**PENDEKATAN *SUPERVISED LEARNING* UNTUK DIAGNOSA KEHAMILAN**

**PENDEKATAN SUPERVISED LEARNING UNTUK DIAGNOSA KEHAMILAN**

**RONI HABIBI, S.KOM., M.T., SFPC**

**FAHIRA**

**ZIAN ASTI DWIYANTI**

**CANTUMKAN**

**Logo Penerbit Disini**

*“Orang yang tak pernah melakukan kesalahan adalah orang yang tak pernah mencoba sesuatu yang baru”*

*Albert Einstein.*

**PENDEKATAN SUPERVISED LEARNING UNTUK DIAGNOSA KEHAMILAN**

©TULISKAN NAMA PENERBIT DISINI

Penulis:

……..

Editor:

(Nama Penguji Sidang)

Cetakan Pertama: Isi dengan Bulan saat upload buku

Cover: Tim Penyusun

Tata Letak: Tim Kreatif Penerbit

Hak Cipta 2023, pada Penulis. Diterbitkan pertama kali oleh:

**ISI NAMA PENERBIT**

ISI ALAMAT PENERBIT

Website: [WEBSITE](http://www.rcipress.rcipublisher.org/) PENERBIT

E-mail: [EMAIL](mailto:rumahcemerlangindonesia@gmail.com) PENERBIT

Copyright © 2023 by NAMA PENERBIT

All Right Reserved

- Cet. I –: NAMA PENERBIT, TAHUN TERBIT

Dimensi : 14,8 x 21 cm

ISBN: KOSONGKAN DULU

Hak cipta dilindungi undang-undang

Dilarang memperbanyak buku ini dalam bentuk dan dengan cara apapun tanpa izin tertulis dari penulis dan penerbit

Undang-undang No.19 Tahun 2002 Tentang

**Hak Cipta Pasal 72**

Undang-undang No.19 Tahun 2002 Tentang Hak Cipta Pasal 72

Barang siapa dengan sengaja melanggar dan tanpa hak melakukan perbuatan sebagaimana dimaksud dalam pasal ayat (1) atau pasal 49 ayat (1) dan ayat (2) dipidana dengan pidana penjara masing-masing paling sedikit 1 (satu) bulan dan/atau denda paling sedikit Rp.1.000.000,00 (satu juta rupiah), atau pidana penjara paling lama 7 (tujuh) tahun dan/atau denda paling banyak Rp.5.000.000.000,00 (lima miliar rupiah).

Barang siapa dengan sengaja menyiarkan, memamerkan, mengedarkan, atau menjual kepada umum suatu ciptaan atau barang hasil pelanggaran hak cipta terkait sebagai dimaksud pada ayat (1) dipidana dengan pidana penjara paling lama 5 (lima) tahun dan/atau denda paling banyak Rp.500.000.000,00 (lima ratus juta rupiah).

# KATA PENGANTAR

Assalamu’alaikum Warahmatullahi Wabarakatuh. Alhamdulillah Puji syukur penulis panjatkan ke hadirat Allah SWT karena hanya dengan rahmat dan hidayahnya, buku ini dapat terselesaikan tanpa halangan berarti. Keberhasilan dalam menyusun buku ini tidak lepas dari bantuan semua pihak yang tulus dan ikhlas memberikan masukan guna sempurnanya buku ini, khususnya terima kasih kepada Bapak Roni Habibi, S.Kom., M.T., SFPC. Selaku dosen pembimbing yang telah memberikan saran serta ide. Tidak lupa penulis berterima kasih kepada Kedua Orang Tua yang selalu memberi dukungan dan doa.

