PROIECT PARTEA 1

Costin Denisa-Nicoleta ~17

Continut:

1. Descrierea problemei.
2. Ce este un aproximator polinomial?
3. Descrierea structurii aproximatorului.
4. Caracteristici esentiale ale solutiei noastre.
5. Rezultate de acordare.
6. Grafice reprezentative.
7. Concluzia.
8. Fragmente de cod.

Descrierea problemei

Obiectivul acestui proiect este de a determina o functie polinomiala de ordin m, care se foloseste de niste date de identificare care trebuie sa aproximeze relatia dintre variabilele furnizate si rezultatul aproximatorului polinomial . Modelul este evaluat cu ajutorul unui set de date de validare pentru a verifica cat de bine se aproximeaza functia pe datele primite initial.

Ce este un aproximator polinomial?

Regresia polinomiala este o tehnica utilizata in identificarea sistemelor pentru a modela si a studia relatia dintre variabilele de intrare si cele de iesire, folosind un aproximator polinomial.

Descrierea structurii aproximatorului

1.Am construit matricea regresorilor, phi, folosindu-ne de datele de identificare (xid1, xid2), urmarind pattern-ul specific.

2.Determinam coeficientii matricei theta ai functiei necunoscute prin folosirea metodei celor mai mici patrate.

3. Am creat matricea regresorilor, phival, folosindu-ne de datele de validare(xval1,xval2).

4.In final, am obtinut functia de aproximare, yprim, facand operatia de inmultire intre phival si theta.

Caracteristici esentiale ale solutiei noastre

Algoritmul pentru determinarea matricii phi a fost realizat atat pentru setul de date de identificare cat si pentru cele de validare, acesta functionand pentru orice valoare furnizata lui m (gradul polinomului).

Algoritmul gasit de noi se bazeaza pe patru for-uri, simple, doua care parcurg elementele din matricea initiala si doua care lucreaza cu puterile elementelor si combinatiile dintre acestea.

Rezultate de acordare

MSE (Eroarea medie patratica), reprezinta o valoare care ne ajuta sa ne dam seama cat de aproape sau de departe sunt valorile prezise de cele initiale.

Calculul MSE-urilor se realizeaza luand diferenta dintre valoarea prezisa si cea reala, la patrat, le adunam pe toate si apoi le impartim la numarul total.

Cu cat un MSE este mai mic, cu atat indica o performanta mai buna a sistemului. (Fig.1)

Graficul MSE-urilor

Cu ajutorul acestui grafic am determinat ca cel mai optim grad al polinomului este 31, aici fiind cel mai mic MSE: 0.0051

