

NAMA : Sherly Mawarni Kusumah

NIM : 1227030033

PRAKTIKUM FISIKA KOMPUTASI

“SUPPORT VECTOR MACHINE (SVM)”

- 1. Buat prediksi integral trapezoid menggunakan Support Vector Machine pada persamaan berikut ini:**

Fungsi pertama: $f(x) = 2x$

Fungsi kedua: $f(x) = 2x + 2$

Fungsi ketiga: $f(x) = 2x + 4$

Fungsi keempat: $f(x) = 4x + 6$

Fungsi kelima: $f(x) = 6x + 8$

Fungsi keenam: $f(x) = 8x + 10$

Fungsi ketujuh: $f(x) = 10x + 12$

Fungsi kedelapan: $f(x) = 12x + 14$

Fungsi kesembilan: $f(x) = 14x + 12$

Fungsi kesepuluh: $f(x) = 20x + 40$

JAWAB :

```
def Trapezoid(a,b,f):
    n = 100
    def trapezoid(f,a,b,n=100):
        h = (b-a)/n
        sum = 0.0
        for i in range(1,n):
            x = a+i*h
            sum = sum + f(x)
        integral = (h/2)*(f(a)+2*sum+f(b))
        return integral
    integral = trapezoid(f,a,b,n)
    print(a,"",b,"",round(integral,2))

# Melakukan looping untuk membuat database dari beberapa soal integral
for i in range(0,5):
    Trapezoid(i+1,i+2,lamba x: 2*x)
for i in range(0,5):
    Trapezoid(i+1,i+2,lamba x: 2*x + 2)
for i in range(0,5):
    Trapezoid(i+1,i+2,lamba x: 2*x + 4)
for i in range(0,5):
    Trapezoid(i+1,i+2,lamba x: 4*x + 6)
for i in range(0,5):
    Trapezoid(i+1,i+2,lamba x: 6*x + 8)
for i in range(0,5):
    Trapezoid(i+1,i+2,lamba x: 8*x + 10)
for i in range(0,5):
    Trapezoid(i+1,i+2,lamba x: 10*x + 12)
for i in range(0,5):
    Trapezoid(i+1,i+2,lamba x: 12*x + 14)
for i in range(0,5):
    Trapezoid(i+1,i+2,lamba x: 14*x + 12)
for i in range(0,5):
    Trapezoid(i+1,i+2,lamba x: 20*x + 40)
```

1, 2, 3.0

2, 3, 5.0

3, 4, 7.0

4, 5, 9.0

5, 6, 11.0
1, 2, 5.0
2, 3, 7.0
3, 4, 9.0
4, 5, 11.0
5, 6, 13.0
1, 2, 7.0
2, 3, 9.0
3, 4, 11.0
4, 5, 13.0
5, 6, 15.0
1, 2, 12.0
2, 3, 16.0
3, 4, 20.0
4, 5, 24.0
5, 6, 28.0
1, 2, 17.0
2, 3, 23.0
3, 4, 29.0
4, 5, 35.0
5, 6, 41.0
1, 2, 22.0
2, 3, 30.0
3, 4, 38.0
4, 5, 46.0
5, 6, 54.0
1, 2, 27.0
2, 3, 37.0
3, 4, 47.0
4, 5, 57.0
5, 6, 67.0
1, 2, 32.0
2, 3, 44.0
3, 4, 56.0
4, 5, 68.0
5, 6, 80.0
1, 2, 33.0
2, 3, 47.0
3, 4, 61.0
4, 5, 75.0
5, 6, 89.0
1, 2, 70.0
2, 3, 90.0
3, 4, 110.0
4, 5, 130.0
5, 6, 150.0

```

import numpy as np
import pandas as pd
from sklearn import svm
from google.colab import drive
import matplotlib.pyplot as plt

# Mount Google Drive
drive.mount('/content/drive')

# Path ke file Database.txt di google drive
file_path = '/content/drive/My Drive/MODUL 11 PRAKFIKOM/TUGAS TRAPEZOID.txt'

# Membaca data dari file
Database = pd.read_csv(file_path, sep=",", header=0)

# X = Data, y = Target
X = Database[['a', 'b']] # Pastikan kolom sesuai dengan nama yang ada di file
y = Database['Target']

# Membuat dan melatih model SVM
clf = svm.SVC()
clf.fit(X.values, y)

# Melakukan prediksi
y_pred = clf.predict(X.values)

# Menampilkan hasil prediksi
print("Hasil prediksi:")
for i, pred in enumerate(y_pred):
    print(f"X: {X.iloc[i, 0]}, {X.iloc[i, 1]} -> {pred}")

# Membuat plot perbandingan nilai asli dengan nilai prediksi
plt.figure(figsize=(10, 6))
plt.plot(range(len(y)), y, 'o-', label='Nilai Asli (Target)', color='blue')
plt.plot(range(len(y_pred)), y_pred, 'x--', label='Nilai Prediksi (SVM)', color='yellow')

# Menambahkan label dan judul
plt.xlabel('Indeks Data')
plt.ylabel('Nilai')
plt.title('Perbandingan Nilai Asli vs Nilai Prediksi Menggunakan SVM')
plt.legend()
plt.grid()
plt.show()