Buku ini merupakan hasil penelitian untuk diagnosa kehamilan menggunakan pendekatan supervised learning. Menjadi ibu bukanlah hal yang mudah karena harus ada kehamilan dan persalinan yang ekstensif. Kehamilan adalah proses sembilan bulan atau lebih dimana harus memperhatikan kesehatan kandungan sedini mungkin. Bahkan hamil di usia muda lebih berisiko mengalami masalah kehamilan. Dalam bidang kesehatan telah dikembangkan suatu kecerdasan yang disebut machine learning yaitu suatu sistem komputer yang menggunakan pengetahuan, fakta dan metode untuk memecahkan masalah yang seringkali tidak dipahami oleh seorang ahli di bidangnya. Penelitian ini nantinya akan menggunakan metode pendekatan supervised learning yang akan melakukan pencarian mendalam berdasarkan pola data yang ada. Tujuan dari penelitian ini adalah untuk membantu ibu melakukan diagnosis kehamilan yang diprediksi normal selama kehamilan tanpa harus ke dokter. Data yang digunakan adalah usia, usia kehamilan, berat dan tinggi badan, sistole, diastole, IMT, lingkar perut dan variabel targetnya adalah diagnosis Normal atau tidak. Hasil dari penelitian ini akan menghasilkan suatu model yang akan diterapkan ke dalam sistem informasi klinik untuk mendiagnosa penyakit kehamilan.

Kami menyadari bahwa penyusunan buku ini jauh dari sempurna. Dengan diselesaikannya buku ini, penulis berharap semoga buku ini dapat bermanfaat bagi seluruh pembaca. Semoga kedepannya buku ini akan terus dikembangkan. Wassalamu’alaikum Warahmatullahi Wabarakatuh.

Bandung, 01 Desember 2022

**Penulis**

# DAFTAR ISI

BUATLAH DAFTAR ISI DISINI

# DAFTAR GAMBAR

BUATLAH DAFTAR ISI DISINI

**BAB 1**

# Pengantar *Supervised Learning*

*Machine learning* dapat diartikan sebuah computer yang memiliki kemampuan belajar tanpa deprogram secara explisit. Program tersebut memanfaatkan data untuk membangun model dan mengambil keputusan berdasarkan model yang telah dibangun. *Machine learning* dapat dikelompokkan berdasarkan bagaimana cara belajar. Sehingga dapat melakukan tugasnya. Pembagian *machine learning* berdasarkan cara belajarnya dibagi menjadi tiga kelompok yaitu *supervised learning*, *unsupervised learning*, dan *reinforcement learning*.

1. **Definisi *Supervised Learning***

Secara bahasa, *supervised learning* merupakan pembelajaran terarah/terawasi. apabila kita analogikan dalam proses pembelajaran, komputer atau mesin akan mempelajari data *training* yang berisi label. apabila dianalogikan ke murid & pengajar, dimana komputer menjadi murid yang belajar maka pengajar akan meminta murid buat belajar berdasarkan soal yang telah mempunyai solusi & kunci jawaban. Pada dasarnya saat melakukan proses pembelajaran, sistem diberikan data yang berisi solusi atau output yang akan diinginkan atau kasus yang akan diselesaikan oleh sistem.

Selain itu supervised learning juga bisa diartikan sebagai pengklasifikasian. Adapun contoh-contoh algoritma yang termasuk ke dalam kategori supervised learning adalah *Linear Regression* (Regresi Linear), *Logistic Regression* (Regresi Logistik), *Linear Discriminant Analysis, k-Nearest Neighbors, Support Verctor Machines* (SVMs), *Random Forest, Decision Tree,* dan *Naïve Bayes.*

Langkah pertama pada proses *supervised learning* merupakan mengumpulkan data pelatihan berlabel. Label merupakan keluaran & memberikan umpan balik buat algoritma. Asalkan tersedia data yang cukup, langkah selanjutnya adalah membagi data berlabel ini menjadi 3 set: pelatihan, pengujian, & validasi. Algoritma memakai set pelatihan buat menyesuaikan model buat meminimalkan kesalahan. Misalnya set pelatihan mungkin berisi banyak sekali gambar fauna dengan label yang terkait dengan setiap gambar, memungkinkan algoritma untuk membandingkan label yg diprediksi menggunakan label yg benar.

Set validasi terpisah dari set pelatihan & memungkinkan seseorang mengukur kemajuan algoritma pembelajaran secara mandiri. Ukuran ini bisa dipakai untuk memilih titik potong pada algoritma pelatihan untuk menyeimbangkan akurasi model yang dipelajari versus overfitting.

Set tes merupakan set terakhir & dimaksudkan untuk dipakai hanya saat model sudah ditemukan optimal dalam set validasi. Kumpulan ini memberikan evaluasi ‘dunia nyata' dari kinerja model dalam data yang belum pernah dicermati sebelumnya. Data uji adalah semacam 'ujian akhir' untuk model yg sudah mempelajari data pelatihannya secara efektif & bisa digeneralisasikan ke data baru.

