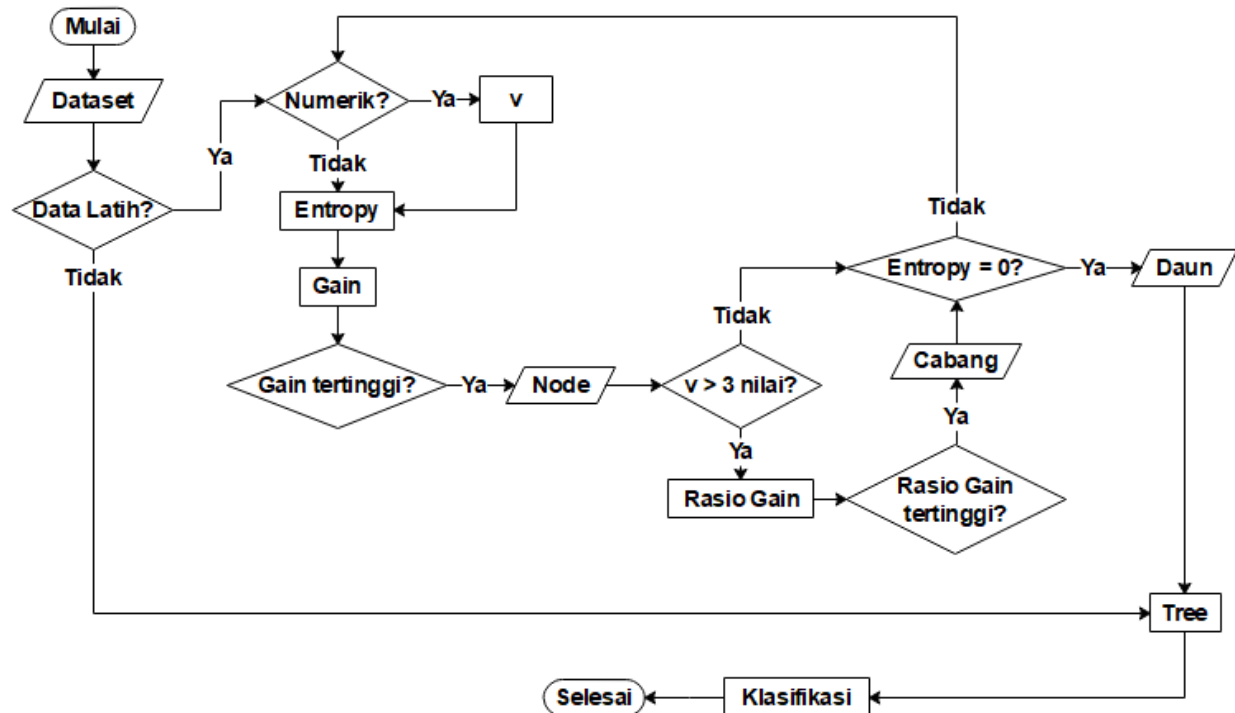


Perhitungan Manual Algoritma *Decision Tree* (C4.5)

Flowchart C4.5



Tabel 1. *Dataset* Latih Golf

<i>Outlook</i>	<i>Temperature</i>	<i>Humidity</i>	<i>Wind</i>	<i>Play</i>
<i>sunny</i>	85	85	<i>false</i>	<i>no</i>
<i>sunny</i>	80	90	<i>true</i>	<i>no</i>
<i>overcast</i>	83	78	<i>false</i>	<i>yes</i>
<i>rain</i>	70	96	<i>false</i>	<i>yes</i>
<i>rain</i>	68	80	<i>false</i>	<i>yes</i>
<i>rain</i>	65	70	<i>true</i>	<i>no</i>
<i>overcast</i>	64	65	<i>true</i>	<i>yes</i>
<i>sunny</i>	72	95	<i>false</i>	<i>no</i>
<i>sunny</i>	69	70	<i>false</i>	<i>yes</i>
<i>rain</i>	75	80	<i>false</i>	<i>yes</i>
<i>sunny</i>	75	70	<i>true</i>	<i>yes</i>
<i>overcast</i>	72	90	<i>true</i>	<i>yes</i>
<i>overcast</i>	81	75	<i>false</i>	<i>yes</i>
<i>rain</i>	71	80	<i>true</i>	<i>no</i>

Dataset: Golf

Tipe data: Numerik (kontinu) dan nominal (disrit)

Karakteristik data: *Balanced class distribution*

Missing value: Tidak

Overlapping data: Ya

Atribut: 4

Algoritma C4.5 [1]

1. Dimulai *node* akar (*node* ke-1)
2. Jika fitur bertipe numerik, cari nilai v
3. Untuk semua fitur dengan tipe numerik atau nominal, hitung *entropy* untuk semua sampel (data latih) pada *node*.
4. Gunakan fitur tersebut sebagai *node* pemecah menjadi cabang.
5. Lakukan secara rekursif pada setiap cabang yang dibuat dengan mengulangi langkah 2 sampai 4 hingga semua data dalam setiap *node* hanya memberikan satu label kelas. *Node* yang tidak dapat dipecah lagi merupakan daun yang berisi keputusan (label kelas).

Proses *node* ke-1 sebagai akar (*root*)

Node akar diperoleh dengan cara wajib melakukan perhitungan terlebih dahulu *Entropy* atau diinisialkan sebagai E (semua data) terhadap komposisi kelas [1].

$$\begin{aligned}
 E(\text{semua}) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua})) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
 &= -\left(\left(\frac{9}{14}\right) \times \log_2 \left(\frac{9}{14}\right) + \left(\frac{5}{14}\right) \times \log_2 \left(\frac{5}{14}\right)\right) \\
 &= -((0,6428 \times \log_2 0,6428) + (0,3571 \times \log_2 0,3571)) \\
 &= -((0,6428 \times (-0,6376)) + (0,3571 \times (-1,4856))) = -(-0,4098 + (-0,5305)) \\
 &= -(-0,9403) = 0,9403
 \end{aligned}$$

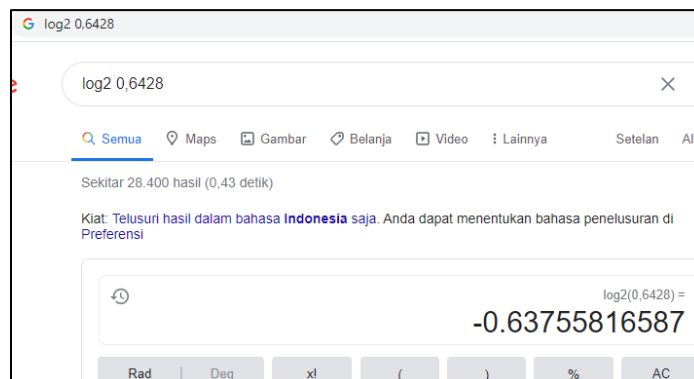
Cara menghitung \log_2

1. Google
2. Microsoft Excel

Google

Cara menghitung $\log_2 a$ di Google, di mana a adalah variable berupa angka, bisa variable maupun positif. Pada Google, tanda variabel boleh menggunakan koma atau titik.

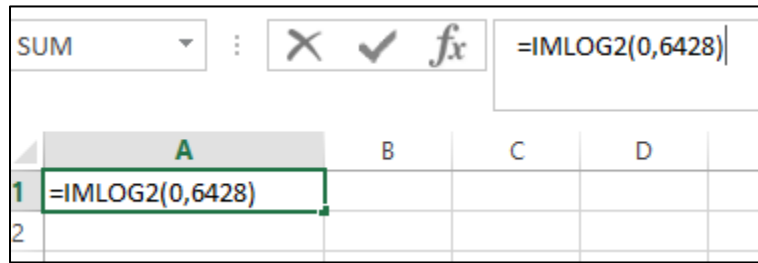
1. Ketikkan $\log_2 0,6428$ di *textbox* yang biasa digunakan untuk mengetikkan url *browser* (lihat Gambar 1)
2. Enter (lihat Gambar 2)



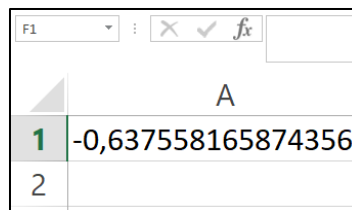
Gambar 1. Mengetikkan nilai \log_2 yang akan dicari di url *browser*

Microsoft Excel

1. Ketikkan $=\text{imlog2}(a)$ di salah satu sel, a adalah nilai yang akan dicari (lihat Gambar 3)
2. Enter (lihat Gambar 4)



Gambar 3. Mengetikkan nilai \log_2 yang dicari di sel Microsoft Excel



Gambar 4. Hasil \log_2 di Microsoft Excel

Karena data Golf terdapat nilai bertipe numerik, maka kita butuh mencari nilai v sebagai nilai pemecah/*split* atau teknik ini biasa disebut sebagai diskretisasi data, yaitu data numerik menjadi data nominal/kontinu. Banyak pendekatan yang bisa digunakan dalam mendapatkan nilai v , yang paling umum dan sering digunakan adalah *Binning*. Salah satu persamaan *binning* yang digunakan adalah *entropy* dan *gain*. *Binning* mendefinisikan kumpulan *class* nominal untuk setiap atribut (3variable input), kemudian menetapkan setiap nilai atribut ke dalam salah satu *class*. Misal jika domain numerik memiliki nilai dari 0 sampai 100, domain tersebut dapat dibagi menjadi 4 bin $\{0...24; 25...49; 50...74; 75...100\}$. Setiap nilai atribut akan dikonversi menjadi atribut nominal/kategorikal yang berkorespondensi dengan salah satu bin. Pendekatan *binning* disebut *unsupervised discretization method* [2]. Namun, pendekatan ini memiliki kelemahan yaitu menyebabkan banyak informasi yang memungkinkan hilang.

