BAB II
TINJUAN PUSTAKA

2.1 Penelitian Terkait

Beberapa penelitian terkait dengan menggunakan teknik Data Mining metode klasifikasi untuk memprediksi penyakit jantung dapat dilihat pada tabel 2.1 dibawah ini :

Table 2.1 Berbagai Teknik Data Mining yang Digunakan dalam Prediksi Penyakit Jantung

PENULIS	TAHUN	TOOLS	METODE	AKURASI
D Derisma	2020	Orange	Naive Bayes,	83%
			Random Forest,	82%
			Neural Network	81%
Sayali Ambekar, Rashmi Phalnikar	2018	Python	KNN	81%
			Naïve Bayes	82%
Md. Touhidul Islam et al	2020	Python	Hybrid Genetic Algorithm dengan k-means clustering	94,06%
Rohit Bharti et al	2021	Python	Random Forest	88%
			Logistic Regression	85,9%
			KNeighbors	79,69%
			Support Vector Machine	84,26%
			Decision Tree	76,35%
			XGBoost	71,1%
Dengqing Zhang et al	2021	Python	LinearSVC algorithm	98,6%
Xiao-Yan Gao et al	2021	Python	KNN	98,1%
			Naïve Bayes	86,7%
			Dessison Tree	98,1%
Yongjie Yuan et al	2021	Python	Convolutional Neural Network	98,5%
			Multimodal Neural Network	98%

Penelitian oleh D Derisma pada tahun 2020, Penelitian ini membahas Perbandingan Kinerja Algoritma untuk Prediksi Penyakit Jantung dengan Teknik Data Mining. Untuk mendapatkan algoritma klasifikasi yang terbaik adalah dengan membandingkan tiga algoritma Naive Bayes, Random Forest, Neural Network yang

sering digunakan untuk memprediksi penderita penyakit jantung. Hasil perbandingan menunjukkan bahwa algoritma Naive Bayes merupakan algoritma yang tepat dan akurat yang digunakan untuk memprediksi penderita penyakit jantung dengan persentase sebesar 83%.

Berikutya penelitian oleh Sayali Ambekar dan Rashmi Phalnikar tahun 2018, Penelitian ini didasarkan pada penanganan dan asimilasi sejumlah besar data rumah sakit, bukan prediksi. Karena sejumlah besar pertumbuhan data di bidang biomedis dan perawatan kesehatan, analisis data medis yang akurat menjadi menguntungkan untuk deteksi dini penyakit dan perawatan pasien. Namun, akurasi menurun ketika data medis sebagian hilang. Untuk mengatasi masalah hilangnya data medis, kami melakukan pembersihan data dan imputasi untuk mengubah data yang tidak lengkap menjadi data yang lengkap. Kami sedang mengerjakan prediksi penyakit jantung berdasarkan kumpulan data dengan bantuan algoritma Naïve bayes dan KNN. Untuk memperluas pekerjaan ini, kami mengusulkan prediksi risiko penyakit menggunakan data terstruktur. Kami menggunakan algoritma prediksi risiko penyakit unimodel berbasis jaringan saraf convolutional. Akurasi prediksi algoritma CNN-UDRP mencapai lebih dari 65%. Selain itu, sistem ini menjawab pertanyaan terkait penyakit yang dihadapi manusia dalam hidupnya.

Selanjutnya Penelitian oleh *Md. Touhidul Islam et al* tahun 2020, dalam penelitian ini Principal Component Analysis (PCA) telah digunakan untuk mereduksi atribut. Terlepas dari algoritma genetika Hybrid (HGA) dengan k-means yang digunakan untuk pengelompokan akhir. Biasanya, metode k-means digunakan untuk mengelompokkan data. Clustering jenis ini dapat terjebak pada local optima karena metode ini bersifat heuristik. Kami menggunakan Algoritma Genetika Hibrid (HGA) untuk pengelompokan data untuk menghindari masalah ini. Metode yang kami usulkan dapat memprediksi penyakit jantung dini dengan akurasi 94,06%.

Selanjutnya Penelitian oleh *Rohit Bharti et al* tahun 2021 dalam penelitian ini mengusulkan tiga metode dimana analisis komparatif dilakukan dan hasil yang menjanjikan dicapai. Banyak peneliti sebelumnya menyarankan agar kita menggunakan ML di mana dataset tidak terlalu besar, yang dibuktikan dalam makalah ini. Metode yang digunakan untuk perbandingan adalah matriks konfusi,

presisi, spesifisitas, sensitivitas, dan skor F1. Untuk 13 fitur yang ada di dataset, classifier KNeighbors tampil lebih baik dalam pendekatan ML saat prapemrosesan data diterapkan. Waktu komputasi juga berkurang yang berguna saat menerapkan model. Ditemukan juga bahwa dataset harus dinormalisasi; jika tidak, model pelatihan terkadang menjadi terlalu pas dan akurasi yang dicapai tidak cukup ketika model dievaluasi untuk masalah data dunia nyata yang dapat bervariasi secara drastis dengan kumpulan data tempat model dilatih.

Selanjutnya Penelitian oleh *Dengqing Zhang et al* tahun 2021 dalam penelitian ini model prediksi penyakit jantung baru diberikan. Penelitian ini mengusulkan algoritma prediksi penyakit jantung yang menggabungkan metode pemilihan fitur tertanam dan jaringan saraf dalam. Metode pemilihan fitur yang disematkan ini didasarkan pada algoritma LinearSVC, menggunakan norma L1 sebagai item penalti untuk memilih subset fitur yang secara signifikan terkait dengan penyakit jantung. Fitur-fitur ini dimasukkan ke dalam jaringan saraf dalam yang kami bangun. Bobot jaringan diinisialisasi dengan penginisialisasi He untuk mencegah pernis gradien atau ledakan sehingga prediktor dapat memiliki kinerja yang lebih baik. Model kami diuji pada dataset penyakit jantung yang diperoleh dari Kaggle. Beberapa indikator termasuk akurasi, recall, presisi, dan F1-score dihitung untuk mengevaluasi prediktor, dan hasilnya menunjukkan bahwa model kami masing-masing mencapai 98,56%.

Selanjutnya Penelitian oleh *Xiao-Yan Gao et al* tahun 2021 dalam penelitian ini metode pembelajaran ensemble digunakan untuk meningkatkan kinerja dalam memprediksi penyakit jantung. Dua fitur metode ekstraksi: analisis diskriminan linier (LDA) dan analisis komponen utama (PCA), digunakan untuk memilih fitur penting dari kumpulan data. Perbandingan antara algoritma pembelajaran mesin dan metode pembelajaran ensemble diterapkan pada fitur yang dipilih. Metode yang berbeda digunakan untuk mengevaluasi model: akurasi, recall, presisi, F-measure, dan ROC. Hasil penelitian menunjukkan metode bagging ensemble learning dengan pohon keputusan telah mencapai kinerja terbaik.

Selanjutnya Penelitian oleh *Yongjie Yuan et al* tahun 2021 dalam Penelitian ini dimaksudkan untuk mengeksplorasi efek dari algoritma jaringan saraf yang berbeda

dalam klasifikasi elektrokardiogram (EKG) pasien dengan penyakit jantung bawaan (PJB). Berdasarkan algoritma EKG jaringan saraf tunggal (CNN) dan algoritma EKG jaringan saraf berulang (RNN), algoritma EKG jaringan saraf multimodal (MNN) dibangun dengan memanfaatkan database MIT-BIH sebagai set pelatihan dan set uji. Selanjutnya, algoritma EKG MNN dioptimalkan untuk membangun algoritma MNN (IMNN) yang ditingkatkan, yang diterapkan pada diagnosis pasien PJK. Pasien PJK yang dirawat antara Agustus 2016 dan Agustus 2019 dipilih untuk analisis untuk membandingkan efek klasifikasi dan tingkat akurasi algoritma IMNN, MNN, CNN ECG, dan RNN ECG. Sensitivitas klasifikasi dan true positive rate algoritma IMNN pada keempat aspek secara signifikan lebih tinggi dibandingkan algoritma MNN. Akurasi klasifikasi algoritma CNN ECG dan algoritma RNN ECG berada di atas 98%, sedangkan algoritma MNN dan algoritma IMNN lebih baik daripada algoritma CNN ECG dan algoritma RNN ECG, dan tingkat akurasi dapat mencapai 98,5% atau lebih. Apalagi tingkat akurasi algoritma IMNN bisa mencapai lebih dari 98%.

