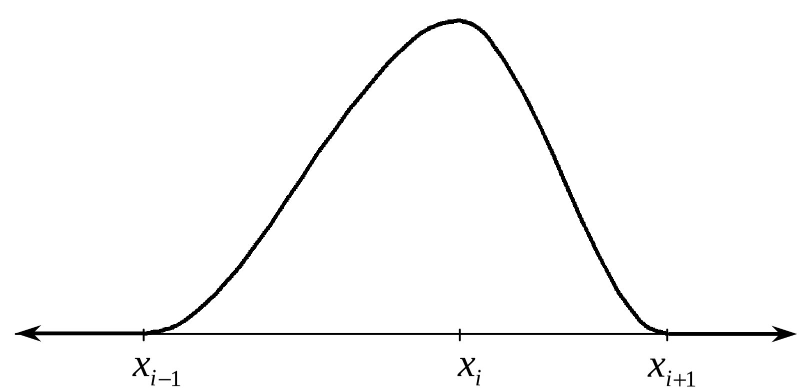
$\theta_2(x)$



$\theta_1(x)$

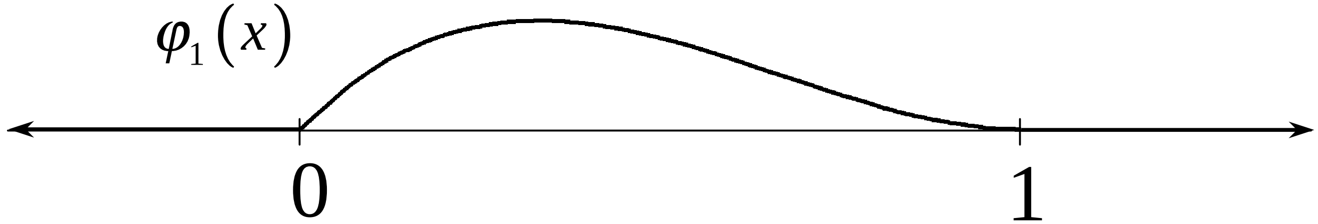
$$\Psi_i(x) = \theta_2\left(\frac{x - x_{i-1}}{x_i - x_{i-1}}\right) + \theta_1\left(\frac{x - x_i}{x_{i+1} - x_i}\right)$$

used to match function values

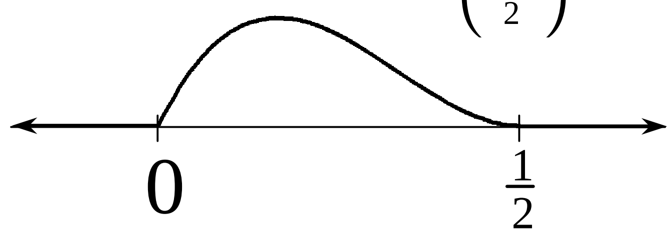


Scaling the derivative basis functions

$\varphi_1(x)$

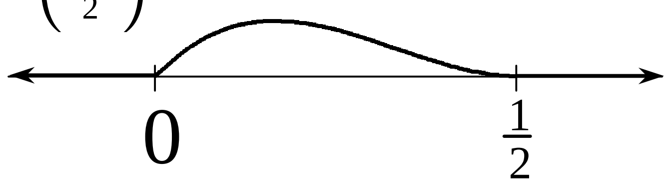


$\varphi_1(2x)$ or $\varphi_1\left(\frac{x}{\frac{1}{2}}\right)$



A horizontal scaling by a factor of ½ changes the slopes by a factor of 2.

$\frac{1}{2}\varphi_1\left(\frac{x}{\frac{1}{2}}\right)$

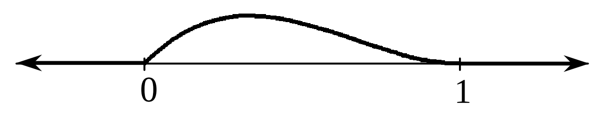


A vertical scaling by a factor of ½ returns the slopes to their corresponding values.

In general the function $hg\left(\frac{x}{h}\right)$ has the same slope at $\frac{x}{h}$ as $g(x)$ has at x

Construction of the cubic spline "derivative" basis functions from the "derivative" elementary basis functions

$\varphi_2(x)$



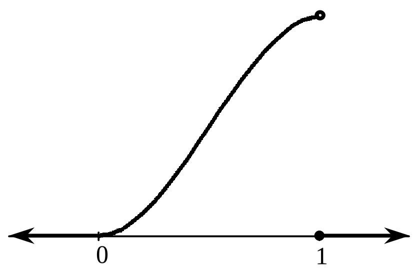
$\varphi_1(x)$

$$\Phi_i(x) = (x_i - x_{i-1})\varphi_2\left(\frac{x - x_{i-1}}{x_i - x_{i-1}}\right) + (x_{i+1} - x_i)\varphi_1\left(\frac{x - x_i}{x_{i+1} - x_i}\right)$$

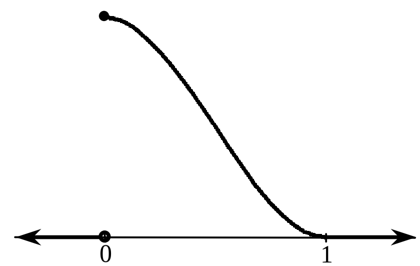
used to match slope values

A summary description of the Cubic Spline Basis

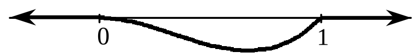
- Elementary basis functions – basically constructed on the unit interval four for the cubics



$$\theta_2(x) = \begin{cases} x^2(3-2x) & if\ x \in [0,1) \\ 0 & if\ x \notin [0,1) \end{cases}$$



$$\theta_1(x) = \theta_2(1-x) = \begin{cases} (2x+1)(x-1)^2 & if\ x \in [0,1) \\ 0 & if\ x \notin [0,1) \end{cases}$$



$$\varphi_2(x) = \begin{cases} x^2(x-1) & if\ x \in [0,1) \\ 0 & if\ x \notin [0,1) \end{cases}$$

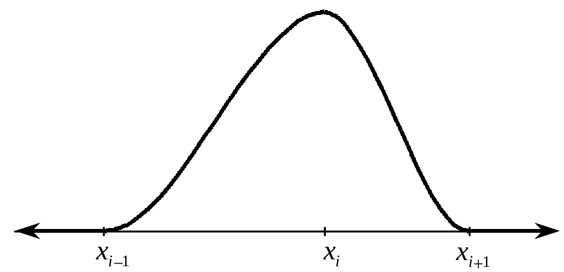
$$\varphi_1(x) = -\varphi_2(1-x) = \begin{cases} x(x-1)^2 & if\ x \in [0,1) \\ 0 & if\ x \notin [0,1) \end{cases}$$

2. A set of nodes -- $x_1 < x_2 < \cdots < x_n$

3. Spline basis functions – sums of (usually) two translated and scaled elementary basis functions

interior: $i = 2, \ldots, n-1$

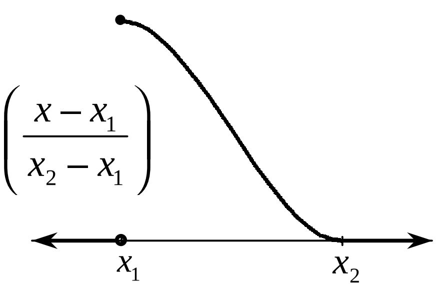
$$\Psi_i(x) = \theta_2\left(\frac{x - x_{i-1}}{x_i - x_{i-1}}\right) + \theta_1\left(\frac{x - x_i}{x_{i+1} - x_i}\right)$$



