

Pengembangan Aplikasi Decision Tree Iterative Dichotomiser 3 untuk Klasifikasi Positif atau Negatif Obesitas pada Balita

Dewi Kurnia Anggraeni¹⁾, Yulius Denny Prabowo²⁾

Teknik Informatika, Institut Teknologi dan Bisnis Kalbis
Jalan Pulomas Selatan kav22, Jakarta 13210

¹⁾Email: dekurnianggraeni@gmail.com

²⁾Email: yulius.prabowo@kalbis.ac.id

Abstract: This research aims to create a grouping of children with positive obesity and negative obesity using ID3. ID3 method using entropy and information gain as the calculation equation to get the results of the decision tree, ID3 also do a search recursively to get a decision tree to best fit result because it removes some of the same data when making the decision tree, to get the proper decision tree ID3 using reduced error pruning (REP) and ruel post-pruning (RPP). The results of this application are the classification of positive obesity and negative obesity. This classification was tested using data taken from the village health center Munjul.

Keywords: iterative dichotomiser 3 (ID3), entropy, information gain, obesity

Abstrak: Penelitian ini bertujuan untuk menentukan anak positif obesitas dan negatif obesitas menggunakan metode ID3. Metode ID3 menggunakan entropy dan information gain sebagai persamaan perhitungan untuk mendapatkan hasil pohon keputusan, ID3 juga melakukan pencarian secara rekursif sehingga mendapatkan pohon keputusan dengan hasil best fit karena menghapus beberapa data yang sama saat membuat pohon keputusan. Untuk mendapatkan pohon keputusan yang tepat ID3 menggunakan reduced error pruning (REP) dan ruel post-pruning (RPP). Hasil dari aplikasi ini adalah klasifikasi positif obesitas dan negatif obesitas. Klasifikasi ini diuji menggunakan data yang diambil dari Puskesmas Kelurahan Munjul.

Kata kunci: iterative dichotomiser 3 (ID3), entropy, information gain, obesitas

I. PENDAHULUAN

Di Indonesia angka anak yang mengalami obesitas tidaklah tinggi tetapi mengalami peningkatan setiap tahunnya. Obesitas adalah kondisi di mana tubuh mengalami kelebihan lemak atau kalori. Dalam jangka panjang obesitas pada anak dapat meningkatkan kemungkinan mengidap penyakit yang menyebabkan kematian terbesar di dunia seperti penyakit kardiovaskuler dan diabetes. Obesitas pada anak bisa terjadi karena pola makan yang buruk seperti memilih makanan cepat saji yang memiliki kalori tinggi, memberikan makanan pendamping ASI yang berlebih, kurangnya pengeluaran kalori, perubahan berat badan ibu saat hamil, orang tua yang mengalami obesitas ataupun mengalami diabetes [1].

Mengetahui anak mengalami obesitas dapat dilihat dengan tabel status gizi yang mengacu pada

standar World Health Organization (WHO) ataupun dengan perhitungan Body Mass Index (BMI) namun tidak semua perhitungan kelebihan berat badan dengan BMI disebut obesitas seperti para atlet yang memiliki masa otot yang baik.

Iterative Dichotomiser 3 (ID3) adalah salah satu metode untuk membuat klasifikasi pohon keputusan. Untuk membuat pohon keputusan ID3 membutuhkan data awal untuk di proses, data awal tersebut adalah data masukan dan data keluaran. Data masukan akan dipergunakan untuk membuat akar dan cabang pohon keputusan sedangkan data keluaran dipakai untuk membuat simpul dari pohon keputusan. Untuk mengatasi masalah tersebut maka akan dibuat perhitungan algoritma *decision tree learning* Iterative Dichotomiser 3 (ID3) untuk menentukan klasifikasi positif atau negatif obesitas pada anak. Aplikasi ini membuat pohon keputusan dari data status gizi anak di

Puskesmas Kecamatan Munjul. Pohon keputusan ini akan dijadikan *rules*, *rules* dipakai untuk melihat data baru apakah data tersebut data anak yang mengalami atau yang tidak mengalami obesitas.

Aplikasi yang dibangun ini, memberikan mesin pola belajar untuk menentukan status obesitas pada anak. Parameter yang digunakan dalam aplikasi ini untuk menentukan anak obesitas atau tidak adalah: berat badan, tinggi badan, jenis kelamin, usia. Data yang digunakan diambil dari Puskesmas Kelurahan Munjul pada bulan Januari. Hasil penelitian ini dapat bermanfaat untuk mengetahui data anak mengalami obesitas atau tidak mengalami obesitas dan mengetahui seberapa akurat algoritma ID3 untuk studi kasus obesitas.

II. METODE PENELITIAN

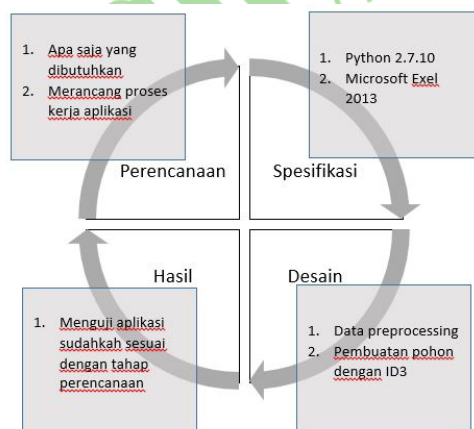
Metode pengembangan perangkat lunak yang digunakan pada penelitian ini adalah prototipe. Prototipe merupakan metode pengembangan aplikasi yang digunakan untuk mewujudkan aspek-aspek dari isi perangkat lunak. Aspek-aspek prototipe adalah teknologi terbaru, kebutuhan dan konten. Aspek teknologi terbaru adalah pemeriksaan perangkat lunak yang dibuat dengan pengetahuan terbaru tim pembuat perangkat lunak [2].

Terdapat empat tahapan metode pengembangan prototipe yang dilakukan yaitu perencanaan, spesifikasi, desain dan hasil. Perencanaan, pada tahapan ini ditentukan apa saja yang dibutuhkan dan tujuan dalam pembuat aplikasi. Dari tahapan ini mendapatkan hasil aplikasi ini dibuat untuk membuat pohon keputusan yang mengklasifikasikan balita yang mengalami atau yang tidak mengalami obesitas.

Dari tahap sebelumnya diperlukan obeservasi untuk mengetahui kriteria obesitas yang dapat digunakan sebagai data awal untuk membuat pohon keputusan, setelah diketahui data tersebut berupa data status gizi balita dengan parameter berat badan, tinggi badan, umur, jenis kelamin dan status gizi yang di ambil dari Puskesmas Kelurahan Munjul. Setelah itu, data tersebut akan diolah dengan algoritma ID3 untuk membuat pohon keputusan. Selain itu dari tahapan ini juga didapatkan hasil berupa rancangan proses kerja aplikasi. Spesifikasi, pada tahapan ini ditentukan alat apa saja yang dibutuhkan untuk membuat aplikasi. Ditahapan ini dihasilkan bahwa dibutuhkannya perangkat keras dan perangkat lunak untuk membuat aplikasi ini. Perangkat keras yang dibutuhkan adalah laptop yang digunakan untuk mengolah data status gizi yang diambil, di mana laptop yang dipakai sudah

di-*install* Python 2.7.10 dan Microsoft Office Excel 2013. Python 2.7.10 digunakan untuk membangun aplikasi di mana Python 2.7.10 menggunakan bahasa pemrograman Python. Microsoft Office Excel 2013 digunakan untuk menyimpan data status gizi balita untuk diolah untuk membuat pohon keputusan.

