

Grupa lab.3	Data wykonania 24.05.2024	Inżynieria Obliczeniowa 2023/2024
Temat ćwiczenia Aproksymacja średniokwadratowa w bazie jednomianów		
Imię i nazwisko Karolina Kurowska		Ocena i uwagi

Cel ćwiczenia:

Celem ćwiczenia jest poznanie metod aproksymacji średniokwadratowej oraz implementacja programu obliczającego współczynniki dla wielomianu aproksymującego dowolnego stopnia. Zadanie obejmuje również aproksymację za pomocą funkcji liniowej dla danych punktów oraz zastosowanie metody eliminacji Gaussa do rozwiązania układu równań.

Wstęp:

Aproksymacja polega na zastąpieniu funkcji trudnej do obliczeń numerycznych inną, bardziej dogodną funkcją, która jest bliska funkcji wyjściowej. Stosuje się ją również do wyznaczania wartości funkcji danej dyskretnie na skończonej liczbie punktów w innych punktach obszaru lub do znalezienia gładkiej funkcji ciągłej przechodzącej w pobliżu zadanych punktów. W naszym przypadku stosujemy aproksymację średniokwadratową, która minimalizuje sumę kwadratów różnic między wartościami funkcji a wartościami aproksymującymi.

Implementacja ćwiczenia:

Potrzebne dane jak wartości wag i węzłów aproksymacji wczytywano z plików txt:

```
int main ()
{
    int m;
    fstream plik_x, plik_y, plik_w;
    //-----X & m
    plik_x.open ("wezly_x.txt", ios::in);
    if (!plik_x.is_open ()) {
        cout << "Plik nie istnieje\n";
        exit (0);
    }
    plik_x >> m;
    cout << "Liczba wezlow: " << m << endl;
    cout << "Wezly aproksymacji" << endl;
    cout << "x:";
    double* x_matrix = wczytaj_matrix (m, plik_x);
    plik_x.close ();

    //-----Y
    plik_y.open ("wezly_y.txt", ios::in);
    if (!plik_y.is_open ()) {
        cout << "Plik nie istnieje\n";
        exit (0);
    }
    cout << endl << "y:";
    double* y_matrix = wczytaj_matrix (m, plik_y);
    plik_y.close ();

    //-----W
    plik_w.open ("wezly_w.txt", ios::in);
    if (!plik_w.is_open ()) {
        cout << "Plik nie istnieje\n";
        exit (0);
    }
    cout << endl << "wagi:";
    double* w_matrix = wczytaj_matrix (m, plik_w);
    plik_w.close ();

    //-----n
    int stop_wielo;
    stop_wielo = 1; //WYBIERZ STOPNIEN WIELOMIANU
    cout << endl << "Stopien wielomianu: " << stop_wielo << endl << endl;
    //-----koniec wczytywania danych-----//
}
```

Następnie zaimplementowano wzory z instrukcji do obliczenia macierzy g i wektora F:

```
double** g_matrix = new double* [stop_wielo + 1];
for (int i = 0; i < stop_wielo + 1; i++) {
    g_matrix[i] = new double[stop_wielo + 2]; // Zwiększam o 1 dla kolumny wektora F

    for (int j = 0; j < stop_wielo + 1; j++) {
        g_matrix[i][j] = suma (m, x_matrix, w_matrix, i, j);
    }
}
cout << "Macierz g: " << endl;
wypisz_macierz (stop_wielo + 1, g_matrix);

cout << "Wektor F: " << endl;
double* F = new double[stop_wielo + 1];
for (int i = 0; i < stop_wielo + 1; i++) {
    F[i] = 0;

    for (int j = 0; j < m; j++) {
        F[i] += pow (x_matrix[j], i) * y_matrix[j] * w_matrix[j];
    }
    g_matrix[i][stop_wielo + 1] = F[i];
    cout << "F[" << i << "]" << F[i] << endl;
}
cout << endl;
```

```
double suma (double m, double* x_matrix, double* waga, int k, int j)
{
    double Suma = 0;
    for (int i = 0; i < m; i++)
    {
        Suma += pow (x_matrix[i], k) * pow (x_matrix[i], j) * waga[i];
    }
    return Suma;
}
```

Następnie rozwiązano układ równań programem napisanym przez nas podczas laboratoriów nr. 3. Ostatecznie obliczono nowe wartości w węzłach aproksymacji.

```
//nowe wartosci
cout << endl << "Wartosci po aproksymacji: " << endl;
y_matrix = nowe_Y(wyniki, x_matrix, m, stop_wielo);
```

```
double* nowe_Y (double* a, double* x_matrix, int m, int stop_wielo)
{
    double* new_matrix_y = new double[m];

    for (int i = 0; i < m; i++) {
        double nowy = 0;

        for (int j = 0; j < (stop_wielo + 1); j++)
        {
            nowy += a[j] * pow (x_matrix[i], j);
        }
        new_matrix_y[i] = nowy;
        cout << "Y[" << i+1 << "]" << " = " << new_matrix_y[i] << endl;
    }
    return new_matrix_y;
}
```

Wyniki:

```
Liczba wezlow: 8
Wezly aproksymacji
x: 1 2 3 4 5 6 7 8
y: 2 4 3 5 6 9 11 11
wagi: 1 1 1 1 1 1 1 1
Stopien wielomianu: 1

Macierz g:
8 36
36 204

Wektor F:
F[0]51
F[1]288

Macierz po postepowaniu prostym:
8 36
0 42

Wyniki:
a0 = 0.107143
a1 = 1.39286

Wartosci po aproksymacji:
Y[1] = 1.5
Y[2] = 2.89286
Y[3] = 4.28571
Y[4] = 5.67857
Y[5] = 7.07143
Y[6] = 8.46429
Y[7] = 9.85714
Y[8] = 11.25
```

Wnioski:

Ćwiczenie pokazuje, że metoda aproksymacji średniokwadratowej pozwala na uzyskanie funkcji przybliżającej, która dobrze oddaje trend danych punktów. Implementacja programu w języku C++ umożliwiła automatyzację procesu obliczeń oraz uproszczenie analizy wyników. Podsumowując, laboratorium pozwoliło na praktyczne zrozumienie i zastosowanie technik aproksymacji średniokwadratowej, co jest niezwykle przydatne w wielu dziedzinach inżynierii i nauk stosowanych.