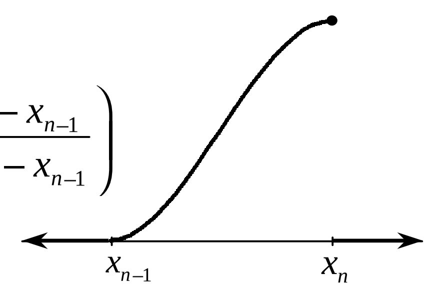
$$\Phi_i(x) = (x_i - x_{i-1})\varphi_2\left(\frac{x - x_{i-1}}{x_i - x_{i-1}}\right) + (x_{i+1} - x_i)\varphi_1\left(\frac{x - x_i}{x_{i+1} - x_i}\right)$$

endpoints: $i = 1$ and n

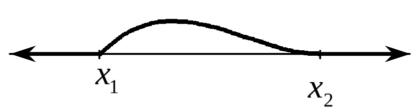
$$\Psi_1(x) = \theta_1\left(\frac{x - x_1}{x_2 - x_1}\right)$$



$$\Psi_n(x) = \overline{\theta_2}\left(\frac{x - x_{n-1}}{x_n - x_{n-1}}\right)$$



$$\Phi_1(x) = (x_2 - x_1)\varphi_1\left(\frac{x - x_1}{x_2 - x_1}\right)$$

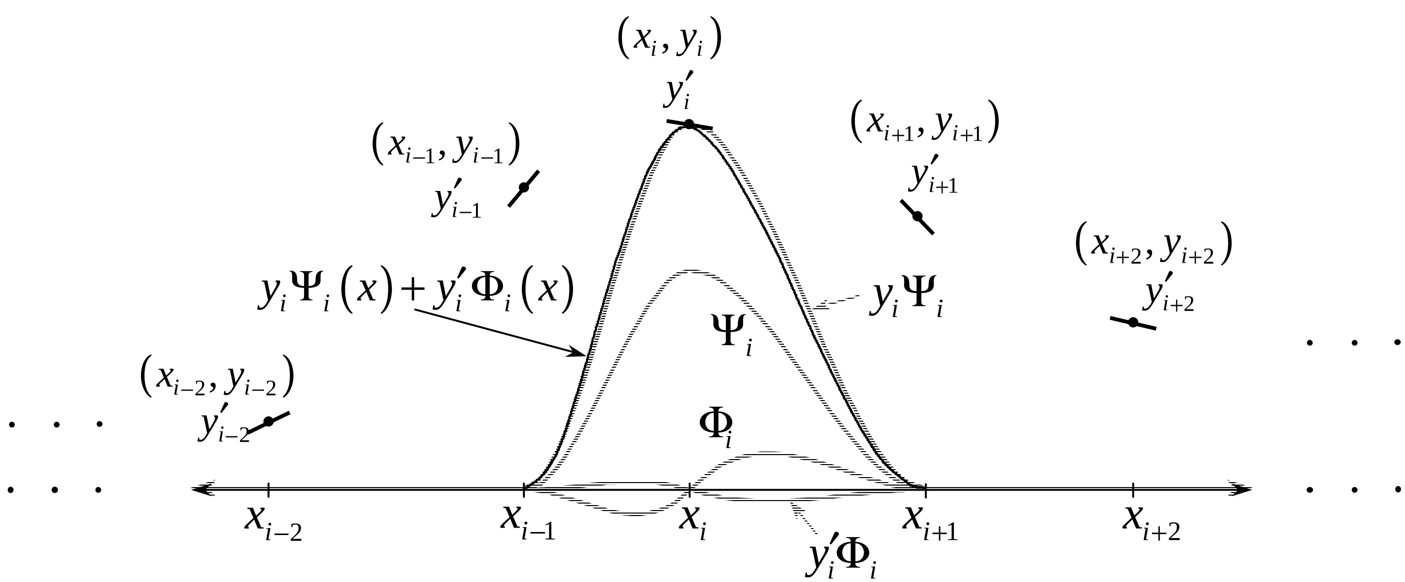


$$\Phi_n(x) = (x_n - x_{n-1})\varphi_2\left(\frac{x - x_{n-1}}{x_n - x_{n-1}}\right)$$

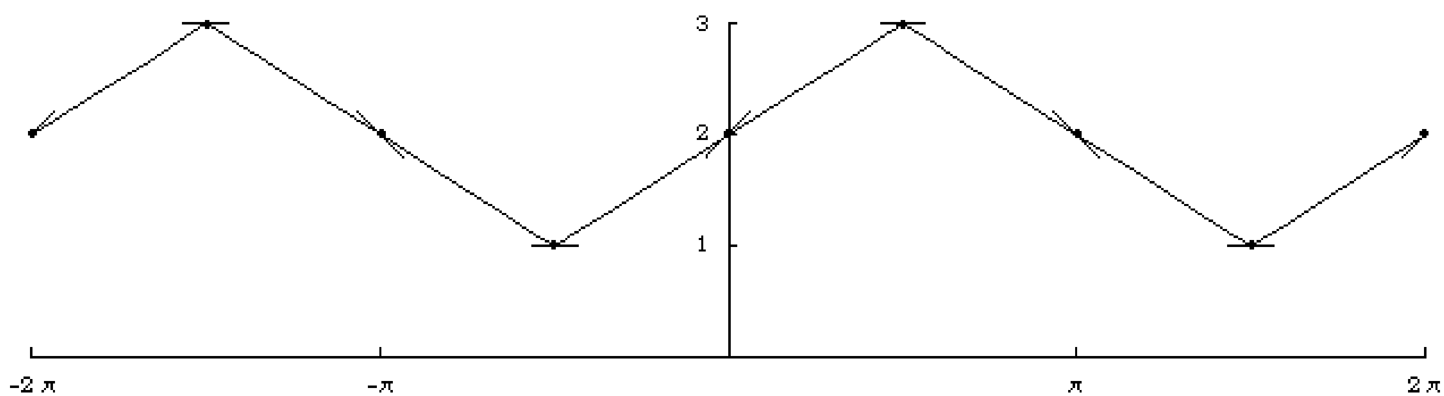
A closer look at a linear combination of cubic basis functions

$$f(x) = y_1\Psi_1(x) + y_1'\Phi_1(x) + y_2\Psi_2(x) + y_2'\Phi_2(x) + \cdots + y_n\Psi_n(x) + y_n'\Phi_n(x)$$

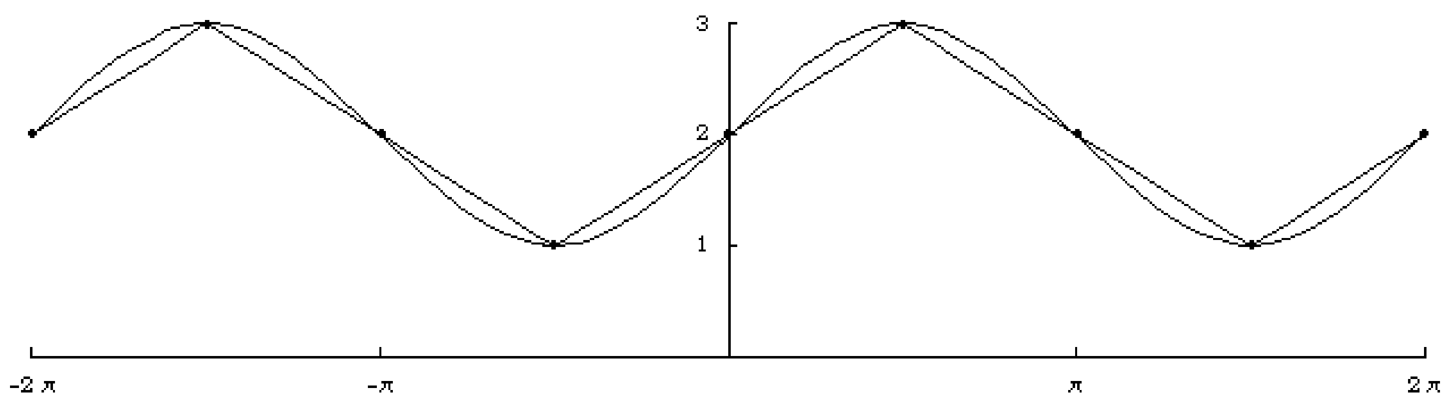
$$= \sum_{i=1}^{n} y_i\Psi_i(x) + y_i'\Phi_i(x)$$



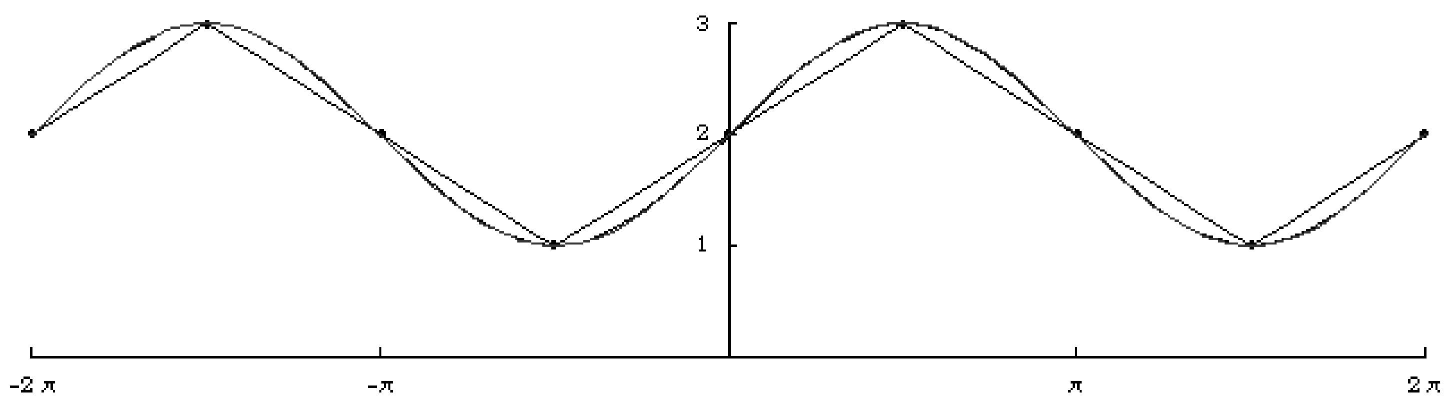
Points from sin(x)
Linear Interpolation & slopes



