Nazwa przedmiotu: **Metody numeryczne**	Temat i nr ćwiczenia: **4. Aproksymacja – metoda najmniejszych kwadratów**	Prowadzący: **Dr hab. Inż. Marcin Hojny**
Imię nazwisko: **Amelia Nalborczyk**		Grupa laboratoryjna: **4**
Data wykonania: **22.03.2024 r.**		Data oddania: **02.04.2024 r.**

1. Cel ćwiczenia

Celem ćwiczenia jest zapoznanie się z zagadnieniem aproksymacji oraz implementacja programu który dopasowuje funkcję liniową do podanych punktów korzystając z metody najmniejszych kwadratów.

2. Wstęp teoretyczny

Aproksymacja polega na znalezieniu najlepszego dopasowania lub przybliżenia funkcji zadanego zbioru danych za pomocą innej funkcji. Aproksymacja ułatwia analizę i obliczenia, co ma zastosowanie w modelowaniu zjawisk fizycznych, analizie danych, prognozowaniu, optymalizacji, grafice komputerowej. Aproksymacja w dziedzinie analizy danych pozwala na stworzenie modeli lub funkcji, które najlepiej odpowiadają danym punktom doświadczalnym, co ułatwia zrozumienie związków między zmiennymi oraz prognozowanie zachowań podstawie zebranych danych.

W ramach laboratorium poznajemy metodę aproksymacji – regresję liniową. Celem regresji liniowej jest znalezienie najlepszego dopasowania prostej linii, która jak najbardziej odpowiada zbiorowi danych. Technika używana jest do analizy związku między dwiema zmiennymi, zmiennej zależnej i zmiennej niezależnej. Wyznaczana prosta jest postaci:

$$y = a_0 + a_1 x \tag{1}$$

Gdzie:

x – zmienna zależna

y – zmienna niezależna

a_0 – współczynnik kierunkowy prostej

a_1 – wyraz wolny

Współczynniki a_0 i a_1 są szukanymi w metodzie regresji liniowej. Poszukiwanie tych parametrów realizujemy poprzez minimalizację sumy kwadratów różnic pomiędzy wartościami zmiennych zależnych a wartościami przewidywanymi:

$$S(a,b) = \sum_{i=1}^{n} [y_i - y(x_i)]^2 = \sum_{i=1}^{n} [y_i - a_0 - a_1 x_i]^2 \tag{2}$$

Funkcja wielu zmiennych ma minimum w punkcie, dla którego pochodne cząstkowe po tej funkcji po wszystkich zmiennych wynoszą 0.

$$\frac{\partial S(a_0, a_1)}{\partial a_0} = 0 \tag{3}$$

$$\frac{\partial S(a_0, a_1)}{\partial a_1} = 0 \tag{4}$$

Po wyliczeniu pochodnych możemy wyznaczyć parametry a_0 i a_1:

$$a_0 = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} {x_i}^2 - (\sum_{i=1}^{n} x_i)^2} \tag{5}$$

$$a_1 = \frac{\sum_{i=1}^{n} y_i \sum_{i=1}^{n} {x_i}^2 - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i x_i}{n\sum_{i=1}^{n} {x_i}^2 - (\sum_{i=1}^{n} x_i)^2} \tag{6}$$

W praktyce, regresja często wykorzystuje współczynnik korelacji do oceny związku między zmiennymi. Współczynnik korelacji służy do mierzenia siły i kierunku związku między dwiema zmiennymi. Jest to liczba z zakresu od -1 do 1, gdzie:

- 1 oznacza doskonałą dodatnią korelację,
- -1 oznacza doskonałą ujemną korelację, a
- 0 oznacza brak korelacji liniowej między zmiennymi.

Współczynnik korelacji wyraża się wzorem:

$$R = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{\sqrt{n\sum_{i=1}^{n} {x_i}^2 - (\sum_{i=1}^{n} x_i)^2}\sqrt{n\sum_{i=1}^{n} {y_i}^2 - (\sum_{i=1}^{n} y_i)^2}} \tag{7}$$

3. Implementacja

Aproksymację metodą najmniejszych kwadratów implementuje w języku programowania c++, korzystając z środowiska programistycznego Visual Studio.

