

2 Indicatori și reprezentări statistice

Realizarea unei histograme în MATLAB folosind generatorul de numere aleatoare

În cazul în care nu aveți la dispoziție date experimentale pentru realizarea unei histograme în MATLAB, puteți folosi funcția *random* pentru a genera numere aleatoare cu o distribuție de probabilitate dată. Acest mod de lucru este util în special pentru a realiza simulări. În acest caz vom simula un set de date pentru a realiza histogramelor.

Sintaxa pentru apelarea funcției *random* este:

- *random*(NUME,A) returnează un vector de numere aleatoare, alese din distribuția de probabilitate cu un parametru, specificată de NUME, cu valoarea parametrului A.
- *random*(NUME,A,B) sau *random*(NUME,A,B,C) returnează un vector de numere aleatoare, alese dintr-o distribuție de probabilitate cu valorile parametrilor A, B (și C).

Există multe tipuri de distribuții ce pot fi alese la utilizarea acestei funcții. Introduceți *help random* în linia de comandă pentru a afișa o listă completă cu numele și parametrii de intrare.

Distribuție	Parametrul de intrare A	Parametrul de intrare B
'bino' sau 'Binomial' pentru fiecare încercare	n: numărul de încercări	p: probabilitatea de succes
'exp' sau 'Exponential'	μ : media	-
'norm' sau 'Normal'	μ : media	σ : deviația standard
'unif' sau 'Uniform'	a: punct inferior (minim)	b: punct superior (maxim)

Pentru a crea un vector sau o matrice cu numere aleatoare folosiți funcțiile de mai sus urmate de dimensiunile matricei. De exemplu:

- *random*('norm',mu,sigma,1,N) va returna o selecție aleatoare de N valori dintr-o distribuție normală și va plasa valorile într-un vector unidimensional de lungime N.
- *random*('bino',n,p,M,N) va returna o selecție aleatoare de valori dintr-o distribuție binomială și va plasa valorile într-o matrice MxN.

Odată creat setul de date cu ajutorul generatorului de numere aleatoare, puteți reprezenta grafic datele, folosind funcția `histogramă`.

Exemplu

Realizați o rutină MATLAB pentru a genera o selecție aleatoare de 1000 de puncte de date dintr-o distribuție Gaussiană cu $\mu = 1$ și $\sigma = 0.5$.

```
media=1; dispersia=0.5; N=1000;
```

```
date=random('Norm',media,dispersia,1,N);
```

```
figure(1); hist(date); % valoarea standard este de 10 clase title('Histograma cu 10 clase');
```

Rezultatul obținut este prezentat în Figura 6.

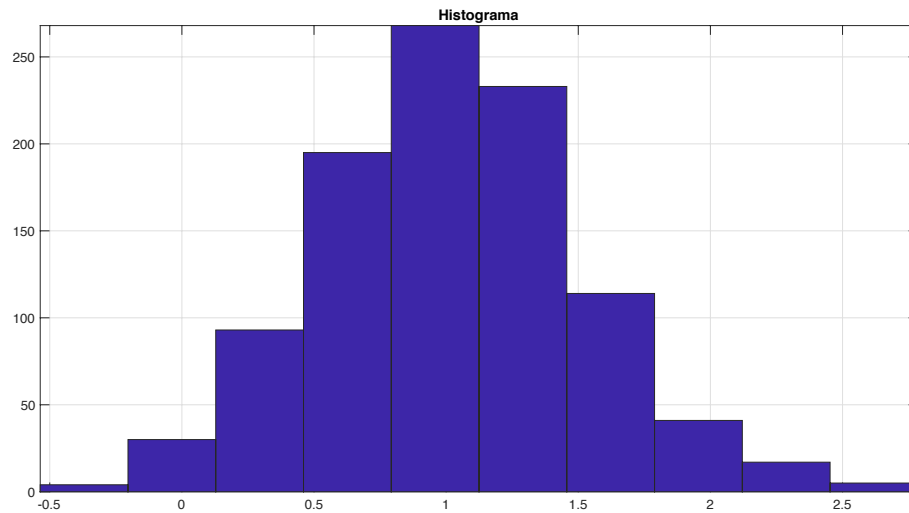


Figure 6: Reprezentare grafică de tip histogramă

Prin intermediul comenzii `histfit` poate fi verificată aderența distribuției empirice de probabilitate reprezentate de histograma setului de date la o repartiție continuă cunoscută. Exemplul din Figura 7 prezintă cazul unei distribuții normale.