Temperature

Tabel 2. Posisi v untuk pemecahan fitur “Temperature” di node akar

Suhu		Jumlah Kasus	Yes	No	Entropy	Gain
$v = 67,2$	\leq	2	1	1	1	0,0103
	$>$	12	8	4	0,9183	
$v = 73$	\leq	8	5	3	0,9544	0,0014
	$>$	6	4	2	0,9183	
$v = 82,25$	\leq	12	8	4	0,9183	0,0103
	$>$	2	1	1	1	

Pada atribut *Temperature* untuk mendapatkan nilai v menggunakan *Binning*. Adapun langkahnya adalah sebagai berikut:

1. Urutkan data dari kecil ke besar: {64, 65, 68, 69, 70; 71, 72, 72, 75, 75; 80, 81, 83, 85}
2. Bagi menjadi 3 bagian atau bin: {64, 65, 68, 69, 70; 71, 72, 72, 75, 75; 80, 81, 83, 85}
3. Hitung rata-rata masing-masing bin.
4. Maka nilai v yang didapatkan adalah 67,2; 73 dan 82,25.

Jika sudah mendapatkan nilai v selanjutnya menghitung nilai *entropy* dan *gain*. Perhitungan nilai *gain* diperlihatkan Tabel 2, sebagai pembandingan, perhitungan yang sama dilakukan oleh Eko ditunjukkan pada Tabel 4-9 [1].

Tabel 4.9 Posisi v untuk pemecahan fitur 'suhu' di node akar

Suhu	70		75		80	
	<=	>	<=	>	<=	>
Ya	4	5	7	2	7	2
Tidak	1	4	3	2	4	1
Gain	0.0453		0.0251		0.0005	

Entropy nilai v atribut *Temperature*

$$\begin{aligned}
 E(\leq 67,2) &= -\left((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua})) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))\right) \\
 &= -\left(\left(\left(\frac{1}{2}\right) \times \log_2 \left(\frac{1}{2}\right)\right) + \left(\left(\frac{1}{2}\right) \times \log_2 \left(\frac{1}{2}\right)\right)\right) \\
 &= -((0,5 \times \log_2 0,5) + (0,5 \times \log_2 0,5)) = -((0,5 \times (-1)) + (0,5 \times (-1))) \\
 &= -(-0,5 + (-0,5)) = -(-1) = 1
 \end{aligned}$$

$$\begin{aligned}
 E(> 67,2) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua})) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
 &= -\left(\left(\left(\frac{8}{12}\right) \times \log_2 \left(\frac{8}{12}\right)\right) + \left(\left(\frac{4}{12}\right) \times \log_2 \left(\frac{4}{12}\right)\right)\right) \\
 &= -((0,6667 \times \log_2 0,6667) + (0,3333 \times \log_2 0,3333)) \\
 &= -((0,6667 \times (-0,5849)) + (0,3333 \times (-1,5851))) = -(-0,3899 + (-0,5283)) \\
 &= -(-0,9182) = 0,9183
 \end{aligned}$$

$$\begin{aligned}
 E(\leq 73) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua})) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
 &= -\left(\left(\left(\frac{5}{8}\right) \times \log_2 \left(\frac{5}{8}\right)\right) + \left(\left(\frac{3}{8}\right) \times \log_2 \left(\frac{3}{8}\right)\right)\right) \\
 &= -((0,625 \times \log_2 0,625) + (0,375 \times \log_2 0,375)) \\
 &= -((0,625 \times (-0,6781)) + (0,375 \times (-1,415))) = -(-0,4238 + (-0,5306)) \\
 &= -(-0,9544) = 0,9544
 \end{aligned}$$

$$\begin{aligned}
 E(> 73) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua})) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
 &= -\left(\left(\left(\frac{4}{6}\right) \times \log_2 \left(\frac{4}{6}\right)\right) + \left(\left(\frac{2}{6}\right) \times \log_2 \left(\frac{2}{6}\right)\right)\right) \\
 &= -((0,6667 \times \log_2 0,6667) + (0,3333 \times \log_2 0,3333)) \\
 &= -((0,6667 \times (-0,5849)) + (0,3333 \times (-1,5851))) = -(-0,3899 + (-0,5283)) \\
 &= -(-0,9182) = 0,9183
 \end{aligned}$$

$$\begin{aligned}
E(\leq 82,25) &= -(p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
&= -\left(\left(\left(\frac{8}{12}\right) \times \log_2 \left(\frac{8}{12}\right)\right) + \left(\left(\frac{4}{12}\right) \times \log_2 \left(\frac{4}{12}\right)\right)\right) \\
&= -((0,6667 \times \log_2 0,6667) + (0,3333 \times \log_2 0,3333)) \\
&= -((0,6667 \times (-0,5849)) + (0,3333 \times (-1,5851))) = -(-0,3899 + (-0,5283)) \\
&= -(-0,9182) = 0,9183 \\
E(> 82,25) &= -(p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
&= -\left(\left(\left(\frac{1}{2}\right) \times \log_2 \left(\frac{1}{2}\right)\right) + \left(\left(\frac{1}{2}\right) \times \log_2 \left(\frac{1}{2}\right)\right)\right) \\
&= -((0,5 \times \log_2 0,5) + (0,5 \times \log_2 0,5)) = -((0,5 \times (-1)) + (0,5 \times (-1))) \\
&= -(-0,5 + (-0,5)) = -(-1) = 1
\end{aligned}$$

Gain nilai v atribut Temperature [1]

$$\begin{aligned}
\text{Gain}(\text{semua}, 67,2) &= E(\text{semua}) - \sum_{i=1}^n \frac{|67,2_i|}{|\text{semua}|} \times E(67,2_i) \\
&= 0,9403 - \left(\left(\left(\frac{2}{14}\right) \times 1\right) + \left(\left(\frac{12}{14}\right) \times 0,9183\right)\right) \\
&= 0,9403 - ((0,1429 \times 1) + (0,8571 \times 0,9183)) = 0,9403 - (0,1429 + 0,7871) \\
&= 0,9403 - 0,93 = 0,0103
\end{aligned}$$

$$\begin{aligned}
\text{Gain}(\text{semua}, 73) &= E(\text{semua}) - \sum_{i=1}^n \frac{|73_i|}{|\text{semua}|} \times E(73_i) \\
&= 0,9403 - \left(\left(\left(\frac{8}{14}\right) \times 0,9544\right) + \left(\left(\frac{6}{14}\right) \times 0,9183\right)\right) \\
&= 0,9403 - ((0,5714 \times 0,9544) + (0,4286 \times 0,9183)) \\
&= 0,9403 - (0,5453 + 0,3936) = 0,9403 - 0,9389 = 0,0014
\end{aligned}$$

$$\begin{aligned}
\text{Gain}(\text{semua}, 82,25) &= E(\text{semua}) - \sum_{i=1}^n \frac{|82,25_i|}{|\text{semua}|} \times E(82,25_i) \\
&= 0,9403 - \left(\left(\left(\frac{12}{14}\right) \times 0,9183\right) + \left(\left(\frac{2}{14}\right) \times 1\right)\right) \\
&= 0,9403 - ((0,8571 \times 0,9183) + (0,1429 \times 1)) = 0,9403 - (0,7871 + 0,1429) \\
&= 0,9403 - 0,93 = 0,0103
\end{aligned}$$

Karena nilai gain tertinggi didapatkan oleh $v = 67,2$ atau $v = 82,25$, maka atribut *temperature* dilakukan diskretisasi pada nilai oleh $v = 67,2$ atau $v = 82,25$ ketika menghitung entropy dan gain pada semua atribut.

Humadity

Pada atribut *Humadity* untuk mendapatkan nilai v menggunakan *Binning*. Adapun langkahnya adalah sebagai berikut:

1. Urutkan data dari kecil ke besar: {65, 70, 70, 70, 75; 78, 80, 80, 80, 85; 90, 90, 95, 96}
2. Bagi menjadi 3 bagian atau bin: {65, 70, 70, 70, 75; 78, 80, 80, 80, 85; 90, 90, 95, 96}
3. Hitung rata-rata masing-masing bin.
4. Maka nilai v yang didapatkan adalah 70; 80,6; dan 92,75.

