

1. Znaleźć wielomian interpolacyjny dla podanych punktów:

x	0	1	3	5
y	-4	0	38	196

- a) z wzoru interpolacyjnego Newtona
- b) z wzoru interpolacyjnego Lagrange'a
- c) z układu równań liniowych z macierzą Vandermonde'a

Rozwiązanie.

a) Wielomian interpolacyjny ma postać

$$p(x) = \sum_{k=0}^n f[x_0, x_1, \dots, x_k] \prod_{j=0}^{k-1} (x - x_j)$$

Tworzymy więc tabelę ilorazów różnicowych

x_0	$f[x_0]$			
		$f[x_0, x_1]$		
x_1	$f[x_1]$		$f[x_0, x_1, x_2]$	
		$f[x_1, x_2]$		$f[x_0, x_1, x_2, x_3]$
x_2	$f[x_2]$		$f[x_1, x_2, x_3]$	
		$f[x_2, x_3]$		
x_3	$f[x_3]$			

W naszym przypadku ma ona postać

0	-4			
		4		
1	0		5	
		19		2
3	38		15	
		79		
5	196			

Współczynniki szukanego wielomianu to $f[x_0]$, $f[x_0, x_1]$, $f[x_0, x_1, x_2]$, $f[x_0, x_1, x_2, x_3]$.
Ma on zatem postać:

$$p(x) = -4 + 4x + 5x(x-1) + 2x(x-1)(x-3) = 2x^3 - 3x^2 + 5x - 4$$

b) Wielomian interpolacyjny ma postać:

$$p(x) = \sum_{i=0}^n y_i l_i(x),$$

gdzie

$$l_i(x) = \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j}.$$

U nas

$$l_0(x) = \frac{(x-1)(x-3)(x-5)}{(0-1)(0-3)(0-5)} = -\frac{1}{15}(x-1)(x-3)(x-5)$$

$$l_1(x) = \frac{x(x-3)(x-5)}{(1-0)(1-3)(1-5)} = \frac{1}{8}x(x-3)(x-5)$$

$$l_2(x) = \frac{x(x-1)(x-5)}{(3-0)(3-1)(3-5)} = -\frac{1}{12}x(x-1)(x-5)$$

$$l_3(x) = \frac{x(x-1)(x-3)}{(5-0)(5-1)(5-3)} = \frac{1}{40}x(x-1)(x-3)$$

Ostatecznie mamy

$$\begin{aligned} p(x) &= -4l_0(x) + 0l_1(x) + 38l_2(x) + 196l_3(x) = \\ &= \frac{4}{15}(x-1)(x-3)(x-5) - \frac{38}{12}x(x-1)(x-5) + \frac{196}{40}x(x-1)(x-3) = \\ &= \frac{16}{60}(x^3 - 9x^2 + 23x - 15) - \frac{190}{60}(x^3 - 6x^2 + 5x) + \frac{294}{60}(x^3 - 4x^2 + 3x) = \\ &= \left(\frac{15-190+294}{60}\right)x^3 + \left(\frac{-16\cdot 9+190\cdot 6-294\cdot 4}{60}\right)x^2 + \left(\frac{16\cdot 23-5\cdot 190+3\cdot 294}{60}\right)x - \frac{15\cdot 16}{60} = \\ &= 2x^3 - 3x^2 + 5x - 4 \end{aligned}$$

c) Układ równań ma postać

$$\begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 1 & x_1 & x_1^2 & x_1^3 \\ 1 & x_2 & x_2^2 & x_2^3 \\ 1 & x_3 & x_3^2 & x_3^3 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

czyli w naszym przypadku

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 3 & 9 & 27 \\ 1 & 5 & 25 & 125 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} -4 \\ 0 \\ 38 \\ 196 \end{bmatrix}$$

Rozwiązując go (np. programem `ulin.x` z zestawu 2, albo poleceniem `solve` w R) dostajemy

$$a_0 = -4, a_1 = 5, a_2 = -3, a_3 = 2$$

Wszystkimi trzema sposobami dostaliśmy ten sam wielomian (bo jest określony jednoznacznie)

Możemy ten wynik sprawdzić w pakiecie R:

```
library(polynom)
x=c(0,1,3,5)
y=c(-4,0,38,196)
poly.calc(x,y)
```

otrzymując

$$-4 + 5x - 3x^2 + 2x^3$$

czyli nasz wielomian.

Funkcja `poly.calc` zwraca wielomian interpolacyjny dla wartości y_i w węzłach x_i .

2. Stosując wzór interpolacyjny Newtona dla równoodległych punktów znajdź wartość $\text{tg}(1.134)$ na podstawie tabeli wartości funkcji tg w przedziale $[1.1, 1.16]$ z krokiem 0.01 .

x	$\text{tg}(x)$
1.10	1.96476
1.11	2.01434
1.12	2.06596
1.13	2.11975
1.14	2.17588
1.15	2.23450
1.16	2.29580

Rozwiązanie.

Budujemy tabelę różnic (dla przejrzystości różnice pomnożone zostały przez 100000)

x	$f(x)$	Δf	$\Delta^2 f$	$\Delta^3 f$	$\Delta^4 f$	$\Delta^5 f$	$\Delta^6 f$
1.10	1.96476						
		4958					
1.11	2.01434		204				
		5162		13			
1.12	2.06596		217		4		
		5379		17		-6	
1.13	2.11975		234		-2		12
		5613		15		6	
1.14	2.17588		249		4		
		5862		19			
1.15	2.23450		268				
		6130					
1.16	2.29580						

Krok wynosi $h = 0.01$

$x_0 = 1.13$

idąc od $f_0 = 2.11975$ po przekątnej w dół mamy kolejno (pamiętając o czynniku 100000)

$$\Delta f_0 = 0.005613, \Delta^2 f_0 = 0.000249, \Delta^3 f_0 = 0.000019$$

Z wzoru Newtona mamy

$$f(x_0 + ph) \approx f_0 + p\Delta f_0 + \frac{1}{2}p(p-1)\Delta^2 f_0 + \frac{1}{6}p(p-1)(p-2)\Delta^3 f_0$$

U nas $p = 0.4$, więc dostajemy

$$f(1.134) \approx 2.11975 + 0.4 \cdot 0.005613 + \frac{1}{2} \cdot 0.4 \cdot (-0.6) \cdot 0.000249 + \frac{1}{6} \cdot 0.4 \cdot (-0.6) \cdot (-1.6) \cdot 0.000019 = 2.14192$$

Wartość dokładna do 6 cyfr znaczących to 2.14191.

