BAB II
DASAR TEORI

2.1 Jatuh

Jatuh merupakan kejadian yang melibatkan seseorang mendadak terbaring, terduduk dilantai, berada ditempat yang lebih rendah, dengan tanpa sadar dan atau menimbulkan luka. Pada tahun 1987 kelompok kerja internasional Kellog melakukan pencegahan jatuh pada orang tua didefinisikan jatuh sebagai " tidak sengaja datang ke tanah, atau beberapa tingkat yang lebih rendah bukan sebagai konsenkuensinya menahan pukulan keras, kehilangan kesadaran, dan seketika mengalami kelumpuhan seperti pada stroke atau serangan epilepsi". Definisi ini digunakan pada banyak penelitian, dan kemudian diperluas termasuk jatuh akibat pusing dan lainnya. Jatuh sering terjadi atau dialami oleh usia lanjut.

Kecelakaan dan cedera yang diakibatkan oleh terjatuh biasa nya lebih rentang terjadi pada lansia.Banyak pencegahan jatuh dengan menggunakan pendekatan-pendekatan yang telah dikembangkan berkisar dari pengobatan dan penyesuaian lingkungan, menggunakan pengekang fisik sampai bantalan pelindung pinggul dan sistem komersil semacam perlindungan diri untuk mengurangi resiko dari terjatuh.Banyak faktor yang berperan sebagai sebab dari terjadinya jatuh, dapat di bagi dua jenis faktor yaitu pertama faktor intrinsik (faktor dari tubuh) dan kedua faktor ekstrinsik (faktor dari lingkungan sekitar).

a. Factor intrisik :
    1. Penyakit Stroke dan TIA, yang mengakibatkan kelemahan tubuh sesisi.
    2. Parkson yang mengakibatkan kekakuan alat gerak pada tubuh.
    3. Depresi yang menyebabkan lansia tidak lerlalu memperhatikan saat berjalan.
    4. Gangguan penglihatan pun seperti misalnya katarak mengingkatkan resiko jatuh pada lansia.
    5. Gangguan system kardiovaskuler akan menyebabkan *syncope*, *syncope* lah yang serin menyebabkan jatuh pada lansia.
    6. Dehidrasi, bisa disebabkan oleh diare,demam, asupan cairan yang kurang atau penggunaan duretik yang berlebihan.

b. Faktor Ekstrinsik :

1. Alat-alat ataupun perlengkapan yang tergeletak dibawah atau dilantai.
2. Tempat tidur yang tidak stabil.
3. Lantai yang memmiliki tingkat kerendahan yang berbeda.
4. Tempat berpegangan yang tidak kuat atau sulit untuk dipegang.
5. Lantai yang tidak datar,licin, ataupun menurun.
6. Karpet yang tidak dilem dengan baik.
7. Keset yang tebal atau menekuk pinggirnya.
8. Benda-benda alas lantai yang licin atau mudah tergeser.
9. Penerangan yang kurang baik (kurang atau menyilaukan).
10. Alat bantu jalan yang tidak tepat ukuran, berat, maupun cara penggunaannya.

2.2 Klasifikasi

Klasifikasi adalah suatu bentuk analisis data yang mengekstrak model yang menggambarkan kelas data yang penting. Model seperti ini disebut penggolongan , memprediksi label kelas untuk dikategorikan (diskrit, tidak berurutan). Misalnya kita dapat membangun klasifikasi untuk mengkategorikan pinjaman bank aman atau beresiko. Analisis semacam ini dapat membantu kita untuk memahammi data dengan lebik baik. Telah banyak metode klasifikasi yang diajukan oleh para peneliti dalam pembelajaran mesin pengenalan pola dan statistik.