Desain, dalam tahapan ini dibuat prototipe berdasarkan rancangan proses kerja yang telah dirancang pada tahapan perencanaan. Hasil, pada tahapan ini, hasil prototipe akan dievaluasi untuk mengetahui apakah prototipe sudah sesuai dengan kebutuhan dan keinginan pengguna. Prototipe ini akan dievaluasi baik oleh pengguna maupun pengembang itu sendiri. Setelah dievaluasi, tahapan yang dilakukan pengulang tahapan prototipe apabila prototipe tersebut belum sesuai keinginan pengguna. Tahapan pengembangan perangkat lunak seperti pada Gambar 1.



Gambar 1 Tahapan metode pengembangan perangkat lunak prototipe

A. Obesitas

Obesitas adalah salah satu masalah gizi yang disebabkan kelebihan kalori yang terdapat di dalam tubuh yang sudah tidak diperlukan pada tubuh yang normal. Obesitas pada anak akan menjadi masalah serius dikarenakan jika anak mempunyai berat badan yang berlebih hingga dewasa, anak akan teancam banyak penyakit yang menyebabkan kematian.

Obesitas dapat dihitung dengan *Body Mass Index* (BMI) yaitu dengan menghitung berat badan (kg) dibagi dengan tinggi badan (m) kuadrat. Namun tidak semua kelebihan berat badan dengan BMI yang tinggi disebut obesitas seperti para atlet yang memiliki masa otot yang baik karena latihan-latihan yang diterimanya.

Ada beberapa faktor yang mempengaruhi obesitas itu seperti, berat badan, tinggi badan, jenis kelamin, faktor pengeluaran kalori dan pola tidur. Ada beberapa penyebab obesitas pada anak yaitu,

pola makan buruk seperti pemberian pendamping ASI yang berlebih atau memakan makanan cepat saji, kurangnya berolahraga, dan perubahan perubahan berat badan ibu saat hamil ataupun ibu yang mengalami obesitas juga [1].

B. Machine Learning

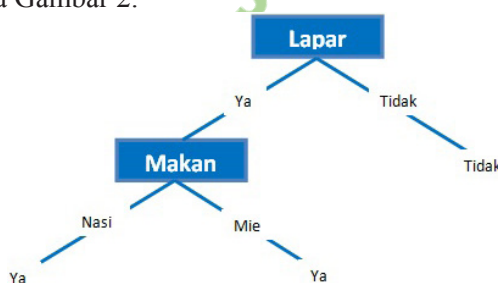
Machine learning adalah salah satu percabangan ilmu dari artificial intelligence (AI) yang berfokus kepada algoritma untuk dapat membuat komputer belajar. Yang algoritmanya terdapat dari data set dan informasi yang terdapat dalam data. Yang nantinya informasi yang dihasilkan bisa digunakan untuk prediksi di kemudian hari. Algoritma yang dipakai sangat bergantung pada matematika dan statistik. Metode ini hanya bisa mengeneralisasi berdasarkan data yang ada dan dengan cara yang terbatas [3].

Machine learning dibagi menjadi dua bagian yaitu [3]: (1) Supervised learning. Supervised learning adalah teknik yang menggunakan input dan output untuk belajar bagaimana memprediksi. Contohnya: neural network, decision trees, support-vector machine, dan Bayesian filtering; dan (2) Unsupervised learning. Berbeda dengan supervised learning, unsupervised learning adalah teknik yang mencari pola dari data yang isi dari data tersebut tidak satupun berisi jawabannya. Contohnya: clustering, non-negative matrix factorization.

C. Decision Tree Learning

Decision tree learning adalah salah satu metode learning dalam machine learning yang mudah dan banyak digunakan. Metode ini mengklasifikasikan data yang nantinya akan mengalami training, setelah itu akan terlihat rules yang jelas dari terbentuknya tree. Decision tree learning ini bisa dipakai untuk klasifikasi ataupun prediksi [4].

Setiap tree mempunyai root dan leaf. Leaf merepresentasikan label kelas. Decision tree ini merupakan metode yang menggunakan pendekatan pencarian fungsi-fungsi yang bernilai diskrit dan terhadap data yang mempunyai kesalahan [5]. Seperti pada Gambar 2.



Gambar 2 Contoh decision tree dalam kasus menentukan seseorang makan atau tidak makan

D. Metode ID3

Salah satu metode decision tree learning yang sering dipakai adalah ID3. Algoritma ini adalah yang paling dasar dalam decision tree learning. Algoritma ini melakukan pencarian dengan menyeluruh (greedy) pada semua kemungkinan pohon keputusan, Karena ID3 memakai fungsi rekursif (fungsi yang memanggil dirinya sendiri) [5].

Node teratas dari sebuah pohon keputusan adalah node akar (root) yang biasanya berupa atribut yang paling memiliki pengaruh. Ada dua perhitungan yang diperlukan untuk menentukan root, yaitu :

1. Entropy

Entropy adalah suatu parameter untuk menghitung keberagaman dari kumpulan sampel data[5].

$$Entropy(S) = \sum_i^c - P_i \log_2 P_i$$

Keterangan :

C = Jumlah nilai pada atribut target

Pi = jumlah sampel pada kelas i

Tabel 1 Contoh data permainan tenis

Cuaca	Angin	Bermain
Cerah	Kencang	Tidak
Cerah	Pelan	Ya
Hujan	Kencang	Tidak
Hujan	Pelan	Tidak
Cerah	Sedang	Tidak
Hujan	Sedang	Ya

Pada contoh Tabel 1 jumlah kelas adalah 2. Kelas 1 ("Ya") terdapat 2 jumlah sampel dan kelas 2 ("Tidak") terdapat 4 sampel. Jadi $p_1 p_1 = 2$ dan $p_2 p_2 = 4$. Maka entropy-nya adalah $Entropy(S) = - (2/6) \log_2 (2/6) - (4/6) \log_2 (4/6) = 0.9183$

Hasil dari perhitungan entropy akan dipergunakan untuk menghitung persamaan information gain.

2. Information Gain

Setelah mengetahui nilai Entropy maka didapat efektivitas suatu atribut dalam mengklasifikasikan data atau disebut information gain [5].

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

Keterangan :

A = atribut.

V = menyatakan suatu nilai yang mungkin untuk atribut A.

Values (A) = himpunan yang mungkin untuk atribut A.

$|S_v|$ = jumlah sampel untuk nilai v.

$|S|$ = jumlah seluruh sampel data.

Entropy (Sv) = Entropy untuk sampel-sampel yang memiliki nilai v.

Pada Tabel 1 terdapat atribut Bermain = "Ya" berjumlah 2 dan atribut Bermain = "Tidak" berjumlah 4. Dari Tabel 1 kita juga dapatkan Values(cuaca) = cerah dan hujan

$$S = [2Y, 4T], |S| = 6$$

$$S_{\text{Cerah}} = [1Y, 2T], |S_{\text{Cerah}}| = 3$$

$$S_{\text{Hujan}} = [1Y, 2T], |S_{\text{Hujan}}| = 3$$

Lalu nilai *information gain* untuk cuaca adalah

$$\text{Entropy}(S) = - (2/6) \log_2(2/6) - (4/6) \log_2(4/6) = 0.9183$$

$$\text{Entropy}(S_{\text{Cerah}}) = - (1/3) \log_2(1/3) - (2/3) \log_2(2/3) = 0.9183$$

$$\text{Entropy}(S_{\text{Hujan}}) = - (1/3) \log_2(1/3) - (2/3) \log_2(2/3) = 0.9183$$

$$\begin{aligned} \text{Gain}(S, \text{Cuaca}) &= \text{Entropy}(S) - (3/6) \text{Entropy}(S_{\text{Cerah}}) - (3/6) \text{Entropy}(S_{\text{Hujan}}) \\ &= 0.9183 - (3/6) 0.9183 - (3/6) 0.9183 \\ &= 0 \end{aligned}$$

Hasil dari perhitungan *information gain* akan dicari nilai yang paling tinggi atau mendekati nilai 1. Nilai *information gain* tersebut akan dijadikan akar atau cabang dalam pembuatan pohon keputusan. Algoritma ID3 dapat diimplementasikan dengan fungsi rekursif, sebagai berikut [5]:

Function ID3 (KumpulanSampel, AtributTarget, KumpulanAtribut)

1. Buat simpul *Root*.

2. If semua sampel adalah kelas i, maka **Return** pohon satu simpul *Root* dengan label = i.

3. If KumpulanAtribut kosong, **Return** pohon satu simpul *Root* dengan label = nilai atribut target yang paling umum (yang paling sering muncul)

Else

A ← Atribut yang merupakan *the best classifier* (dengan *information gain* terbesar)

Atribut keputusan untuk *Root* ← A

For v_i (setiap nilai pada A)

Tambahkan suatu cabang di bawah *Root* sesuai dengan nilai n.