Używając interpolacji liniowej (czyli jedynie członów $f_0 + p\Delta f_0$) dostalibyśmy 2.14221.

3. (Zjawisko Rungego) Używając pakietu R obejrzyj dopasowanie wielomianu do funkcji

$$\frac{1}{1+25x^2}$$

Użyj wzrastającej liczby węzłów (3, 5, 11, 21)

a) równoodległych

b) węzłów Czebyszewa zadanych wzorem

$$x_j = \cos\left(\frac{2j-1}{2k}\pi\right),$$

gdzie k to liczba węzłów.

Rozwiązanie.

a) Ładujemy potrzebny pakiet, definiujemy funkcję i rysujemy ją:

```
library(polynom)
f=function(x) 1/(1+25*x^2)
curve(f,xlim=c(-1,1),ylim=c(-1,1))
```

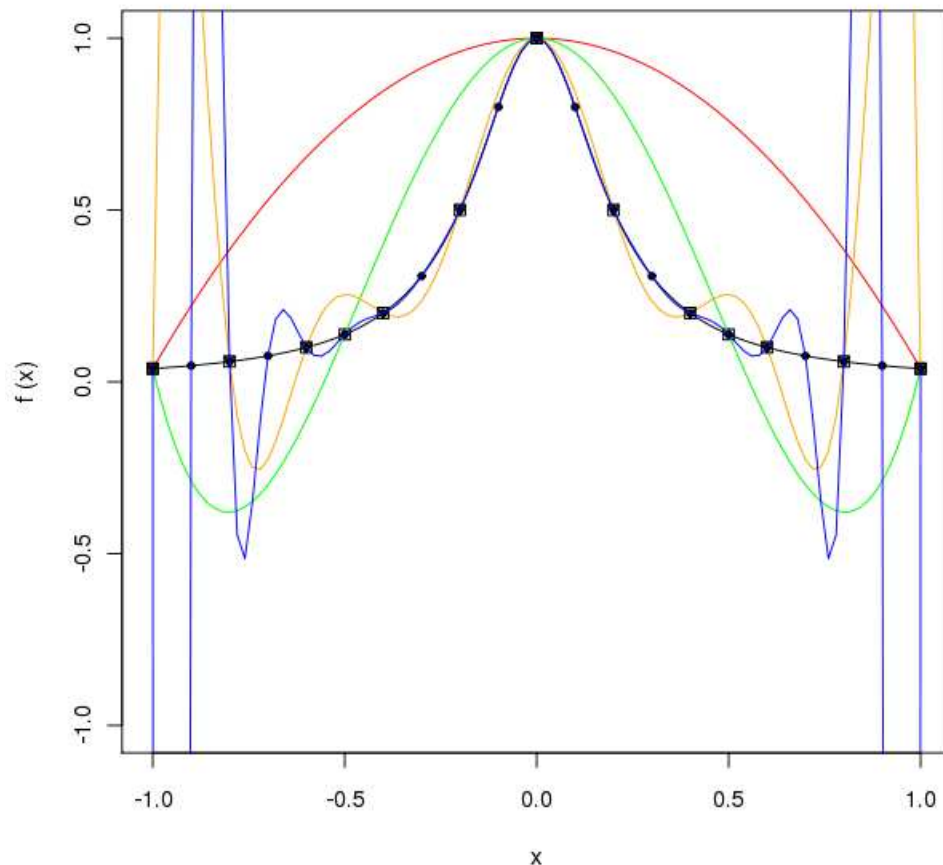
następnie generujemy wartości funkcji w 3 punktach, nanosimy je na wykres, dopasowujemy wielomian i rysujemy go:

```
x=seq(-1,1,by=1)
y=f(x)
points(x,y)
f1=as.function(poly.calc(x,y))
curve(f1,add=T,col="red")
```

następnie operacje powtarzamy dla 5, 11 i 21 punktów:

```
x=seq(-1,1,by=0.5)
y=f(x)
points(x,y,pch=12)
f1=as.function(poly.calc(x,y))
curve(f1,add=T,col="green")
x=seq(-1,1,by=0.2)
y=f(x)
points(x,y,pch=14)
f1=as.function(poly.calc(x,y))
curve(f1,add=T,col="orange")
x=seq(-1,1,by=0.1)
y=f(x)
points(x,y,pch=20)
f1=as.function(poly.calc(x,y))
curve(f1,add=T,col="blue")
```

W efekcie dostajemy następujący wykres:



Jak widać, dla coraz większej liczby węzłów wielomian interpolacyjny dobrze przybliża funkcję w środku przedziału, ale wykazuje coraz większe oscylacje na jego końcach. Jest to przykład zjawiska Rungego.

