

BAB II

TINJAUAN PUSTAKA

2.1 Diabetes

Diabetes adalah penyakit kronis pada kesehatan yang mempengaruhi cara kerja metabolisme tubuh dalam merubah makanan menjadi energi. Tubuh manusia umumnya memecah sebagian besar makanan yang dikonsumsi menjadi gula (glukosa) dan melepaskannya menuju ke aliran darah [2]. Ketika kadar gula darah seseorang naik, tubuh akan memberi sinyal pada pankreas untuk melepaskan insulin. Insulin bertindak seperti pos pengawas yang memberi gula darah akses untuk masuk ke sel tubuh yang nantinya digunakan sebagai energi. Namun berbeda dengan penderita diabetes, tubuh dari orang yang memiliki diabetes tidak dapat membuat insulin yang cukup atau bahkan tidak dapat digunakan sebagaimana mestinya. Ketika tubuh kekurangan insulin dan berhenti merespons adanya insulin, akan ada banyak gula darah yang menetap dan terkumpul di aliran darah.

Seiring waktu, hal ini beresiko dapat menyebabkan masalah kesehatan yang serius seperti penyakit jantung, kehilangan penglihatan, dan juga penyakit ginjal. Hingga saat ini masih belum ada obat yang bisa menyembuhkan diabetes, tetapi dengan menurunkan berat badan, mengkonsumsi makanan sehat dan sering melakukan banyak aktifitas bisa membantu mengurangi resiko komplikasi yang disebabkan oleh diabetes. Diabetes memiliki tiga jenis utama yaitu tipe 1, tipe 2 dan gestasional [3]. Diabetes tipe 1 terjadi karena reaksi dimana tubuh menyerang dirinya sendiri secara tidak sengaja atau biasa disebut autoimun. Hal ini menyebabkan tubuh berhenti dalam membuat insulin. Penderita diabetes 5-10% diantaranya mengidap diabetes tipe 1. Gejala dari diabetes tipe 1 seringkali dapat berkembang dengan cepat dan umumnya terjadi pada anak-anak, remaja dan dewasa muda. Jika seseorang menderita diabetes tipe 1, maka mereka perlu mengkonsumsi makanan yang mengandung insulin setiap hari untuk mengurangi kemungkinan terjadinya kondisi yang semakin memburuk.

Diabetes tipe 2 adalah kondisi dimana tubuh seseorang tidak bisa menggunakan insulin dengan baik sehingga gula darah tidak terjaga dan berada di tingkat yang tidak normal. Tingkat penderitanya sendiri sangatlah tinggi karena 90-95% dari penderita diabetes mengidap tipe 2. Tipe ini berkembang bisa selama bertahun-tahun dan dialami oleh orang dewasa tetapi lebih banyak terjadi pada anak-anak, remaja, dan dewasa muda. Gejala dari tipe 2 jarang terlihat maupun dirasakan dan biasanya baru terlihat saat mulai terjadi komplikasi, sehingga penting untuk memeriksakan gula darah untuk seseorang yang beresiko memiliki tipe ini.

Diabetes Gestasional biasanya terjadi pada perempuan yang sedang hamil sekalipun orang tersebut tidak pernah menderita diabetes selama hidupnya. Ketika seseorang menderita diabetes gestasional, bayi yang dikandung kemungkinan berisiko lebih tinggi untuk mengalami masalah kesehatan. Diabetes gestasional biasanya akan mulai hilang setelah melahirkan. Namun, hal ini bisa meningkatkan resiko munculnya diabetes tipe 2 dimana bayi dari penderita akan cenderung mengalami obesitas pada saat masih anak-anak atau remaja.

2.2 Sistem Pendukung Keputusan

Sistem pendukung keputusan atau dalam bahasa Inggris *Decision Support System* (DSS) adalah program sistem terkomputerisasi yang digunakan untuk mendukung dalam penentuan, penilaian, dan pemberian solusi dalam suatu masalah [4]. Sistem pendukung keputusan menyaring dan menganalisis sejumlah besar data, mengumpulkan informasi komprehensif yang dapat digunakan untuk memecahkan masalah maupun dalam pengambilan keputusan atas permasalahan tersebut. *Decision Support Sistem* (DSS) didukung oleh sebuah sistem informasi berbasis komputer yang diharapkan membantu seseorang meningkatkan kinerjanya dalam pengambilan tindakan untuk menyelesaikan suatu permasalahan [5].

Sistem pendukung keputusan dapat menjadi sistem terkomputerisasi secara sepenuhnya yang bekerja secara *stand-alone* dan bisa juga didukung oleh manusia, namun dalam beberapa kasus bisa juga keduanya dapat diterapkan secara bersamaan, tergantung dari seberapa rumit permasalahan yang sedang dihadapi.

Sistem pendukung keputusan yang ideal bisa menganalisis informasi dan membuat keputusan maupun solusi untuk membantu penggunanya. Sistem pendukung keputusan juga dibuat untuk memungkinkan pengguna manusia membuat solusi yang lebih tepat secara cepat dalam membuat sebuah keputusan.

Pengertian sistem pendukung keputusan diperkenalkan oleh McLeod pada tahun 1998. Ia menyatakan bahwa sistem pendukung keputusan merupakan sistem yang dapat menghasilkan informasi yang ditujukan pada suatu permasalahan. Selain itu sistem pendukung keputusan merupakan suatu sistem informasi yang ditujukan untuk membantu seseorang dalam memecahkan masalah yang sedang dihadapi [6].

Sistem pendukung keputusan memiliki beberapa jenis, diantaranya adalah :

- *Data-Driven DSS*

Data-Driven DSS atau sistem pendukung keputusan berbasis data yaitu sistem yang membuat keputusan dari sebuah data melalui database internal atau database eksternal. Sistem ini menggunakan teknik data mining yang memungkinkan untuk memprediksi kejadian yang akan terjadi. Sistem pendukung keputusan berbasis data ini juga sudah banyak digunakan untuk membantu membuat keputusan di sektor publik, contohnya seperti ramalan cuaca atau memprediksi kemungkinan kapan akan terjadi kemacetan.