Dapat dibilang klasifikasi dibangun untuk memprediksi kelas (kategori).Klasifikasi data adalah proses dua langkah terdiri dari langkah belajar (dimana klasifikasi dibangun) dan tahap mengklasifikasikan (dimana model ini digunakan untuk memprediksi label kelas untuk data yang diberikan). Pada langkah pertama, klasifikasi dibangun untuk menggambarkan kumpulan data kelas yang telah ditentukan sebelumnya. Konsep ini adalah langkah pembelajaran (atau tahap pelatihan), dimana algoritma klasifikasi membangun classifier dengan menganalisi atau “belajar” dari training data yang terdiri dari data tupel dan label kelas terkait.

Sebuah tupel X diwakili oleh n-dimensi *vector* atribut X () , yang menggambarkan n pengukuran yang dilakukan pada tupel dari n atribut database, masing masing, setiap tupel X diasumsikan termasuk kelas yang telah ditetapkan seperti yang ditentukan oleh atribut database lain yang disebut atribut label kelas. Atribut label kelas bernilai dikrit dan tidak teratur. Kategoris atau nominal dalam setiap nilai tersebut memiliki fungsi sebagai kategori atau

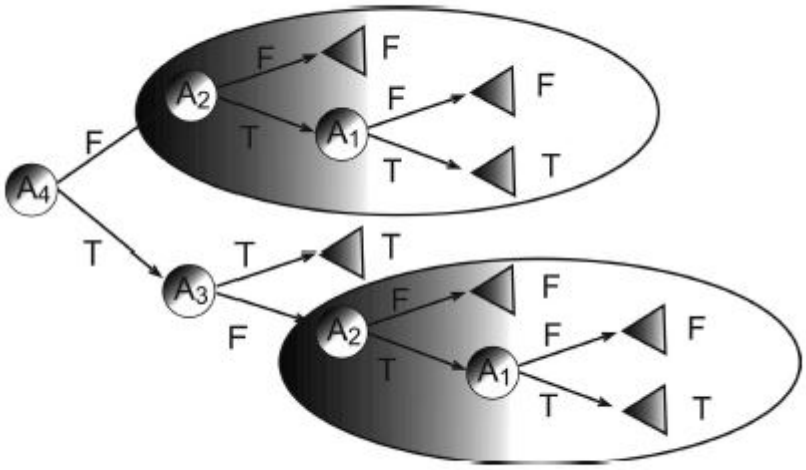
kelas.Tupel individu membuat set training disebut sebagai tupel training dan diambil secara acak dari database yang sedang dianalisis.

2.3 *Decision tree*

Decision tree pertama kali diperkenalkan pada tahun 1960-an oleh Fredkin. Dalam ilmu *computer trie* atau *prefix tree* adalah sebuah struktur data dengan representasi *ordered tree* yang digunakan untuk menyimpan *associative array* yang berupa string. Berbeda dengan *binary search tree* (BTS) yang tidak ada *node* di *tree* yang menyimpan elemen yang berhubungan dengan *node* sebelumnya dan posisi setiap elemen di *tree* sangat menentukan. Semua keturunan dari suatu *node* mempunyai *prefix string* yang mengandung elemen dari *node* itu dengan root merupakan string kosong. *value* biasanya tidak terdapat di setiap *node* melainkan hanya pada daun dan beberapa *node* di tengah yang cocok dengan elemen tertentu. (Han, Kammber, & Pei, 2012)

Decision tree merupakan salah satu metode klasifikasi pada Text Mining. Klasifikasi adalah proses menemukan kumpulan pola atau fungsi-fungsi yang mendeskripsikan dan memisahkan kelas data satu dengan lainnya, untuk dapat digunakan memprediksi data yang belum memiliki kelas data tertentu.

Decision tree adalah sebuah struktur pohon, dimana setiap *node* pohon merepresentasikan atribut yang telah diuji setiap cabang merupakan suatu pembagian hasil uji dan *node* daun (leaf) merepresentasikan kelompok kelas tertentu.Level *node* teratas dari sebuah *decision tree* adalah *node* akar (root) ang biasanya berupa atribut yang paling memiliki pengaruh terbesar pada suatu kelas tertentu. Pada proses mengklasifikasi data yang tidak diketahui,seperti Gambar 2.2.3.1 Struktur *decision tree* nilai atribut akan diuji dengan cara melacak jalur dari *node* akar (*root*) sampai *node* akhir (daun) dan kemudian akan diprediksi kelas yang dimiliki oleh suatu data baru tertentu.



