

**ANALISIS PERBANDINGAN KLASIFIKASI DATA MINING
MENGUNAKAN ALGORITMA C5.0 DAN CART PADA
PENERIMAAN PROGRAM KELUARGA HARAPAN**

SKRIPSI

**Diajukan untuk Memenuhi Sebagian Syarat Guna Memperoleh
Gelar Sarjana Matematika
dalam Ilmu Matematika**



Oleh : RIZKA ZULIYANTI
NIM : 2008046041

**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI WALISONGO
SEMARANG
2024**

PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini :

Nama : Rizka Zuliyanti
NIM : 2008046041
Jurusan/Program Studi : Matematika/ Matematika

menyatakan bahwa skripsi yang berjudul :

ANALISIS PERBANDINGAN KLASIFIKASI DATA MINING MENGUNAKAN ALGORITMA C5.0 DAN CART PADA PENERIMAAN PROGRAM KELUARGA HARAPAN

secara keseluruhan adalah hasil penelitian/karya saya sendiri,
kecuali bagian tertentu yang dirujuk sumbernya.

Semarang, 28 Mei 2024
Pembuat pernyataan,



Rizka Zuliyanti
NIM : 2008046041



KEMENTERIAN AGAMA R.I.
UNIVERSITAS ISLAM NEGERI WALISONGO
FAKULTAS SAINS DAN TEKNOLOGI
Jl. Prof. Dr. Hamka (Kampus II) Ngaliyan Semarang
Telp. 024-7601295 Fax. 7615387

PENGESAHAN

Naskah skripsi berikut ini :

Judul : **ANALISIS PERBANDINGAN KLASIFIKASI DATA MINING MENGGUNAKAN ALGORITMA C5.0 DAN CART PADA PENERIMAAN PROGRAM KELUARGA HARAPAN**

Penulis : Rizka Zuliyanti

NIM : 2008046041

Jurusan : Matematika

Telah diujikan dalam sidang *tugas akhir* oleh Dewan Penguji Fakultas Sains dan Teknologi UIN Walisongo dan dapat diterima sebagai salah satu syarat memperoleh gelar sarjana dalam Ilmu Matematika.

Semarang, 24 Juni 2024

DEWAN PENGUJI

Penguji I,

Seftina Diyah Miasary, M.Sc.

NIP : 198709212019032010

Penguji III,

Any Muanalifah, M.Si., Ph.D.

NIP : 198201132011012009

Penguji II,

Ariska Kurnia Rachmawati, M.Sc.

NIP : 198908112019032019

Penguji IV,

Purnadi Kurniawan, M.Sc.

NIP : 199012262019031012

Pembimbing,

Ariska Kurnia Rachmawati, M.Sc.

NIP : 198908112019032019

NOTA DINAS

Semarang, 28 Mei 2024

Yth. Ketua Program Studi Matematika
Fakultas Sains dan Teknologi
UIN Walisongo Semarang

Assalamu'alaikum warahmatullahi wabarakatuh

Dengan ini diberitahukan bahwa saya telah melakukan bimbingan, arahan dan koreksi naskah skripsi dengan:

Judul : ANALISIS PERBANDINGAN KLASIFIKASI DATA
MINING MENGGUNAKAN ALGORITMA C5.0 DAN
CART PADA PENERIMAAN PROGRAM KELUARGA
HARAPAN
Nama : Rizka Zuliyanti
NIM : 2008046041
Jurusan : Matematika

Saya memandang bahwa naskah skripsi tersebut sudah dapat diajukan kepada Fakultas Sains dan Teknologi UIN Walisongo untuk diujikan dalam Sidang Munaqasyah.

Wassalamu'alaikum warahmatullahi wabarakatuh

Pembimbing,



Ariska Kurnia Rachmawati, M.Sc.
NIP : 198908112019032019

ABSTRAK

Dalam upaya mengurangi angka kemiskinan di Indonesia pemerintah membuat sebuah program bantuan sosial salah satunya program keluarga harapan. Proses penerimaan bantuan program keluarga harapan belum menggunakan teknik khusus untuk menentukan manakah yang berhak menerima bantuan. Untuk menangani hal tersebut perlu dibangun sistem klasifikasi status penerima bantuan menggunakan model algoritma C5.0 dan algoritma CART terhadap data penerima bantuan program keluarga harapan Tahun 2023. Dalam penelitian ini, status penerima bantuan akan diklasifikasikan menjadi dua yaitu diterima dan ditolak. Pengujian dilakukan dengan cara mengukur kinerja dua algoritma tersebut menggunakan metode *confussion matrix* dan nilai kurva AUC-ROC. Hasil pengujian menunjukkan bahwa algoritma C5.0 memberikan hasil lebih baik dibandingkan algoritma CART yang ditunjukkan dengan rata-rata tingkat akurasi sebesar 90,12% dan nilai AUC sebesar 0,8998 dan dibandingkan hasil rata-rata algoritma CART yaitu tingkat akurasi sebesar 88,20% dengan nilai AUC sebesar 0,8813. Secara keseluruhan, proporsi yang paling efektif untuk menjalankan kedua algoritma klasifikasi tersebut dalam memprediksi status penerima bantuan yaitu menggunakan proporsi 90%:10%.

