

BAB II

LANDASAN TEORI

2.1 Tinjauan Studi

Tinjauan studi berguna bagi peneliti untuk menjadi bahan referensi berdasarkan pada penelitian-penelitian yang telah dilakukan sebelumnya. Dalam melaksanakan penelitian ini, penulis telah mengambil beberapa referensi sebagai dasar pelaksanaan penelitian, diantaranya adalah:

Menurut penelitian yang dilakukan oleh Dwi Kinasih Widiyati, Masna Wati, dan Herman Santoso Pakpahan pada 2018 dengan judul Penerapan Algoritma ID3 *Decision Tree* pada Penentuan Penerima Program Bantuan Pemerintah Daerah di Kabupaten Kutai Kartanegara. Dalam penelitian ini, peneliti menggunakan data berjumlah 123 yang diperoleh dari Dinas Sosial Kabupaten Kutai Kartanegara. Penentuan penerima bantuan dilakukan dengan menggunakan enam parameter yaitu umur, jenis kelamin, status pernikahan, pendidikan terakhir, pekerjaan, dan keterampilan dengan target menerima atau tidak menerima. Data yang ada selanjutnya akan diolah menggunakan algoritma *Iterative Dichotomiser Three* (ID3) untuk membentuk suatu model dalam menentukan penerima bantuan pemerintah daerah yang selanjutnya dilakukan pengujian menggunakan *Confusion Matrix* dan kurva ROC (*Receiver Operating Characteristics*)/AUC (*Area Under Cover*) menggunakan *RapidMiner*. Pengujian dilakukan berdasarkan rasio data training dengan interval 55%, 65%, 75%, 85%, dan 95%. Hasil dari penelitian ini menunjukkan nilai akurasi sebesar 97,89% dari data training 75% dengan data berjumlah 92 [5].

Menurut penelitian yang dilakukan oleh Abdul Rohman dan Anief Rufiyanto pada 2019 dengan judul Penerapan Algoritma *Decision Tree* ID3 untuk Prediksi Kelulusan Mahasiswa Jenjang Pendidikan D3 di Fakultas Teknik Universitas Pandanaran. Dalam penelitian ini, peneliti menggunakan data berjumlah 235 mahasiswa dengan jumlah mahasiswa lulus tepat waktu sebanyak 151 dan mahasiswa terlambat lulus sebanyak 84. Data tersebut diperoleh dari data

kelulusan mahasiswa D3 di Fakultas Teknik dari beberapa program studi yaitu Teknik Sipil, Teknik Mesin, Teknik Elektronika, Teknik Lingkungan, dan Teknik Kimia. Parameter yang ditentukan dalam penelitian ini berjumlah delapan, yaitu jurusan, umur, jenis kelamin, pekerjaan, IPS 1, IPS 2, IPS 3, dan IPS 4 dengan target lulus tepat waktu atau terlambat. Dari pengolahan data, dihasilkan sebuah pohon keputusan dengan aturan *rule* sebanyak 38. Setelah dilakukan pengolahan data menggunakan algoritma *Iterative Dichotomiser Three* (ID3) dan menghasilkan sebuah pohon keputusan, selanjutnya dilakukan pengujian dengan *K-Fold Cross Validation*. Dari pengujian yang telah dilakukan menghasilkan nilai akurasi sebesar 73,91% dengan nilai AUC 0,806 sehingga termasuk klasifikasi data yang baik [6].

Menurut penelitian yang dilakukan oleh Ibu Rasyid Munthe dan Volvo Sihombing pada 2018 berjudul Klasifikasi Algoritma *Iterative Dichotomiser* (ID3) untuk Tingkat Kepuasan pada Sarana Laboratorium Komputer. Dalam penelitian ini, peneliti mengumpulkan data dengan pemberian kuisioner kepada mahasiswa AMIK Labuan Batu. Parameter yang digunakan untuk proses pengolahan data berjumlah tiga, yaitu fasilitas penunjang, laboratorium komputer, dan minat berkunjung kembali dengan target puas atau tidak puas. Dari proses penerapan algoritma *Iterative Dichotomiser Three* (ID3), peneliti menghasilkan sebuah pohon cabang dengan jumlah aturan sebanyak empat *rule*. Pengolahan data menggunakan *RapidMiner* menghasilkan 73,8% dengan mahasiswa sebanyak 31 orang menyatakan puas dan 26,2% dengan mahasiswa sebanyak 11 orang menyatakan tidak puas. Untuk pengujian, dilakukan dengan pengukuran *Confusion Matrix* dengan nilai akurasi 0,95-1,00 sangat baik [7].

Menurut penelitian yang dilakukan oleh Rusda Wajhillah, Nurul Diah Komala, dan Jamal Maulana pada 2019 dengan judul Penerapan Metode Algoritma ID3 untuk Prediksi Diagnosa Gagal Ginjal Kronis (Studi Kasus: RSUD Sekarwangi Sukabumi). Dalam penelitian ini, peneliti memperoleh data dari RSUD Sekarwangi dengan jumlah data sebanyak 400, berupa data pasien positif GGK sebanyak 253 dan pasien negative GGK sebanyak 147. Parameter yang digunakan dalam pengolahan data sebanyak sembilan, yaitu usia, tekanan darah,

urine dipstick, *serum creatinine*, hemoglobin, hipertensi, diabetes mellitus, nafsu makan, dan anemia dengan target positif dan negatif dari Gagal Ginjal Kronis (GGK). Pengolahan data untuk membentuk sebuah model dilakukan menggunakan algoritma *Iterative Dichotomiser Three* (ID3). Dari proses yang dilakukan, terdapat aturan-aturan *rule* dari pohon keputusan sebanyak 20 *rule*. Untuk pengujian dilakukan dengan *confusion matrix* dengan hasil akurasi sebesar 96,08%, sehingga menunjukkan bahwa algoritma bekerja dengan sangat baik [8].

Berdasarkan penelitian diatas yang merupakan penelitian terdahulu, maka peneliti bermaksud untuk melakukan penelitian menggunakan algoritma yang sama tapi dengan data yang berbeda. Penelitian bertujuan untuk menentukan penerima program Hibah Air Minum Perdesaan (HAMP) di Kabupaten Jepara yang tepat sasaran. Alasan peneliti menggunakan algoritma ini karena hasil akhir berupa pohon keputusan sehingga mempermudah dalam menentukan penerima berdasarkan aturan-aturan *rule* yang nantinya dihasilkan. Selain itu, algoritma ini bersifat relatif mudah, cepat, namun tetap akurat sehingga mampu meminimalisir terjadinya kesalahan dalam menentukan penerima. Dalam pengujian terhadap model yang dihasilkan, peneliti menggunakan *k-fold validation* serta evaluasi dan validasi hasil menggunakan *confusion matrix* dan kurva ROC (*Receiver Operating Characteristics*) yang nantinya akan menghasilkan nilai AUC (*Area Under Cover*).

2.2 Tinjauan Pustaka

2.2.1 Program Hibah Air Minum Perdesaan (HAMP)

Program Hibah Air Minum Perdesaan (HAMP) merupakan hibah yang diberikan kepada Pemerintah Daerah dari Pemerintah Pusat sebagai sebuah tanggungjawab dalam menyelenggarakan pemenuhan sistem penyediaan air minum di perdesaan hingga terbangun dan berfungsinya layanan Sambungan Rumah (SR) berdasarkan kinerja yang terukur (*output based*).

