Warszawa, 10.12.2018r.

Politechnika Warszawska

Wydział Elektroniki i Technik Informacyjnych

Metody numeryczne

Zadanie 2: Aproksymacja funkcji

Wykonawca zadania:

Joanna Kiesiak #22

Prowadzący projekt:

dr inż. Andrzej Miękina

Spis treści

1.Wstęp

1.1 Środowisko programu

Program został wykonany w programie MATLAB R2017b. MATLAB to pakiet programowy, którego zadaniem jest wykonywanie złożonych obliczeń matematycznych i wizualizacja wyników. Podstawowym elementem konstrukcyjnym w tym programie jest macierz, a tablica fundamentalnym typem danych.

1.2 Aproksymacja średniokwadratowa funkcji

Poszukiwany jest wektor parametrów $p = [p_1 \dots p_K]^T$, dla którego liniowa kombinacja liniowo niezależnych funkcji $\{\varphi_k(x) \mid k = 1, \dots, K\}$:

$$\hat{f}(x; p) = \sum_{k=1}^{K} p_k \varphi_k(x)$$

Najlepiej przybliża ciąg dyskretnych wartości nieznanej funkcji $f(x)$: $\{f(x_n) \mid n = 1, \dots, N\}$ w sensie następującego kryterium:

$$J_2(p) = \sum_{n=1}^{N} [\hat{f}(x_n; p) - f(x_n)]^2$$

Warunek konieczny osiągania minimum:

$$\frac{\partial J_2(p)}{\partial p_k} = 0 \;\; dla \;\; k = 1, \dots, K \;\leftrightarrow\; \boldsymbol{\Phi^T} \cdot \boldsymbol{\Phi} \cdot \boldsymbol{p} = \boldsymbol{\Phi^T} \cdot \boldsymbol{y}$$

Gdzie:

$$\Phi = \begin{bmatrix} \varphi_1(x_1) & \varphi_2(x_1) & \cdots & \varphi_K(x_1) \\ \varphi_1(x_2) & \varphi_2(x_2) & \cdots & \varphi_K(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_1(x_N) & \varphi_2(x_N) & \cdots & \varphi_K(x_N) \end{bmatrix} \text{ oraz } y = [f(x_1) \quad f(x_2) \quad \cdots \quad f(x_N)]^T$$

Możliwe jest użycie metody Cholesky'ego-Banachiewicza, bo $\boldsymbol{\Phi^T} \cdot \boldsymbol{\Phi}$ jest macierzą dodatnio określoną.

1.3 Wskaźnik błędu średniokwadratowego i maksymalnego

Wskaźnik błędu średniokwadratowego:

$\delta_2(K, N) = \frac{\|\hat{f}(x;K,N) - f(x)\|_2}{\|f(x)\|_2}$, gdzie $\hat{f}(x; K, N)$ jest funkcją aproksymującą.

Wskaźnik błędu maksymalnego:

$\delta_\infty(K, N) = \frac{\|\hat{f}(x;K,N) - f(x)\|_\infty}{\|f(x)\|_\infty}$, gdzie $\hat{f}(x; K, N)$ jest funkcją aproksymującą.

2.Zadanie 1

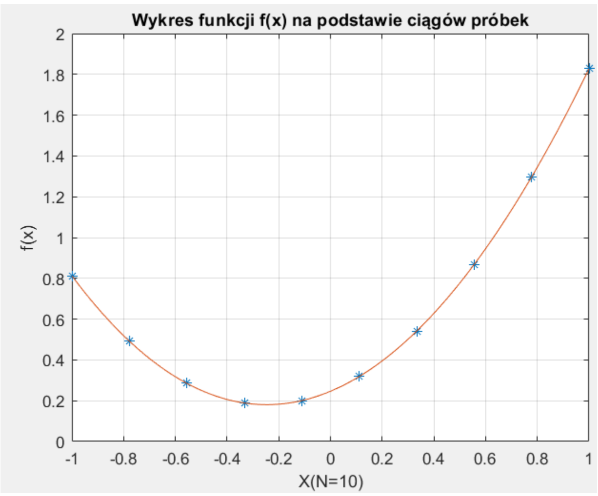
2.1 Treść zadania

Sporządzić wykresy funkcji $f(x) = \left(x + \frac{1}{3}\right)^{2} + e^{-x-2}$ *w przedziale [-1;1] i zaznaczyć ciąg próbek jej wartości, na podstawie których dokonana zostanie jej aproksymacja:* $\{y = f(x_n) | n = 1,2,\ldots,N\}$*, gdzie* $x_n = -1 + 2\frac{n-1}{N-1}$ *dla* $N = 10,20,30.$

