



PROIECT IDENTIFICAREA SISTEMELOR 1

Modelarea unei functii necunoscute

Nume Studenti:

Ciceu Robert

Florea Ioana-Valentina

Tabircea Georgiana-Maria

Grupa:30135

Semigrupa:1

Set de date: `proj_fit_14.mat`

Cuprins

1.Introducere

2.Descrierea structurii si a procedurii

3.Caracteristici esentiale ale solutiei

4.Rezultate de acordare

5.Grafice reprezentative

6.Concluzie

INTRODUCERE

Această prezentare se concentrează pe dezvoltarea unui aproximant polinomial pentru o funcție necunoscută, statică și neliniară, folosind date de intrare-ieșire furnizate. Scopul este să dezvoltăm un model care să poată estima funcția dată, în ciuda naturii sale neliniare.

Acest proces de modelare se bazează pe două seturi distincte de date:

- Setul de identificare, notat cu "id," folosit pentru antrenarea modelului.
- Setul de validare, notat cu "val," utilizat pentru evaluarea performanței modelului dezvoltat.

DESCRIEREA STRUCTURII SI A PROCEDURII

Aproximatorul polinomial trebuie sa aiba gradul configurabil, iar antrenarea modelului consta în gasirea vectorului optim de parametri pentru care polinomul se apropie cel mai mult de functia necunoscuta, dată de setul de date de identificare.

Structura aproximatorului polinomial:

Forma generala: $y(k) = [\phi_1(k), \phi_2(k), \dots, \phi_n(k)] \cdot [\theta_1, \theta_2, \dots, \theta_n]^T + e(k)$

Unde:

- $\phi_1(k), \phi_2(k), \dots, \phi_n(k)$ sunt regresorii
- $\theta_1, \theta_2, \dots, \theta_n$ sunt vectori necunoscuți
- $e(k)$ este eroarea modelului

Procedura de Găsire a Parametrilor:

Antrenarea modelului constă în găsirea vectorului optim de parametri θ pentru care polinomul $y(n)$ se apropie cât mai bine de funcția necunoscută. Aceasta se realizează prin minimizarea erorii pătratice medii (MSE) între ieșirile estimate și cele reale ale funcției.

În continuare va vom prezenta codul care sta la baza regresiei liniare având comentarii explicite.

```
load("proj_fit_14.mat")
id.X; %date identificare
id.Y; %date identificare
val.X; %date validare
val.Y; %date validare
%mesh urile sunt folosite pentru crearea graficelor initiale
%tridimensionale
figure;
mesh(id.X{1},id.X{2},id.Y)
mesh(val.X{1},val.X{2},val.Y)
title("Grafic Initial")

phi = [];
phi_val=[];
theta=[];
n=30; %gradul polinomului 30
MSE1=zeros(1,n);
MSE2=zeros(1,n);

for n=1:30 %luam n de la 1 la 30 pentru a vedea evolutia
%matricea phi de identificare
N=length(id.X{1}); %nr de puncte din setul de identificare
il = 1; %indexul liniei
for i = 1:N
for j = 1:N
ic = 1; %indexul coloanei
for m = 0:n %putere
for w = 0:n %putere
if m + w <= n %puterile adunate trebuie sa fie mai mici sau egale cu gradul polinomului
if m == 0 && w == 0 %conditie pt prima coloana
phi(il, ic) = 1; % Prima coloană este întotdeauna 1
else
phi(il, ic) = id.X{1}(i)^m * id.X{2}(j)^w; % Calculează valorile pentru celelalte coloane
end
ic = ic + 1; %trecem la urmatoarea coloana
end
end
end
```

2023-2024

```
end
end
il = il + 1; %trecem la urmatoarea linie
end
end
% matricea phi de validare se face la fel ca cea de identificare
M=length(val.X{1}); %nr de puncte din setul de validare
il_val = 1;
for i= 1:M
for j= 1:M
    ic_val = 1;
for m= 0:n
for w = 0:n
    if m+w <= n
    if m== 0 && w == 0
        phi_val(il_val, ic_val) = 1; % Prima coloană este întotdeauna 1
    else
        phi_val(il_val, ic_val) = val.X{1}(i)^m* val.X{2}(j)^w; % Calculează valorile pentru
        celelalte coloane
    end
    ic_val = ic_val + 1;
end
end
end
il_val = il_val + 1;
end
end

% Calcularea vectorului de parametri theta
Y_id=reshape(id.Y,N*N,1);
theta=phi\Y_id;

% Estimarea ieşirilor aproximative pentru setul de identificare
y_aprox_id=phi*theta;
y_aprox1_id=reshape(y_aprox_id,N,N);

% Estimarea ieşirilor aproximative pentru setul de validare
Y_val=reshape(val.Y,M*M,1);
y_aprox_val=phi_val*theta;
y_aprox1_val=reshape(y_aprox_val,M,M);

% Afişarea graficelor doar pentru n=4 gradul minim pentru a nu se
% supraantrena aproximările
if n==4
    figure
    mesh(y_aprox1_id)
    title("Aproximare si identificare")

    figure
    mesh(y_aprox1_val)
    title("Aproximare si validare")
end

%MSE pentru identificare
e1=ones(N, 1); %matrice care sa stocheze toate valorile e1
for k=1:N
e1(k)=id.Y(k)-y_aprox1_id(k);
end
MSE1(n)=1/N*sum(e1.^2);

%MSE pentru validare
```