Program Hibah Air Minum Perdesaan (HAMP) memiliki *output* berupa berfungsinya sambungan rumah dengan membangun sistem penyediaan air minum yang dapat diterima oleh MBR. Tujuan utama dari penyelenggaraan

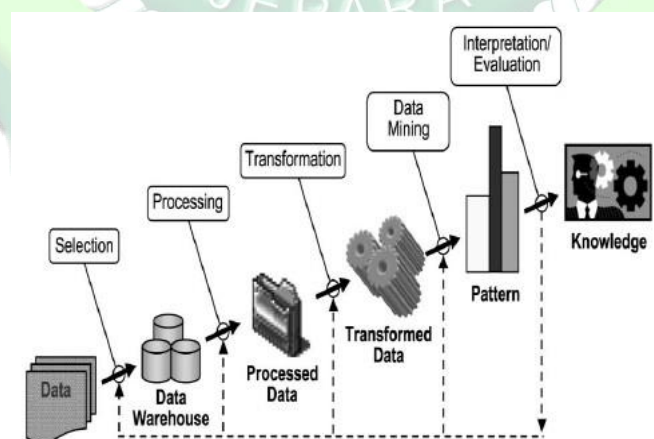
kegiatan ini adalah untuk meningkatkan kebutuhan pelayanan air minum perpipaan di kawasan perdesaan yang diprioritaskan bagi MBR sebagai upaya meningkatkan derajat kualitas kesehatan masyarakat [4].

2.2.2 Data Mining

Data mining atau biasa disebut *Knowledge Discovery in Database* merupakan sebuah kegiatan yang berhubungan dengan pengumpulan data atau penggunaan data historis dengan tujuan menemukan informasi, pengetahuan, keteraturan, pola atau hubungan data yang berukuran besar. Hasil dari data mining dapat digunakan sebagai alternatif dalam penentuan atau penentuan keputusan pada masa mendatang [9].

Dari definisi yang tertulis di atas, maka penulis dapat berkesimpulan bahwa data mining merupakan rangkaian proses pengumpulan data, baik dengan jumlah data yang besar atau kecil untuk memperoleh informasi dan pengetahuan di masa mendatang yang berguna dalam pengambilan keputusan.

Data mining atau *Knowledge Discovery in Databases* (KDD) kerap kali digunakan secara bergantian untuk menguraikan proses pencarian informasi dalam suatu data yang berukuran besar. Jika dibandingkan data mining dan *Knowledge Discovery in Databases* (KDD) memiliki konsep yang berbeda, namun memiliki keterkaitan satu sama lain [10]. *Knowledge Discovery in Databases* (KDD) memiliki proses yang dapat dijelaskan sebagai berikut:



Gambar 2.1 Proses *Knowledge Discovery in Databases* (KDD)

Sumber : Buku Data Mining untuk Perguruan Tinggi, 2020

1. *Data Selection*

Tahap awal dalam menjalankan kegiatan adalah dengan menyeleksi data yang diperoleh dari sekumpulan data berskala besar, sebelum nantinya akan diproses oleh data mining. Data hasil seleksi yang nantinya digunakan pada proses data mining, selanjutnya akan disimpan pada sebuah berkas dengan basis data operasional secara terpisah.

2. *Pre-processing / Cleaning*

Setelah proses pemilihan data, selanjutnya data akan melewati proses cleaning. Proses cleaning bertujuan untuk membuang data yang bersifat ganda, memeriksa data yang inkonsisten, dan memperbaiki kekeliruan pada data seperti kesalahan cetak (tipografi). Selain itu dilakukan proses *enrichment* yang memiliki tujuan untuk memperkaya data yang telah tersedia dengan data atau informasi yang relevan.

3. *Transformation*

Tahap ketiga dalam proses data mining adalah transformasi dalam pemilihan data. Peneliti harus mengenali data mining yang akan diterapkan dalam pengolahan data. Seperti pada algoritma *Iterative Dichotomiser Three* (ID3), tipe data yang digunakan adalah string. Jika data awal yang diperoleh peneliti bersifat *numeric*, maka perlu dilakukan pengolahan untuk mengubah tipe data tersebut.

4. *Data Mining*

Data mining adalah sebuah proses pencarian data atau informasi dari data yang terpilih menggunakan teknik atau metode tertentu. Pemilihan metode atau algoritma dapat menentukan hasil perhitungan dalam data mining. Hal tersebut karena karakteristik dari tiap metode yang berbeda.

5. *Interpretation / Evaluation*

Interpretation / evaluation adalah sebuah pola berisi informasi yang ditampilkan ke dalam bentuk yang mudah dan dapat dimengerti oleh pihak yang berkepentingan melalui proses yang dihasilkan dalam data mining. Hal tersebut dikarenakan mencakup pemeriksaan pola atau informasi yang ditemukan bersifat fakta atau hipotesis yang ada sebelumnya.

Menurut Budi Santosa dan Ardian Umam, data mining memiliki beberapa tugas yang biasa dilakukan [11]. Tugas tersebut diantaranya sebagai berikut:

1. Klastering

Klastering merupakan sebuah metode dalam pengelompokan obyek ke dalam beberapa kelompok berdasarkan kemiripan antar obyek. Dalam satu klaster, harus didapati obyek yang mirip dan obyek salin tidak mirip. Berbeda dengan metode klasifikasi, dalam pengolahannya klastering tidak memerlukan data latih yang sudah diberi label.

2. Klasifikasi

Klasifikasi merupakan sebuah metode untuk mengelompokkan obyek berdasarkan kelompok yang sudah tersedia. Seperti yang dijelaskan sebelumnya, jika klastering tidak memerlukan data latih yang sudah diberi label, maka pada klasifikasi data latih harus sudah diberi label. Proses prediksi dilakukan dengan membangun model melalui proses pelatihan dengan menggunakan data *training*.

3. Regresi/Estimasi

Regresi merupakan sebuah metode untuk mengetahui atau mencari model hubungan antara atribut predictor dan atribut dependen yang berupa nilai kontinyu. Berbeda dengan klasifikasi, output yang dihasilkan pada metode regresi bersifat nilai kontinyu. Sedangkan pada klasifikasi merupakan nilai bersifat diskrit.

4. Asosiasi

Asosiasi merupakan sebuah metode dengan melakukan asosiasi antar obyek dalam suatu set data. Data yang digunakan biasanya merupakan data transaksional. Asosiasi dilakukan dengan menghitung berapa kali set data suatu transaksi mengandung dua item atau lebih yang memiliki hubungan. Metode ini sering disebut *Market Basket Analysis*.

2.2.3 Algoritma Iterative Dichotomiser Three (ID3)

Iterative Dichotomiser Three (ID3) adalah sebuah algoritma pembelajaran pohon keputusan paling dasar yang dikembangkan oleh J. Ross Quinlan.