b) podobnie jak poprzednio rysujemy funkcję:

```
f=function(x) 1/(1+25*x^2)
curve(f,xlim=c(-1,1),ylim=c(-1,1))
```

definiujemy 3 węzły i nanosimy je na wykres

```
ii=seq(1:3)
x=cos((2*ii-1)/(2*3)*pi)
y=f(x)
points(x,y)
```

wyznaczamy i rysujemy wielomian interpolacyjny

```
f1=as.function(poly.calc(x,y))
curve(f1,add=T,col="red")
```

i powtarzamy operacje dla 5, 11 i 21 węzłów

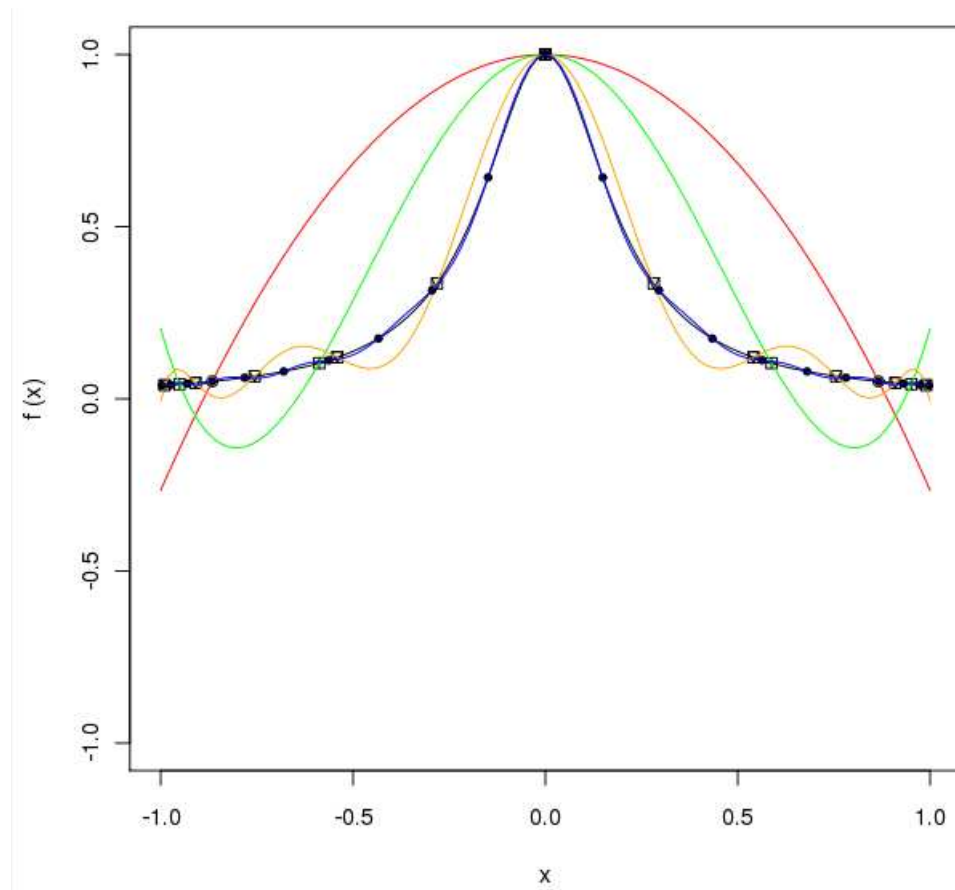
```
ii=seq(1:5)
x=cos((2*ii-1)/(2*5)*pi)
```

```

y=f(x)
points(x,y,pch=12)
f1=as.function(poly.calc(x,y))
curve(f1,add=T,col="green")
ii=seq(1:11)
x=cos((2*ii-1)/(2*11)*pi)
y=f(x)
points(x,y,pch=14)
f1=as.function(poly.calc(x,y))
curve(f1,add=T,col="orange")
ii=seq(1:21)
x=cos((2*ii-1)/(2*21)*pi)
y=f(x)
points(x,y,pch=20)
f1=as.function(poly.calc(x,y))
curve(f1,add=T,col="blue")

```

Tym razem dostajemy wykres



Dla większej liczby węzłów wielomian w środku przedziału coraz lepiej przybliża interpolowaną funkcję a oscylacje na końcach przedziału nie rosną tak gwałtownie, jak przy węzłach równoodległych.

4. Powtórz interpolację funkcji z zadania 3 dla 21 równoodległych punktów wykorzystując funkcje sklejane (sześciennie).

Rozwiązanie:

Odpowiednie polecenia w R to:

(definicja i rysowanie funkcji):

```
f=function(x) 1/(1+25*x^2)  
curve(f,xlim=c(-1,1))
```

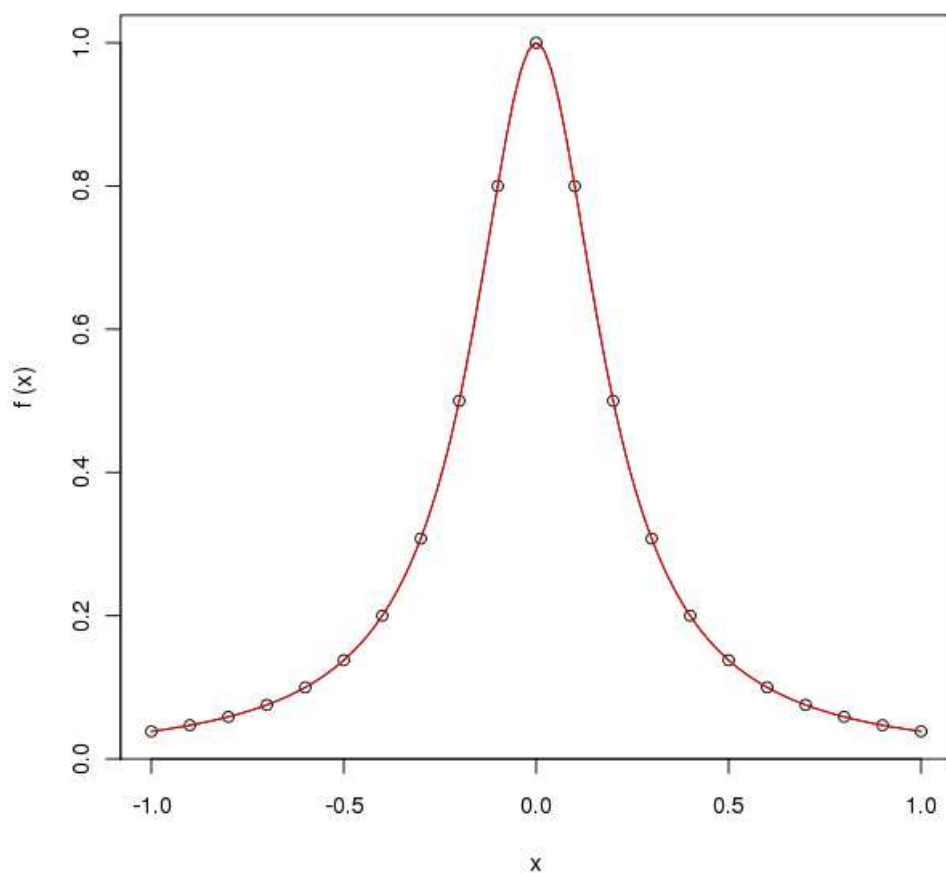
(obliczenie węzłów interpolacji i naniesienie ich na wykres)

```
x=seq(-1,1,by=0.1)  
y=f(x)  
points(x,y)
```