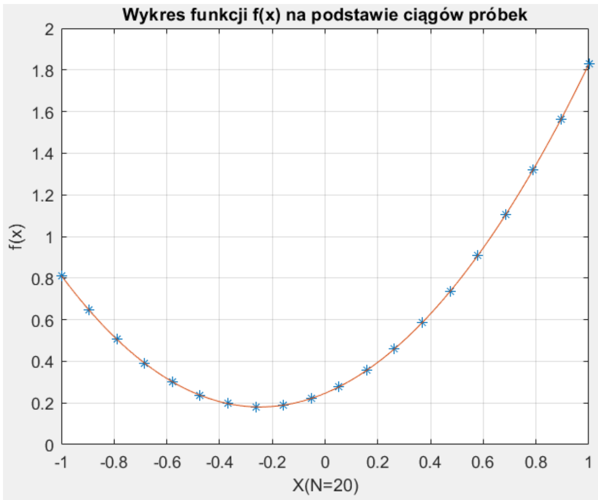
2.2 Opis zastosowanych algorytmów numerycznych

Funkcja **wyliczWezly** zwraca argumenty wyjściowe x_n i f(x_n) w zależności od argumentu wejściowego N. Na początku algorytmu zostają zadeklarowane dwie funkcje anonimowe, które obliczają x_n oraz f(x_n). Zadeklarowana zostaje zmienna x1000, za pomocą wygenerowania 1000 punktów w przedziale [-1;1]. W pętli for zostaje wyliczony xn oraz yn w zależności od argumentu wejściowego N. Zostają wygenerowane trzy wykresy, za pomocą odwołania zmiennej x do funkcji f. Wykresy zamieszczone poniżej, są wygenerowane dzięki wbudowanej funkcji plot oraz opisane poleceniami xlabel, ylabel oraz title. Przedstawiają funkcję yx oraz ciąg próbek(xn), które będą aproksymowane w kolejnym etapie, tak aby uzyskać najlepsze przybliżenie do funkcji f.

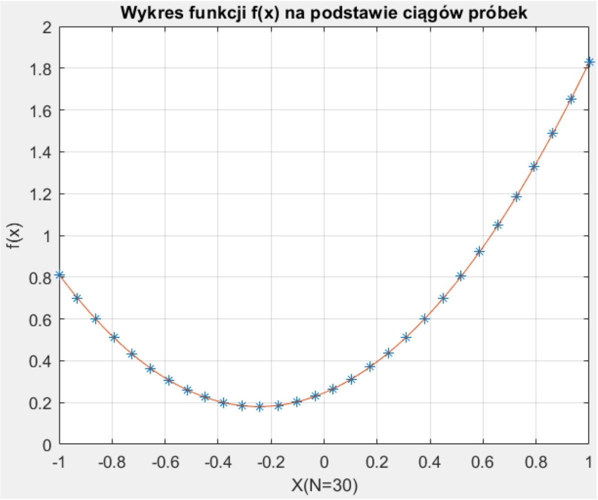
2.3 Przykład działania programu



Rys.1 Wykres funkcji f(x) dla N=10.



Rys.2 Wykres funkcji f(x) dla N=20.



Rys.3 Wykres funkcji f(x) dla N=30.

3. Zadanie 2

Opracować program aproksymacji funkcji $f(x)$ *na podstawie danych* $\{(x_n, y_n)|n = 1, \dots, N\}$ *metodą najmniejszych kwadratów za pomocą operatora wyznaczania pseudo-odwrotności macierzy „\"; jako bazę funkcji liniowo niezależnych przyjąć:* *dla* $k = 1,2, \dots, K$, *gdzie* $x_k = -1 + 2\frac{k-1}{K-1}$*:*

$$W_k(x) = \begin{Bmatrix} 2|x - x_k'|^{\,2} - 3|x - x_k'|^{\,2} + 1 & ; dla\ |x - x_k'| < 1 \\ 0 & ; dla\ pozostałych \end{Bmatrix}$$

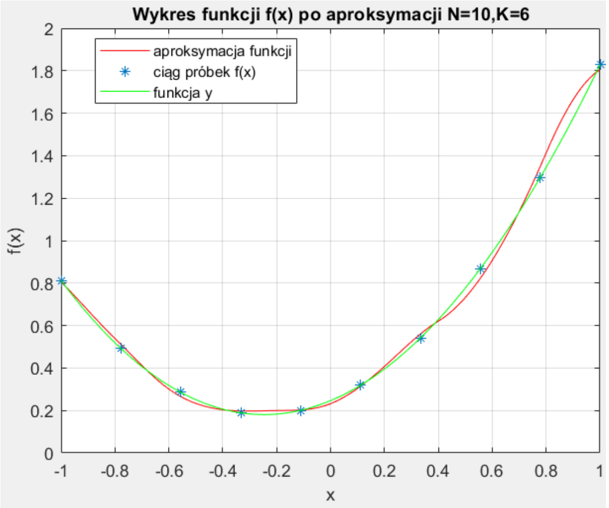
Sprawdzić poprawność działania programu dla kilku par niewielkich wartości parametrów N i K*. Do wykresów z punktu 1 dorysować wynik aproksymacji dla wybranych wartości K.*