Algoritma ini menjalankan proses pencarian dengan menyeluruh, bahkan mungkin serakah untuk semua kemungkinan pada pohon keputusan. Algoritma ID3 berupaya untuk membangun pohon keputusan secara *top-down* atau dari atas ke bawah, mulai dengan mencari atribut pertama dan meletakkannya pada root [12] [13]. Dalam algoritma *Iterative Dichotomiser Three* (ID3) terdapat perhitungan untuk mencari nilai *entropy* dan *information gain* yang masing-masing memiliki rumus persamaan dan dijelaskan di bawah ini:

a. *Entropy*

Entropy merupakan keberbedaan atau keberagaman. Dalam data mining, *entropy* didefinisikan sebagai suatu parameter untuk mengukur heterogenitas (keberagaman) dalam suatu himpunan data. Semakin heterogen suatu himpunan data, semakin besar pula nilai *entropy*-nya [14]. *Entropy* dirumuskan dengan persamaan berikut:

$$\mathbf{Entropy} = \sum_i^c (-p_i * \log_2 p_i) \dots\dots\dots (2.1)$$

b. *Information Gain*

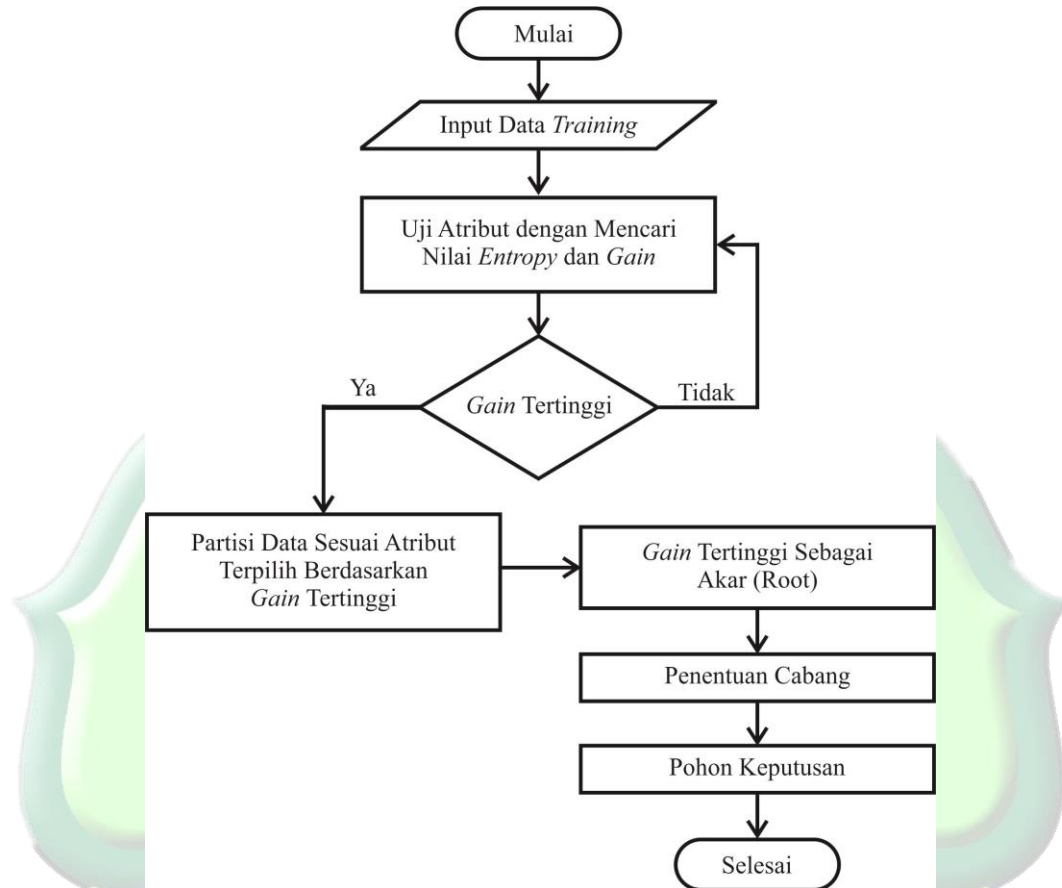
Information gain merupakan perolehan informasi. *Information gain* didefinisikan sebagai ukuran efektifitas suatu atribut dalam mengklasifikasi data [14]. *Information gain* dirumuskan dengan persamaan berikut:

$$\mathbf{Gain(S, A)} = \mathbf{Entropy(S)} - \sum_{A=1} \left(\frac{|S_i|}{|S|} * \mathbf{Entropy(S_i)} \right) \dots\dots\dots (2.2)$$

Keterangan:

- C = Jumlah nilai yang terdapat pada atribut target (jumlah kelas)
 Pi = Rasio antar jumlah sampel di kelas I dengan jumlah sampel pada himpunan data
 A = Atribut
 C = Nilai yang mungkin untuk atribut A
 |Si| = Jumlah sampel untuk nilai i
 |S| = Jumlah seluruh sampel data
 Entropy(Si) = Entropy untuk sampel-sampel yang memiliki nilai i

Berikut merupakan langkah-langkah dalam penerapan algoritma *Iterative Dichotomiser Three* (ID3) yang diilustrasikan pada gambar di bawah ini.



Gambar 2.2 Flowchart Algoritma *Iterative Dichotomiser Three* (ID3)

1. Menghitung jumlah sampel dan kejadian
2. Menghitung *Entropy* (Total)
3. Menghitung *Entropy Ratio* masing-masing atribut
4. Menghitung *Information Gain* untuk setiap atribut dan cari nilai tertinggi sebagai *root* (akar)
5. Menentukan cabang untuk tiap-tiap nilai
6. Pohon keputusan dan aturan

Setiap algoritma tentunya memiliki kelebihan dan kekurangan tersendiri. Berikut merupakan kelebihan dan kekurangan dari algoritma *Iterative Dichotomiser Three* (ID3) antara lain [14]:

1. Kelebihan
 - a. Mempunyai konsep yang jelas dan mudah dipahami karena menghasilkan pohon keputusan yang menjadi penentu model klasifikasi data mampu bekerja atau tidak melalui proses pelatihan yang telah dilakukan.
 - b. Mudah diimplementasikan dengan menggunakan algoritma rekursif.
2. Kekurangan
 - a. Sulit diterapkan pada kumpulan data berskala besar dengan ribuan hingga milyaran atribut dan objek data.
 - b. Mudah mengalami *overfit* karena proses pelatihan *greedy* tidak menjamin menghasilkan pohon keputusan yang optimum.

Untuk mengetahui lebih lanjut mengenai algoritma *Iterative Dichotomiser Three* (ID3), berikut adalah contoh perhitungannya:

Tabel 2.1 Contoh *Dataset*

Peserta SPMB	NS	UAN	PS	DTR
Peserta 1	Tinggi	Bagus	Baik	Ya
Peserta 2	Tinggi	Cukup	Baik	Ya
Peserta 3	Tinggi	Cukup	Buruk	Ya
Peserta 4	Tinggi	Kurang	Buruk	Tidak
Peserta 5	Sedang	Bagus	Baik	Ya
Peserta 6	Sedang	Cukup	Baik	Ya
Peserta 7	Sedang	Cukup	Buruk	Ya
Peserta 8	Sedang	Kurang	Buruk	Tidak
Peserta 9	Rendah	Bagus	Baik	Ya
Peserta 10	Rendah	Cukup	Buruk	Tidak
Peserta 11	Rendah	Kurang	Baik	Ya

Tabel 2.1 merupakan contoh *dataset* yang sudah ditransformasikan dari data *numeric* menjadi *string* yang sudah dikategorikan, dengan ketentuan sebagai berikut:

NS = Nilai SPMB, dengan kategori Tinggi, Sedang, dan Rendah

UAN	= UAN, dengan kategori Bagus, Kurang, dan Cukup
PS	= Psikotest, dengan kategori Baik dan Buruk
DTR	= Diterima, dengan kategori Ya dan Tidak

Setelah menentukan *dataset*, langkah selanjutnya adalah menerapkan algoritma *Iterative Dichotomiser Three* (ID3). Adapun tahapan perhitungannya adalah sebagai berikut:

1. Menghitung jumlah sampel dan kejadian untuk DITERIMA (DTR) sehingga diperoleh:

Jumlah sampel, dinotasikan dengan $n = 11$.

Jumlah kejadian Diterima Ya, dinotasikan dengan $n(\text{Ya}) = 8$.

Jumlah kejadian Diterima Tidak, dinotasikan dengan $n(\text{Tidak}) = 3$.