Kata kunci : Algoritma C5.0, Algoritma CART, Klasifikasi, Pohon Keputusan, Program Keluarga Harapan.

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT yang telah melimpahkan rahmat serta hidayah-Nya sehingga penulis dapat menyelesaikan skripsi ini. Skripsi ini disusun untuk memenuhi persyaratan guna memperoleh gelar Sarjana Matematika (S.Mat) di Fakultas Sains dan Teknologi Program Studi Matematika Jurusan Matematika UIN Walisongo Semarang, dengan judul **"Analisis Perbandingan Klasifikasi Data Mining Menggunakan Algoritma C5.0 dan CART pada Penerimaan Program Keluarga Harapan"**.

Penulis menyadari bahwa dalam proses penyusunan skripsi ini tidak lepas dari doa, bantuan, bimbingan, motivasi dan peran dari banyak pihak. Sehingga penulis mengucapkan terimakasih yang sebesar-besarnya kepada :

1. Allah SWT yang telah memberikan kemudahan dan kelancaran dalam proses penyusunan skripsi.
2. Bapak Mulyadi, Ibu Mustikomah, Adik Ari Sugiharto dan seluruh keluarga yang telah memberikan semangat, nasihat serta do'a yang tulus dalam setiap langkah yang diambil oleh penulis.
3. Bapak Prof. Dr. H. Nizar Ali, M.Ag. selaku Rektor Universitas Islam Negeri Walisongo Semarang.
4. Bapak Prof. Dr. H. Musahadi, M.Ag. selaku Dekan Fakultas Sains dan Teknologi UIN Walisongo Semarang.
5. Ibu Any Muanalifah M.Si., Ph.D. selaku Ketua Program Studi Matematika Fakultas Sains dan Teknologi UIN Walisongo

Semarang.

6. Ibu Zulaikha, M.Si. selaku dosen wali yang memberikan arahan dan motivasi selama perkuliahan.
7. Ibu Ariska Kurnia Rachmawati, M.Sc. selaku dosen pembimbing yang telah bersedia meluangkan waktu, tenaga dan pikiran untuk memberikan bimbingan dan arahan kepada penulis dalam menyelesaikan skripsi
8. Bapak Moh. Habib Mahfud dan Bapak Moh. Cholid Fauzi selaku petugas pendamping PKH yang telah membantu dalam proses penelitian.
9. Segenap Dosen Fakultas Sains dan Teknologi khususnya dosen matematika yang telah memberikan ilmunya selama perkuliahan.
10. Teman-teman S1 Prodi Matematika angkatan 2020 yang telah memberikan pengalaman selama perkuliahan di UIN Walisongo Semarang.
11. Kepada Puji Lestari selaku sahabat seperjuangan yang telah memberikan semangat, motivasi, dan dukungan selama proses pembuatan skripsi ini.
12. Semua Pihak yang tidak dapat penulis sebutkan satu persatu yang telah memberikan kontribusi hingga selesainya skripsi ini.

Semoga kebaikan semuanya menjadi amal ibadah yang diterima dan mendapat pahala yang berlimpah dari Allah SWT. Aamiin.

Penulis berharap apa yang telah disusun dapat bermanfaat bagi perkembangan ilmu pengetahuan khususnya ilmu matematika

dalam bidang data mining. Penulis menyadari atas segala kekurangan dan kelemahan dalam skripsi ini. Oleh karena itu, penulis mengharapkan saran dan kritik yang membangun guna perbaikan kedepannya.

Semarang, Mei 2024

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
PERNYATAAN KEASLIAN	ii
PENGESAHAN	iv
NOTA PEMBIMBING	v
KATA PENGANTAR	viii
DAFTAR ISI	ix
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiv
DAFTAR SIMBOL	xvi
DAFTAR LAMPIRAN	xvii
BAB 1 PENDAHULUAN	1
1.1. Latar Belakang Masalah	1
1.2. Rumusan Masalah	6
1.3. Tujuan Penelitian	6
1.4. Manfaat Penelitian	6
1.5. Batasan Penelitian	7
BAB 2 LANDASAN PUSTAKA	8
2.1. Landasan Teori	8
2.1.1. Program Keluarga Harapan (PKH)	8
2.1.2. Data Mining	12
2.1.3. <i>Decision Tree</i>	14
2.1.4. Algoritma C5.0	22
2.1.5. <i>Information Gain</i> dan <i>Gain Ratio</i>	26
2.1.6. Algoritma CART (<i>Clasification and Regression Tree</i>)	29
2.1.7. Perbedaan Algoritma ID3, C4.5, C5.0 dan CART	33
2.1.8. <i>Data Training</i> dan <i>Data Testing</i>	34
2.1.9. Mengukur Ketepatan Klasifikasi	34
2.1.10. <i>Confusion Matrix</i>	35
2.1.11. SMOTE (<i>Synthetic Minority Oversampling Technique</i>)	38