1. **Algoritma *Supervised Learning***

Berbagai algoritma dan teknik komputasi digunakan dalam proses *supervised machine learning*. Di bawah ini adalah penjelasan dari beberapa metode pembelajaran yang paling umum digunakan, biasanya dihitung melalui penggunaan program seperti R atau Python:

* 1. ***Decision Tree***

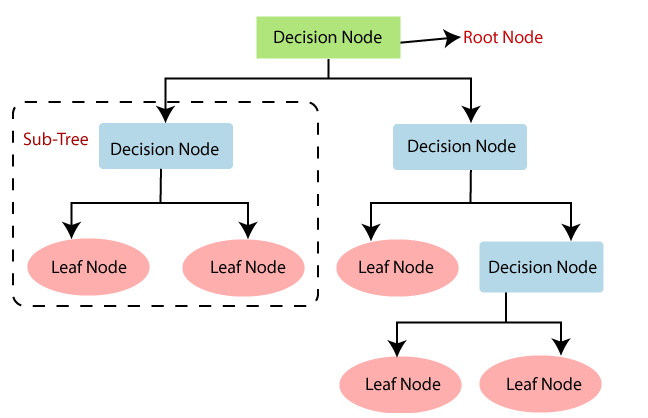
*Decision Tree*  adalah salah satu algoritma yang paling bagus, karena berbagai fitur yang menarik, sederhana, lengkap, tanpa parameter, dan mampu menangani tipe data campuran. Decision Tree adalah teknik supervised learning yang dapat digunakan untuk masalah klasifikasi dan Regresi, tetapi sebagian besar lebih banyak digunakan untuk menyelesaikan masalah Klasifikasi. Ini adalah pengklasifikasi berstruktur pohon, dimana simpul internal mewakili fitur kumpulan data, cabang mewakili aturan keputusan dan setiap simpul daun mewakili hasilnya.

Pada pohon keputusan terdapat dua simpul yaitu simpul keputusan dan simpul daun. Simpul keputusan digunakan untuk membuat keputusan apa pun dan memiliki banyak cabang, sedangkan simpul Daun adalah keluaran dari keputusan tersebut dan tidak mengandung cabang lebih lanjut.

Keputusan atau pengujian dilakukan berdasarkan fitur dari kumpulan data yang diberikan. Ini adalah representasi grafis untuk mendapatkan semua kemungkinan solusi untuk masalah/keputusan berdasarkan kondisi yang diberikan.

Ini disebut pohon keputusan karena, mirip dengan pohon, dimulai dengan simpul akar, yang meluas ke cabang lebih lanjut dan membangun struktur seperti pohon.

Sebuah pohon keputusan hanya mengajukan sebuah pertanyaan, dan berdasarkan jawaban (Ya/Tidak), selanjutnya membagi pohon menjadi subpohon. Diagram di bawah menjelaskan struktur umum pohon keputusan:



Dalam pohon keputusan, untuk memprediksi kelas dari himpunan data tertentu, algoritma dimulai dari simpul akar pohon. Algoritma ini membandingkan nilai atribut root dengan atribut record (dataset nyata) berdasarkan perbandingan, mengikuti cabang dan melompat ke node berikutnya.

Untuk node berikutnya, algoritma kembali membandingkan nilai atribut dengan sub-node lain dan melanjutkan prosesnya. Proses berlanjut hingga mencapai simpul daun pohon.

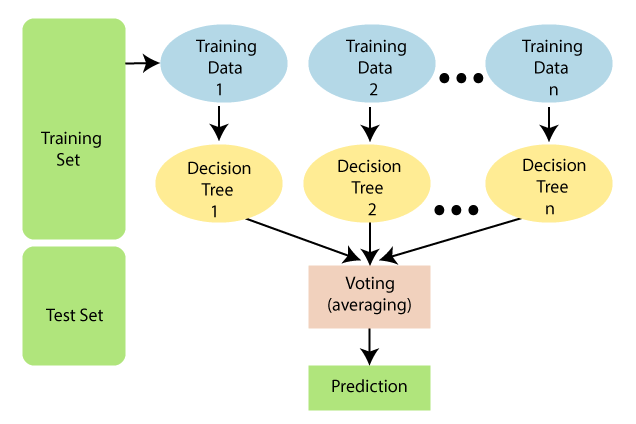
Kelebihan dari algoritma *decision tree* adalah *decision tree* mampu menghasilkan aturan yang dapat dimengerti, *decision tree* melakukan klasifikasi tanpa memerlukan banyak perhitungan, *decision tree* mampu menangani variabel kontinyu dan kategoris dan *decision tree* memberikan indikasi yang jelas bidang mana yang paling penting untuk prediksi atau klasifikasi.