Exercițiu

1. Modificați μ , σ , N și numărul de clase. Cum afectează aceste modificări forma histogramei?

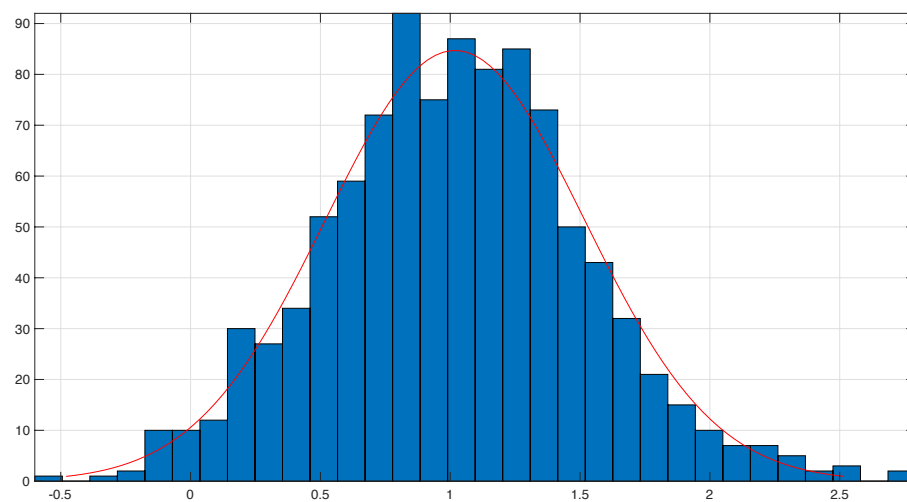


Figure 7: Utilizare *histfit* pentru verificarea aderenței la o distribuție de probabilitate cunoscută - Cazul distribuției normale

3 Referințe

- [1] B. Hahn, D. Valentine, Essential MATLAB for Engineers and Scientists, Third Edition, Elsevier, 2007.

Laborator 4

December 2, 2020

1 Metode de regresie pentru modelarea seturilor de date

Introducere

Curve Fitting Tool este utilitarul MATLAB care oferă interfețe grafice și funcții pentru linia de comandă pentru atribuirea de funcții și suprafețe seriilor de date. Utilitarul permite realizarea de analize de date exploratorii, pre-procesarea și post-procesarea datelor, compararea modelelor candidate și înlăturarea excepțiilor. Pot fi implementate regresii folosind librăriile existente de modele liniare și neliniare sau pot fi specificate propriile ecuații. Librăria oferă parametri de rezolvare optimizați și condiții inițiale pentru îmbunătățirea calității fiturilor. Pachetul suportă și tehnici de modelare neparametrice, cum sunt spline, interpolarea și netezirea.

Caracteristici cheie:

- Instrumente grafice pentru fitarea curbilor și suprafețelor
- Regresii liniare și neliniare cu ecuații personalizate
- Bibliotecă de modele de regresii cu puncte inițiale optimizate și parametri de rezolvare
- Metode de interpolare, inclusiv B-splines, splines tensor-produs, etc.
- Tehnici de netezire, inclusiv splines de netezire, regresii localizate, filtre Savitsky-Golay și medii alunecătoare
- Rutine de pre-procesare, inclusiv înlăturarea extremelor și secționare, scalare și ponderarea datelor
- Rutine de post-procesare, inclusiv interpolare, extrapolare, intervale de încredere, integrale și derivate

Utilizare

Pentru deschiderea Curve Fitting Tool, introduceți următoarea comandă în linia de comenzi MATLAB:

```
>> cftool
```

iar utilitarul se va deschide sub forma unei ferestre noi (vezi Figura 1).