2.2. Kajian Pustaka	38
BAB 3 Metode Penelitian	42
3.1. Sumber Data	42
3.2. Waktu dan Tempat Penelitian	42
3.3. Populasi dan Sampel	43
3.3.1. Populasi	43
3.3.2. Sampel	43
3.4. Teknik Pengumpulan Data	44
3.5. Atribut Penelitian	45
3.6. Teknik Analisis Data	48
3.6.1. <i>preprocessing</i> Data	49
3.6.2. Oversampling Menggunakan SMOTE	49
3.6.3. Algoritma C5.0	51
3.6.4. Algoritma CART	52
3.6.5. Perbandingan 2 Algoritma	54
3.7. Diagram Alur Penelitian	55
BAB 4 Hasil dan Pembahasan	56
4.1. Analisis Statistik Deskriptif	56
4.1.1. Atribut Status Penerima Bantuan	56
4.1.2. Atribut Usia	57
4.1.3. Atribut Pendidikan	58
4.1.4. Atribut Pekerjaan	59
4.1.5. Atribut Penghasilan	60
4.1.6. Atribut Aset	61
4.1.7. Atribut Luas Lantai	62
4.1.8. Atribut Jenis Lantai	63
4.1.9. Atribut Dinding	64
4.1.10. Atribut Penerangan	65
4.2. <i>Preprocecing</i> data	66
4.2.1. <i>Missing Value</i>	66
4.2.2. <i>Redundant Data</i>	67
4.3. SMOTE (<i>Synthetic Minority Oversampling Technique</i>)	69
4.4. Pembagian Data <i>Training</i> dan Data <i>Testing</i>	71
4.5. Teknik Data Mining	73
4.5.1. Analisis Algoritma C5.0	73

4.5.2. Analisis Algoritma CART	83
4.5.3. Mengukur Ketepatan Klasifikasi	93
BAB 5 PENUTUP	102
5.1. Kesimpulan	102
5.2. Saran	103
DAFTAR PUSTAKA	104
Lampiran-lampiran	109

DAFTAR TABEL

Tabel	Judul	Halaman
Tabel 2.1	Nominal Penerima PKH Sumber: Kemensos RI (2019)	12
Tabel 2.2	Contoh Dataset Pelanggan Perusahaan <i>E-commerce</i>	28
Tabel 2.3	Contoh Calon Cabang	32
Tabel 2.4	Perbandingan Algoritma ID3, C4.5, C5.0 dan CART	33
Tabel 2.5	<i>Confusion matrix</i>	35
Tabel 3.1	Atribut Bebas	47
Tabel 4.1	Usia berdasarkan Status Penerima Bantuan	57
Tabel 4.2	Pendidikan berdasarkan Status Penerima Bantuan	58
Tabel 4.3	Pekerjaan berdasarkan Status Penerima Bantuan	59
Tabel 4.4	Penghasilan berdasarkan Status Penerima Bantuan	60
Tabel 4.5	Aset berdasarkan Status Penerima Bantuan	61
Tabel 4.6	Luas Lantai berdasarkan Status Penerima Bantuan	62
Tabel 4.7	Jenis Lantai berdasarkan Status Penerima Bantuan	63
Tabel 4.8	Dinding berdasarkan Status Penerima Bantuan	64

Tabel 4.9	Penerangan berdasarkan Status Penerima Bantuan	65
Tabel 4.10	Jumlah Data <i>Training</i> dan Data <i>Testing</i>	71
Tabel 4.11	Data <i>Training</i>	72
Tabel 4.12	Perhitungan <i>Entropy</i> Semua Atribut	74
Tabel 4.13	Perhitungan <i>Gain</i> , dan <i>Gain Ratio Node</i> akar ke-1	76
Tabel 4.14	Calon Cabang Iterasi ke-1	84
Tabel 4.15	Hasil Perhitungan PL dan PR Iterasi ke-1	86
Tabel 4.16	Hasil Perhitungan $P(j t_L)$ dan $P(j t_R)$ Iterasi ke-1	88
Tabel 4.17	Hasil Perhitungan $\varphi(s t)$ Iterasi ke-1	90
Tabel 4.18	<i>Confussion Matrix</i> Algoritma C5.0	94
Tabel 4.19	Nilai <i>Accuracy</i> , <i>Sentivity</i> , <i>Specificity</i> dan AUC untuk Semua Proporsi Algoritma C5.0	96
Tabel 4.20	<i>Confussion Matrix</i> Algoritma CART	96
Tabel 4.21	Nilai <i>Accuracy</i> , <i>Sentivity</i> , <i>Specificity</i> dan AUC untuk Semua Proporsi Algoritma CART	98
Tabel 4.22	Perbandingan Tingkat Akurasi dan Nilai AUC untuk Kedua Algoritma	99
Tabel 4.23	Hasil Prediksi pada Data <i>Testing</i>	100