Buat suatu variabel, misalnya $S_{\text{Sampel}_{v_i}}$, sebagai himpunan bagian (subset) dari KumpulanSampel yang bernilai v_i pada atribut A.

If $S_{\text{Sampel}_{v_i}}$ kosong

Then di bawah cabang ini tambahkan suatu simpul daun (*leaf node*, simpul yang tidak anak di

bawahnya) dengan label = nilai atribut target yang paling umum (yang paling sering muncul)

Else di bawah cabang ini tambahkan *subtree* dengan memanggil fungsi ID3(KumpulanSampel, AtributTarget, KumpulanAtribut)

End

End

End

4. Return *Root*

Dalam *decision tree* dibutuhkan *reduced error pruning* (REP) dan *ruel post-pruning* (RPP) untuk membuat pohon yang tepat. REP dibagi menjadi tiga bagian yaitu *training set*, *validation set* dan *test set*. REP akan mengurangi simpul-simpul pada pohon yang nantinya akan menurunkan *error* dan *validation set*. Cara paling sederhana pada metode REP adalah melakukan pembagian sampel ke dalam tiga kelompok secara berulang kali sehingga dapat memilih pembagian sampel data yang menghasilkan pohon dengan tingkat akurasi yang paling tinggi pada ketiga kelompok data tersebut. [5]

Untuk mengatasi kelemahan REP dapat menggunakan metode RPP yang terdiri dari 4 langkah, yaitu [5]: (1) Menggunakan sampel data pada *training set*, untung membangun pohon yang paling sesuai. Dan membiarkan *overfitting* terjadi; (2) Merubah pohon yang dihasilkan menjadi sekumpulan aturan, yang setiap aturan menggambarkan satu jalur pohon dari akar sampai daun; (3) Hapus setiap aturan dengan cara menghilangkan setiap prekondisi yang membuat akurasi menjadi lebih baik; dan (4) Urutkan aturan-aturan tersebut berdasarkan akurasinya. Dan pilihlah aturan-aturan hasil pemangkasan berdasarkan urutannya.

E. Python

Python adalah bahasa pemrograman yang diciptakan oleh Guido van Rossum dan sebagai bahasa skrip yang berorientasi objek. Python dapat berjalan cepat dan diberbagai *platform*. Python lengkap dengan *source code*, *debugger*, *profiler*, dan *graphical user interface* (GUI).

Ada beberapa fitur yang dimiliki python seperti berorientasi objek, dapat juga dibangun dengan Bahasa C/C++, sudah terdapat banyak *library*, memiliki Bahasa yang mudah dipelajari.[6]

F. Alur Proses Aplikasi

Pembangunan aplikasi klasifikasi balita yang mengalami atau yang tidak mengalami obesitas dimulai dengan data *preprocessing*. Data *preprocessing* dilakukan karena data yang didapatkan

dalam bentuk *hardcopy* maka data harus dimasukkan ke dalam bentuk *csv*.

Dalam data *preprocessing* dilakukan penghilangan data yang bernilai Null, pengubahan tipe data pada parameter penentu status gizi, pengelompokan parameter berat badan dan tinggi badan berdasarkan standar WHO. Lalu disimpan ke dalam bentuk file. Tahapan selanjutnya adalah pembuatan pohon dengan algoritma ID3. Proses pembuatan pohon keputusan menggunakan ID3. Dimulai dari perhitungan *entropy* untuk dipakai di dalam perhitungan *information gain*. Seperti pada Gambar 4.

Atribut dengan hasil tertinggi dari nilai *information gain* akan dipergunakan untuk membuat akar jika pohon tersebut belum memiliki akar setelah itu kembali melakukan perhitungan *entropy* dan *information gain*, untuk membuat cabang jika pohon tersebut sudah memiliki akar tetapi belum memiliki cabang dan hasil *information gain* mendekati nilai satu setelah itu diperiksa kembali apakah masih ada kriteria lain jika ya kembali melakukan perhitungan

entropy dan *information* jika tidak akan dibuat simpul jika pohon sudah memiliki akar dan cabang tetapi belum memiliki simpul dan setelah itu dilakukan kembali perhitungan *entropy* dan *information*.

Pembuatan algoritma ID3 dilakukan secara rekursif untuk membuat pohon keputusan. Jadi dalam ID3 akan dilakukan perhitungan *information gain* lalu dari perhitungan tersebut diambil nilai *information gain* tertinggi atau mendekati nilai satu untuk dijadikan sebuah *root* setelah itu akan dilakukan perhitungan secara berulang menentukan cabang setiap pohon.

Perulangan terus terjadi sampai semua data kriteria menjadi sebuah *decision tree*. Setelah terbentuk *decision tree* akan dibuat sebuah *rules*. Dan ketika pengguna menggunakan program ini kriteria *rules* akan dipakai untuk menentukan positif atau negatif obesitas.

G. Pengujian

Aplikasi ini diuji dalam bentuk prototipe yang telah dibuat pada tahapan sebelumnya. Aplikasi ini diuji dengan cara memberikan *input* berupa data balita yang ingin diketahui status gizinya apakah mengalami atau tidak mengalami obesitas. Jika hasil dari prototipe belum memenuhi apa yang sudah ditentukan oleh pengguna, maka tahapan pembuatan prototipe harus dilakukan kembali. Hal ini dilakukan secara berulang sampai prototipe yang dibuat sudah memenuhi kebutuhan dan batasan yang ada.

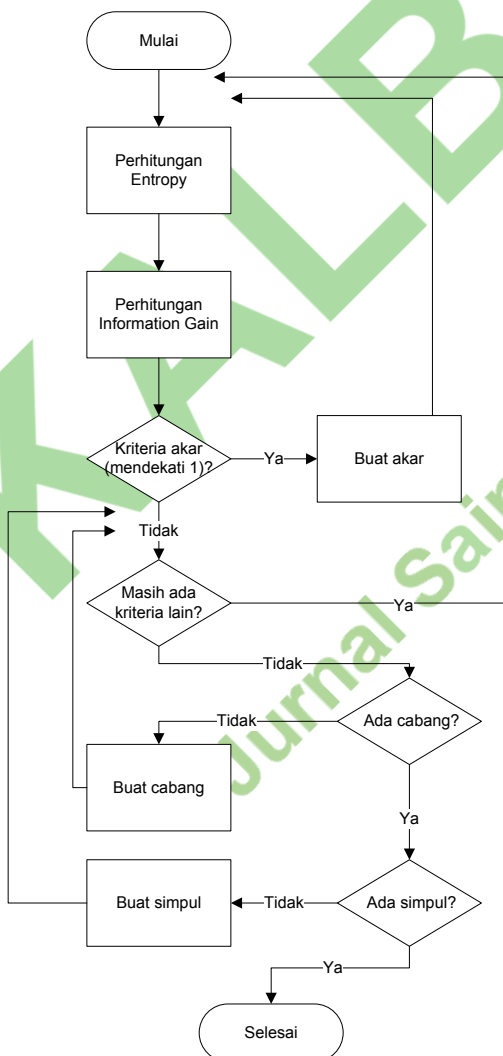
III. HASIL DAN PEMBAHASAN

Aplikasi klasifikasi obesitas positif atau obesitas negatif pada balita menggunakan bahasa pemrograman python. Untuk penyimpanan data, aplikasi ini menggunakan *Microsoft Office Excel* dengan ekstensi *file csv* dan ekstensi *file file*. Aplikasi ini akan dibangun menggunakan beberapa tahapan sebagai berikut:

