BAB 2
TINJAUAN PUSTAKA

2.1 Tinjauan Studi

Sebelum menyusun tugas akhir ini dilakukan tinjauan pustaka terlebih dahulu terhadap penelitian-penelitian terkait sebagai bahan referensi. Penelitian tentang klasifikasi penjurusan siswa Sekolah Menengah Atas sebelumnya sudah pernah dilakukan dengan menggunakan algoritma C4.5 dan *Naïve Bayes,* tetapi dengan jumlah kelas yang berbeda.

Penelitian yang pertama yaitu penelitian yang dilakukan oleh David Hartanto Kamagi dan Seng Hansun tentangprediksi tingkat kelulusan mahasiswa. Metode yang digunakan adalah algoritma C4.5. Dari penelitian tersebut membuktikan bahwa algoritma C4.5 dapat diterapkan untuk memprediksi tingkat kelulusan mahasiswa dengan 4kategori/kelas yaitu drop out, lulus terlambat, lulus tepat, dan lulus cepat. IPS semester enam adalah atribut yang paling berpengaruhdalam penetuan hasil prediksi. Prediksi kelulusan mahasiswa dengan algoritma C4.5 menghasilkan presentase sebanyak 87.5% dari enam puluh *data training* dan empat puluh *data testing*. Prediksi tingkat kelulusan yang dihasilkan dari penelitian tersebut dapat membantu bagian program studi untuk mengetahui status kelulusan mahasiswa, sehingga dapat dijadikan sebagaidasar dalam pengambilan mata kuliah bagi mahasiswa untuk semester berikutnya seperti tugas akhir dan kerja praktek, sehingga memudahkan mahasiswa agar lulus tepat waktu.[2]

Penelitian yang kedua yaitu penelitian tentangpenentuan jurusan mahasiswa yang dilakukan oleh Liliana Swastina. Metode yang digunakan yaitu algoritma C4.5. Dari penelitian tersebut menyimpulkan bahwa algoritma C4.5 memprediksi lebih akurat dibandingkan dengan algoritma Naive Bayes dalam menentukan kesesuaian jurusan dan rekomendasi jurusan mahasiswa. Karena hal itu, maka dapat disimpulkan bahwa algoritma *decision tree* C4.5 akurat diterapkan untuk

menentukan kesesuaian jurusan mahasiswa dari pada algoritma *Naïve Bayes*.Tingkat keakuratan yang dihasilkan algoritma C4.5 sebesar 93,31% serta akurasi rekomendasi jurusan sebesar 82,64%, sedangkan rekomendasi jurusan dengan algoritma Nave Bayes hanya sebesar 66,36%.[6]

Penelitian yang ketiga yaitu penelitian yang dilakukan oleh Maghriza Fakri Adillatentangklasifikasi penjurusan Sekolah Menengah Atas (SMA). Metode yang digunakan dalam penelitian tersebutyaitu algoritma *Naïve Bayes*, sedangkan atribut yang digunakan berjumlah delapan yaitu nilai Ujian Nasional IPA dan Matematika Sekolah Menengah Pertama (SMP), nilai raport IPA dan Matematika Sekolah Menengah Pertama (SMP) selama 5 semester, nilai kualitas, nilai IQ, dan minat jurusan. Dari hasil penelitian tersebut menyimpulkan bahwa klasifikasi data siswa baru SMA 1 Kajen tahun ajaran 2015/2016 dapat dilakukan dengan menggunakan teknik *data mining*, yaitu dengan metode klasifikasi menggunakan algoritma Naive Bayes Classifier.Akurasi yang dihasilkan dari metode algoritma *Naïve Bayes Classifier* menggunakan matlab adalah sebesar 86,1842 % dengan *error rate* sebesar 13.8158 %.[7]

Penelitian yang keempat dilakukan oleh Obbie Kristantotentang penentuan jurusan Siswa SMA. Metode yang digunakan yaitu algoritma *decision tree* ID3 atau *Iterative Dichotomiser* 3. Peneletian tersebut menghasilkan sebuah aplikasi yang telah berhasil dirancang sesuai dengan kebutuhan.Aplikasi tersebut dapat berjalan sebagai media pembantu dalam proses penentuan jurusan pada SMAN6 Semarang.Tingkat akurasi dari aplikasi tersebut dapat diketahui dengan cara membandingkannyadengan data dari guru BP. Penelitian menyimpulkan bahwaterdapat kasus yang tidak sesuai atau meleset yaitu sebanyak4 kasus, sedangkanyangberhasil sebanyak 16 kasus dari 20 data uji371 dataset, dari hasil tersebut didapat akurasi sebesar 80%.[8]

Tabel 2.1 Penelitian Terkait

No	Nama Peneliti dan Tahun	Masalah	Metode	Hasil
1.	David Hartanto Kamagi dan Seng Hansun, 2014	Prediksi tingkat kelulusan mahasiswa	Algoritma C4.5	Implementasi algoritma C4.5 dapat memprediksi kelulusan mahasiswa. Dari enam puluh data training dan empat puluh data testing diperoleh prosentase sebesar 87.5%.
2.	Liliana Swastina, 2013	Klasifikasi penentuan jurusan mahasiswa	Algoritma C4.5	Algoritma C4.5 terbukti akurat diaplikasikandalam penentuan kesesuaian jurusan mahasiswa dari pada algoritma *Naïve Bayes.* Tingkat keakuratan yang dihasilkan

No	Nama Peneliti dan Tahun	Masalah	Metode	Hasil
				algoritma C4.5 sebesar 93,31% serta akurasi rekomendasi jurusan sebesar 82,64%, sedangkan rekomendasi jurusan dengan algoritma Nave Bayes hanya sebesar 66,36%.
3.	Maghriza Fakri Adilla, 2016	Klasifikasi penjurusan siswa SMA	Algoritma *Naive Bayes Classifier*	Akurasi yang dihasilkan dari metode algoritma *Naïve Bayes Classifier* menggunakan matlab adalah sebesar 86,1842 % dengan *error rate* sebesar 13.8158 %.
4.	Obbie Kristanto, 2014	Klasifikasi penjurusan siswa SMA	Algoritma Klasifikasi *Data Mining* ID3 atau	Penelitian menyimpulkan bahwa terdapat kasus yang tidak sesuai atau

No	Nama Peneliti dan Tahun	Masalah	Metode	Hasil
			Iterative Dichotomiser 3	meleset yaitu sebanyak 4 kasus, sedangkan yang berhasil sebanyak 16 kasus dari 20 data uji 371 dataset, dari hasil tersebut didapat akurasi sebesar 80%.