DAFTAR GAMBAR

Gambar	Judul	Halaman
Gambar 1.1	Grafik Jumlah dan Tingkat Kemiskinan di Indonesia Sumber : Badan Pusat Statistika	1
Gambar 2.1	Syarat pengujian fitur biner	16
Gambar 2.2	Syarat Pengujian fitur bertipe nominal	17
Gambar 2.3	Syarat pengujian fitur bertipe ordinal	17
Gambar 2.4	Syarat pengujian bertipe numerik	18
Gambar 3.1	Diagram Alir Analisis Penelitian	48
Gambar 3.2	Diagram Alir Algoritma C5.0 (Pratiwi, 2020)	52
Gambar 3.3	Diagram Alir Algoritma CART	54
Gambar 3.4	Diagram Alur Penelitian	55
Gambar 4.1	Diagram Lingkaran untuk Atribut Status	56
Gambar 4.2	Atribut Usia	57
Gambar 4.3	Diagram Batang untuk Atribut Pendidikan	59
Gambar 4.4	Diagram Batang untuk Atribut Pekerjaan	60
Gambar 4.5	Diagram Batang untuk Atribut Penghasilan	61
Gambar 4.6	Diagram Batang untuk Atribut Aset	62
Gambar 4.7	Diagram Batang untuk Atribut Luas Lantai	63
Gambar 4.8	Diagram Batang untuk Atribut Jenis Lantai	64
Gambar 4.9	Diagram Batang untuk Atribut Dinding	65
Gambar 4.10	Diagram Batang untuk Atribut Penerangan	66
Gambar 4.11	Output <i>Missing Value</i> Sumber: Output RStudio, 2024	67

Gambar 4.12	Output Koefisien Korelasi Sumber: Output RStudio, 2024	68
Gambar 4.13	<i>Scatterplot</i> Koefisien Korelasi Sumber: Output RStudio, 2024	69
Gambar 4.14	Penerapan Metode SMOTE	70
Gambar 4.15	<i>Node</i> Akar ke-1 Algoritma C5.0	77
Gambar 4.16	Model Pohon Keputusan Algoritma C5.0 Sumber: Output RStudio, 2024	79
Gambar 4.17	<i>Root Node</i> Iterasi ke-1 Algoritma CART	91
Gambar 4.18	Model Pohon Keputusan Algoritma CART	92
Gambar 4.19	Kurva ROC Algoritma C5.0	95
Gambar 4.20	Kurva ROC Algoritma CART	97
Gambar 0.1	Wawancara dengan Pendamping PKH	144
Gambar 0.2	Pengisian Kuesioner dengan Responden	144
Gambar 0.3	Mengikuti Pendampingan dengan Penerima PKH	145
Gambar 0.4	Mengikuti Pendampingan dengan Penerima PKH	145

DAFTAR SIMBOL

Simbol	Arti
A	Atribut
S	Himpunan kasus
m	Jumlah partisi pada atribut A
p_i	Proporsi dari S_i dan S
S_i	Jumlah kasus pada partisi ke- i
S	Jumlah kasus dalam S
s	Kandidat <i>split</i> $s(1, 2, 3, \dots)$
t	<i>Node</i> t
$Q(s t)$	Nilai dari $\sum_{j=1}^{\neq \text{classes}} P(j t_L) - P(j t_R) $
$\varphi(s t)$	Nilai besaran kesesuaian
t_L	<i>Node</i> anak kiri dari <i>node</i> t
t_R	<i>Node</i> anak kanan dari <i>node</i> t
P_L	Jumlah <i>record</i> pada t_L
P_R	Jumlah <i>record</i> pada t_R
$P(j t_L)$	Jumlah <i>record</i> pada kelas j dalam t_L
$P(j t_R)$	Jumlah <i>record</i> pada kelas j dalam t_R
$Entropy(S)$	Nilai keteracakan dari suatu atribut
$Gain(S, A)$	Nilai <i>gain</i> dari suatu atribut
<i>Gain ratio</i>	Alternatif dari <i>information gain</i> yang digunakan untuk memilih atribut yang akan dipecah dalam pohon keputusan
$SplitInfo(S, A)$	Nilai <i>split info</i> dalam suatu atribut
n	Ukuran sampel
N	Ukuran populasi
e	<i>Error</i> (tingkat kesalahan)
<i>Tuple</i>	Berupa kasus-kasus atau <i>record</i>