Sedangkan kelemahan dari *decision tree* adalah *decision tree* kurang sesuai untuk tugas estimasi di mana tujuannya adalah untuk memprediksi nilai atribut kontinu, *decision tree* rentan terhadap kesalahan dalam masalah klasifikasi dengan banyak kelas dan jumlah contoh pelatihan yang relatif kecil.

* 1. ***Random Forest***

*Random Forest* adalah algoritma *machine learning* populer yang termasuk dalam teknik *supervised learning*. Algoritma ini dapat digunakan untuk masalah Klasifikasi dan Regresi di ML. Ini didasarkan pada konsep pembelajaran ensemble, yang merupakan proses menggabungkan beberapa pengklasifikasi untuk memecahkan masalah yang kompleks dan untuk meningkatkan kinerja model.

Seperti namanya, "*Random Forest* (Hutan Acak) adalah pengklasifikasi yang berisi sejumlah pohon keputusan pada berbagai himpunan bagian dari kumpulan data yang diberikan dan mengambil rata-rata untuk meningkatkan akurasi prediktif dari kumpulan data tersebut." Alih-alih mengandalkan satu pohon keputusan, hutan acak mengambil prediksi dari setiap pohon dan berdasarkan suara mayoritas prediksi, dan memprediksi hasil akhir. Semakin banyak jumlah pohon di hutan mengarah pada akurasi yang lebih tinggi dan mencegah masalah overfitting.



Karena hutan acak menggabungkan banyak pohon untuk memprediksi kelas kumpulan data, ada kemungkinan bahwa beberapa pohon keputusan dapat memprediksi keluaran yang benar, sementara yang lain mungkin tidak. Tapi, semua pohon memprediksi hasil yang benar. Oleh karena itu, di bawah ini adalah dua asumsi untuk pengklasifikasi hutan Acak yang lebih baik:

* Harus ada beberapa nilai aktual dalam variabel fitur dari kumpulan data sehingga pengklasifikasi dapat memprediksi hasil yang akurat daripada hasil yang ditebak.
* Prediksi dari setiap pohon harus memiliki korelasi yang sangat rendah.

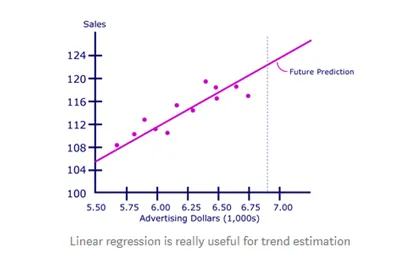
Terdapat beberapa perbedaan antara *random forest* dengan *decision tree* yaitu, jika kita memasukkan dataset latih dengan fitur dan label ke dalam pohon keputusan, itu akan merumuskan beberapa aturan, yang akan digunakan untuk membuat prediksi.

Misalnya, untuk memprediksi apakah seseorang akan mengklik iklan online, Anda dapat mengumpulkan iklan yang diklik orang tersebut di masa lalu dan beberapa fitur yang menggambarkan keputusan mereka. Sebagai perbandingan, algoritma random forest secara acak memilih pengamatan dan fitur untuk membangun beberapa pohon keputusan dan kemudian hasilnya berupa rata-rata yang akan digunakan.