Gambar 2.2.3.1 Struktur *decision tree*
(Rokach & Maimon, DATA MINING WITH DECISION TREES Theory and Applications, 2015)

Decision tree menggunakan metoda *rekursif top down divide and-conquer*. *Top down* disini menunjukan bahwa menggunakan manajemen dimana tindakan dimulai pada tingkat tertinggi lalu menurun kebawah, membagi ke arah yang sesuai dengan kriteria, dan berhenti pada kriteria yang sesuai.

Dalam data mining, *decision tree* adalah model prediktif yang dapat digunakan untuk mewakili kedua pengklasifikasi dan model regresi. Dalam riset operasi, pada sisi alin *decision tree* mengacu pada podel keputusan ihrarki dan konsekuensinya. Pembuatan keputusan menggunakan *decision tree* untuk diidentifikasi strategi yang paling mungkin untuk mencapai tujuannya. Ketika *decision tree* digunakan untuk tugas klasifikasi, itu lebih tepat disebut sebagai *classification tree*. Ketika digunakan untuk regresi makan akan disebut *regression tree.* (Rokach & Maimon, DATA MINING WITH DECISION TREES, 2008)

2.3.1 ***Information Gain***

Ukuran ini berdasarkan pada karya Claude Shannon pada teori informasi, yang mempelajari nilai atau “isi informasi” dari sebuah pesan. Biarkan *node* mewakili atau menahan tupel partisi D. atribut dengan gain informasi tertinggi dipilih sebagai atribut pemisah *node*. Atribut ini meminimalisir informasi yang dibutuhkan untuk mengklasifikasikan tupel dalam partisi yang

dihasilkan dan mencerminkan keacakan atau ketidakamanan terkecil dalam partisi ini. Pendekatan ini meminimalkan jumlah tes yang diharapkan yang diperlukan untuk mengklasifikasikan tupel yang diberikan dan memjamin bahwa *tree* yang sederhana (tapi tidak harus yang paling sederhana) dapat ditemukan.

Informasi yang diharapkan untuk mengklasifikasi tupel di D didapat dari

$$\text{Info(D)} = -\sum_{i=1}^{m} p_i \log_2(pi) \qquad (2.1)$$

Dimana pi adalah probabilitas tak berhingga bahwa tupel acak di D termasuk kelas Ci dan diperkirakan oleh$|Ci, D|/|D|$. Fungsi log ke basis 2 digunakan, karena informasinya dikodekan dalam bit. Info (D) hanyalah jumlah rata-rata informasi yang dibutuhkan untuk mengidentifikasi label kelas tupel di D. informasi yang kita miliki didasarkan pada proporsi tupel dari setiap kelas info (D) juga dikenal sebagai entropi dari D.

Beberapa banyak lagi informasi yang masih kita butuhkan (setelah partisi) untuk sampai pada klasifikasi yang tepat jumlah ini diukur dengan

$$\text{info}_A(D) = \sum_{j=1}^{v} \frac{|Dj|}{|D|} \text{ x info}(D_j) \qquad (2.2)$$

Syarat $\frac{|Dj|}{|D|}$ bertindak sebagai bobot partisi jth. infoA (D) adalah informasi yang diharapkan diperlukan untuk mengklasifikasikan tupel dari D berdasarkan partisi oleh A. semakin kecil informasi yang diharapkan (masih) dibutuhkan, semakin besar kemurnian partisi. Keuntungan informasi didefinisikan sebagai perbedaan antara informasi asli persyaratan (yaitu, hanya berdasarkan proporsi kelas) dan persyaratan baru (yaitu, diperoleh setelah partisi pada A). itu adalah

