3 (done)

24 November, 2023 23:46

L7.3. **Włącz komputer!** 1 punkt Przy pomocy programu umożliwiającego rysowanie wykresów funkcji, przygotuj wykresy wielomianów

$$p_{n+1}(x) := (x - x_0)(x - x_1) \dots (x - x_n) \qquad (n = 4, 5, \dots, 20)$$

dla x_k $(0 \le k \le n)$ będących węzłami równoodległymi w przedziale $[-1, 1]$. Następnie powtórz eksperyment dla *węzłów Czebyszewa*. Skomentuj wyniki porównując odpowiednie wykresy. Jakie i dlaczego płyną stąd wnioski dla sposobu wyboru węzłów interpolacji?

Funkcje generujące wielomian oraz punkty Czebyszewa

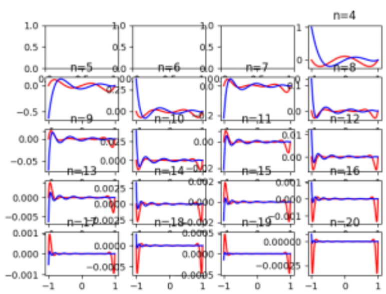
```python
# Funkcja zwracajaca wartość wielomianu
# Działa rekurencyjne - zwracamy wielomian dla pierwszego punktu z listy
# i rekurencyjnie wywolujemy dla reszty
def p(x, xs):
    if len(xs) == 0:
        return 1
    return (x - xs[0]) * p(x, xs[1:])


# Funkcja zwracajaca n punktów Czebyszewa w przedziale [-1, 1]
def chebyshev(n):
    chebyshev_points = []
    for k in range(n):
        cosval = (k * math.pi) / n
        chebyshev_point = math.cos(cosval)
        chebyshev_points.append(chebyshev_point)

    return chebyshev_points
```

Miejsca zerowe wielomianu Czebyszewa n-tego stopnia

Generowanie wykresu

```python
# Tworzymy wykresy - 5 wierszy, 4 kolumny
fig, ax = plt.subplots(5, 4)

# Oś x - 1000 punktów w przedziale [-1, 1]
x = np.linspace(-1, 1, 1000)

# Zmienne do generowania wykresów
 = 0 # Numer wiersza
change_i = 0 # Zmienna pomocnicza do zmiany kolumny

# Generujemy wykresy dla n = 4, 5, ..., 20
for i in range(4, 21):
    # Generujemy równoodległe węzły
    xs = np.linspace(-1, 1, i)

    # Wyliczamy i rysujemy wielomian p(x)
    y = [p(xn, xs) for xn in x] # Wyliczamy wartości wielomianu
    ax[j, i - change_i * 4 - 1].set_title(f'n={i}')
    ax[j, i - change_i * 4 - 1].plot(x, y, color="red")

    # Wyliczamy i rysujemy wielomian p(x) dla węzłów Czebyszewa
    cheb = [p(xn, chebyshev(i)) for xn in x]
    ax[j, i - change_i * 4 - 1].plot(x, cheb, color="blue")

    # Aktualizujemy zmienne
    if i > 0 and i % 4 == 0:
        j += 1
        change_i += 1

plt.show() # Wyświetlamy wykresy
```

Dla równoodległych



Niemiecki - Czebyszew
Czerwony - równoodległe węzły

Wnioski: Odpowiednie dobranie węzłów (czyt. Czebyszew) pozwala zminimalizować efekt Rungego (wartości odjeżdżające od rzeczywistości na krawędziach przedziałów)