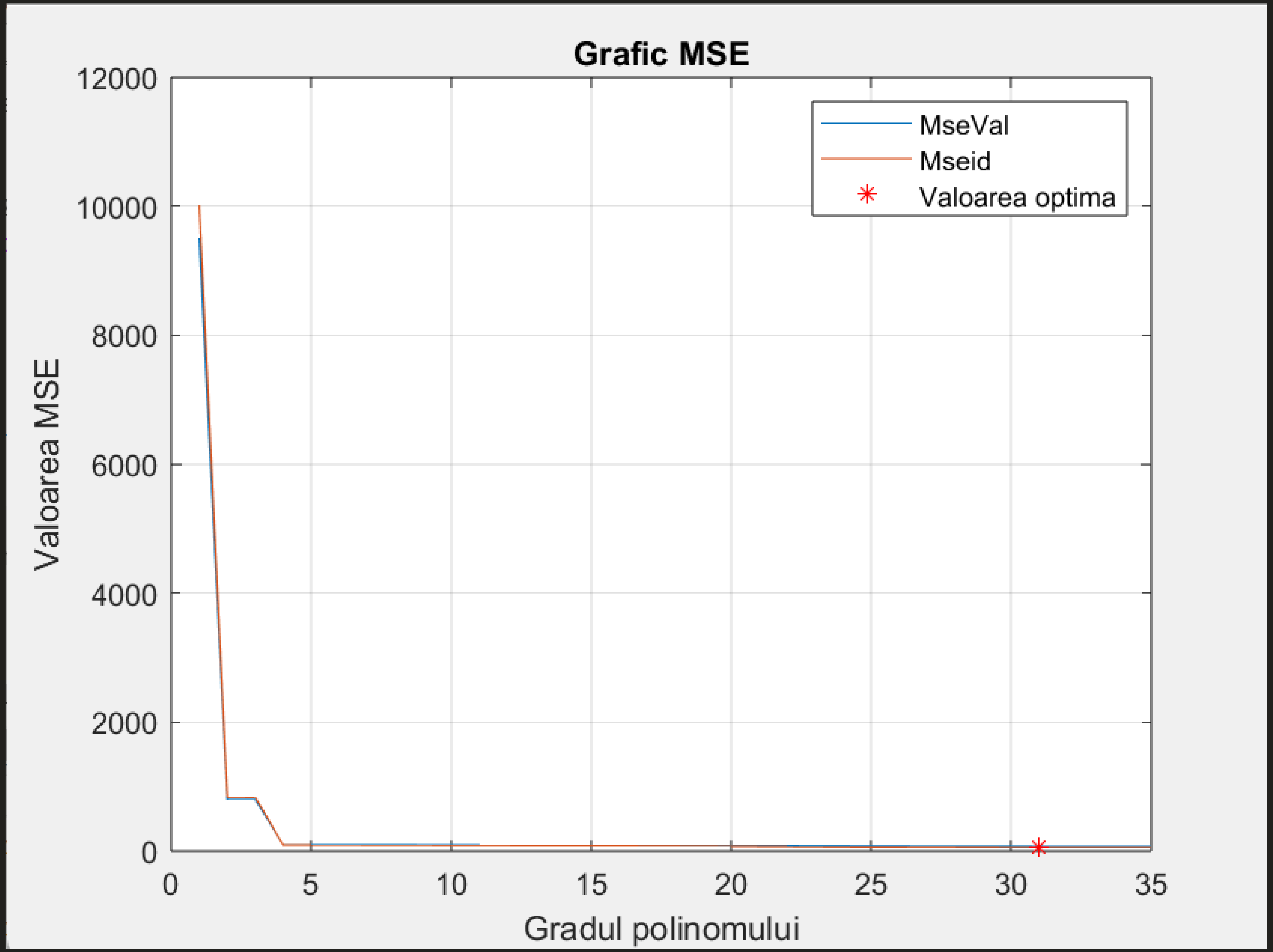

Fig.1

m	m=21	m=22	m=23	m=24	m=25	m=26	m=27	m=28	m=29	m=30
MSE	0,0071	0,0070	0,0069	0,0068	0,0065	0,0062	0,0059	0,0056	0,0054	0,0053

tabelul 1

m	m=31	m=32	m=33	m=34	m=35	m=36	m=37	m=38	m=39	m=40
MSE	0,0051	0,0052	0,0052	0,0052	0,0052	0,0052	0,0052	0,0053	0,0053	0,0055