2.2 Penyakit Jantung

Penyakit jantung mengacu pada beberapa jenis kondisi jantung. Jenis penyakit jantung yang paling umum di Amerika Serikat adalah penyakit arteri koroner (CAD), yang mempengaruhi aliran darah ke jantung. Penurunan aliran darah dapat menyebabkan serangan jantung. Terkadang penyakit jantung mungkin tidak terdiagnosis sampai seseorang mengalami tanda atau gejala serangan jantung, gagal jantung, atau aritmia. Ketika peristiwa ini terjadi, gejala dapat mencakup Serangan jantung : Nyeri atau ketidaknyamanan dada, nyeri punggung atas atau leher, gangguan pencernaan, mulas, mual atau muntah, kelelahan ekstrem, ketidaknyamanan tubuh bagian atas, pusing, dan sesak napas. Aritmia: Perasaan berdebar-debar di dada (palpitasi). Gagal jantung : Sesak napas, kelelahan, atau pembengkakan pada kaki, pergelangan kaki, tungkai, perut, atau vena leher. Ada beberapa faktor resikopenyakit jantung yaitu Tinggi tekanan darah , darah tinggi kolesterol , dan merokok merupakan faktor risiko utama untuk penyakit jantung. Sekitar setengah dari orang di Amerika Serikat (47%) memiliki setidaknya satu dari

tiga faktor risiko ini. 2 Beberapa kondisi medis dan pilihan gaya hidup lainnya juga dapat menempatkan orang pada risiko penyakit jantung yang lebih tinggi, termasuk: Diabetes, Kegemukan dan obesitas, Pola makan tidak sehat, Ketidakaktifan fisik dan Penggunaan alkohol yang berlebihan (Hester et al. 2016). Sesuai penelitian yang dilakukan oleh American Heart Society (American Heart Association, 2016), berikut ini kemungkinan faktor risiko untuk kondisi CAD:

1. Usia

   Mayoritas orang yang meninggal karena penyakit jantung koroner berusia 65 tahun atau lebih. Pada usia, lebih dari 65 tahun, wanita yang mengalami serangan jantung lebih mungkin meninggal karena serangan jantung daripada pria dalam beberapa minggu.

2. Jenis Kelamin

   Pria memiliki risiko lebih besar terkena serangan jantung daripada wanita, dan mereka mengalami serangan lebih awal dalam hidup. Pada wanita, bahkan setelah menopause, peningkatan angka kematian akibat penyakit jantung tidak sebesar pria.

3. Keturunan (Termasuk Ras)

   Di mana anak-anak memiliki orang tua dengan penyakit jantung, mereka lebih mungkin untuk mengembangkannya sendiri. Kebanyakan orang dengan riwayat keluarga yang kuat penyakit jantung memiliki satu atau lebih faktor risiko lain yang mungkin menyebabkan penyakit.

4. Asap tembakau

   Risiko seorang perokok terkena PJK jauh lebih tinggi daripada bukan perokok. Merokok merupakan faktor risiko independen yang kuat untuk kematian jantung mendadak pada pasien dengan PJK. Merokok juga bekerja dengan faktor risiko lain untuk sangat meningkatkan risiko PJK. Paparan asap orang lain meningkatkan risiko penyakit jantung bahkan untuk non-perokok.

5. Kolesterol darah tinggikolesterol

   Saat darah meningkat, begitu pula risiko PJK. Ketika faktor risiko lain (seperti tekanan darah tinggi dan asap tembakau) ada, risiko ini bahkan lebih

meningkat. Kadar kolesterol seseorang juga dipengaruhi oleh usia, jenis kelamin, keturunan dan pola makan.

6. Tekanan darah tinggi

   Tekanan darah tinggi meningkatkan beban kerja jantung, menyebabkan otot jantung menebal dan menjadi kaku. Kekakuan otot jantung ini tidak normal, dan mencegah jantung bekerja dengan baik. Ini juga meningkatkan risiko stroke, serangan jantung, gagal ginjal, dan gagal jantung kongestif.

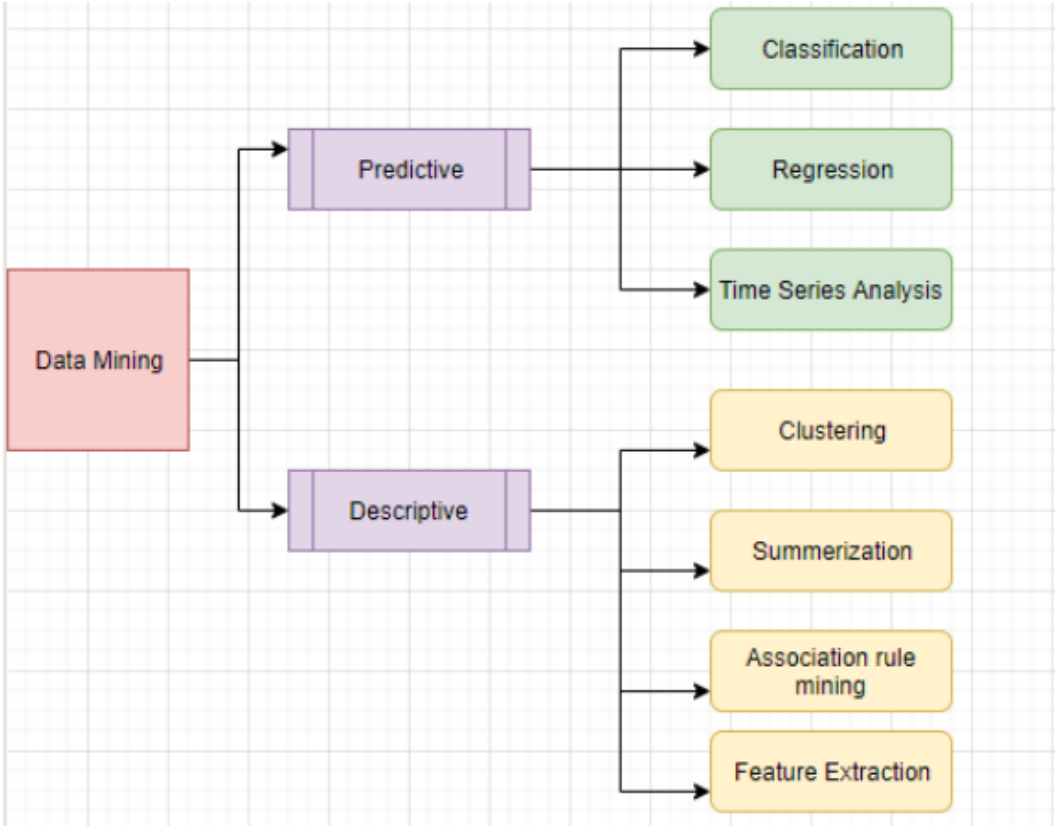
7. Kurang aktivitas fisik

   Gaya hidup yang tidak aktif merupakan faktor risiko penyakit jantung koroner. Aktivitas fisik yang teratur, sedang hingga berat membantu mengurangi risiko penyakit jantung dan pembuluh darah. Bahkan aktivitas dengan intensitas sedang sangat membantu jika dilakukan secara teratur dan dalam jangka panjang. Aktivitas fisik dapat membantu mengontrol kolesterol darah, diabetes dan obesitas, dan pada beberapa orang, menurunkan tekanan darah.