(interpolacja naturalnymi spline'ami w 200 punktach i narysowanie funkcji interpolacyjnej)

```
z=spline(x,y,method="natural",n=200)  
lines(z,col="red")
```

W wyniku otrzymujemy wykres:



Krzywa interpolacyjna jest praktycznie nieodróżnialna od interpolowanej funkcji.

5. (Iteracja odwrotna) Mamy tabelkę wartości pewnej funkcji

x	0	1	2	3
$f(x)$	0	1	4	9

Rozwiąż równanie $f(x) = 3$

a) stosując wzór interpolacyjny Newtona dla równoodległych punktów

b) wyznaczając postać funkcji $f^{-1}(y)$

Rozwiązanie.

a) Tworzymy tabelę różnic

x	$f(x)$	Δ	Δ^2	Δ^3
0	0			
		1		
$x_0 = 1$	1		2	
		3		0
2	4		2	
		5		
3	9			

Wzór interpolacyjny Newtona dla węzłów równoodległych ma postać

$$f(x_0 + ph) = f_0 + \sum_{j=1}^m \Delta^j f_0 \frac{p(p-1)\dots(p-j+1)}{j!}$$

U nas mamy

$h = 1$, $f(x_0 + ph) = 3 = c$, musimy znaleźć p .

Czyli

$$c = f_0 + \Delta f_0 p + \frac{1}{2} \Delta^2 f_0 p(p-1)$$

Stąd

$$\Delta f_0 p = c - f_0 - \frac{1}{2} \Delta^2 f_0 p(p-1)$$

$$p = \frac{1}{\Delta f_0} \left(c - f_0 - \frac{1}{2} \Delta^2 f_0 p(p-1) \right)$$

Inaczej

$$p = \varphi(p),$$

gdzie $\varphi(p)$ jest określone przez prawą stronę równości wyżej.

Wyznaczamy

$$p_0 = \frac{c - f_0}{\Delta f_0}$$

i dalej iteracyjnie $p_{i+1} = \varphi(p_i)$

U nas $f_0 = 1$, $\Delta f_0 = 3$, $\Delta^2 f_0 = 2$, skąd

$$p_0 = \frac{3-1}{3} = \frac{2}{3}$$

oraz

$$\varphi(p) = \frac{1}{3}(3 - 1 - \frac{1}{2} \cdot 2p(p-1)) = \frac{2}{3} - \frac{1}{3}p(p-1)$$

Iterując dostajemy więc

$$p_0 = 0.666667$$

$$p_1 = 0.740741$$

$$p_2 = 0.730681$$

$$p_3 = 0.732262$$

$$p_4 = 0.732018$$

$$p_5 = 0.732056$$

$$p_6 = 0.732050$$

$$p_7 = 0.732051$$

$$p_8 = 0.732051$$

Zatem $p = 0.732051$ i szukana wartość $x = x_0 + ph = 1 + 0.732051 = 1.732051$

b) Próbuje znaleźć wielomian interpolacyjny dla funkcji $f^{-1}(y) = g(y)$.

Tworzymy tabelę ilorazów różnicowych

y	$g(y)$			
0	0			
		1		
1	1		-1/6	
		1/3		1/60
4	2		-1/60	
		1/5		
9	3			

Wielomian interpolacyjny ma więc postać

$$\begin{aligned} g(y) &\approx 0 + 1 \cdot (y-0) - \frac{1}{6}(y-0)(y-1) + \frac{1}{60}(y-0)((y-1)(y-4)) = \\ &= y - \frac{1}{6}y(y-1) + \frac{1}{60}y(y-1)(y-4) \end{aligned}$$

Możemy więc obliczyć, że

$$g(3) = 3 - 1 - 1/10 = 1.90$$

Nietrudno zauważyć, że naszą funkcją f była funkcja $f(x) = x^2$, zatem szukana wartość to $\sqrt{3} \approx 1.732051$. Pierwsza metoda dała bardzo dobry wynik, druga mniej dokładny. Jest tak, ponieważ próbowaliśmy interpolować wielomianem funkcję $g(y) = \sqrt{y}$, która niezbyt dobrze nadaje się do interpolacji wielomianami w badanym zakresie. Z drugiej strony $f(x) = x^2$, więc wzór interpolacyjny Newtona działał bardzo dobrze.

6. Przy pomocy pakietu R obejrzyj wykresy wielomianów ortogonalnych

a) Czebyszewa

b) Lagrange'a

do stopnia 5-tego.

