KSN — III FK — teoria 3
Interpolacja wielomianowa

Zajmujemy się interpolacją wielomianową, tj. szukaniem wielomianu $w(x)$ o tej własności, że $w(x_i) = y_i$, gdzie dane N punktów (x_i, y_i) nazywamy węzłami interpolacyjnymi.

Łatwo sprawdzić, że wielomian stopnia $N-1$ dany wzorem *Lagrange'a*

$$w(x) = \sum_{i=1}^{N} \Big(y_i \cdot \prod_{\substack{j=1 \\ j \neq i}}^{N} \frac{x - x_j}{x_i - x_j} \Big)$$

jest rozwiązaniem problemu interpolacyjnego.

Przyglądając się macierzy Vandermonde'a z zadania 2.1 łatwo zauważyć, że jeśli tylko wszystkie odcięte węzłów są różne, to istnieje dokładnie jeden wielomian interpolacyjny stopnia $N-1$ spełniający $w(x_i) = y_i$ dla $i = 1, \ldots, N$ — wynika to wprost z twierdzeniem Cramera.

Wyznaczenie wartości $w(x)$ bezpośrednio ze wzoru Lagrange'a nie nosi znamion przestępstwa, na ogół jednak korzystamy w tym celu ze schematu *Neville'a*:

- Niech $P_i(x)$ będzie wartością wielomianu stopnia zerowego (stałą) przechodzącego przez (x_i, y_i) dla pewnego x (a więc $P_i(x) \equiv y_i$),
- $P_{i,i+1}(x)$ będzie wartością wielomianu stopnia pierwszego (funkcji liniowej) przechodzącej przez parę punktów $\{(x_i, y_i), (x_{i+1}, y_{i+1})\}$ dla pewnego x,
- $P_{i,i+1,i+2}(x)$ będzie wartością wielomianu stopnia drugiego (trójmianu kwadratowego) przechodzącego przez punkty $\{(x_i, y_i), (x_{i+1}, y_{i+1}), (x_{i+2}, y_{i+2})\}$ dla pewnego x,
- itd., aż do $P_{1,2,\ldots,N}(x)$ będącego wartością dla x szukanego wielomianu interpolacyjnego przechodzącego przez wszystkie zadane węzły.

Wówczas algorytm Neville'a pozwala na konstrukcję wielomianu interpolacyjnego $w(x)$ poprzez iteracyjne wyznaczanie dla każdego i kolejnych P_i, $P_{i,i+1}$, $P_{i,i+1,i+2}$, $\ldots$:

$$\begin{array}{lllll}
x_1: & y_1 = P_1 & & & \\
 & & P_{12} & & \\
x_2: & y_2 = P_2 & & P_{123} & \\
 & & P_{23} & & P_{1234} \\
x_3: & y_3 = P_3 & & P_{234} & \\
 & & P_{34} & & \\
x_4: & y_4 = P_4 & & &
\end{array}$$

zgodnie z formułą

$$P_{i,i+1,\ldots,(i+m)} = \frac{(x - x_{i+m}) P_{i,i+1,\ldots,i+m-1} + (x_i - x) P_{i+1,i+2,\ldots,i+m}}{x_i - x_{i+m}}.$$

W bibliotece *Numerical Recipes* schemat Neville'a wyznaczania $w(x)$ realizuje procedura `polint`. W procedurze

```
SUBROUTINE polint(xa,ya,n,x,y,dy)
INTEGER n,NMAX
REAL dy,x,y,xa(n),ya(n)
PARAMETER (NMAX=10)
...
```

wektory `xa` i `ya` służą do przetrzymywania węzłów interpolacyjnych (x_i, y_i) zaś n informuje o ich liczbie. W wyniku wywołania procedury `polint` wartość wielomianu interpolacyjnego dla `x` zostaje wpisana do `y`.

Biblioteka *GNU Scientific Library (GSL)* zawiera funkcję `gsl_interp_eval` zwracającą wartość wielomianu interpolacyjnego w punkcie `x`. Funkcja zadeklarowana jest w pliku nagłówkowym `gsl_interp.h`.

```
double gsl_interp_eval(const gsl_interp * interp, const double xa[],
                       const double ya[], double x, gsl_interp_accel * acc)
```

Pozostałe (oprócz `x`) parametry tej funkcji to:

- `xa, ya` – tablice zawierające węzły interpolacyjne,
- `interp` – wskaźnik do obiektu interpolacyjnego zaalokowanego przez funkcję
  `gsl_interp * gsl_interp_alloc (const gsl_interp_type * T, size_t size)`,
  w której jako typ interpolacji `T` podano `gsl_interp_polynomial` (`size` jest rozmiarem tablic `xa` i `ya`),
  np. `gsl_interp_alloc (gsl_interp_polynomial, N);`
  Obiekt musi zostać też wcześniej wypełniony danymi za pomocą funkcji
  `int gsl_interp_init(gsl_interp * interp, const double xa[], const double ya[], size_t size)`
  przy czym tablica `xa` musi być posortowana – w przeciwnym wypadku wynik jest nieokreślony.
- `acc` – wskaźnik do obiektu zaalokowanego funkcją `gsl_interp_accel * gsl_interp_accel_alloc (void)`.

Funkcje tworzące pomocnicze obiekty wywołujemy tylko raz przed późniejszym (nawet wielokrotnym) użyciem fukcji `gsl_interp_eval`.

Po zakończeniu obliczeń dobrze jest zwolnić zarezerwowaną wcześniej pamięć – do tego celu przeznaczone są funkcje `gsl_interp_free` oraz `gsl_interp_accel_free`.

Krzysztof Malarz & Maciej Wołoszyn, Kraków, 5 listopada 2003