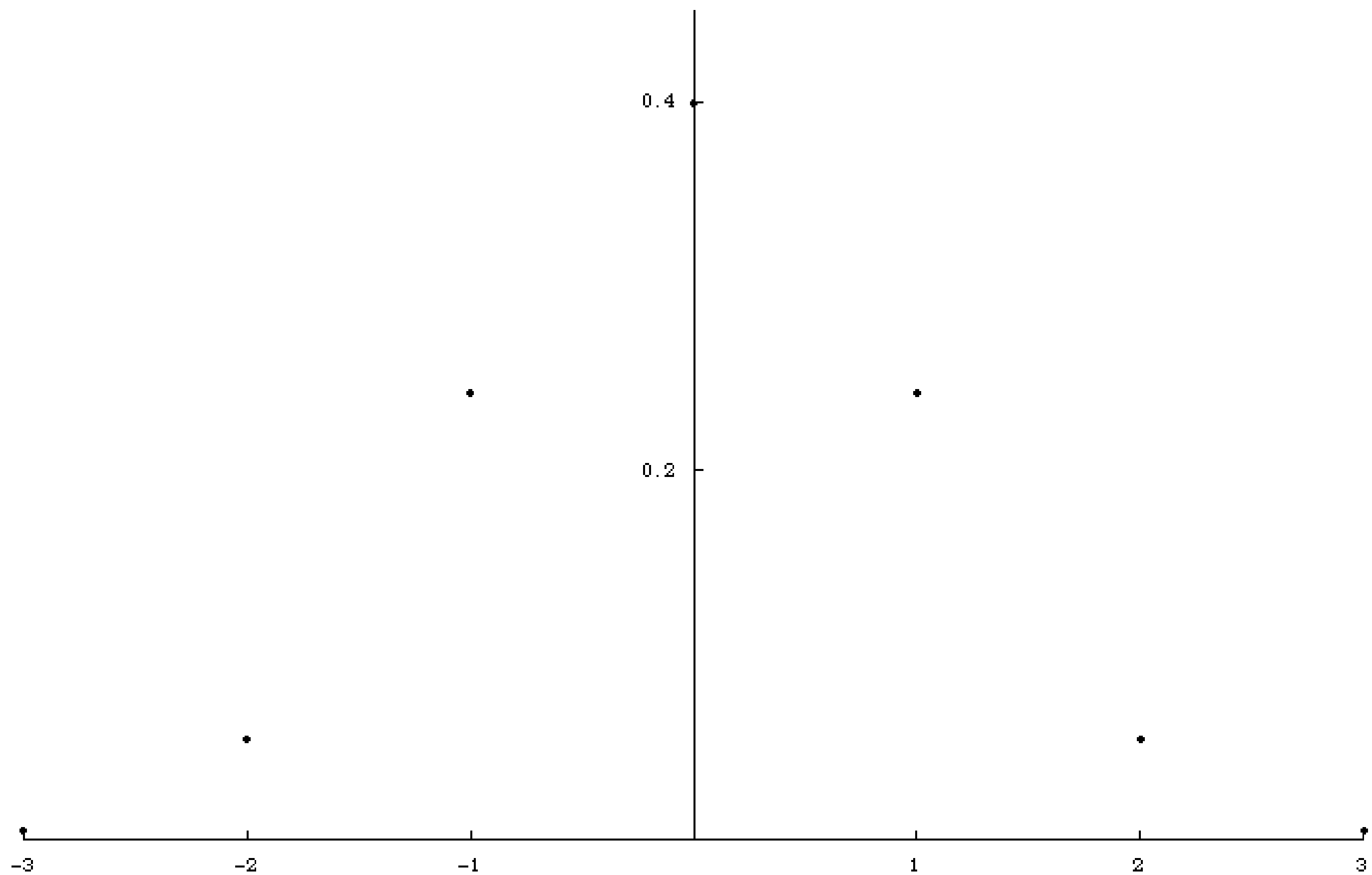
Points from sin(x)
Cubic Interpolation



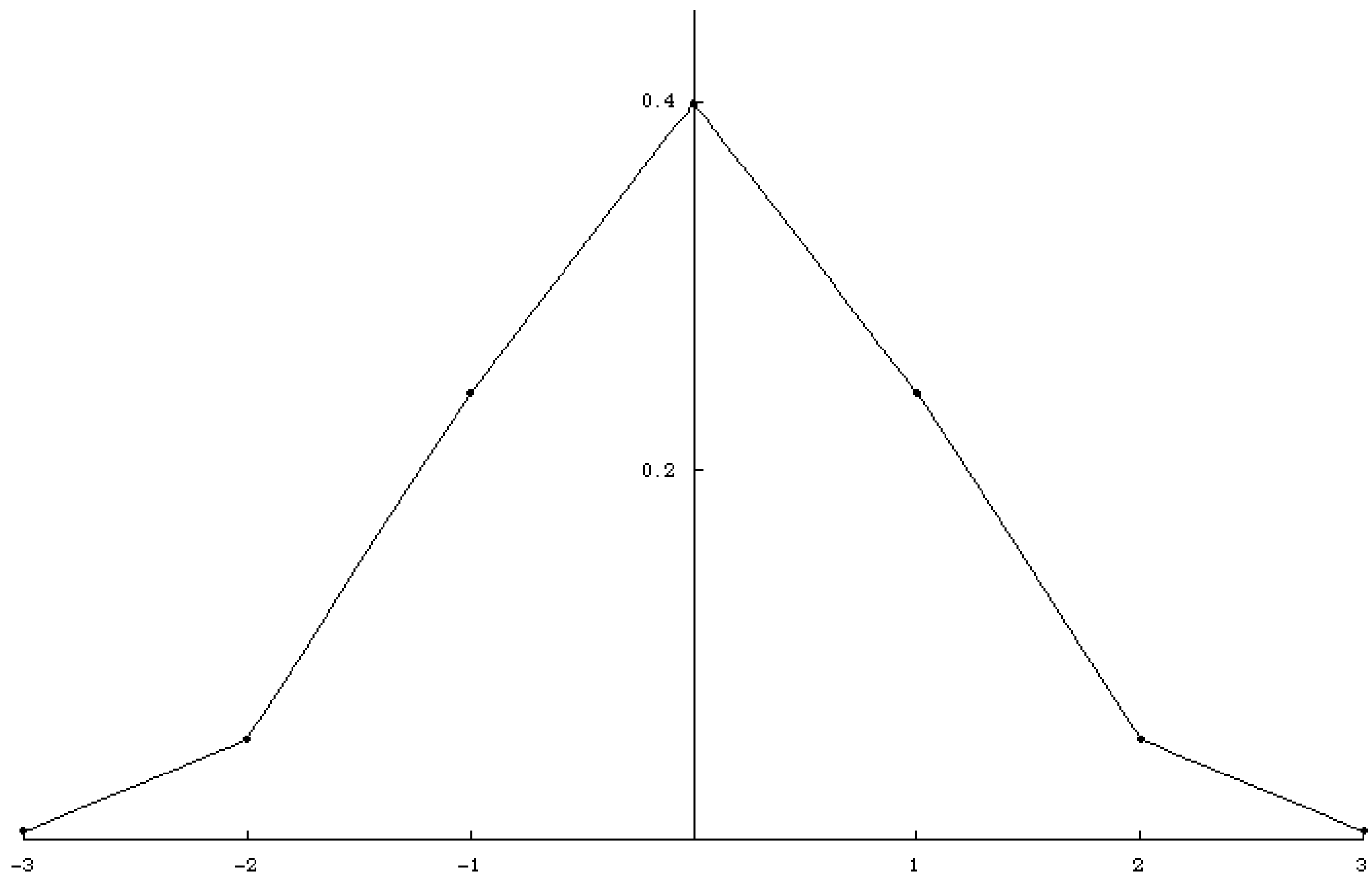
Points from sin(x)
Interpolations and “Retail”