- *Model-Driven DSS*

Model-Driven DSS atau sistem pendukung keputusan berbasis model adalah sistem yang dibangun berdasarkan model keputusan dasar untuk memutuskan suatu permasalahan dari individu. Sistem pendukung keputusan berbasis model biasanya dirancang berdasarkan kesesuaian maupun kebutuhan dari pengguna sehingga sudah ditentukan jauh sebelum sistemnya akan dirancang, sistem pendukung dengan menggunakan basis model ini dibuat untuk membantu menganalisis skenario yang dihadapi pengguna.

- *Communication-Driven and Group DSS*

Communication-Driven and Group DSS atau sistem pendukung keputusan berbasis komunikasi dan data yaitu sistem pendukung keputusan terintegrasi dengan alat komunikasi seperti email atau pesan suara yang memungkinkan lebih dari satu orang mengerjakan tugas yang sama. Tujuan sistem pendukung keputusan ini adalah untuk meningkatkan komunikasi antara pengguna dan sistem untuk meningkatkan efisiensi dan efektivitas sistem yang terbagi secara menyeluruh.

- *Knowledge-Driven DSS*

Knowledge-Driven DSS atau sistem pendukung keputusan berbasis pengetahuan adalah sistem yang dirancang untuk pengambilan keputusan yang tepat dan efektif dengan menggunakan data, informasi, dan manajemen pengetahuan. Sistem ini berfokus pada pengambilan keputusan berdasarkan pengetahuan yang relevan berdasarkan pada kecerdasan buatan, maupun penerapan teknologi informasi dan komunikasi. Selain itu juga sistem ini mendukung pengambilan keputusan dengan teknik prediksi dan rekomendasi, namun tergantung pada kriteria dan berbagai klasifikasi.

2.3 Penelitian-Penelitian Sebelumnya

Ada beberapa penelitian berkaitan dengan sistem pendukung keputusan terkait penyakit diabetes yang sudah dilakukan oleh sejumlah peneliti lain sebelumnya. Penelitian pertama yaitu penelitian yang berjudul "Sistem Pendukung Keputusan Pengobatan Penderita Diabetes Menggunakan Integrasi Decision Table Dan Algoritma Genetika." yang dilakukan oleh Dana Sumartini (2012). Penelitian ini membahas tentang metode pengobatan dan terapi dari penderita diabetes melalui pengaturan jadwal makan, porsi makan, kandungan nutrisi dan juga makanan apa yang dikonsumsi.

Metode yang digunakan dalam menentukan hal tersebut adalah *Decision Table* yang berperan dalam penentuan pengobatan diabetes namun karena metode tersebut masih menggunakan logika sederhana diperlukan juga adanya integrasi dengan metode Algoritma Genetika untuk menghasilkan output yang lebih baik.

Dengan integrasi kedua metode tersebut ditemukan optimasi bahan makanan yang cocok untuk tipe diabetes penderita berdasarkan nutrisi dari makanan tersebut. Dalam pengujian, dilakukan penjumlahan dari 20 bahan makanan yang digunakan dan percobaan tersebut dilakukan secara berulang-ulang namun dengan data input yang berbeda-beda, sehingga didapatkan hasil analisa berupa total kebutuhan kalori perorang, berat badan ideal dan jenis diabetes. Selain itu juga ada hasil dari nutrisi seperti kebutuhan nutrisi tiap orang dan juga bahan makanan yang disarankan maupun yang harus dihindari [7].

Penelitian kedua juga pernah dilakukan oleh Umikulsum Indah Lestari, Anis Yusrotun Nadhiroh, dan Cahyuni Novia dengan judul penelitian yaitu “Penerapan Metode K-Nearest Neighbor Untuk Sistem Pendukung Keputusan Identifikasi Penyakit Diabetes”. Penelitian tersebut menggunakan teknik di dalam *data mining*, yaitu klasifikasi dengan K-Nearest Neighbor (KNN). Pada teknik tersebut dibuat 8 nilai variabel yaitu tingkat kehamilan, kadar gula darah atau glukosa, tekanan darah, ketebalan dari lipatan kulit pada trisep, kadar insulin, index masa tubuh atau berat badan, riwayat penyakit diabetes melitus pada anggota keluarga, dan juga usia. Pada penelitian tersebut juga dilakukan pengumpulan dataset dari penyakit diabetes yang diambil dari data publik Pima Indians Diabetes Database. Peneliti yang sebelumnya juga melakukan wawancara dengan seorang dokter untuk mengetahui informasi yang lebih detail mengenai penyakit diabetes [8].

Setelah pengumpulan data, peneliti melakukan tahap pre-processing dengan menganalisis dataset tersebut, namun pada data ditemukan atribut yang tidak lengkap sehingga untuk mengatasi kekurangan atribut tersebut dibuat aturan nilai nol untuk atribut yang kosong. Penggunaan metode KKN dilakukan sebanyak dua kali percobaan, yaitu pertama pada sistem dan kedua pada perhitungan dengan excel yang membuat hasil perhitungan *Euclidean*.

Langkah-langkah yang diterapkan pada algoritma KKN untuk memproses data tersebut adalah menentukan nilai k , menghitung kuadrat Euclid, mengurutkan objek, mengumpulkan label class Y atau klasifikasi KKN dan terakhir melakukan prediksi berdasarkan label dari kelas Y yang paling mayoritas. Setelah itu dilakukan uji coba sistem. Proses uji coba dilakukan dengan mencari nilai dari k yang paling optimal melalui 50 data. Tingkat keberhasilan pun diukur berdasarkan keakuratan yang sudah dihasilkan oleh sistem sehingga semua data yang diproses dapat mengidentifikasi setiap orang yang mereka memiliki diabetes atau tidak.

