

Chapter 5: Support Vector Machines (SVMs)

A. Tujuan Bab Ini

Memahami dan menerapkan Support Vector Machines (SVM), salah satu algoritma yang sangat kuat untuk klasifikasi dan regresi, terutama di dataset kecil hingga menengah.

B. Konsep Utama dalam Bab Ini

1. Intuisi SVM untuk Klasifikasi

SVM mencoba mencari hyperplane terbaik yang memisahkan dua kelas dengan margin maksimum.

Support Vectors = Titik data terdekat ke hyperplane.

SVM akan tetap stabil walaupun data di luar support vector berubah.

Fungsi Tujuan:

Maksimalkan margin antara dua kelas, sambil meminimalkan kesalahan klasifikasi.

2. Soft Margin Classification

Kadang data tidak bisa dipisahkan sempurna (noisy, overlapping).

Maka digunakan "soft margin" → izinkan pelanggaran kecil terhadap aturan margin.

Parameter penting:

C → ukuran penalti untuk kesalahan:

Kecil → lebih longgar, toleran noise

Besar → lebih ketat, bisa overfitting

3. Nonlinear SVM (Trick Kernel)

Jika data tidak bisa dipisahkan secara linear, SVM tetap bisa digunakan dengan kernel trick.

Alih-alih menambahkan fitur manual (misal x^2 , $x*y$, x^3), kernel function menghitung jarak di ruang berdimensi tinggi tanpa transformasi eksplisit.

Kernel Umum:

Polynomial kernel: cocok untuk data dengan kurva

RBF (Radial Basis Function) / Gaussian kernel: cocok untuk data kompleks & nonlinier

Parameter penting:

γ (gamma): menentukan seberapa jauh pengaruh satu training example.

4. SVM untuk Regresi (SVR)

Alih-alih memisahkan kelas, SVR mencoba menemukan fungsi prediksi dengan deviasi maksimal tertentu dari nilai aktual (ϵ -insensitive zone).

Sama seperti klasifikasi tapi prinsipnya adalah: "cukup dekat".

5. Skalabilitas & Komputasi

SVM bekerja sangat baik pada dataset kecil hingga menengah

Tapi tidak efisien untuk dataset besar atau banyak fitur (high-dimensional)

Kernelized SVMs memerlukan $O(m^2 - m^3)$ waktu \rightarrow perlu hati-hati dengan ukuran data

Proyek / Kode Praktik Bab Ini

Isi Notebook / Proyek

1. Linear SVM Classification

```
from sklearn.svm import SVC
```

```
svm_clf = SVC(kernel="linear", C=1)
```

```
svm_clf.fit(X, y)
```

Membangun model klasifikasi linear sederhana

Mencoba berbagai nilai C untuk melihat efek margin ketat vs longgar

2. Nonlinear Classification

```
poly_kernel_svm_clf = SVC(kernel="poly", degree=3, coef0=1, C=5)
```

Menggunakan polynomial kernel untuk menangani data yang tidak bisa dipisahkan secara linear

```
rbf_kernel_svm_clf = SVC(kernel="rbf", gamma=5, C=0.001)
```

Menggunakan Gaussian (RBF) kernel yang sangat powerful untuk bentuk keputusan kompleks

3. SVM untuk Regresi

```
from sklearn.svm import SVR
```

```
svm_reg = SVR(kernel="linear", C=100, epsilon=1.5)
```

Implementasi Support Vector Regression

Menyesuaikan parameter C dan epsilon untuk melihat toleransi terhadap error

- Visualisasi

Banyak plot decision boundary untuk membandingkan:

Linear vs nonlinear

C besar vs kecil

Gamma tinggi vs rendah

Sangat membantu untuk memahami dampak parameter

- Hal yang dipelajari

Konsep	Manfaat
SVM	Algoritma klasifikasi kuat, bagus untuk dataset kecil
Support Vectors	Data kritis yang menentukan model
C Parameter	Kontrol kompromi antara akurasi dan margin
Kernel Trick	Menangani data non-linear tanpa komputasi mahal
SVR	Regresi dengan prinsip SVM

- Contoh Aplikasi Nyata SVM

- Deteksi spam
- Klasifikasi gambar
- Pengklasifikasian teks
- Pengenalan wajah
- Regresi untuk prediksi harga/permintaan