2. Menghitung *Entropy* (Total)

Langkah selanjutnya yang dilakukan adalah menghitung *entropy* total menggunakan persamaan (2.1), sehingga diperoleh:

$$Entropy(\text{Total}) = - \left(\left(\frac{n(\text{Ya})}{n} \right) * \frac{\log(n(\text{Ya})/n)}{\log(2)} + \left(\frac{n(\text{Tidak})}{n} \right) * \frac{\log(n(\text{Tidak})/n)}{\log(2)} \right)$$

$$Entropy(\text{Total}) = - \left(\left(\frac{8}{11} \right) * \frac{\log(8/11)}{\log(2)} + \left(\frac{3}{11} \right) * \frac{\log(3/11)}{\log(2)} \right)$$

$$Entropy(\text{Total}) = -(-0,845350)$$

$$Entropy(\text{Total}) = 0,845350$$

3. Menghitung *Entropy Ratio* masing-masing atribut

Untuk mencari *entropy ratio* dari *dataset* pada Tabel 2.1, atribut yang akan melalui proses perhitungan adalah Nilai SPMB, UAN, dan Psikotest sebagai berikut:

- Nilai SPMB

- a. Kategori Tinggi

Jumlah kejadian Nilai SPMB Tinggi, dinotasikan dengan $n(\text{Tinggi}) = 4$, dengan $n(\text{Ya}) = 3$ dan $n(\text{Tidak}) = 1$.

$$Entropy(\text{Tinggi}) = - \left(\left(\frac{3}{4} \right) * \frac{\log(3/4)}{\log(2)} + \left(\frac{1}{4} \right) * \frac{\log(1/4)}{\log(2)} \right)$$

$$\text{Entropy (Tinggi)} = -(-0,811278)$$

$$\text{Entropy (Tinggi)} = 0,811278$$

b. Kategori Sedang

Jumlah kejadian Nilai SPMB Sedang, dinotasikan dengan $n(\text{Sedang}) = 4$, dengan $n(\text{Ya}) = 3$ dan $n(\text{Tidak}) = 1$.

$$\text{Entropy(Sedang)} = -\left(\left(\frac{3}{4}\right) * \frac{\log(3/4)}{\log(2)} + \left(\frac{1}{4}\right) * \frac{\log(1/4)}{\log(2)}\right)$$

$$\text{Entropy (Sedang)} = -(-0,811278)$$

$$\text{Entropy (Sedang)} = 0,811278$$

c. Kategori Rendah

Jumlah kejadian Nilai SPMB Rendah, dinotasikan dengan $n(\text{Rendah}) = 3$, dengan $n(\text{Ya}) = 2$ dan $n(\text{Tidak}) = 1$.

$$\text{Entropy(Sedang)} = -\left(\left(\frac{2}{3}\right) * \frac{\log(2/3)}{\log(2)} + \left(\frac{1}{3}\right) * \frac{\log(1/3)}{\log(2)}\right)$$

$$\text{Entropy (Sedang)} = -(-0,918295)$$

$$\text{Entropy (Sedang)} = 0,918295$$

• UAN

a. Kategori Bagus

Jumlah kejadian UAN Bagus , dinotasikan dengan $n(\text{Bagus}) = 3$, dengan $n(\text{Ya}) = 3$ dan $n(\text{Tidak}) = 0$.

$$\text{Entropy(Bagus)} = -\left(\left(\frac{3}{3}\right) * \frac{\log(3/3)}{\log(2)} + \left(\frac{0}{3}\right) * \frac{\log(0/3)}{\log(2)}\right)$$

$$\text{Entropy (Bagus)} = -(-0)$$

$$\text{Entropy (Bagus)} = 0$$

b. Kategori Kurang

Jumlah kejadian UAN Kurang, dinotasikan dengan $n(\text{Kurang}) = 3$, dengan $n(\text{Ya}) = 1$ dan $n(\text{Tidak}) = 2$.

$$\text{Entropy(Kurang)} = -\left(\left(\frac{1}{3}\right) * \frac{\log(1/3)}{\log(2)} + \left(\frac{2}{3}\right) * \frac{\log(2/3)}{\log(2)}\right)$$

$$\text{Entropy (Kurang)} = -(-0,918295)$$

$$\text{Entropy (Kurang)} = 0,918295$$

c. Kategori Cukup

Jumlah kejadian UAN Cukup, dinotasikan dengan $n(\text{Cukup}) = 5$, dengan $n(\text{Ya}) = 4$ dan $n(\text{Tidak}) = 1$.

$$\text{Entropy}(\text{Cukup}) = - \left(\left(\frac{4}{5} \right) * \frac{\log(4/5)}{\log(2)} + \left(\frac{1}{5} \right) * \frac{\log(1/5)}{\log(2)} \right)$$

$$\text{Entropy}(\text{Cukup}) = -(-0,721928)$$

$$\text{Entropy}(\text{Cukup}) = 0,721928$$

- Psikotest

- a. Kategori Baik

Jumlah kejadian Psikotest Baik, dinotasikan dengan $n(\text{Baik}) = 6$, dengan $n(\text{Ya}) = 6$ dan $n(\text{Tidak}) = 0$.

$$\text{Entropy}(\text{Baik}) = - \left(\left(\frac{6}{6} \right) * \frac{\log(6/6)}{\log(2)} + \left(\frac{0}{6} \right) * \frac{\log(0/6)}{\log(2)} \right)$$

$$\text{Entropy}(\text{Baik}) = -(-0)$$

$$\text{Entropy}(\text{Baik}) = 0$$

- b. Kategori Buruk

Jumlah kejadian Psikotest Buruk, dinotasikan dengan $n(\text{Buruk}) = 5$, dengan $n(\text{Ya}) = 2$ dan $n(\text{Tidak}) = 3$.

$$\text{Entropy}(\text{Kurang}) = - \left(\left(\frac{2}{5} \right) * \frac{\log(2/5)}{\log(2)} + \left(\frac{3}{5} \right) * \frac{\log(3/5)}{\log(2)} \right)$$

$$\text{Entropy}(\text{Buruk}) = -(-0,970950)$$

$$\text{Entropy}(\text{Buruk}) = 0,970950$$

4. Menghitung *Information Gain* untuk setiap atribut dan cari nilai tertinggi sebagai *root* (akar)

Untuk mengetahui atribut dengan nilai tertinggi, maka perlu dilakukan perhitungan *information gain* yang diperlihatkan pada perhitungan dan tabel sebagai berikut:

- Nilai SPMB

$\text{Gain} = \text{Entropy}(\text{Total}) -$

$$\left(\left(\frac{n(\text{Tinggi})}{n} * \text{Entropy } n(\text{Tinggi}) \right) + \left(\frac{n(\text{Sedang})}{n} * \text{Entropy } n(\text{Sedang}) \right) + \left(\frac{n(\text{Rendah})}{n} * \text{Entropy } n(\text{Rendah}) \right) \right)$$

$$\text{Gain} = 0,845350 - \left(\left(\frac{4}{11} * 0,811278 \right) + \left(\frac{4}{11} * 0,811278 \right) + \left(\frac{3}{11} * 0,918295 \right) \right)$$

$$Gain = 0,004886$$

- UAN

$$Gain = Entropy (Total) -$$

$$\left(\left(\frac{n(Bagus)}{n} * Entropy n(Bagus) \right) + \left(\frac{n(Kurang)}{n} * Entropy n(Kurang) \right) + \left(\frac{n(Cukup)}{n} * Entropy n(Cukup) \right) \right)$$

$$Gain = 0,845350 - \left(\left(\frac{3}{11} * 0 \right) + \left(\frac{3}{11} * 0,918295 \right) + \left(\frac{5}{11} * 0,721928 \right) \right)$$

$$Gain = 0,266757$$

- Psikotest

$$Gain = Entropy (Total) -$$

$$\left(\left(\frac{n(Baik)}{n} * Entropy n(Baik) \right) + \left(\frac{n(Buruk)}{n} * Entropy n(Buruk) \right) \right)$$

$$Gain = 0,845350 - \left(\left(\frac{6}{11} * 0 \right) + \left(\frac{5}{11} * 0,970950 \right) \right)$$

$$Gain = 0,404009$$

Tabel 2.2 Hasil Perhitungan *Gain* Tertinggi pada *Node* 1

Node	Atribut	Kategori	Jumlah Sampel (n)	Ya	Tidak	Entropy	Gain
1	Total		11	8	3	0,84535	
	Nilai SPMB	Tinggi	4	3	1	0,81128	0,004886
		Sedang	4	3	1	0,81128	
		Rendah	3	2	1	0,9183	
	UAN	Bagus	3	3	0	0	0,266757
		Kurang	3	1	2	0,9183	
		Cukup	5	4	1	0,72193	
	Psikotest	Baik	6	6	0	0	0,404009
Buruk		5	2	3	0,97095		