Jika sudah mendapatkan nilai v selanjutnya menghitung nilai *entropy* dan *gain*. Perhitungan nilai *gain* diperlihatkan Tabel 3, sebagai pembanding, perhitungan yang sama dilakukan oleh Eko ditunjukkan pada Tabel 4-10 [1].

Tabel 3. Posisi v untuk pemecahan fitur “Humadity” di node akar

Suhu		Jumlah Kasus	Yes	No	Entropy	Gain
$v = 70$	\leq	4	3	1	0,8112	0,0149
	$>$	10	6	4	0,971	
$v = 80,6$	\leq	9	7	2	0,7641	0,1024
	$>$	5	2	3	0,971	
$v = 92,75$	\leq	12	8	4	0,9182	0,0104
	$>$	2	1	1	1	

Tabel 4.10 Posisi v untuk pemecahan fitur ‘kelembaban’ di node akar

Kelembaban	70		75		80		85	
	\leq	$>$	\leq	$>$	\leq	$>$	\leq	$>$
Ya	2	7	3	6	7	2	7	2
Tidak	1	4	1	4	2	3	3	2
Gain	0.0005		0.0150		0.1022		0.0251	

Entropy nilai v atribut Humadity

$$\begin{aligned}
 E(\leq 70) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua})) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
 &= -\left(\left(\left(\frac{3}{4}\right) \times \log_2 \left(\frac{3}{4}\right)\right) + \left(\left(\frac{1}{4}\right) \times \log_2 \left(\frac{1}{4}\right)\right)\right) \\
 &= -((0,75 \times \log_2 0,75) + (0,25 \times \log_2 0,25)) \\
 &= -((0,75 \times (-0,415)) + (0,25 \times (-2))) = -(-0,3112 + (-0,5)) = -(-0,8112) \\
 &= 0,8112
 \end{aligned}$$

$$\begin{aligned}
 E(> 70) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua})) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
 &= -\left(\left(\left(\frac{6}{10}\right) \times \log_2 \left(\frac{6}{10}\right)\right) + \left(\left(\frac{4}{10}\right) \times \log_2 \left(\frac{4}{10}\right)\right)\right) \\
 &= -((0,6 \times \log_2 0,6) + (0,4 \times \log_2 0,4)) \\
 &= -((0,6 \times (-0,737)) + (0,4 \times (-1,3219))) = -(-0,4422 + (-0,5288)) \\
 &= -(-0,971) = 0,971
 \end{aligned}$$

$$\begin{aligned}
E(\leq 80,6) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
&= -\left(\left(\left(\frac{7}{9}\right) \times \log_2 \left(\frac{7}{9}\right)\right) + \left(\left(\frac{2}{9}\right) \times \log_2 \left(\frac{2}{9}\right)\right)\right) \\
&= -((0,7778 \times \log_2 0,7778) + (0,2222 \times \log_2 0,2222)) \\
&= -((0,7778 \times (-0,3625)) + (0,2222 \times (-2,1701))) = -(-0,2819 + (-0,4822)) \\
&= -(-0,7641) = 0,7641
\end{aligned}$$

$$\begin{aligned}
E(> 80,6) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua})) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
&= -\left(\left(\left(\frac{2}{5}\right) \times \log_2 \left(\frac{2}{5}\right)\right) + \left(\left(\frac{3}{5}\right) \times \log_2 \left(\frac{3}{5}\right)\right)\right) \\
&= -((0,4 \times \log_2 0,4) + (0,6 \times \log_2 0,6)) \\
&= -((0,4 \times (-1,3219)) + (0,6 \times (-0,737))) = -(-0,5288 + (-0,4422)) \\
&= -(-0,971) = 0,971
\end{aligned}$$

$$\begin{aligned}
E(\leq 92,75) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
&= -\left(\left(\left(\frac{8}{12}\right) \times \log_2 \left(\frac{8}{12}\right)\right) + \left(\left(\frac{4}{12}\right) \times \log_2 \left(\frac{4}{12}\right)\right)\right) \\
&= -((0,6667 \times \log_2 0,6667) + (0,3333 \times \log_2 0,3333)) \\
&= -((0,6667 \times (-0,5849)) + (0,3333 \times (-1,5851))) = -(-0,3899 + (-0,5283)) \\
&= -(-0,9182) = 0,9182
\end{aligned}$$

$$\begin{aligned}
E(> 92,75) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
&= -\left(\left(\left(\frac{1}{2}\right) \times \log_2 \left(\frac{1}{2}\right)\right) + \left(\left(\frac{1}{2}\right) \times \log_2 \left(\frac{1}{2}\right)\right)\right) \\
&= -((0,5 \times \log_2 0,5) + (0,5 \times \log_2 0,5)) = -((0,5 \times (-1)) + (0,5 \times (-1))) \\
&= -(-0,5 + (-0,5)) = -(-1) = 1
\end{aligned}$$

Gain nilai v atribut Humadity

$$\begin{aligned}
\text{Gain}(\text{semua}, 70) &= E(\text{semua}) - \sum_{i=1}^n \frac{|70_i|}{|\text{semua}|} \times E(70_i) \\
&= 0,9403 - \left(\left(\frac{4}{14}\right) \times 0,8112\right) + \left(\left(\frac{10}{14}\right) \times 0,971\right) \\
&= 0,9403 - ((0,2857 \times 0,8112) + (0,7143 \times 0,971)) \\
&= 0,9403 - (0,2318 + 0,6936) = 0,9403 - 0,9254 = 0,0149
\end{aligned}$$

$$\begin{aligned}
\text{Gain}(\text{semua}, 80,6) &= E(\text{semua}) - \sum_{i=1}^n \frac{|80,6_i|}{|\text{semua}|} \times E(80,6_i) \\
&= 0,9403 - \left(\left(\frac{9}{14}\right) \times 0,7641\right) + \left(\left(\frac{5}{14}\right) \times 0,971\right) \\
&= 0,9403 - ((0,6429 \times 0,7641) + (0,3571 \times 0,971)) \\
&= 0,9403 - (0,4912 + 0,3467) = 0,9403 - 0,8379 = \mathbf{0,1024}
\end{aligned}$$

$$\begin{aligned}
Gain(semua, 92,75) &= E(semua) - \sum_{i=1}^n \frac{|92,75_i|}{|semua|} \times E(92,75_i) \\
&= 0,9403 - \left(\left(\frac{12}{14} \right) \times 0,9182 \right) + \left(\left(\frac{2}{14} \right) \times 1 \right) \\
&= 0,9403 - ((0,8571 \times 0,9182) + (0,1429 \times 1)) = 0,9403 - (0,787 + 0,1429) \\
&= 0,9403 - 0,9299 = 0,0104
\end{aligned}$$

Karena nilai gain tertinggi didapatkan oleh $v = 80,6$, maka atribut *temperature* dilakukan diskretisasi pada nilai $v = 80,6$ ketika menghitung entropy dan gain pada semua atribut.

Tabel 4. Hasil perhitungan *entropy* dan *gain* untuk node akar

Node	Atribut	v	Jumlah Kasus	Yes	No	Entropy	Gain
1	Total		14	9	5	0,9403	
	Outlook	<i>Sunny</i>	5	2	3	0,971	0,2469
		<i>Overcast</i>	5	3	2	0,971	
		<i>Rain</i>	4	4	0	0	
	<i>Temperature</i>	≤ 73	8	5	3	0,9544	0,0251
		> 73	6	4	2	0,9182	
	<i>Humadity</i>	$\leq 80,6$	9	7	2	0,7641	0,1024
		$< 80,6$	5	2	3	0,971	
	<i>Wind</i>	<i>True</i>	6	3	3	1	0,0482
		<i>False</i>	8	6	2	0,8112	

Outlook

Entropy Outlook

$$\begin{aligned}
E(Sunny) &= - \left((p(\text{yes}|semua) \times \log_2 p(\text{no}|semua)) + (p(\text{no}|semua) \times \log_2 p(\text{no}|semua)) \right) \\
&= - \left(\left(\left(\frac{2}{5} \right) \times \log_2 \left(\frac{2}{5} \right) \right) + \left(\left(\frac{3}{5} \right) \times \log_2 \left(\frac{3}{5} \right) \right) \right) \\
&= - \left((0,4 \times \log_2 0,4) + (0,6 \times \log_2 0,6) \right) \\
&= - \left((0,4 \times (-1,3219)) + (0,6 \times (-0,737)) \right) = -(-0,5288 + (-0,4422)) \\
&= -(-0,971) = 0,971
\end{aligned}$$

$$\begin{aligned}
E(Overcast) &= - \left((p(\text{yes}|semua) \times \log_2 p(\text{no}|semua)) + (p(\text{no}|semua) \times \log_2 p(\text{no}|semua)) \right) \\
&= - \left(\left(\left(\frac{3}{5} \right) \times \log_2 \left(\frac{3}{5} \right) \right) + \left(\left(\frac{2}{5} \right) \times \log_2 \left(\frac{2}{5} \right) \right) \right) \\
&= - \left((0,6 \times \log_2 0,6) + (0,4 \times \log_2 0,4) \right) \\
&= - \left((0,6 \times (-0,737)) + (0,4 \times (-1,3219)) \right) = -(-0,4422 + (-0,5288)) \\
&= -(-0,971) = 0,971
\end{aligned}$$

$$\begin{aligned}
E(\text{Rain}) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua})) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
&= -\left(\left(\left(\frac{4}{4}\right) \times \log_2 \left(\frac{4}{4}\right)\right) + \left(\left(\frac{0}{4}\right) \times \log_2 \left(\frac{0}{4}\right)\right)\right) = -((1 \times \log_2 1) + (0 \times \log_2 0)) \\
&= -((1 \times (0)) + (0 \times (-\infty))) = -(0 + 0) = -(0) = 0
\end{aligned}$$