2.2 Landasan Teori

2.2.1 Penjurusan Siswa SMA

Penjurusan siswa Sekolah Menengah Atas (SMA) yang berlakusekarang didasarkan pada kurikulum 2013. Dalam pelaksanaannya, pada kurikulum 2013 terdapat perbedaan-perbedaan dengan kurikulum sebelumnya, diantaranya adalah dalam proses penjurusan. Dalam kurikulum 2013penentuan jurusan siswa SMA dilaksanakan pada kelas X[1]. Faktor-faktor yang dijadikan dasar untuk menentukan jurusan siswa pada SMA Negeri 2 Pemalang yaitu nilaiUjian Nasional Bahasa Indonesia, nilai Ujian Nasional Bahasa Inggris, nilai Ujian Nasional Matematika, dan nilai Ujian Nasional IPA Sekolah Menengah Pertama (SMP), nilai rata-rata raport Bahasa Indonesia, Bahasa Inggris, Matematika, IPA, dan IPS Sekolah Menengah Pertama (SMP) selama 5 semester, serta minat siswa [3].

2.2.2 *Data Mining*

Data mining atau dalam istilah lain disebut dengan *Knowlege Discovery in Database* (KDD) merupakan suatu kegiatan yang berkaitan dengan pengumpulan data historis guna menemukan keteraturan, serta pola keterkaitan dalam sebuah dataset yang memiliki kapasitas sangat besar [12]. Menurut Durairaj dan Vijitha dalam penelitiannya yang berjudul "*Educational Data mining for Prediction of Student Performance Using Clustering Algorithms*"[9] menyatakan bahwa data mining adalah sebuah metodologi analisa data yang dipergunakan untuk pengidentifikasian pola-pola yang tersembunyi dengan menggunakan teknik pada sebuah metodologi analisa data untuk memperoleh pola-pola yang unik serta menarik dalam sebuah dataset dengan record yang banyak.

Data mining berdasarkan cara menganalisa dataset yang akan diteliti dapat dibagikan menjadi lima metode dalam menarik kesimpulan tentang pola data yang tersembunyi [5]. Metode-metode tersebut adalah sebagai berikut:

1. Estimasi

   Sebuah teknik untuk memperoleh pola tersembunyi pada sebuah*dataset* dengan caramelihat target variabel kategori. Teknik ini hampir mirip dengan teknik klasifikasi namun teknik ini cendrung digunakakn untuk tipe data numerik dan memiliki label. Metode ini menggunakan pembelajaran supervised learning. Beberapa algoritma yang dapat digunakan dalam teknik ini adalah: *Linear Regression, Support Vector Machine, dan Neural Network*.

2. Klasifikasi

   Fungsinya untuk mengelompokan pola data yang sama berdasarkan atribut-atribut yang dimiliki. Pada teknik ini data yang diolah cenderung menggunakan tipe data nominal namun tidak menutup kemungkinan untuk pengolahan dengan tipe data numerik. Teknik ini juga bersifat *supervised learning* artinya dalam penemuan pola yang baru memerlukan guru atau label target. Beberapa algoritma yang dapat digunakan dalam teknik ini adalah :

Naive Bayes, C4.5, ID3, *K-Nearest Neighbor*, *Linear Discriminant Analysis*, dan CART.

3. Prediksi

   Teknik prediksi memilki kesaman dengan teknik estimasi dan klasifikasi dalam menganalisa kumpulan data, namun tipe data yang digunakan adalah numerik baik dalam variabel maupun label. Ciri khas dalam metode ini adalah salah satu variabel yang digunakan memiliki tipe data time series. Teknik ini juga termasuk dalam kategori supervised learning.Algoritma yang dapat digunakan dalam teknik prediksi diantaranya adalah : *Linear Regression, Support Vector Machine, dan Neural Network.*

4. Klastering

   Metode ini sering disebut juga sebagai metode segmentation, metode ini berfungsi untuk mengidentifikasi kelompok alami dari kasus yang didasarkan pada satu kelompok atribut, dengan cara mengelompokkan data-data yang memiliki kemiripan pada setiap atribut-atributnya. Ciri khas dari teknik ini adalah dataset yang digunakan tidak memiliki lebel target. Metode pembelajaran dalam teknik klastering ini adalah unsupervised learning (tidak membutuhkan guru dalam menemukan sebuah pola tertentu). Algoritma yang sering digunakan dalam teknik ini adalah: K-Means, K-Medoids, Fuzzy C-Means, dan Self-Organizing Map (SOM).

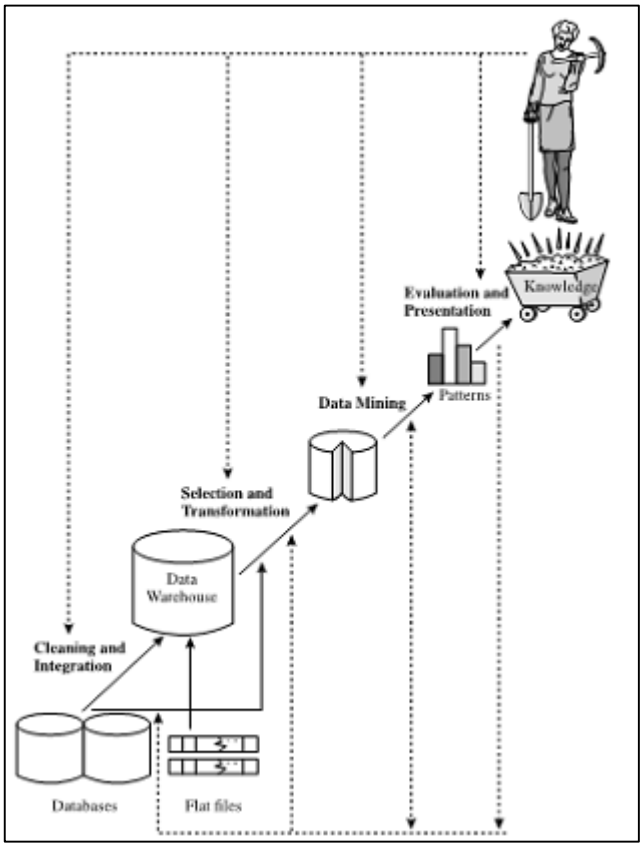
5. Asosiasi

   Teknik asosiasi atau yang sering dikenal sebagai *association rules* berfungsi untukmenemukan relasi diantara item-item data serta menemukan sejumlah atribut yang muncul secara bersamaan. Teknik ini menggunakan pembelajaran yang paling berbeda sendiri diantara empat teknik diatas, pembelajaran yang digunakan adalah association learning (menemukan pola item yang muncul pada transaksi yang sama). Algoritma yang sering digunakan dalam teknik ini adalah: *Apriori* , dan *FP-Growth*.