8. Obesitas dan kelebihan berat badan

   Orang yang memiliki kelebihan lemak tubuh — terutama di bagian pinggang — lebih mungkin terkena penyakit jantung dan stroke meskipun tidak ada faktor risiko lain. Orang dewasa yang kelebihan berat badan dan obesitas dengan faktor risiko penyakit kardiovaskular seperti tekanan darah tinggi, kolesterol tinggi, atau gula darah tinggi dapat membuat perubahan gaya hidup untuk menurunkan berat badan dan menghasilkan pengurangan trigliserida, glukosa darah, HbA1c, dan risiko pengembangan Tipe 2 yang bermakna secara klinis diabetes. Banyak orang mungkin mengalami kesulitan menurunkan berat badan, tetapi penurunan berat badan yang berkelanjutan dari 3 sampai 5 persen berat badan dapat menyebabkan pengurangan yang bermakna secara klinis dalam beberapa faktor risiko. Penurunan berat badan di atas 5 persen dapat menurunkan tekanan darah, kolesterol, dan kadar glukosa darah.

2.3 Data Mining



Gambar 2.1 Tugas Data Mining

Data mining adalah bidang untuk memeriksa database yang sudah ada sebelumnya untuk menggali informasi baru. Bidang ini membentuk dasar Analisa dan digunakan untuk membuat prediksi untuk berbagai bidang seperti pemasaran, keuangan, Medis, cuaca, dll. Sering kali, istilah penambangan data ditafsirkan untuk menemukan data yang relevan dari kumpulan data yang sudah ada sebelumnya, itu adalah ekstraksi informasi atau pola yang relevan untuk membuat prediksi. Data mining adalah bidang yang pasti digunakan di bidang Medis (Chatterjee et al. 2019). Dalam data mining, tugas dapat dikategorikan menjadi dua jenis yaitu Prediktif dan Deskriptif.

1. Prediktif

   Dengan bantuan pendekatan ini, pengguna dapat membuat prediksi tentang nilai data yang digunakan dalam berbagai database dengan bantuan hasil yang sudah diketahui dari beberapa data yang berbeda atau berdasarkan data

historis. Prediktif dapat dicirikan lebih lanjut menjadi empat bagian lain yang tercantum di bawah ini:

a. Klasifikasi : Pada dasarnya bertanggung jawab untuk menemukan fungsi yang mengklasifikasikan item data ke dalam salah satu dari beberapa kelas yang telah ditentukan sebelumnya. Ini dapat dipahami lebih lanjut seperti ini:
b. Regresi : Regresi adalah fungsi datamining yang digunakan untuk memprediksi angka. Misalnya, model regresi dapat digunakan untuk memprediksi nilai barang antik tertentu berdasarkan tampilannya, berapa banyak penambahan atau penghapusan yang dilakukan, kondisi penyimpanannya, dan faktor lainnya. Umumnya, tugas regresi dimulai dengan sekumpulan data di mana nilai target sudah dikenal atau dipelajari.
c. Analisis Deret Waktu : Deret waktu dapat didefinisikan sebagai rangkaian atau rangkaian titik data yang diturunkan pada titik waktu yang berbeda. Ini biasanya diturunkan dalam interval waktu yang teratur seperti detik, jam, hari, bulan, tahun, dan seterusnya.
d. Prediksi : Prediksi adalah teknik data mining yang dapat digunakan untuk mengekstrak model dan mewakili kelas data untuk memprediksi tren data masa depan. Prediksi dapat didefinisikan sebagai output dari suatu algoritma setelah diinstruksikan untuk beroperasi pada kumpulan data historis dan kemudian diterapkan pada data baru. Misalnya, meramalkan cuaca adalah contoh prediksi data mining yang paling populer. Dengan kata lain, itu juga dapat didefinisikan sebagai ramalan atau ramalan. Contoh lain dari prediksi bisa menjadi paranormal yang memberi tahu seseorang tentang masa depannya.

2. deskriptif :

   Jenis tugas data mining yang kedua adalah tugas Deskriptif. Jenis ini mencakup fungsi-fungsi berikut:

a. Aturan Asosiasi : Dalam data mining, aturan asosiasi dapat digunakan untuk mengungkap asosiasi atau koneksi di antara berbagai set item yang berbeda. Asosiasi juga dapat digunakan untuk mendapatkan hubungan antara objek yang berbeda.
b. Pengelompokan : Dalam data mining, proses clustering dapat digunakan untuk mendapatkan objek data yang memiliki beberapa kesamaan. Kesamaan tersebut dapat dimanifestasikan atas dasar faktor yang berbeda seperti perilaku pembelian, daya tanggap terhadap tindakan tertentu, lokasi geografis dan sebagainya. Dengan mengelompokkan informasi ini, akan sangat membantu untuk memahami pelanggan dengan lebih baik dan akibatnya memberikan layanan yang lebih baik kepada pelanggan.
c. Ringkasan : Peringkasan dapat didefinisikan sebagai proses generalisasi data. Dalam proses ini, sekumpulan data yang relevan dan penting diringkas yang kemudian menghasilkan sekumpulan data yang lebih kecil yang memberikan informasi yang terkumpul dari data tersebut. Informasi yang diringkas ini dapat berguna untuk penjualan atau hubungan pelanggan.
d. Penemuan Urutan: Sequence discovery atau sequential pattern mining, adalah teknik data mining yang digunakan untuk menemukan pola yang relevan dan penting dalam data sekuensial. Program penambangan ini menilai kriteria tertentu yang merupakan frekuensi kemunculan, durasi, atau nilai dalam satu set urutan untuk menemukan pola tersembunyi atau tersembunyi.

Model prediktif bekerja dengan membuat prediksi tentang nilai data, yang menggunakan hasil yang diketahui yang ditemukan dari kumpulan data yang berbeda. Tugas-tugas yang termasuk dalam model data mining prediktif meliputi klasifikasi, prediksi, regresi dan analisis deret waktu. Model datamining prediktif memprediksi hasil masa depan berdasarkan catatan masa lalu yang ada dalam database atau dengan jawaban yang diketahui. Model deskriptif sebagian besar mengidentifikasi pola atau hubungan dalam kumpulan data. Ini berfungsi untuk

mengeksplorasi properti dari data yang diperiksa sebelumnya dan bukan untuk memprediksi properti baru. Model deskriptif mencakup tugas yang harus dilakukan sebagai Pengelompokan, Aturan Asosiasi, Peringkasan, dan Analisis Urutan. Model data mining deskriptif menemukan pola dalam data dan memahami hubungan antara atribut yang diwakili oleh data (Srivastava and Khare 2017).

2.4 Klasifikasi

Klasifikasi adalah teknik data mining yang paling umum pada pembelajaran mesin. Pada dasarnya klasifikasi digunakan untuk mengklasifikasikan setiap elemen dalam kumpulan data ke dalam satu set kelas atau grup yang telah ditentukan. Teknik klasifikasi adalah macam-macam teknik matematika seperti SVM, pohon keputusan, KNN dan statistik. Untuk mengembangkan perangkat lunak yang dapat mempelajari cara mengklasifikasikan item data ke dalam kelompok. Misalnya -Untuk menerapkan klasifikasi dalam aplikasi bahwa "diberikan semua catatan pasien yang akan mendapatkan masalah jantung, memprediksi siapa yang mungkin akan mendapatkan masalah jantung di masa mendatang." Untuk kasus ini, Untuk membagi catatan pasien menjadi dua kelompok yang diberi nama “CMS ya” dan “CMS Tidak”. Dapat meminta perangkat lunak penambangan data kami untuk mengklasifikasikan pasien ke dalam kelompok terpisah (*Marconi et al. 2019*).