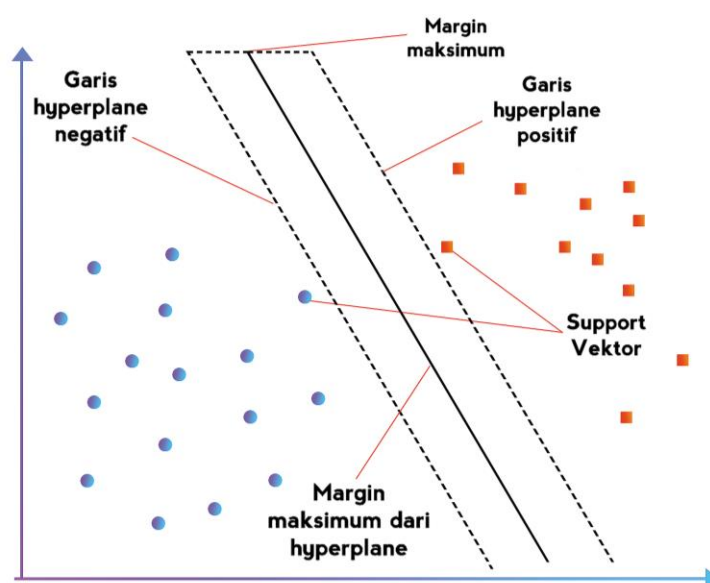
Penelitian ketiga yang pernah diteliti sebelumnya adalah penelitian yang berjudul “Sistem Pendukung Keputusan Menggunakan Fuzzy Logic Tahani Untuk Penentuan Golongan Obat Sesuai Dengan Penyakit Diabetes” oleh Charolina Debora, Josua Armando Watuseke, Prince David Gibrael Saerang, dan Reynaldo Joshua. Sebelum klasifikasi dari golongan obat akan dilakukan, keadaan diabetes pada setiap orang yang menderita penyakit tersebut akan dimasukkan menjadi parameter berupa, umur, data tes hasil HbA1C, data tes hasil GDP, dan data tes hasil TTGO [9].

Variabel fungsi keanggotaan yang pertama yaitu adalah umur, karena usia adalah salah satu bagian yang sangat vital untuk mengetahui di usia berapakah tingkat kerentanan resiko diabetes terjadi. Selanjutnya ada juga fungsi keanggotaan hasil HbA1C atau Hemoglobin A1C, yaitu data pengukuran dari rata-rata kadar gula darah seseorang selama tiga bulan, pemeriksaan ini biasanya digunakan untuk menganalisa perkembangan dan tingkat keberhasilan pada pasien yang melakukan terapi diabetes. Kemudian ada fungsi keanggotaan GDP atau Gula Darah Puasa yaitu rentang dari kadar gula darah normal saat pasien belum melakukan makan, pemeriksaan dari GDP dilakukan dengan tujuan untuk melihat pasien apakah pasien tersebut memiliki resiko akan prediabetes atau diabetes. Terakhir adalah fungsi keanggotaan TTGO atau Tes Toleransi Glukosa Oral, yaitu pemeriksaan resistensi insulin pada seseorang yang membantu dalam klasifikasi gula darahnya.

Setelah semua fungsi keanggotaan dibuat maka metode Fuzzy bisa memproses semua data yang dimasukkan. Penulis mengambil 5 dari 20 data penderita diabetes yang akan diuji sehingga metode yang digunakan oleh sistem yang sudah dibuat dapat membantu dokter dalam mengambil keputusan untuk menentukan jenis obat apa yang boleh dikonsumsi oleh penderita.

2.4 Support Vector Machine (SVM)

Support Vector Machine atau biasa disingkat SVM adalah salah satu algoritma *supervised learning* yang biasanya digunakan untuk memecahkan masalah klasifikasi dalam *machine learning*. Tujuan dari algoritma SVM adalah membuat sebuah garis atau batas keputusan terbaik yang dapat memisahkan ruang n-dimensi ke dalam kelas sehingga seseorang dapat dengan mudah menempatkan titik data baru ke dalam kategori yang benar di masa mendatang. Batas keputusan terbaik ini disebut *hyperplane*, garis dari *hyperplane* yang akan dicari yaitu garis yang letaknya berada berdekatan antar satu kelas dengan kelas lainnya, dengan cara menghitung lebar dari margin secara maksimal. Margin adalah jarak dari *hyperplane* atau bidang pemisah optimal terhadap poin-poin terdekat yang berada dimasing-masing kelas. Poin dengan kelas berbeda yang saling berdekatan ini disebut sebagai *support vector* [10].



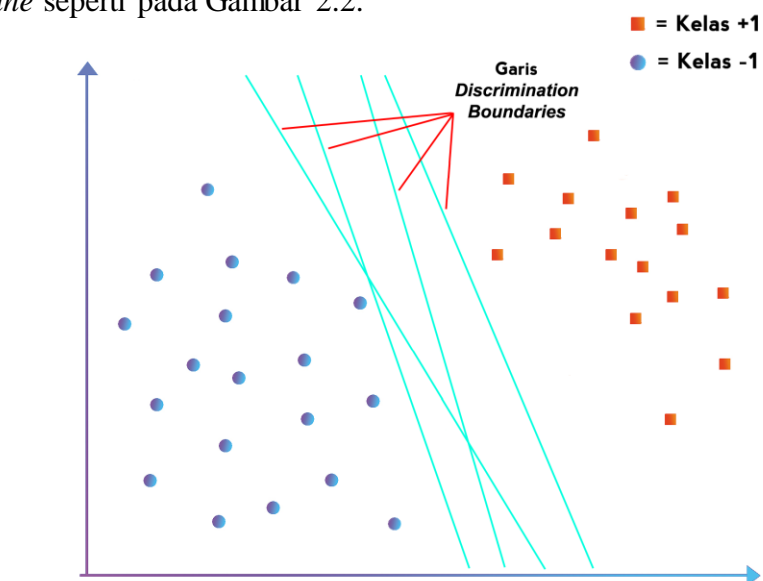
Gambar 2.1 Diagram *Support Vector Machine*

SVM memilih titik atau vektor yang akan dijadikan patokan dalam pembuatan *hyperplane*, dari gambar diatas dapat dilihat jika ada dua titik data dengan kategori yang berbeda berada di posisi yang hampir berdekatan, Kedua data tersebut akan dipilih untuk menjadi perwakilan dalam membuat garis *hyperplane*, titik yang dipilih ini disebut sebagai *Support Vector*.