2.2.2.1 Tahapan dalam *Data Mining*

Dalam data mining terbagi beberapa tahap, Tahapan tersebut dilakukan sebagai suatu rangkaian proses yang bersifat interaktif dimana knowledge base terlibat oleh pengguna[10].

Berikut adalah tahapan dalam data mining :



Gambar 2.1 : Tahapan Data Mining[10]

Keterangan:

1. Pembersihan Data

   Pembersihan data dilakukan untuk menghilangkan noise atau missing value. Sering kali data yang diperoleh dari hasil penelitian, terdapat data yang tidak lengkap diantaranya data yang hilang, atau salah ketik dalam penulisan. Beratribut tidak relevan dengan hipotesa data mining yang dimiliki. Data yang tidak relevan akan dibuang dan tidak digunakan dalam proses.

2. Integrasi Data

   Penggabungan data dari database ke dalam satu database yang baru. Integrasi data dilakukan dengan cermat untuk mengidentifikasikan beberapa entitas agar tidak menyimpang, beberapa entitas diantaranya atribut nama, jenis produk, dan sebagainya. Melakukan integrasi data diperlukan transformasi dan pemberisihan data dikarenakan sering kali dari dua database cara penulisannya berbeda.

3. Seleksi Data

   Dalam database tidak semua data akan dipakai, hanya data yang sesuai yang akan diambil sebagai bahan analisa. Sebagai contoh ada kasus yang menyeleksi kecenderungan orang saat membeli dalam kasus market, cukup id pelanggan yang diambil dan tidak perlu dengan nama pelanggannya.

4. Transformasi Data

   Sebelum diaplikasikan, data mining memerlukan format data, sebelumnya data akan dirubah dan digabungkan sesuai dengan fotmat data mining. Sebagai contoh analisis, asosiasi, dan clustering hanya mampu menerima input data kategorikal yang berupa data numerik, data akan dibagi dalam bentuk interval. Proses ini disebut transformasi data.

5. Proses Mining

   Dalam Tahap ini metode yang telah diterapkan bertujuan untuk menemukan informasi yang berharga dari data yang tersembunyi.

6. Evaluasi Pola

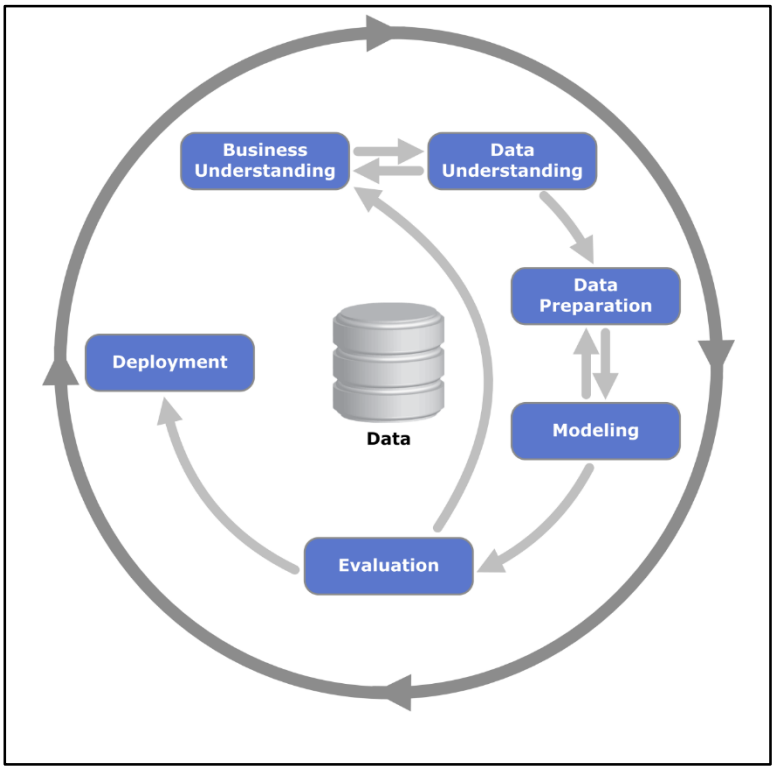
   Untuk tahapan ini data mining menghasilkan prediksi guna evaluasi untuk menilai apakah hipotesa tersebut terapai. Masih ada cara alternative jika hasil yang didapatkan tidak sesuai hipotesa.

7. Evaluasi Pola

   Yang terakhir dari tahap data mining adalah bagaimana memformulasikan keputusan dari analisis. Tentang data mining setidaknya melibatkan orang-orang yang paham agar semua orang yang terlibat dalam persentasi data memahaminya.

2.2.2.2 *Cross-Indstry Standard Process for Data Mining*(CRISP-DM)

CRISP-DM (*Cross-Industry Standard Process for Data Mining*) adalah konsorsium perusahaan yang berdiri sejak tahun 1996 oleh Komisi Eropa yang ditetapkan sebagai acuan standar data mining untuk seluruh sector industry. Gambar 2.2 menggambarkan pola siklus hidup dalam CRISP-DM dengan penjelasan *data mining.*



Gambar 2.2 : Gambar Siklus Hidup CRISP-DM

Ada enam tahapan siklus hidup dalam data mining[11][12] (Chapman, 2000) :

1. *Business Understanding*

   Tahap pertama adalah melakukan pemahaman dari sudut pandang bisnis untuk suatu kebutuhan dan masuk kedalam masalah definisi data mining sehingga tujuan yang akan dicapai telah ditentukan oleh rencana dan strategi.

2. *Data Understanding*

   Tahap kedua adalah melakukan pengumpulan data kemudian dilakukannya proses agar pemahaman tentang data dapat diperoleh. Dapat mengidentifikasi

suatu masalah kualitas data untuk hipotesa sebagai informasi yang tersembunyi.

3. *Data Preparation*

   Tahap ketiga adalah proses dari data mentah untuk pembentukan dataset akhir dan dapat diulang berkali-kali. Proses pembersihan yang mencakup tabel, *record*, dan atribut data yang akan dijadikan untuk tahap pemodelan.

4. *Modeling*

   Tahap keempat adalah melakukan penerapan untuk beberapa teknik pemodelan termasuk dengan parameternya untuk menghasilkan nilai yang optimal.

5. *Evaluation*

   Tahap kelima adalah menganalisa dan mengevaluasi data yang sudah ditetapkan pada fase awal untuk mendapatkan kualitas yang baik dengan menerapkan sudut pandang model yang sudah terbentuk.