2.5 K-Nearest Neighbors Classification

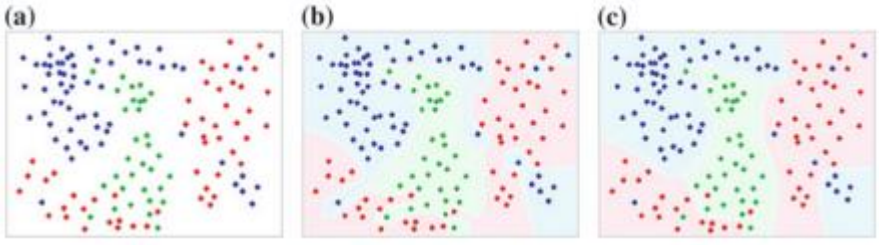
K-Nearest Neighbors atau K-NN adalah algoritma sederhana yang menyimpan semua kasus yang tersedia dan mengklasifikasikan kasus baru berdasarkan fungsi keputusan (misalnya, ukuran jarak). Diberikan dataset pelatihan D dan jarak ukuran

- $(x_i, y_i), i = 1, 2, ..., N$
- x_i adalah data pelatihan dalam R^n
- y_i adalah kelas yang sesuai dari data x_i, dan $y_i \{c_j, j\ 1, 2, ... M\}$
- $dist\ (x - x_i) = || x - x_i ||$

Data pengamatan baru x diklasifikasikan ke dalam salah satu kelas y_j menggunakan algoritma berikut :

1. Masukkan data baru x
2. Hitung jarak x ke semua sampel pelatihan xi dalam kumpulan data: *dist* $(x - x_i)$
3. Urutkan *dist* $(x - x_i)$ $(i = 1, 2, \ldots, N)$ dalam urutan menaik dan urutkan semua x_i sesuai dengan: $x_{r1}, x_{r2}, \ldots, x_{rk}, \ldots, x_{rN}$
4. Untuk klasifikasi tetangga terdekat (NN) mengklasifikasikan *x* ke y_{rl}
   a. Untuk klasifikasi *K-NN*, klasifikasikan *x* ke kelas mayoritas y_{rp} di antara data peringkat k teratas: { $x_{r1}, x_{r2}, \ldots, x_{rk}$, }.

Meskipun Euclidean (L_2) dan jarak blok kota (L_1) adalah pilihan tipikal untuk ukuran jarak, jarak lain dapat digunakan tergantung pada aplikasinya. Tetangga terdekat (NN atau 1-NN) menghasilkan terlalu banyak kelas, sedangkan *K-NN* memberikan hasil klasifikasi yang lebih andal. Hal ini dikarenakan nilai k memiliki efek smoothing yang membuat classifier lebih tahan terhadap outlier. Namun, kinerja pengklasifikasi *K-NN* tergantung pada pilihan k yang biasanya ditentukan secara empiris. Gambar 2.2 menunjukkan perbandingan antara *classifier NN* dan *classifier K-NN* (*Karpathy A, 2019*). Hal ini dapat dilihat dari dua hasil klasifikasi, pada kasus pengklasifikasi NN (setelah penggabungan), titik data outlier membuat pulau-pulau kecil dalam suatu kelas (misalnya, titik merah dalam kelas hijau) dan sudut tajam pada batas kelas, pulau-pulau, dan sudut-sudut tajam kemungkinan mengarah pada prediksi yang salah untuk lebih jelasnya bisa melihat gambar 2.2 dibawah ini.



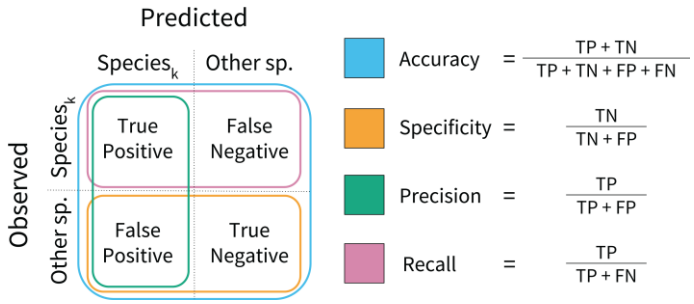
Gambar 2.2 Perbandingan antara NN dan K-NN. a Data yang akan diklasifikasikan; b hasil klasifikasi

Pengklasifikasi menghaluskan outlier ini, yang mengarah pada klasifikasi yang lebih baik pada data. Namun, pengklasifikasi 5-NN juga menyebabkan kesalahan klasifikasi yang ditandai dengan titik biru di wilayah merah dan titik merah di wilayah hijau. Bisa juga terjadi kebingungan dengan perolehan suara yang sama di antara lima tetangga terdekat (misalnya, dua tetangga berwarna merah, dua tetangga berikutnya berwarna biru, dan tetangga terakhir berwarna hijau). Kesalahan klasifikasi semacam ini dapat diatasi sampai batas tertentu dengan menggunakan K-NN berbobot. Idenya adalah untuk memberi bobot lebih pada tetangga dengan jarak yang lebih pendek ke data uji daripada ke tetangga yang jauh. K-NN berbobot yang umum digunakan adalah K-NN berbobot Gaussian. Tidak seperti pengklasifikasi lain yang independen dari data pelatihan asli setelah dilatih, pengklasifikasi K-NN tidak memiliki memori. Jika kita menganalogikan pengklasifikasi dengan seorang ahli yang berkeliling dunia untuk menilai (mengklasifikasikan) berbagai jenis barang antik untuk orang-orang. Sementara jenis penikmat lain hanya perlu mengambil perangkat yang merangkum karakteristik utama barang antik, penikmat K-NN harus membawa setiap jenis barang antik asli dalam koleksinya untuk membuat penilaian baru. Ini mungkin terdengar terlalu rumit, namun, salah satu keuntungan utama pengklasifikasi K-NN adalah dapat mengklasifikasikan data yang tidak dapat dipisahkan secara nonlinier. Ini adalah ide kunci di balik mesin vektor dukungan berbasis kernel (SVM) *(Kantardzic, 2019).*

2.6 *Confusion Matrix*

Matriks konfigurasi adalah tabel yang terdiri dari jumlah baris data uji yang diprediksi benar dan salah dengan model klasifikasi yang digunakan. Tabel Confusion Matrix diperlukan untuk memilih kinerja terbaik dari sebuah model klasifikasi (Romadhon & Kurniawan, 2021). Selama pelatihan model, kinerja dinilai berdasarkan per-sampel menggunakan akurasi model dan skor kerugian log. Akurasi model menghitung proporsi sampel yang diklasifikasikan dengan benar dalam data uji (Gambar. 2.3), dan nilai akurasi model yang tinggi diinginkan. Log loss menilai apakah probabilitas prediksi dikalibrasi dengan baik, menghukum

prediksi yang salah dan tidak pasti. Skor kehilangan log yang rendah menunjukkan bahwa kesalahan klasifikasi terjadi pada tingkat yang mendekati tingkat probabilitas yang diprediksi. Selama pengujian model, kinerja dinilai menggunakan akurasi peringkat-1 dan biaya lintas-entropi (*Marconi et al., 2019*). Akurasi peringkat-1 dihitung berdasarkan ID spesies mana yang diprediksi dengan probabilitas tertinggi. Skor lintas-entropi mirip dengan fungsi kehilangan log, tetapi diskalakan menggunakan fungsi indikator. Ini dapat ditafsirkan dengan cara yang mirip dengan akurasi dan kehilangan log; akurasi peringkat-1 yang tinggi dan skor entropi silang yang rendah diinginkan (*Hastie, Tibshirani & Friedman, 2009*). Metrik pengujian model sekunder dihitung untuk setiap spesies menggunakan data uji. Ini termasuk model spesifisitas, presisi, dan recall yang perumusanya dapat dilihat pada Gambar 2.3 dibawah ini.