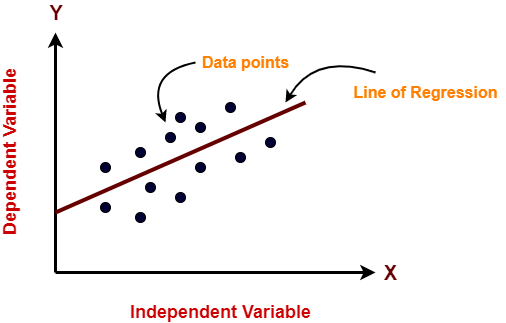
Perbedaan lainnya adalah pohon keputusan mungkin mengalami overfitting. Sebagian besar waktu, hutan acak mencegah hal ini dengan membuat subset acak dari fitur dan membangun pohon yang lebih kecil menggunakan subset tersebut. Setelah itu, menggabungkan subtree. Penting untuk dicatat bahwa ini tidak berfungsi setiap saat dan juga memperlambat perhitungan, tergantung pada berapa banyak pohon yang dibangun oleh hutan acak.

* 1. ***Linear Regression* (Regresi Linear)**

Regresi linier digunakan untuk mengidentifikasi hubungan antara variabel dependen dan satu atau lebih variabel independen dan biasanya dimanfaatkan untuk membuat prediksi tentang hasil masa depan. Ketika hanya ada satu variabel independen dan satu variabel dependen, itu dikenal sebagai regresi linier sederhana. Ketika jumlah variabel independen meningkat, ini disebut sebagai regresi linier berganda. Untuk setiap jenis regresi linier, ia mencari garis yang paling cocok, yang dihitung melalui metode kuadrat terkecil. Namun, tidak seperti model regresi lainnya, garis ini lurus saat diplot pada grafik.



Teknik regresi ini menemukan hubungan linier antara variabel dependen dan variabel independen lainnya yang diberikan. Oleh karena itu, nama algoritma ini adalah Regresi Linier.



Pada gambar di atas, pada sumbu X adalah variabel bebas dan pada sumbu Y adalah keluarannya. Garis regresi adalah garis yang paling cocok untuk sebuah model. Dan tujuan utama dalam algoritma ini adalah menemukan garis yang paling cocok.

Kelebihan dari Regresi Linear ini adalah Regresi Linier mudah diterapkan, lebih sedikit kompleksitas dibandingkan dengan algoritma lain dan regresi linear dapat menyebabkan over-fitting tetapi dapat dihindari dengan menggunakan beberapa teknik reduksi dimensi, teknik regularisasi, dan validasi silang.

Sedangkan kekurangannya yaitu terdapat outlier yang mempengaruhi algoritma ini dengan buruk dan itu terlalu menyederhanakan masalah dunia nyata dengan mengasumsikan hubungan linier antara variabel, karenanya tidak direkomendasikan untuk kasus penggunaan praktis.

* 1. ***Logistic Regression* (Regresi Logistik)**

Regresi logistik adalah salah satu algoritma *Machine Learning* yang paling populer, yang berada di bawah teknik *Supervised Learning*. Ini digunakan untuk memprediksi variabel dependen kategoris menggunakan seperangkat variabel independen yang diberikan.

Regresi logistik memprediksi output dari variabel dependen kategori. Oleh karena itu hasilnya harus menjadi nilai kategoris atau diskrit. Itu bisa berupa Ya atau Tidak, 0 atau 1, benar atau Salah, dll. tetapi alih-alih memberikan nilai pasti sebagai 0 dan 1, ini memberikan nilai probabilistik yang terletak antara 0 dan 1.

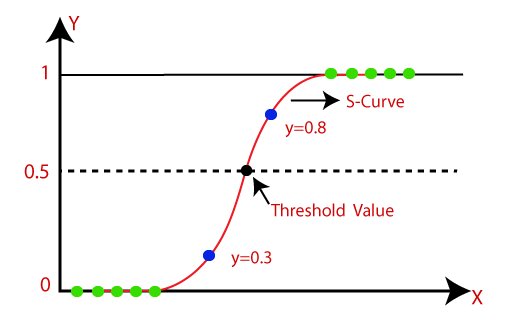
Regresi Logistik sangat mirip dengan Regresi Linier kecuali cara penggunaannya. Regresi Linier digunakan untuk menyelesaikan masalah Regresi, sedangkan regresi Logistik digunakan untuk menyelesaikan masalah klasifikasi.

Dalam regresi Logistik, alih-alih memasang garis regresi, kami memasukkan fungsi logistik berbentuk "S", yang memprediksi dua nilai maksimum (0 atau 1).