6. *Deployment*

   Tahap keenam adalah informasi yang didapat akan diatur dan dipresentasikan sehingga seseorang dapat menggunakannya. Tahap iniberupa pengulangan proses implementasikan *data mining* dalam perusahaan yang melibatkan konsumen, agar para konsumen dapat memahami dengan menggunakan model yang sudah dibuat.

2.2.3 Konsep Klasifikasi

Berdasarkan tugas yang dilakukan, data mining dibagi beberapa kelompok, yaitu : Deskripsi, Estimasi, Prediksi, Klasifikasi, Clustering, dan Asosiasi (Larose, 2005). Klasifikasi adalah salah satu algoritma data mining, menggunakan data dengan target yang berupa nilai nominal. Klasifikasi didasarkan pada empat komponen mendasar (Gorunescu), yaitu:

a. Kelas (*Class*)

   Merepresentasikan label yang merupakan dari variabel kategorikal pada objek setelah klasifikasinya. Contohnya yaitu adanya kelas diagnose penyakit amnemia, kelas bencana alam, dll.

b. Prediktor (*Predictor*)

   Merepresentasikan atribut data yang akan diklasifikasikan. Sebagai contoh : konsumsi narkoba, konsumsi alkohol, tekanan darah, status kekeluargaan, kecepatan arah mata angin, pergantian musim, dll.

c. Pelatihan dataset (*Training dataset*)

   Berdasarkan prediktor yang tersedia, data yang digunakan terkait dengan nilai-nilai dari kedua komponen sebelumnya, guna melatih model dalam mengenali kelas yang sesuai. Contohnya adalah database yang terdapat gambar untuk monitoring teleskopik dan basis data pada penelitian gempa.

d. Dataset Pengujian (*Testing Dataset*)

   Data yang telah diklasifikasikan oleh model sehingga akurasi klasifikasi dapat dievaluasi.

2.2.4 Pohon Keputusan (*Decision Tree*)

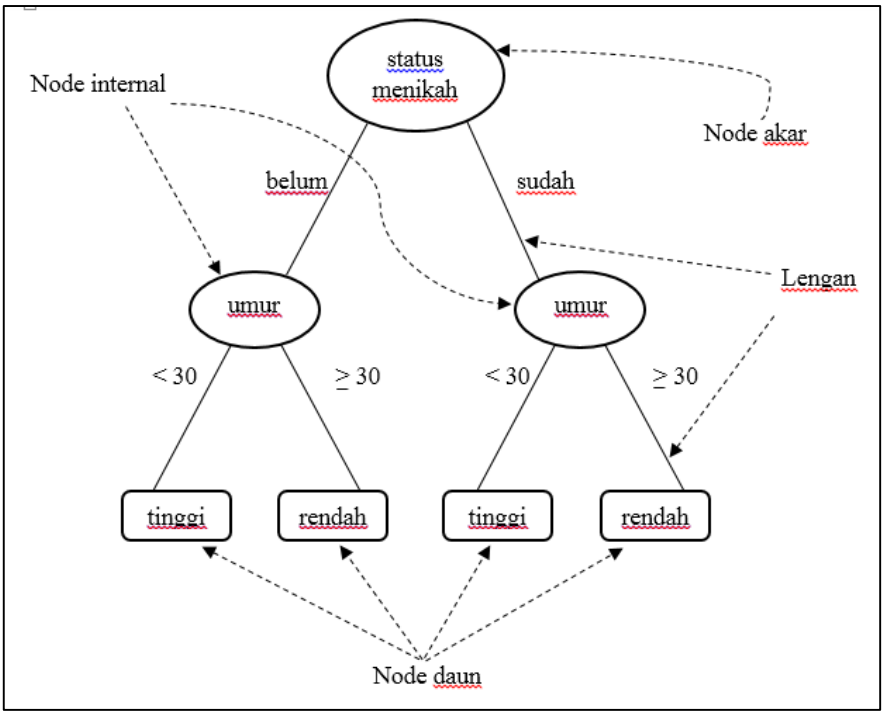
Pohon keputusan yaitu pohon dalam analisis pemecahan masalah pengambilan keputusan mengenai pemetaan mengenai alternatif-alternatif pemecahan masalah yang dapat diambil dari masalah tersebut. Pohon tersebut juga memperlihatkan faktor-faktor kemungkinan/probablitas yang akan mempengaruhi alternatif-alternatf keputusan tersebut, disertai dengan estimasi hasil akhir yang akan didapat bila kita mengambil alternatif keputusan tersebut.

Decision tree menggunakan struktur hierarki untuk pembelajaran *supervised*. Proses dari decision tree dimulai dari *root node* hingga *leaf node* yang dilakukan secara rekursif. Di mana setiap percabangan menyatakan suatu kondisi yang harus dipenuhi dan pada setiap ujung pohon menyatakan kelas dari suatu data.

Proses dalam pohon keputusan yaitu mengubah bentuk data (tabel) menjadi model pohon (*tree*) kemudian mengubah model pohon tersebut menjadi aturan (*rule*).

Metode pohon keputusan digunakan untuk memperkirakan nilai diskret dari fungsi target yang mana fungsi pembelajaran direpresentasikan oleh sebuah pohon keputusan (*decision tree*). Pohon keputusan terdiri dari himpunan *IF…THEN*. Setiap *path* dalam tree dihubungkan dengan sebuah aturan, dimana premis terdiri atas sekumpulan node-node yang ditemui dan kesimpullannya dari aturan atas kelas yang terhubung dengan *leaf node* dari path.

Berikut ini adalah contoh dari *decision tree* :



Gambar 2.3 Contoh D*ecision Tree*[13]

Karakteristik *decision tree*seperti pada gambar 2.1, dibentuk oleh sejumlah elemen antara lain (Tan, 2006)[13] :

a. Node akar, node ini tidak mempunyai lengan masukan dan memiliki nol atau lebih lengan keluaran. Node ini terletak pada bagian atas pohon.
b. Node internal, node yang memiliki tepat satu lengan masukan dan dua atau lebih lengan keluaran. Node ini merupakan node percabangan.

c. Lengan, setiap cabang menyatakan nilai hasil pengujian di node selain node daun.
d. Node daun, node yang memiliki tepat satu lengan masukan dan tidak mempunyai lengan keluaran. Node ini yang menyatakan label kelas.

Untuk membentuk pohon keputusan, terdapat langkah-langkah sebagai berikut :

a. Membuat root dengan memilih atribut.
b. Setiap nilai dibuat bercabang.
c. Cabang dibagi kedalam kelas.
d. Mengulang kembali disetiap cabang sehingga semua kasus memiliki kelas yang sama.