3.2 Opis zastosowanych algorytmów numerycznych

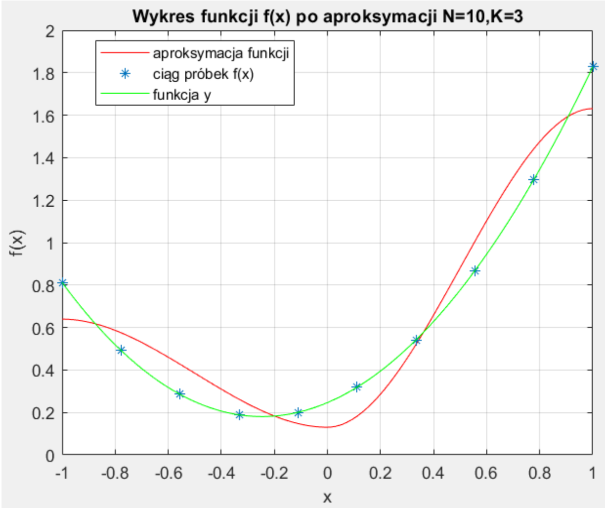
W celu opracowania algorytmu do wykonania zadania skorzystano z funkcji z poprzedniego zadania **wyliczWezly** oraz utworzono dwie funkcje **macierzP** oraz **wyznaczFunkAp**. W funkcji macierzP pobieramy argumenty [xn, yn] z funkcji wyliczWezly oraz argument wejściowy K. W algorytmie korzystamy z zaimplementowanej funkcji anonimowej iks, macierzy W wypełnionej zerami oraz zmienna xK wyznaczona za pomocą funkcji anonimowej wyznaczającej x_k. W zagnieżdżonej pętli for zostaje wyznaczona funkcja $W_k(x)$ według wzoru z polecenia. Następnie za pomocą operatora pseudo-odwrotności macierzy „/" wyznaczona zostaje macierz P, która jest argumentem wyjściowym o wymiarze [K x 1].

Funkcja wyznaczFunkAp pobiera jako argument wejściowy cztery zmienne. Wartości xK, macierz p, wartość K oraz węzły wyznaczone w funkcji wyliczWezly**.** Na początku za pomocą podwójnej pętli zostaje wyznaczona macierz liniowo niezależnych funkcji $\varphi_k(x)$ dla dużej ilości wierszy, tak aby można było narysować funkcje po aproksymacji gładką na wykresie. Następnie odbywa się wyliczenie funkcji wyjściowej po aproksymacji, poprzez iloczyn macierzy φ_k ×p'. Generacja wykresów odbywa się analogicznie jak w poprzednim zadaniu.

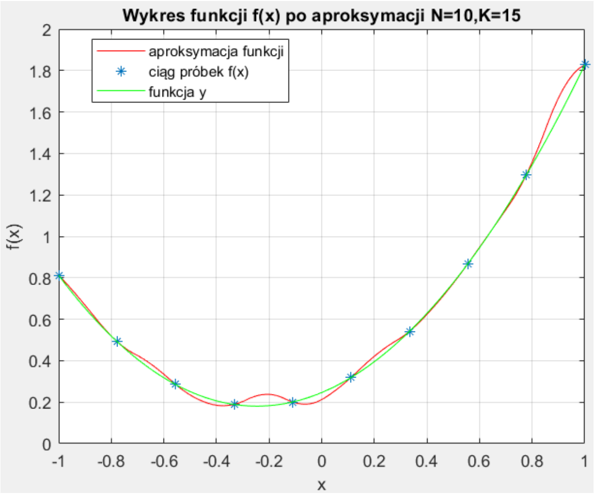
3.3 Przykład działania programu



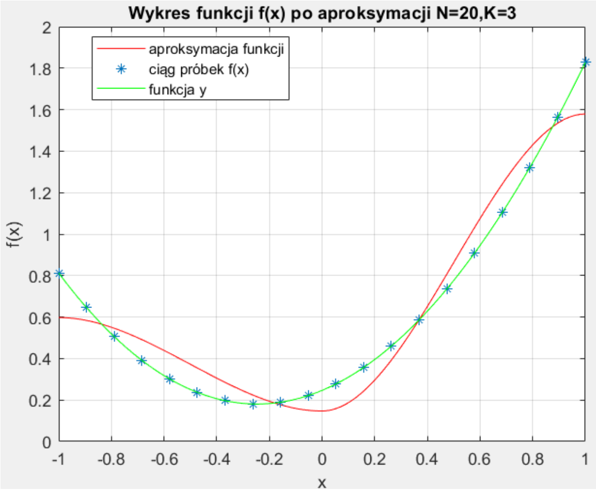
Rys.4 Wykres funkcji f(x) i jej aproksymacji dla N=10, K=6