Kurva dari fungsi logistik menunjukkan kemungkinan sesuatu seperti apakah sel-sel itu kanker atau tidak, seekor tikus gemuk atau tidak berdasarkan beratnya, dll.

Regresi Logistik adalah algoritma pembelajaran mesin yang signifikan karena memiliki kemampuan untuk memberikan probabilitas dan mengklasifikasikan data baru menggunakan dataset kontinu dan diskrit.

Regresi Logistik dapat digunakan untuk mengklasifikasikan pengamatan menggunakan berbagai jenis data dan dapat dengan mudah menentukan variabel yang paling efektif digunakan untuk klasifikasi. Gambar di bawah ini menunjukkan fungsi logistic :



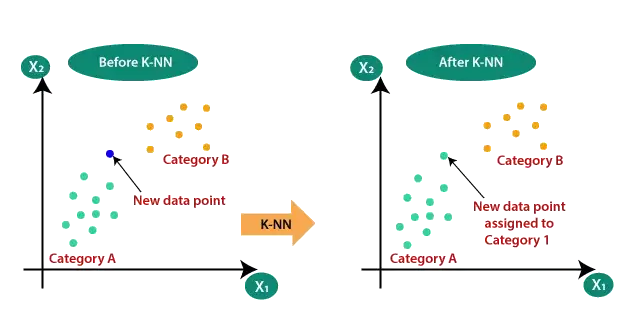
Berdasarkan kategorinya, Regresi Logistik dapat diklasifikasikan menjadi tiga jenis, yaitu:

* **Binomial**: Dalam regresi Logistik binomial, hanya ada dua kemungkinan jenis variabel dependen, seperti 0 atau 1, Lulus atau Gagal, dll.
* **Multinomial**: Dalam regresi Logistik multinomial, mungkin ada 3 atau lebih jenis variabel dependen yang tidak terurut, seperti "kucing", "anjing", atau "domba"
* **Ordinal**: Dalam regresi Logistik ordinal, mungkin ada 3 atau lebih jenis variabel dependen yang mungkin diurutkan, seperti "rendah", "Sedang", atau "Tinggi".
  1. ***k-Nearest Neighbors***

Algoritma K-Nearest Neighbors (KNN) adalah metode klasifikasi data untuk memperkirakan kemungkinan bahwa suatu titik data akan menjadi anggota dari satu grup atau lainnya berdasarkan grup apa yang dimiliki oleh titik data terdekat.

KNN mencoba memprediksi kelas yang benar untuk data uji dengan menghitung jarak antara data uji dan semua titik latih. Kemudian pilih K jumlah titik yang paling dekat dengan data uji. Algoritma KNN menghitung probabilitas data uji yang termasuk dalam kelas data latih 'K' dan kelas yang memiliki probabilitas tertinggi akan dipilih. Dalam kasus regresi, nilainya adalah rata-rata dari poin pelatihan yang dipilih 'K'.

Misalkan ada dua kategori, yaitu Kategori A dan Kategori B, dan kita memiliki titik data baru x1, maka titik data ini akan terletak pada kategori yang mana. Untuk mengatasi jenis masalah ini, kita membutuhkan algoritma K-NN. Dengan bantuan K-NN, kita dapat dengan mudah mengidentifikasi kategori atau kelas dari dataset tertentu. Pertimbangkan diagram di bawah ini:



KNN adalah salah satu algoritma tertua namun akurat yang digunakan untuk klasifikasi pola dan model regresi. Berikut adalah beberapa area di mana algoritma *k-nearest neighbor* dapat digunakan:

