

Esercitazione 3

Esercizio 1 (smoothing spline)

Il comando MatLab `csaps` restituisce una spline cubica liscia dato un insieme di dati x e y (doc `csaps`) eseguendo uno smoothing al variare di un parametro p che pesa, mediante combinazione convessa, lo scostamento dai dati e la curvatura (derivata seconda) della funzione. Questo comando trova applicazione quando si desidera “ricostruire” una funzione a partire da una serie di dati affetti da rumore. L’esercizio consiste nel partire da una funzione del tipo

$$f(x) = \sin(t)$$

nell’intervallo $[0, 4\pi]$, aggiungere un rumore gaussiano a media nulla e varianza nota (ad esempio 0.1), ed approssimare la funzione con il comando `csaps` al variare di p .

Per generare un rumore gaussiano si utilizzi il comando MatLab `randn`.

Risultati

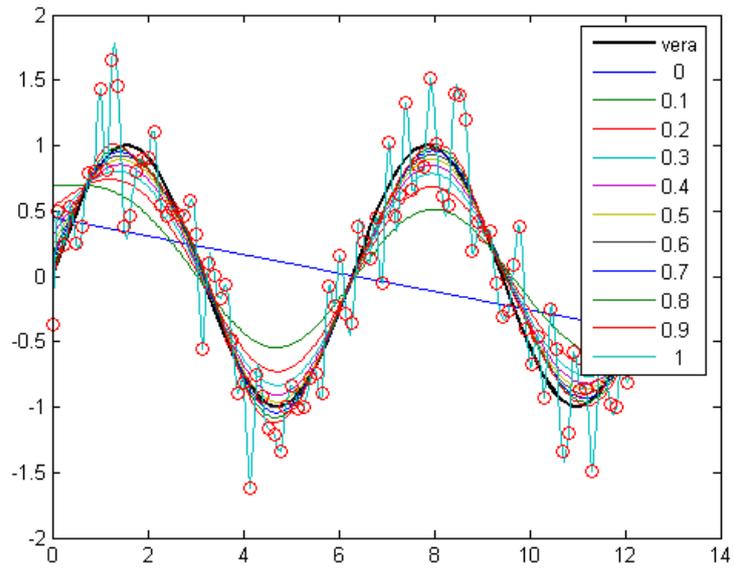


Figura 1: Approssimazione al variare di p .

Richiami di Teoria

Curve di Bèzier

I polinomi di Bernstein di grado n sono definiti da:

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}, \quad i = 0, \dots, n.$$

Ci sono $n + 1$ polinomi di Bernstein di grado n . Per convenienza matematica, si usa porre $B_i^n = 0$, se si verifica che $i < 0$ o $i > n$.

Nei casi semplici otteniamo:

- il polinomio di Bernstein di grado 0 è:

$$B_0^0 = 1$$

- i polinomi di Bernstein di grado 1 sono:

$$B_0^1 = 1 - t \quad B_1^1 = t$$

- i polinomi di Bernstein di grado 2 sono:

$$B_0^2 = (1-t)^2 \quad B_1^2 = 2t(1-t) \quad B_2^2 = t^2$$

- i polinomi di Bernstein di grado 3 sono:

$$B_0^3 = (1-t)^3 \quad B_1^3 = 3t(1-t)^2 \quad B_2^3 = 3t^2(1-t) \quad B_3^3 = t^3$$

I polinomi di Bernstein di grado n , possono essere definiti dalla fusione di due polinomi di Bernstein di grado $n - 1$, cioè il k -esimo polinomio di Bernstein di grado n può essere scritto come:

$$B_k^n = (1-t)B_k^{n-1} + tB_{k-1}^{n-1}$$

Alcune proprietà dei polinomi di Bernstein:

- Non negatività;
- formano una partizione dell'unità.

Algoritmo di De Casteljau

Trattiamo inizialmente le parabole. Consideriamo quindi tre punti b_0, b_1, b_2 , appartenenti a E^3 ed un parametro $t \in \mathbb{R}$. Costruiamo quindi:

$$\begin{array}{l} b_0 \\ b_1 \quad b_0^1(t) = (1-t)b_0 + tb_1 \\ b_2 \quad b_1^1(t) = (1-t)b_1 + tb_2 \quad b_0^2(t) = (1-t)b_0^1 + tb_1^1 \end{array}$$