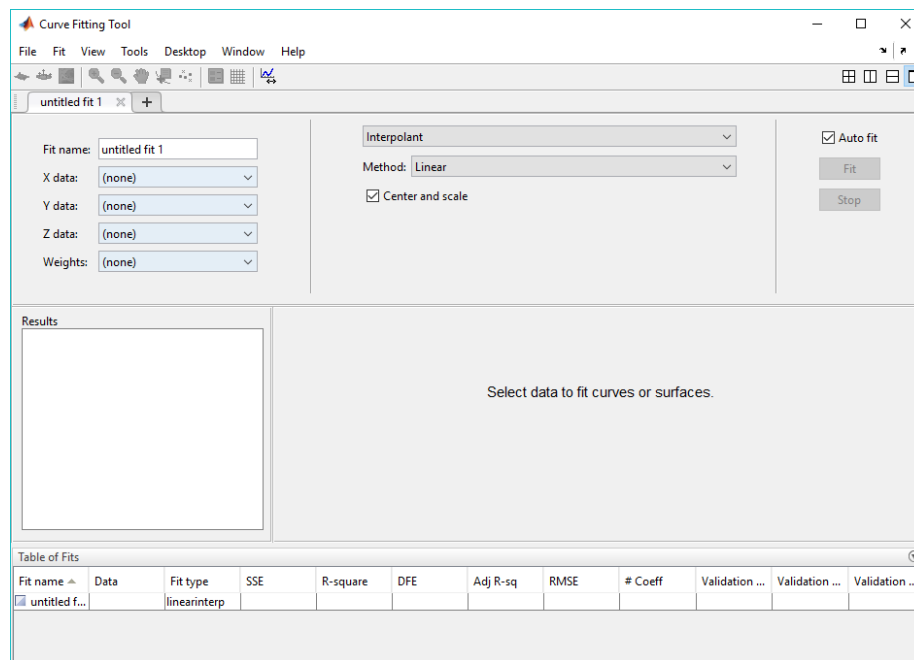


Figure 1: Fereastra principală - Curve Fitting Tool

Fereastra principală cuprinde următoarele secțiuni:

- 1...: O zonă care permite utilizatorului importarea de date din spațiul de lucru MATLAB în mediul Curve Fitting Tool. Acesta va fi denumit set de date.
- 2...: O zonă ce permite alegerea funcției de fitare și a parametrilor ei.
- 3... O zonă denumită "Table of Fits" de unde sunt selectate opțiunile de afișare pentru seturile de date, Pot fi alese selectiv cele pe care dorim să le reprezentăm grafic. Graficele pot fi vizualizate în zona denumită "Results".

Exemplu 1.

Se definesc în "Command Window" vectorii:

```
>> x = [0 : 0.1 : 2 * pi];
```

» $y = \sin(x)$;

Se crează un set de date alegând x , y , numele setului de date și opțional variabilele pondere. Setul de date va fi afișat în listă din zona "Table of Fits" și va putea fi utilizat în pașii următori.

Acest scurt exemplu nu include setare unor valori pentru ponderi deoarece variabilele x și y au fost create. În cazul analizei unor date, cum este măsurarea la un interval de timp finit e.g. la fiecare secundă, parametrul x este unul ales (parametrul liber). Parametrul y este unul măsurat (dependent) care prezintă erori diferite între punctele de date. Fitarea normală, fără ponderi, consideră toate punctele egale dar în cazul în care unele puncte au bare de erori mai mari, relativ la punctele celelalte, asta înseamnă că sunt mai puțin precise și că fitarea ar trebui să le ia în considerare mai puțin în calcule. Acesta este rolul ponderilor. Punctele de date mai precise primesc ponderi mai mari în raport cu punctele de date mai puțin precise.

Modul de a crea ponderile, corelat cu logica expusă mai sus, este ca ponderea unui punct de date să fie inversul pătratului barei de eroare. Exprimată matematic: dacă w_i este ponderea și σ_i este eroarea punctului de date i , atunci:

$$w_i = \frac{1}{\sigma_i^2} \quad (1)$$

Pentru a fi datale, trebuie ales setul de date predefinit, funcția de fitare (ex. "Polynomial", "Exponential", etc.) și apăsat butonul "Fit" (în cazul în care nu este selectată deja opțiunea "Auto fit").

Există o serie de funcții de fitare incluse, care sunt grupate după tip: "Sumă de sinusoid", "Polinomiale", etc. Trebuie aleasă funcția specifică care se dorește a fi folosită pentru fitare. În acest caz vom alege o funcție din grupul "Sum of Sine".