Gain Outlook

$$\begin{aligned}
\text{Gain}(\text{semua}, \text{Outlook}) &= E(\text{semua}) - \sum_{i=1}^n \frac{|\text{Outlook}_i|}{|\text{semua}|} \times E(\text{Outlook}_i) \\
&= 0,9403 - \left(\left(\frac{5}{14}\right) \times 0,971\right) + \left(\left(\frac{5}{14}\right) \times 0,971\right) + \left(\left(\frac{4}{14}\right) \times 0\right) \\
&= 0,9403 - ((0,3571 \times 0,971) + (0,3571 \times 0,971) + 0) \\
&= 0,9403 - (0,3467 + 0,3467) = 0,9403 - 0,6934 = 0,2469
\end{aligned}$$

Wind

Entropy Wind

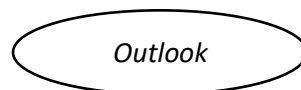
$$\begin{aligned}
E(\text{True}) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua})) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
&= -\left(\left(\left(\frac{3}{6}\right) \times \log_2 \left(\frac{3}{6}\right)\right) + \left(\left(\frac{3}{6}\right) \times \log_2 \left(\frac{3}{6}\right)\right)\right) \\
&= -((0,5 \times \log_2 0,5) + (0,5 \times \log_2 0,5)) = -((0,5 \times (-1)) + (0,5 \times (-1))) \\
&= -(-0,5 + (-0,5)) = -(-1) = 1
\end{aligned}$$

$$\begin{aligned}
E(\text{False}) &= -((p(\text{yes}|\text{semua}) \times \log_2 p(\text{no}|\text{semua})) + (p(\text{no}|\text{semua}) \times \log_2 p(\text{no}|\text{semua}))) \\
&= -\left(\left(\left(\frac{6}{8}\right) \times \log_2 \left(\frac{6}{8}\right)\right) + \left(\left(\frac{2}{8}\right) \times \log_2 \left(\frac{2}{8}\right)\right)\right) \\
&= -((0,75 \times \log_2 0,75) + (0,25 \times \log_2 0,25)) \\
&= -((0,75 \times (-0,415)) + (0,25 \times (-2))) = -(-0,3112 + (-0,5)) = -(-0,8112) \\
&= 0,8112
\end{aligned}$$

Gain Wind

$$\begin{aligned}
\text{Gain}(\text{semua}, \text{Wind}) &= E(\text{semua}) - \sum_{i=1}^n \frac{|\text{Wind}_i|}{|\text{semua}|} \times E(\text{Wind}_i) \\
&= 0,9403 - \left(\left(\frac{6}{14}\right) \times 1\right) + \left(\left(\frac{8}{14}\right) \times 0,8112\right) \\
&= 0,9403 - ((0,4286 \times 1) + (0,5714 \times 0,8112)) = 0,9403 - (0,4286 + 0,4635) \\
&= 0,9403 - 0,8921 = 0,0482
\end{aligned}$$

Outlook memiliki nilai gain tertinggi sehingga terpilih sebagai *root* seperti Gambar 4. Langkah selanjutnya adalah menghitung posisi split “*Outlook*” dengan menghitung rasio gain.



Gambar 4. Node akar

Rasio Gain

Perhitungan posisi *split* untuk opsi satu sebagai berikut [1]:

$$\begin{aligned}
 SplitInfo(semua, outlook) &= -\sum_{i=1}^k p(v_i|s) \log_2 p(v_i|s) = -\left((p(outlook|semua) \times \log_2 p(sunny|semua)) + (p(outlook|semua) \times \log_2 p(overcast|semua)) + (p(outlook|semua) \times \log_2 p(rain|semua)) \right) \\
 &= -\left(\left(\frac{5}{14} \times \log_2 \left(\frac{5}{14} \right) \right) + \left(\frac{5}{14} \times \log_2 \left(\frac{5}{14} \right) \right) + \left(\frac{4}{14} \times \log_2 \left(\frac{4}{14} \right) \right) \right) = \\
 &= -\left((0,3571 \times \log_2(0,3571)) + (0,3571 \times \log_2(0,3571)) + (0,2857 \times \log_2(0,2857)) \right) = \\
 &= -\left((0,3571 \times (-1,4856)) + (0,3571 \times (-1,4856)) + (0,2857 \times (-1,8074)) \right) = -(-0,5305 + \\
 &(-0,5305) + (-0,5164)) = -(-1,5774) = 1,5774
 \end{aligned}$$

$$RasioGain(Semua, Outlook) = \frac{Gain(Semua, Outlook)}{SplitInfo(Semua, Outlook)} = \frac{0,2469}{1,5774} = 0,1565$$

Dengan cara yang sama, akan didapatkan nilai rasio gain untuk opsi yang lain.

Hasil dari semua perhitungan *Split Info* di sajikan pada Tabel 5.

Tabel 5. Perhitungan Rasio Gain untuk fitur “*Outlook*”

Node			Jumlah	Entropy	Gain	Rasio Gain
1	Total		14	1,5774	0,2469	
Opsi 1	Outlook			1,5774		0,1565
		Sunny	5			
		Overcast	4			
		Rain	5			
Opsi 2	Outlook			0,9403		0,2624
		Sunny	5			
		Overcast & Rain	9			
Opsi 3	Outlook			0,9403		0,2624
		Sunny & Overcast	9			
		Rain	5			
Opsi 4	Outlook			0,8631		0,2859
		Sunny & Rain	10			
		Overcast	4			

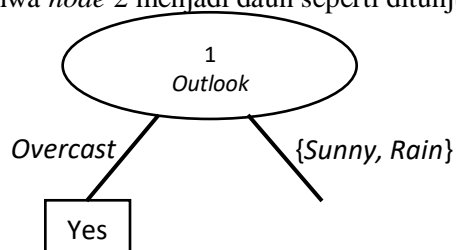
Hasil pemisahan data menurut *node* akan disajikan pada Tabel 6.

Tabel 6. Pemisahan *Dataset Golf* Iterasi 1

<i>Outlook</i>	<i>Temperature</i>	<i>Humidity</i>	<i>Wind</i>	<i>Play</i>
<i>sunny</i>	85	85	<i>FALSE</i>	<i>no</i>
<i>sunny</i>	80	90	<i>TRUE</i>	<i>no</i>
<i>sunny</i>	72	95	<i>FALSE</i>	<i>no</i>
<i>sunny</i>	69	70	<i>FALSE</i>	<i>yes</i>
<i>sunny</i>	75	70	<i>TRUE</i>	<i>yes</i>
<i>rain</i>	70	96	<i>FALSE</i>	<i>yes</i>
<i>rain</i>	68	80	<i>FALSE</i>	<i>yes</i>
<i>rain</i>	65	70	<i>TRUE</i>	<i>no</i>
<i>rain</i>	75	80	<i>FALSE</i>	<i>yes</i>
<i>rain</i>	71	80	<i>TRUE</i>	<i>no</i>
<i>overcast</i>	83	78	<i>FALSE</i>	<i>yes</i>
<i>overcast</i>	64	65	<i>TRUE</i>	<i>yes</i>
<i>overcast</i>	72	90	<i>TRUE</i>	<i>yes</i>
<i>overcast</i>	81	75	<i>FALSE</i>	<i>yes</i>

Proses *node* ke-2

Untuk *node* 2, nilai *entropy* yang didapat adalah 0 (karena semua baris pada *overcast* memiliki kelas yang sama). Oleh karena itu dipastikan bahwa *node* 2 menjadi daun seperti ditunjukkan pada Gambar 5.



Gambar 5. Hasil pembentukan cabang di akar untuk kasus “Apakah harus bermain golf?”

Proses *node* ke-3

Selanjutnya, di *node* 3 harus dihitung dulu *entropy* untuk sisa data terhadap komposisi yang tidak masuk dalam *node* 2.