A. Data Preprocessing

Pembuatan aplikasi ini dibutuhkan *data preprocessing* karena data yang didapatkan terdapat data yang bernilai Null seperti pada Tabel 2 dan beberapa data kriteria yang harus diolah terlebih dahulu sebelum dipergunakan. Data yang bernilai Null akan dihilangkan karena dalam data tersebut tidak banyak dan tidak dapat digunakan. Seperti pada Tabel 2-Tabel4.

Data yang telah dilakukan penghilangan nilai Null, data di masukkan ke dalam *list*. Ada dua *list*



Gambar 4 Proses pembuatan dalam pohon keputusan dengan ID3

Tabel 2 Contoh sebagian data sebelum preprocessing

Umur (Bulan)	Jenis kelamin	Berat Badan (kg)	Tinggi Badan (cm)	Obesitas (Positif/Negatif)
32	P	12.5	92	Negatif
4	P	16.2	102	Negatif
29	P	11.1	90	Negatif
4	L	6.1	66	Negatif
43	L	14.1	99	Negatif
30	L	10.3	94	Negatif
20	P	9	83	Negatif
17	L	8.3	80	Negatif
7	L	7.6	Null	Negatif
31	L	12	80	Negatif
54	L	26.2	105	Positif
48	P	22	106	Positif
55	L	23.1	108	Positif

Tabel 3 Contoh sebagian data standar berat badan sesuai umur

Umur (Bln)	Berat Badan (Kg) Laki-laki						
	-3 SD	-2 SD	-1 SD	Med -ian	1 SD	2 SD	3 SD
0	2.1	2.5	2.9	3.3	3.9	4.4	5
1	2.9	3.4	3.9	4.5	5.1	5.8	6.6
2	3.8	4.3	4.9	5.6	6.3	7.1	8
3	4.4	5	5.7	6.4	7.2	8	9
4	4.9	5.6	6.2	7	7.8	8.7	9.7
5	5.3	6	6.7	7.5	8.4	9.3	10.4

Tabel 4 Contoh sebagian data standar tinggi badan sesuai umur

Umur (Bln)	Tinggi Badan (cm)						
	-3 SD	-2 SD	-1 SD	Med -ian	1 SD	2 SD	3 SD
0	44.2	46.1	48	49.9	51.8	53.7	55.6
1	43.9	50.8	52.8	54.7	56.7	58.6	60.6
2	52.4	54.4	56.4	58.4	60.4	62.4	64.4
3	55.3	57.3	59.4	61.4	63.5	65.5	67.6
4	57.6	59.7	61.8	63.9	66	68	70.1
5	59.6	61.7	63.8	65.9	68	70.1	72.2

yang digunakan, pertama *list* yang berisi data per kolom dan yang kedua adalah *list* yang berisi data per baris. Pembuatan *list* untuk mempermudah saat membentuk *range* nilai atribut tinggi badan dan berat badan. Dalam berat badan dan tinggi badan akan dikelompokkan berdasarkan standarisasi berat badan dan tinggi badan menurut umur dan jenis kelamin berdasarkan standar World Health Organization.

Range dibuat hanya untuk obesitas atau tidak obesitas maka *range* yg digunakan adalah *range* yang dibuat dengan batas bawah -3SD dan batas atas 2SD untuk yang tidak obesitas. Untuk yang obesitas dengan batas bawah 2SD dan batas atas 3SD, dimana baris 3SD adalah tabel untuk obesitas, baris 1SD - 2SD tabel untuk gemuk, baris Median untuk normal, baris -2SD sampai -1SD untuk kurus dan baris -3SD untuk anak yang sangat kurus. SD dalam Tabel 3 dan Tabel 4 adalah standar deviasi.

Proses akhir *data preprocessing* adalah melakukan penyimpanan ke dalam file text dengan format dot file.

B. Perhitungan Algoritma dan Pembuatan Pohon Keputusan

Entropy digunakan untuk mengukur parameter homogenitas dari suatu sampel data. Yang dapat dihitung dengan perhitungan

$$Entropy (S) = \sum_i^c - P_i \log_2 P_i$$

S adalah banyaknya data positif dan negatif dari 50 contoh data. Nilai positif adalah nilai status obesitas = Positif, sedangkan nilai negatif adalah nilai status obesitas = Negatif.

Banyaknya data yang digunakan 35 data negatif dan 15 data positif, yang ditulis dengan notasi [15+, 35-].

$$Entropy (S) = - (15/50) \log_2 (15/50) - (35/50) \log_2 (35/50) = 0,88129089923$$

Information Gain adalah ukuran efektivitas suatu atribut dalam mengklasifikasikan data dengan perhitungan.

$$Gain (S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

Nilai Berat Badan = Karena setiap nilai berat badan yang bernilai 4.9-8.7, 9.4-16.9, >=19.5, >=22.9 frekuensi . Perhitungan *entropy* akan dicontohkan dari sebagian nilai berat badan.