La apăsarea butonului "Fit", este calculat fitul și rezultatele sunt afișate în zona dedicată din fereastra principală "Curve Fitting Tool" împreună cu reprezentarea grafică. Figura 2 prezintă rezultatul operațiilor efectuate. Inspectia vizuală a rezultatului arată că fitarea este bună și coeficientul de determinare R-square și R-square ajustat confirmă acest lucru. Putem vedea de asemenea că amplitudinea, $a1$, este 1, frecvența (sau mai exact, $\omega = 2\pi f$, care este $b1$ dacă variabila x este timpul) este, de asemenea, 1 și faza, $c1$, este 0, identic cu parametrii aleși la crearea vectorului y ($y = \sin(x)$).

Numerele din paranteză, de lângă valorile parametrilor (coeficienții) sunt intervalele de încredere - eroarea fitului. Deoarece fitul nostru se potrivește exact setului de date, aceste valori sunt zero. Această observație nu va mai fi valabilă

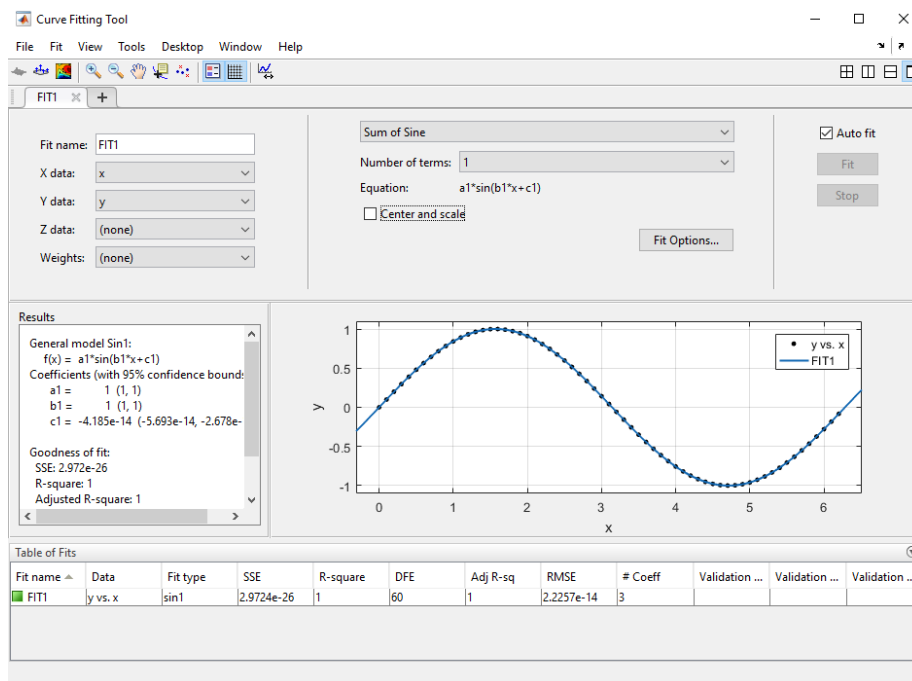


Figure 2: Fitarea datelor cu o funcție $y = a1 \cdot \sin(b1 \cdot x + c1)$

în exemplele ce vor urma.

La modul general, algoritmul de fitare pornește de la niște valori ale constantelor funcției de fitare, $a1$, $b1$ și $c1$ în acest caz, și încearcă să convergă de la acești parametri inițiali, la cei finali. Uneori parametrii inițiali nu sunt destul de apropiați de cei finali pentru a converge la aceștia sau nu converg deloc. Meniul "Fit options..." permite modificarea acestor parametri.

În cazul în care o funcție de fitare de care avem nevoie nu există în lista de funcții, putem alege ecuații personalizate prin butonul "Custom equation" și definirea funcției dorite.

Din meniul "Fit" → "Save to workspace..." se vor salva datele de fitare în spațiul de lucru și de acolo vom putea lucra cu aceste date în linia de comandă.

Din meniul "View" → "Residuals Plot" se afișează încă un grafic care va prezenta reziduurile, diferența finală dintre punctele de date inițiale și fit. Figura 3 ilustrează rezultatele pentru exemplul dat.