$$\text{Gain(A)} = \text{info(D)} - \text{info}_A(D) \qquad (2.3)$$

Dengan kata lain, Gain(A) memberitahukan kita ada berapa banyak yang akan diperoleh dengan bercabang pada A. ini penurunan yang diharapkan dalam kebutuhan informasi yang disebabkan oleh mengetahui nilai A. atribut A dengan gain(A), dipilih sebagai pemecahan atribut pada *node* N ini setara dengan mengatakan bahwa kita ingin mempartisi atribut tersebut A yang akan melakukan “klasifikasi terbaik”, sehingga jumlah informasi masih diperlukan untuk menyelesaikan pengklasifikasian tupel minimal(yaitu, infoA(D)).

2.3.2 **Algoritma C4.5**

Algoritma C4.5 merupakan algoritma yang digunakan untuk membentuk pohon keputusan. C4.5 menggunakan metode yang disebut pessimistic *pruning*, yang mirip dengan komplesitas biaya metode yang juga menggunakan estimasi tingkat kesalahan untuk membuat keputusan mengenai pemangkasan *subtree*. Pemangkasan ini tidak menggunakan satu set prune. Sebagai gantinya , pemangkasan ini menggunakan set training untuk memperkirakan tikat kesalahan. Perkiraan akurasi atau kesalahan berdasarkan set training terlalu sering terjadi dan oleh karena itu dapat dibilang sangat bias. Maka metode pemangkasan tersebut dapat menyesuaikan tingkat kesalahan yang diperoleh dari training dan ditetapkan dengan menambahkan penalti, sehingga dapat menanggulangi bias yang terjadi.

Secara umum algoritma C4.5 untuk membangun pohon keputusan adalah sebagai berikut.

a. Pilih atribut sebagai akar.
b. Buat cabang untuk tiap tiap nilai.
c. Bagi kasus dalam cabang.
d. Ulangi proses untuk setiap cabang sampai semua kasus pada cabang memiliki kelas yang sama.

Untuk memilih atribut sebagai akar, didasarkan pada nilai gain tertinggi dari atribut-atribut yang ada. Untuk menghitung gain digunakan rumus seperti tertera dalam persamaan berikut.

$$\text{Gain(S,A)} = \text{Entropy(S)} - \sum_{i=1}^{n} \frac{|Si|}{|S|} \text{ x Entropy(Si)} \qquad (2.4)$$

S : himpunan kasus

A : atribut

N : jumlah partisi atribut A

$|Si|$: jumlah kasus pada partisi ke-i

$|Si|$: jumlah kasus dalam S

Sementara itu, nilai entropi dapat dilihat pada persamaan berikut

$$\text{Entropy(S)} = \sum_{i=1}^{n} -pi \text{ x } \log_2 pi \quad (2.5)$$

Keterangan :

S : himpunan Kasus

A : fitur

N : jumlah partisi S

P_i : proporsi dari Si terhadap S

2.4 Validasi

Validasi dapat diartikan tindakan penilaian suatu parameter tertentu berdasarkan suatu percobaan, untuk membuktikan bahwa parameter yang diberikan memenuhi persyaratan. Dalam validasi dimungkinkan untuk mendisain pekerjaan eksperimen sedemikian rupa agar sesuai dengan karakteristik dapat memberikan pengujian secara keselruhan data mengenai prosedur analitis, seperti *specificity* (spesifikasi) ,*linearity* (linearitas) ,*range* (jarak) ,*accuracy* (akurasi) dan *precision* (presisi). Tujuan dari prosedur analitik harus dipahami dengan tujuan yang jelas karena akan mengatur karakteristik validasi yang perlu dievaluasi.