Karena atribut psikotest memiliki nilai *gain* terbesar, maka atribut Psikotest menjadi *root node*. Pada atribut Psikotest, kategori Baik memiliki 6 kasus dengan semua jawabannya adalah Ya. Sehingga tidak diperlukan perhitungan lebih lanjut dan Psikotest = Baik menjadi daun atau leaf.

5. Menentukan cabang untuk tiap-tiap nilai

Setelah diperoleh root node 1. Psikotest, maka langkah selanjutnya adalah mencari atribut sebagai node 1.1. Untuk memudahkan, *dataset* difilter dengan mengambil data yang memiliki Psikotest = Buruk untuk membuat tabel Node 1.1.

Langkah ini dilakukan dengan membuat cabang untuk tiap-tiap nilai, diantaranya sebagai berikut:

- Menghitung *entropy* total dari atribut Psikotest = Buruk

Jumlah sampel Psikotest = Buruk, dinotasikan dengan $n = 5$.

Jumlah kejadian Psikotest = Buruk dan Diterima Ya, dinotasikan dengan $n(Ya) = 2$.

Jumlah kejadian Psikotest = Buruk dan Diterima Tidak, dinotasikan dengan $n(Ya) = 3$.

Maka:

$$Entropy(Total) = - \left(\left(\frac{n(Ya)}{n} \right) * \frac{\log(n(Ya)/n)}{\log(2)} + \left(\frac{n(Tidak)}{n} \right) * \frac{\log(n(Tidak)/n)}{\log(2)} \right)$$

$$Entropy(Total) = - \left(\left(\frac{2}{5} \right) * \frac{\log(2/5)}{\log(2)} + \left(\frac{3}{5} \right) * \frac{\log(3/5)}{\log(2)} \right)$$

$$Entropy(Total) = -(-0,970950)$$

$$Entropy(Total) = 0,970950$$

- Menghitung *entropy rasion* dari atribut Nilai SPMB

a. Kategori Tinggi

Jumlah kejadian Nilai SPMB Tinggi, dinotasikan dengan $n(Tinggi) = 2$, dengan $n(Ya) = 1$ dan $n(Tidak) = 1$.

$$Entropy(Tinggi) = - \left(\left(\frac{1}{2} \right) * \frac{\log(1/2)}{\log(2)} + \left(\frac{1}{2} \right) * \frac{\log(1/2)}{\log(2)} \right)$$

$$Entropy(Tinggi) = -(-1)$$

$$Entropy(Tinggi) = 1$$

b. Kategori Sedang

Jumlah kejadian Nilai SPMB Sedang, dinotasikan dengan $n(Sedang) = 2$, dengan $n(Ya) = 1$ dan $n(Tidak) = 1$.

$$Entropy(Sedang) = - \left(\left(\frac{1}{2} \right) * \frac{\log(1/2)}{\log(2)} + \left(\frac{1}{2} \right) * \frac{\log(1/2)}{\log(2)} \right)$$

$$Entropy(Sedang) = -(-1)$$

$$Entropy(Sedang) = 1$$

c. Kategori Rendah

Jumlah kejadian Nilai SPMB Rendah, dinotasikan dengan $n(\text{Rendah}) = 1$, dengan $n(\text{Ya}) = 0$ dan $n(\text{Tidak}) = 1$.

$$\text{Entropy}(\text{Rendah}) = - \left(\left(\frac{0}{1} \right) * \frac{\log(0/1)}{\log(2)} + \left(\frac{1}{1} \right) * \frac{\log(1/1)}{\log(2)} \right)$$

$$\text{Entropy}(\text{Rendah}) = -(-0)$$

$$\text{Entropy}(\text{Rendah}) = 0$$

- Menghitung *entropy ration* dari atribut UAN

a. Kategori Bagus

Berdasarkan perhitungan sebelumnya, pada UAN = Bagus memiliki 3 kasus dengan semua jawabannya adalah Ya. Sehingga tidak diperlukan perhitungan lebih lanjut dan nantinya akan menjadi daun atau *leaf*.

b. Kategori Kurang

Jumlah kejadian UAN Kurang, dinotasikan dengan $n(\text{Kurang}) = 2$, dengan $n(\text{Ya}) = 0$ dan $n(\text{Tidak}) = 2$.

$$\text{Entropy}(\text{Kurang}) = - \left(\left(\frac{0}{2} \right) * \frac{\log(0/2)}{\log(2)} + \left(\frac{2}{2} \right) * \frac{\log(2/2)}{\log(2)} \right)$$

$$\text{Entropy}(\text{Kurang}) = -(-0)$$

$$\text{Entropy}(\text{Kurang}) = 0$$

c. Kategori Cukup

Jumlah kejadian UAN Cukup, dinotasikan dengan $n(\text{Cukup}) = 3$, dengan $n(\text{Ya}) = 2$ dan $n(\text{Tidak}) = 1$.

$$\text{Entropy}(\text{Cukup}) = - \left(\left(\frac{2}{3} \right) * \frac{\log(2/3)}{\log(2)} + \left(\frac{1}{3} \right) * \frac{\log(1/3)}{\log(2)} \right)$$

$$\text{Entropy}(\text{Cukup}) = -(-0,918295)$$

$$\text{Entropy}(\text{Cukup}) = 0,918295$$

- Menghitung *Information Gain* pada atribut Nilai SPMB dan UAN untuk mengetahui atribut dengan nilai Gain tertinggi menggunakan persamaan (2.2), sehingga diperoleh:

a. Nilai SPMB

$$\text{Gain} = \text{Entropy}(\text{Total}) -$$

$$\left(\left(\frac{n(\text{Tinggi})}{n} * \text{Entropy } n(\text{Tinggi}) \right) + \left(\frac{n(\text{Sedang})}{n} * \text{Entropy } n(\text{Sedang}) \right) + \left(\frac{n(\text{Rendah})}{n} * \text{Entropy } n(\text{Rendah}) \right) \right)$$

$$\text{Gain} = 0,970950 - \left(\left(\frac{2}{5} * 1 \right) + \left(\frac{2}{5} * 1 \right) + \left(\frac{1}{5} * 0 \right) \right)$$

$$\text{Gain} = 0,0,170950$$

b. UAN

$$\text{Gain} = \text{Entropy } (Total) - \left(\left(\frac{n(\text{Kurang})}{n} * \text{Entropy } n(\text{Kurang}) \right) + \left(\frac{n(\text{Cukup})}{n} * \text{Entropy } n(\text{Cukup}) \right) \right)$$

$$\text{Gain} = 0,970950 - \left(\left(\frac{2}{5} * 0 \right) + \left(\frac{3}{5} * 0,918295 \right) \right)$$

$$\text{Gain} = 0,419973$$

Tabel 2.3 Hasil Perhitungan *Gain* Tertinggi pada Node 1.1

Node	Atribut	Kategori	Jumlah Sampel (n)	Ya	Tidak	Entropy	Gain
1.1	Psikotest = Buruk		5	2	3	0,97095	0,17095
	Nilai SPMB	Tinggi	2	1	1	1	
		Sedang	2	1	1	1	
		Rendah	1	0	1	0	
	UAN	Kurang	2	0	2	0	0,419973
Cukup		3	2	1	0,918295		

Karena atribut UAN memiliki nilai *gain* terbesar, maka atribut UAN menjadi cabang dari Psikotest = Buruk. Pada atribut UAN, kategori Kurang memiliki 2 kasus dengan semua jawabannya adalah Tidak. Sehingga tidak diperlukan perhitungan lebih lanjut dan UAN = Kurang menjadi daun atau *leaf*.