Rozwiązanie.

a) W pakiecie R ładujemy bibliotekę orthopolynom

```
library(orthopolynom)
```

a następnie wywołujemy funkcję generującą wielomiany Czebyszewa

```
(wiel.C=chebyshev.t.polynomials(5))
```

Dostajemy postać wielomianów $T_0, \dots, T_5$.

Możemy przekształcić T_0 na funkcję, którą da się wyrysować i jej obejrzyć wykres

```
wiel=as.function(wiel.C[[1]])
```

```
curve(wiel,xlim=c(-1,1),ylim=c(-1,1.1),lwd=2,col=1)
```

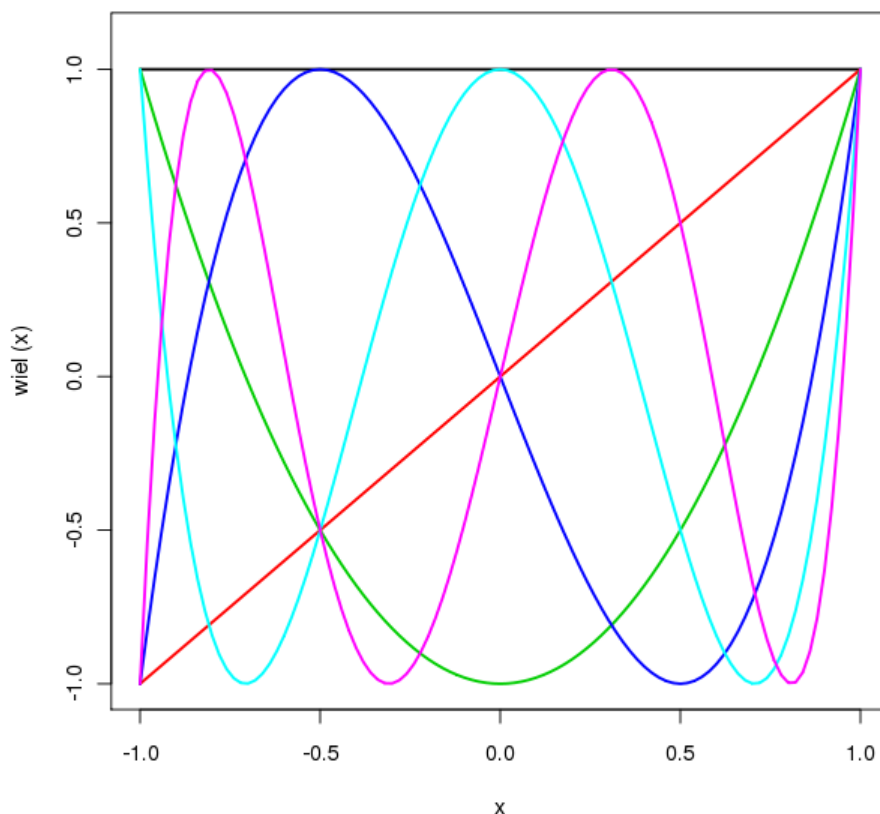
Analogicznie rysujemy pozostałe wielomiany

```
for (i in 2:6) {
```

```
wiel=as.function(wiel.C[[i]])
```

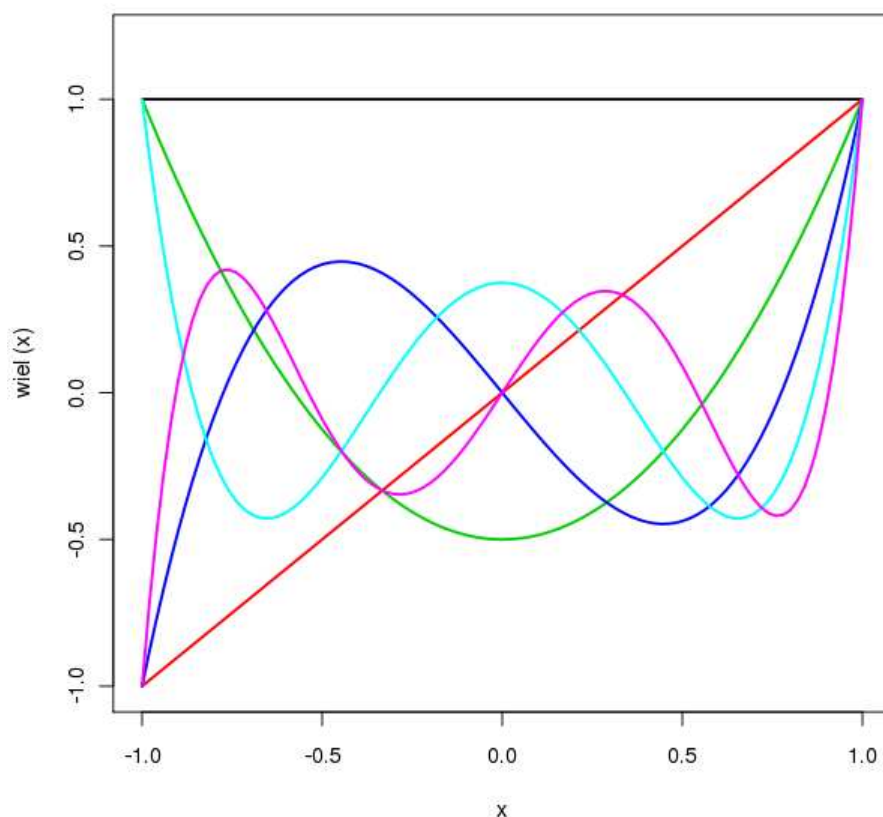
```
curve(wiel,add=T,lwd=2,col=i)}
```

otrzymując ostatecznie wykres



b) W podobny sposób otrzymujemy współczynniki i wykres wielomianów Legendre'a:

```
(wiel.L=legendre.polynomials(5))  
wiel=as.function(wiel.L[[1]])  
curve(wiel,xlim=c(-1,1),ylim=c(-1,1.2),lwd=2,col=1)  
for (i in 2:6) {  
  wiel=as.function(wiel.L[[i]])  
  curve(wiel,add=T,lwd=2,col=i)}
```



7. Obliczyć numerycznie pierwszą pochodną funkcji $\sin(x)$ dla $x = 0.5$. Użyć kroku $h = 0.04$, następnie zmniejszyć krok dwukrotnie i wyznaczyć pochodną z pierwszego kroku ekstrapolacji Richardsona.