Gambar 2.3 Metrik kinerja model

Gambar 2.3 merupakan metrik kinerja model yang mengungkapkan perilaku model yang skor akurasinya mungkin tidak jelas. Spesifisitas menilai kinerja model pada spesies non-target, menghukum overprediksi spesies target (yaitu, sejumlah besar positif palsu). Presisi juga menghukum overprediksi, tetapi menilai tingkat overprediksi relatif terhadap tingkat prediksi positif yang benar. Recall menghitung proporsi prediksi positif sejati dengan jumlah total pengamatan positif per spesies. Nilai yang lebih tinggi diinginkan untuk masing-masing. Metrik ini dihitung untuk membantu interpretasi, tetapi tidak digunakan untuk memeringkat kinerja model secara formal. Akurasi adalah salah satu metrik untuk mengevaluasi model klasifikasi. Secara informal, akurasi adalah sebagian kecil dari prediksi model kami yang benar. Secara formal, akurasi memiliki definisi sebagai berikut:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

Untuk klasifikasi biner, akurasi juga dapat dihitung dalam hal positif dan negatif sebagai berikut:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Dimana TP = Positif Benar, TN = Negatif Benar, FP = Positif Palsu, dan FN = Negatif Palsu (*Mari 2015*).

2.7 Klasifikasi: Kurva ROC dan AUC

2.7.1 Kurva ROC

Sebuah ROC kurva (*Receiver Operating Characteristic Curve*) adalah grafik yang menunjukkan kinerja model klasifikasi sama sekali ambang klasifikasi. Kurva ini memplot dua parameter:

- Tingkat Positif Benar
- Tingkat Positif Palsu

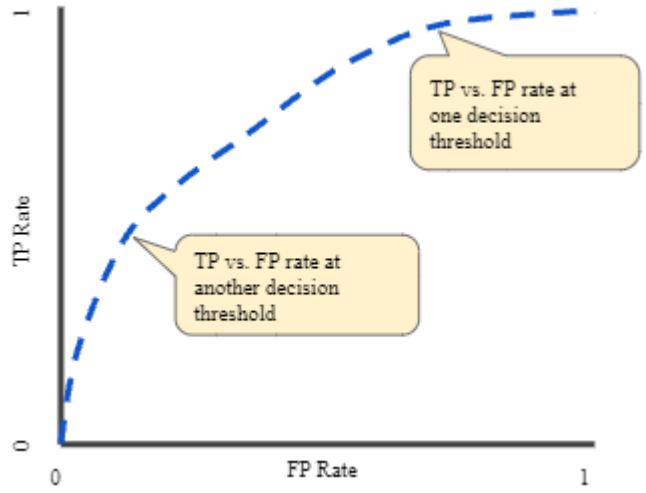
True Positive Rate (TPR) adalah sinonim untuk mengingat dan oleh karena itu didefinisikan sebagai berikut:

$$TPR = \frac{TP}{TP + FN}$$

False Positive Rate (FPR) didefinisikan sebagai berikut:

$$TPR = \frac{TP}{TP + FN}$$

Kurva ROC memplot TPR vs. FPR pada ambang klasifikasi yang berbeda. Menurunkan ambang klasifikasi mengklasifikasikan lebih banyak item sebagai positif, sehingga meningkatkan Positif Palsu dan Positif Benar. Gambar 2.4 menunjukkan kurva ROC yang khas.

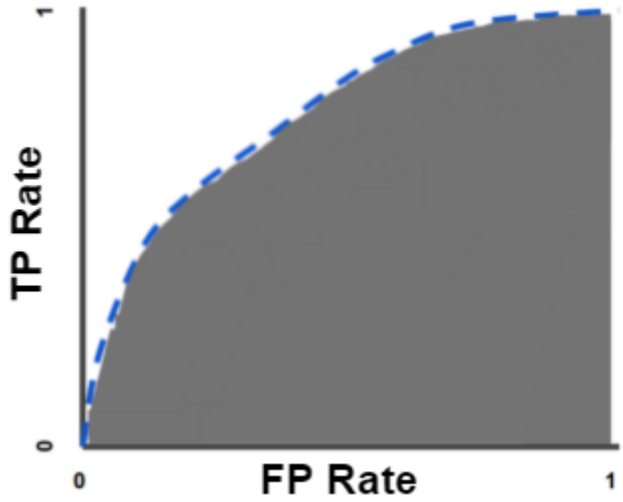


Gambar 2.4 Tingkat TP vs. FP pada ambang klasifikasi yang berbeda

Gambar 2.4 Tingkat TP vs. FP pada ambang klasifikasi untuk menghitung titik-titik dalam kurva ROC, kita dapat mengevaluasi model regresi logistik berkali-kali dengan ambang klasifikasi yang berbeda, tetapi ini akan menjadi tidak efisien. Untungnya, ada algoritma berbasis penyortiran yang efisien yang dapat memberikan informasi ini untuk kita, yang disebut AUC.

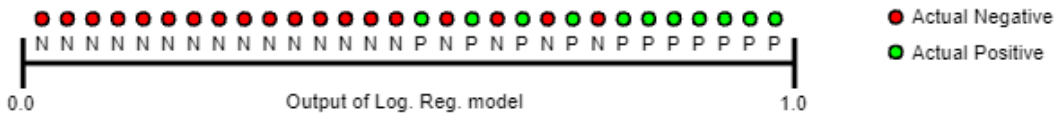
2.7.2 AUC: Area Under the Kurva ROC

AUC adalah singkatan dari "Area di bawah Kurva ROC." Artinya, AUC mengukur seluruh area dua dimensi di bawah seluruh kurva ROC (pikirkan kalkulus integral) dari (0,0) hingga (1,1) untuk lebih jelasnya bisa dilihat pada gambar 2.5 dibawah ini.



Gambar 2.5 AUC (Area di bawah Kurva ROC)

AUC memberikan ukuran kinerja agregat di semua ambang klasifikasi yang mungkin. Salah satu cara untuk menginterpretasikan AUC adalah sebagai probabilitas bahwa model memberi peringkat contoh positif acak lebih tinggi daripada contoh negatif acak. Sebagai contoh, diberikan contoh berikut, yang disusun dari kiri ke kanan dalam urutan menaik dari prediksi regresi logistic seperti pada gambar 2.6 dibawah ini:



Gambar 2.6 Prediksi peringkat dalam urutan skor regresi logistik

AUC mewakili probabilitas bahwa contoh acak positif (hijau) diposisikan di sebelah kanan contoh acak negatif (merah).

Rentang nilai AUC dari 0 hingga 1. Model yang prediksinya 100% salah memiliki AUC 0,0; yang prediksinya 100% benar memiliki AUC 1,0.

AUC diinginkan karena dua alasan berikut:

- AUC adalah skala-invarian . Ini mengukur seberapa baik prediksi diberi peringkat, bukan nilai absolutnya.
- AUC adalah klasifikasi-ambang-invarian . Ini mengukur kualitas prediksi model terlepas dari ambang klasifikasi apa yang dipilih.

Namun, kedua alasan ini disertai dengan peringatan, yang dapat membatasi kegunaan AUC dalam kasus penggunaan tertentu:

- Skala invarians tidak selalu diinginkan. Misalnya, terkadang kami benar-benar membutuhkan keluaran probabilitas yang terkalibrasi dengan baik, dan AUC tidak akan memberi tahu kami tentang itu.
- Klasifikasi-ambang invarians tidak selalu diinginkan. Dalam kasus di mana ada perbedaan besar dalam biaya negatif palsu vs positif palsu, mungkin penting untuk meminimalkan satu jenis kesalahan klasifikasi. Misalnya, saat melakukan deteksi spam email, Anda mungkin ingin memprioritaskan meminimalkan positif palsu (bahkan jika itu menghasilkan peningkatan negatif palsu yang signifikan). AUC bukan metrik yang berguna untuk jenis pengoptimalan ini.