Osserviamo che inserendo le prime due equazioni (colonna 2) nella terza (colonna 3), otteniamo:

$$b_0^2(t) = (1-t)^2 b_0 + 2t(1-t)b_1 + t^2 b_2 = \sum_{k=0}^2 B_k^2(t) b_k$$

Questa è una espressione quadratica in t , quindi $b_0^2(t)$ traccia una parabola con t che varia da $-\infty$ a $+\infty$. Per t tra 0 e 1, $b_0^2(t)$ è all'interno del triangolo formato da b_0 , b_1 e b_2 ; dove in particolare abbiamo, $b_0^2(0) = b_0$ e $b_0^2(1) = b_2$

Possiamo generalizzare la precedente costruzione della parabola generando una curva polinomiale di grado arbitrario n : dati b_0, b_1, \dots, b_n e $t \in [0, 1]$ fissiamo:

$$b_i^r(t) = (1-t)b_i^{r-1}(t) + t b_{i+1}^{r-1}(t) \quad r = 1, \dots, n, \quad i = 0, \dots, n-r$$

e $b_i^0 = b_i$. Quindi $b_0^n(t)$ è il punto che come parametro usa t nella curva di Bézier b^n .

Il poligono P formato da b_0, \dots, b_n è chiamato poligono di Bézier o poligono di controllo della curva b^n . In modo simile, i vertici del poligono b_i , sono chiamati punti di controllo o punti di Bézier.

I coefficienti b_i^r possono essere scritti in forma matriciale

$$\begin{aligned} b_0 &= b_0^0 \\ b_1 &= b_1^0 \quad b_0^1 \\ b_2 &= b_2^0 \quad b_1^1 \quad b_0^2 \\ b_3 &= b_3^0 \quad b_2^1 \quad b_1^2 \quad b_0^3 = b^3 \end{aligned}$$

dove il calcolo dei vari coefficienti avviene per colonna e deve essere ripetuta al variare di t .

Quindi, dati $n+1$ punti definiti punti di controllo b_0, b_1, \dots, b_n , la curva di Bézier di grado n è definita come:

$$q(t) = \sum_{k=0}^n B_k^n(t) b_k, \quad t \in [0, 1]$$

dove i B_k^n sono i polinomi di Bernstein di grado n .

Ad esempio, prendendo i punti di controllo $b_0(1, 2)$ e $b_1(3, 4)$, i polinomi di Bernstein diventano $1-t$ e t e quindi la curva di Bézier diventa

$$\begin{pmatrix} x \\ y \end{pmatrix} = (1-t)b_0 + t b_1 = (1-t) \begin{pmatrix} 1 \\ 2 \end{pmatrix} + t \begin{pmatrix} 3 \\ 4 \end{pmatrix}$$

che sviluppata

$$\begin{aligned} x &= 1(1-t) + 3t = 1 + 2t \\ y &= 2(1-t) + 4t = 2 + 2t \end{aligned}$$

porta alle due equazioni parametriche di un segmento di linea su un piano per $t \in [0, 1]$.

Esempio di codifica ed utilizzo è riportato nelle seguenti function.

```

function val = deCasteljau(puntiBezier,t)
% Algoritmo di De Casterljau

% schema algoritmo
% b_0^0, b_1^0, b_2^0, ..., b_n^0
% b_0^1, b_1^1, ....., b_{n-1}^1
% b_0^2, b_1^2, ...
% ...

bt = puntiBezier;

[dim,numpunti] = size(bt);
for r = 2:numpunti,
    for i=1:numpunti-r+1,
        %  $b_i^r = (1-t)b_{i-1}^{r-1} + t b_{i+1}^{r-1}$ 
        bt(:,i) = (1-t)*bt(:,i) + t*bt(:,i+1);
    end
end
val = bt(:,1);

% ESEMPIO DI CURVA DI BEZIER
% -----

clear all
close all
clc

puntiBezier = [1 1 0 -1/2; 0 1 1 1; 0 0 0 1/2]; % punti in uno spazio 3D
np = size(puntiBezier,2); % numero di punti del poligono di controllo
dim = size(puntiBezier,1); % dimensione
sudd = 100; % suddivisioni di [0,1]
t = linspace(0,1,sudd); % t
curvaBezier = zeros(dim,sudd); % succ. dei punti della curva di Bezier
% valutazione della curva di Bezier per ogni istante del parametro
for i=1:sudd,
    % valutazione del punto
    bt = deCasteljau(puntiBezier,t(i));
    % salvataggio del punto
    curvaBezier(:,i) = bt;
end

% disegno della curva come spezzata e del poligono di controllo
xBezier = curvaBezier(1,:);
yBezier = curvaBezier(2,:);
zBezier = curvaBezier(3,:);

plot3(xBezier,yBezier,zBezier,'k');