La analiza reziduurilor este important să vedem că acestea sunt împrăștiate

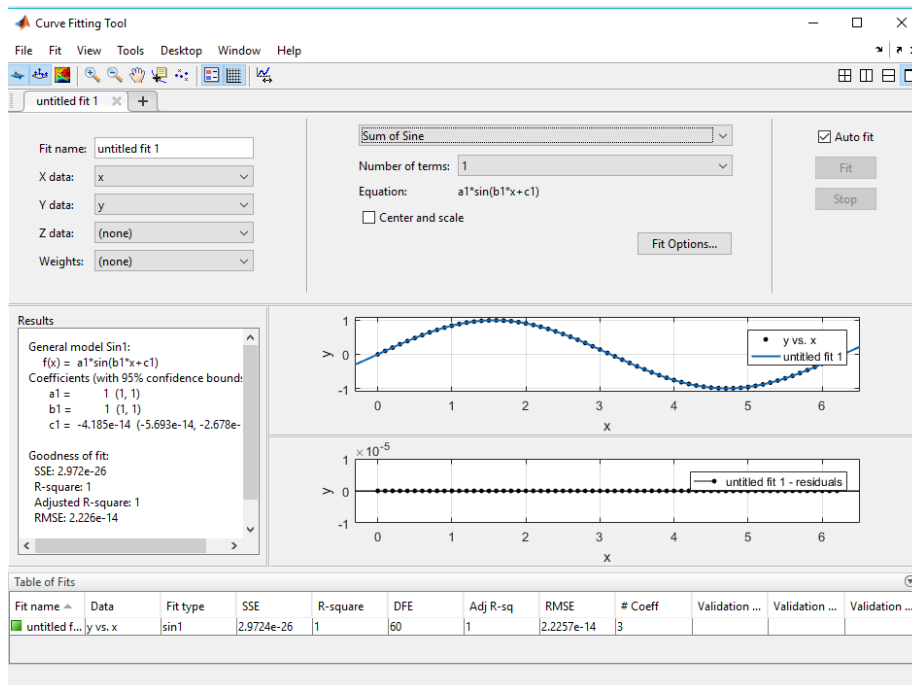


Figure 3: Graficul cu afişarea reziduurilor

aleatoriu și uniform în jurul valorii de zero. Asta presupune ca să nu fie pozitive într-o parte a fitului și negative în altă parte, sau ca valoarea lor nu este mică sau mare în zone diferite.

Exemplu 2. Fit liniar simplu, cu bare de eroare.

În acest exemplu vor fi fitate 5 puncte de date cu o funcție liniară. Introduceți următoarele comenzi:

```
>>m=[1 2 3 4 5];
```

```
>>n=[1 2 3 4 4.5];
```

și barele de erori și ponderile:

```
>> err=[0.1 0.2 0.3 0.4 0.5];
```

```
>> weight=1./err.^2;
```

Se poate observa cum este generată o linie dreaptă cu panta 1 în care ultimul punct pare a fi greșit (Figura 4). O opțiune este excluderea acestui punct din grafic, dar nu este cea preferată. Vom arăta cum adăugarea ponderilor are un efect similar (Figura 5).