Temperature

Pada atribut *Temperature* untuk mendapatkan nilai v menggunakan *Binning*. Adapun langkahnya adalah sebagai berikut:

1. Urutkan data dari kecil ke besar: {65, 68, 69, 70, 71; 72, 75, 75, 80, 85}
2. Bagi menjadi 2 bagian atau bin: {65, 68, 69, 70, 71; 72, 75, 75, 80, 85}
3. Hitung rata-rata masing-masing bin.
4. Maka nilai v yang didapatkan adalah 68,6 dan 77,4.

Jika sudah mendapatkan nilai v selanjutnya menghitung nilai *entropy* dan *gain*. Perhitungan nilai *gain* diperlihatkan Tabel 7.

Tabel 7. Posisi v untuk pemecahan fitur “*Temperature*” di node 3

Suhu		Jumlah Kasus	Yes	No	Entropy	Gain
$v = 68,6$	\leq	2	1	1	1	0
	$>$	8	4	4	1	
$v = 77,4$	\leq	8	5	3	0,9544	0,2365
	$>$	2	0	1	0	

Humadity

Pada atribut *Humadity* untuk mendapatkan nilai v menggunakan *Binning*. Adapun langkahnya adalah sebagai berikut:

1. Urutkan data dari kecil ke besar: {70, 70, 70, 80, 80; 80, 85, 90, 95, 96}
2. Bagi menjadi 2 bagian atau bin: {70, 70, 70, 80, 80; 80, 85, 90, 95, 96}
3. Hitung rata-rata masing-masing bin.
4. Maka nilai v yang didapatkan adalah 74 dan 89,2.

Jika sudah mendapatkan nilai v selanjutnya menghitung nilai *entropy* dan *gain*. Perhitungan nilai *gain* diperlihatkan Tabel 8.

Tabel 8. Posisi v untuk pemecahan fitur “*Humadity*” di node 3

Suhu		Jumlah Kasus	Yes	No	Entropy	Gain
$v = 74$	\leq	3	2	1	0,9183	0,0349
	$>$	7	3	4	0,9852	
$v = 80,6$	\leq	6	4	2	0,9183	0,1245
	$>$	4	1	3	0,8113	

Selanjutnya dihitung *entropy* untuk setiap nilai fitur terhadap kelas, kemudian dihitung *gain* untuk setiap fitur. Hasil perhitungan disajikan pada Tabel 9.

Tabel 9. Hasil perhitungan *entropy* dan *gain* untuk node 3

Node	Atribut	v	Jumlah Kasus	Yes	No	Entropy	Gain
3	Total		14	9	5	0,9403	
	<i>Outlook</i>	<i>Sunny</i>	5	2	3	0,971	0,029
		<i>Rain</i>	5	3	2	0,971	
	<i>Temperature</i>	$\leq 77,4$	8	5	3	0,9544	0,2365
		$> 77,4$	2	0	1	0	
	<i>Humadity</i>	$\leq 80,6$	6	4	2	0,9183	0,1245
		$< 80,6$	4	1	3	0,8113	
	<i>Wind</i>	<i>True</i>	4	1	3	0,8113	0,1245
		<i>False</i>	6	4	2	0,9183	

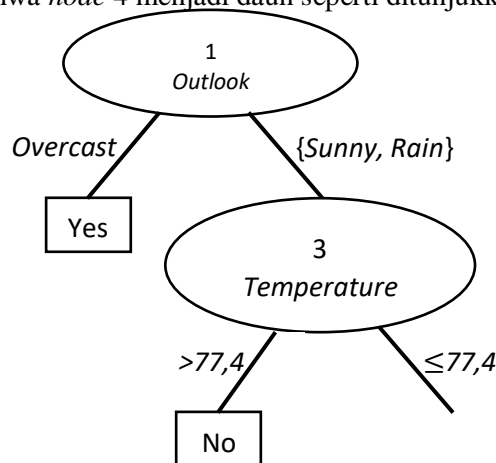
Hasil yang ditunjukkan pada Tabel 9 menunjukkan bahwa gain tertinggi ada di fitur “Temperature”, sehingga fitur “Temperature” dijadikan kondisi di node 3 seperti ditunjukkan pada Tabel 10.

Tabel 10. Pemisahan *Dataset* Golf Iterasi 2

<i>Outlook</i>	<i>Temperature</i>	<i>Humidity</i>	<i>Wind</i>	<i>Play</i>
rain	65	70	TRUE	no
rain	68	80	FALSE	yes
sunny	69	70	FALSE	yes
rain	70	96	FALSE	yes
rain	71	80	TRUE	no
sunny	72	95	FALSE	no
sunny	75	70	TRUE	yes
rain	75	80	FALSE	yes
sunny	80	90	TRUE	no
sunny	85	85	FALSE	no

Proses *node* ke-4

Untuk node 4, nilai *entropy* yang didapat adalah 0 (karena semua baris pada $> 77,4$ memiliki kelas yang sama). Oleh karena itu dipastikan bahwa *node* 4 menjadi daun seperti ditunjukkan pada Gambar 6.



Gambar 6. Hasil pembentukan cabang di node ke-3 untuk kasus “Apakah harus bermain golf?”

Proses *node* ke-5

Selanjutnya, di *node* 5 harus dihitung dulu *entropy* untuk sisa data terhadap komposisi yang tidak masuk dalam *node* 4.

Temperature

Pada atribut *Temperature* untuk mendapatkan nilai v menggunakan *Binning*. Adapun langkahnya adalah sebagai berikut:

1. Urutkan data dari kecil ke besar: {65, 68, 69, 70; 71, 72, 75, 75}
2. Bagi menjadi 2 bagian atau bin: {65, 68, 69, 70; 71, 72, 75, 75}
3. Hitung rata-rata masing-masing bin.
4. Maka nilai v yang didapatkan adalah 68 dan 73,25.

Jika sudah mendapatkan nilai v selanjutnya menghitung nilai *entropy* dan *gain*. Perhitungan nilai *gain* diperlihatkan Tabel 11.

Tabel 11. Posisi v untuk pemecahan fitur “*Temperature*” di node 5

Suhu		Jumlah Kasus	Yes	No	Entropy	Gain
$v = 68$	\leq	2	1	1	1	0,0157
	$>$	6	4	2	0,9183	
$v = 73,25$	\leq	6	3	3	1	0,2044
	$>$	2	1	1	0	

Humidity

Pada atribut *Humidity* untuk mendapatkan nilai v menggunakan *Binning*. Adapun langkahnya adalah sebagai berikut:

1. Urutkan data dari kecil ke besar: {70, 70, 70, 80; 80, 80, 95, 96}
2. Bagi menjadi 2 bagian atau bin: {70, 70, 70, 80; 80, 80, 95, 96}
3. Hitung rata-rata masing-masing bin.
4. Maka nilai v yang didapatkan adalah 72,5 dan 87,75.

Jika sudah mendapatkan nilai v selanjutnya menghitung nilai *entropy* dan *gain*. Perhitungan nilai *gain* diperlihatkan Tabel 8.

Tabel 8. Posisi v untuk pemecahan fitur “*Humidity*” di node 5

Suhu		Jumlah Kasus	Yes	No	Entropy	Gain
$v = 72,5$	\leq	3	2	1	0,9183	0,0032
	$>$	5	3	2	0,971	
$v = 87,75$	\leq	6	4	2	0,9183	0,0157
	$>$	2	1	1	1	

Selanjutnya dihitung *entropy* untuk setiap nilai fitur terhadap kelas, kemudian dihitung *gain* untuk setiap fitur. Hasil perhitungan disajikan pada Tabel 9.

Tabel 9. Hasil perhitungan *entropy* dan *gain* untuk node 5

Node	Atribut	v	Jumlah Kasus	Yes	No	Entropy	Gain
5	Total		8	5	3	0,9544	
	<i>Outlook</i>	<i>Sunny</i>	3	2	1	0,9183	0,0032
		<i>Rain</i>	5	3	2	0,971	
	<i>Temperature</i>	$\leq 73,25$	6	3	3	1	0,2044
		$> 73,25$	2	2	0	0	
	<i>Humidity</i>	$\leq 87,75$	6	4	2	0,9183	0,0157
		$< 87,75$	2	1	1	1	
	<i>Wind</i>	<i>True</i>	3	1	2	0,9183	0,1589
		<i>False</i>	5	4	1	0,7219	

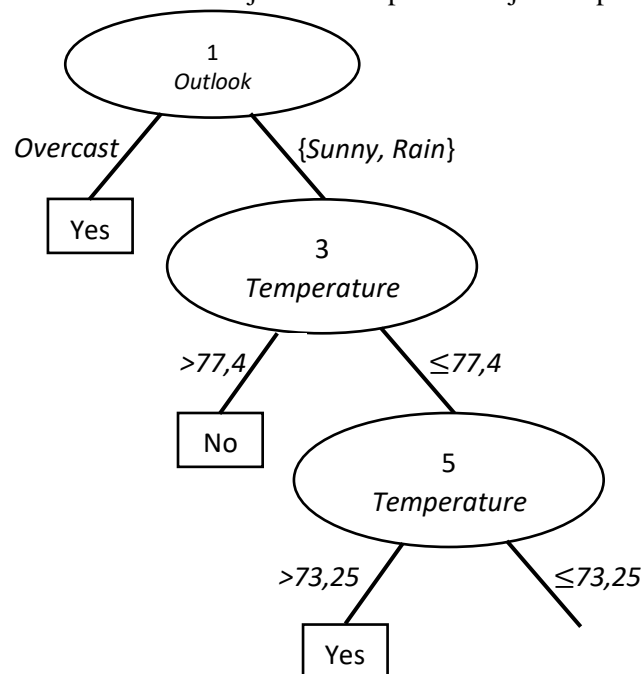
Hasil yang ditunjukkan pada Tabel 9 menunjukkan bahwa gain tertinggi ada di fitur “Temperature”, sehingga fitur “Temperature” dijadikan kondisi di node 5 seperti ditunjukkan pada Tabel 10.