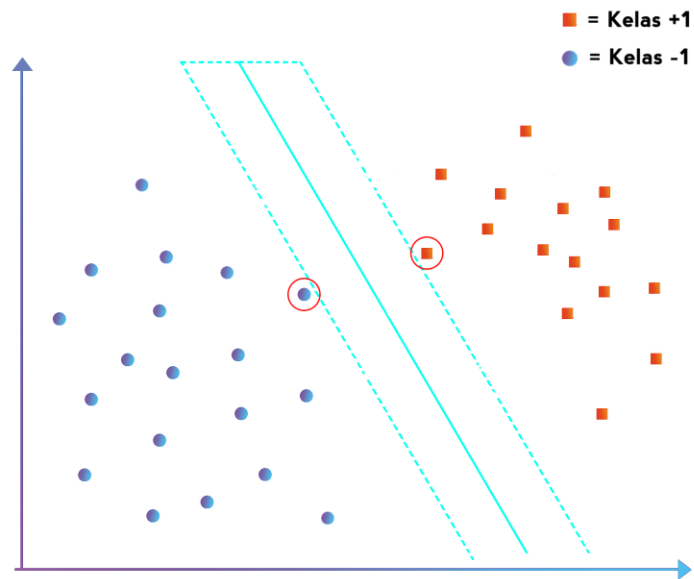
Selain itu, SVM juga dibagi menjadi dua jenis diantaranya:

- SVM Linier, yaitu SVM yang biasanya digunakan untuk data yang dapat dipisahkan secara linier. Jika suatu dataset bisa diklasifikasikan menjadi dua kelas hanya dengan menggunakan satu garis lurus, maka data tersebut dapat dianggap sebagai data yang dapat dipisahkan secara linier.
- SVM Non-Linier, yaitu SVM yang digunakan pada data yang dipisahkan secara non-linear, artinya jika suatu dataset tidak dapat diklasifikasikan dengan menggunakan garis lurus, maka dataset ini disebut sebagai data non-linear.

Karena algoritma SVM adalah salah satu bagian dari metode klasifikasi maka algoritma ini akan mencari garis *hyperplane* dalam memisahkan dua data yang kategorinya berbeda. Adapun contoh bagaimana SVM melakukan proses pencarian *hyperplane* seperti pada Gambar 2.2.



Gambar 2.2 Pencarian *hyperplane* dengan membuat banyak garis



Gambar 2.3 Pengambilan *hyperplane* terbaik untuk memisah kelas -1 dan +1

Pada Gambar 2.3 dapat dilihat terdapat dua kelas yaitu -1 yang ditunjukkan dengan lingkaran berwarna biru dan +1 yang ditunjukkan dengan kotak berwarna oranye. Proses klasifikasi bisa ditemukan dengan membuat beberapa garis yang disebut *discrimination boundaries* untuk memisahkan kedua kelas seperti pada Gambar 2.2. Setelah itu SVM mengambil salah satu garis *hyperplane* terbaik yang bisa menjadi pemisah dengan cara mengukur *margin* yang digambarkan dengan garis biru putus-putus pada Gambar 2.3 dan juga mencari titik maksimumnya.

SVM Linear memiliki persamaan *hyperplane* yaitu:

$$w^T x + b = 0 \quad (\text{rumus 2.1})$$

w = Penentu support vektor

T = Transpos nilai W

x = Variabel x dan y pada vektor W

b = Titik potong garis

Untuk benar-benar memahami proses dari *Support Vector Machine*, kita perlu memahami bagaimana persamaan hyperplane bisa diterapkan dalam mencari garis pemisah antar kelas +1 dan kelas -1.

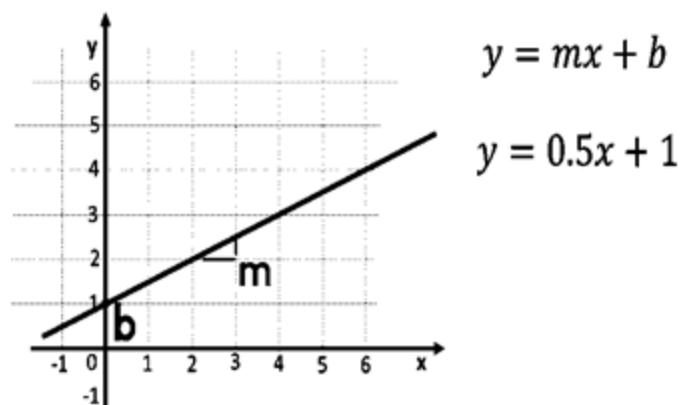
Karena *hyperplane* adalah persamaan yang berhubungan dengan garis, maka rumus sederhana dari persamaan garis gradien adalah seperti pada Gambar 2.4.

$$y = mx + b \quad (\text{rumus 2.2})$$

y = Gradien persamaan garis

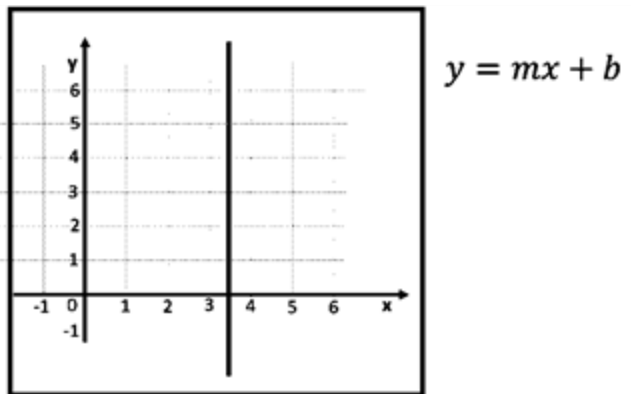
mx = Nilai kemiringan

b = Titik potong



Gambar 2.4 Contoh persamaan garis gradien

Misalkan sebagai contoh kita asumsikan bahwa ada sebuah garis lurus yang sama seperti Gambar 2.4, yaitu sebuah garis yang memiliki nilai $y = 0.5x + 1$. Variabel m adalah nilai kemiringan atau gradient, dan b adalah titik potong dari nilai y . Namun persamaan tersebut akan mengalami kesulitan apabila memperkirakan nilai dari garis vertikal seperti Gambar 2.5.



Gambar 2.5 Garis vertikal

Sehingga karena hal itu, kita perlu untuk menggunakan bentuk umum lain dari persamaan garis gradient dari $y = mx + b$ yaitu:

$$Ax + By + C = 0 \quad (\text{rumus 2.3})$$

A, B dan C = bilangan bulat

x = koordinat titik X yang ditentukan oleh A

y = koordinat titik Y yang ditentukan oleh B

Setelah itu kita perlu merubah persamaan tersebut dengan cara memindahkan A dengan C kesisi kanan, lalu jadikan B sebagai pembagi masing- masing A dan C. Jika telah dilakukan transformasi maka bentuk dari persamaan akan berubah menjadi persamaan dibawah ini yaitu:

$$y = -\frac{A}{B}x - \frac{C}{B} \quad (\text{rumus 2.4})$$

$-\frac{A}{B}$ adalah penentu kemiringan, sedangkan $-\frac{C}{B}$ adalah titik potong.