Performance keakurasian AUC dapat diklasifikasikan menjadi beberapa kelompok yaitu: (Gorunescu, 2011)

1. 0.90 – 1.00 = *Exellent Classification*
2. 0.80 – 0.90 = *Good Classification*
3. 0.70 – 0.80 = *Fair Classification*
4. 0.60 – 0.70 = *Poor Classification*
5. 0.50 – 0.60 = *Failure Classification*

2.8 *Feature Selection*

Kita semua mungkin menghadapi masalah dalam mengidentifikasi fitur terkait dari kumpulan data dan menghapus fitur yang tidak relevan atau kurang penting dengan tidak berkontribusi banyak pada variabel target kami untuk mencapai akurasi yang lebih baik untuk model kami. Proses Pemilihan Fitur sangat penting dalam pembelajaran mesin yang sangat memengaruhi kinerja model Anda. Fitur data yang Anda gunakan untuk melatih model atau algoritme pembelajaran mesin memiliki pengaruh besar pada kinerja yang dapat Anda capai. Fitur yang tidak relevan atau hilang dapat berdampak negatif pada kinerja sistem Pemilihan fitur adalah salah satu langkah pertama dan penting saat melakukan tugas pembelajaran mesin apa pun. Fitur dalam kasus kumpulan data berarti kolom. Ketika kita mendapatkan dataset apa pun, belum tentu setiap kolom (fitur) akan berdampak pada variabel output. Jika kami menambahkan fitur yang tidak relevan ini ke dalam model, itu hanya akan memperburuk model dan dapat mengurangi keakuratan model dan membuat model Anda belajar berdasarkan fitur yang tidak relevan. Hal ini menimbulkan perlunya melakukan seleksi fitur. Ketika datang ke implementasi pemilihan fitur dalam fitur Numerik dan Kategoris harus diperlakukan secara berbeda. Disini kita akan membahas tentang pemilihan fitur Numerik. Oleh karena itu sebelum menerapkan metode berikut, kita perlu memastikan bahwa Data Frame hanya berisi fitur Numerik.

Manfaat melakukan pemilihan fitur sebelum memodelkan data Anda adalah sebagai berikut:

- Mengurangi Overfitting: Data yang lebih sedikit berarti lebih sedikit kesempatan untuk membuat keputusan berdasarkan noise.
- Meningkatkan Akurasi: Data yang kurang menyesatkan berarti akurasi pemodelan meningkat.
- Mengurangi Kompleksitas: lebih sedikit titik data mengurangi kompleksitas algoritme dan membuatnya lebih mudah dipahami.
- Pelatihan Lebih Cepat: Ini memungkinkan algoritme pembelajaran mesin untuk berlatih lebih cepat.

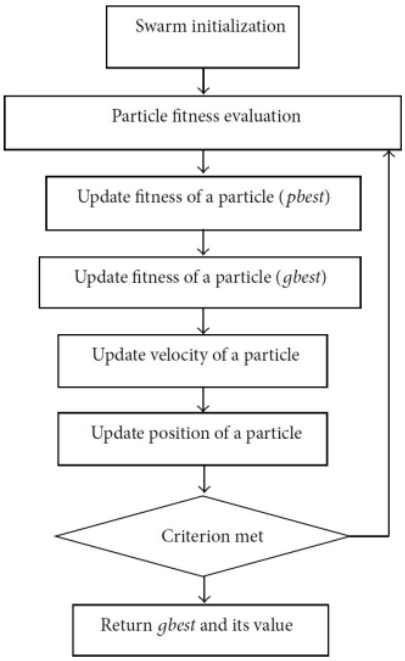
Pada sistem ini digunakan dua proses seleksi fitur yaitu proses seleksi fitur sekuensial/forward dan proses seleksi fitur mundur. Pada sistem ini digunakan dua proses seleksi fitur yaitu proses seleksi fitur sekuensial/forward dan proses seleksi fitur mundur (Bolón-Canedo et al. 2015).

2.9 Particle swarm optimization (PSO)

Pada awal 1990-an, beberapa penelitian tentang perilaku sosial kelompok hewan dikembangkan. Studi-studi ini menunjukkan bahwa beberapa hewan yang termasuk dalam kelompok tertentu, yaitu burung dan ikan, dapat berbagi informasi di antara kelompok mereka, dan kemampuan tersebut memberi hewan-hewan ini keuntungan bertahan hidup yang besar. Terinspirasi oleh karya-karya ini, Kennedy dan Eberhart mengusulkan pada tahun 1995 algoritma PSO (Kennedy dan Eberhart 1995), sebuah algoritma metaheuristik yang sesuai untuk mengoptimalkan fungsi kontinu nonlinier. Penulis memperoleh algoritme yang terinspirasi oleh konsep kecerdasan kawanan, yang sering terlihat pada kelompok hewan, seperti kawanan dan kawanan. Untuk menjelaskan bagaimana PSO telah mengilhami perumusan algoritma optimasi untuk memecahkan masalah matematika yang kompleks, diskusi tentang perilaku kawanan disajikan. Kawanan burung yang terbang di atas suatu tempat harus menemukan titik untuk mendarat dan, dalam hal ini, definisi titik mana seluruh kawanan harus mendarat adalah masalah yang kompleks, karena tergantung pada beberapa masalah, yaitu memaksimalkan ketersediaan makanan. dan meminimalkan risiko keberadaan predator. Dalam konteks ini, gerakan burung dapat dipahami sebagai sebuah koreografi; burung-burung secara sinkron bergerak

untuk suatu periode sampai tempat terbaik untuk mendarat ditentukan dan semua kawanan mendarat sekaligus. Dalam contoh yang diberikan, pergerakan kawanan hanya terjadi seperti yang dijelaskan setelah semua anggota kawanan dapat berbagi informasi di antara mereka sendiri; jika tidak, setiap hewan kemungkinan besar akan mendarat di titik dan waktu yang berbeda. Studi tentang perilaku sosial hewan dari awal 1990-an yang dinyatakan sebelumnya dalam teks ini menunjukkan bahwa semua burung dari kawanan yang mencari titik yang baik untuk mendarat dapat mengetahui titik terbaik sampai ditemukan oleh salah satu anggota kawanan. Dengan cara itu, setiap anggota kawanan menyeimbangkan pengalaman pengetahuan individu dan kawanannya, yang dikenal sebagai pengetahuan sosial. Seseorang mungkin memperhatikan bahwa kriteria untuk menilai apakah suatu titik baik atau tidak dalam kasus ini adalah kondisi kelangsungan hidup yang ditemukan di titik pendaratan yang memungkinkan, seperti yang disebutkan sebelumnya dalam teks ini. Masalah untuk menemukan titik terbaik ke tanah yang dijelaskan memiliki masalah optimasi. Kawanan harus mengidentifikasi titik terbaik, misalnya, garis lintang dan garis bujur, untuk memaksimalkan kondisi kelangsungan hidup anggotanya. Untuk melakukannya, setiap burung terbang mencari dan menilai titik-titik yang berbeda menggunakan beberapa kriteria bertahan hidup pada saat yang sama. Masing-masing memiliki keunggulan untuk mengetahui di mana titik lokasi terbaik ditemukan hingga diketahui oleh seluruh swarm. Kennedy dan Eberhart terinspirasi oleh perilaku sosial burung, yang memberi mereka keuntungan besar untuk bertahan hidup ketika memecahkan masalah menemukan titik aman ke daratan, mengusulkan sebuah algoritma yang disebut PSO yang bisa meniru perilaku ini. Versi inersia, juga dikenal sebagai versi klasik, dari algoritma diusulkan pada tahun 1995 (Kennedy dan Eberhart 1995). Algoritma ini terinspirasi dari perilaku kawanan seperti kawanan burung dan sekolah di alam. PSO telah banyak digunakan dan merupakan inspirasi untuk area penelitian baru yang disebut swarm intelligence (Yang, 2008). PSO adalah teknik pencarian global yang sukses dan bernilai. Ini adalah algoritma yang cocok untuk mengatasi masalah pemilihan fitur karena alasan berikut: pengkodean fitur yang mudah, fasilitas pencarian global, komputasi yang masuk akal, parameter yang lebih sedikit, dan implementasi

yang lebih mudah. Alur PSO untuk pemilihan fitur ditunjukkan pada Gambar 2.7 dibawah ini.