Setelah diperoleh *root node* 1.1 UAN, maka langkah selanjutnya adalah mencari atribut sebagai *node* 1.1.2. Untuk memudahkan, dataset difilter dengan mengambil data yang memiliki UAN = Cukup untuk membuat tabel *Node* 1.1.2 Langkah ini dilakukan dengan membuat cabang untuk tiap-tiap nilai, diantaranya sebagai berikut:

- Menghitung *entropy* total dari atribut UAN = Cukup

Jumlah sampel UAN = Cukup, dinotasikan dengan $n = 5$.

Jumlah kejadian UAN = Cukup dan Diterima Ya, dinotasikan dengan $n(\text{Ya}) = 4$.

Jumlah kejadian Psikotest = Buruk dan Diterima Tidak, dinotasikan dengan $n(\text{Ya}) = 1$.

Maka:

$$Entropy(Total) = - \left(\left(\frac{n(Ya)}{n} \right) * \frac{\log(n(Ya)/n)}{\log(2)} + \left(\frac{n(Tidak)}{n} \right) * \frac{\log(n(Tidak)/n)}{\log(2)} \right)$$

$$Entropy(Total) = - \left(\left(\frac{4}{5} \right) * \frac{\log(4/5)}{\log(2)} + \left(\frac{1}{5} \right) * \frac{\log(1/5)}{\log(2)} \right)$$

$$Entropy(Total) = -(-0,721928)$$

$$Entropy(Total) = 0,721928$$

- Menghitung *entropy ration* dari atribut Nilai SPMB

a. Kategori Tinggi

Jumlah kejadian Nilai SPMB Tinggi, dinotasikan dengan $n(\text{Tinggi}) = 2$, dengan $n(Ya) = 2$ dan $n(\text{Tidak}) = 0$.

$$Entropy(Tinggi) = - \left(\left(\frac{2}{2} \right) * \frac{\log(2/2)}{\log(2)} + \left(\frac{0}{2} \right) * \frac{\log(0/2)}{\log(2)} \right)$$

$$Entropy(Tinggi) = -(-0)$$

$$Entropy(Tinggi) = 0$$

b. Kategori Sedang

Jumlah kejadian Nilai SPMB Sedang, dinotasikan dengan $n(\text{Sedang}) = 2$, dengan $n(Ya) = 2$ dan $n(\text{Tidak}) = 0$.

$$Entropy(Sedang) = - \left(\left(\frac{2}{2} \right) * \frac{\log(2/2)}{\log(2)} + \left(\frac{0}{2} \right) * \frac{\log(0/2)}{\log(2)} \right)$$

$$Entropy(Sedang) = -(-0)$$

$$Entropy(Sedang) = 0$$

c. Kategori Rendah

Jumlah kejadian Nilai SPMB Rendah, dinotasikan dengan $n(\text{Rendah}) = 1$, dengan $n(Ya) = 0$ dan $n(\text{Tidak}) = 1$.

$$Entropy(Rendah) = - \left(\left(\frac{0}{1} \right) * \frac{\log(0/1)}{\log(2)} + \left(\frac{1}{1} \right) * \frac{\log(1/1)}{\log(2)} \right)$$

$$Entropy(Rendah) = -(-0)$$

$$Entropy(Rendah) = 0$$

- Menghitung *entropy ration* dari atribut Psikotest

a. Kategori Baik

Jumlah kejadian Psikotest Baik, dinotasikan dengan $n(\text{Baik}) = 2$, dengan $n(\text{Ya}) = 2$ dan $n(\text{Tidak}) = 0$.

$$\text{Entropy}(\text{Baik}) = - \left(\left(\frac{2}{2} \right) * \frac{\log(2/2)}{\log(2)} + \left(\frac{0}{2} \right) * \frac{\log(0/2)}{\log(2)} \right)$$

$$\text{Entropy}(\text{Baik}) = -(-0)$$

$$\text{Entropy}(\text{Baik}) = 0$$

b. Kategori Buruk

Jumlah kejadian Psikotest Buruk, dinotasikan dengan $n(\text{Buruk}) = 3$, dengan $n(\text{Ya}) = 2$ dan $n(\text{Tidak}) = 1$.

$$\text{Entropy}(\text{Buruk}) = - \left(\left(\frac{2}{3} \right) * \frac{\log(2/3)}{\log(2)} + \left(\frac{1}{3} \right) * \frac{\log(1/3)}{\log(2)} \right)$$

$$\text{Entropy}(\text{Buruk}) = -(-0,918295)$$

$$\text{Entropy}(\text{Buruk}) = 0,918295$$

- Menghitung *Information Gain* pada atribut Nilai SPMB dan Psikotest untuk mengetahui atribut dengan nilai Gain tertinggi

c. Nilai SPMB

$\text{Gain} = \text{Entropy}(\text{Total}) -$

$$\left(\left(\frac{n(\text{Tinggi})}{n} * \text{Entropy } n(\text{Tinggi}) \right) + \left(\frac{n(\text{Sedang})}{n} * \text{Entropy } n(\text{Sedang}) \right) + \left(\frac{n(\text{Rendah})}{n} * \text{Entropy } n(\text{Rendah}) \right) \right)$$

$$\text{Gain} = 0,721928 - \left(\left(\frac{2}{5} * 0 \right) + \left(\frac{2}{5} * 0 \right) + \left(\frac{1}{5} * 0 \right) \right)$$

$$\text{Gain} = 0,721928$$

d. UAN

$\text{Gain} = \text{Entropy}(\text{Total}) -$

$$\left(\left(\frac{n(\text{Baik})}{n} * \text{Entropy } n(\text{Baik}) \right) + \left(\frac{n(\text{Buruk})}{n} * \text{Entropy } n(\text{Buruk}) \right) \right)$$

$$\text{Gain} = 0,721928 - \left(\left(\frac{2}{5} * 0 \right) + \left(\frac{3}{5} * 0,918295 \right) \right)$$

$$\text{Gain} = 0,721928$$

Tabel 2.4 Hasil Perhitungan *Gain* Tertinggi pada *Node* 1.1.2

Node	Atribut	Kategori	Jumlah Sampel (n)	Ya	Tidak	Entropy	Gain
1.1.2	UAN = Cukup		5	4	1	0,721928	0,17095
	Nilai SPMB	Tinggi	2	2	0	0	
		Sedang	2	2	0	0	
		Rendah	1	0	1	0	
	Psikotest	Baik	2	2	0	0	
		Buruk	3	2	1	0,918295	

6. Pohon Keputusan dan Aturan

Karena atribut Nilai SPMB memiliki nilai *Gain* terbesar, maka atribut Nilai SPMB menjadi *node* cabang dari atribut UAN = Cukup. Pada Nilai SPMB = Tinggi dan Nilai SPMB = Sedang masing-masing memiliki 2 kasus dengan jawaban Ya. Sedangkan pada Nilai SPMB = Rendah memiliki 1 kasus dengan jawaban Tidak. Maka, Nilai SPMB = Tinggi, Nilai SPMB = Sedang, Nilai SPMB = Rendah menjadi daun atau *leaf* yang selanjutnya tidak perlu dilakukan perhitungan kembali. Demikian telah peneliti uraikan langkah-langkah pengerjaan dataset menggunakan algoritma *Iterative Dichotomiser Three* (ID3). Untuk pohon keputusan dan *rule* akan diuraikan lebih jelas pada tinjauan teori pohon keputusan.