Kemiringan

$$Ax + By + C = 0$$

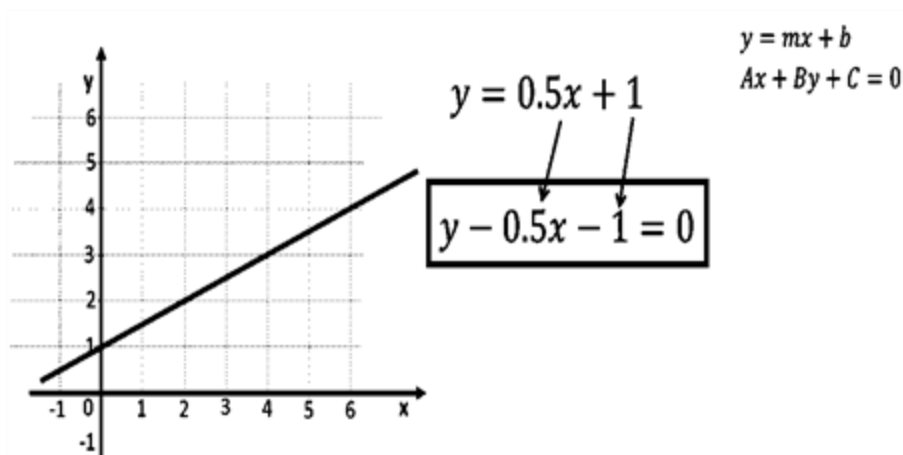
Titik Potong y

$$y = -\frac{A}{B}x - \frac{C}{B}$$

Kemiringan dan titik potong

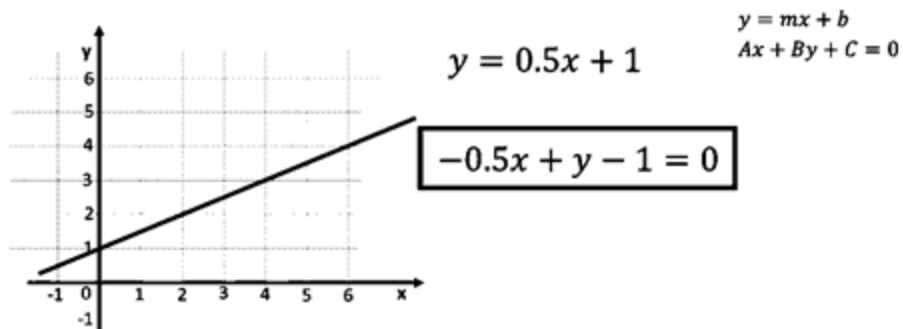
Gambar 2.6 Persamaan umum lain awal dan setelah diubah

Jika dibandingkan dengan rumus sebelumnya kita bisa mengetahui jika koefisien A akan mempengaruhi kemiringan, lalu koefisien C mempengaruhi titik potong y, dan koefisien B akan mempengaruhi keduanya. Sehingga contoh persamaan $y = 0.5x + 1$ pada Gambar 2.4 sebelumnya, bisa kita ubah bentuknya menjadi $y - 0.5x - 1 = 0$ seperti Gambar 2.7.



Gambar 2.7 Merubah persamaan gradient ke bentuk umum lain.

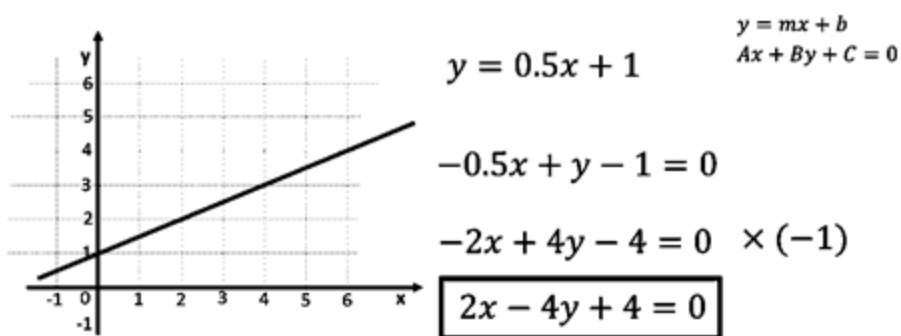
Namun agar bentuknya konsisten seperti persamaan $Ax + By + C = 0$, maka kita perlu melakukan penukaran posisi y dengan $0.5x$, sehingga menjadi $-0.5x + y - 1 = 0$ atau seperti Gambar 2.8.



Gambar 2.8 Menukar posisi variable $-0.5x$ dan y

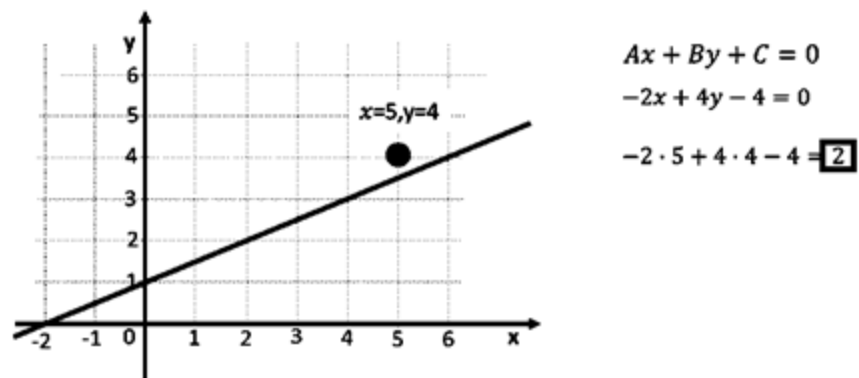
Dalam bentuk $Ax + By + C = 0$, koefisien biasanya di ekspresikan sebagai integer. Jika kita misal kita mengalikan persamaan $-0.5x + y - 1 = 0$ semuanya dengan 4, maka kita akan mendapatkan $-2x + 4y - 4 = 0$ yang persamaannya tetap mendeskripsikan garis yang sama seperti sebelumnya, karena jika kita selesaikan persamaan tersebut untuk nilai y maka hasilnya adalah $y = 0.5x + 1$.