```

Mounted at /content/drive

Hasil prediksi:

X: 1, 2 -> 7.0
 X: 2, 3 -> 7.0
 X: 3, 4 -> 9.0
 X: 4, 5 -> 11.0
 X: 5, 6 -> 11.0
 X: 1, 2 -> 7.0
 X: 2, 3 -> 7.0
 X: 3, 4 -> 9.0
 X: 4, 5 -> 11.0
 X: 5, 6 -> 11.0
 X: 1, 2 -> 7.0
 X: 2, 3 -> 7.0
 X: 3, 4 -> 9.0
 X: 4, 5 -> 11.0
 X: 5, 6 -> 11.0
 X: 1, 2 -> 7.0
 X: 2, 3 -> 7.0
 X: 3, 4 -> 9.0
 X: 4, 5 -> 11.0

X: 5, 6 -> 11.0
 X: 1, 2 -> 7.0
 X: 2, 3 -> 7.0
 X: 3, 4 -> 9.0
 X: 4, 5 -> 11.0
 X: 5, 6 -> 11.0
 X: 1, 2 -> 7.0
 X: 2, 3 -> 7.0
 X: 3, 4 -> 9.0
 X: 4, 5 -> 11.0
 X: 5, 6 -> 11.0
 X: 1, 2 -> 7.0
 X: 2, 3 -> 7.0
 X: 3, 4 -> 9.0
 X: 4, 5 -> 11.0
 X: 5, 6 -> 11.0
 X: 1, 2 -> 7.0
 X: 2, 3 -> 7.0
 X: 3, 4 -> 9.0
 X: 4, 5 -> 11.0
 X: 5, 6 -> 11.0
 X: 1, 2 -> 7.0
 X: 2, 3 -> 7.0
 X: 3, 4 -> 9.0
 X: 4, 5 -> 11.0
 X: 5, 6 -> 11.0
 X: 1, 2 -> 7.0
 X: 2, 3 -> 7.0
 X: 3, 4 -> 9.0
 X: 4, 5 -> 11.0
 X: 5, 6 -> 11.0



2. Analisis grafik perbandingan nilai asli dan nilai prediksi pada soal nomor 1!

JAWAB :

Grafik tersebut menggambarkan nilai asli (biru) yang menunjukkan pola peningkatan yang signifikan (eksponensial atau non-linear), sementara nilai prediksi (kuning) cenderung konstan. Hal itu yang membuat model SVM mengalami underfitting dan gagal menangkap pola kompleks dalam data, kemungkinan disebabkan oleh penggunaan kernel (fungsi yang digunakan) tidak sesuai (misalnya, kernel linier untuk data non-linear), parameter model yang belum dioptimalkan, atau kurangnya transformasi data.