```

```

hold on
plot3(puntiBezier(1,:),puntiBezier(2,:),puntiBezier(3,:), 'ro');
hold off
grid on
xlabel('x')
ylabel('y')
zlabel('z')

```

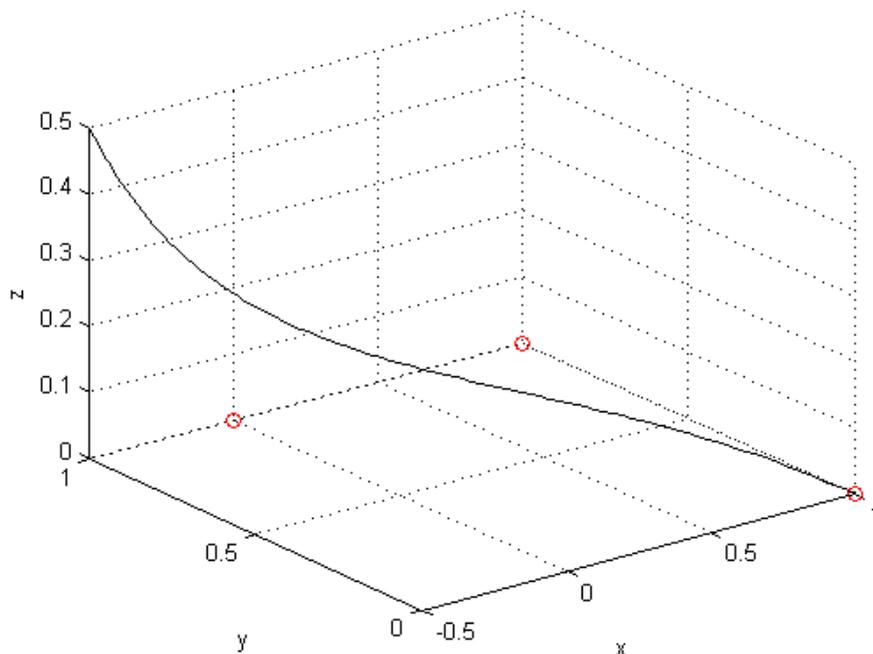


Figura 2: Curva di Bézier.

Una proprietà molto importante delle curve di Bézier è l'invarianza affine, ovvero se viene applicata una trasformazione affine

$$\Phi(x) = Ax + c$$

il risultato è dato dalle immagini affini dei suoi punti di controllo

$$\Phi(q(t)) = \Phi \left(\sum_{k=0}^n B_k^n(t) b_k \right) = \sum_{k=0}^n \Phi(b_k) B_k^n(t)$$

riducendone notevolmente la complessità nell'applicazione di queste trasformazioni.

Esempio di queste trasformazioni sono la rotazione, la traslazione, la scalatura,

Esercizio 2

Applicare la rotazione alla curva di Bézier definita dai seguenti punti di controllo $(0, 0)$, $(0, 1)$, $(1, 1)$, $(1, 0)$, $(1/2, 0)$, $(1/2, 1/2)$.

Nello spazio, la rotazione attorno all'asse z è della forma

$$\begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Ripete l'esercizio con un'altra trasformazione affine a scelta dello studente (es. traslazione, scalatura, ...).