Pemilihan atribut sebagai *root* dari suatu atribut berdasarkan nilai tertinggi dari *gian*. Sementara itu, jika ingin mendapat nilai *gain* tertinggi kita harus menghitung nilai *entropy* dari semua nilai didalam atribut. *Entropy* berperan sebagai parameter untuk mengukur varian dari data sampel. Setelah nilai *entropy* dalam data sampel diketahui, atribut yang paling berpengaruh akan menjadi pengukur dalam pengklasifikasian data, ukuran ini disebut sebagai *Information gain*.

Terdapat beberapa algoritma yang dapat digunakan dalam pembentukan pohon keputusan, antara lain ID3, CART, dan C4.5 (Larose, 2005)[5].

2.2.5 Algoritma C4.5

Decision Tree (Pohon Keputusan) merupakan metode klasifikasi yang berguna untuk memprediksi dengan cara menggunakan struktur pohon. Decision tree dapat digunakan sebagaimana mendapatkan informasi guna pengambilan keputusan. Konsep decision tree yaitu dengan cara mengubah data menjadi pohon keputusan serta aturan-aturan keputusan[2].

Algoritma C4.5 mengkonstruksi pohon keputusan dari data pelatihan, yang berupa kasus-kasus atau *record* (tupel) dalam basisdata. Setiap kasus berisikan nilai dari atribut-atribut untuk sebuah kelas. Setiap atribut dapat berisi data diskret atau kontinyu (numerik). C4.5 juga menangani kasus yang tidak memiliki nilai untuk

sebuah atau lebih atribut. Akan tetapi, atribut kelas hanya bertipe diskret dan tidak boleh kosong.

Ada tiga prinsip kerja algoritma C4.5 pada tahap belajar dari data, yaitu sebgai berikut :

1. Pembuatan Pohon Keputusan

   Obyektif dari algoritma pohon keputusan adalah mengkonstruksi struktur data pohon (dinamakan pohon keputusan) yang dapat digunakan untuk memprediksi kelas dari sebuah kasus atau *record* baru yang belum memiliki kelas.Algoritma ini memilih pemecahan kasus-kasus yang terbaik dengan menghitung dan membandingkan *gain ratio*, kemudian pada node-node yang terbentuk di level berikutnya. Demikian seterusnya sampai terbentuk daun-daun.

2. Pemangkasan Pohon Keputusan dan Evaluasi (Opsional)

   Karena pohon yang dikonstruksi dapat berukuran besar dan tidak mudah dibaca, C4.5 dapat menyederhanakan pohon dengan melakukan pemangkasan berdasarkan nilai tingkat kepercayaan (*confidence level*). Selain untuk pengurangan ukuran pohon, pemangkasan juga bertujuan untuk mengurangi tingkat kesalahan prediksi pada kasus (rekord) baru.

3. Pembuatan Aturan-Aturan dari Pohon Keputusan (Opsional)

   Aturan-aturan dalam bentuk if-then diturunkan dari pohon keputusan dengan melakukan penelusuran dari akar sampai ke daun. Setiap node dan syarat pencabangannya akan diberikan di if, sedangkan nilai pada daun akan menjadi ditulis di then. Setelah semua aturan dibuat, maka aturan akan disederhanakan (digabung atau diperumum).

2.2.5.1 *Entropy*

Dalam teori informasi, entropi mengukur ketidakpastian antar variabel acak dalam file data. Claude E. Shannon telah mengembangkan gagasan tentang entropi dari variabel acak. Entropi dan informasi terkait menyediakan perilaku jangka panjang

dari proses acak yang sangat berguna untuk menganalisis data. Perilaku dalam proses acak juga merupakan faktor kunci untuk mengembangkan pengkodean untuk teori informasi. Entropi merupakan pengukuran ketidakpastian rata-rata kumpulan data ketika kita tidak tahu hasil dari sumber informasi. Itu berarti bahwa seberapa banyak pengukuran informasi yang kita tidak punya. Ini juga menunjukkan jumlah rata-rata informasi yang kami akan menerima dari hasil sumber informasi. Untuk mendapatkan nilai gain ratio dalam pembentukan pohon keputusan, perlu menghitung dulu nilai informasi dalam satuan bits dari suatu kumpulan objek

Bentuk perhitungan untuk entropi adalah sebagai berikut :

$$Entropy(X) = \sum_{j=1}^{k} -pj \times \log 2\, pj \qquad (1)$$

Keterangan :

X : Himpunan Kasus

k : jumlah partisi X

pj : Proporsi Xj terhadap X

2.2.5.2 *Gain*

Pada kontruksi pohon C4.5, di setiap simpul pohon, atribut dengan nilai *gain* tertinggi dipilih sebagai atribut untuk simpul. Rumus dari *Gain* adalah sebagai berikut :

$$Gain(X, A) = Entropy(X) - \sum_{j=1}^{k} \frac{|X_i|}{|X|} * Entropy(X_i) \qquad (2)$$

Keterangan :

X : Himpunan Kasus

A : atribut

Xi : Proporsi atribut ke X terhadap jumlah kasus

2.2.5.3 Studi Kasus

Berikut ini adalah tahapan Decision Tree menggunakan algoritma C4.5 dengan studi kasus yaitu : Klasifikasi kelayakan penerima beasiswa dengan menganalisa

data penerima beasiswa dengan atribut-atributnya adalah IPK, Piagam Penghargaan dan Penghasilan Orang Tua. Disetiap atribut memiliki nilai, dan kelasnya ada pada kolom Kelayakan Beasiswa dengan kelas "Layak" dan kelas "Tidak Layak". Dataset terdiri dari 20 kasus dengan 5 kasus Layak dan 15 kasus Tidak Layak pada kolom Kelayakan Beasiswa.