DAFTAR LAMPIRAN

	Halaman
Lampiran 1 Kuesioner Penelitian	109
Lampiran 2 Data Penerima Bantuan PKH Tahap I	110
Lampiran 3 Data Testing Proporsi 90:10	114
Lampiran 4 <i>Script Software</i> RStudio	115
Lampiran 5 <i>Script dan Output Software</i> RStudio Algoritma C5.0	117
Lampiran 6 <i>Script dan Output Software</i> RStudio Algoritma CART	132
Lampiran 7 Foto Dokumentasi Penelitian	144
Lampiran 8 Daftar Riwayat Hidup	146

BAB 1

PENDAHULUAN

1.1. Latar Belakang Masalah

Kemiskinan adalah kondisi dimana seseorang mengalami kesulitan dalam memenuhi kebutuhan dasar. Kemiskinan menjadi masalah utama yang dihadapi oleh semua negara, terutama di negara berkembang seperti di Indonesia. Angka kemiskinan di Indonesia mengalami naik turun dari bulan Maret 2016 sampai September 2022. Badan Pusat Statistika mencatat pada bulan Maret 2022 jumlah penduduk miskin di Indonesia mencapai angka 26,16 juta orang dengan persentase penduduk miskin mencapai 9,54%, pada periode tersebut mengalami penurunan dengan persentase 0,17% dibandingkan pada September 2021 yang mencapai 9,71%, namun pada September 2022 persentase penduduk miskin mengalami kenaikan mencapai 9,57% dimana total penduduk miskin mencapai 26,36 juta orang. Berikut grafik jumlah dan angka kemiskinan di Indonesia:



Gambar 1.1. Grafik Jumlah dan Tingkat Kemiskinan di Indonesia
Sumber : Badan Pusat Statistika

Dalam upaya mengurangi angka kemiskinan di Indonesia, pemerintah membuat sebuah program bantuan sosial untuk masyarakat, diantaranya yaitu Program Keluarga Harapan (PKH). PKH merupakan program bantuan bersyarat yang telah dilaksanakan sejak tahun 2007 dalam rangka meningkatkan kesejahteraan sosial penduduk miskin dan mengurangi angka kemiskinan. Sebagai program bantuan bersyarat, PKH memberikan kesempatan bagi keluarga miskin, khususnya ibu hamil/nifas/menyusui dan anak belita/anak sekolah untuk mengakses beberapa fasilitas layanan kesehatan dan pendidikan yang ada di wilayahnya. Akses PKH juga mengikutsertakan penyandang disabilitas berat dan lanjut usia dengan tetap menjaga tingkat kepedulian sosial (Kemenkeu, 2015). Allah SWT telah berfirman di dalam kitab Al-Qur'an tentang kepedulian terhadap masyarakat melalui surah An-Nisa' ayat 1 yang berbunyi :

يَا أَيُّهَا النَّاسُ اتَّقُوا رَبَّكُمُ الَّذِي خَلَقَكُمْ مِنْ نَفْسٍ وَاحِدَةٍ وَخَلَقَ مِنْهَا زَوْجَهَا وَبَثَّ مِنْهُمَا رِجَالًا كَثِيرًا وَنِسَاءً ۚ وَاتَّقُوا اللَّهَ الَّذِي تَسَاءَلُونَ بِهِ وَالْأَرْحَامَ إِنَّ اللَّهَ كَانَ عَلَيْكُمْ رَقِيبًا

Artinya: *“Wahai sekalian manusia, bertakwalah kepada Tuhan-mu yang telah menciptakan kamu dari seorang diri (Adam), dari padanya Allah menciptakan isterinya (Hawa), dan dari keduanya Allah memperkembangbiakan laki-laki dan perempuan yang banyak. Bertakwalah kepada Allah dengan namanya-Nya kamu meminta dan peliharalah hubungan kekeluargaan. Susungguhnya Allah senantiasa menjaga dan mengawasimu.”*

Menurut Quraish Shihab (2002), penjelasan dari ayat diatas yaitu sebagai pendahuluan untuk mengantarkan lahirnya kesatuan dan persatuan dalam masyarakat, serta saling menyayangi dan

membantu dikarenakan semua manusia berasal dari keturunan yang sama, tidak membedakan antara perempuan dan laki-laki, besar dan kecil, beragama maupun tidak beragama. Semua dituntut untuk mewujudkan kedamaian dan rasa aman dalam bermasyarakat, serta saling menghargai hak setiap manusia.