Rozwiązanie

Dokładną wartością pochodnej jest oczywiście $\cos(0.5) \approx 0.877583$

Sprawdźmy najpierw wynik obliczenia ze wzoru

$$f'(x) \approx (f(x+h) - f(x))/h = (\sin(0.54) - \sin(0.5))/0.04 = 0.867761$$

Zastosujmy teraz wzór

$$f'(x) \approx (f(x+h) - f(x-h))/2h = (\sin(0.54) - \sin(0.46))/0.08 = 0.877349$$

Przybliżenie jest znacznie lepsze.

Zmniejszamy teraz krok dwukrotnie do 0.02:

$$f'(x) \approx (\sin(0.52) - \sin(0.48))/0.04 = 0.877524$$

Przy zmniejszeniu kroku dokładność polepszyła się.

Oznaczając oszacowania pochodnej otrzymane dla kroku h i $h/2$ odpowiednio jako $\varphi(h)$ i $\varphi(h/2)$ dostaniemy ekstrapolowaną wartość jako

$$f'(x) \approx 4/3 \varphi(h/2) - 1/3 \varphi(h) = 4/3 \cdot 0.877524 - 1/3 \cdot 0.877349 = 0.877582$$

Różnica do wartości dokładnej jest już bardzo mała (i wynika w znacznej części z zaokrągleń na pośrednich etapach).

8. Bazując na wynikach obliczeń kwantowochemicznych wyznacz metodą skończonego pola

a) moment dipolowy cząsteczki HCl

b) polaryzowalność cząsteczki CH₄

Rozwiązanie.

a) Zgodnie z rozważaniami z wykładu składowe momentu dipolowego możemy wyznaczyć jako pierwsze pochodne energii cząsteczki po składowych pola elektrycznego.

Ponieważ cząsteczka jest dwuatomowa, musimy wyznaczyć tylko jedną składową momentu – wzdłuż osi cząsteczki.

Poniżej zestawiono wyniki obliczeń programem Gaussian 09, metodologia DFT z funkcjonalem B3LYP, baza 6-31+G**. Zebrano energie cząsteczki obliczone dla różnych wartości pola przykładanego wzdłuż osi wiązania (wartości w jednostkach atomowych).

F	E
0,0004	-460,803438316
0,0002	-460,803322053
0,0000	-460,803206327
-0,0002	-460,803091140
-0,0004	-460,802976491

Wykorzystując wzór

$$f'(x) \approx \frac{1}{2h} [f(x+h) - f(x-h)]$$

Dostajemy dla $h = 0.0004$ $f' \approx -0,577281250$

dla $h = 0.0002$ $f' \approx -0,577282500$

Z ekstrapolacji Richardsona

$$f' \approx 4/3 f'(h/2) - 1/3 f'(h) = -0,577282917$$

Uwzględniając, że atomowa jednostka momentu dipolowego to 2,541746313 D dostajemy wartość momentu dipolowego $\mu_{\text{HCl}} = 1,4673067$ D

Jest to wynik zgodny z wartością 1.4673 D obliczoną przez program na podstawie rozkładu gęstości elektronowej.

b) Stosując metodę skończonego pola obliczamy składowe tensora polaryzowalności jako odpowiednie drugie pochodne cząstkowe energii po składowych pola. W naszym przypadku ze względu na symetrię cząsteczki jej polaryzowalność jest izotropowa i wystarczy po przyłożeniu pola w dowolnym kierunku wyznaczyć drugą pochodną energii.

Programem Gaussian 09, metoda B3LYP, baza 6-31+G** uzyskano przykładowe wyniki

F	E
0,0004	-40,5261452957
0,0002	-40,5261444732
0,0000	-40,5261441990
-0,0002	-40,5261444732
-0,0004	-40,5261452957

Ze wzoru

$$f''(x) \approx \frac{1}{h^2} [f(x-h) - 2f(x) + f(x+h)]$$

dostajemy dla $h = 0.0004$ $f'' = -13,7087500$

dla $h = 0.0002$ $f'' = -13,7099999$

Ponieważ polaryzowalność ma przeciwny znak niż pochodna, z drugiego wyniku mamy

$$\alpha = 13.710 \text{ a.u.}$$

Wynik metody skończonego pola jest tu w idealnej zgodności z wartością 13.710 a.u. obliczoną przez program metodą CPHF.

9. Wyprowadź z reguły Newtona-Cotesa wzór Simpsona

$$\int_a^b f(x) dx \approx \frac{1}{6} (b-a) \left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right]$$

Rozwiązanie

Zacznijmy od wzoru dla przedziału $[0,1]$.

Szukamy współczynników trójpunktowej kwadratury

$$\int_0^1 f(x) dx \approx \sum_{i=0}^2 A_i f(x_i),$$

gdzie x_i są równe kolejno 0, 1/2 i 1.