Nello spazio, la scalatura attorno è una matrice del tipo

$$\begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{pmatrix} .$$

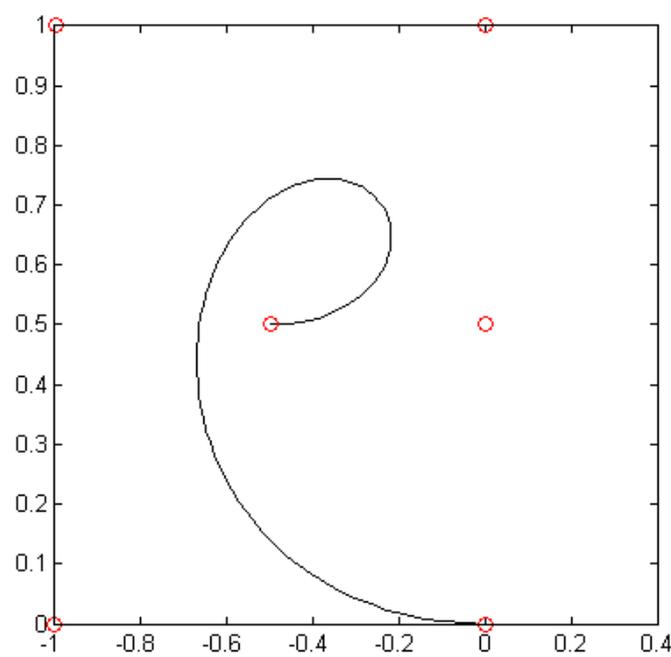
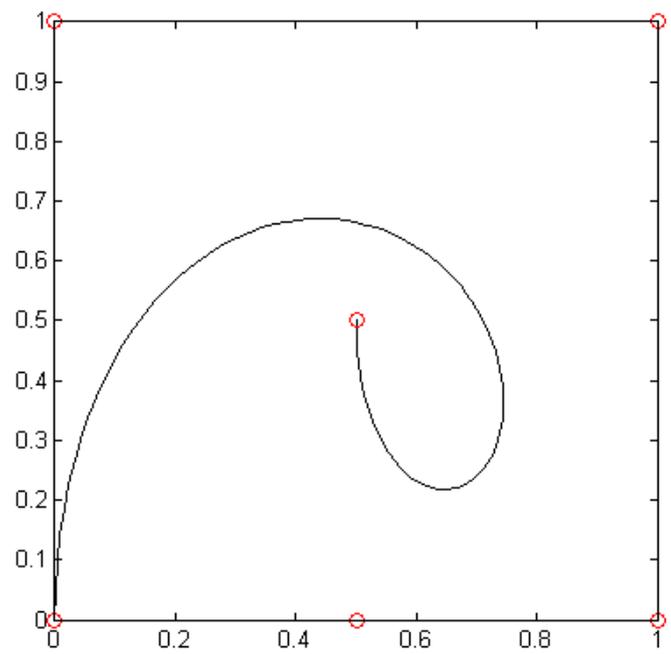


Figura 3: Curva di Bézier e sua rotazione di 90°.

Richiami di teoria

Con il termine "Degree elevation" si indica il processo in cui presa una curva di Bézier con k punti di controllo si ottiene la stessa curva di Bézier ma espressa con $k + 1$ punti di controllo e quindi con una curva di grado maggiore. Ossia, data una curva di Bézier $q(t)$ di grado k , definita nei punti di controllo b_i con $i = 0, \dots, k$, vogliamo definire dei nuovi punti di controllo \hat{b}_i , con $i = 0, \dots, k + 1$, in modo che la curva di Bézier $\hat{q}(t)$ di grado $k + 1$ sia uguale alla curva di Bézier $q(t)$ di grado k , ovvero $\hat{q}(t) = q(t)$ per tutti i valori assegnati di t .

Per capire meglio il problema, si può analizzare il caso della curva quadratica in cui $k = 2$ e con i punti di controllo b_0, b_1, b_2 . Poichè in questa curva $q(0) = b_0$ e $q(1) = b_2$, per la curva trasformata dovranno valere le seguenti uguaglianze: $\hat{p}_0 = b_0$ e $\hat{p}_2 = b_2$ (in questo modo la curva cubica ha gli estremi coincidenti a quella quadratica). Oltre a questo, anche le derivate all'inizio e alla fine della curva devono essere uguali a: $q'(0) = 2(b_1 - b_0)$ e $q'(1) = 2(b_2 - b_1)$.

Infine, per la derivata della curva cubica devono valere le relazioni:

$$\hat{b}_1 = \hat{b}_0 + \frac{1}{3}q'(0) = \frac{1}{3}b_0 + \frac{2}{3}b_1,$$

$$\hat{b}_2 = \hat{b}_3 - \frac{1}{3}q'(1) = \frac{2}{3}b_1 + \frac{1}{3}b_2.$$

Nel caso generale, supponiamo di avere una curva $q(t)$ di grado k , con punti di controllo b_0, \dots, b_k . Per ottenere i $k + 1$ punti di controllo $\hat{b}_0, \dots, \hat{b}_{k+1}$ che definiscono la curva $\hat{q}(t)$ identica a $q(t)$, devono valere le seguenti relazioni:

$$\hat{b}_0 = b_0 \quad , \quad \hat{b}_{k+1} = b_k$$

$$\hat{b}_i = \frac{i}{k+1}b_{i-1} + \frac{k-i+1}{k+1}b_i.$$

Esercizio 3

Con riferimento ai dati dell'esercizio 2 eseguire il degree elevation.