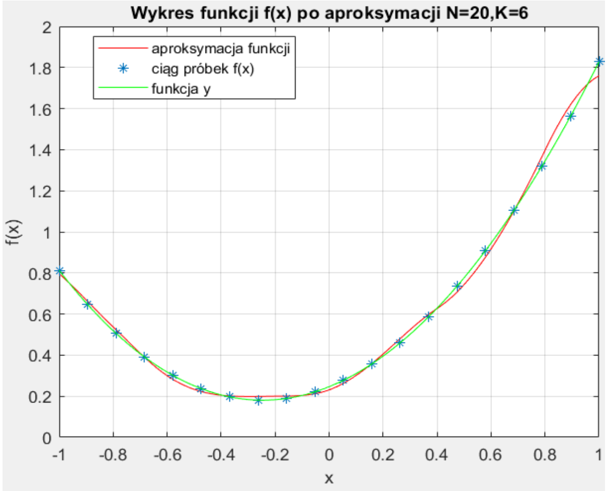
Rys.5 Wykres funkcji f(x) i jej aproksymacji dla N=10, K=3



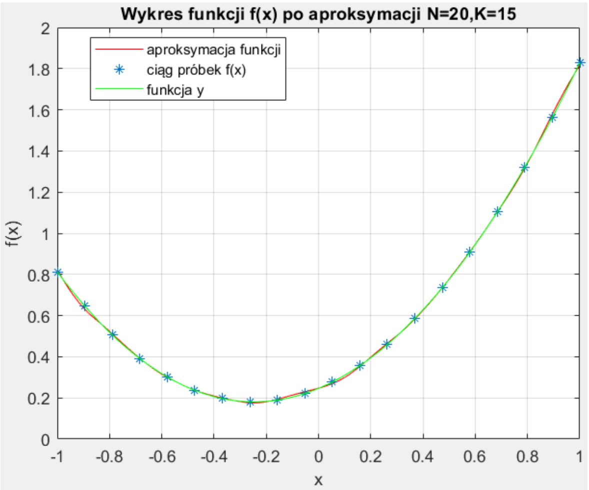
Rys.6 Wykres funkcji f(x) i jej aproksymacji dla N=10, K=15



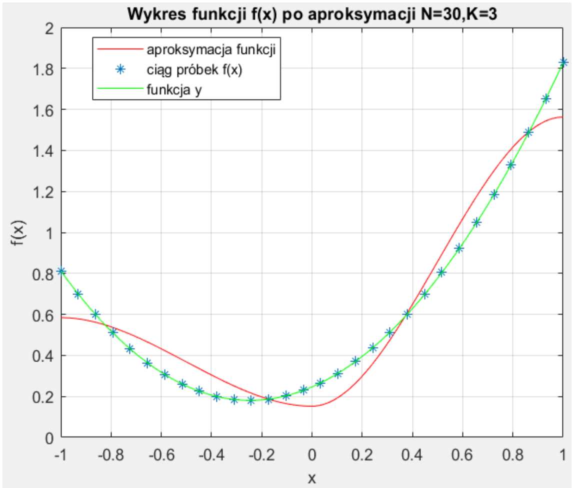
Rys.7 Wykres funkcji f(x) i jej aproksymacji dla N=20, K=3



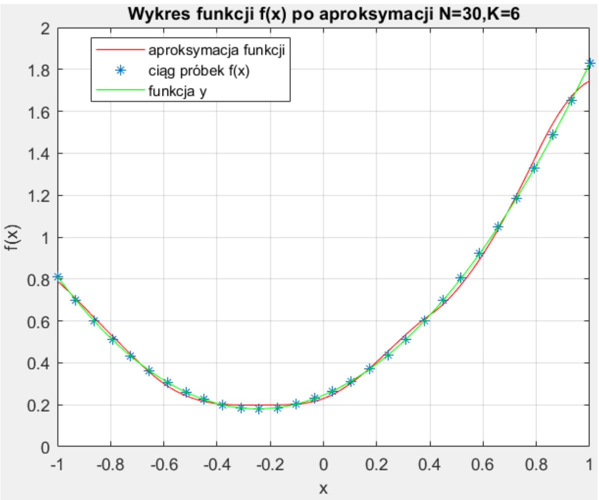
Rys.8 Wykres funkcji f(x) i jej aproksymacji dla N=20, K=6