3. Kerjakan soal nomor 1 dengan nilai $a = i+2$ dan $b = i+4$!

```
def Trapezoid(a,b,f):
    n = 100
    def trapezoid(f,a,b,n=100):
        h = (b-a)/n
        sum = 0.0
        for i in range(1,n):
            x = a+i*h
            sum = sum + f(x)
            integral = (h/2)*(f(a)+2*sum+f(b))
        return integral
    integral = trapezoid(f,a,b,n)
    print(a," ",b," ",round(integral,2))

# Melakukan looping untuk membuat database dari beberapa soal integral
for i in range(0,5):
    Trapezoid(i+2,i+4,lambda x: 2*x)
for i in range(0,5):
    Trapezoid(i+2,i+4,lambda x: 2*x + 2)
for i in range(0,5):
    Trapezoid(i+2,i+4,lambda x: 2*x + 4)
for i in range(0,5):
    Trapezoid(i+2,i+4,lambda x: 4*x + 6)
for i in range(0,5):
    Trapezoid(i+2,i+4,lambda x: 6*x + 8)
for i in range(0,5):
    Trapezoid(i+2,i+4,lambda x: 8*x + 10)
for i in range(0,5):
    Trapezoid(i+2,i+4,lambda x: 10*x + 12)
for i in range(0,5):
    Trapezoid(i+2,i+4,lambda x: 12*x + 14)
for i in range(0,5):
    Trapezoid(i+2,i+4,lambda x: 14*x + 12)
for i in range(0,5):
    Trapezoid(i+2,i+4,lambda x: 20*x + 40)
```

```
2 , 4 , 12.0
3 , 5 , 16.0
4 , 6 , 20.0
5 , 7 , 24.0
6 , 8 , 28.0
2 , 4 , 16.0
3 , 5 , 20.0
4 , 6 , 24.0
5 , 7 , 28.0
6 , 8 , 32.0
2 , 4 , 20.0
3 , 5 , 24.0
4 , 6 , 28.0
5 , 7 , 32.0
6 , 8 , 36.0
2 , 4 , 36.0
3 , 5 , 44.0
```

4 , 6 , 52.0
5 , 7 , 60.0
6 , 8 , 68.0
2 , 4 , 52.0
3 , 5 , 64.0
4 , 6 , 76.0
5 , 7 , 88.0
6 , 8 , 100.0
2 , 4 , 68.0
3 , 5 , 84.0
4 , 6 , 100.0
5 , 7 , 116.0
6 , 8 , 132.0
2 , 4 , 84.0
3 , 5 , 104.0
4 , 6 , 124.0
5 , 7 , 144.0
6 , 8 , 164.0
2 , 4 , 100.0
3 , 5 , 124.0
4 , 6 , 148.0
5 , 7 , 172.0
6 , 8 , 196.0
2 , 4 , 108.0
3 , 5 , 136.0
4 , 6 , 164.0
5 , 7 , 192.0
6 , 8 , 220.0
2 , 4 , 200.0
3 , 5 , 240.0
4 , 6 , 280.0
5 , 7 , 320.0
6 , 8 , 360.0

```

import numpy as np
import pandas as pd
from sklearn import svm
from google.colab import drive
import matplotlib.pyplot as plt

# Mount Google Drive
drive.mount('/content/drive')

# Path ke file Database.txt di google drive
file_path = '/content/drive/My Drive/MODUL 11 PRAKTIKUM/Trapezoid No 3.txt'

# Membaca data dari file
Database = pd.read_csv(file_path, sep=",", header=0)

# X = Data, y = Target
X = Database[['a', 'b']] # Pastikan kolom sesuai dengan nama yang ada di file
y = Database['Target']

# Membuat dan melatih model SVM
clf = svm.SVC()
clf.fit(X.values, y)

# Melakukan prediksi
y_pred = clf.predict(X.values)

# Menampilkan hasil prediksi
print("Hasil prediksi:")
for i, pred in enumerate(y_pred):
    print(f"X: {X.iloc[i, 0]}, {X.iloc[i, 1]} -> {pred}")

# Membuat plot perbandingan nilai asli dengan nilai prediksi
plt.figure(figsize=(10, 6))
plt.plot(range(len(y)), y, 'o-', label='Nilai Asli (Target)', color='blue')
plt.plot(range(len(y_pred)), y_pred, 'x--', label='Nilai Prediksi (SVM)', color='yellow')

# Menambahkan label dan judul
plt.xlabel('Indeks Data')
plt.ylabel('Nilai')
plt.title('Perbandingan Nilai Asli vs Nilai Prediksi Menggunakan SVM')
plt.legend()
plt.grid()
plt.show()