Tabel 10. Pemisahan *Dataset* Golf Iterasi 3

<i>Outlook</i>	<i>Temperature</i>	<i>Humidity</i>	<i>Wind</i>	<i>Play</i>
rain	65	70	TRUE	no
rain	68	80	FALSE	yes
sunny	69	70	FALSE	yes
rain	70	96	FALSE	yes
rain	71	80	TRUE	no
sunny	72	95	FALSE	no
sunny	75	70	TRUE	yes
rain	75	80	FALSE	yes

Proses node ke-6

Untuk node 6, nilai *entropy* yang didapat adalah 0 (karena semua baris pada $> 73,25$ memiliki kelas yang sama). Oleh karena itu dipastikan bahwa node 6 menjadi daun seperti ditunjukkan pada Gambar 7.



Gambar 7. Hasil pembentukan cabang di node ke-5 untuk kasus “Apakah harus bermain golf?”

Proses node ke-7

Selanjutnya, di node 7 harus dihitung dulu *entropy* untuk sisa data terhadap komposisi yang tidak masuk dalam node 6.

Temperature

Pada atribut *Temperature* untuk mendapatkan nilai v menggunakan *Binning*. Adapun langkahnya adalah sebagai berikut:

1. Urutkan data dari kecil ke besar: {65, 68, 69; 70, 71, 72}
2. Bagi menjadi 2 bagian atau bin: {65, 68, 69; 70, 71, 72}

3. Hitung rata-rata masing-masing bin.
4. Maka nilai v yang didapatkan adalah 67,33 dan 71.

Jika sudah mendapatkan nilai v selanjutnya menghitung nilai *entropy* dan *gain*. Perhitungan nilai *gain* diperlihatkan Tabel 11.

Tabel 11. Posisi v untuk pemecahan fitur “*Temperature*” di node 7

Suhu		Jumlah Kasus	Yes	No	Entropy	Gain
$v = 67,33$	\leq	1	0	1	0	0,1909
	$>$	5	3	2	0,971	
$v = 71$	\leq	5	3	2	0,971	0,1909
	$>$	1	0	1	0	

Humadity

Pada atribut *Humadity* untuk mendapatkan nilai v menggunakan *Binning*. Adapun langkahnya adalah sebagai berikut:

1. Urutkan data dari kecil ke besar: {70, 70, 80; 80, 95, 96}
2. Bagi menjadi 2 bagian atau bin: {70, 70, 80; 80, 95, 96}
3. Hitung rata-rata masing-masing bin.
4. Maka nilai v yang didapatkan adalah 73,33 dan 90,33.

Jika sudah mendapatkan nilai v selanjutnya menghitung nilai *entropy* dan *gain*. Perhitungan nilai *gain* diperlihatkan Tabel 12.

Tabel 12. Posisi v untuk pemecahan fitur “*Humadity*” di node 7

Suhu		Jumlah Kasus	Yes	No	Entropy	Gain
$v = 73,33$	\leq	2	1	1	1	0
	$>$	4	2	2	1	
$v = 90,33$	\leq	4	2	2	1	0
	$>$	2	1	1	1	

Selanjutnya dihitung *entropy* untuk setiap nilai fitur terhadap kelas, kemudian dihitung *gain* untuk setiap fitur. Hasil perhitungan disajikan pada Tabel 13.

Tabel 13. Hasil perhitungan *entropy* dan *gain* untuk node 7

Node	Atribut	v	Jumlah Kasus	Yes	No	Entropy	Gain
7	Total		6	3	3	1	
	<i>Outlook</i>	<i>Sunny</i>	2	1	1	1	0
		<i>Rain</i>	4	2	2	1	
	<i>Temperature</i>	$\leq 67,33$	1	0	1	0	0,1909
		$> 67,33$	5	3	2	0,971	
	<i>Humadity</i>	$\leq 73,33$	2	1	1	1	0
		$< 73,33$	4	2	2	1	
	<i>Wind</i>	<i>True</i>	2	0	2	0	0,4591
		<i>False</i>	4	3	1	0,8113	

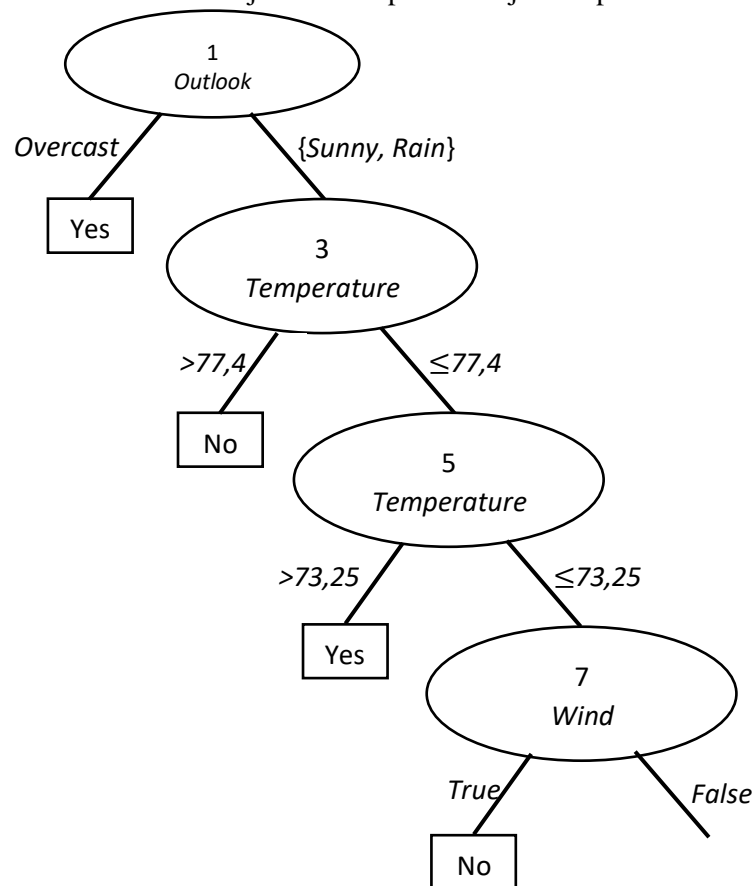
Hasil yang ditunjukkan pada Tabel 13 menunjukkan bahwa gain tertinggi ada di fitur “*Wind*”, sehingga fitur “*Wind*” dijadikan kondisi di node 7 seperti ditunjukkan pada Tabel 14.

Tabel 14. Pemisahan *Dataset Golf* Iterasi 4

<i>Outlook</i>	<i>Temperature</i>	<i>Humidity</i>	<i>Wind</i>	<i>Play</i>
<i>rain</i>	68	80	<i>FALSE</i>	<i>yes</i>
<i>sunny</i>	69	70	<i>FALSE</i>	<i>yes</i>
<i>rain</i>	70	96	<i>FALSE</i>	<i>yes</i>
<i>sunny</i>	72	95	<i>FALSE</i>	<i>no</i>
<i>rain</i>	65	70	<i>TRUE</i>	<i>no</i>
<i>rain</i>	71	80	<i>TRUE</i>	<i>no</i>

Proses *node ke-8*

Untuk node 8, nilai *entropy* yang didapat adalah 0 (karena semua baris pada *True* memiliki kelas yang sama). Oleh karena itu dipastikan bahwa *node 8* menjadi daun seperti ditunjukkan pada Gambar 8.



Gambar 8. Hasil pembentukan cabang di node ke-7 untuk kasus “Apakah harus bermain golf?”

Proses *node ke-9*

Selanjutnya, di *node 9* harus dihitung dulu *entropy* untuk sisa data terhadap komposisi yang tidak masuk dalam *node 8*.

Temperature

Pada atribut *Temperature* untuk mendapatkan nilai v menggunakan *Binning*. Adapun langkahnya adalah sebagai berikut:

1. Urutkan data dari kecil ke besar: {68, 69; 70, 72}
2. Bagi menjadi 2 bagian atau bin: {68, 69; 70, 72}
3. Hitung rata-rata masing-masing bin.
4. Maka nilai v yang didapatkan adalah 68,5 dan 71.