W programie wykorzystuję obsługę pliku tekstowego, którą realizuję przy pomocy biblioteki fstream. W pliku tekstowym data.txt znajdują się ilość punktów size oraz wartości x i y w punktach. W poniższym fragmencie kodu (Fragment programu 1) realizuję obsługę pliku „data.txt", odczyt danych do zmniennej size oraz do macierzy M.

W dalszej części kodu u wywołuje funkcję wyliczającą współczynniki a_0 i a_1 oraz współczynnik korelacji R.

```
    fstream read("data.txt");

    if (read.is_open()) {
        int size;
        read >> size;

        double** M = new double* [size];

        for (int i = 0; i < size; i++) {
            M[i] = new double[2];
        }

        for (int i = 0; i < size; i++)
            read >> M[i][0] >> M[i][1];
```

Fragment kodu 1: obsługa pliku.

Następnym krokiem w działaniu programu jest wykonanie obliczeń opartych o aproksymację metodą najmniejszych kwadratów. Wszystkie obliczenia zamieszczam w funkcji o nazwie akroksymacja. W funkcji współczynniki korzystając ze wzorów (5) i (6) oraz współczynnik korelacji ze wzoru (7).

Do obliczenia naszych szukanych wykorzystuje zmienne pomocnicze: suma_x, suma_y, suma_x2, suma_y2, suma_xy w których sumuje wartości, będące wykorzystywane potem we wzorach (5) i (6). Do podniesienia wartości do potęgi używam funkcji pow(podtswa, wykładnik) z biblioteki cmath. Następnie wykorzystując zmienne pomocnicze wyznaczam współczynniki a_0 i a_1 oraz współczynnik korelacji R jako zmienne kolejno b, a, R . Fragment funkcji interpolacja zamieszczam jako Fragment kodu 2.

```
for (int i = 0; i < size; ++i) {
    suma_x += M[i][0];
    suma_y += M[i][1];
    suma_x2 += pow(M[i][0], 2);
    suma_y2 += pow(M[i][1], 2);
    suma_xy += M[i][1] * M[i][0];
}

a = (size * suma_xy - suma_x * suma_y) / (size * suma_x2 - pow(suma_x, 2));
b = (suma_y * suma_x2 - suma_x * suma_xy) / (size * suma_x2 - pow(suma_x, 2));
R = (size * suma_xy - suma_x * suma_y) / (sqrt(size * suma_x2 - pow(suma_x, 2)) *
sqrt(size * suma_y2 - pow(suma_y, 2)));
```

Fragment kodu 2: wyznaczenie współczynniki a_0 i a_1 oraz współczynnik korelacji R.

W następnym fragmencie funkcji aproksymacja następuje wypisanie komunikatu opisującego poziom korelacji (idealny, dobry lub zły) na podstawie wartości zmiennej r, która jest współczynnikiem korelacji. Zmienna pomocnicza r jest wartością bezwzględną zmiennej R w której przechowywana jest oryginalna wartość współczynnika korelacji, zmienna r utworzona została w celu zmniejszenia ilości warunków w wydrukowanym tekście. Fragment funkcji aproksymacja zamieszczam jako Fragment Kodu 3.

```
double r = abs(R);
if (r == 1)
    cout << "wartosci sa idealne skorelowane, wspolczynnik korelacji wynosi " <<
R << endl;
else if (r > 0.2 && r < 1)
    cout << "wartosci sa dobrze skorelowane, wspolczynnik korelacji wynosi " << R
<< endl;
else
    cout << "wartosci sa zle skorelowane, wspolczynnik korelacji wynosi " << R <<
endl;
```

Fragment kodu 3: Interpretacja współczynnika korelacji.

W dalszej części funkcji aproksymacja generuje równanie prostej w zależności od wartości współczynników a i b, uwzględniając różne przypadki dla wartości b (czy jest większe od zera, równe zero lub mniejsze od zera). Kod zamieszczam jako Fragment kodu 4.

```
    if (b > 0)
        cout << "y=" << a << "x+" << b << endl;
    else if (b==0)
        cout << "y=" << a << "x"<< endl;
    else
        cout << "y=" << a << "x" << b << endl;
```

Fragment kodu 4: wydrukowanie równania dopasowanej funkcji.