În continuare, salvați fitul în "Workspace" păstrând denumirile așa cum este

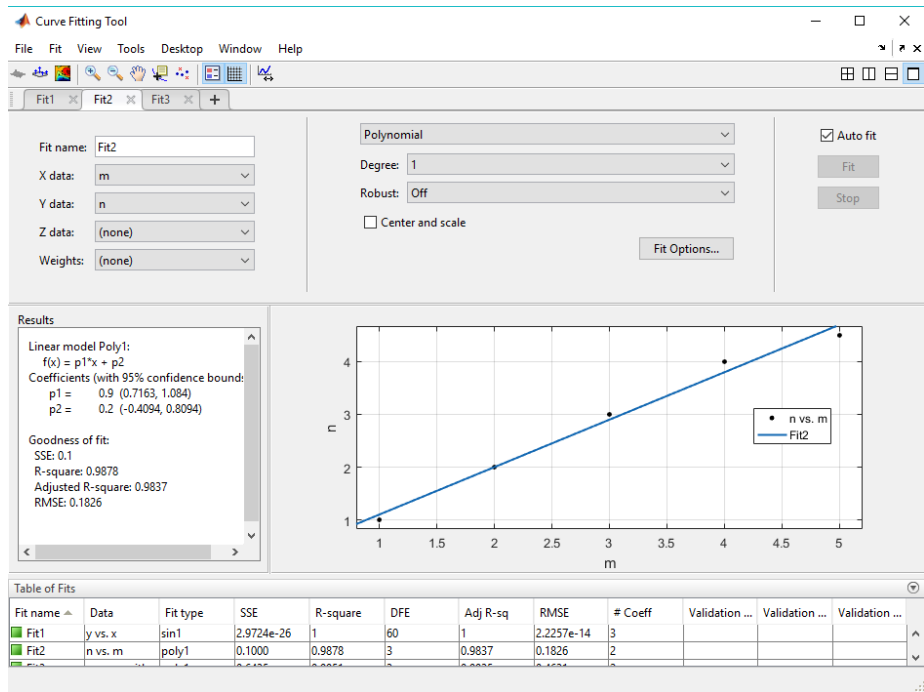


Figure 4: Fitare cu funcție liniară fără ponderi

ilustrat în Figura 6. Faceți dublu clic pe aceste variabile în spațiul de lucru pentru a afișa conținutul lor.

În continuare va fi realizat un grafic care include și barele de eroare alături de funcția de fitare și limitele de încredere.

Introduceți următoarele comenzi în linia de comandă MATLAB:

```
>> figure
>> plot(m,n,'o')
>> xlim([0,10])
>> hold on
>> plot(fittedmodel,'predobs')
>> h=errorbar(m,n,err,'.')
>> hold off
>> legend('m vs. n','poly1-fit','95 % prediction bounds')
```

Figura 7 prezintă rezultatele obținute.