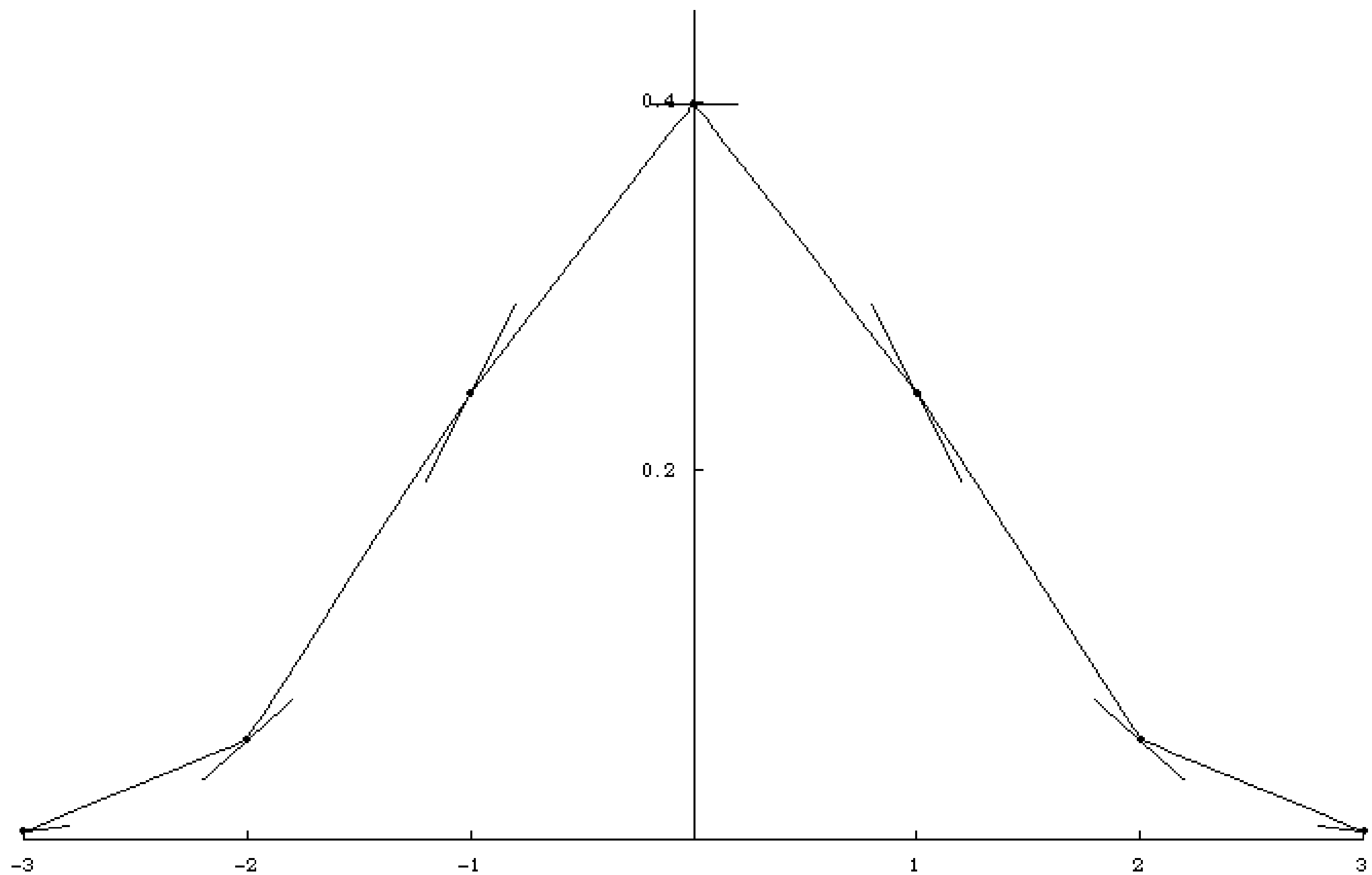
Points from $g\left(x\right)=\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}$