Zastosujemy metodę współczynników nieoznaczonych, czyli będziemy szukać rozwiązania układu

$$\sum_{i=0}^2 A_i x_i^j = \int_0^1 x^j dx, \quad j = 0, 1, 2$$

Całki po stronie prawej są równe

$$\int_0^1 x^0 dx = 1, \quad \int_0^1 x^1 dx = \frac{1}{2}, \quad \int_0^1 x^2 dx = \frac{1}{3}$$

Wstawiając x_i do lewych stron dostajemy układ

$$A_0 + A_1 + A_2 = 1$$

$$\frac{1}{2} A_1 + A_2 = \frac{1}{2}$$

$$\frac{1}{4} A_1 + A_2 = \frac{1}{3}$$

Odejmując trzecie równanie od drugiego dostajemy $A_1 = \frac{2}{3}$, a potem łatwo $A_2 = \frac{1}{6}$ i $A_0 = \frac{1}{6}$.

Zatem otrzymaliśmy wzór

$$\int_0^1 f(x) dx \approx \frac{1}{6} f(0) + \frac{2}{3} f\left(\frac{1}{2}\right) + \frac{1}{6} f(1)$$

zapisywany zwykle w formie

$$\int_0^1 f(x) dx \approx \frac{1}{6} \left[f(0) + 4f\left(\frac{1}{2}\right) + f(1) \right].$$

Skorzystajmy teraz z zamiany przedziału całkowania w całe

$$\int_c^d f(t) d(t) \approx \sum_{i=0}^n A_i f(t_i)$$

na przedział $[a, b]$

$$\int_a^b f(x) dx \approx \frac{b-a}{d-c} \sum_{i=0}^n A_i f\left(\frac{(b-a)t_i + ad - bc}{d-c}\right).$$

Po podstawieniu otrzymujemy wzór Simpsona

$$\int_a^b f(x) dx \approx \frac{1}{6}(b-a) \left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right].$$

10. Wyprowadź wzory na dwu- i trójpunktową kwadraturę Gaussa dla całki

$$\int_{-1}^1 f(x) dx.$$

Sprawdź ich działanie dla całki

$$I = \int_{-1}^1 e^x dx$$

Porównaj z wynikiem z wzoru Simpsona z zadania poprzedniego i wartością dokładną.

Rozwiązanie.

Ponieważ funkcja wagowa $w(x) = 1$, właściwymi wielomianami ortogonalnymi do wyznaczenia punktów kwadratury w przedziale $[-1, 1]$ są wielomiany Legendre'a.

Węzły kwadratury dwupunktowej dostaniemy jako zera wielomianu $P_2(x) = \frac{1}{2}(3x^2 - 1)$.

Są to oczywiście $x_0 = -1/\sqrt{3}$ i $x_1 = 1/\sqrt{3}$.

Do obliczenia współczynników A_0 i A_1 wykorzystamy metodę współczynników nieoznaczonych. Potrzebne całki to

$$\int_{-1}^1 x^0 dx = 2 \quad \text{ i } \quad \int_{-1}^1 x^1 dx = 0$$

Wstawiając wartości x_0 i x_1 otrzymujemy równania na współczynniki

$$A_0 + A_1 = 2$$

$$-\frac{1}{\sqrt{3}} A_0 + \frac{1}{\sqrt{3}} A_1 = 0$$

skąd łatwo znajdujemy $A_0 = A_1 = 1$.

Odpowiednia kwadratura więc to

$$\int_{-1}^1 f(x) dx \approx f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right).$$

Dla wyznaczenia węzłów kwadratury trójpunktowej potrzebujemy zer wielomianu $P_3(x) = \frac{1}{2}(5x^3 - 3x)$.

Są to oczywiście $-\sqrt{3/5}$, 0 i $\sqrt{3/5}$.

Potrzebujemy jeszcze wartości całki

$$\int_{-1}^1 x^2 dx = \frac{2}{3}$$

i dostajemy równania na współczynniki

$$\begin{aligned}
A_0 + A_1 + A_2 &= 2 \\
-\sqrt{\frac{3}{5}}A_0 + \sqrt{\frac{3}{5}}A_2 &= 0 \\
\frac{3}{5}A_0 + \frac{3}{5}A_2 &= \frac{2}{3}
\end{aligned}$$

Skąd $A_0 = A_2 = 5/9$ i $A_1 = 8/9$

Dostaliśmy więc kwadraturę

$$\int_{-1}^1 f(x) dx \approx \frac{5}{9} f\left(-\sqrt{\frac{3}{5}}\right) + \frac{8}{9} f(0) + \frac{5}{9} f\left(\sqrt{\frac{3}{5}}\right).$$

Dla $f(x) = e^x$ z pierwszego wzoru (dwupunktowego) dostajemy

$$I \approx \exp(-1/\sqrt{3}) + \exp(1/\sqrt{3}) \approx 2.3426961$$

Z wzoru drugiego (trójpunktowego) otrzymujemy

$$I \approx \frac{5}{9} \exp(-\sqrt{0.6}) + \frac{8}{9} + \frac{5}{9} \exp(\sqrt{0.6}) \approx 2.3503369$$

Wzór Simpsona daje

$$I \approx \frac{2}{6} [\exp(-1) + 4 + \exp(1)] \approx 2.3620538$$

Wartość dokładna to oczywiście

$$I = \exp(1) - \exp(-1) \approx 2.3504024$$

Widać, że kwadratury Gaussa (nawet dwupunktowa) dały lepszy wynik.

Możemy jeszcze sprawdzić wyniki w pakiecie R

Definiujemy funkcję:

```
f=function(x) exp(x)
```

i możemy policzyć całkę

```
integrate(f, -1, 1)
```

otrzymując wartość 2.350402.

Możemy też sprawdzić nasze wzory na kwadratury Gaussa. Biblioteka `gaussquad` umożliwia pracę z kwadraturami Gaussa dla wielu rodzin wielomianów ortogonalnych.

```
library(gaussquad)
```

Wyznamy węzły kwadratur do trójpunktowej włącznie:

```
(rules=legendre.quadrature.rules(3))
```

```
[[1]]
```

```
  x w
1 0 2
```

```
[[2]]
```

```
      x w
1 0.5773503 1
2 -0.5773503 1
```

```
[[3]]
```

```
      x      w
1  7.745967e-01 0.5555556
2  7.771561e-16 0.8888889
3 -7.745967e-01 0.5555556
```

Jak widać, wyniki są zgodne z naszymi wyprowadzeniami (w oznacza współczynnik kwadratury dla węzła x).