Gambar 2.7 Alur PSO untuk seleksi fitur

PSO diterapkan untuk pemilihan fitur karena alasan yang disebutkan di atas di mana subset dari komponen utama atau fitur utama dieksplorasi dan dipilih melalui PSO. Dalam PSO, partikel mewakili kandidat solusi dalam partikel ruang pencarian dan membentuk populasi yang juga dikenal sebagai swarm. Segerombolan partikel dihasilkan dengan mendistribusikan 1 s dan 0 s secara acak. Untuk setiap partikel, jika komponen utama adalah 1, dipilih dan komponen utama dengan 0 diabaikan. Jadi, setiap partikel menunjukkan subset yang berbeda dari komponen utama. Partikel swarm diinisialisasi secara acak dan kemudian bergerak di ruang pencarian atau ruang utama untuk mencari subset fitur yang optimal dengan memperbarui posisi dan kecepatannya. Posisi partikel i saat ini dan kecepatannya dinyatakan dalam perumusan (1) dan perumusan (2):

$$x_i = \{x_{i1}, x_{i2}, \ldots, x_{iD}\}, \tag{1}$$

di mana D adalah dimensi ruang pencarian utama,

$$v_i = \{v_{i1}, v_{i2}, \ldots, v_{iD}\}. \tag{2}$$

Kecepatan dan posisi partikel i dihitung dengan perumusan (3)

$$v_{id}^{t+1} = \omega * v_{id}^{t} + c_1 * r_{1i} * \left(p_{id} - x_{id}^{t}\right) + c_2 * r_{2i} * \left(p_{gd} - x_{id}^{t}\right), \quad (3)$$

$$x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1},$$

t menunjukkan iterasi ke-t dalam proses dan d menunjukkan dimensi ke-d dalam ruang pencarian. w adalah berat inersia dan c_1 dan c_2 adalah konstanta percepatan. r_{1i} dan r_{2i} adalah nilai acak yang terdistribusi merata di [0,1]. p_{id} dan p_{gd} mewakili elemen $_{pbest}$ dan $_{gbest}$ dalam dimensi ke-d. Nilai posisi dan kecepatan setiap partikel terus diperbarui untuk mencari kumpulan fitur terbaik hingga kriteria penghentian terpenuhi yang dapat berupa jumlah maksimum iterasi atau nilai fitness yang memuaskan. Algoritma PSO yang diterapkan dijelaskan.

Langkah 1 (inisialisasi kawanan). Inisialisasi posisi dan kecepatan setiap partikel secara acak.

Langkah 2 (evaluasi kebugaran partikel)

Langkah 3. Perbarui kecepatan partikel I dengan perumusan

$$v_{id}^{t+1} = \omega * v_{id}^{t} + c_1 * r_{1i} * \left(p_{id} - x_{id}^{t}\right) + c_2 * r_{2i} * \left(p_{gd} - x_{id}^{t}\right),$$

Perbarui posisi partikel I dengan perumusan

$$x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1},$$

Langkah 4. Jika kriteria berhenti tidak terpenuhi, lanjutkan Langkah 2 dan 3.

Langkah 5. Kembali $_{gbest}$ dan nilai fitnessnya .

2.10 *CRoss Industry Standard Process for Data Mining (CRISP-DM).*

Analisis data menunjukkan bahwa metode *CRISP-DM* adalah metodologi utama yang digunakan oleh para penambang data. *CRoss Industry Standard Process for Data Mining (CRISP-DM). CRISP-DM* merupakan proses dengan enam fase yang secara alami menggambarkan siklus hidup ilmu Data (Larose 2005). Ini seperti seperangkat pagar pembatas untuk membantu Anda merencanakan, mengatur, dan mengimplementasikan proyek ilmu data (atau pembelajaran mesin). Enam fase tersebut meliputi :

1. Pemahaman Bisnis (Business Understanding)

   Fase ini berfokus pada pemahaman tujuan dan kebutuhan proyek. Selain tugas ketiga, tiga tugas lain dalam fase ini adalah aktivitas manajemen proyek dasar yang bersifat universal untuk sebagian besar proyek:

   a. Tentukan tujuan bisnis: Pertama-tama Anda harus "memahami secara menyeluruh, dari perspektif bisnis, apa yang benar-benar ingin dicapai pelanggan." (CRISP-DM Guide) dan kemudian tentukan kriteria keberhasilan bisnis.
   b. Menilai situasi: Menentukan ketersediaan sumber daya, persyaratan proyek, menilai risiko dan kontinjensi, dan melakukan analisis biaya-manfaat.
   c. Tentukan tujuan penambangan data: Selain menentukan tujuan bisnis, Anda juga harus menentukan seperti apa kesuksesan dari perspektif penambangan data teknis.
   d. Menghasilkan rencana proyek: Pilih teknologi dan alat dan tentukan rencana terperinci untuk setiap fase proyek.

   Sementara banyak tim terburu-buru melalui fase ini, membangun pemahaman bisnis yang kuat seperti membangun fondasi rumah – sangat penting.

2. Pemahaman Data (Data Understanding)

   Selanjutnya adalah fase Data Understanding. Menambah dasar Pemahaman Bisnis, ini mendorong fokus untuk mengidentifikasi, mengumpulkan, dan menganalisis kumpulan data yang dapat membantu Anda mencapai tujuan proyek. Fase ini juga memiliki empat tugas:

   a. Kumpulkan data awal: Dapatkan data yang diperlukan dan (jika perlu) masukkan ke dalam alat analisis Anda.
   b. Jelaskan data: Periksa data dan dokumentasikan properti permukaannya seperti format data, jumlah catatan, atau identitas bidang.
   c. Jelajahi data: Gali data lebih dalam. Query, visualisasikan, dan identifikasi hubungan antar data.

d. Verifikasi kualitas data: Seberapa bersih/kotor datanya? Dokumentasikan masalah kualitas apa pun.

3. Persiapan data (Data Preparation)

   Fase ini, yang sering disebut sebagai “data munging”, menyiapkan kumpulan data akhir untuk pemodelan. Ini memiliki lima tugas:

   a. Pilih data: Tentukan kumpulan data mana yang akan digunakan dan dokumentasikan alasan penyertaan/pengecualian.
   b. Membersihkan data: Seringkali ini adalah tugas terlama. Tanpa itu, Anda mungkin akan menjadi korban sampah-masuk, sampah-keluar. Praktik umum selama tugas ini adalah mengoreksi, mengaitkan, atau menghapus nilai yang salah.
   c. Bangun data: Dapatkan atribut baru yang akan membantu. Misalnya, dapatkan indeks massa tubuh seseorang dari bidang tinggi dan berat badan.
   d. Integrasikan data: Buat kumpulan data baru dengan menggabungkan data dari berbagai sumber.
   e. Format data: Format ulang data seperlunya. Misalnya, Anda dapat mengonversi nilai string yang menyimpan angka menjadi nilai numerik sehingga Anda dapat melakukan operasi matematika.