$$S_{9.1-17.1} = [0+, 1-], |S_{9.1-17.1}|=1$$

$$S_{>=24.9} = [1+, 0-], |S_{>=24.9}|=1$$

$$S_{4.9-8.7} = [0+, 2-], |S_{4.9-8.7}|=2$$

$$S_{>=19.5} = [2+, 0-], |S_{>=19.5}|=2$$

$$Entropy S_{9.1-17.1} = [0+, 1-]$$

$$= - (0/1) \log_2 (0/1) - (1/1) \log_2 (1/1)$$

$$= 0$$

$$Entropy S_{>=24.9} = [1+, 0-]$$

$$= - (1/1) \log_2 (1/1) - (0/1) \log_2 (0/1)$$

$$= 0$$

Tabel 5 Contoh beberapa data

Umur (Bulan)	Jenis Kelamin	Berat Badan (kg)	Tinggi Badan (cm)	Obesitas (Positif/Negatif)
32	P	9.1-17.1	81.3-99.4	Negatif
48	P	10.9-21.5	89.8-111.3	Negatif
29	P	8.8-16.2	79.5-96.9	Negatif
4	L	4.9-8.7	57.6-68	Negatif
43	L	10.7-20	8.4-108.5	Negatif
30	L	9.4-16.9	81.7-98.7	Negatif
20	P	7.5-13.7	73.7-88.7	Negatif
17	L	7.7-13.4	73.3-86.5	Negatif
41	P	10.2-19.5	86.3-106.4	Negatif
47	P	10.8-21.2	89.3-110.7	Negatif
51	P	11.2-22.4	91.2-113.3	Negatif
33	P	9.3-17.3	81.9-100.3	Negatif
7	P	5.3-9.8	60.3-71.9	Negatif
4	L	4.9-8.7	>68	Negatif
15	P	6.7-12.4	>83	Negatif
31	L	9.5-19.3	<82.3	Negatif
49	L	11.3-21.4	91.2-112.4	Negatif
53	L	>=22.4	93-114.9	Negatif
27	L	9-16.17	9.9-96.1	Negatif
13	P	6.4-11.8	67.3-80.5	Negatif
8	P	5.6-10.2	>73.5	Negatif
22	P	7.8-14.3	75.2-90.8	Negatif
30	L	9.4-16.9	81.7-98.7	Negatif
40	L	10.4-19.3	87-106.4	Negatif
59	L	12.3-23.9	95.6-118.6	Negatif
52	P	11.3-22.6	91.7-114	Negatif
17	P	7-12.9	71.1-85.4	Negatif
13	L	7.1-12.3	69.6-81.8	Negatif
44	P	10.5-20.4	87.9-108.6	Negatif
51	L	11.5-21.9	92.1-113.6	Negatif
23	P	7.9-14.6	76-91.8	Negatif
1	P	2.7-5.5	47.8-57.6	Negatif
46	P	10.7-20.9	88.9-110	Negatif
48	L	11.2-21.2	90.7-111.7	Negatif
41	L	>=19.5	87.5-107.1	Positif
49	L	>=21.4	91.2-112.4	Positif
52	L	>=22.2	>114.2	Positif
60	P	>=24.9	95.2-118.9	Positif
45	L	>=20.5	89.4-109.8	Positif
41	L	>=19.5	87.5-107.1	Positif
60	L	>=24.2	>119.2	Positif
53	P	>=22.9	92.1-114.6	Positif
42	P	>=19.8	>107.2	Positif
21	L	>=14.5	>90.9	Positif
42	L	>=19.7	>107.8	Positif
43	P	>=20.1	87.4-107.9	Positif
54	L	>=22.7	93.4-115.5	Positif
47	P	>=21.2	89.3-110.7	Positif
55	L	>=22.9	93.9-116.1	Positif
50	L	11.4-21.7	91.6-113	Negatif

$$\begin{aligned} \text{Entropy } S_{4.9-8.7} &= [0+, 2-] \\ &= - (0/2) \log_2 (0/2) - (2/2) \log_2 (2/2) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Entropy }_{>=19.5} &= [2+, 0-] \\ &= - (2/2) \log_2 (2/2) - (0/2) \log_2 (0/2) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Gain Berat Badan} &= \text{Gain}(S, \text{Berat Badan}) \\ &= \text{Entropy}(S) - (((1/50) \text{Entropy}(S_{\text{beratbadan}}))^*42) - \\ &(((2/50) \text{Entropy}(S_{\text{beratbadan}}))^*8) = 0,88129089923 \end{aligned}$$

Nilai Umur = nilai umur 48, 4, 43, 30, 17, 47, 51, 49, 53, 13, 52, 42, 60 memiliki frekuensi 2, nilai umur 41 memiliki frekuensi 3 dan nilai umur lainnya memiliki frekuensi 1.

$$\begin{aligned} S_{32} &= [0+, 1-], |S_{32}|=1 \\ S_{45} &= [1+, 0-], |S_{45}|=1 \\ S_{47} &= [1+, 1-], |S_{47}|=2 \\ S_4 &= [0+, 2-], |S_4|=2 \\ S_{60} &= [2+, 0-], |S_{60}|=2 \\ S_{41} &= [2+, 1-], |S_{41}|=3 \end{aligned}$$

$$\begin{aligned} \text{Entropy } S_{32} &= [0+, 1-] \\ &= - (0/1) \log_2 (0/1) - (1/1) \log_2 (1/1) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Entropy } S_{45} &= [1+, 0-] \\ &= - (1/1) \log_2 (1/1) - (0/1) \log_2 (0/1) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Entropy } S_{47} &= [1+, 1-] \\ &= - (1/2) \log_2 (1/2) - (1/2) \log_2 (1/2) \\ &= 1 \end{aligned}$$

$$\begin{aligned} \text{Entropy } S_4 &= [0+, 2-] \\ &= - (0/2) \log_2 (0/2) - (2/2) \log_2 (2/2) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Entropy } S_{60} &= [2+, 0-] \\ &= - (2/2) \log_2 (2/2) - (0/2) \log_2 (0/2) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Entropy } S_{41} &= [2+, 1-] \\ &= - (2/3) \log_2 (2/3) - (1/3) \log_2 (1/3) \\ &= 0,91829583405 \end{aligned}$$

$$\begin{aligned} \text{Gain Umur} &= \text{Gain}(S, \text{Umur}) = \text{Entropy}(S) - \\ &(((1/50) \text{Entropy}(S_{\text{umur}})) - ((1/50) \text{Entropy}(S_{\text{umur}})) - \\ &(((1/50) \text{Entropy}(S_{\text{umur}}))^*48) = 0,842924982549 \end{aligned}$$

Nilai Jenis Kelamin = Laki-laki, Perempuan

$$\begin{aligned} S_{\text{laki-laki}} &= [10+, 16-], |S_{\text{laki-laki}}|=26 \\ S_{\text{perempuan}} &= [5+, 19-], |S_{\text{perempuan}}|=24 \end{aligned}$$

$$\begin{aligned} \text{Entropy } S_{\text{laki-laki}} &= [10+, 16-] \\ &= - (10/26) \log_2 (10/26) - (16/26) \log_2 (16/26) \\ &= 0,96123660472 \end{aligned}$$

$$\begin{aligned} \text{Entropy } S_{\text{perempuan}} &= [5+, 19-] \\ &= - (5/24) \log_2 (5/24) - (19/24) \log_2 (19/24) \\ &= 0,73828486614 \end{aligned}$$

$$\begin{aligned} \text{Gain Jenis Kelamin} &= \text{Gain}(S, \text{Jenis Kelamin}) \\ &= \text{Entropy}(S) - (26/50) \text{Entropy}(S_{\text{laki-laki}}) - (24/50) \end{aligned}$$

$$\begin{aligned} \text{Entropy}(S_{\text{perempuan}}) & \\ &= 0,0270711290328 \end{aligned}$$

Nilai Tinggi Badan=karena nilai tinggi badan bernilai kontinu maka ada beberapa data yang memiliki frekuensi sama dengan 1 dan frekuensi tertinggi bernilai sama dengan 2.

$$\begin{aligned} \text{Entropy } S_{89.3-110.7} &= [1+, 1-], |S_{89.3-110.7}| = 2 \\ \text{Entropy } S_{81.7-98.7} &= [0+, 2-], |S_{81.7-98.7}| = 2 \\ \text{Entropy } S_{89.3-110.7} &= [1+, 1-] \\ &= - (1/2) \log_2 (1/2) - (1/2) \log_2 (1/2) \\ &= 0,22575424759 \end{aligned}$$

$$\begin{aligned} \text{Entropy } S_{81.7-98.7} &= [0+, 2-] \\ &= - (0/2) \log_2 (0/2) - (2/2) \log_2 (2/2) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Gain Tinggi Badan} &= \text{Gain}(S, \text{Tinggi Badan}) = \\ \text{Entropy}(S) - ((1/50) \text{Entropy}(S_{\text{tinggibadan}})) * 2 - ((1/50) \\ \text{Entropy}(S_{\text{tinggibadan}})) * 48 &= 0,87677581427 \end{aligned}$$

Setelah semua nilai *information gain* diketahui, maka nilai tertinggi yang mendekati angka 1 adalah nilai *information gain* pada atribut berat badan. Dengan begitu atribut berat badan dijadikan sebagai akar pohon keputusan. Untuk menghitung simpul pohon selanjutnya dihitung dengan nilai berat badan. Karena nilai berat badan bernilai kontinu maka akan dicontohkan dengan nilai berat badan 7.5-13.7 dan ≥ 19.5

$$\begin{aligned} \text{Entropy Berat Badan (7.5-13.7)} &= [0+, 1-] \\ &= - (0/1) \log_2 (0/1) - (1/1) \log_2 (1/1) \\ &= 0 \end{aligned}$$