```

Mounted at /content/drive

Hasil prediksi:

X: 2, 4 -> 20.0

X: 3, 5 -> 20.0

X: 4, 6 -> 24.0

X: 5, 7 -> 28.0

X: 6, 8 -> 28.0

X: 2, 4 -> 20.0

X: 3, 5 -> 20.0

X: 4, 6 -> 24.0

X: 5, 7 -> 28.0

X: 6, 8 -> 28.0

X: 2, 4 -> 20.0

X: 3, 5 -> 20.0

X: 4, 6 -> 24.0

X: 5, 7 -> 28.0

X: 6, 8 -> 28.0

X: 2, 4 -> 20.0

X: 3, 5 -> 20.0

X: 4, 6 -> 24.0

X: 5, 7 -> 28.0

X: 6, 8 -> 28.0

X: 2, 4 -> 20.0

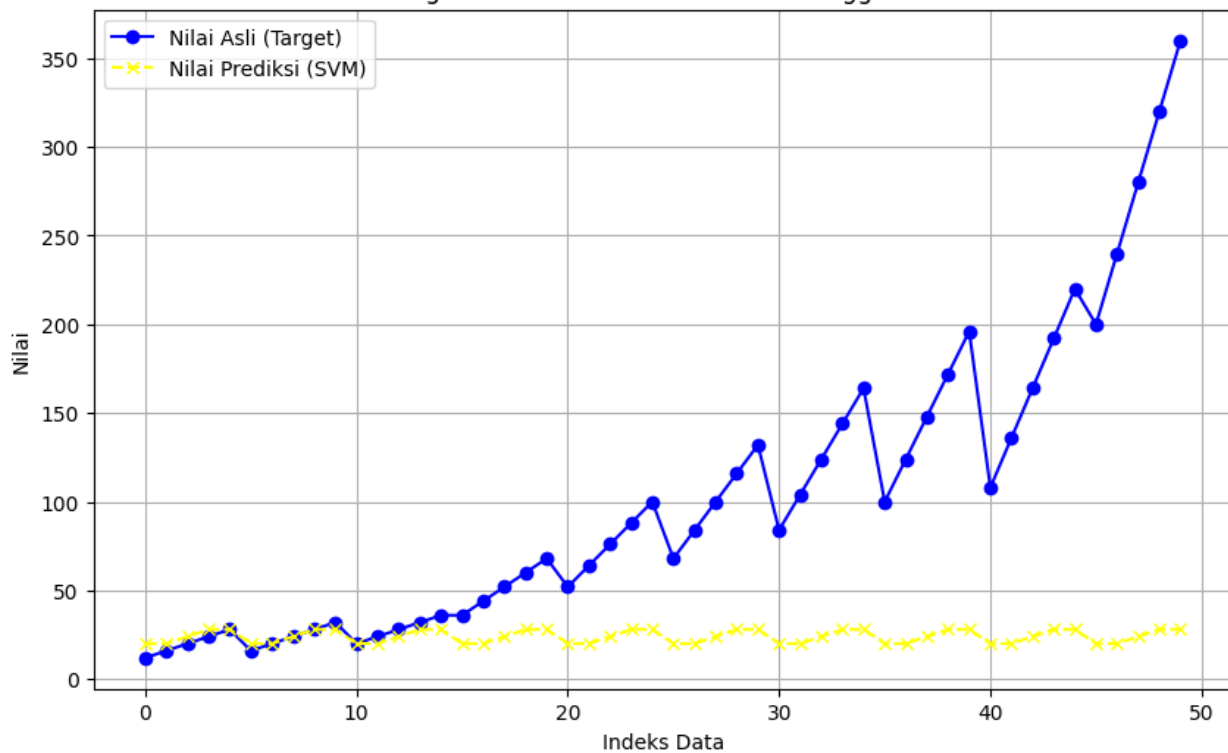
X: 3, 5 -> 20.0

X: 4, 6 -> 24.0

X: 5, 7 -> 28.0

X: 6, 8 -> 28.0
 X: 2, 4 -> 20.0
 X: 3, 5 -> 20.0
 X: 4, 6 -> 24.0
 X: 5, 7 -> 28.0
 X: 6, 8 -> 28.0
 X: 2, 4 -> 20.0
 X: 3, 5 -> 20.0
 X: 4, 6 -> 24.0
 X: 5, 7 -> 28.0
 X: 6, 8 -> 28.0
 X: 2, 4 -> 20.0
 X: 3, 5 -> 20.0
 X: 4, 6 -> 24.0
 X: 5, 7 -> 28.0
 X: 6, 8 -> 28.0
 X: 2, 4 -> 20.0
 X: 3, 5 -> 20.0
 X: 4, 6 -> 24.0
 X: 5, 7 -> 28.0
 X: 6, 8 -> 28.0
 X: 2, 4 -> 20.0
 X: 3, 5 -> 20.0
 X: 4, 6 -> 24.0
 X: 5, 7 -> 28.0
 X: 6, 8 -> 28.0

Perbandingan Nilai Asli vs Nilai Prediksi Menggunakan SVM



4. Analisis grafik perbandingan nilai asli dan nilai prediksi pada soal nomor 3!

JAWAB :

Pada grafik ini sebenarnya sama dengan grafik sebelumnya, yaitu menunjukkan nilai asli yang pola peningkatannya tajam dengan prediksi SVM yang masih tidak bisa menangkap polanya. Namun yang membedakan antara grafik di no 1 adalah pada skala nilai asli (target) dan tingkat kenaikannya. Grafik sebelumnya memiliki nilai asli maksimum sekitar 