Exemplu 3. Fit avansat

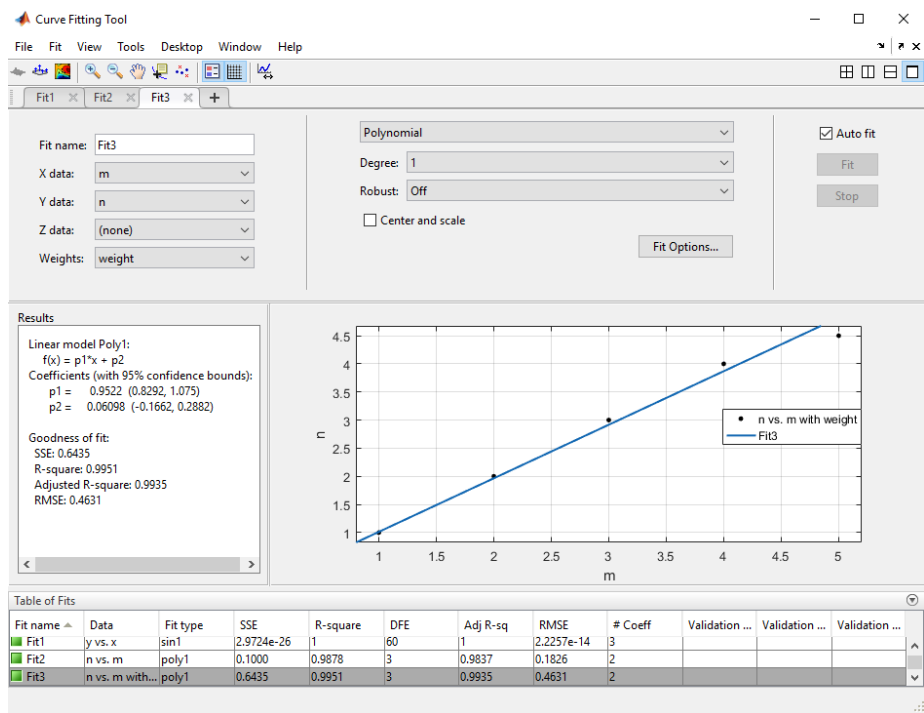


Figure 5: Fitare cu funcție liniară cu ponderi

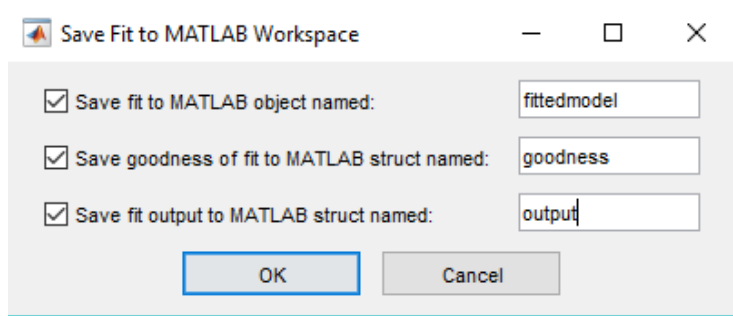


Figure 6: Save Fit to MATLAB Workspace

Acest exemplu prezintă fitarea unui set de date cu o funcție de tipul \sin^2 . Funcția de fitare nu este inclusă în opțiunile standard și va trebui creată. De asemenea, vor trebui definiți alți parametri inițiali. În acest exemplu este folosit un set de date afectate de zgomot. Acesta este introdus în linia de comandă MATLAB, după cum urmează:

```
>>a=[0:0.1:2*pi];
```

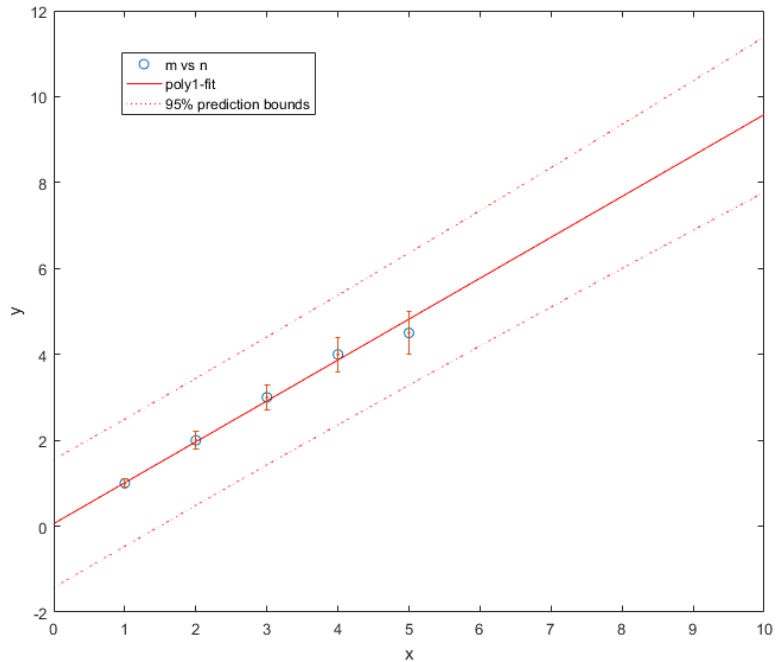


Figure 7: Grafic cu setul de date, bare de eroare, funcția de fitare și limite de încredere

```
>>b=8*sin(2*pi*0.2*a+1).^2+randn(size(a));
```

Se poate observa că datele sunt un \sin^2 cu amplitudinea 8, frecvența 0.2 și faza 1 iar scopul este obținerea acestor valori din funcția de fitare.

Pentru a ajunge la rezultatele ilustrate, trebuie creată în "Curve Fitting Tool" fitarea folosind "New Fit" și o ecuație introdusă de la tastatură "Custom Equation". Aici trebuie inserați și parametri inițiali de fitare. Prin apăsarea "Fit Options..." se va deschide o nouă fereastră unde pot fi modificați parametrii de fitare. După identificarea corectă a acestora, afișați și reziduurile pentru a verifica distribuția lor aleatoare. Urmăriți Figura 8 și Figura 9 pentru a vedea rezultatele obținute.

Exercițiu: Predicția unui set de date

Fie setul de date primar determinist definit după cum urmează, pentru x aleator:

```
>>x=15*randn(1,100)+25;
```

```
>>y = 0,0001 * (2x3 + 14x2) + 20;
```