Berdasarkan hasil wawancara dengan Ibu Indah (Wawancara, 21 Mei 2023) selaku petugas pendamping PKH Kecamatan Pulokulon, beliau menyatakan bahwa dalam proses penerimaan bantuan tersebut, petugas pendamping PKH akan melakukan survey dengan mendatangi Kepala Desa/Kadus untuk mendapatkan informasi terkait kriteria-kriteria yang dibutuhkan petugas. Kriteria-kriteria tersebut akan menyatakan bahwa seseorang tersebut layak atau tidak untuk mendapatkan bantuan PKH. Jika seseorang itu dikatakan layak mendapatkan bantuan maka untuk usulan-usulan yang diterima akan di verifikasi dan di validasi ke sistem data terpadu kesejahteraan sosial (DTKS) begitupun sebaliknya. Sejauh ini petugas pendamping PKH salah satunya yaitu di Desa Sembungharjo Kecamatan Pulokulon Kabupaten Grobogan belum menggunakan teknik khusus untuk menentukan manakah orang yang berhak menerima bantuan PKH. Dalam bantuan PKH masih banyak ditemukan bahwa penerima bantuan tidak tepat sasaran. Untuk menangani hal tersebut perlu dibangun sistem klasifikasi pada status penerima bantuan secara bertahap. Sehingga Dari permasalahan tersebut bisa disimpulkan bahwa petugas pendamping PKH akan sangat membutuhkan teknik khusus untuk mengklasifikasi penerima bantuan PKH.

Salah satu teknik pengklasifikasian yaitu menggunakan teknik klasifikasi data mining. Dalam data mining sejumlah besar data diperiksa dan dianalisis secara otomatis atau semi-otomatis

untuk menemukan pola dan *rules* yang berguna. Macam-macam klasifikasi yang ada di data mining yaitu *Naive Bayes*, *Decision Tree*, *Support Vector Machine* (SVM), dan yang lainnya (Larose, 2005).

Decision tree atau pohon keputusan dapat dibangun dengan relatif cepat dibandingkan dengan metode klasifikasi lainnya. Pengklasifikasian *decision tree* juga memperoleh *accuracy* yang serupa dan terkadang lebih baik daripada metode klasifikasi lainnya. Algoritma *decision tree* dapat diimplementasikan secara parsial atau paralel berdasarkan volume data, ruang memori yang tersedia pada sumber daya komputer dan skabilitas algoritma (Priyam dkk, 2013). Beberapa algoritma yang digunakan untuk menghasilkan pohon keputusan, seperti ID3, *Clasification and Regression Tree* (CART), C4.5, CHAID, C5.0, dan yang lainnya (Prasetyo, 2014).

Algoritma C5.0 merupakan perluasan dari C4.5 hanya saja C5.0 mempunyai kelebihan dengan algoritma sebelumnya. C5.0 adalah algoritma klasifikasi yang dapat diimplementasikan pada kumpulan data besar. C5.0 lebih baik dari C4.5 dalam hal performa dan memori. Menurut Kusriani dan Luthfi, pemilihan atribut yang akan diproses dalam C5.0 menggunakan *gain ratio*. *Gain ratio* tersebut digunakan sebagai pembentukan atau pemilihan *node* pada pohon. *Parent* bagi *node* selanjutnya, dipilih berdasarkan atribut nilai *gain ratio* tertinggi (Pratiwi dkk, 2020).

Algoritma CART merupakan metode yang fleksibel untuk menggambarkan bagaimana atribut terikat berdistribusi setelah menetapkan atribut bebasnya. Model ini menggunakan pohon biner untuk membagi ruang peramalan menjadi himpunan bagian tertentu yang atribut terikatnya berdistribusi genap secara kontinu (Kharat dan Anil, 2014) (Patil dkk, 2012).

Dalam penelitian ini menggunakan algoritma C5.0 dan CART untuk membandingkan algoritma manakah yang lebih baik dalam mengklasifikasi status penerima bantuan PKH. Karena melihat banyaknya penelitian terdahulu telah menggunakan penelitian ini dalam berbagai konteks, termasuk untuk analisis data sosial, seperti program bantuan sosial. Diantaranya yaitu ada Pakpahan dkk (2018) melakukan penelitian tentang penentuan penerima bantuan dari pemerintah daerah Kabupaten Kutai Kartanegara dengan menerapkan algoritma CART. Menggunakan lima model percobaan diperoleh hasil bahwa dalam percobaan dalam rasio 85% memperoleh tingkat akurasi tertinggi.

Kharat dan Anil (2014) melakukan penelitian tentang klasifikasi tumor dengan menggunakan algoritma CART dan C5.0. Bukti empiris menunjukkan bahwa metode ini sangat cepat dan berhasil. Tetapi tidak memberikan informasi manakah algoritma yang lebih baik.

Pratiwi dkk (2020) melakukan penelitian tentang klasifikasi rata-rata pendapatan masyarakat Desa Teluk Baru menggunakan metode C5.0 dan CART menggunakan diperoleh hasil bahwa metode CART lebih baik dalam mengklasifikasi rata-rata pendapatan masyarakat Desa Teluk Baru dibandingkan metode C5.0. Berdasarkan penelitian tersebut, menjadikan salah satu yang melatarbelakangi penelitian ini, karena secara umum keakuratan kedua algoritma tersebut dapat bervariasi tergantung dataset yang digunakan.