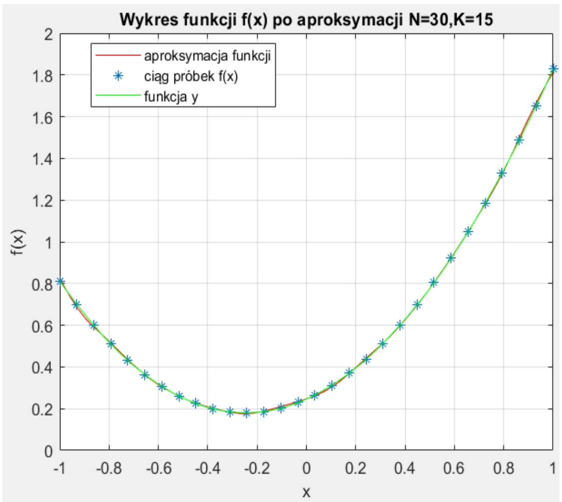
Rys.9 Wykres funkcji f(x) i jej aproksymacji dla N=20, K=15



Rys.10 Wykres funkcji f(x) i jej aproksymacji dla N=30, K=3



Rys.11 Wykres funkcji f(x) i jej aproksymacji dla N=30, K=6



Rys.12 Wykres funkcji f(x) i jej aproksymacji dla N=30, K=15

3.4 Wnioski

Wniosek jaki można wysnuć z powyższych wykresów to im większy stopień wielomianu $W_k(x)$ tym zjawisko aproksymacji jest dokładniejsze i szybciej zbliża się kształtem do funkcji odniesienia. Im większy parametr N tym zjawisko aproksymacji jest dokładniejsze i zachodzi szybciej. Dla K=15 obserwujemy na wykresach coraz bardziej dopasowany wykres do funkcji $\mathrm{f}(x)$. Wniosek: wzrost parametru K jest proporcjonalny do aproksymacji funkcji, pod warunkiem że K<N, wtedy zachodzi tutaj zjawisko źle uwarunkowanego wielomianu.

4.Zadanie 3

4.1 Treść zadania

Przeprowadzić systematyczne badania zależności dokładności aproksymacji od parametrów N i K ; w tym celu wyznaczyć wskaźniki błędu średniokwadratowego oraz błędu maksymalnego:

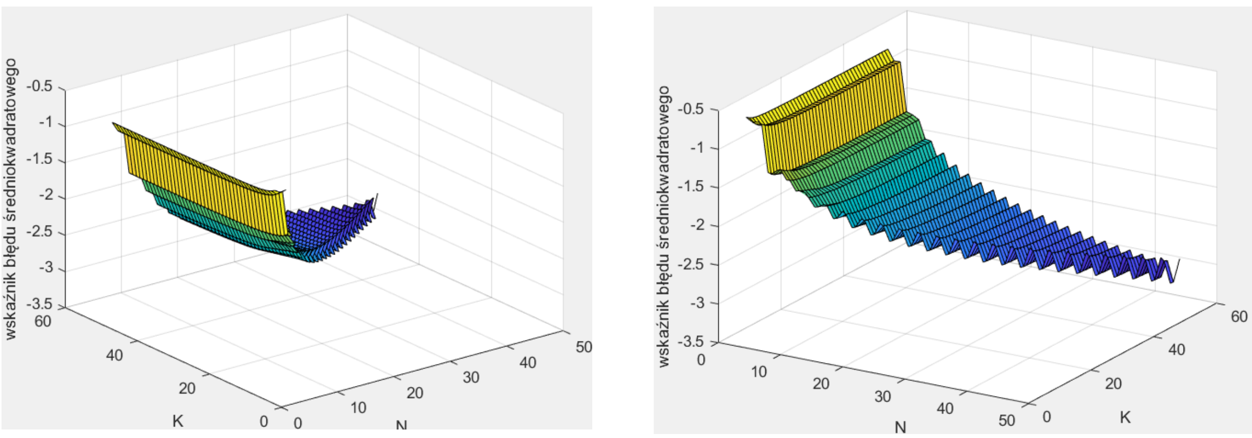
$$\delta_2(K,N) = \frac{\left\|\hat{f}(x;K,N) - f(x)\right\|_2}{\|f(x)\|_2} \qquad i \qquad \delta_\infty(K,N) = \frac{\left\|\hat{f}(x;K,N) - f(x)\right\|_\infty}{\|f(x)\|_\infty}$$

Gdzie $\hat{f}(x;K,N)$ jest funkcją aproksymującą, uzyskaną dla wskazanych wartości N i K. Sporządzić wykresy zależności $\delta_2(K,N)$ i $\delta_\infty(K,N)$ dla N w zakresie 5-50 oraz wartości K<N.