150, sedangkan grafik baru mencapai sekitar 350 dengan pola peningkatan yang jauh lebih tajam. Meskipun pada kedua grafik prediksi SVM tetap datar dan mengalami underfitting, perbedaan antara nilai asli dan prediksi pada grafik baru menjadi lebih mencolok karena nilai asli meningkat secara drastis. Hal ini semakin memperjelas bahwa model SVM tidak mampu menangkap pola non-linear atau eksponensial dalam data.

5. Jelaskan algoritma program integral trapezoid dan prediksi nilai integral trapezoid menggunakan metode Support Vector Machine!

JAWAB :

Kode program ini menghitung integral numerik menggunakan metode trapezoid dan menganalisis data hasilnya dengan Support Vector Machine (SVM). Fungsi trapezoid membagi interval $[a,b]$ menjadi 100 subinterval untuk menghitung integral dari berbagai fungsi linear, seperti $2x$ hingga $20x+40$, pada interval $[i+1,i+2]$ melalui looping dan juga memvariasikannya menjadi interval $[i+1,i+4]$ pada nomor 3. Hasil integral dicetak sebagai database, lalu dianalisis menggunakan SVM. Data dibaca dari file TUGAS TRAPEZOID.txt, dipisahkan menjadi fitur X (kolom a, b) dan target y, kemudian model SVM dilatih untuk memprediksi y berdasarkan X. Hasil prediksi dibandingkan dengan nilai asli melalui teks dan grafik, yang menampilkan performa model dalam mempelajari hubungan antara fitur dan target. Kode ini digunakan untuk melihat akurasi dari metode SVM untuk memvisualkan data integral trapezoid.