* Peringkat kredit: Algoritma KNN membantu menentukan peringkat kredit individu dengan membandingkannya dengan yang memiliki karakteristik serupa.
* Persetujuan pinjaman: Mirip dengan peringkat kredit, algoritma *k-nearest neighbor* bermanfaat dalam mengidentifikasi individu yang lebih cenderung gagal bayar dengan membandingkan sifat mereka dengan individu serupa.
* Pemrosesan awal data: Kumpulan data dapat memiliki banyak nilai yang hilang. Algoritma KNN digunakan untuk proses yang disebut imputasi data yang hilang dengan memperkirakan nilai yang hilang.
* Pengenalan pola: Kemampuan algoritma KNN untuk mengidentifikasi pola menciptakan berbagai macam aplikasi. Misalnya, ini membantu mendeteksi pola penggunaan kartu kredit dan menemukan pola yang tidak biasa. Deteksi pola juga berguna dalam mengidentifikasi pola perilaku pembelian pelanggan.
* Prediksi harga saham: Karena algoritma KNN memiliki bakat untuk memprediksi nilai entitas yang tidak diketahui, ini berguna dalam memprediksi nilai saham di masa mendatang berdasarkan data historis.
* Sistem rekomendasi: Karena KNN dapat membantu menemukan pengguna dengan karakteristik serupa, KNN dapat digunakan dalam sistem rekomendasi. Misalnya, ini dapat digunakan dalam platform streaming video online untuk menyarankan konten yang kemungkinan besar akan ditonton pengguna dengan menganalisis apa yang ditonton oleh pengguna serupa.
* *Computer vision*: Algoritma KNN digunakan untuk klasifikasi gambar. Karena mampu mengelompokkan titik data yang serupa, misalnya mengelompokkan kucing dan anjing dalam kelas yang berbeda, ini berguna dalam beberapa aplikasi *computer vision*.
  1. ***Support Vector Machines (SVMs)***

*Support Vector MAchines* adalah seperangkat metode *supervised learning* yang digunakan untuk klasifikasi, regresi, dan deteksi outlier. Kita dapat menggunakannya untuk mendeteksi sel kanker berdasarkan jutaan gambar atau Anda dapat menggunakannya untuk memprediksi rute mengemudi di masa depan dengan model regresi yang sesuai.

Ada jenis SVM tertentu yang dapat kita gunakan untuk masalah *machine learning* tertentu, seperti *Support Vector Regression* (SVR) yang merupakan perpanjangan dari Support Vector Classification (SVC).

Hal utama yang perlu diingat di sini adalah bahwa SVM ini hanyalah persamaan matematika yang disetel untuk memberi kita jawaban seakurat mungkin dan secepat mungkin.

SVM berbeda dari algoritma klasifikasi lainnya karena cara mereka memilih batas keputusan yang memaksimalkan jarak dari titik data terdekat dari semua kelas. Batas keputusan yang dibuat oleh SVM disebut *maximum margin classifier* atau *maximum margin hyperplane*.

SVM digunakan dalam aplikasi seperti pengenalan tulisan tangan, deteksi intrusi, deteksi wajah, klasifikasi email, klasifikasi gen, dan di halaman web. Inilah salah satu alasan kita dapat menggunakan SVM dalam *machine learning*. Itu dapat menangani klasifikasi dan regresi pada data linier dan non-linier.

Alasan lain menggunakan SVM adalah karena SVM dapat menemukan hubungan kompleks antara data kita tanpa kita perlu melakukan banyak transformasi sendiri. Ini adalah opsi yang bagus ketika kita bekerja dengan kumpulan data yang lebih kecil yang memiliki puluhan hingga ratusan ribu fitur. SVM biasanya menemukan hasil yang lebih akurat jika dibandingkan dengan algoritma lain karena kemampuannya menangani kumpulan data yang kecil dan kompleks.

Berikut adalah beberapa kelebihan dan kekurangan untuk menggunakan SVM.

**Kelebihan** :

* Efektif pada kumpulan data dengan banyak fitur, seperti data keuangan atau medis.
* Efektif dalam kasus di mana jumlah fitur lebih besar dari jumlah titik data.
* Menggunakan subkumpulan titik pelatihan dalam fungsi keputusan yang disebut vektor dukungan yang menjadikannya hemat memori.
* Fungsi kernel yang berbeda dapat ditentukan untuk fungsi keputusan. Anda dapat menggunakan kernel umum, tetapi juga memungkinkan untuk menentukan kernel kustom.