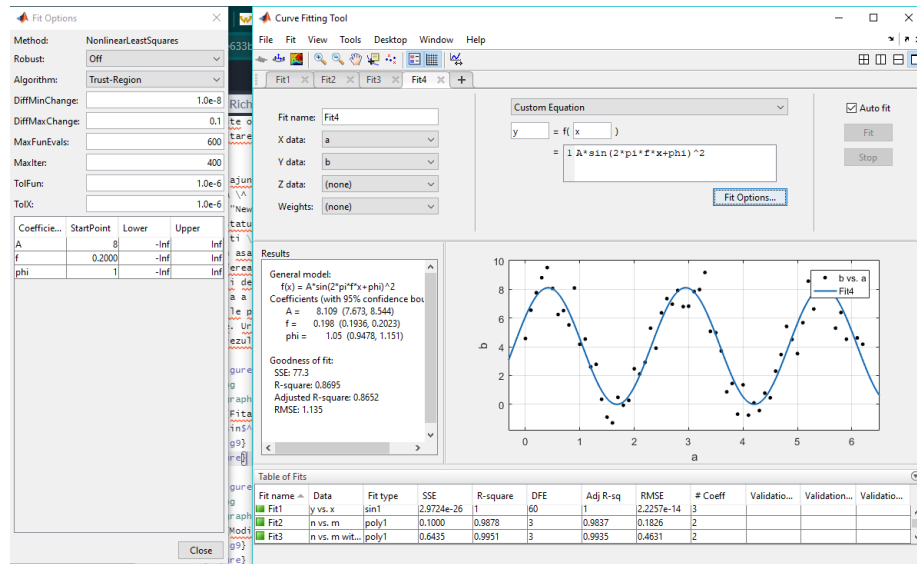


Figure 8: Fitare cu funcție tip \sin^2

Setul de date secundar y , este format pe baza setului de date primar, la care se adaugă o componentă aleatoare. Să se utilizeze **cftool** pentru identificarea celui mai potrivit model pentru cele două seturi de date. Justificare.

2 Referințe

- [1] B. Hahn, D. Valentine, Essential MATLAB for Engineers and Scientists, Third Edition, Elsevier, 2007.
- [2] MathWorks, MATLAB Curve Fitting Toolbox User's Guide, R2011a.

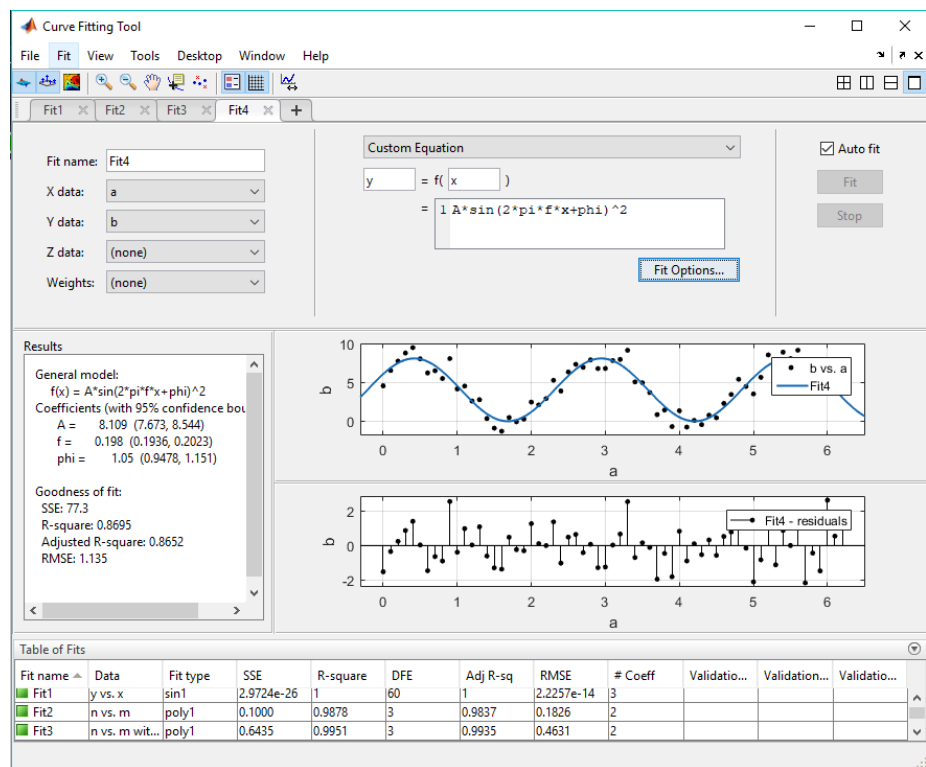


Figure 9: Rezultate finale cu erori reziduale