Berdasarkan latar belakang tersebut, peneliti tertarik untuk melakukan penelitian dengan judul “Analisis Perbandingan Klasifikasi Data Mining Menggunakan Algoritma C5.0 dan CART pada Penerimaan Program Keluarga Harapan”.

1.2. Rumusan Masalah

1. Bagaimana model klasifikasi penerima bantuan program keluarga harapan menggunakan algoritma C5.0 ?
2. Bagaimana model klasifikasi penerima bantuan program keluarga harapan menggunakan algoritma CART ?
3. Bagaimana perbandingan model klasifikasi algoritma C5.0 dan algoritma CART pada klasifikasi penerima bantuan program keluarga harapan ?

1.3. Tujuan Penelitian

1. Mendapatkan model klasifikasi penerima bantuan program keluarga harapan menggunakan algoritma C5.0.
2. Mendapatkan model klasifikasi penerima bantuan program keluarga harapan menggunakan algoritma CART.
3. Mengetahui perbandingan model klasifikasi algoritma C5.0 dan algoritma CART pada klasifikasi penerima bantuan program keluarga harapan.

1.4. Manfaat Penelitian

1. Memberikan wawasan dan gambaran mengenai algoritma C5.0 dan algoritma CART.
2. Hasil dari penelitian ini diharapkan bisa memberi masukan untuk Dinas Sosial Kabupaten Grobogan dalam menentukan calon dari penerima bantuan program keluarga harapan.

3. Bisa digunakan untuk referensi pada penelitian selanjutnya khususnya dalam bidang ini.

1.5. Batasan Penelitian

1. Bantuan sosial yang dibahas dari penelitian ini adalah penerima program keluarga harapan di Desa Sembungharjo Kecamatan Pulokulon Kabupaten Grobogan Tahun 2023.
2. Menggunakan lima macam proporsi data *training* dan data *testing*
3. Pada penelitian ini menggunakan *Microsoft Excel* 2013 dan *software* RStudio.

BAB 2

LANDASAN PUSTAKA

2.1. Landasan Teori

2.1.1. Program Keluarga Harapan (PKH)

Program keluarga harapan atau yang disingkat dengan PKH adalah program bantuan sosial kepada masyarakat yang telah memenuhi persyaratan tertentu dalam rangka mengurangi angka kemiskinan. Program ini dikenal sebagai program bantuan tunai bersyarat yang telah dimulai sejak tahun 2007. PKH bertujuan untuk mengurangi tingkat kemiskinan dan memperbaiki perilaku yang kurang mendukung dalam meningkatkan kesejahteraan (Kemenkeu, 2015). Setiap tahunnya PKH mengalami kenaikan target sasaran penerima bantuan tunai tersebut.

Menurut Kemenkeu (2015) target sasaran PKH adalah keluarga miskin yang telah terdaftar dalam data terpadu kesejahteraan sosial (DTKS) yang memiliki salah satu kriteria peserta program bantuan tersebut, yaitu :

- a. Kategori ibu hamil/menyusui/nifas
- b. Kategori anak pada usia 0-6 tahun
- c. Kategori anak sekolah (SD, SMP, SMA)
- d. Kategori lanjut usia atau lansia (60 tahun keatas)
- e. Kategori penyandang disabilitas berat

PKH diharapkan dapat mengurangi angka kemiskinan dalam jangka panjang serta mengurangi biaya pengeluaran keluarga

miskin dalam jangka pendek. Sebagai upaya peningkatan kualitas pendidikan, kesehatan dan meningkatkan kualitas diri pada masyarakat.

Menurut Kemensos RI (2020) dalam penerimaan bantuan PKH yang menjadi acuan seseorang dapat diterima atau tidaknya tergantung dari data di DTKS. DTKS adalah kumpulan data yang mencakup informasi tentang kesejahteraan sosial, penerima bantuan dan upaya pemberdayaan sosial, serta potensi dan sumber daya yang mendukung kesejahteraan sosial. DTKS dijadikan acuan lembaga-lembaga untuk menerima bantuan berupa PKH, sembako, dan PBI JK. Salah satu syarat mutlak untuk mendapatkan bantuan PKH yaitu keluarga penerima manfaat (KPM) wajib terdaftar dalam DTKS. Langkah-langkah untuk pendaftaran DTKS sebagai berikut:

- a. Warga mendaftarkan diri ke Desa/Kelurahan atau bisa melalui usulan dari RT/RW ke Desa/Kelurahan.
- b. Usulan tersebut akan dibawa ke musyawarah Desa/Kelurahan.
- c. Petugas Desa melakukan verifikasi dan validasi di lapangan, kemudian hasil verifikasi dan validasi di input melalui aplikasi SIKS-NG dan akan diteruskan ke Dinas Sosial daerah Kabupaten/Kota atau.
- d. Masyarakat mendaftar secara mandiri melalui aplikasi Cek Bansos dan pastikan akun sudah teraktivasi oleh admin Kemensos.
- e. Usulan tersebut akan masuk di disposisi SIKS-NG kemudian dilakukan verifikasi dan validasi oleh Dinas Sosial daerah Kabupaten/Kota.