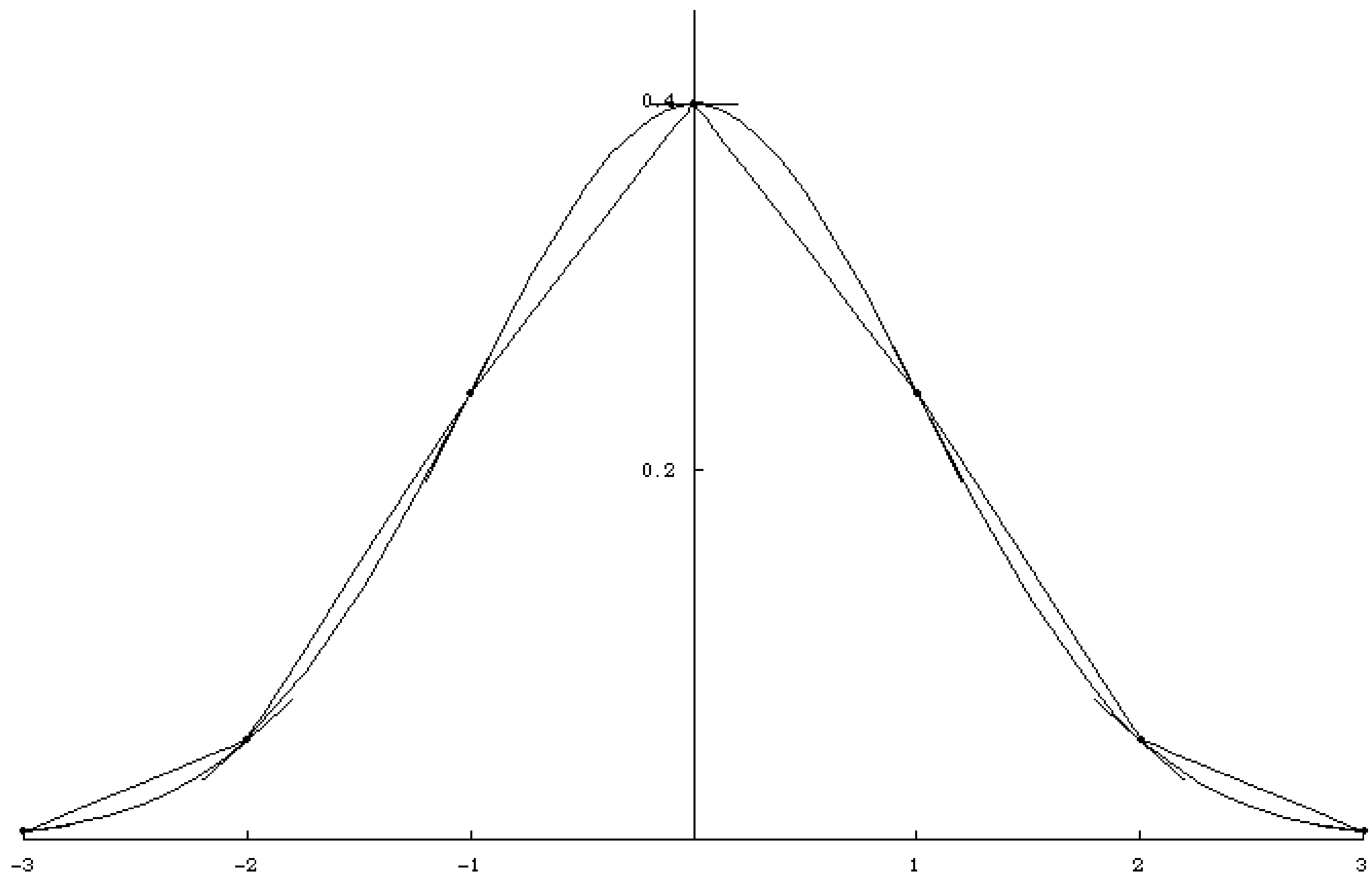
Points from g(x)
Linear Interpolation



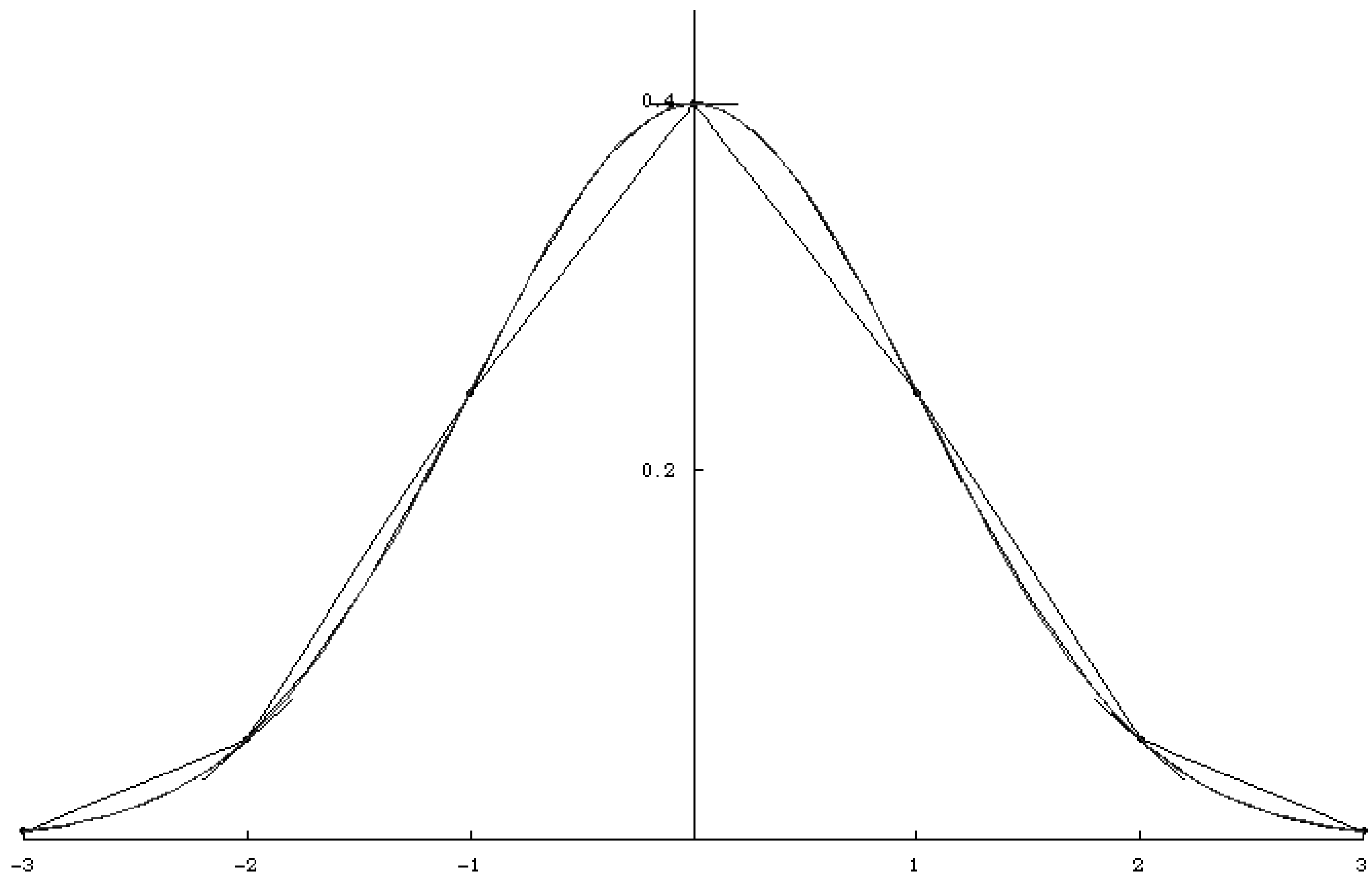
Points from g(x)
Linear Interpolation & slopes



Points from g(x)
Cubic Interpolation



Points from g(x) Interpolations and “Retail”



Geometer’s sketch pad demonstration

Some nice properties of cubic splines

Localization – changing the data at a point only changes the interpolant on the two adjacent subintervals

Simple construction - starts with really only two elementary basis functions; the rest follows from symmetry, scaling, translation and linear combinations

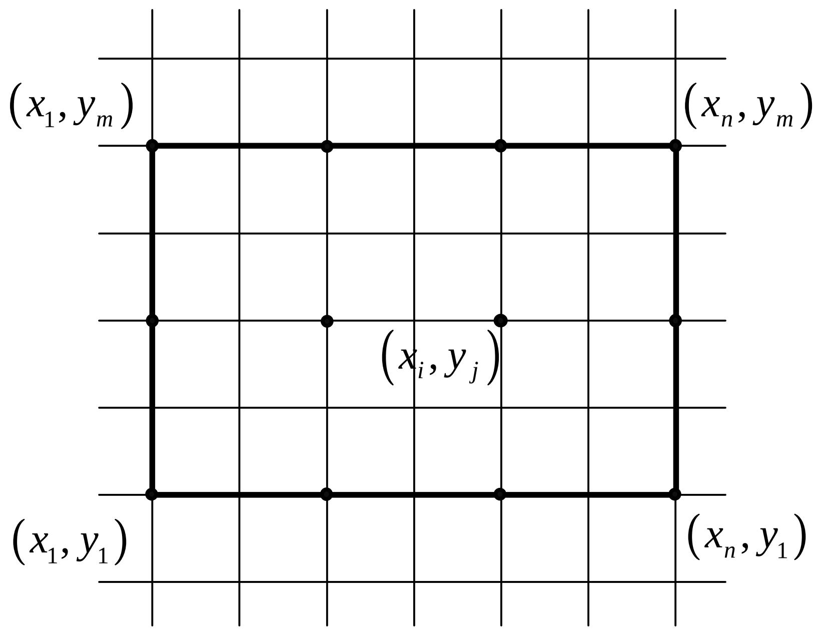
Easy “synthesis” - since the interpolating function is a finite linear combination with the data as the coefficients it is easy to construct

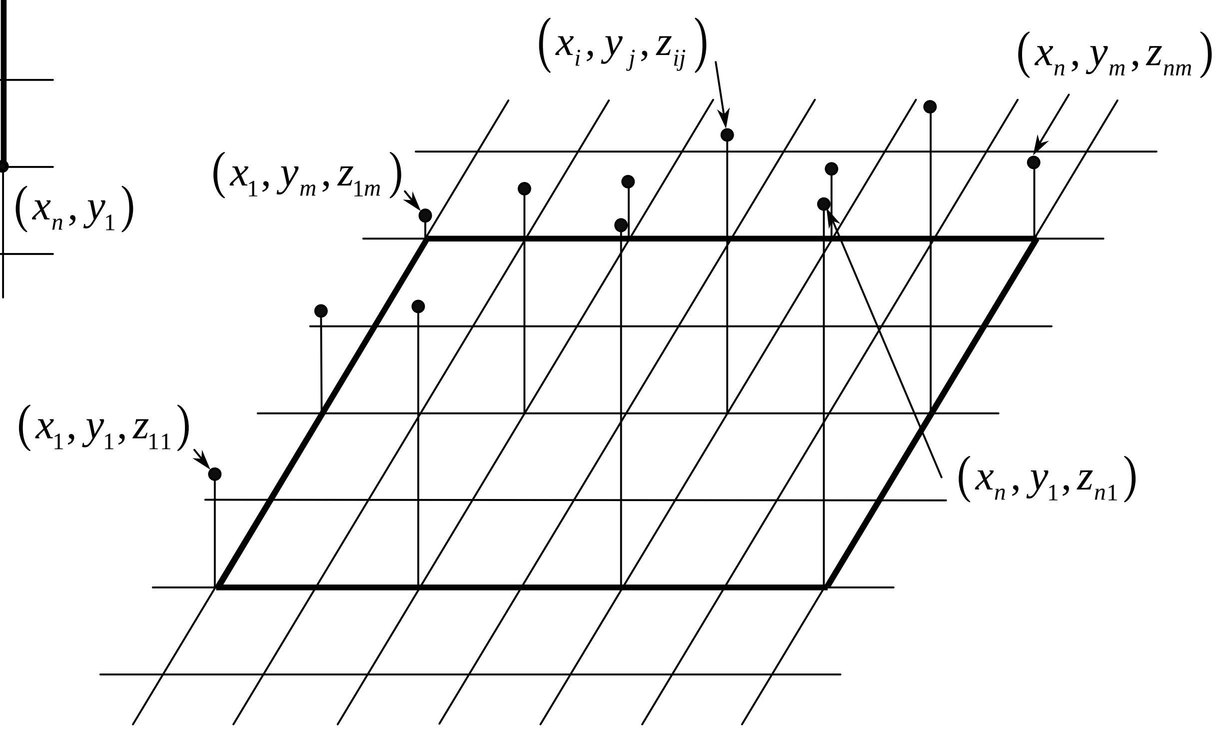
Easy “analysis” - if we are given a cubic splines function we can find the coefficients by evaluating function and its derivative at the nodes

Smoothness – good but not great. We have continuous first derivatives, but generally discontinuous second derivatives.