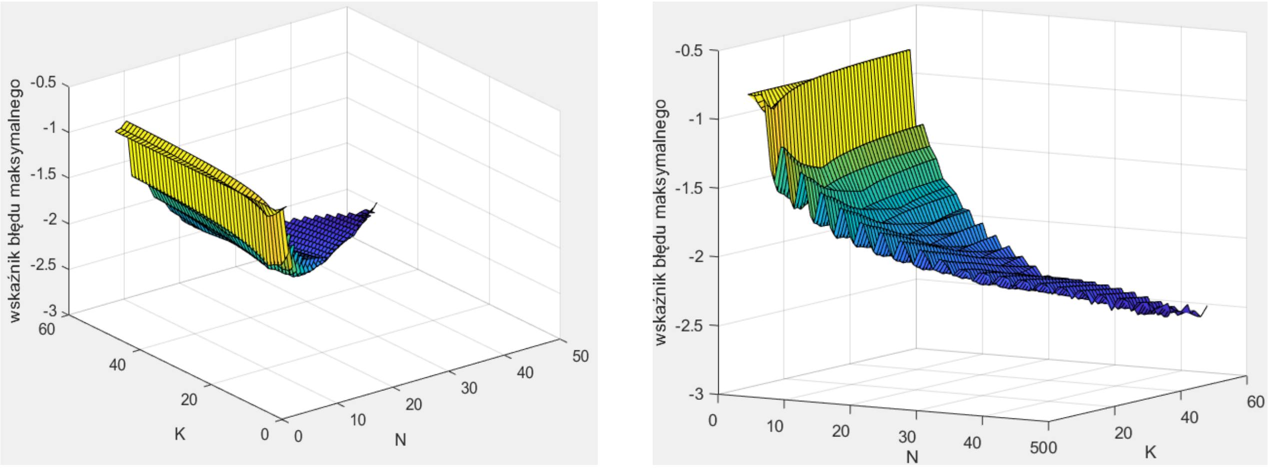
4.2 Opis zastosowanych algorytmów numerycznych

Zadanie trzecie jest realizowane przez funkcje **wyliczWezly**, **macierzP**, **wyznaczFunkAp** oraz nową funkcję **wyznaczBledy**. Funkcja wyznaczBledy pobiera na wejście macierz po aproksymacji i zwraca błąd średniokwadratowy oraz maksymalny za pomocą wbudowanej w Matlaba funkcji norm. Błąd średniokwadratowy czyli kwadrat różnicy pomiędzy estymatorem a wartością estymowaną. Do narysowania wykresów przestrzennych wykorzystano polecenia surf w zależności od parametrów N=5:50 i K<N. Wykresy zostały utworzone w skali logarytmicznej.

4.3 Przykład działania programu



rys.13 Wskaźnik błędu średniokwadratowego dla N=5:50 i K<N.



rys.14 Wskaźnik błędu maksymalnego dla N=5:50 i K<N.

4.4 Wnioski

Uzyskanie dobrej aproksymacji średniokwadratowej wygładzającej wiąże się z obliczeniem błędów, które oczekiwane są jako wartości minimalne, co oznaczałoby że algorytm spełnia swoje funkcje. Na wykresie obserwujemy że błąd średniokwadratowy oraz błąd maksymalny przyjmuje wartości ujemne dla wartości błędu – wynika to z zastosowania skali logarytmicznej. Na przedstawionych wykresach obserwujemy że błąd zarówno średniokwadratowy jak i maksymalny dla wzrastających zmiennych N i K zaczyna stabilizować swoją wartość, co w konsekwencji prowadzi do wniosku że wzrost zmiennych prowadzi do zmniejszenia błędu aproksymacji. Dla porównania zostały wygenerowane wykresy dla N=5:500 i K<N. Algorytm charakteryzuje się stabilnością i poprawnością ze względu na brak pików na wykresach błędów.