Tabel 2.2 Kasus Klasifikasi Penerima Beasiswa

Kategori IPK	Piagam Penghargaan	Penghasilan Orang Tua	Kelayakan Beasiswa
Rendah	Tidak Ada	Tinggi	Tidak Layak
Rendah	Tidak Ada	Rendah	Tidak Layak
Tinggi	Tidak Ada	Tinggi	Layak
Cukup	Tidak Ada	Tinggi	Layak
Cukup	Ada	Tinggi	Layak
Cukup	Ada	Rendah	Layak
Tinggi	Ada	Rendah	Layak
Rendah	Tidak Ada	Tinggi	Tidak Layak
Rendah	Ada	Tinggi	Layak
Cukup	Ada	Tinggi	Layak
Rendah	Ada	Rendah	Layak
Tinggi	Tidak Ada	Rendah	Layak
Tinggi	Ada	Tinggi	Layak
Cukup	Tidak Ada	Rendah	Tidak Layak
Cukup	Tidak Ada	Tinggi	Layak
Tinggi	Tidak Ada	Rendah	Layak

Kategori IPK	Piagam Penghargaan	Penghasilan Orang Tua	Kelayakan Beasiswa
Rendah	Tidak Ada	Rendah	Tidak Layak
Cukup	Ada	Tinggi	Layak
Tinggi	Ada	Tinggi	Layak
Rendah	Ada	Tinggi	Layak

1. Menentukan akar dari pohon, node awal diambil dari atribut yang akan dipilih, menghitung dan kemudian memilih nilai *gain* yang tertinggi dari masing-masing atribut. Sebelum menghitung gain, harus dihitung dahulu nilai *entropy* dari setiap tupel berikut :

Tabel 2.3 Jumlah Kasus Tiap Atribut

Simpul		Jumlah Kasus	Tidak Layak	Layak
Jumlah Kasus		20	5	15
IPK				
	Tinggi	6	0	6
	Cukup	7	1	6
	Rendah	7	4	3
Piagam Penghargaan				
	Ada	10	0	10
	Tidak Ada	10	5	5
Penghasilan Orang Tua				

Simpul		Jumlah Kasus	Tidak Layak	Layak
	Tinggi	8	3	5
	Rendah	12	2	10

2. Lakukan perhitungan dengan metode *information Gain* :

Untuk menghitung *Gain* diwajibkan untuk mencari *Entropy* dari setiap tupel atribut masing-masing, contohnya sebagai berikut :

Entropy(Jumlah kasus)

$= (-(\frac{5}{20}) * \log_2\frac{5}{20}) + (-(\frac{15}{20}) * \log_2\frac{15}{20})$

$= 0.918295834$

Entropy(IPK-Tinggi)

$= (-(\frac{0}{6}) * \log_2\frac{0}{6}) + (-(\frac{6}{6}) * \log_2\frac{6}{6})$

$= 0$

Entropy(IPK-Cukup)

$= (-(\frac{1}{7}) * \log_2\frac{1}{7}) + (-(\frac{6}{7}) * \log_2\frac{6}{7})$

$= 0.591672779$

Entropy(IPK-Rendah)

$= (-(\frac{4}{7}) * \log_2\frac{4}{7}) + (-(\frac{3}{7}) * \log_2\frac{3}{7})$

$= 0.985228136$

Entropy(Piagam-Ada)

$= (-(\frac{0}{10}) * \log_2\frac{0}{10}) + (-(\frac{10}{10}) * \log_2\frac{10}{10})$

$= 0$

Entropy(Piagam-Tidak Ada)

$= (-(\frac{5}{10}) * \log_2\frac{5}{10}) + (-(\frac{5}{10}) * \log_2\frac{5}{10})$

$= 1$

Entropy(Penghasilan Orang Tua-Tinggi)

$= (-(\frac{3}{8}) * \log_2\frac{3}{8}) + (-(\frac{5}{8}) * \log_2\frac{5}{8})$

= 0.9544344

Entropy(Penghasilan Orang Tua-Rendah)

= (- ($\frac{2}{12}$) * $\log_2\frac{2}{12}$) + (- ($\frac{10}{12}$) * $\log_2\frac{10}{12}$)

= 0.650022

Setelah itu menghitung Gain untuk setiap atribut :

Gain(IPK)

= 0.918296 – (($\frac{6}{20}$*0)+($\frac{7}{20}$*0.591673) +($\frac{7}{20}$*0.985228))

= 0.918296 – (0.55191532)

= 0.366380514

Gain(Piagam Penghargaan)

= 0.918296 – (($\frac{10}{20}$*0)+($\frac{10}{20}$*1))

= 0.918296 – (0.5)

= 0.418295834

Gain(Penghasilan Orang Tua)

= 0.918296 – (($\frac{8}{20}$*0.954434003)+($\frac{12}{20}$*0.650022422))

= 0.918296 – (0.771787054)

= 0.14650878

3. Partisi pohon keputusan ini akan berhenti saat proses saat atribut didalam tupel tidak ada yang dipartisi lagi dan tidak ada didalam cabang yang kosong.

Menghitung *Entropy* dan *Gain* selengkapnya akan ditampilkan pada tabel seperti berikut :

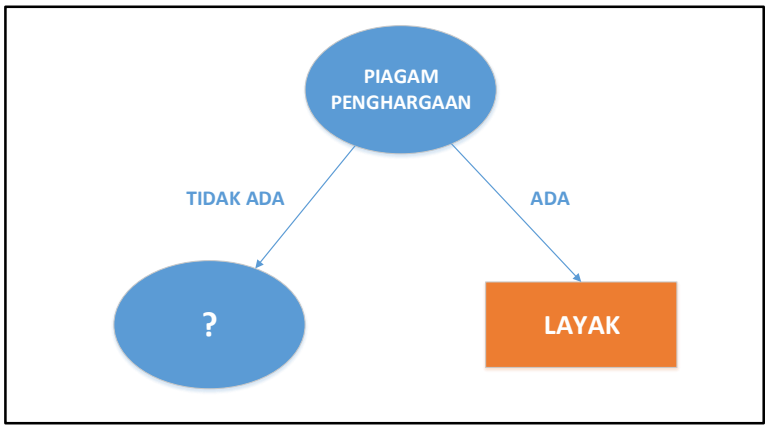
Tabel 2.4 Hasil Perhitungan Gain dan Entropy

		Jumlah kasus	**Tidak Layak**	**Layak**	**Entropy**	**Gain**
Total		20	5	15	0.918296	
IPK						0.366380514

		Jumlah kasus	Tidak Layak	Layak	Entropy	Gain
	Tinggi	6	0	6	0	
	Cukup	7	1	6	0.591672779	
	Rendah	7	4	3	0.985228	
Piagam Penghargaan						0.418295834
	Ada	10	0	10	0	
	Tidak Ada	10	5	5	1	
Penghasilan Keluarga						0.14650878
	Tinggi	8	3	5	0.954434003	
	Rendah	12	2	10	0.650022422	

Tabel 2.5 menghasilkan nilai gian tertinggi adalah Piagam Penghargaan yaitu 0.418295834. Maka dari itu atribut Piagam Penghargaan bisa menjadi node akar. Terdapat 2 variabel dari atribut Piagam Penghargaan, diantaranya Ada dan Tidak Ada. Nilai variabel Piagam Penghargaan mengklasifikasikan sebuah kasus menjadi 1 yaitu keputusannya Layak untuk variabel "Ada" karena dari 10 kasus dan semua mempunyai jawaban yang sama Layak (Sum(Total)/Sum(Layak)=10/10=1), selanjutnya nilai variabel tidak perlu diperhitungkan kembali.