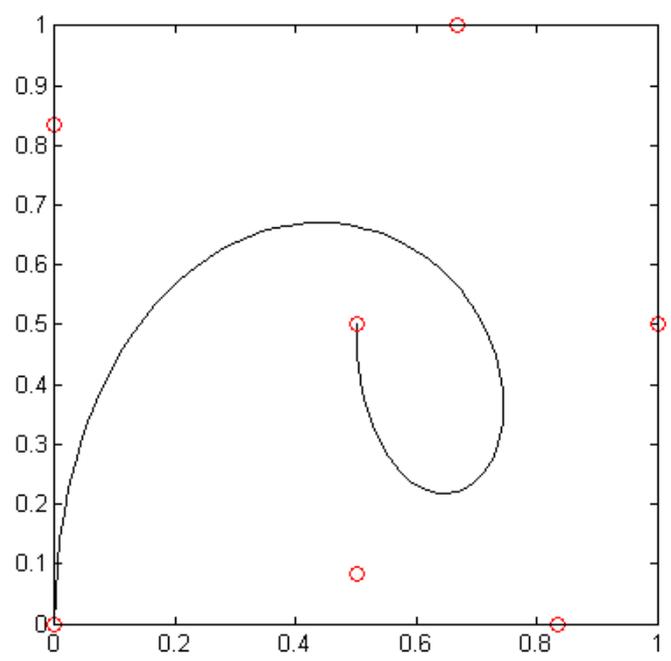
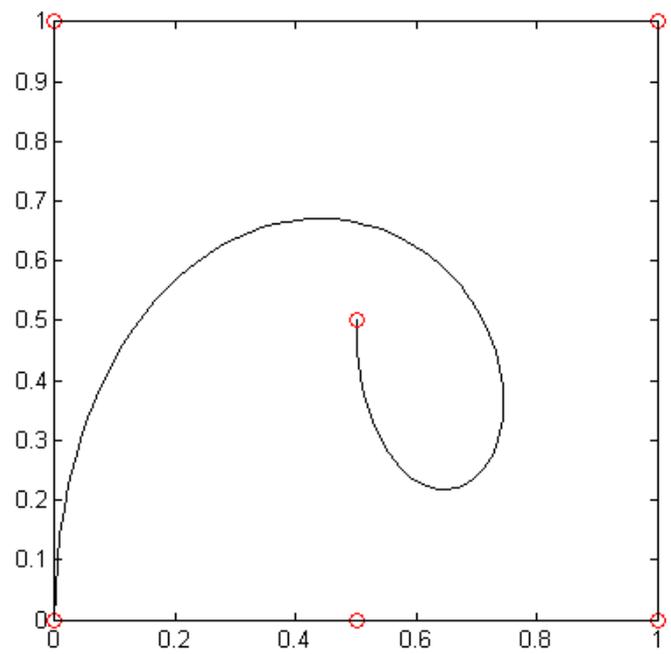


Figura 4: Curva di Bézier originale e esempi di degree elevation.

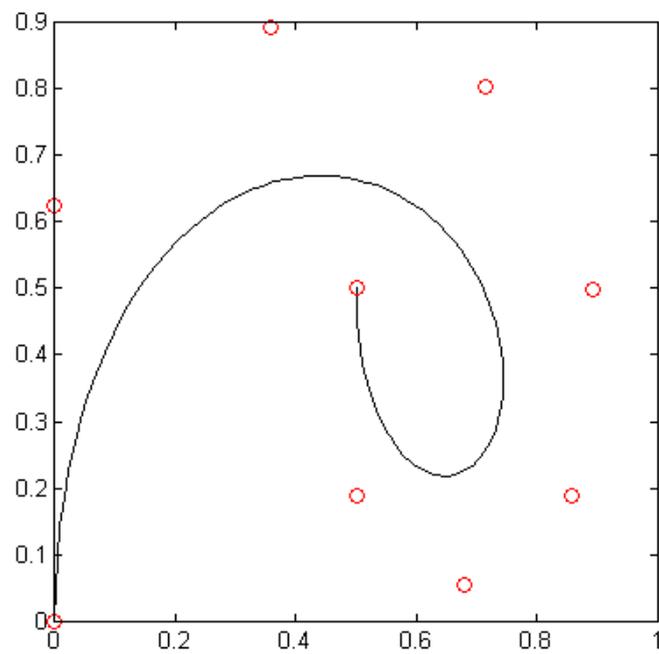
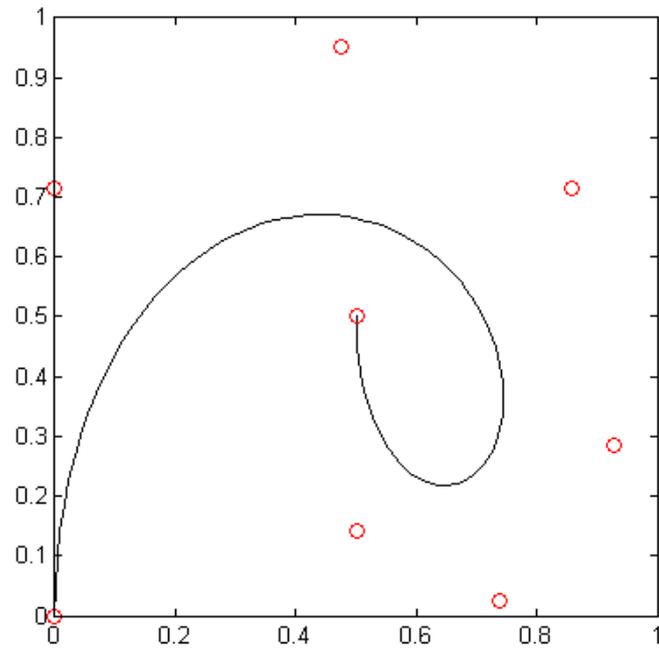