**Kekurangan** :

* Jika jumlah fitur jauh lebih besar daripada jumlah titik data, menghindari pemasangan yang berlebihan saat memilih fungsi kernel dan istilah regularisasi sangatlah penting.
* SVM tidak secara langsung memberikan perkiraan probabilitas.
* Berfungsi paling baik pada kumpulan sampel kecil karena waktu pelatihannya yang tinggi.
  1. ***Naïve Bayes***

Algoritma Naïve Bayes terdiri dari dua kata Naïve dan Bayes, yang dapat digambarkan sebagai:

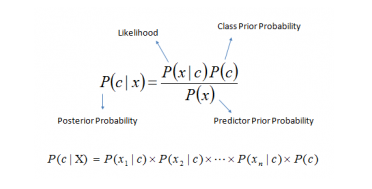
* Naïve: Disebut Naïve karena mengasumsikan bahwa kemunculan fitur tertentu tidak bergantung pada kemunculan fitur lainnya. Seperti jika buah diidentifikasi berdasarkan warna, bentuk, dan rasa, maka buah merah, bulat, dan manis dikenali sebagai buah apel. Karenanya setiap fitur secara individual berkontribusi untuk mengidentifikasi bahwa itu adalah sebuah apel tanpa saling bergantung satu sama lain.
* Bayes: Disebut Bayes karena bergantung pada prinsip Teorema Bayes.

*Naive Bayes* ini adalah teknik klasifikasi berdasarkan Teorema Bayes dengan asumsi independensi antar prediktor. Secara sederhana, pengklasifikasi Naive Bayes mengasumsikan bahwa keberadaan fitur tertentu di kelas tidak terkait dengan keberadaan fitur lainnya.

Misalnya, buah dapat dianggap sebagai apel jika berwarna merah, bulat, dan berdiameter sekitar 3 inci. Bahkan jika ciri-ciri ini bergantung satu sama lain atau pada keberadaan ciri-ciri lain, semua sifat ini secara mandiri berkontribusi pada kemungkinan bahwa buah ini adalah apel dan itulah sebabnya ia dikenal sebagai 'Naif'.

Model Naive Bayes mudah dibuat dan sangat berguna untuk kumpulan data yang sangat besar. Seiring dengan kesederhanaannya, Naive Bayes dikenal mengungguli bahkan metode klasifikasi yang sangat canggih.

Teorema Bayes memberikan cara menghitung probabilitas posterior P(c|x) dari P(c), P(x) dan P(x|c). Perhatikan persamaan di bawah ini:



**BAB 2**

# Pengantar Kehamilan

Sesuaikan jenis font dan paragraph dengan template ini. Jangan diubah-ubah, anda hanya tinggal menggunakannya

**BAB 3**

# Tinjauan Pustaka

Sesuaikan jenis font dan paragraph dengan template ini. Jangan diubah-ubah, anda hanya tinggal menggunakannya

**BAB 4**

# Bermain dengan *Python*

Sesuaikan jenis font dan paragraph dengan template ini. Jangan diubah-ubah, anda hanya tinggal menggunakannya

# DAFTAR PUSTAKA

Blogs.reed.edu. 2017. Text analysis using Voyant Tools. [online] Available at: <https://blogs.reed.edu/ed- tech/2017/03/text-analysis-using-voyant-tools/> [Accessed 11 September 2022].

Calado, F., 2018. Using Voyant-Tools to Formulate Research Questions for Textual Data - GC Digital Fellows. [online] GC Digital Fellows. Available at:

<https://digitalfellows.commons.gc.cuny.edu/2018/11/01

/using-voyant-tools-to-formulate-research-questions-for- textual- data/#:~:text=One%20of%20the%20benefits%20of,text

%20is%20displayed%20for%20reading.> [Accessed 11

August 2022].

En.wikipedia.org. 2022. Voyant Tools - Wikipedia. [online] Available at:

<https://en.wikipedia.org/wiki/Voyant\_Tools#:~:text=Ran ge%20of%20Uses-

,History,Lisa%20Goddard%2C%20and%20Mark%20Turc ato.> [Accessed 10 August 2022].

GitHub. 2016. *GitHub - voyanttools/Voyant*. [online] Available at:

<https://github.com/voyanttools/Voyant> [Accessed 10

August 2022].

-oo00oo-

DIHALAMAN INI, SILAHKAN ANDA ISI DENGAN PROFIL PENULIS, BAIK ITU PEMBIMBING INTERNSHP 1 DAN MAHASISWA

BUAT COVER BELAKANG BUKU

YANG BERISI :

SINOPSIS DARI BUKU YANG ANDA TULIS, SINOPSIS DAPAT DIAMBIL DARI ABSTRAK PADA JURNAL YANG ANDA BUAT

CATATAN :

DESAIN HARUS ORIGINAL