4. Pemodelan (Modeling)

   Di sini Anda mungkin akan membangun dan menilai berbagai model berdasarkan beberapa teknik pemodelan yang berbeda. Fase ini memiliki empat tugas:

   a. Pilih teknik pemodelan: Tentukan algoritma mana yang akan dicoba (misalnya regresi, jaringan saraf).
   b. Hasilkan desain pengujian: Sambil menunggu pendekatan pemodelan, Anda mungkin perlu membagi data menjadi set pelatihan, pengujian, dan validasi.
   c. Model build: Meski terdengar glamor, ini mungkin hanya mengeksekusi beberapa baris kode seperti “reg = LinearRegression().fit(X, y)”.

d. Menilai model: Umumnya, beberapa model bersaing satu sama lain, dan ilmuwan data perlu menginterpretasikan hasil model berdasarkan pengetahuan domain, kriteria keberhasilan yang telah ditentukan sebelumnya, dan desain pengujian.

Meskipun panduan CRISP-DM menyarankan untuk "mengulangi pembuatan model dan penilaian sampai Anda sangat yakin bahwa Anda telah menemukan model terbaik", dalam praktiknya tim harus terus mengulangi sampai mereka menemukan model yang "cukup baik", lanjutkan melalui CRISP -DM siklus hidup, kemudian lebih meningkatkan model di iterasi mendatang.

5. Evaluasi (Evaluation)

   Sementara tugas Model Penilaian pada fase Pemodelan berfokus pada penilaian model teknis, fase Evaluasi melihat lebih luas model mana yang paling sesuai dengan bisnis dan apa yang harus dilakukan selanjutnya. Fase ini memiliki tiga tugas:

   a. Evaluasi hasil: Apakah model memenuhi kriteria keberhasilan bisnis? Yang mana yang harus kami setujui untuk bisnis?
   b. Proses peninjauan: Tinjau pekerjaan yang diselesaikan. Apakah ada yang terlewatkan? Apakah semua langkah dijalankan dengan benar? Ringkas temuan dan perbaiki apa pun jika diperlukan.
   c. Tentukan langkah selanjutnya: Berdasarkan tiga tugas sebelumnya, tentukan apakah akan melanjutkan penerapan, mengulangi lebih lanjut, atau memulai proyek baru.

6. Penyebaran (Deployment)

   Sebuah model tidak terlalu berguna kecuali pelanggan dapat mengakses hasilnya. Kompleksitas fase ini sangat bervariasi. Fase terakhir ini memiliki empat tugas:

   a. Merencanakan penyebaran : Kembangkan dan dokumentasikan rencana untuk menerapkan model.

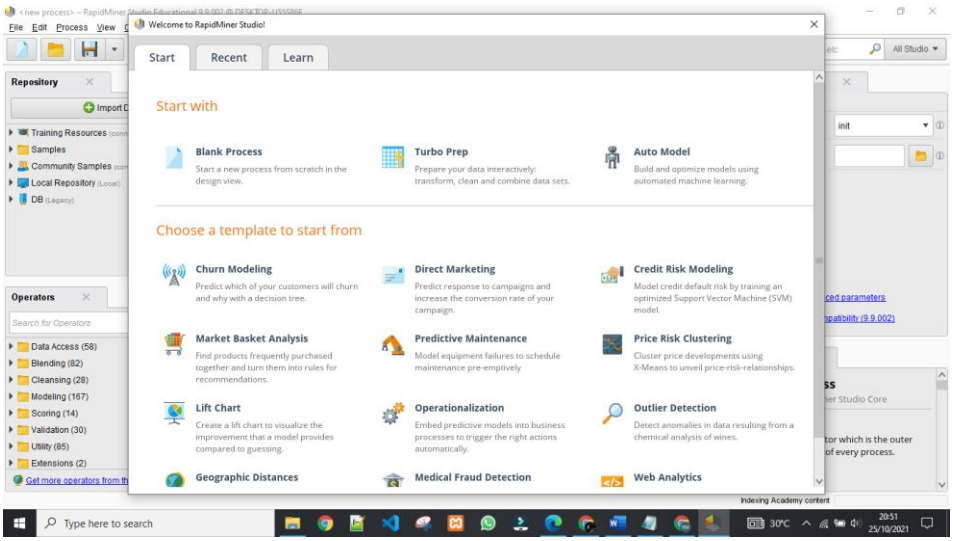
b. Merencanakan pemantauan dan pemeliharaan: Kembangkan rencana pemantauan dan pemeliharaan yang menyeluruh untuk menghindari masalah selama fase operasional (atau fase pasca proyek) suatu model.
c. Menghasilkan laporan akhir: Tim proyek mendokumentasikan ringkasan proyek yang mungkin mencakup presentasi akhir hasil penambangan data. d. Tinjau proyek: Lakukan retrospektif proyek tentang apa yang berjalan dengan baik, apa yang bisa lebih baik, dan bagaimana meningkatkannya di masa depan.

Pekerjaan organisasi Anda mungkin tidak berakhir di situ. Sebagai kerangka kerja proyek, CRISP-DM tidak menguraikan apa yang harus dilakukan setelah proyek (juga dikenal sebagai "operasi"). Tetapi jika model akan diproduksi, pastikan Anda mempertahankan model dalam produksi. Pemantauan konstan dan penyetelan model sesekali sering diperlukan.

2.11 Rapid Miner

Rapid miner adalah alat analisis penambangan data yang digunakan untuk menganalisis data dan mendukung berbagai teknik data mining. Digunakan untuk aplikasi industri, penelitian, pelatihan, pengembangan aplikasi dan pendidikan. Itu mengandung sekitar 100 skema pembelajaran untuk pengelompokan, klasifikasi dan regresi analisis. Ini mendukung sekitar 22 format file seperti .xls, .csv dan sebagainya. Dalam informasi ini dapat diimpor dari berbagai database untuk analisis dan tujuan prediksi (Periasamy et al. 2017). RapidMiner , sebelumnya dikenal sebagai YALE (Yet Another Learning Environment), dikembangkan mulai tahun 2001 oleh Ralf Klinkenberg, Ingo Mierswa, dan Simon Fischer di Artificial Intelligence Unit of the Technical University of Dortmund. Mulai tahun 2006, perkembangannya dimotori oleh Rapid-I, perusahaan yang didirikan oleh Ingo Mierswa dan Ralf Klinkenberg pada tahun yang sama. Pada tahun 2007, nama perangkat lunak diubah dari YALE menjadi RapidMiner. Pada tahun 2013, perusahaan berganti nama dari Rapid-I menjadi RapidMiner. RapidMiner memiliki beberapa sifat sebagai berikut:

1. Ditulis dengan bahasa pemrograman Java sehingga dapat dijalankan diberbagai sistem operasi;
2. Proses penemuan pengetahuan dimodelkan sebagai operator trees;
3. Representasi XML internal untuk memastikan format standar pertukaran data;
4. Bahasa scripting memungkinkan untuk eksperimen skala besar dan otomatisasi eksperimen;
5. Konsep multi-layer untuk menjamin tampilan data yang efisien dan menjamin penanganan data;
6. Memiliki GUI, command line mode, dan Java API yang dapat dipanggil dari program lain. RapidMiner memiliki tampilan antarmuka yang *user friendly* yang memberikan kemudahan kepada pengguna dalam memakainya dan dikenal dengan sebutan *Perspective*. Berikut adalah tampilan aplikasi RapidMiner seperti yang ditunjukan pada gambar 2.9



Gambar 2.8 Tools Rapid Miner

Beberapa fitur RapidMiner antara lain:

1. Banyaknya algoritma data mining, seperti decision tree dan self organization map;
2. Bentuk grafis yang canggih, seperti tumpang tindih diagram histogram, tree chart dan 3D Scatter plots;
3. Banyaknya variasi plugin, seperti text plugin untuk melakukan analisis teks;

4. Menyediakan prosedur data mining dan machine learning termasuk:ETL(extraction, transformation, loading), data preprocessing, visualisasi,modelling dan evaluasi;
5. Proses data mining tersusun atas operator-operator yang nestable, dideskripsikan dengan XML, dan dibuat dengan GUI;
6. Mengintegrasikan proyek data mining Weka dan statistika R.