Nilai Umur = 32, 29, 20, 33, 7, 15, 31, 27, 8, 22, 40, 59, 50, 55, 54, 21, 45, 46, 1, 23, 44, 48, 4, 43, 30, 17, 47, 51, 49, 53, 13, 52, 42, 60, 41

$$\begin{aligned} S_{20} &= [0+, 1-], |S_{20}| = 1 \\ S_{45} &= [0+, 0-], |S_{45}| = 0 \\ \text{Entropy } S_{20} &= [0+, 1-] \\ &= - (0/1) \log_2 (0/1) - (1/1) \log_2 (1/1) \\ &= 0 \end{aligned}$$

$$\text{Entropy } S_{45} = [0+, 0-]$$

Karena entropi S_{45} tidak memiliki data positif dan negatif maka hasilnya tidak dipergunakan.

$$\text{Gain Umur} = \text{Gain}(S, \text{Umur}) = \text{Entropy}(S) - ((1/1) \text{Entropy}(S_{\text{umur}})) - (((0/1) \text{Entropy}(S_{\text{umur}})) * 49) = 0$$

Nilai Jenis Kelamin = Laki-laki, Perempuan

$$\begin{aligned} S_{\text{laki-laki}} &= [0+, 0-], |S_{\text{laki-laki}}| = 0 \\ S_{\text{perempuan}} &= [0+, 1-], |S_{\text{perempuan}}| = 1 \\ \text{Entropy } S_{\text{laki-laki}} &= [0+, 0-] \end{aligned}$$

Karena entropi $S_{\text{laki-laki}}$ tidak memiliki data positif dan negatif maka hasilnya tidak dipergunakan.

$$\begin{aligned} \text{Entropy } S_{\text{perempuan}} &= [0+, 1-] \\ &= - (0/1) \log_2 (0/1) - (1/1) \log_2 (1/1) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Gain Jenis Kelamin} &= \text{Gain}(S, \text{Jenis Kelamin}) = \text{Entropy}(S) - (0/1) \text{Entropy}(S_{\text{laki-laki}}) - (1/1) \\ \text{Entropy}(S_{\text{perempuan}}) &= 0 \end{aligned}$$

Nilai Tinggi Badan = karena nilai tinggi badan bernilai kontinu maka ada beberapa data yang memiliki frekuensi sama dengan 1 dan frekuensi tertinggi bernilai 2.

$$\begin{aligned} \text{Entropy } S_{73.7-88.7} &= [0+, 1-], |S_{73.7-88.7}| = 1 \\ \text{Entropy } S_{81.7-98.7} &= [0+, 0-], |S_{81.7-98.7}| = 0 \\ \text{Entropy } S_{73.7-88.7} &= [0+, 1-] \\ &= - (0/1) \log_2 (0/1) - (1/1) \log_2 (1/1) \\ &= 0 \end{aligned}$$

$$\text{Entropy } S_{81.7-98.7} = [0+, 0-]$$

Karena entropi $S_{81.7-98.7}$ tidak memiliki data positif dan negatif maka hasilnya tidak dipergunakan.

$$\begin{aligned} \text{Gain Tinggi Badan} &= \text{Gain}(S, \text{Tinggi Badan}) = \\ \text{Entropy}(S) - ((1/1) \text{Entropy}(S_{\text{tinggibadan}})) - ((0/1) \\ \text{Entropy}(S_{\text{tinggibadan}})) * 49 &= 0 \end{aligned}$$

$$\begin{aligned} \text{Entropy Berat Badan } (\geq 19.5) &= [2+, 0-] \\ &= - (2/2) \log_2 (2/2) - (0/2) \log_2 (0/2) \\ &= 0 \end{aligned}$$

Nilai Umur = nilai umur 48, 4, 43, 30, 17, 47, 51, 49, 53, 13, 52, 42, 60 memiliki frekuensi 2, nilai umur 41 memiliki frekuensi 3 dan nilai umur lainnya memiliki frekuensi 1.

$$\begin{aligned} S_{41} &= [0+, 1-], |S_{41}| = 1 \\ S_{45} &= [0+, 0-], |S_{45}| = 0 \\ \text{Entropy } S_{41} &= [2+, 0-] \\ &= - (2/2) \log_2 (2/2) - (0/2) \log_2 (0/2) \\ &= 0 \end{aligned}$$

$$\text{Entropy } S_{45} = [0+, 0-]$$

Karena entropi S_{45} tidak memiliki data positif dan negatif maka hasilnya tidak dipergunakan.

$$\text{Gain Umur} = \text{Gain}(S, \text{Umur}) = \text{Entropy}(S) - ((2/2) \text{Entropy}(S_{\text{umur}})) - (((0/2) \text{Entropy}(S_{\text{umur}})) * 49) = 0$$

Nilai Jenis Kelamin = Laki-laki, Perempuan

$$\begin{aligned} S_{\text{laki-laki}} &= [2+, 0-], |S_{\text{laki-laki}}| = 2 \\ S_{\text{perempuan}} &= [0+, 0-], |S_{\text{perempuan}}| = 0 \end{aligned}$$

$$\begin{aligned} \text{Entropy } S_{\text{laki-laki}} &= [2+, 0-] \\ &= - (2/2) \log_2 (2/2) - (0/2) \log_2 (0/2) \\ &= 0 \end{aligned}$$

$$\text{Entropy } S_{\text{perempuan}} = [0+, 0-]$$

Karena entropi $S_{\text{perempuan}}$ tidak memiliki data positif dan negatif maka hasilnya tidak dipergunakan.

$$\begin{aligned} \text{Gain Jenis Kelamin} &= \text{Gain}(S, \text{Jenis Kelamin}) = \text{Entropy}(S) - (0/1) \text{Entropy}(S_{\text{laki-laki}}) - (1/1) \\ \text{Entropy}(S_{\text{perempuan}}) &= 0 \end{aligned}$$

Nilai Tinggi Badan = karena nilai tinggi badan bernilai kontinu maka hanya ada beberapa data yang memiliki frekuensi sama dengan 1 dan frekuensi tertinggi bernilai 2.

$$\begin{aligned} \text{Entropy } S_{87.5-107.1} &= [0+, 1-], |S_{87.5-107.1}| = 1 \\ \text{Entropy } S_{81.7-98.7} &= [0+, 0-], |S_{81.7-98.7}| = 0 \\ \text{Entropy } S_{87.5-107.1} &= [2+, 0-] \\ &= - (2/2) \log_2 (2/2) - (0/2) \log_2 (0/2) \\ &= 0 \end{aligned}$$

$$\text{Entropy } S_{81.7-98.7} = [0+, 0-]$$

Karena entropi $S_{81.7-98.7}$ tidak memiliki data positif dan negatif maka hasilnya tidak dipergunakan.

$$\begin{aligned} \text{Gain Tinggi Badan} &= \text{Gain}(S, \text{Tinggi Badan}) \\ &= \text{Entropy}(S) - (((1/1) \text{Entropy}(S_{\text{tinggibadan}})) - (((0/1) \text{Entropy}(S_{\text{tinggibadan}})) * 49) = 0 \end{aligned}$$

Karena semua nilai *gain*-nya sama maka akan dipilih secara acak nilai untuk cabang selanjutnya disini sebagai contoh akan diambil nilai tinggi badan sebagai cabang selanjutnya. Lalu nilai "Negatif" dapat disimpulkan sebagai node di bawah atribut berat badan. Maka selanjutnya akan dihitung untuk *node* dibawah atribut tinggi badan.