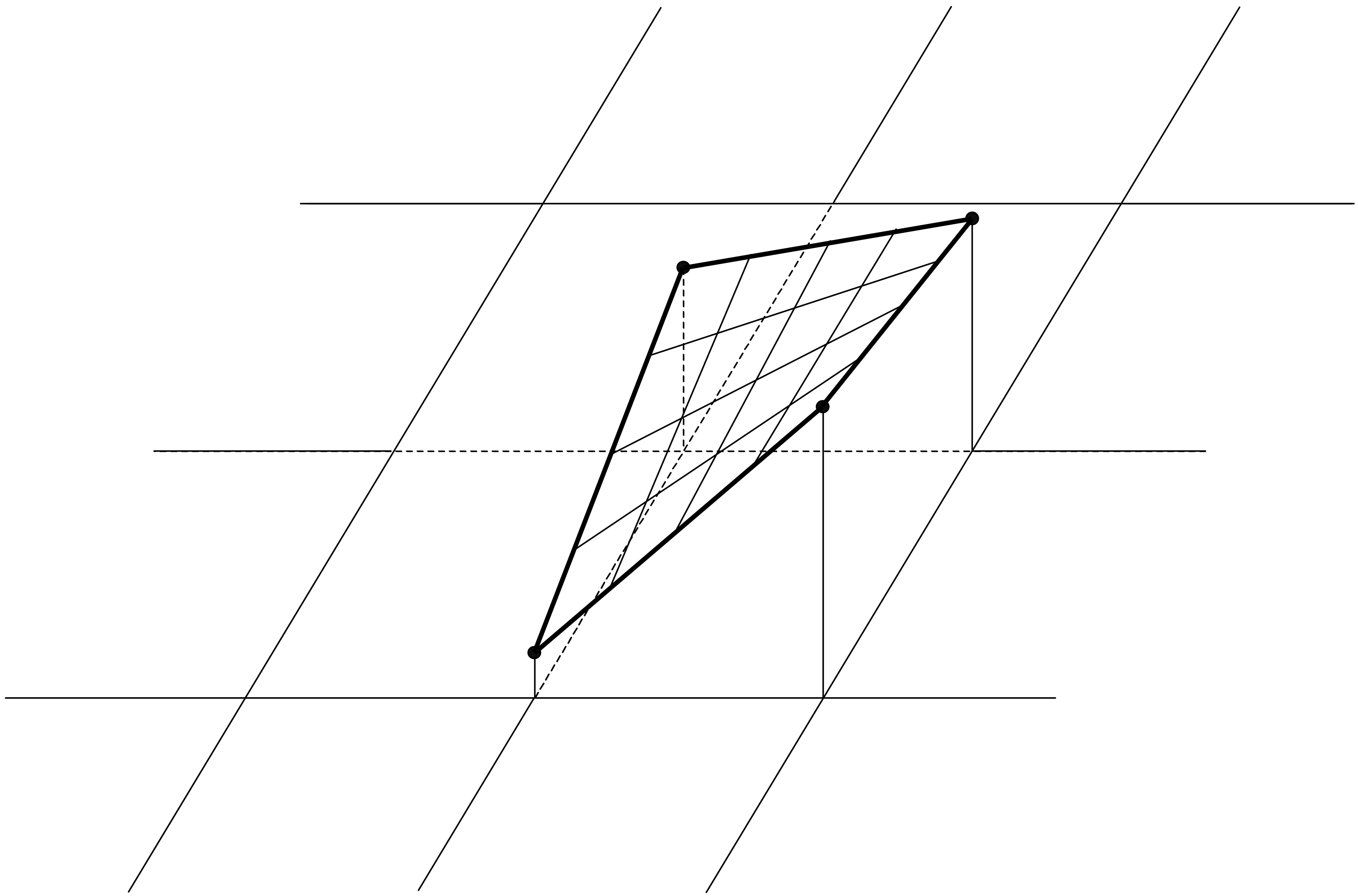
Easy implementation - it is straight forward to program a computer to set up and evaluate cubic splines efficiently.

Find a two-dimensional analog of linear interpolation for specified function values at *mn* given data points

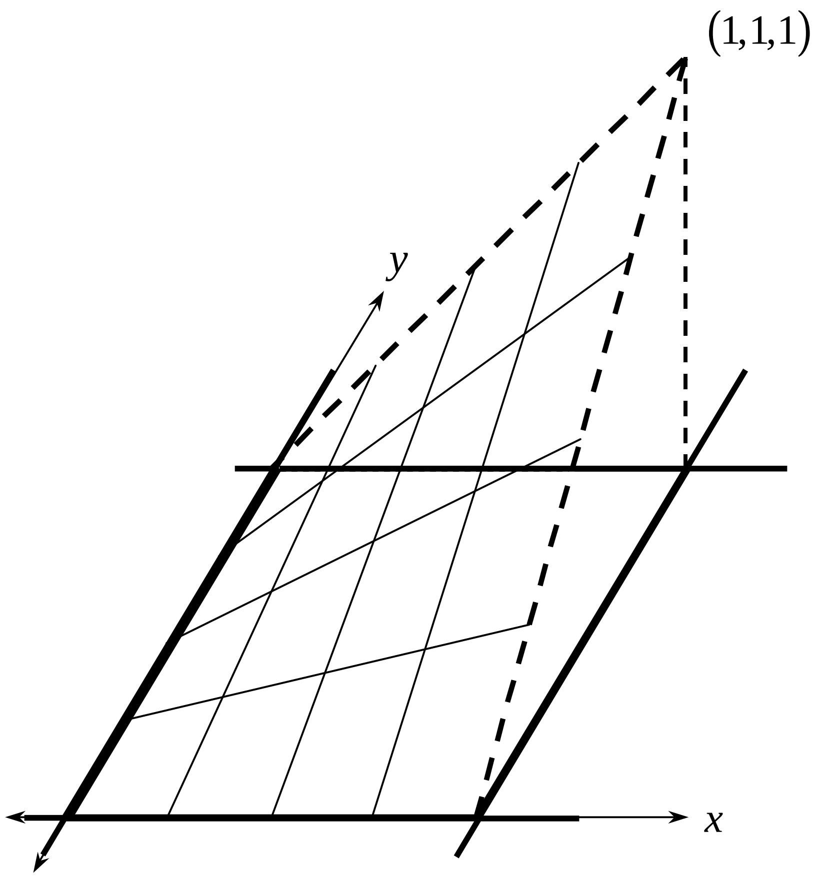


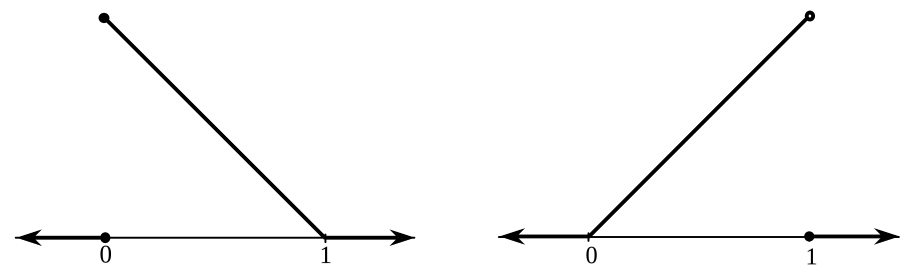


A closer view of four data points



There a are four "bilinear" 2-D elementary basis functions. They are constructed from the 1-D elementary basis functions





$\theta_1(t)$ $\theta_2(t)$

We define the bilinear 2-D elementary basis functions as follows:

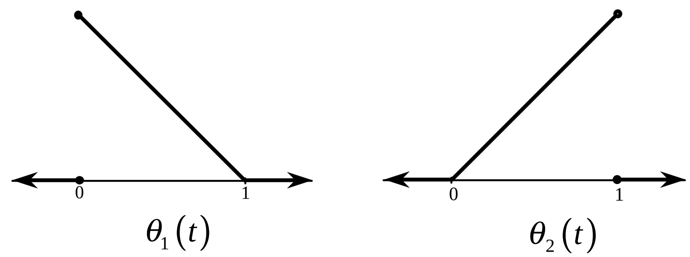
$$\theta_{11}(x,y) = \theta_1(x)\theta_1(y)$$

$$\theta_{12}(x,y) = \theta_1(x)\theta_2(y)$$

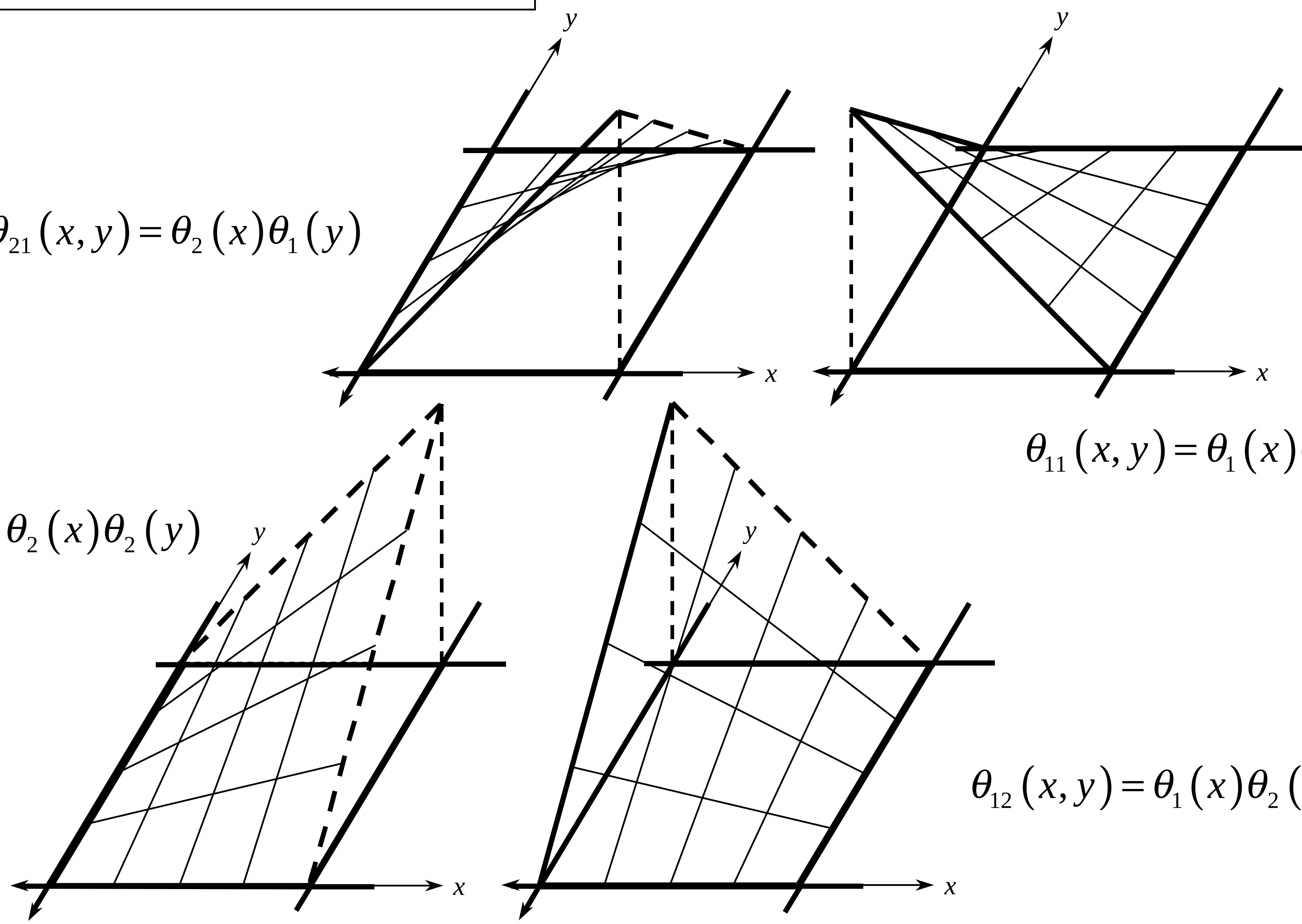
$$\theta_{21}(x,y) = \theta_2(x)\theta_1(y)$$

