Sprawozdanie do zadania **P2.03**

Maurycy Borkowski

4.12.2020

Przedstawienie problemu

W zadaniu mamy przedstawić algorytm konstrukcji wielomianu w możliwie najniższego stopnia, który dla $\varepsilon > 0$ spełnia nierówność:

$$\int_0^1 x[f(x) - w(x)]^2 dx < \varepsilon \tag{1}$$

Szukamy więc wielomianu optymalnego w^* w sensie aproksymacji średnio kwadratowej dla funkcji wagowej $p(x) = x$.
Najniższego stopnia żeby spełniony był warunek (1).

Mamy zdefiniowany iloczyn skalarny:

$$\langle f, g \rangle = \int_0^1 p(x)f(x)g(x)dx = \int_0^1 xf(x)g(x)dx \tag{2}$$

Szukany w^* ma spełniać:

$$\|f - w^*\|^2 = \langle f - w^*, f - w^* \rangle = \int_0^1 x[f(x) - w^*(x)]^2 dx < \varepsilon \tag{3}$$

Szukanie wielomianu optymalengo

Przypomnijmy potrzebne twierdzenia z wykładu:

Dla dowolnego ciągu wielomianów ortogonalnych $\{P_k\} \subset C_p[0, 1]$, wielomian optymalny n-tego stopnia istnieje i wyraża się jednoznacznie wzorem:

$$w_n^*(x) = \sum_{i=0}^{n} \frac{\langle f, P_i \rangle}{\langle P_i, P_i \rangle} P_i(x) \tag{4}$$

a n-ty błąd ma postać:

$$\|f - w^*\| = \sqrt{\|f\|^2 - \sum_{i=0}^{n} \frac{\langle f, P_i \rangle^2}{\langle P_i, P_i \rangle}} \tag{5}$$

Zwiększanie stopnia wielomianu w_n^* zwiększa dokładność tj. zmniejsza $\|f - w_n^*\|$.

Z powyższych obserwacji, można zaproponować wstępny algorytm szukania w^*

---

input: ε, f
$n \leftarrow 0$
$w^* \leftarrow 0$
$error \leftarrow \infty$
while $error > \varepsilon$ **do**
  $w^* \leftarrow update\_w(w^*)$
  $error \leftarrow get\_error(f, w^*)$
end while
return w^*

Zauważmy, że jeżeli w_n^* spełnia (3) to dla dowolnego $k > n$ w_k^* również spełnia warunek (3).
Dostępna jest więc optymalizacja powyższego algorytmu polegająca na używania wyszukiwania binarnego do wyznaczenia najmniejszego n zamiast iteracji. My jednak użyjemy zwykłej iteracji z dwóch powodów: stać nas na to obliczeniowo, iteracyjnie łatwiejsze jest wyznaczanie kolejnych wielomianów w_n^*.

Używając wzoru (4) wyznaczanie w^* kolejnego stopnia (funkcja $udpate\_w(w^*)$) sprowadza się do:

$$w_{n+1}^*(x) = \sum_{i=0}^{n+1} \frac{\langle f, P_i \rangle}{\langle P_i, P_i \rangle} P_i(x) = \sum_{i=0}^{n} \frac{\langle f, P_i \rangle}{\langle P_i, P_i \rangle} P_i(x) + \frac{\langle f, P_{n+1} \rangle}{\langle P_{n+1}, P_{n+1} \rangle} P_{n+1}(x) =$$
$$= w_n^*(x) + \frac{\langle f, P_{n+1} \rangle}{\langle P_{n+1}, P_{n+1} \rangle} P_{n+1}(x)$$

Obliczanie nowego błędu (funkcja $get\_error(f, w^*)$) to aplikacja wzoru (5) do nowego wielomianu w_n^*.

Mamy już więc szkic rozwiązania, pozostało jeszcze kilka podproblemów.

Wybór $\{P_k\}$

Z twiedzenia Farvarda wiemy, że dowolny ciąg wielomianów spełniający poniższą zależność rekurencyjną jest ciągiem wielomianów ortogonalnych:

$$P_k(x) = (\alpha_k x - \beta_k) P_{k-1}(x) - \gamma_k P_{k-2} \tag{6}$$

Zatem wybór $\{P_k\}$ sprowadza się do wyznaczenia współczynników: $\alpha_k, \beta_k, \gamma_k$.
Patrząc na wzór (4), interesuje nas wyliczanie pewnej kombinacji liniowej wielomianów P_k.

Ten problem tj. obliczanie $s_n(x) = a_0P_0(x) + a_1P_1(x) + \ldots + a_nP_n(x)$ możemy rozwiązać stosując algorytm Clenshaw'a, polega on na:

Obliczamy pomocnicze wartości V_k:

$$V_k = \begin{cases} a_k + (\alpha_{k+1}x - \beta_{k+1})V_{k+1} - \gamma_{k+2}V_{k+2} & \text{gdy } k \leqslant n \\ 0 & \text{gdy } k > n \end{cases} \tag{7}$$

wtedy:

$$s_n(x) = \alpha_0 V_0 \tag{8}$$

Algorytm Clenshew'a jest mocnym ułatwieniem obliczeniowym dla skomplikowanych funkcji P_k, ale za to dla łatwych do wyznaczenia współczynników $\alpha_k, \beta_k, \gamma_k$.

Dla ułatwienia obliczeń użyjemy ciągu wielomianów Czebyszewa z rekurencyjną definicją:

$$T_k = \begin{cases} 2xT_{k-1}(x) - T_{k-2}(x) & \text{gdy } k > 1 \\ x & \text{gdy } k = 1 \\ 1 & \text{gdy } k = 0 \end{cases} \tag{9}$$

Czyli współczynników:

$$\alpha_k = 2, \beta_k = 0, \gamma_k = 1 \tag{10}$$

Obliczanie $\langle f, g \rangle$

Implementacja

Wyniki i interpretacja

Podsumowanie