$$\begin{aligned} \text{Entropy Tinggi Badan (89.3-110.7)} &= [1+, 1-] \\ &= - (1/2) \log_2 (1/2) - (1/2) \log_2 (1/2) \\ &= 1 \end{aligned}$$

Nilai Umur = nilai umur 48, 4, 43, 30, 17, 47, 51, 49, 53, 13, 52, 42, 60 memiliki frekuensi 2, nilai umur 41 memiliki frekuensi 3 dan nilai umur lainnya memiliki frekuensi 1.

$$\begin{aligned} S_{47} &= [1+, 1-], |S_{47}|=2 \\ S_{44} &= [0+, 0-], |S_{44}|=0 \\ \text{Entropy } S_{47} &= [1+, 1-] \\ &= - (1/2) \log_2 (1/2) - (1/2) \log_2 (1/2) \\ &= 1 \end{aligned}$$

$$\text{Entropy } S_{44} = [0+, 0-]$$

Karena entropi S_{44} tidak memiliki data positif dan negatif maka hasilnya tidak dipergunakan.

$$\begin{aligned} \text{Gain Umur} &= \text{Gain}(S, \text{Umur}) = \text{Entropy}(S) - \\ &(((2/2) \text{Entropy}(S_{47})) - (((0/2) \text{Entropy}(S_{\text{umur}})) * 49) \\ &= 1 \end{aligned}$$

Nilai Jenis Kelamin = Laki-laki, Perempuan

$$\begin{aligned} S_{\text{laki-laki}} &= [0+, 0-], |S_{\text{laki-laki}}|=0 \\ S_{\text{perempuan}} &= [1+, 1-], |S_{\text{perempuan}}|=2 \\ \text{Entropy } S_{\text{laki-laki}} &= [0+, 0-] \end{aligned}$$

Karena entropi $S_{\text{laki-laki}}$ tidak memiliki data positif dan negatif maka hasilnya tidak dipergunakan.

$$\begin{aligned} \text{Entropy } S_{\text{perempuan}} &= [1+, 1-] \\ &= - (1/2) \log_2 (1/2) - (1/2) \log_2 (1/2) \\ &= 1 \end{aligned}$$

$$\begin{aligned} \text{Gain Jenis Kelamin} &= \text{Gain}(S, \text{Jenis Kelamin}) \\ &= \text{Entropy}(S) - (0/2) \text{Entropy}(S_{\text{laki-laki}}) - (2/2) \text{Entropy}(S_{\text{perempuan}}) = 1 \end{aligned}$$

$$\begin{aligned} \text{Entropy tinggi badan (95.2-118.9)} &= [1+, 0] \\ &= - (1/1) \log_2 (1/1) - (0/1) \log_2 (0/1) \\ &= 0 \end{aligned}$$

Nilai Umur = nilai umur 48, 4, 43, 30, 17, 47, 51, 49, 53, 13, 52, 42, 60 memiliki frekuensi 2, nilai umur 41 memiliki frekuensi 3 dan nilai umur lainnya memiliki frekuensi 1

$$\begin{aligned} S_{60} &= [0+, 1-], |S_{60}|=1 \\ S_{45} &= [0+, 0-], |S_{45}|=0 \\ \text{Entropy } S_{60} &= [1+, 0-] \\ &= - (1/1) \log_2 (1/1) - (0/1) \log_2 (0/1) \\ &= 0 \end{aligned}$$

$$\text{Entropy } S_{45} = [0+, 0-]$$

Karena entropi S_{45} tidak memiliki data positif dan negatif maka hasilnya tidak dipergunakan.

$$\begin{aligned} \text{Gain Umur} &= \text{Gain}(S, \text{Umur}) = \text{Entropy}(S) - \\ &(((2/2) \text{Entropy}(S_{\text{umur}})) - (((0/2) \text{Entropy}(S_{\text{umur}})) * 49) \\ &= 0 \end{aligned}$$

Nilai Jenis Kelamin = Laki-laki, Perempuan

$$\begin{aligned} S_{\text{laki-laki}} &= [0+, 0-], |S_{\text{laki-laki}}|=0 \\ S_{\text{perempuan}} &= [0+, 1-], |S_{\text{perempuan}}|=1 \\ \text{Entropy } S_{\text{laki-laki}} &= [0+, 0-] \end{aligned}$$

Karena entropi $S_{\text{laki-laki}}$ tidak memiliki data positif dan negatif maka hasilnya tidak dipergunakan.

$$\begin{aligned} \text{Entropy } S_{\text{perempuan}} &= [0+, 1-] \\ &= - (0/1) \log_2 (0/1) - (0/1) \log_2 (0/1) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Gain Jenis Kelamin} &= \text{Gain}(S, \text{Jenis Kelamin}) \\ &= \text{Entropy}(S) - (0/1) \text{Entropy}(S_{\text{laki-laki}}) - (1/1) \text{Entropy}(S_{\text{perempuan}}) = 0 \end{aligned}$$

Karena nilai yang mendekati 1 adalah gain dari Tinggi Badan (89.3-110.7) yang mendapatkan nilai "Negatif" dan untuk cabang selanjutnya adalah nilai jenis kelamin. Dan perhitungannya adalah

$$\begin{aligned} \text{Entropy Jenis Kelamin (Laki-Laki)} &= [10+, 16-] \\ &= - (1/2) \log_2 (1/2) - (1/2) \log_2 (1/2) \\ &= 1 \end{aligned}$$

Nilai Umur = nilai umur 48, 4, 43, 30, 17, 47, 51, 49, 53, 13, 52, 42, 60 memiliki frekuensi 2, nilai umur 41 memiliki frekuensi 3 dan nilai umur lainnya memiliki frekuensi 1

$$\begin{aligned} S_{47} &= [1+, 1-], |S_{32}|=2 \\ S_{44} &= [0+, 0-], |S_{45}|=0 \\ \text{Entropy } S_{32} &= [0+, 1-] \\ &= - (1/2) \log_2 (1/2) - (1/2) \log_2 (1/2) \\ &= 1 \end{aligned}$$

$$\text{Entropy } S_{45} = [0+, 0-]$$

Karena entropi S_{45} tidak memiliki data positif dan negatif maka hasilnya tidak dipergunakan.

$$\begin{aligned} \text{Gain Umur} &= \text{Gain}(S, \text{Umur}) = \text{Entropy}(S) - \\ &(((2/2) \text{Entropy}(S_{47})) - (((0/2) \text{Entropy}(S_{\text{umur}})) * 49) \\ &= 1 \end{aligned}$$

Nilai Umur = nilai umur 48, 4, 43, 30, 17, 47, 51, 49, 53, 13, 52, 42, 60 memiliki frekuensi 2, nilai umur 41 memiliki frekuensi 3 dan nilai umur lainnya memiliki frekuensi 1

$$\begin{aligned} S_{60} &= [1+, 0-], |S_{32}|=1 \\ S_{40} &= [0+, 1-], |S_{45}|=1 \\ \text{Entropy } S_{41} &= [1+, 0-] \\ &= - (1/1) \log_2 (1/1) - (0/1) \log_2 (0/1) \\ &= 0 \\ \text{Entropy } S_{45} &= [0+, 1-] \\ &= - (0/1) \log_2 (0/1) - (1/1) \log_2 (1/1) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Gain Umur} &= \text{Gain}(S, \text{Umur}) = \text{Entropy}(S) - \\ &((2/2) \text{Entropy}(S_{\text{umur}})) - (((0/2) \text{Entropy}(S_{\text{umur}})) * 49) \\ &= 0,961236604720 \end{aligned}$$

$$\begin{aligned} \text{Entropy Jenis Kelamin(Perempuan)} &= [5+, 19-] \\ &= - (5/24) \log_2 (5/24) - (19/24) \log_2 (19/24) \\ &= 0,73828486614 \end{aligned}$$