$$\theta_{22}(x,y) = \theta_2(x)\theta_2(y)$$

The two 1-D elementary basis functions



The four bilinear 2-D elementary basis functions



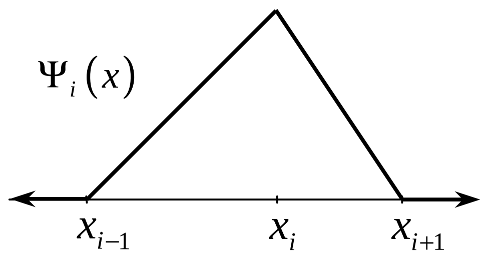
$$\theta_{21}(x,y)=\theta_2(x)\theta_1(y)$$

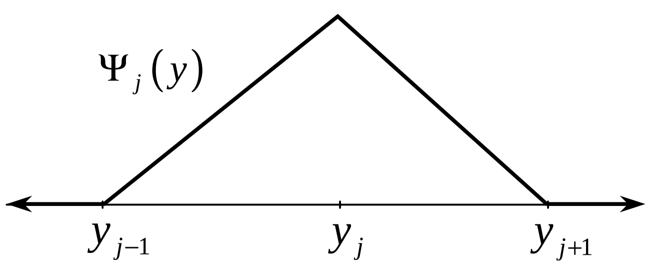
$$\theta_{11}(x,y)=\theta_1(x)\theta_1(y)$$

$$\theta_{22}(x,y)=\theta_2(x)\theta_2(y)$$

$$\theta_{12}(x,y)=\theta_1(x)\theta_2(y)$$

We can construct the 2-D bilinear spline basis functions directly from the 1-D linear spline basis functions

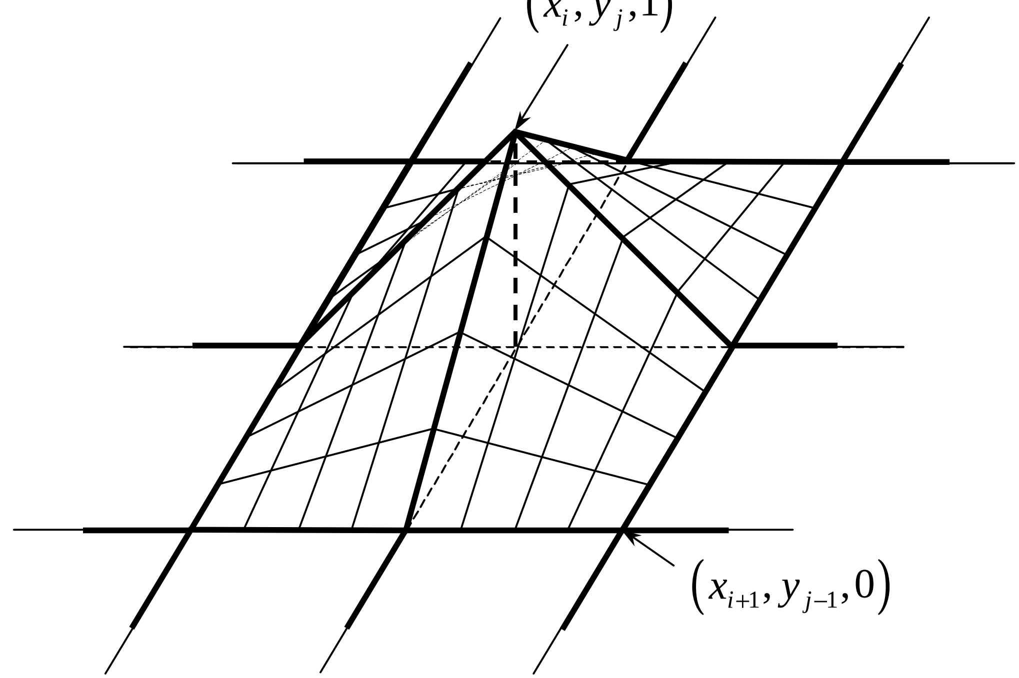




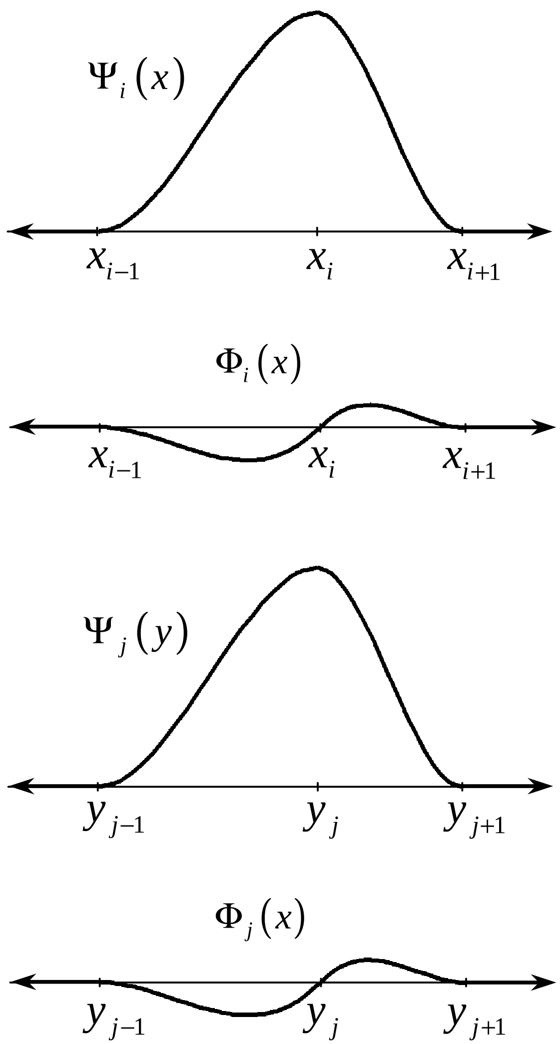
As in the 1-D case we want our basis functions to be zero at all but one of the data points.