2023-2024

```
e2=ones(M, 1); %matrice care sa stocheze toate valorile e2
for K=1:M
e2(K)=val.Y(K)-y_aprox1_val(K);
end
MSE2(n)=1/M*sum(e2.^2);
%functie pentru calcularea punctului minim si gradului minim a erorii medii
%patratice de validare
[minim,min_gr]=min(MSE2);
end
%afisarea graficelor pentru erori
figure
plot(1:n,MSE1)
title("Eroarea Medie Patratice 1")

figure
plot(1:n,MSE2)
title("Eroarea Medie Patratice 2")
%afisarea in consola a punctului si gradului minim
fprintf('MSE MINIM ESTE IN PUNCTUL:%f \nmin grad%d',min(MSE2),min_gr)
```

CARACTERISTICI ESENTIALE ALE SOLUTIEI

Datele de identificare și validare sunt reprezentate de variabilele $id.X$, $id.Y$, $val.X$, și $val.Y$.

Odata ce datele sunt incarcate, utilizam functia “mesh()” pentru a afisa graficele initiale tridimensionale.

Mai apoi, am construit matricile ϕ si ϕ_{val} , care sunt esentiale pentru procesul de identificare si validare a modelului. Aceasta constructie presupune generarea si combinarea unor puteri ale datelor de intrare, cu scopul de a crea un set complex de caracteristici utilizate pentru modelare.

Am determinat parametrii modelului, reprezentati de vectorul θ , prin rezolvarea unui sistem linear in sensul celor mai mici patrate, realizata prin operatorul \ (“backslash()”)

Dupa determinarea parametrilor, se efectueaza aproximarea datelor de identificare si validare.

Urmatoarea etapa consta in calcularea erorilor medii patratice(MSE), unde MSE reprezinta o masura a discrepantei intre datele reale si cele approximate de model.

In final, se afiseaza graficele erorilor pentru identificare si validare si se evidentiaza punctul in care MSE pentru validare atinge minimul, oferind astfel o evaluare concludenta a performantei modelului.

REZULTATE DE ACORDARE

În continuare, vom analiza rezultatele obținute în urma variației numărului de termeni 'n', examinând eroarea medie pătratică (MSE) în intervalul de la 1 la 30. Am ales gradul polinomului 30 deoarece așa se observa cel mai bine la eroarea medie patratice de validare ca va creste.

Fig 1: Se observa o scadere a erorii pana in $n=4$. Pentru $n>4$ sistemul va avea o eroare medie patratice constanta

Fig 2: Se observa o scadere a erorii pana in $n=4$. Dupa eroarea medie patratice este constanta pana In punctul 23 unde observam o urcare a erorii.

Valoarea minima a parametrului este $n=4$, pentru ca y_{aprox_val} se apropie cel mai bine de valorile reale y pe setul de date de identificare.