Validasi umum yang harus dipertimbangkan sebagai berikut :

a. Accuracy
b. Precision
c. Repeatability
d. Intermediate Precision
e. Reproducibity
f. Reproducibility
g. Specificity
h. Detection

2.4.1 **Confusion Matrix**

Confusion Matrix berfungsi untuk mengindikasi sifat-sifat aturan klasifikasi. Mengetahui setiap data yang ada diklasifikasikan dengan benar atau tidak. Utamanya pada jumlah diagonal observasi yang telah diklasifikasikan untuk setiap data klasifikasi, data-data yang tidak termasuk dalam diagonal observasi menunjukkan seberapa banyak data yang diklasifikasikan

dengan tidak sesuai. Salah satu manfaat dari *confusion matrix* adalah melihat, apakah sistem tepat atau tidak dalam mengklasifikasikan antar dua opsi atau lebih (yaitu pada umumnya salah menafsirkan satu nilai sebagai niai yang lain). Untuk setiap set data tes, kita membandingkan kelas yang sebenarnya dengan kelas yang diprediksi oleh *trained classifier*.

Actual	Predicted		
		Positive	Negative
	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

Gambar 2.4.1 *confusion matrix*

Metode ini menggunakan tabel matriks, *dataset* yang diuji dianggap sebagai positif dan yang lainnya negative. Seperti pada Gambar 2.4.1 *confusion matrix* yang mengelompokkan perbandingan data prediksi dengan data hasil menjadi empat kategori : *true positive* (jumlah hasil positif yang diklasifikasikan sebagai positif), *false positive* (jumlah hasil negatif yang diklasifikasikan sebagai positif), *false negative* (jumlah hasil positif yang diklasifikasikan negatif), *true negative* (jumlah hasil negatif yang diklasifikasikan sebagai negatif). Menggunakan perhitungan sebagai berikut (Han, Kammber, & Pei, 2012):

a. Accuracy

Keakuratan prosedur analitik menyatakan kedekatan kesepakatan antara nilai yang diterima baik sebagai nilai nyata kovensional atau nillai referensi yang diterima dan nilai yang ditemukan. *Accuracy* terkadang disebut *trueness*. Untuk menghitung akurasi dapat menggunakan rumus berikut :

$$Accuracy : \left(Sensitifity\ x\ \frac{\text{pos}}{pos+neg}\right) + \left(Sensitifity\ x\ \frac{\text{neg}}{pos+neg}\right) \quad (2.6)$$

b. Precision

Ketepatan prosedur analitik menyyatakan kedekatan perjanjian(tingkat scatter) antara serangkaian pengukuran yang diperoleh dari beberapa contoh yang sama dalam kondisi yang ditentukan. Presisi dapat dipertimbangkan pada tiga tingkat : pengulangan, presisi antara dan

reproduktifitas. Preasisi seharusnya diselidiki menggunakan contoh yang homogeny dan otentik dengan menggunakan ruus berikut :

c. Precision : $\frac{\text{t\_pos}}{t\_pos + f\_pos}$ (2.7)

Namun ,jika itu tidak mungkin diperoleh sampel yang homogen dapat diselidiki menggunakan sampel yang dibuat secara artifisial atau larutan sampel. Ketepatan prosedur analitik biasanya dinyatakan sebagai varians, standar deviasi atau koefisien variasi dari serangkaian pengukuran

d. Specificity

Spesifisitas adalah kemampuan untuk menilai dengan tegas analit di hadapan komponen yang mungkin diharapkan hadir dengan menggunakan rumus berikut :

Specify : $\frac{\text{t\_neg}}{neg}$ (2.8)

Biasanya ini mungkin termasuk kotoran, degradasi, matriks dan lainnya. Kekurangan kekhususan prosedur analitis individu dapat dikompensasi oleh prosedur analitis individu dapat dikompensasi oleh prosedur analitis pendukung lainnya.

e. Sensitifitas

Sensitifitas mengukur poroporsi *true* (positif) yang diidentifikasi secara tepat, seperti spesifitas yang mengukur porporsi *true* (negatif).

Sensitifity(recall) : $\frac{\text{t\_pos}}{pos}$ (2.9)

Pada dasarnya sensitifitas dan speksifitas memiliki peran penting pada klasifikasi data.