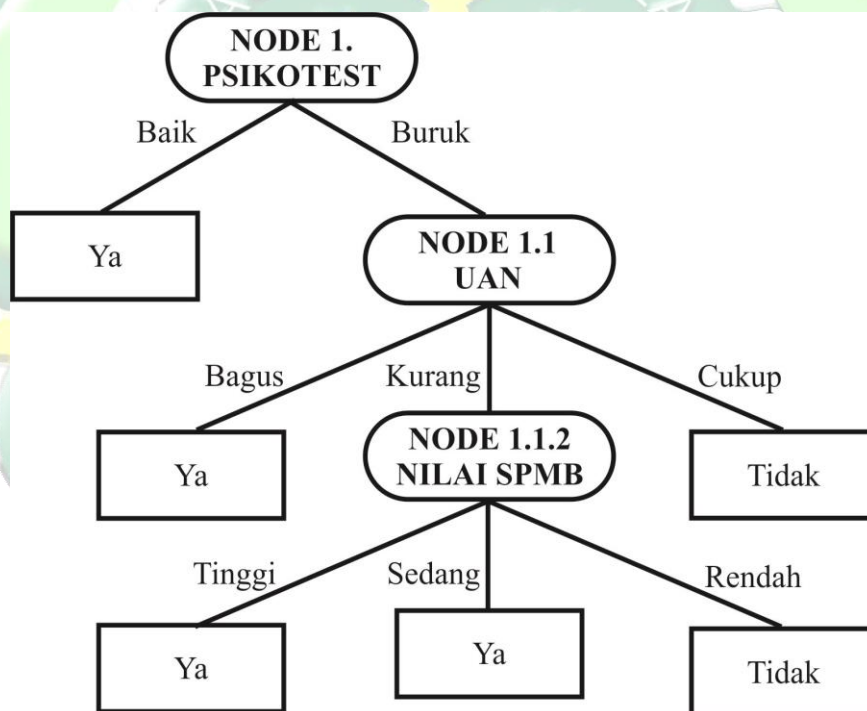
2.2.4 Pohon Keputusan

Pohon keputusan (*Decision Tree*) adalah proses dalam menganalisis pemecahan masalah atau pemetaan terkait beberapa alternatif dalam pemecahan masalah menggunakan gambaran sebuah pohon [15]. Pohon keputusan juga dapat dikatakan sebagai salah satu metode dalam klasifikasi yang mudah dipahami oleh setiap orang (umum). Dari pohon keputusan yang dihasilkan dalam decision tree akan menghasilkan beberapa aturan keputusan/*rule* [5] [16]. Pohon keputusan memiliki tiga jenis node, yaitu:

1. *Root Node*, merupakan *node* diposisi paling atas yang tidak memiliki *input* dan memiliki *output* lebih dari satu atau tidak sama sekali.

2. *Internal Node*, merupakan *node* dengan tugas sebagai cabang yang hanya memiliki satu *input* dengan minimal *output* sebanyak dua.
3. *Leaf Node* atau *Terminal Node*, merupakan *node* dengan posisi paling akhir yang memiliki satu *input* dan tidak memiliki *output*.

Manfaat yang paling diunggulkan dalam penggunaan pohon keputusan adalah kemampuannya dalam menyimpulkan hasil yang mulanya kompleks menjadi simpel sehingga lebih mudah dipahami karena langsung menginterpretasikan solusi dari permasalahan yang ada. Selain itu, kegunaan lain dari pohon keputusan adalah mampu mengeksplorasi data, menentukan hubungan yang belum diketahui antar calon variabel inputan dan variabel target [17]. Berikut peneliti sertakan pohon keputusan dari hasil perhitungan dataset sebelumnya.



Gambar 2.3 Pohon Keputusan

2.2.5 Cross Validation

Cross validation adalah sebuah permodelan statistik yang berfungsi atau memiliki tujuan untuk mengevaluasi kinerja algoritma atau model. Dalam penelitian ini, peneliti menggunakan *K-fold cross validation* karena dapat mengurangi waktu komputasi tanpa mengurangi hasil dari akurasi estimasi. Sebagai contoh metode *3-fold cross validation*, dengan pembagian *dataset* secara acak menjadi 3 bagian dan dilakukan percobaan sebanyak 3 kali untuk *training* dan *testing*. Pada tiap bagian, akan disisakan satu *dataset* sebagai *testing* dan satu *dataset* lainnya sebagai *training* [18]. Berikut merupakan pembagian *dataset* pada *3-fold cross validation* :

Tabel 2.5 Contoh Pembagian *Dataset*

Data Training	Data Testing
<i>Dataset 2 dan Dataset 3</i>	<i>Dataset 1</i>
<i>Dataset 1 dan Dataset 3</i>	<i>Dataset 2</i>
<i>Dataset 1 dan Dataset 2</i>	<i>Dataset 3</i>

2.2.6 Confusion Matrix

Confusion matrix berfungsi untuk menampilkan hasil evaluasi pada sebuah algoritma. *Confusion matrix* merupakan sebuah *tool* yang digunakan untuk evaluasi metode klasifikasi dengan memperkirakan objek yang benar atau salah. Sebuah matriks dari prediksi yang akan dibandingkan dengan kelas yang asli dari inputan atau dengan kata lain berisi informasi nilai aktual dan prediksi pada klasifikasi [5]. Berikut merupakan contoh penerapan *confusion matrix* dalam klasifikasi dua kelas yang diperlihatkan pada tabel berikut ini:

Tabel 2.6 *Confusion Matrix*

	Prediction Class 1	Prediction Class 2
Actual Class 1	<i>True Positive (TP)</i>	<i>False Positive (FP)</i>
Actual Class 2	<i>False Negative (FN)</i>	<i>True Negative (TN)</i>

Tabel di atas menjelaskan sebuah identifikasi dari suatu prediksi, dimana [19]:

- ❖ *True Positive* (TP) merupakan jumlah *record* positif yang diklasifikasikan sebagai positif.
- ❖ *False Positive* (FP) merupakan jumlah *record* negatif yang diklasifikasikan sebagai positif.
- ❖ *False Negative* (FN) merupakan jumlah *record* positif yang diklasifikasikan sebagai negatif.
- ❖ *True Negative* (TN) merupakan jumlah *record* negatif yang diklasifikasikan sebagai negatif.