Figura 5: Esempi di degree elevation.

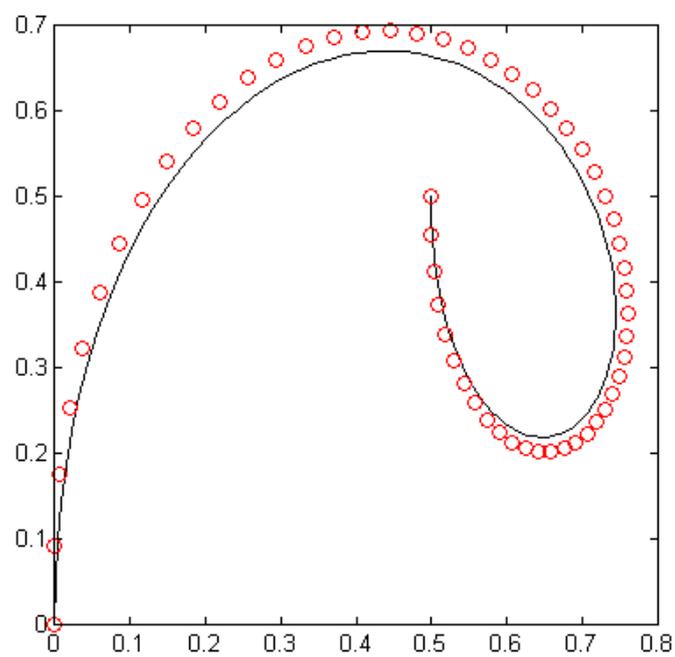
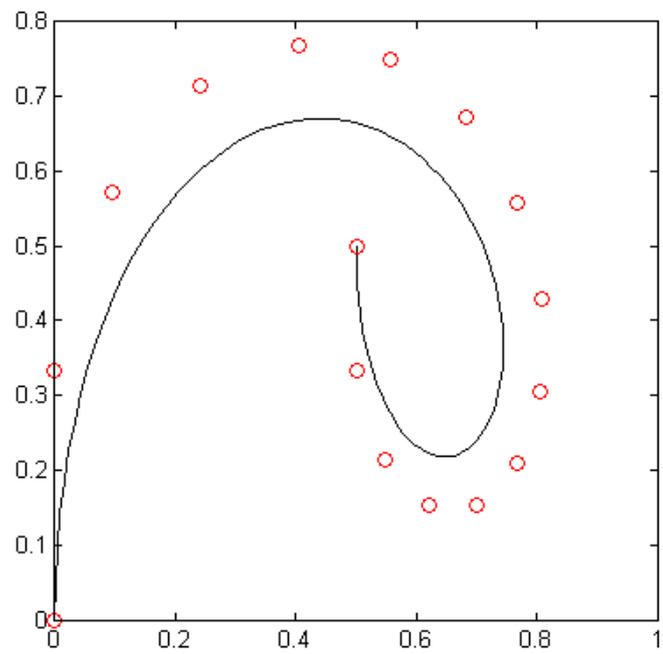


Figura 6: Esempi di degree elevation.