tabelul 2

Reprezentarile grafice ale functiilor

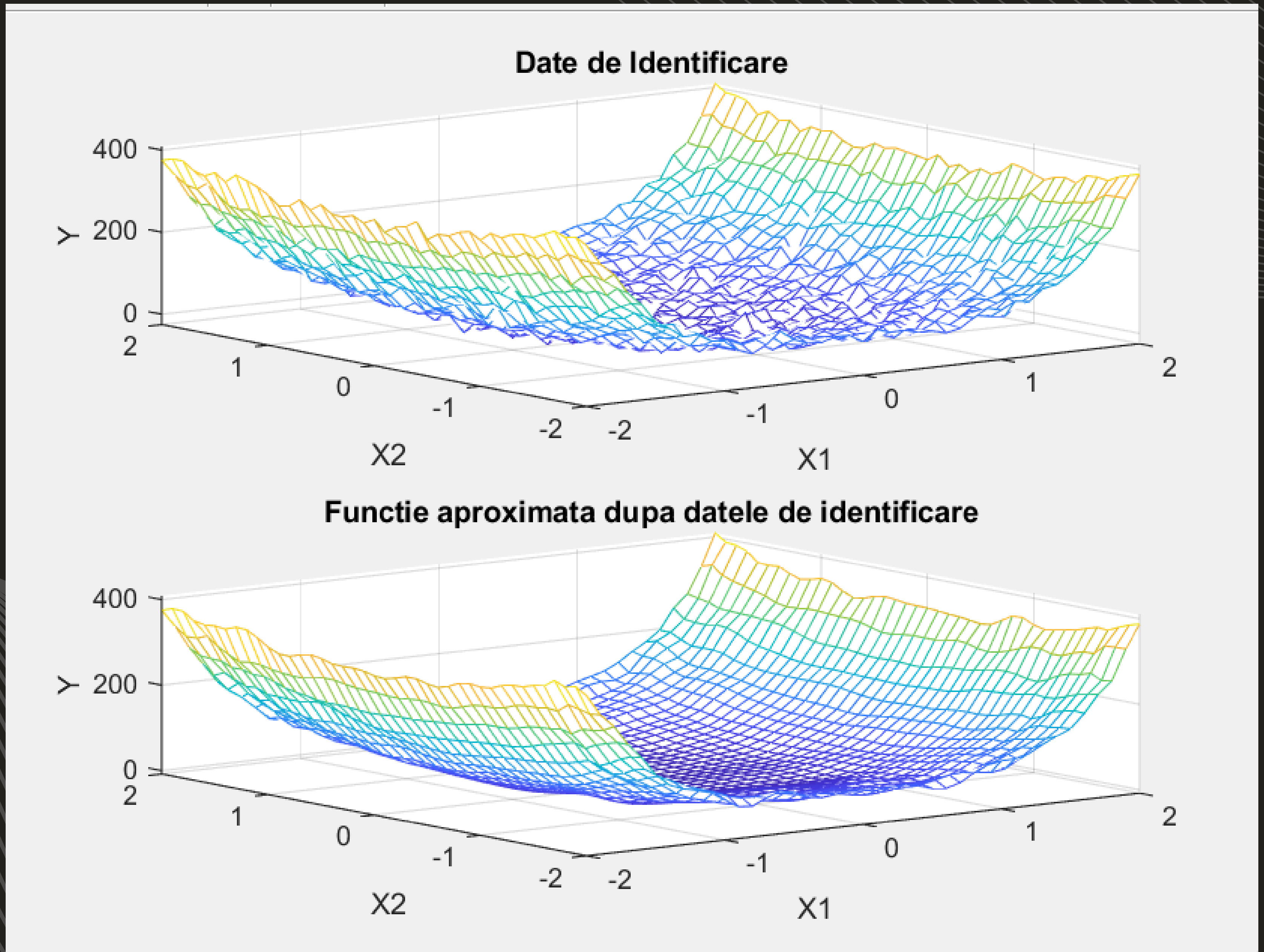

Fig.2

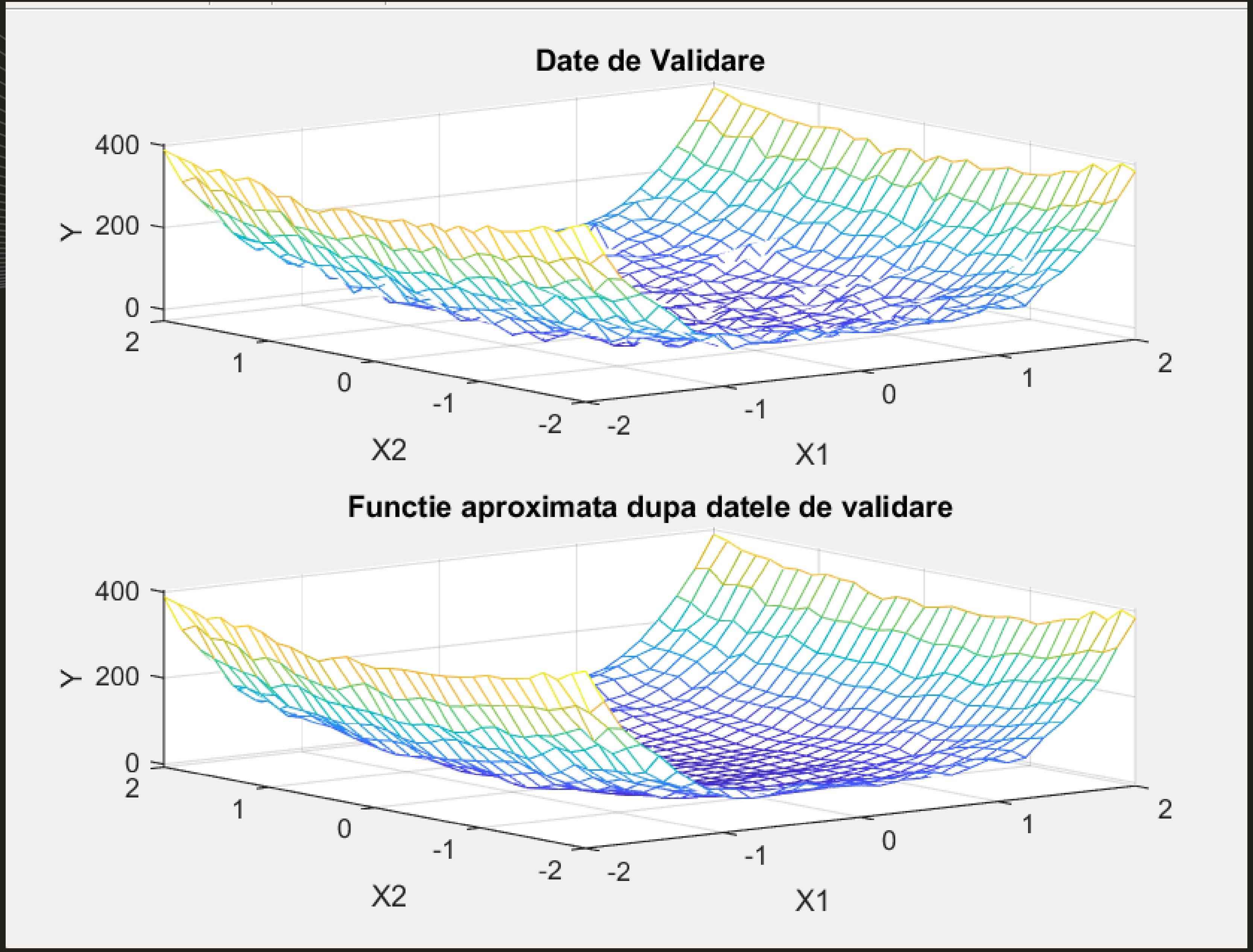

Fig.3

Concluzia

Modelul de regresie liniară estimează relația dintre variabilele x1, x2 și ieșirea y într-un mod simplu și interpretabil, cu rezultate bune atunci când eroarea de predicție este mică. Deși sensibil la zgomot și la relații neliniare, regresia liniară este o metodă utilă pentru a aproxima funcții necunoscute în diverse situații.

Matricea phi pentru datele de identificare:

```
%%Calculul erorii pătratice medii (MSE) pentru identificare și validare
m= 31;

MSEid= zeros(1, m);
MSEval= zeros(1, m);

for l=1:m
    f = factorial(l+2) / (factorial(l) * factorial(l+2 - l));
    c=int32(f);
    phi= ones(length(xid1)*length(xid2), c);

philinie=1;
for i=1:length(xid1)
    for j=1:length(xid2)
        phicoloana=1;
        for p1=0:l
            for p2=0:l
                if p1+p2<=l
                    phi(philinie, phicoloana)= xid1(i).^p1*xid2(j).^p2;
                    phicoloana= phicoloana+1;
                end
            end
        end
        philinie= philinie+1;
    end
end

yid= reshape(yid, 1, length(xid1)*length(xid2));
thetaid= phi\yid';