We define the ijth 2-D bilinear spline basis function as follows:

$$\Psi_{ij}(x,y) = \Psi_i(x)\Psi_j(y)$$



We can construct the 2-D bicubic spline basis functions directly from the 1-D bicubic spline basis functions



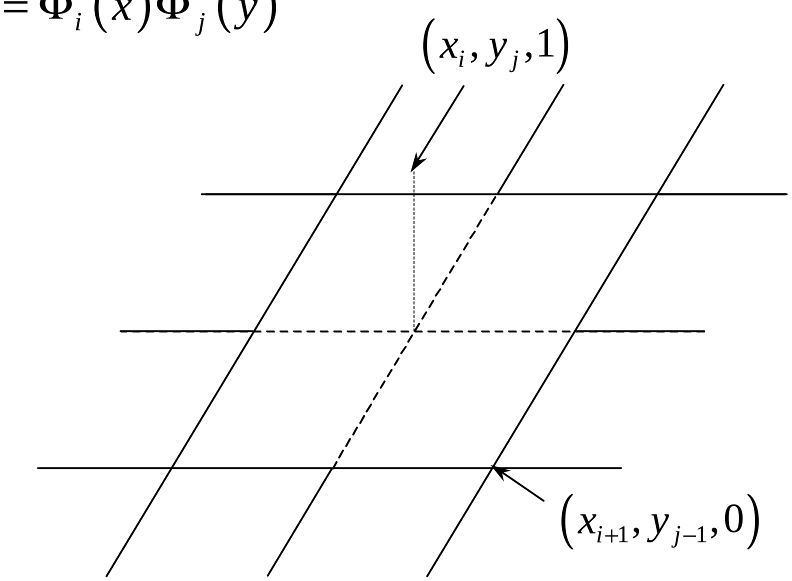
At each data point we define four 2-D bicubic spline basis functions as follows:

$$\Theta^{\Psi\Psi}{}_{ij}(x,y) = \Psi_i(x)\Psi_j(y)$$

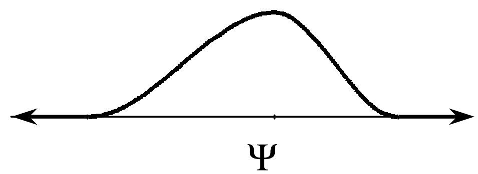
$$\Theta^{\Psi\Phi}{}_{ij}(x,y) = \Psi_i(x)\Phi_j(y)$$

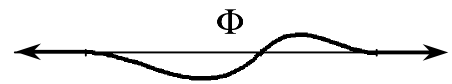
$$\Theta^{\Phi\Psi}{}_{ij}(x,y) = \Phi_i(x)\Psi_j(y)$$

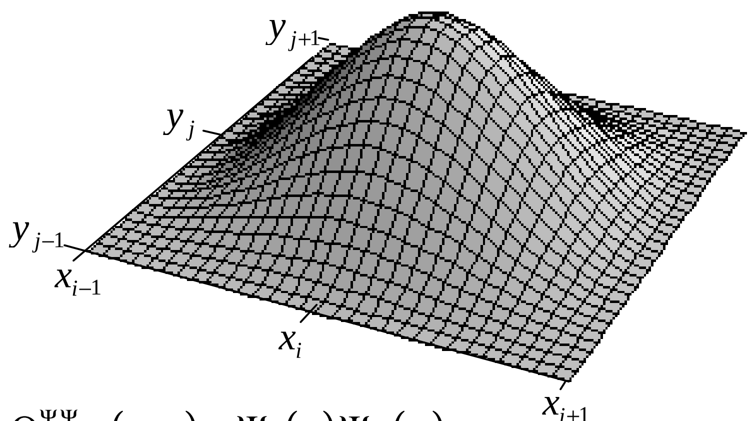
$$\Theta^{\Phi\Phi}{}_{ij}(x,y) = \Phi_i(x)\Phi_j(y)$$



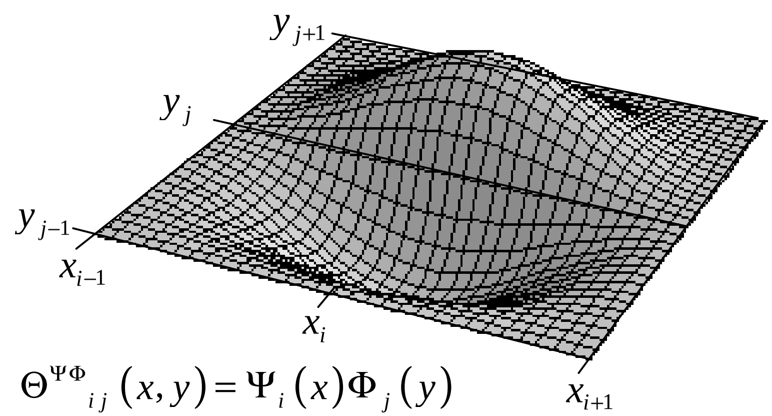
The 2-D bicubic spline basis functions



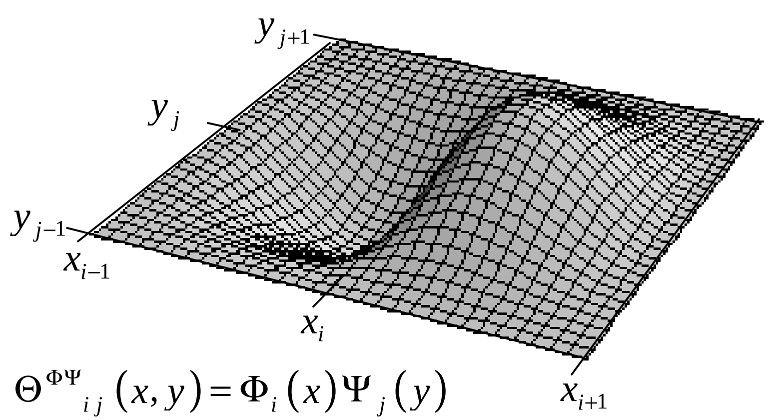




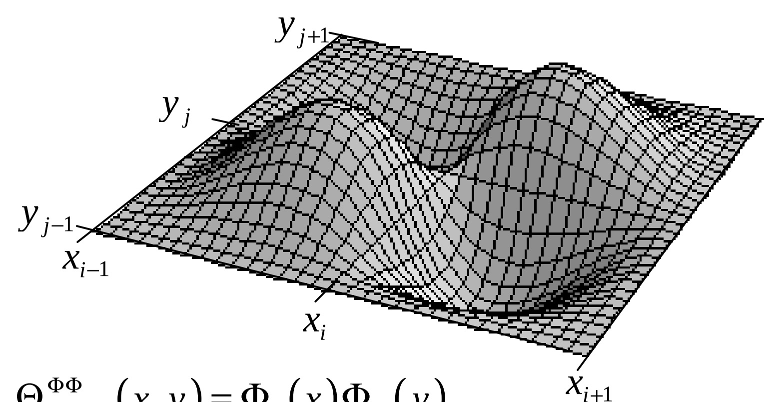
$\Theta^{\Psi\Psi}{}_{ij}(x,y) = \Psi_i(x)\Psi_j(y)$



$\Theta^{\Psi\Phi}{}_{ij}(x,y) = \Psi_i(x)\Phi_j(y)$



$\Theta^{\Phi\Psi}{}_{ij}(x,y) = \Phi_i(x)\Psi_j(y)$



$\Theta^{\Phi\Phi}{}_{ij}(x,y) = \Phi_i(x)\Phi_j(y)$

What have we done?

Connecting the dots – continuous piecewise linear interpolation

- using the data to compute a linear equation for each subinterval
- using a linear combination of "Spline basis functions"

Connecting the dashes – smooth, piecewise cubic, interpolation

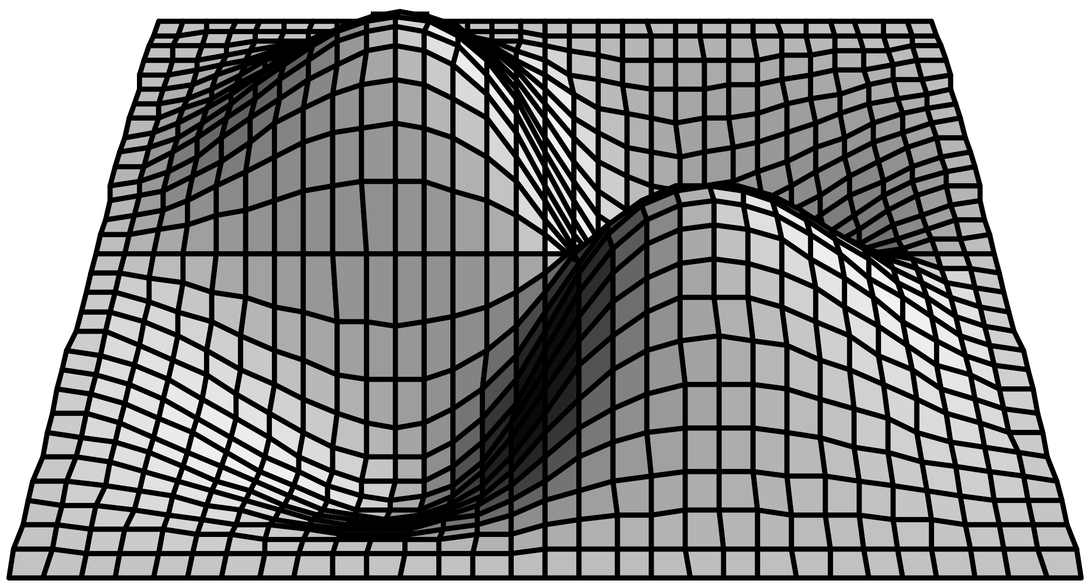
- using slope and function data to compute a cubic equation for each subinterval
- using a linear combination of "Cubic spline basis functions"

Interpolation using 2-D splines

- Bilinear spline basis functions
- Bicubic spline basis functions

Overview – "elementary mathematics from an advanced standpoint"

linear algebra, finite dimensional function spaces, inner products, small support, almost orthogonal bases, tensor products, finite element methods, wavelets, …



The end