Jika sudah mendapatkan nilai v selanjutnya menghitung nilai *entropy* dan *gain*. Perhitungan nilai *gain* diperlihatkan Tabel 15.

Tabel 15. Posisi v untuk pemecahan fitur "*Temperature*" di node 9

Suhu		Jumlah Kasus	Yes	No	Entropy	Gain
$v = 68,5$	\leq	1	1	0	0	0,1226
	$>$	3	2	1	0,9183	
$v = 71$	\leq	3	3	0	0	0,8113
	$>$	1	0	1	0	

Humadity

Pada atribut *Humadity* untuk mendapatkan nilai v menggunakan *Binning*. Adapun langkahnya adalah sebagai berikut:

1. Urutkan data dari kecil ke besar: {70, 80; 95, 96}
2. Bagi menjadi 3 bagian atau bin: {70, 80; 95, 96}
3. Hitung rata-rata masing-masing bin.
4. Maka nilai v yang didapatkan adalah 75 dan 95,3.

Jika sudah mendapatkan nilai v selanjutnya menghitung nilai *entropy* dan *gain*. Perhitungan nilai *gain* diperlihatkan Tabel 16.

Tabel 16. Posisi v untuk pemecahan fitur "*Humadity*" di node 9

Suhu		Jumlah Kasus	Yes	No	Entropy	Gain
$v = 75$	\leq	1	1	0	0	0,1226
	$>$	3	2	1	0,9183	
$v = 95,5$	\leq	3	2	1	0,9183	0,1226
	$>$	1	1	0	0	

Selanjutnya dihitung *entropy* untuk setiap nilai fitur terhadap kelas, kemudian dihitung *gain* untuk setiap fitur. Hasil perhitungan disajikan pada Tabel 17.

Tabel 17. Hasil perhitungan *entropy* dan *gain* untuk node 9

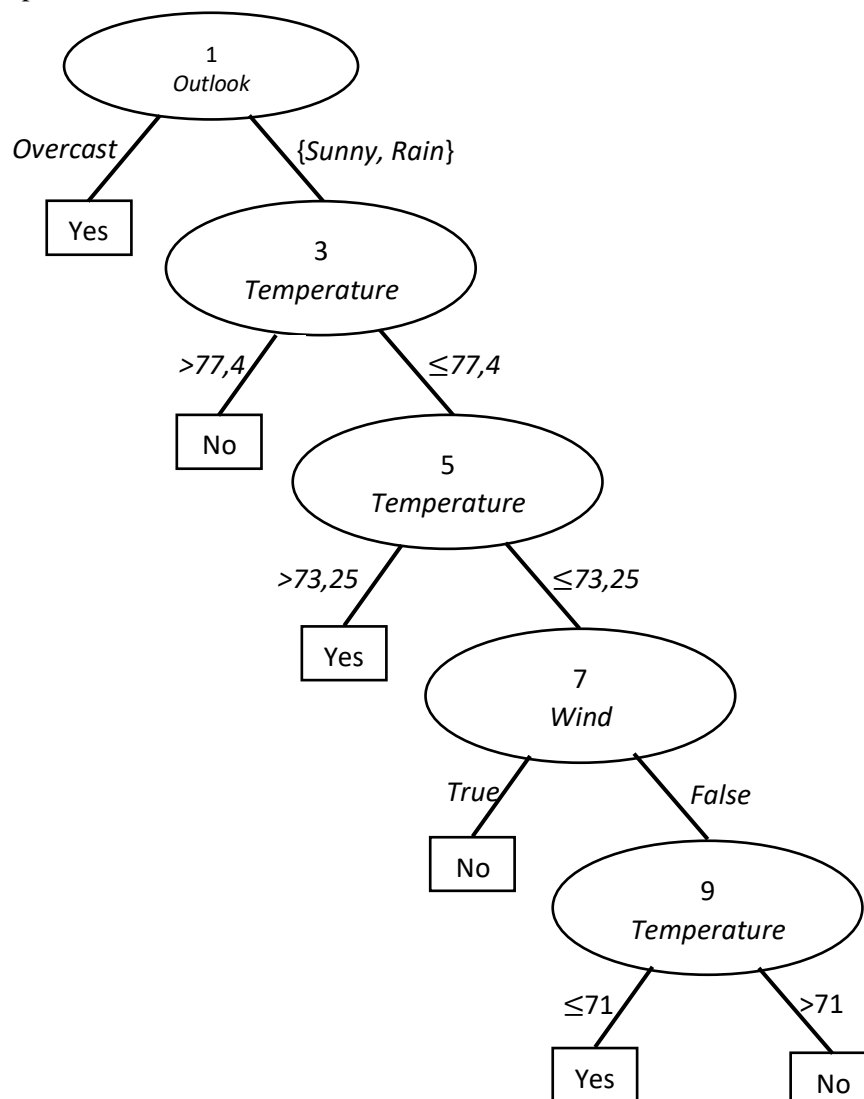
Node	Atribut	v	Jumlah Kasus	Yes	No	Entropy	Gain
9	Total		4	3	1	0,8113	
	<i>Outlook</i>	<i>Sunny</i>	2	1	1	1	0,3113
		<i>Rain</i>	2	0	2	0	
	<i>Temperature</i>	≤ 71	3	3	0	0	0,8113
		> 71	1	0	1	0	
	<i>Humadity</i>	≤ 75	3	2	1	0,9183	0,1226
		< 75	1	1	0	0	

Hasil yang ditunjukkan pada Tabel 17 menunjukkan bahwa gain tertinggi ada di fitur “Temperature”, sehingga fitur “Temperature” dijadikan kondisi di node 9 seperti ditunjukkan pada Tabel 18.

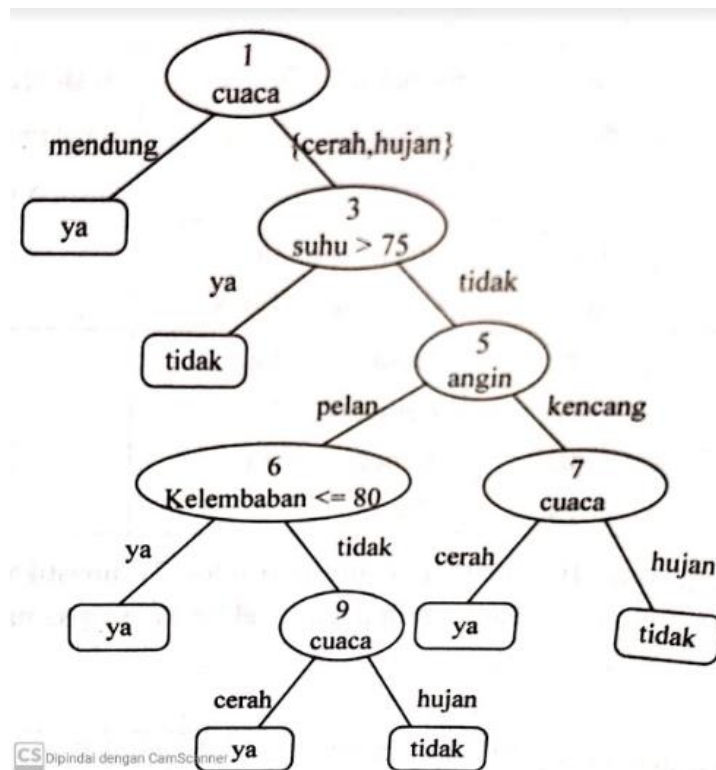
Tabel 18. Pemisahan *Dataset* Golf Iterasi 5

Outlook	Temperature	Humidity	Wind	Play
rain	68	80	FALSE	yes
sunny	69	70	FALSE	yes
rain	70	96	FALSE	yes
sunny	72	95	FALSE	no

Untuk node 10, nilai *entropy* yang didapat adalah 0 (karena semua baris pada ≤ 71 memiliki kelas yang sama). Begitu pula dengan node 11, nilai *entropy* yang didapat adalah 0 (karena semua baris pada > 71 memiliki kelas yang sama). Oleh karena itu dipastikan bahwa node 10 dan node 11 menjadi daun seperti ditunjukkan pada Gambar 9.