Namun dikarenakan koefisien A pada $Ax + By + C = 0$ bernilai positif, maka kita perlu merubah nilai $-2x$ pada $-2x + 4y - 4 = 0$ menjadi positif, caranya dengan mengalikan (-1) . Sehingga persamaan tersebut berubah menjadi $2x - 4y + 4 = 0$ atau digambarkan seperti Gambar 2.9.



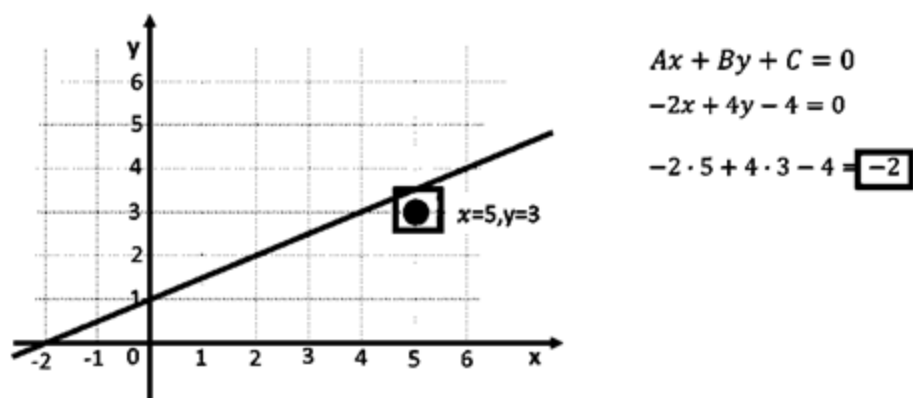
Gambar 2.9 Mengubah koefisien $-2x$ menjadi positif seperti koefisien A

Setelah proses-proses perubahan persamaan diatas kita bisa menemukan nilai positif dan negative dari suatu titik vektor, misalkan ada sebuah vektor dengan nilai $x=5$ dan $y=4$, dengan memasukan x dan y tersebut kedalam $-2x + 4y - 4 = 0$ maka hasilnya adalah $-2 \cdot 5 + 4 \cdot 4 - 4 = 2$, karena hasilnya adalah positif 2 maka titik tersebut diklasifikasikan sebagai kelas +1 seperti Gambar 2.10.



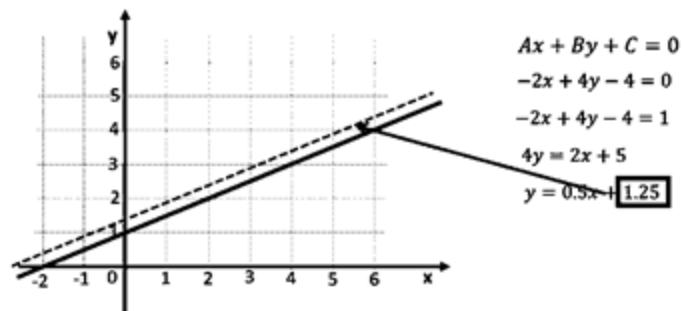
Gambar 2.10 Contoh klasifikasi setelah garis hyperplane dibuat bagian 1

Adapun misal titik vektor adalah $x=5$ dan $y=3$, dengan memasukan x dan y tersebut kedalam $-2x + 4y - 4 = 0$ maka hasilnya adalah $-2 \cdot 5 + 4 \cdot 3 - 4 = -2$, karena hasilnya adalah negatif 2 maka titik tersebut diklasifikasikan sebagai kelas -1 seperti gambar 2.11.



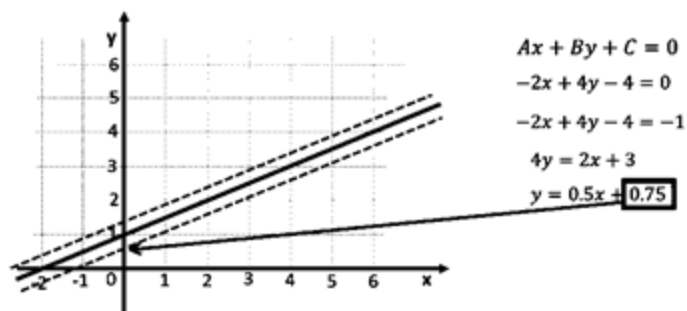
Gambar 2.11 Contoh klasifikasi setelah garis hyperplane dibuat bagian 2

Pembahasan diatas baru membahas bagaimana margin maximum ditemukan, maka selanjutnya adalah kita perlu mencari margin positif dan negatif. Untuk menemukan margin positifnya kita perlu merubah nilai 0 pada $-2x + 4y - 4 = 0$, menjadi 1 sehingga bentuk persamaannya menjadi $-2x + 4y - 4 = 1$ sehingga apabila kita mencari nilai untuk y , hasilnya adalah $y = 0.5x + 1.25$. Jika dibandingkan dengan margin maximal yang memiliki persamaan $y = 0.5x + 1$ dan persamaan yang ditemukan untuk margin positif $y = 0.5x + 1.25$, kita menemukan jika nilai yang berbeda untuk persamaan $y = mx + b$ adalah koefisien b , karena *intercept* ditentukan oleh b , maka garis memiliki kemiringan yang sama namun dengan posisi yang berbeda, atau dapat digambarkan dengan Gambar 2.12.