Dodatkowo całokształt funkcji aproksymacja zamieszczam jako Fragment kodu 5.

```
void aproksymacja(double** M, double& a, double& b, int size, double& R) {
    double suma_x = 0, suma_y = 0, suma_x2 = 0, suma_xy = 0, suma_y2 = 0;
    //zmienne pomocnicze do regresji i Przejscie przez wszystkie punkty
    for (int i = 0; i < size; ++i) {
        suma_x += M[i][0];
        suma_y += M[i][1];
        suma_x2 += pow(M[i][0], 2);
        suma_y2 += pow(M[i][1], 2);
        suma_xy += M[i][1] * M[i][0];
    }

    a = (size * suma_xy - suma_x * suma_y) / (size * suma_x2 - pow(suma_x, 2));
    b = (suma_y * suma_x2 - suma_x * suma_xy) / (size * suma_x2 - pow(suma_x,
2));
    R = (size * suma_xy - suma_x * suma_y) / (sqrt(size * suma_x2 - pow(suma_x,
2)) * sqrt(size * suma_y2 - pow(suma_y, 2)));

    //Wspoczynnik korelacji R
    if (R == 1)
        cout << "wartosci sa idealne skorelowane, wspolczynnik korelacji wynosi"
<< R << endl;
    else if (R > 0 && R < 1)
        cout << "wartosci sa dobrze skorelowane, wspolczynnik korelacji wynosi"
<< R << endl;
    else
        cout << "wartosci sa zle skorelowane, wspolczynnik korelacji wynosi" << R
<< endl;

    if (b > 0)
        cout << "y=" << a << "x+" << b << endl;
    else if (b==0)
        cout << "y=" << a << "x"<< endl;
    else
        cout << "y=" << a << "x" << b << endl;
}
```

Fragment kodu 5: Funkcja aproksymacja.

4. Testy jednostkowe

Test 1.1

Punkty podane w pliku data.txt są punktami funkcji $f(x) = 3x - 2$. Punkty znane są podawane przy uruchomieniu programu, na Zdjęciu.

```
0 -2
1 1
2 4
-2 -8
3 7
wartosci sa idealne skorelowane, wspolczynnik korelacji wynosi 1
y=3x-2
```

Zdjęcie 1: Test 1.1.

Wyniki działania programu:

$a_0 = $-2

$a_1 = 3$

$R = 1$

Wyniki narzędzia REGLIMP oraz PEARSON w programie Excel:

$a_0 = $-2

$a_1 = 3$

$R = 1$

Test 1.2

Punkty podane w pliku data.txt są punktami funkcji $g(x) = -3x + 2$. Punkty znane są podawane przy uruchomieniu programu, na Zdjęciu .

```
0 2
1 -1
2 -4
-2 8
3 -7
wartosci sa idealne skorelowane, wspolczynnik korelacji wynosi -1
y=-3x+2
```

Zdjęcie 2: Test 1.2.

.

Wyniki działania programu:

$a_0 = 2$

$a_1 = $-3

$R = $-1

Wyniki narzędzia REGLIMP oraz PEARSON w programie Excel:

$a_0 = 2$

$a_1 = $-3

$R = $-1

Test 2.1

W pliku data.txt znajduje się 5 punktów, które wydrukowywane są podczas uruchomienia programu, na Zdjęciu 2.

```
1 0
2 1
3 3.5
3.5 4
5 7
wartosci sa dobrze skorelowane, wspolczynnik korelacji wynosi0.992889
y=1.79891x-2.11685
```

Zdjęcie 3: Test 2.1.

Wyniki działania programu:

a_0 = -2.11685

a_1 = 1.79891

R = 0.992889

Wyniki narzędzia REGLIMP oraz PEARSON w programie Excel:

a_0 = -2,11685

a_1 = 1,798913

R = 0.992889

Test 3.1

Punkty podane w pliku data.txt są punktami losowymi z danych przedziałów. Jest ich 20. Punkty znane są podawane przy uruchomieniu programu, na zdjęciu 5.

```
16.65 0.48
17.9 0.88
16.73 0.42
16.52 0.41
17.83 0.72
17.21 0.42
17.11 0.67
17.07 0.15
17.99 0.44
17.23 0.07
16.58 0.7
16.78 0.65
17.11 0.39
16.78 0.32
16.98 0.95
16.99 0.97
17.01 0.95
16.51 0.45
16.56 0.15
17.88 0.34
wartosci sa zle skorelowane, wspolczynnik korelacji wynosi 0.126442
y=0.0718204x-0.699545
```

Zdjęcie 4: Test 3.1.