Fig 1

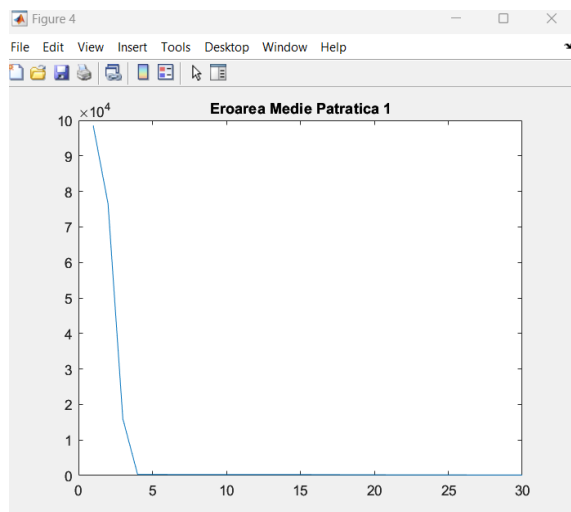
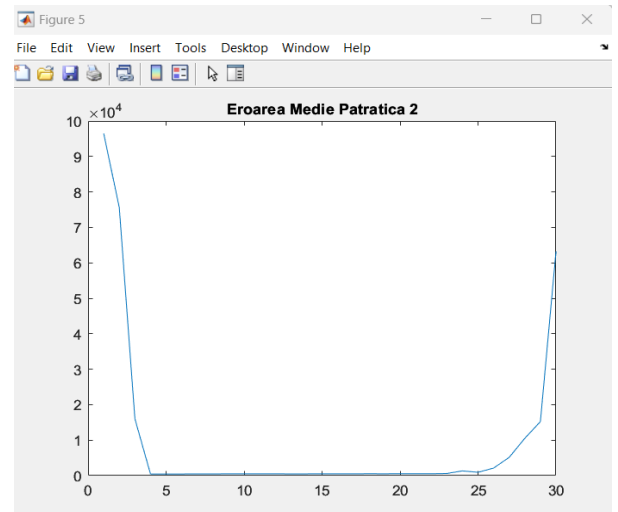


Fig 2



GRAFICE REPREZENTATIVE

Fig 3(Date identificare): Acest grafic reflecta procesul de ajustare a modelului la setul de identificare. Pentru valoarea optima a parametrului "n", in acest caz, $n=4$, modelul ofera o aproximatie semnificativa si precisa a datelor reale. Acest aspect este evidentiat prin modul in care suprafata graficului se apropie de forma datelor de identificare.

Fig 4(Date validare): Modelul alege un numar optim de parametrii, evitand astfel atat sub-ajustarea,cat si supra-ajustarea si manifestandu-se intr-o generalizare eficienta pe datele de validare.

Fig 3

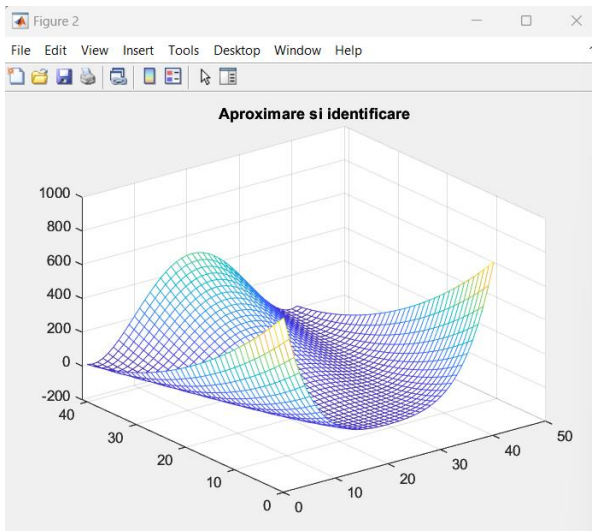
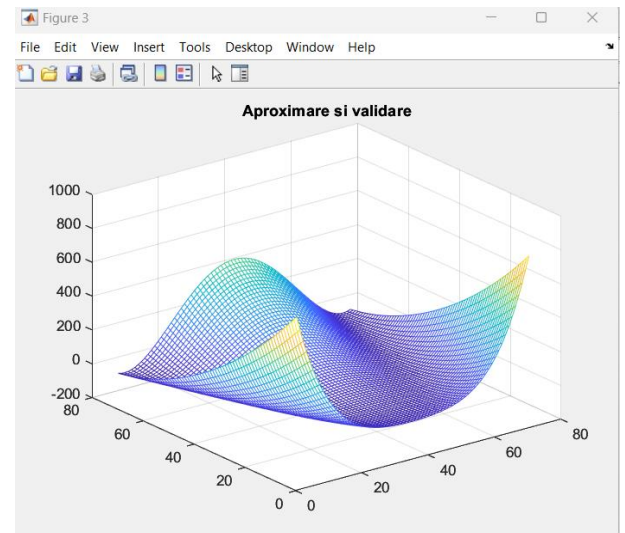


Fig 4



CONCLUZIE

În concluzie, codul implementează o metodă de identificare a unui sistem utilizând regresia liniară polinomială de gradul n , evaluată prin eroarea medie pătratică (MSE). Un aspect important al analizei este găsirea gradului optim al polinomului, care a fost determinat pe baza minimului local al MSE pe setul de validare.

Sugestie de îmbunătățire și optimizarea codului:

Se poate îmbunătăți eficiența codului prin utilizarea unor funcții și vectorizarea operațiilor.