Keterangan :

t_pos : jumlah *true positive*

t_neg : Jumlah *true negative*

pos : Jumlah *record positive*

Neg : Jumlah *record negative*

f_neg : Jumlah *false positive*

2.5 RapidMiner

Merupakan perangkat lunak berbasis java sehingga dapat bekerja di semua system operasi dan bersifat terbuka *(open source).* RapidMiner menggunakan berbagai teknik deskriptif dan prediksi dalam memberikan wawasan kepada pengguna sehingga dapat mendapatkan keputusan terbaik. 500 operator data mining kurang lebih terdapat dalam RapidMiner , termasuk operator untuk *input, output* data *preprocessing,* dan visualisasi.

RapidMiner menyediakan GUI (*Graphic User Interface*) untuk merancang sebuah *pipeline analitis*. GUI ini akan menghasilkan file XML (*Extensible Markup Language*) yang mendefenisikan proses analitis keinginan penggunaan untuk diterapkan dalam data. File ini kemudian dibaca oleh RapidMiner untuk menjalankan analisis secara otomatis.

Beberapa fitur yang bias didapatkan dari RapidMiner :

a. Banyaknya algoritma data mining (decision tree, k-NN, Naïve Bayes, dll)
b. Bentuk grafis yang canggih, seperti dapat menampilkan secara bersamaan diagram histogram, tree chart dan 3D Scatter plot.
c. Prosedur data mining dan *machine learning* termasuk: ETL (*extraction, transformation loading*), data preprocecing, visualisasi, modeling dan evaluasi.
d. Memiliki variasi plugin, seperti text plugin untuk melakukan analisis teks.

Proses data mining tersusun atas operator – operator yang *nestable*, dideskripsikan dengan *XML*, dan dibuat dengan *GUI*.

2.5.1 **Binning**

Binning atau diskritisasi adalah proses transformasi variabel numerik menjadi kelompok-kelompok kategori. Variabel numerik biasanya didiskritkan dalam metode pemodelan berdasarkan tabel frekuensi (misalnya, pohon keputusan). Selain itu, Binning dapat memungkinkan idendifikasi mudah dari *outlier*, nilai variabel numerik yang tidak valid dan hilang. (carthheel technologies, 2018)

Binning sendiri terbagi menjadi dua tipe yaitu :

e. Unsupervised Binning

Metode *unsupervised binning* mengubah *variable numeric* menjadi kelompok-kelompok kategoris tetapi tidak menggunakan informasi target(kelas). *Equal Width* dan *Equal Freuency* merupakan dua metode binning yang termasuk dalam *Unspervised Binning*.

1. *Equal Width*

Algoritma membagi data kedalam interval k dengan ukuran yang sama. Rumus untuk lebar interval nya :

$$W = (max-min) / k \quad (2.10)$$

Rumus untuk batas interval :

$$Min+w, min+2w, \ldots, min + (k-1)w \quad (2.11)$$

2. Equel Frequency

Algoritma ini membagi data kedalam kelompok k yang masing-masing kelompok berisi jumlah nilai yang hamper sama. Untuk kedua metode, cara terbaik untuk menentukan k adalah dengan melihat histogram dan mencoba interval atau group yang berbeda.

f. Supervised Binning

Metode supervised binning mengubah variable numeric menjadi kelompok-kelompok kategori dan mengacu pada informasi target (kelas) ketika memilih *discretizeation* untuk memotong point. *Entropy-based* binning adalah contoh metode *supervised binning.*

c. Entropy-based binning

Metode berbasis entropi menggunakan pendekatan split. *Entropi* (atau konten informasi) dihitung berdasarkan label kelas. Secara *intuitif,* metode ini menemukan perpecahan terbaik sehingga nilai bin semurni mungkin, mayoritas nilai-nilai dalam bin sesuai dengan label kelas yang sama. Secara formal, ini ditandai dengan menemukan pepercahan dengan perolehan informasi maksimal.