Wyniki działania programu:

$a_0 =$ -0.699545

$a_1 = 0.0718204$

$R = 0.126442$

Wyniki narzędzia REGLIMP oraz PEARSON w programie Excel:

$a_0 =$ -0,6897

$a_1 = 0{,}07127$

$R = 0{,}124823$

5. Opracowanie wyników

Zestawienie wszystkich wyznaczonych współczynników funkcji oraz współczynnika korelacji zamieszczam jako Tabela 1.

Tabela 1: Opracowanie wyników.

	Excel			Program		
	a_0	a_1	R	a_0	a_1	R
Test 1.1	-2	3	1	-2	3	1
Test 1.2	2	-3	-1	2	-3	-1
Test 2.1	-2,116848	1,798913	0,992889	-2.11685	1,798910	0.992889
Test 3	-0,689703	0,071270	0,124823	-0.699545	0.0718204	0.126442

W teście 1.1 analizowano punkty danych z pliku data.txt, które reprezentują punkty funkcji $f(x) = 3x - 2$. Wyniki działania programu oraz narzędzi analizy danych (REGLIMP i PEARSON w Excelu) są zgodne i dobrze odzwierciedlają funkcję f(x), a współczynnik korelacji R wskazuje na doskonałą liniową zależność między zmiennymi.

W teście 1.2 analizowano punkty danych z pliku data.txt, które są punktami funkcji $g(x) = -3x + 2$. Otrzymane wyniki w programie i w Excel są zgodne i również dobrze odzwierciedlają funkcję g(x), a wartość współczynnika korelacji R wskazuje na silną negatywną zależność między zmiennymi.

W teście 2.1 analizowano 5 punktów danych z pliku data.txt. Parametry regresji liniowej dobrze dopasowują się do danych, a wysoka wartość współczynnika korelacji potwierdza silną zależność między zmiennymi. Różnice wartości parametrów wyznaczonych przez program i Excel wynikają z dokładności wyników, wyższa w Excel.

Wyniki Testu 3.1 wskazują na brak znaczącej liniowej zależności między zmiennymi, co oznacza słabą zgodność wyników z modelem liniowym. Dla takich danych należałoby rozważyć inne modele regresji lub inne techniki analizy danych, które lepiej odpowiadają charakterowi danych losowych z przedziałów. Wyniki uzyskane w skutek

obliczeń programu i użycia narzędzi w Excel nieco się różnią. Zakładając, że Excel podaje wynik bardziej bliski rzeczywistemu, wyznaczam błędy względne, by ustalić dokładność wyników.

Dla współczynnika a_0 błąd względny wynosi 1.42%

Dla współczynnika a_1 błąd względny wynosi 0.77%

Dla współczynnika korelacji R błąd względny wynosi 1.30%

Wyniki programu mają niewielkie odchylenie, co sugeruje, że program może być stosunkowo dokładny, ale istnieje miejsce na poprawę dokładności wyników, szczególnie dla danych o słabej liniowej zależności.

Wnioskiem z tej analizy jest zalecenie rozważenia innych modeli regresji lub technik analizy danych, które lepiej odpowiadają charakterowi danych losowych z przedziałów.

6. Wnioski

Analiza aproksymacji metodą najmniejszych kwadratów wskazuje na jej skuteczność w dopasowywaniu funkcji do zbioru danych. Metoda ta pozwala na znalezienie najlepszego przybliżenia funkcji. Zgodność programu z rzeczywistością zależy od dokładności obliczeń i precyzji wyników. Im dokładniejsze wyniki programu, tym lepiej odzwierciedlają one rzeczywiste zależności między zmiennymi.

Mimo odchyleń w obliczeniach przy mocno nieskorelowanych danych nie możemy zarzucić błędnego działania programowi, gdyż zgodność programu z rzeczywistością można oceniać poprzez analizę współczynnika korelacji R, który sugerował inną dokładność wyników w teście 3.1.

7. Źródła

1. Prezentacja „Metody numeryczne. Aproksymacja – metoda najmniejszych kwadratów” dr hab. inż. Marcin Hojny, prof. AGH