Pracownia 2

Analiza Numeryczna (M)
Prowadzący: Witold Karczewski
Zadanie P2.9

Wojciech Adamiec, 310064

30 grudnia 2019

Spis treści

1 Wstęp

1.1 Treść Zadania

Wykorzystaj aproksymację średniokwadratową, aby dopasować funkcję postaci $y = ab^x$ do danych:

x	1	2	3	...	m
y	y_1	y_2	y_3	...	y_m

Opracowaną metodę przetestuj na wybranych przykładach.

1.2 Plan Działania

Zgodnie z poleceniem, do rozwiązania naszego problemu zastosujemy aproksymację średniokwadratową. Nie możemy jednak zastosować jej wprost, bowiem funkcja postaci $y = ab^x$ nie jest wielomianem. Aby rozwiązać ten problem, dokonamy kilku przekształceń:

$$y = ab^x$$

Logarytmujemy obustronnie równianie:

$$ln(y) = ln(ab^x)$$

Korzystamy z własności logarytmów:

$$ln(y) = ln(a) + ln(b^x)$$

$$ln(y) = ln(a) + x \cdot ln(b)$$

Zauważamy, że mamy już teraz postać wielomianu - w naszym przypadku funkcji liniowej. Użyjmy teraz zmiennych pomocniczych: $Y = ln(y)$, $A = ln(a)$ i $B = ln(b)$. Dostajemy wówczas:

$$Y = A + x \cdot B$$

Teraz możemy już bez problemu zastosować aproksymację średniokwadratową. Szukamy minimum wyrażenia:

$$||Y - (A + x \cdot B))||^2$$

W tym celu musimy rozwiązać równanie:

$$\begin{bmatrix} \langle 1, 1 \rangle & \langle 1, x \rangle \\ \langle x, 1 \rangle & \langle x, x \rangle \end{bmatrix} \cdot \begin{bmatrix} A \\ B \end{bmatrix} = \begin{bmatrix} \langle Y, 1 \rangle \\ \langle Y, x \rangle \end{bmatrix}$$

W naszym przypadku przyjmujemy dyskretną definicję iloczynu skalarnego, co upraszcza nam równanie do:

$$\begin{bmatrix} m & \sum_{k=1}^{m} x_k \\ \sum_{k=1}^{m} x_k & \sum_{k=1}^{m} x_k^2 \end{bmatrix} \cdot \begin{bmatrix} A \\ B \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^{m} Y_k \\ \sum_{k=1}^{m} Y_k x_k \end{bmatrix}$$

Teraz pozostaje nam rozwiązać układ 2 równań liniowych, aby otrzymać A i B. Końcowym krokiem będzie wyliczenie a i b zgodnie ze wzorami:

$$a = e^A$$

$$b = e^B$$

Zatem nasz plan działania sprowadza się do:

- Obliczenia $Y_k = ln(y_k)$ dla $k = 1, 2, 3, ..., m$
- Obliczenia $\sum_{k=1}^{m} x_k$ oraz $\sum_{k=1}^{m} x_k^2$
- Obliczenia $\sum_{k=1}^{m} Y_k$ oraz $\sum_{k=1}^{m} Y_k x_k$
- Rozwiązania układu 2 równań liniowych
- Obliczeniu a i b

Do zadania dołączony jest program, który wykonuje wszystkie te czynności i od razu zwraca wartości a i b.

1.3 Sprawy Techniczne

Wszystkie obliczenia numeryczne zostały wykonane w języku *Julia* w wersji *v1.2.0*, w podwójnej precyzji - *binary64* zgodnie ze standardem *IEEE 754*, którego opis można znaleźć np. tu: `https://en.wikipedia.org/wiki/IEEE_754`

2 Rozwiązanie Zadania

W treści zadania nie mamy żadnych konkretnych danych, zatem naszą metodę zastosujemy do kilku własnoręcznie wymyślonych przykładów. Będziemy je tworzyć w taki sposób, że najpierw wybierzemy pewną funkcję postaci $y = ab^x$, następnie dla m różnych wartości x odczytamy i zapiszemy odpowiadające wartości y. Na koniec dokonamy lekkiego *zaburzenia* wartości x i y, tak aby zasymulować rzeczywistą sytuację.

2.1 Przykład 1

Dla sprawdzenia poprawności przedstawionej metody zacznijmy od prostej funkcji $y = 2^x$. W tym przykładzie wyjątkowo nie dokonamy zaburzenia wartości, aby upewnić się, że nasza metoda jest poprawna. Jako dane weźmiemy:

x	1	2	3	4	5
y	2	4	8	16	32

Czyli $m = 5$.
Stosując nasz algorytm otrzymujemy:
$a = 1$ oraz $b = 2.0000000000000004$, zatem otrzymaliśmy w niemal idealnym przybliżeniu funkcję wejściową. Pozwala to nabrać intuicji, że nasz algorytm faktycznie działa.

2.2 Przykład 2

Weźmy teraz funkcję $y = 2 \cdot 3^x$. Zaburzymy nieznacznie wartości - nasze dane będą takie:

x	1	2	3	4	5
y	5.87	18.05	54.1	162.6	487.1

Stosując nasz algorytm otrzymujemy:
$a \approx 1.9662069$ i $b \approx 3.014812$, w poszczególnych punktach daje nam to:

y	5.87	18.05	54.1	162.6	487.1
y'	5.93	17.87	53.88	162.43	489.70
$\|\Delta\|$	0.06	0.18	0.22	0.17	2.6

2.3 Przykład 3

Sprawdźmy teraz jakość algorytmu w zależności od ilości badanych punktów. W tym celu weźmy teraz funkcję $y = 7.4284 \cdot 1.5672^x$ dla $m = 3$ i $m = 5$.
Zacznijmy od $m = 3$:

x	12.5677	23.2512	44.1234
y	2104.556	255724.457	3023070299.5619

Otrzymujemy:
$a \approx 7.428399896$ i $b \approx 1.5672$ Zauważamy, że już dla $m = 3$ wynik jest niemal perfekcyjny, wynika to z tego, że nie zaburzyliśmy naszych danych. Zróbmy zatem lekkie zaburzenie:

x	12.56	23.25	44.12
y	2102	255725	3023070315

Otrzymujemy wówczas:
$a \approx 7.441683$ i $b \approx 1.567179$
Dołóżmy dwa punktu do naszych danych (lekko zaburzone, $m = 5$), sprawdzimy jak zmieni się nasze dopasowanie:

x	12.56	16.2	23.25	25.5	44.12
y	2102	10760	255725	702365	3023070315

Otrzymujemy:
$a \approx 7.43292$ i $b \approx 1.567213$
Otrzymane dopasowanie jest trochę lepsze, ale nie jesteśmy w stanie stwierdzić, czy to ze względu na dodatkowe punkty czy przez *inne* zaburzenie dwóch dodanych punktów. Łatwo jednak dostrzec, że różnica jest niewielka.

3 Podsumowanie

Metoda aproksymacji średniokwadratowej pozwala szybko i z dużą dokładnością dopasowywać funkcję postaci $y = ab^x$ do odpowiednich danych. Przykład 1 pokazał, że metoda pozwala niemal perfekcyjnie odtworzyć funkcję wykładniczą na bazie zaledwie kilku punktów. Pozostałe przykłady pozwalają nabrać przekonania, że metoda nie tylko działa, ale również daje stosunkowo dokładny wynik.