5.Zadanie 4

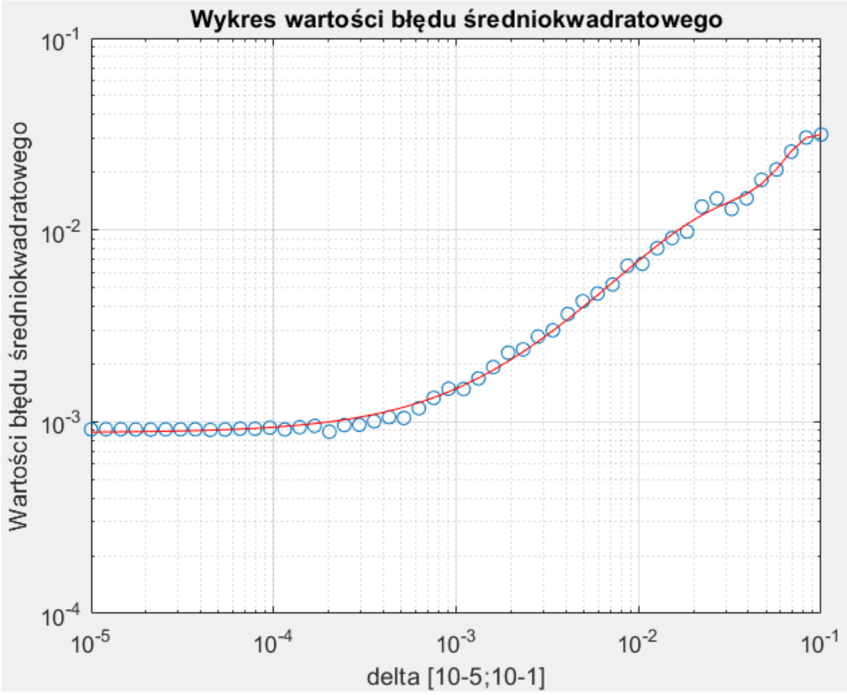
5.1 Treść zadania

Wyznaczyć w skali logarytmicznej zależność $\delta_{2,MIN}(\delta_y)$ *dla* $(\delta_y = [10^{-5}; 10^{-1}])$ *gdzie* $\delta_{2,MIN}$ *to minimalna wartość błędu* $\delta_{2,MIN}(K,N)$ *uzyskana dla danego poziomu zaburzenia danych. W tym celu dokonać aproksymacji za pomocą funkcji polyfit na podstawie wartości wyznaczonych dla kilkudziesięciu poziomów zaburzenia. Jeżeli zachodzi taka potrzeba podzielić przedział zmienności na dwa podprzedziały. Addytywne zaburzenie danych zrealizować za pomocą liczb pseudolosowych o zerowej wartości oczekiwanej i wariancji, wygenerowanych za pomocą zaimplementowanej w Matlabie funkcji randn.*

5.2 Opis zastosowanych algorytmów numerycznych

Zadanie cztery jest realizowane analogicznie jak zadanie trzecie. Korzysta z napisanych wcześniej funkcji, z wyjątkiem że po wyliczeniu węzłów addytywnie zaburzamy yn. Następnie przy pomocy zapisywania macierzowego zapisujemy błąd, który jest widoczny na wykresie.

5.3 Przykład działania programu



rys.15 Wskaźnik błędu średniokwadratowego dla N=5:50 i K<N.

5.4 Wnioski

Na wykresie obserwujemy błąd, który utrzymuje się na poziomie stałym dla zaburzeń w zakresie $[10^{-5}; 5{*}10^{-4}]$. Rząd wielkości błędu rośnie proporcjonalnie do wielkości zaburzenia.