Możemy teraz obliczyć naszą całkę wykorzystując najpierw kwadraturę dwu- a potem trójpunktową:

```
legendre.quadrature(f, rules[[2]], -1, 1)
[1] 2.342696
legendre.quadrature(f, rules[[3]], -1, 1)
[1] 2.350337
```

Całkę możemy obliczyć też w innym przedziale (funkcja odpowiednio przetransformuje wartości węzłów), np.:

```
legendre.quadrature(f, rules[[3]], -1, 2)
[1] 7.019261
```

Wynik dokładny to oczywiście:

```
f(2) - f(-1)
[1] 7.021177
```

11. Przy użyciu biblioteki `animation` w pakiecie R obejrzyj ideę działania całkowania Monte Carlo obliczając całkę

$$\int_0^1 x^2 dx$$

(Uwaga: funkcje całkowania MC z biblioteki `animation` służą jedynie do demonstracji działania metody dla funkcji jednej zmiennej i nie nadają się do żadnych prawdziwych obliczeń!)

Rozwiązanie.

W bibliotece `animation` dostępne są dwie funkcje obrazujące całkowania Monte Carlo. Pierwsza, `MC.hitormiss`, szacuje całkę na podstawie procentu losowo wybranych w pewnym prostokącie punktów, które trafiają pod wykres funkcji.

Druga, `MC.samplemean`, szacuje całkę uśredniając wartości funkcji podcałkowej w punktach wybieranych losowo w przedziale $[0,1]$.

Przed rozpoczęciem animacji warto skrócić odstęp między kolejnymi ramkami poleceniem `ani.options(interval=...)`

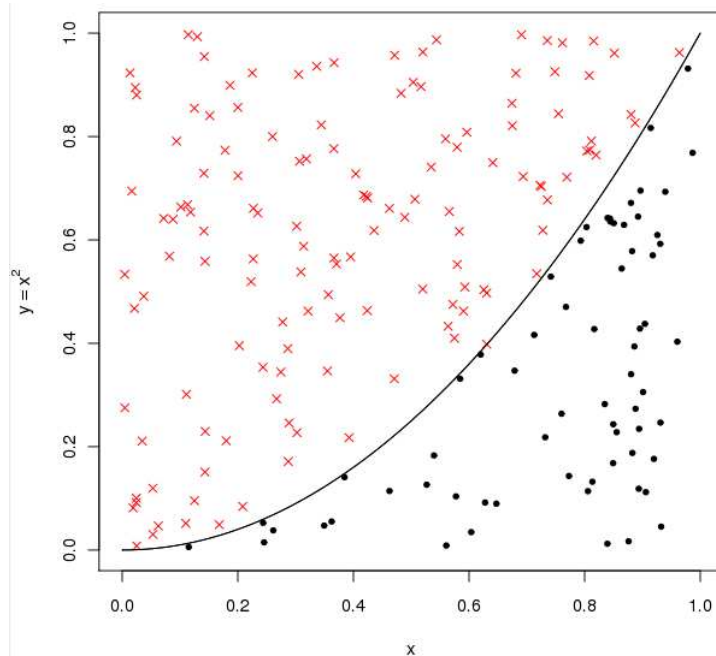
(podając w miejsce kropek dostatecznie małą wartość)

Ponieważ procedury MC używają liczb pseudolosowych, każde ich użycie powinno dać nieco inny wynik.

Najpierw należy załadować bibliotekę:
`library(animation)`

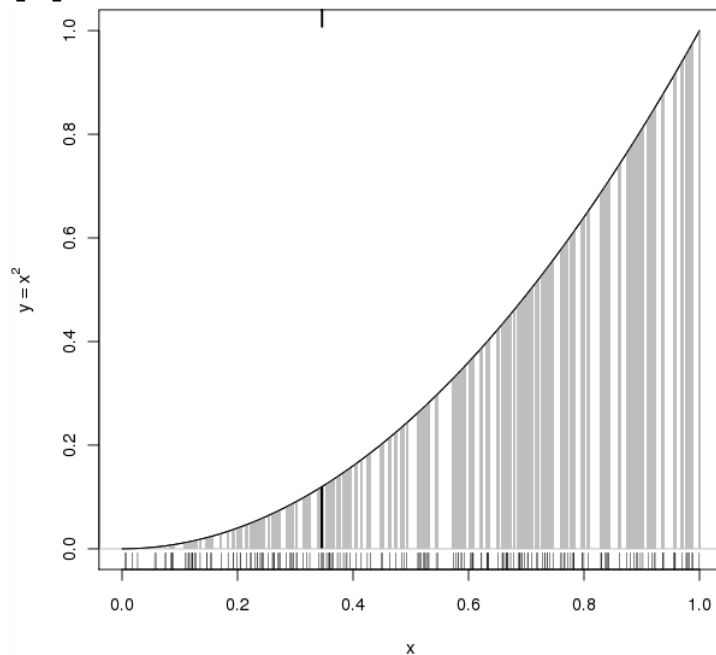
Użycie `MC.hitormiss` i przykładowy wynik:

```
MC.hitormiss(function(x) x^2, from=0, to=1, n=200)$est  
[1] 0.3299564
```



Użycie `MC.samplemean` i przykładowy wynik:

```
MC.samplemean(function(x) x^2, n=200, border=NA, adj.x=FALSE)$est  
[1] 0.3487592
```



Oba wyniki są dość odległe od dokładnej wartości całki równej $1/3$. Metody MC opłaca się stosować dopiero dla wielu wymiarów.