Hasil ini sementara bisa digambarkan bentuk pohon keputusan seperti pada Gambar 2.5.



Gambar 2.4 Perhitungan Node 1 Pada Pohon Keputusan

Kemudian untuk menentukan Node 1.1 hitung jumlah kasus untuk Piagam Penghargaan dengan atribut "Tidak Ada" untuk keputusan Layak dan Tidak Layak, kemudian tentukan Entropy dari semua kasus. Untuk mempermudah, Tabel 2.3 difilter dengan mengambil data dari variabel Piagam Penghargaan dengan atribut Tidak Ada, sehingga jadilah tabel seperti berikut :

Tabel 2.5 Data Variabel Piagam Penghargaan - Tidak Ada

Kategori IPK	Piagam Penghargaan	Penghasilan Orang Tua	Kelayakan Beasiswa
Rendah	Tidak Ada	Tinggi	Tidak Layak
Rendah	Tidak Ada	Rendah	Tidak Layak
Tinggi	Tidak Ada	Tinggi	Layak
Cukup	Tidak Ada	Tinggi	Layak
Rendah	Tidak Ada	Tinggi	Tidak Layak
Tinggi	Tidak Ada	Rendah	Layak
Cukup	Tidak Ada	Rendah	Tidak Layak
Cukup	Tidak Ada	Tinggi	Layak
Tinggi	Tidak Ada	Rendah	Layak

Kategori IPK	Piagam Penghargaan	Penghasilan Orang Tua	Kelayakan Beasiswa
Rendah	Tidak Ada	Rendah	Tidak Layak

Selanjutnya melakukan penghitungan Gain pada masing-masing atribut, sebelum menghitung Gain, lakukan perhitungan Entropy dari setiap atribut. Hasil perhitungannya sebagai berikut :

Entropy(Jumlah Kasus)

$= (- (\frac{5}{10}) * \log_2\frac{5}{10}) + (- (\frac{5}{10}) * \log_2\frac{5}{10})$

$= 1$

Entropy(IPK-Tinggi)

$= (- (\frac{0}{3}) * \log_2\frac{0}{3}) + (- (\frac{3}{3}) * \log_2\frac{3}{3})$

$= 0$

Entropy(IPK-Cukup)

$= (- (\frac{1}{3}) * \log_2\frac{1}{3}) + (- (\frac{2}{3}) * \log_2\frac{2}{3})$

$= 0.918295834$

Entropy(IPK-Rendah)

$= (- (\frac{0}{4}) * \log_2\frac{0}{4}) + (- (\frac{4}{4}) * \log_2\frac{4}{4})$

$= 0$

Entropy(Penghasilan Orang Tua-Tinggi)

$= (- (\frac{3}{5}) * \log_2\frac{3}{5}) + (- (\frac{2}{5}) * \log_2\frac{2}{5})$

$= 0.970950594$

Entropy(Penghasilan Orang Tua-Rendah)

$= (- (\frac{2}{5}) * \log_2\frac{2}{5}) + (- (\frac{3}{5}) * \log_2\frac{3}{5})$

$= 0.970950594$

Setelah itu menghitung Gain untuk semua atribut diantaranya :

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

Gain(Piagam Penghargaan – Tidak Ada & IPK)

= 1–(($\frac{3}{10}$*0)+($\frac{3}{10}$*0.91829)+($\frac{4}{10}$* 0))

= 1 – (0.27548875)

= 0.72451125

Gain(Piagam Penghargaan – Tidak Ada & Penghasilan Orang Tua)

= 1 – (($\frac{5}{10}$*0.970950594)+($\frac{5}{10}$*0.970950594))

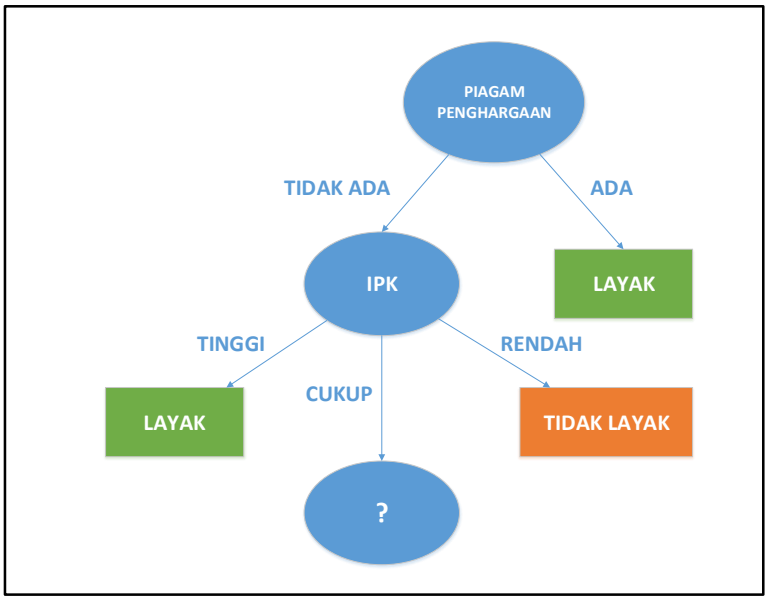
= 1 – (0.970950594)

= 0.029049406

Tabel 2.6 Proses Penghitungan Node 1.1

		Jumlah kasus	**STROKE**	**NON-STROKE**	**Entropy**	**Gain**
Piagam Penghargaan – Tidak Ada		10	5	5	1	
IPK						0.724511
	Tinggi	3	0	3	0	
	Cukup	3	1	2	0.918296	
	rendah	4	4	0	0	
Penghasilan Orang Tua						0.029049406
	Tinggi	5	3	2	0.970950594	
	Rendah	5	2	3	0.970950594	

Pada tabel 2.6 menghasilkan atribut dengan Gain tertinggi yaitu IPK sebesar 0.724511. Maka atribut IPK bsa menjadi node cabang atribut Piagam Penghargaan – Tidak Ada. Ada 3 variabel dari IPK yaitu Tinggi, Cukup dan Rendah. Dengan ini nilai variabel Tinggi dan Rendah sudah mengklasifikasikan kasus menjadi 1 yaitu keputusanya Layak untuk IPK - Tinggi dan Tidak Layak untuk IPK – Rendah sehingga dijadikan daun atau *leaf*, sedangkan untuk variabel Cukup masih perlu dianalisis kembali untuk menentukan daun atau *leaf*. Sehingga pohon keputusan (Decision Tree) sementara dari node-1.1 seperti gambar dibawah ini :



Gambar 2.5 Hasil Perhitungan Node 1.1 Pada Pohon Keputusan

Untuk menentukan daun atau *leaf* terakhir dari pohon keputusan, hitung kembali jumlah kasus untuk keputusan Layak dan Tidak Layak, dan entropy dari semua kasus berdasarkan variabel Piagam Penghargaan dengan atribut Tidak Ada dan variabel IPK dengan atribut Cukup. Untuk mempermudah, Tabel 2.5 difilter kembali dengan mengambil data dari variabel IPK dengan atribut Cukup, sehingga jadilah tabel seperti berikut.