Nilai Umur = nilai umur 48, 4, 43, 30, 17, 47, 51, 49, 53, 13, 52, 42, 60 memiliki frekuensi 2, nilai umur 41 memiliki frekuensi 3 dan nilai umur lainnya memiliki frekuensi 1

$$S_{46} = [0+, 1-], |S_{32}|=1$$

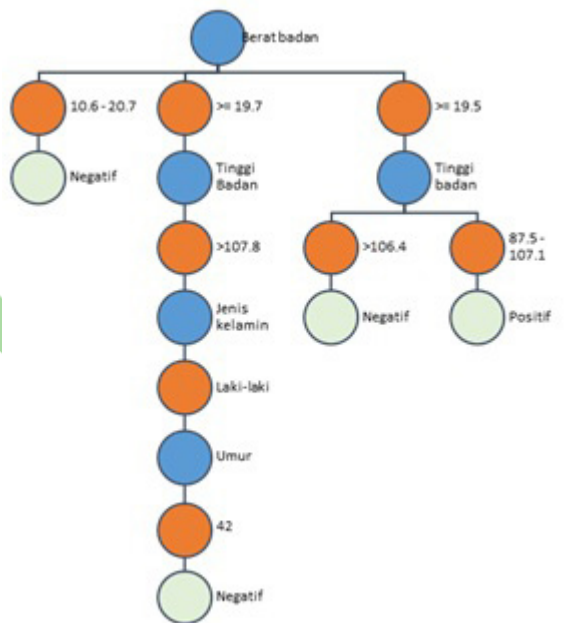
$$S_{43} = [1+, 0-], |S_{45}|=1$$

$$\begin{aligned} \text{Entropy } S_{46} &= [0+, 1-] \\ &= - (0/1) \log_2 (0/1) - (1/1) \log_2 (1/1) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Entropy } S_{41} &= [1+, 0-] \\ &= - (1/1) \log_2 (1/1) - (0/1) \log_2 (0/1) \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Gain Umur} &= \text{Gain}(S, \text{Umur}) = \text{Entropy}(S) - \\ &((1/1) \text{Entropy}(S_{\text{umur}})) - (((0/1) \text{Entropy}(S_{\text{umur}})) * 49) \\ &= 0,73828486614 \end{aligned}$$

Dari hasil tersebut atribut jenis kelamin mendekati angka satu maka atribut tersebut berada dalam *node* "Laki-Laki". Dan dalam *node* "Perempuan" terdapat lagi cabang untuk nilai atribut berdasarkan umur.



Gambar 5 Contoh sebagian pohon dari perhitungan

Pada Gambar 5 pohon tersebut adalah sebagian kecil dari keseluruhan pohon. Pohon tersebut digunakan dengan cara mengubahnya menjadi *rules*, *rules* dibuat dengan menggunakan jika maka. Jadi dari pohon tersebut didapatkan 4 *rule*: (1) Jika berat badan = 10.6 - 20.7 maka obesitas = Negatif; (2) Jika

berat badan = >=19.7 dan tinggi badan = >=107. 8 dan jenis kelamin = laki-laki dan umur = 42 maka obesitas = Negatif; (3) Jika berat badan = >=19.5 dan tinggi badan = >106.4 maka obesitas = Negatif; dan (4) Jika berat badan = >=19.5 dan tinggi badan = 87.5 - 107.1 maka obesitas = Positif

Rules dipakai untuk mengetahui data yang belum memiliki keputusan obesitas = Negatif atau obesitas = Positif.

C. Pengujian Aplikasi

Pengujian dilakukan dengan mengambil sepuluh data yang ingin diketahui status gizinya. Hasil implementasinya dapat dilihat pada Tabel 6.

Tabel 6 Hasil perbandingan status gizi dari data dan dari program

Umur (Bulan)	Jenis Kelamin	Berat Badan (kg)	Tinggi Badan (cm)	Status gizi dari data	Status gizi dari program
24	L	14.6	88	Negatif	Negatif
38	P	11.5	95	Negatif	Negatif
46	P	14.7	102	Negatif	Negatif
1	P	4	50	Negatif	Negatif
17	L	9.1	80	Negatif	Negatif
46	L	24	106	Positif	Negatif
55	L	25.7	108	Positif	Positif
42	L	24.6	108	Positif	Negatif
49	L	23.9	108	Positif	Positif
43	P	25	108	Positif	Positif

Pada Tabel 6 terlihat dari 10 data yang dicoba ada 8 data diketahui status gizinya secara tepat dan 2 data diketahui status gizi yang ditentukan program tidak sama dengan hasil dari data status gizi yang diambil dari Puskesmas Kecamatan Munjul.

D. Analisis Hasil

Hasil dari penelitian ini adalah pohon keputusan yang *best fit* karena tidak melakukan perulangan atribut yang sama untuk cabang yang ada, sebab di dalam ID3 dilakukan REP dan RRP. Di dalam program, pohon keputusan telah dibuat menjadi *rules* yang digunakan untuk *testing*, *testing* menggunakan 10 data untuk mencoba apakah pohon keputusan bisa memutuskan secara akurat. Dari 10 data tersebut yang sesuai dengan hasil data perhitungan dari puskesmas hanya ada delapan yang sesuai. Dua sisanya tidak sesuai, setelah dilihat dari *rules* yang dibuat

ID3 terdapat beberapa hasil keputusan yang tidak sesuai dengan yang ada dalam data *training* sebab dari hasil perhitungan *entropy* dan *information gain* nilai data dengan atribut obesitas = "Positif" lebih kecil dari atribut obesitas = "Negatif". Nilai atribut obesitas = "Positif" lebih kecil karena data

yang di pergunakan dalam *training* lebih sedikit dari data yang memiliki atribut obesitas = “Negatif”.

IV. SIMPULAN

Dari hasil pengujian aplikasi dapat disimpulkan bahwa: (1) Aplikasi ini menggunakan algoritma ID3 untuk menentukan status obesitas pada anak sesuai dengan standar tabel gizi dari WHO; (2) Aplikasi ini sudah dapat membuat pohon keputusan yang *best fit* di mana pohon keputusan yang dibuat lebih ringkas karena di terapkannya REP dan RPP dalam algoritma ID3; dan (3) Aplikasi ini masih memiliki keakuratan yang terbatas dalam memprediksi *input* pengguna karena ketimpangan data yang dipakai data positif obesitas hanya 15 data dari sekitar 300 data.

V. DAFTAR RUJUKAN

- [1] Seotjningsih. *Tumbuh Kembang Anak*. Jakarta: Buku Kedokteran EGC. 1995. hml 183-190.
- [2] A. Jonathan et al. *Effective Prototyping for Software Makers*. San Francisco: Diane Cerra. 2007. hlm 21-25.
- [3] T. Segaran. *Programming Collective Intelligence: Building Smart Web 2.0 Applications*. United States of America. : O'Reilly Media, Inc., 2007. hlm 3-4.
- [4] T. Segaran. *Programming Collective Intelligence: Building Smart Web 2.0 Applications*. United States of America. : O'Reilly Media, Inc., 2007. hlm 142-148.
- [5] Suyanto. *Artificial Intelligence Searching Reasoning Planning and Learning*. Bandung: Penerbit Informatika, 2014. hlm 143-169.
- [6] T. R. Perkasa et al.. “Rancangan Bangun Pendeteksi Gerak Menggunakan Metode Image Subtraction Pada Single Board Computer (SBC)”. In *Journal of Control and Network Systems*. Vol 3. Hml 92. 2014.

KALBIS
Jurnal Sains dan Teknologi