5.Kod funkcji w języku programu MATLAB

```
clear all
close all

N = [10 20 30];
K = 8;
x100= linspace(-1,1,100);
x1000 = linspace(-1,1,1000);


%wzór funkcji - funkcja anonimowa
F=@(u) (u+1/3).*(u+1/3)+exp(-u-2);
iks=@(w) (2*((1:w)-1))/(w-1)-1;


%%%Zadanie 1 - sprządzanie wykresu funkcji
for i=1:length(N)
   [xn, yn] = wyliczWezly(N(i));

   figure;
   plot(xn,yn,'o',x1000,F(x1000));
   title(['Wykres funkcji f(x) na podstawie ciągu próbek N=',num2str(N(i))]);
   legend({'ciąg próbek N','funkcja y(x)'},'Location','Best');
   xlabel("x");  %nazwa osi x
   ylabel("f(x)");   %nazwa osi y
   grid on;
   hold on;
end


%%%Zadanie 2 - aproksymacja funkcji dla kilku par N i K
N2 = [10 20 30];
K2 = [3 6 15];
for i = 1:length(N2)
   for j = 1:length(K2)
      [xn, yn] = wyliczWezly(N2(i));
      [p, xK] = macierzP(xn, yn, K2(j));
      [yZ] = wyznaczFunkAp(xK, p, K2(j), xn);

      figure;
      plot(x100,yZ,'r',xn,F(xn),'*',x1000,F(x1000),'g' );
      title(['Wykres funkcji f(x) po aproksymacji K=',num2str(K2(j)),'dla N=',num2str(N2(i))]);
      legend({'aproksymacja funkcji','ciąg próbek f(x)','funkcja y'},'Location','Best');
      xlabel("x");  %nazwa osi x
      ylabel("f(x)");   %nazwa osi y
      grid on;
      hold on;
   end

end
```

```
N=5:50;
%%%Zadanie 3 - wykresy błędów
for i = N
[xn, yn] = wyliczWezly(i);
   for j = 1: i-1
   [p, xK] = macierzP(xn, yn, j);
   [yZ] = wyznaczFunkAp(xK, p, j);
   [bladSra, bladMaxa] = wyznaczBledy(yZ);
   bladSr(i,j) = bladSra;
   bladMax(i,j) = bladMaxa;
   end
end
figure;
title("Wykres wskaźniks błędu średniokwadratowego");
surf(log10(bladSr));     %rysuje wykres
xlabel("N");  %nazwa osi x
ylabel("K");    %nazwa osi y
zlabel("wskaźnik błędu średniokwadratowego"); %nazwa osi z
grid on;     %tworzenie linie na wykresie
hold on;
figure;
title("Wykres wskaźnika błędu maksymalnego");
surf(log10(bladMax));     %rysuje wykres
xlabel("N");  %nazwa osi x
ylabel("K");    %nazwa osi y
zlabel("wskaźnik błędu maksymalnego"); %nazwa osi z
grid on;       %tworzenie linie na wykresie
hold on;



%%%Zadanie 4 - aproksymacja przy zaburzonych danych
bladSred1=[];
for z = logspace(-5,-1)
   for i = N   %N = 5:50
   [xn, yn] = wyliczWezly(i);
   yn = yn + randn(size(yn))*z; %addytywne zaburzenie danych
      for j = 3: i-1         %K<N
      [p, xK] = macierzP(xn, yn, j);
      [yZ] = wyznaczFunkAp(xK, p, j);
      [bladSra, bladMaxa] = wyznaczBledy(yZ);
      bladSred(i-4,j-2) = bladSra;
      end
   end
   bladSred1 = [bladSred1;min(min(bladSred(bladSred>0)))];
end
figure;
semilogx(logspace(-5,-1), bladSred1, 'o');
xlabel("delta [10-5;10-1]");
ylabel("Wartości błędu średniokwadratowego");
title("Wykres wartości błędu średniokwadratowego");
hold on;
p=polyfit(logspace(-5,-1),bladSred1',7);
semilogx(logspace(-5,-1),polyval(p, logspace(-5,-1)),'r');
hold on;
grid on;

%% Funkcja wyliczająca węzły
function [xn, yn] = wyliczWezly(N)
F =@(u) (u+1/3).*(u+1/3)+exp(-u-2);
iks=@(w) (2*((1:w)-1))/(w-1)-1;
x1000 = linspace(-1,1,1000);
for i = 1:N
   xn = iks(i);
   yn = F(xn);
end
end


%% Funkcja wyznaczająca macierz-fi
function [p, xK] = macierzP(xn, yn, K)
iks=@(u) (2*((1:u)-1))/(u-1)-1;
[z,z]=size(xn);
W = zeros(z,K);
xK = zeros(K);
xK = iks(K);
for n = 1:z
   for k = 1:K
      if (abs(xn(n) - xK(k)) < 1)
         W(n,k) = 2*(abs(xn(n)-xK(k)))^3 - 3*((xn(n)-
xK(k))^2)+1;
      else
         W(n,k) = 0;
      end
   end
end
p = (W'*W) \ (W'*(yn)');
end


%% Funkcja aproksymująca węzły
function [yZ] = wyznaczFunkAp(xK, p, K, xn)
z=100;
Fz=@(u) (u+1/3).*(u+1/3)+exp(-u-2);
x100 = linspace(-1,1,z);
x1000 = linspace(-1,1,1000);
for n = 1:z
   for k = 1:K
      if (abs(x100(n) - xK(k)) < 1)
         Wp(n,k) = 2*(abs(x100(n)-xK(k)))^3 - 3*((x100(n)-
xK(k))^2)+1;
      else
         Wp(n,k) = 0;
      end
   end
end
%mnzenie każdego elementu do wykresu
for n=1:z
      yZ(n)=sum(Wp(n,:).*p');
end
end


%% Funkcja zwracająca błąd średniokwadratowy i maksymalny
function [bladSra, bladMaxa] = wyznaczBledy(yZ)
x = linspace(-1,1,100);
Fun =@(u) (u+1/3).*(u+1/3)+exp(-u-2);
bladSra=norm(yZ-Fun(x))/norm(Fun(x));
bladMaxa=norm(yZ-Fun(x),Inf)/norm(Fun(x),Inf);
end
```

6. Bibliografia

1. Morawski Roman, „Metody numeryczne (MNUB) Materiały do wykładu prowadzonego w semestrze zimowym 2017/2018”
2. Morawski Roman, „Zadania z rozwiązaniami do przedmiotów WNUM MNUB”
3. Pratap Rudra, „MATLAB 7 dla naukowców i inżynierów”, Warszawa, PWN, 2013
4. „Wstęp do metod numerycznych dla studentów elektroniki i technik informacyjnych” pod red. Romana Z. Morawskiego, Warszawa, Oficyna wydawnicza Politechniki Warszawskiej, 2009
5. **https://www.mathworks.com/**