Tabel 2.7 Data Variabel IPK - Cukup

Kategori IPK	Piagam Penghargaan	Penghasilan Orang Tua	Kelayakan Beasiswa
Cukup	Tidak Ada	Tinggi	Layak
Cukup	Tidak Ada	Rendah	Tidak Layak
Cukup	Tidak Ada	Tinggi	Layak

Kemudian hitung kembali Gain dari atribut dengan menghitung entropy dari setiap variabel terlebih dahulu.

Hasil penghitungannya sebagai berikut :

Entropy(Jumlah Kasus)

$= (- (\frac{1}{3}) * \log_2 \frac{1}{3}) + (- (\frac{2}{3}) * \log_2 \frac{2}{3})$

$= 0.918295834$

Entropy(Penghasilan Orang Tua - Tinggi)

$= (- (\frac{1}{1}) * \log_2 \frac{1}{1}) + (- (\frac{0}{1}) * \log_2 \frac{0}{1})$

$= 0$

Entropy(Penghasilan Orang Tua - Rendah)

$= (- (\frac{0}{2}) * \log_2 \frac{0}{2}) + (- (\frac{2}{2}) * \log_2 \frac{2}{2})$

$= 0$

Setelah itu menghitung *Gain* diantaranya :

Gain(Penghasilan Orang Tua)

$= 0.918295834 – ((\frac{1}{3} * 0) + (\frac{2}{3} * 0))$

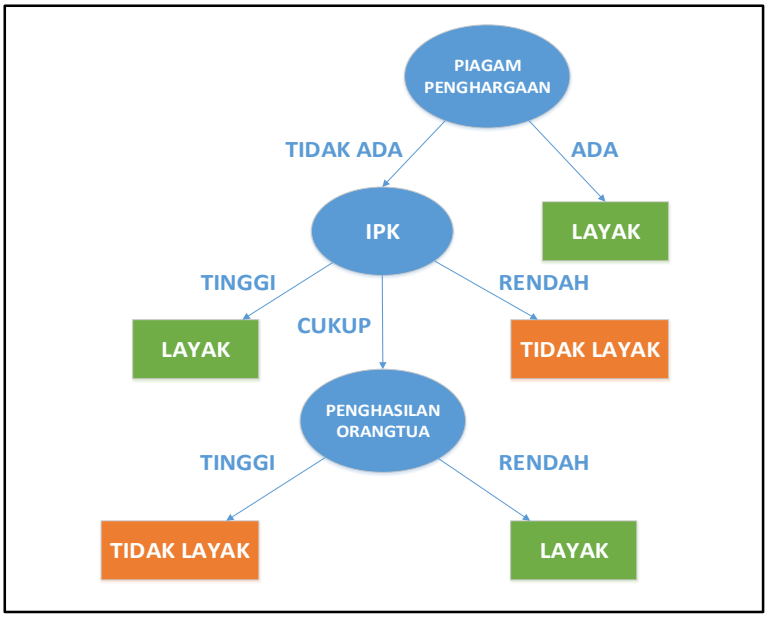
$= 0.918295834 – (0)$

$= 0.918295834$

Tabel 2.8 Pemilihan Node 1.1.2

		Jumlah kasus	Tidak Layak	Layak	Entropy	Gain
Piagam Penghargaan – Tidak Ada & IPK – Cukup		3	1	2	0.918296	
Penghasilan Orang Tua						0.918295834
	Tinggi	1	1	0	0	
	Rendah	2	0	2	0	

Dari hasil Tabel 2.9 sebenarnya dapat diketahui daun atau *leaf* dari node 1.1.2 tanpa menghitung nilai entropy dan gain. Dari kedua nilai tersebut sudah dapat mengklasifikasikan kasus dengan kasus Layak untuk Penghasilan Orang Tua rendah dan kasus Tidak Layak untuk Penghasilan Orang Tua Tinggi. Sehingga tidak perlu dilakukan penghitungan kembali, sehingga pohon keputusan dari node 1.1.2 atau yang terakhir membentuk seperti gambar dibawah ini :



Gambar 2.6 Pohon Keputusan Hasil Perhitungan Node 1.1.2

2.2.6 *Confusion Matrix*

Untuk mengevaluasi model klasifikasi guna memperkirakan apakah objek tersebut benar atau salah maka digunakan *Confussion matrix*. Berikutt adalah tabel *Confussion matrix* :

Tabel 2.9 *Confusion Matrix*[7]

Classification	Predicted class	
	Class = Yes	Class = No
Class=Yes	a (true positive-TP)	b (false negative-FN)
Class=No	c (false positive-FP)	d (true negative-TN)

Setelah data-data telah masuk ke dalam *confusion matrix* maka dapat dihitung nilai akurasinya dengan rumus dibawah ini (Olson & Yong, 2008) [9]:

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FP} \qquad (3)$$

2.2.7 Matlab

Matlab adalah bahasa canggih untuk komputasi teknik. Di dalamnya terdapat kemampuan penghitungan, visualisasi, dan pemrograman dalam suatu lingkungan yang mudah untuk digunakan karena permasalahan dan pemecahannya dinyatakan dalam notasi matematika biasa. Kegunaan Matlab secara umum yaitu untuk :

a. Matematika dan komputasi.
b. Pengembangan algoritma.
c. Pemodelan, simulasi dan pembuatan prototype.
d. Analisis data, eksplorasi dan visualisasi.
e. Pembuatan aplikasi termasuk pembuatan antarmuka grafis.

Matlab merupakan sistem interaktif dengan elemen dasar *database array* yang dimensinya tidak perlu dinyatakan secara khusus. Sehingga memungkinkan untuk memecahkan banyak masalah perhitungan teknik, khususnya yang melibatkan matriks dan vektor [14].