```

Matricea phi pentru datele de validare:

```
%construim matricea de regresie pentru validare
phival= ones(length(xval1)*length(xval2), c);

philinie=1;% indexul liniilor
for i=1:length(xval1)
    for j=1:length(xval2)
        phicoloana=1;% indexul coloanelor
        for p1=0:m
            for p2=0:m
                % condiție pentru ca gradul maxim să fie `m`
                if p1+p2<=m
                    phival(philinie, phicoloana)= xval1(i).^p1*xval2(j).^p2;
                    phicoloana= phicoloana+1;
                end
            end
        end
        philinie= philinie+1;
    end
end

yval=reshape(yval, 1, length(xval1)*length(xval2));% se transforma 'yval', car
thetaval= phival\yval';% se construieste matricea coeficientilor
```

Rezolvarea erorilor de identificare si validare:

```
eror= yid-yprimid(:)';
sumerori=0;
for i=1:length(xid1)*length(xid2)
    sumerori= sumerori+eror(i).^2;
end
MSEid(l)=(1/(length(xid1)*length(xid2)))*sumerori


eror1= yval-yprimval(:)';
sumerori1=0;
for i=1:length(xval1)*length(xval2)
    sumerori1= sumerori1+eror1(i).^2;
end
MSEval(l)=(1/(length(xval1)*length(xval2)))*sumerori1
end

figure;
plot(MSEid);
grid on;
hold on;
plot(MSEval);
```