Gambar 2.12 Garis hyperplane positif

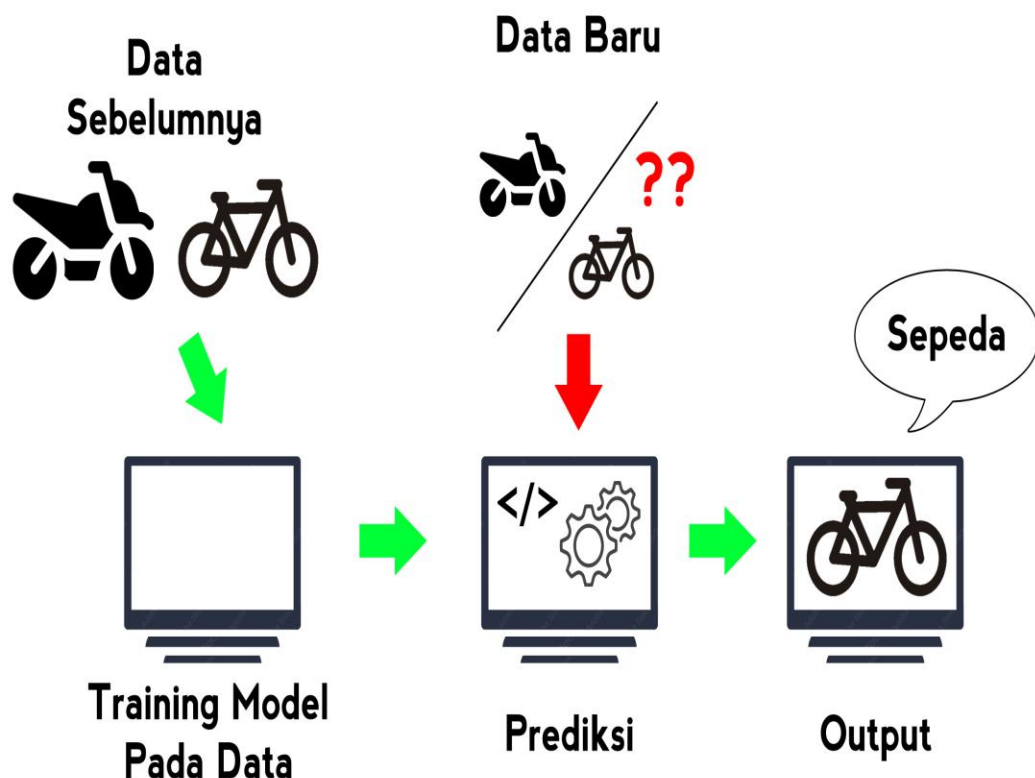
Selanjutnya kita mencari garis hyperplane negative dengan cara merubah nilai 0 pada $-2x + 4y - 4 = 0$, menjadi -1 sehingga bentuk persamaannya yaitu menjadi $-2x + 4y - 4 = -1$ sehingga hasilnya adalah $y = 0.5x + 0.75$. dan ditemukan garis hyperplane negatifnya seperti Gambar 2.13



Gambar 2.13 Garis hyperplane negatif

Selain itu, ada juga contoh dari penerapan aplikasi SVM, misalkan seseorang ingin membuat sebuah mesin klasifikasi untuk membedakan antara sepeda dan juga motor secara akurat, maka model tersebut dapat dibuat dengan menggunakan algoritma SVM. Langkah pertama yaitu melakukan pelatihan model, dalam proses *training* ini user akan memberikan model kumpulan data gambar dari motor dan sepeda, sehingga model bisa mempelajari ciri-ciri, karakteristik, dan perbedaan dari motor ataupun sepeda. Dalam proses ini, SVM akan membuat batas keputusan antara dua data, dan memilih *support vector*.

Setelah itu akan dilakukan pengujian model dengan cara memasukan salah satu data antara sepeda dan motor sehingga SVM akan melakukan prediksi berdasarkan vektor pendukung untuk melakukan proses klasifikasi pada data dan membuat prediksi apakah gambar yang dimasukan tersebut adalah sepeda atau motor. Contoh tersebut dapat digambarkan seperti Gambar 2.14



Gambar 2.14 Contoh dari sistem prediksi SVM

2.5 Website

Website atau situs web adalah kumpulan halaman digital internet yang dapat diakses secara publik dan saling terkait dengan berbagi satu nama domain. Situs web dapat dibuat dan dikelola oleh individu atau kelompok organisasi untuk memberikan layanan dengan berbagai tujuan yang beragam. Situs web pertama kali dibuat pada tahun 1990 oleh seorang yang bernama Tim Berners-Lee, ia adalah seorang fisikawan Inggris di CERN. 3 tahun kemudian, pada tahun 1993, CERN mengumumkan bahwa setiap orang dapat mengakses dan menggunakan World Wide Web secara gratis. Semua jaringan situs web yang dapat diakses oleh publik dengan skala besar membuat sebuah *World Wide Web* [11].

Situs web memiliki banyak variasi yang hampir tak ada habisnya, diantaranya seperti pendidikan, berita, forum, media sosial, *e-commerce*, dan sebagainya.

Halaman dalam situs web biasanya memiliki banyak isi berupa teks dan media seperti foto ataupun video. Situs web pada umumnya juga mempunyai beranda yang menjadi halaman bagian paling depan pada suatu situs web, halaman beranda ini memiliki *link* untuk mengakses halaman lain di situs tersebut.

Halaman web adalah file digital yang ditulis menggunakan format bahasa *HyperText Markup Language* atau disingkat HTML. Agar situs web bisa menjadi tersedia untuk setiap orang secara publik, situs web tersebut harus disimpan atau dihosting melalui sebuah komputer yang sudah terhubung ke internet sepanjang waktu dan komputer tersebut biasa disebut *web server*. Karena situs web adalah sekumpulan dari halaman web yang berasal dari *web server*.

Ada beberapa komponen penting yang membangun sebuah website diantaranya :

- *Webhost*

Webhost adalah lokasi fisik dari sebuah komputer yang menjadi webserver. Sekumpulan halaman web bisa disebut sebagai situs web hanya ketika situs web tersebut di-*hosting* pada webserver. Webserver tersebut memiliki sekumpulan file yang nantinya dikirimkan ke komputer pengguna lain saat mereka mengunjungi suatu alamat situs web.

- URL

Alamat situs web atau juga dikenal sebagai *Uniform Resource Locator* (URL) adalah alamat yang dibutuhkan seorang pengguna ketika ingin membuka sebuah situs web. Ketika alamat URL sudah dimasukkan pada kotak pencarian pada aplikasi web browser, situs web yang diminta akan langsung dikirimkan oleh server web.

- *Homepage*

Homepage atau Beranda adalah bagian yang sangat umum dan penting untuk membangun sebuah halaman web karena biasanya alaman ini muncul ketika seorang pengguna mengunjungi suatu URL.

Situs web juga memiliki domain untuk mengidentifikasi jenis layanan apa yang mereka sediakan di antaranya seperti .gov untuk pemerintahan, .edu untuk institusi pengetahuan, .org untuk organisasi non-profit, .com untuk situs komersial dan .info sebagai situs informasi. Selain itu ada juga domain yang berhubungan dengan Negara seperti .id untuk Indonesia, .uk untuk Inggris, .fr untuk perancis dan sebagainya yang diambil dari kata depan atau singkatan negara.