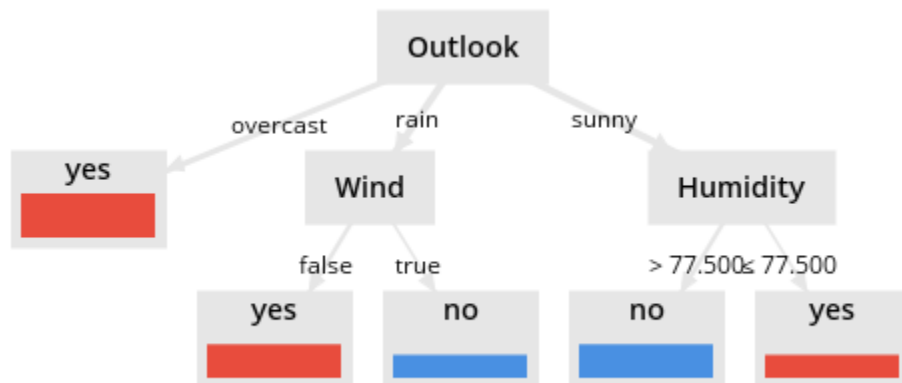


Gambar 9. Hasil pembentukan cabang di node ke-9 untuk kasus “Apakah harus bermain golf?”



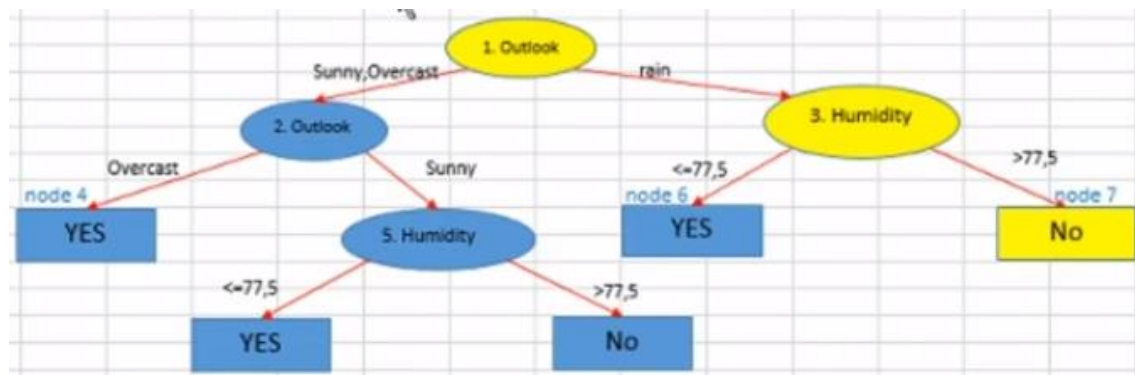
Gambar 10. Hasil pembentukan cabang di node ke-9 untuk kasus “Apakah harus bermain *baseball*?” [1]

Gambar adalah hasil perhitungan di buku Data Mining dengan penulis Eko Prasetyo dengan data yang sama, walaupun nama datasetnya berbeda. Perbedaan hanya pada penentuan nilai v untuk data tipe numerik.



Gambar 11. Hasil *tree* oleh RapidMiner untuk kasus “Apakah harus bermain golf?”

Gambar 11 adalah hasil tree dari Decision Tree (C4.5) RapidMiner.

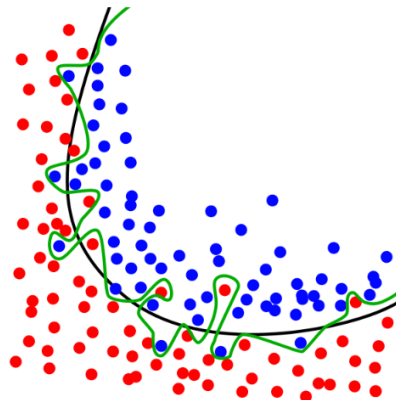
Gambar 12. Hasil pembentukan *tree* oleh kelompok Zunnah

Tabel 19. Data uji

Outlook	Temperature	Humidity	Wind	Play	Mut	Eko	RM	Zunnah
overcast	80.0	90.0	TRUE	No	Yes	Yes	Yes	Yes
rain	68.0	80.0	TRUE	Yes	No	No	No	No
sunny	75.0	80.0	FALSE	No	Yes	Yes	No	No
overcast	81.0	75.0	TRUE	No	Yes	Yes	Yes	Yes

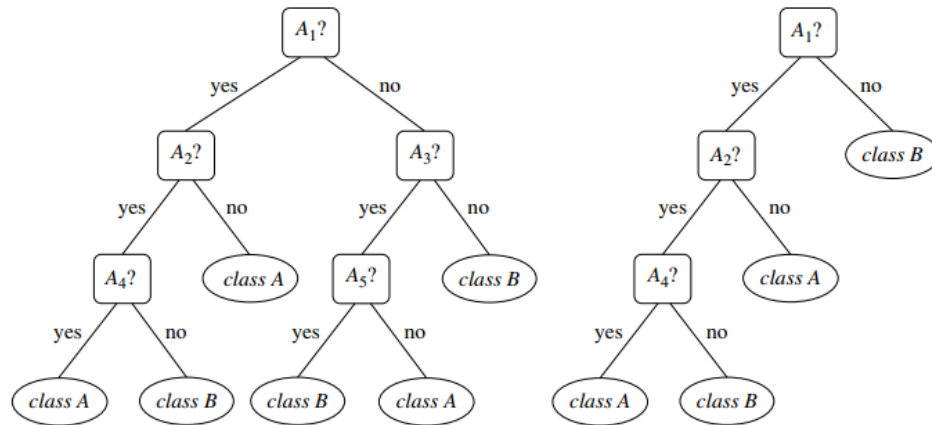
Berdasarkan Tabel 19, didapatkan kesimpulan bahwa hasil *tree* dari RapidMiner dan Zunnah, dkk yang penentuan nilai v untuk data bertipe numerik menggunakan *Gini Split* lebih baik dari pada hasil perhitungan menggunakan *Binning* seperti yang digunakan oleh Mut dan Eko.

Jika dilihat dari sisi kompleksitas *tree*-nya, milik Eko lebih tinggi (13 node), dilanjutkan milik Mut (11 node), Zunnah (9 node), dan RapidMiner (8 node) lebih sederhana karena menggunakan mekanisme *pruning*. Eko sudah merekomendasikan untuk melakukan *pruning* atau pemotongan, karena akan menyebabkan *overfitting* (Gambar 13), di mana hal ini dapat mempengaruhi *trade-off* antara akurasi prediksi terhadap pelatihan. Untuk permasalahan komputasi perlu untuk membuat aplikasi untuk membandingkan kecepatan prosesnya, apakah penentuan v pada nilai numerik menggunakan *binning* dengan *gini split* lebih lama mana? Ataukah node yang lebih sedikit yang lebih cepat proses komputasinya?.

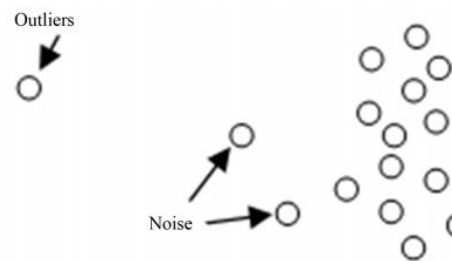
Gambar 13. Ilustrasi *overfitting* [3]

Keunggulan mekanisme *pruning* adalah pohon lebih ringkas, kompleksitas rendah, lebih baik dalam kinerja klasifikasi, dan lebih mudah dipahami. Metode *pruning* (Gambar 14) biasanya menggunakan langkah-langkah statistik untuk menghapus cabang yang tidak dapat diandalkan. Karena cabang yang banyak akan

menyebabkan anomali pada saat pelatihan, yang biasanya terjadi karena ada *outlier* dan *noise* (Gambar 15) [4].



Gambar 14. Hasil Tree sebelum dan sesudah di lakukan *pruning* [5]



Gambar 15. Ilustrasi *outliers* vs *noise* [6]

Outlier adalah data tetapi memiliki karakteristik yang aneh dibandingkan data lainnya, sedangkan *noise* adalah bukan data, atau sampah yang menyebabkan data tidak bersih. Contoh *outlier* adalah kaki kambing ada

References

- [1] E. Prasetyo, Data Mining Mengolah Data Menjadi Informasi Menggunakan Matlab, Yogyakarta: Andi Publisher, 2014.
- [2] B. Santoso, A. I. S. Azis and Zohrahayaty, Machine Learning & Reasoning Fuzzy Logic Algoritma, Manual, Matlab, RapidMiner, Yogyakarta: Deepublish Publisher, 2020.
- [3] Wikipedia, "Wikipedia," 2019. [Online]. Available: <https://id.wikipedia.org/wiki/Overfitting>.
- [4] D. T. Larose and C. D. Larose, Discovering Knowledge in Data: An Introduction to Data Mining, 2nd Edition ed., Wiley, 2014.
- [5] J. Han, M. Kamber and J. Pei, Data Mining: Concepts and Techniques, 3rd ed. ed., Waltham: Morgan Kaufmann Publishers, 2012.
- [6] V. A. Tran, O. Hirose, T. Saethang, L. A. T. Nguyen, X. T. Dang, T. K. T. Le, D. L. Ngo, G. Sergey, M. Kubo, Y. Yamada and K. Satou, "D-IMPACT: A Data Preprocessing Algorithm to Improve the Performance of Clustering," *Journal of Software Engineering and Applications*, vol. 7, pp. 639-654, 2014.