Berikut merupakan rumus perhitungan pada confusion matrix:

$$Akurasi = \frac{(TP+TN)}{(TP+FN+FP+TN)} * 100\% \dots\dots\dots (2.3)$$

$$Presisi = \frac{TP}{(FP+TP)} * 100\% \dots\dots\dots (2.4)$$

$$Recall = \frac{TP}{(TP+TN)} * 100\% \dots\dots\dots (2.5)$$

2.2.7 Kurva ROC (*Receiver Operating Characteristic*)

Kurva ROC merupakan sebuah grafik dua dimensi yang digunakan untuk mengevaluasi tingkat keberhasilan rata-rata dari kinerja metode klasifikasi dengan menggunakan *false positives* sebagai garis horizontal (sumbu X) dan *true positives* sebagai garis vertikal (sumbu Y). Kurva ROC bertujuan melakukan sebuah pengamatan terhadap model dalam klasifikasi yang dihasilkan melalui perhitungan *confusion matrix*, yaitu antara *False Positive Rate* dengan *True Positive Rate*. Dimana:

$$False Positive Rate (FPR) = \frac{FP}{(FP+TN)} \dots\dots\dots (2.6)$$

$$True Positive Rate (TPR) = \frac{TP}{(TP+FN)} \dots\dots\dots (2.7)$$

Kurva ROC nantinya akan menghasilkan pengujian yang baik jika garis kurva berada di atas garis *threshold*. Kurva ROC akan menghasilkan sebuah nilai yaitu *Area Under Curve* (AUC) [20] [21]. Menurut Gorunescu (2011), dalam nilai AUC klasifikasi data mining dibagi menjadi beberapa kelompok [22], yaitu:

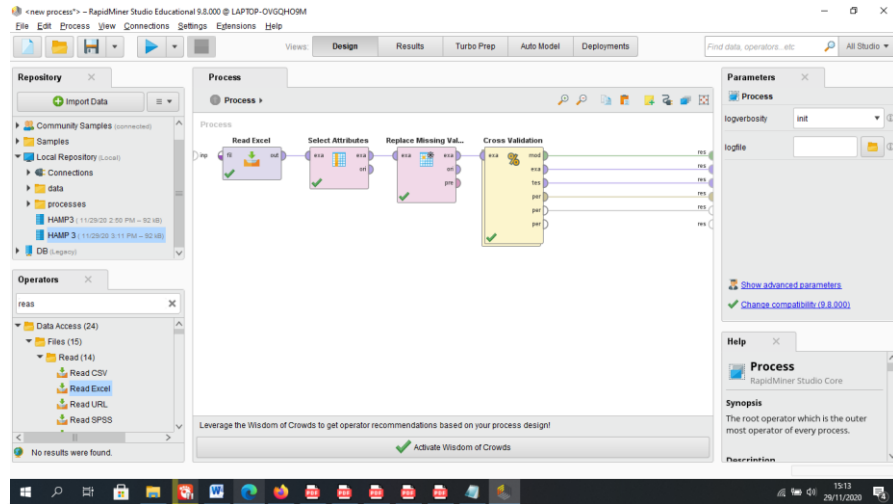
- a. 0,90 – 1,00 = Klasifikasi Sangat Baik (*Excellent Classification*)
- b. 0,80 – 0,90 = Klasifikasi Baik (*Good Classification*)
- c. 0,70 – 0,80 = Klasifikasi Cukup (*Fair Classification*)
- d. 0,60 – 0,70 = Klasifikasi Buruk (*Poor Classification*)
- e. 0,50 – 0,60 = Klasifikasi Gagal (*Failure*)

2.2.8 RapidMiner

RapidMiner merupakan sebuah perangkat lunak yang bersifat *open source*. *RapidMiner* adalah sebuah upaya penyelesaian dalam melakukan tahapan analisis terhadap beberapa keilmuan seperti data mining, *text mining*, dan memprediksi. Dalam memberikan pemahaman pada pengguna, *RapidMiner* memanfaatkan beberapa teknik seperti prediksi dan deskriptif yang nantinya mampu menghasilkan keputusan paling baik. Jumlah operator dalam *RapidMiner* berkisar 500 yang diantaranya merupakan operator untuk *input*, *output*, data *preprocessing* dan visualisasi. *RapidMiner* dapat digunakan dan bekerja pada semua sistem operasi karena menggunakan bahasa *java* [23].

Pada awalnya *RapidMiner* memiliki nama YALE (*Yet Another Learning Enviroment*), yang dikembangkan pada tahun 2001 oleh Ralf Klinkenberg, Ingo Mierswa, dan Simon Fischer di Artificial Intelligence Unit dari University of Dortmund sebagai versi pertama. *RapidMiner* didistribusikan di bawah lisensi AGPL (*GNU Affero General Public License*) versi 3. *RapidMiner* mampu menyediakan GUI (*Graphic User Interface*) untuk merancang sebuah *pipeline* analisis. GUI ini yang nantinya menghasilkan file XML (*Extensible Markup Language*) yang mendefinisikan proses analisis penerapan keinginan dan kebutuhan pengguna ke dalam data. File ini yang selanjutnya akan dibaca oleh *RapidMiner* untuk menjalankan analisis secara otomatis [24].

Dalam penelitian ini, peneliti menggunakan *RapidMiner Studio* versi 9.6. Berikut merupakan tampilan dari *RapidMiner*.

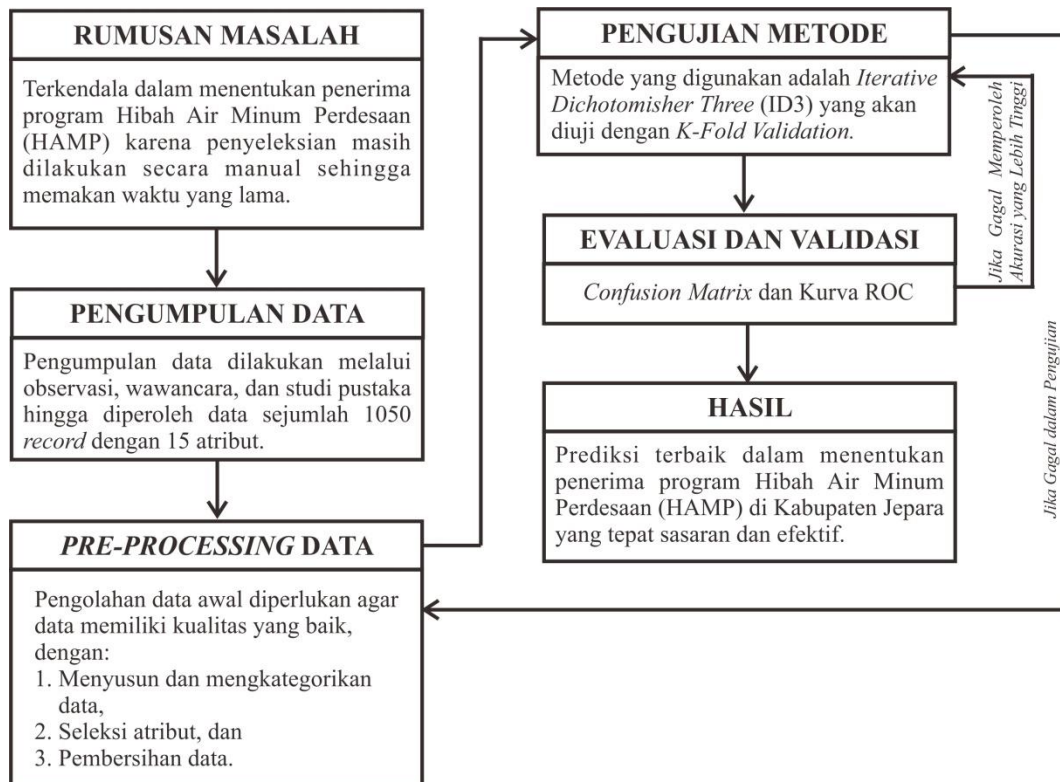


Gambar 2.4 Tampilan *RapidMiner*



2.3 Kerangka Pemikiran

Kerangka pemikiran merupakan sebuah gambaran pola pikir penelitian yang akan dilakukan. Dalam penelitian ini, peneliti membuat sebuah kerangka pemikiran yang diilustrasikan pada gambar di bawah ini.



Gambar 2.5 Kerangka Pemikiran