Dalam beberapa situs web, ada juga webserver yang memiliki *database*. Database adalah informasi yang disimpan untuk memberi akses, manajemen, dan pembaruan yang mudah. Database komputer biasanya menyimpan kumpulan catatan data atau file yang berisi informasi, seperti transaksi penjualan, data pelanggan, keuangan, dan informasi produk. Database digunakan untuk menyimpan dan mengakses segala jenis data. Selain itu database juga dianggap sebagai kumpulan informasi yang terorganisir dan mengumpulkan informasi tentang individu, tempat atau benda. Informasi tersebut dikumpulkan di satu tempat sehingga dapat diamati dan dianalisis.

2.6 MySQL

MySQL adalah aplikasi manajemen database, yang memiliki sistem *Relational Database Management System* atau disingkat dengan RDBMS dan dikembangkan oleh Oracle yang didasarkan pada bahasa *Structured Query Language* (SQL).



Gambar 2.15 Logo MySQL

MySQL bersifat open source dan sama seperti kebanyakan sistem manajemen database lainnya, MySQL memiliki arsitektur client-server dan dapat digunakan dalam pembangunan jaringan. Program server berada pada sistem fisik atau virtual yang sama di mana file database disimpan, dan bertanggung jawab atas semua interaksi pada database.

Database yaitu sekumpulan informasi yang disimpan untuk akses, manajemen dan pembaruan yang sangat mudah. Database biasanya berisi kumpulan dari banyak catatan data atau file yang berisi informasi, contohnya seperti histori pembelian atau penjualan, data dari pelanggan, catatan keuangan, dan informasi sebuah produk.

Database juga berfungsi untuk menyimpan, mengamankan, dan mengakses segala jenis data. Ada juga informasi yang dikumpulkan berupa informasi seseorang, tempat atau benda, informasi tersebut disimpan pada tempat penyimpanan yang sama sehingga dapat diamati dan juga dapat dianalisis. Sehingga database bisa disebut sebagai kumpulan data yang isinya terorganisir [12].

Database pertama kali dikembangkan pada tahun 1960. Awalnya database adalah model sebuah jaringan di yang menyimpan dua jenis data yaitu primer dan sekunder. Database pertama ini memiliki skema pohon dengan direktori akar catatan yang dihubungkan pada beberapa tempat sub-direktori. Database kemudian berkembang lagi dengan dibuatnya database relasional yang diperkenalkan pada tahun 1970. Sepuluh tahun kemudian yaitu pada tahun 1980, dikembangkan yang namanya database berorientasi objek dan terus berkembang jenis-jenis database hingga saat ini. Database yang sering digunakan pada saat ini contohnya seperti *Structured Query Language* (SQL), NoSQL, dan database cloud.

SQL atau *Structured Query Language* adalah bahasa yang digunakan untuk penggunaan database dan juga bahasa standar untuk sistem manajemen basis data relasional. SQL berperan untuk melakukan tugas seperti memperbaiki data maupun mengambil data dari dalam database. Beberapa manajemen database relasional yang menggunakan SQL diantaranya adalah Oracle, Sybase, Microsoft SQL Server, Access, dan Ingres.

Walaupun saat ini hampir semua manajemen database menggunakan SQL, sebagian besar manajemen database memiliki ekstensi fitur tambahan yang hanya digunakan pada sistem manajemen database mereka masing-masing. Perintah SQL yang sangat umum seperti *Select*, *Insert*, *Update*, *Delete*, *Create* dan *Drop* dapat digunakan untuk melakukan hampir semua hal yang perlu dilakukan saat melakukan interaksi dengan database [13]. MySQL digunakan untuk berbagai tujuan, termasuk database web, data pergudangan, *e-commerce*, database pengguna media sosial dan aplikasi *logging*.

Selain itu juga MySQL juga membantu pembuat situs web dalam pengembangan dan menghosting sistem manajemen situs web online seperti WordPress, Drupal, dan Joomla. MySQL juga mendukung banyak bahasa pemrograman seperti Java, Python, dan sebagainya.

2.7 Python

Python adalah bahasa pemrograman yang dikembangkan oleh Guido van Rossum pada tahun 1991 dan juga sangat umum digunakan untuk implementasi aplikasi maupun program.



Gambar 2.16 Logo Python

Pemrograman Python mudah dipelajari karena memiliki syntax sederhana yang meniru bahasa alami, sehingga lebih mudah untuk dibaca dan dipahami. Python dapat digunakan untuk berbagai tugas, mulai dari pengembangan web, aplikasi database hingga *Machine Learning*. Python menerjemahkan setiap kode sumber menjadi instruksi kode byte dengan menguraikannya menjadi langkah-langkah individual. Menerjemahkan kode byte dilakukan untuk tujuan mempercepat eksekusi. Kode byte dapat dijalankan jauh lebih cepat daripada kode sumber asli.

Bahasa pemrograman Python juga sangat umum digunakan dalam pengembangan kecerdasan buatan atau *artificial intelligence* (AI), karena pemrograman untuk mengembangkan sebuah AI sangatlah berbeda dibandingkan proyek perangkat lunak sederhana.

Perbedaan dari pengembangan AI terletak pada teknologi, keterampilan yang diperlukan, dan keperluan penelitian yang mendalam. Untuk mengimplementasikan kecerdasan buatan diperlukan bahasa pemrograman yang stabil dan fleksibel, dan Python adalah salah satu bahasa pemrograman yang memiliki kelebihan tersebut [14].

Salah satu kemampuan Python juga dapat dilihat dari ekstensinya. Python memiliki sejumlah besar modul yang dapat diinstal dan digunakan dengan cepat. Modul-modul tersebut mencakup setiap aspek pemrograman, mulai dari akses data hingga penerapan algoritma umum. Selain itu Python bersifat open source dan memiliki komunitas developer yang cukup besar.

Python bisa berjalan pada sistem operasi yang berbeda seperti Windows, Mac, Linux, Raspberry Pi. Selain itu bahasa pemrograman ini menggunakan teknik baris baru untuk menyelesaikan sebuah perintah program dan berbeda dengan bahasa pemrograman lain yang sering menggunakan titik koma atau tanda kurung.