- f. Disahkan oleh Bupati/Walikota melalui Dinas Sosial.
- h. Selanjutnya proses usulan data yang diajukan oleh pemerintah daerah Kabupaten/Kota diteruskan kepada Menteri Sosial RI.
- i. Kemudian kementerian Sosial RI melakukan pengolahan usulan data tersebut.
- j. Menteri Sosial RI menetapkan dan mengumumkan DTKS.

Menurut Kemensos RI (2020) bahwa KPM yang terdaftar di DTKS merupakan keluarga miskin atau rentan miskin yang kriteria penilainnya ditentukan oleh Keputusan Menteri Sosial RI No. 146/HUK/2013 tentang penetapan kriteria dan pendataan fakir miskin dan orang tidak mampu. Menurut standar Badan Pusat Statistika (BPS) kriteria masyarakat miskin sebagai berikut:

- a) Luas lantai rumah kurang dari $8m^2$ /orang.
- b) Jenis lantai rumah (tanah, bambu, kayu murahan).
- c) Jenis dinding rumah (bambu, kayu dengan kualitas rendah, tembok tanpa diplester).
- d) Tidak memiliki toilet atau wc umum yang digunakan dengan rumah tangga lain.
- e) Sumber penerangan bukan listrik atau penerangan listrik dari listrik negara 450 watt.
- f) Sumber air minum (air sungai, mata air tidak terlindungi, air sumur, air hujan).
- g) Bahan bakar untuk memasak (kayu, minyak tanah, arang).

- h) Hanya mampu mengkonsumsi ayam/daging/susu perminggu.
- i) Hanya mampu membeli satu stel pakaian pertahun.
- j) Hanya mampu makan satu/dua kali perhari.
- k) Tidak mampu membayar biaya perawatan di puskesmas atau poliklinik
- l) Sumber penghasilan kepala keluarga (petani dengan lahan $500m^2$, buruh tani, buruh bangunan, buruh perkebunan, nelayan dan pekerjaan yang lainnya) dengan pendapatan kurang Rp. 600.000/bulan.
- m) Tingkat pendidikan tertinggi kepala keluarga (tidak sekolah, belum tamat SD, tamat SD).
- n) Tidak mempunyai tabungan atau barang yang bisa dijual dengan minimal penjualan sebesar Rp. 500.000,-.

Dasar pelaksanaan PKH adalah sebagai berikut:

- (a) UU No. 15 Tahun 2011 tentang penanganan fakir miskin.
- (b) KEPMENSOS RI 146/HUK/2013 tentang penetapan kriteria dan pendataan fakir miskin dan orang tidak mampu.
- (c) PERMENSOS No. 1 Tahun 2018 program keluarga harapan.
- (d) PERMENSOS No. 5 Tahun 2021 pengelolaan data terpadu kesejahteraan sosial.

Kementerian Sosial menetapkan jumlah nominal yang diterima oleh penerima PKH dari setiap kriteria/kategori berbeda-beda, dapat dilihat pada tabel 2.1

Tabel 2.1. Nominal Penerima PKH
Sumber: Kemensos RI (2019)

Kriteria/Kategori	Jumlah Nominal
Ibu hamil/ menyusui/nifas	Rp. 2.400.000,-
Anak usia 0-6 tahun	Rp. 2.400.000,-
Anak SD/Sederajat	Rp. 900.000,-
Anak SMP/Sederajat	Rp. 1.500.000,-
Anak SMA/ sederajat	Rp. 2.000.000,-
Lansia (60 tahun keatas)	Rp. 2.400.000,-
Disabilitas berat	Rp. 2.400.000,-

Berdasarkan hasil wawancara dengan bapak Habib (Wawancara, 5 Juni 2023) petugas pendamping PKH Desa Sembungharjo, beliau menyatakan bahwa pencairan PKH setiap tahunnya akan cair 3 atau 4 tahap. Dari nominal yang sudah ditetapkan akan dibagi 3 atau 4 tahap pencairannya. Dalam satu rumah tangga seluruh keluarga berhak untuk menerima bantuan tersebut apabila memenuhi kriteria peserta program bantuan tersebut.

2.1.2. Data Mining

Data mining adalah proses penemuan melalui kumpulan data pola, hubungan, dan wawasan besar yang memandu perusahaan mengukur dan mengelola dimana mereka berada dan memprediksi dimana mereka akan berada di masa depan. Sejumlah besar data dan database dapat diperoleh dari berbagai sumber data dan dapat dijadikan informasi serta pengetahuan yang berguna. Metode data mining seperti *artificial intelligence* (AI), *machine learning* dan pemodelan prediktif dapat disertakan (Maung, 2020).

Menurut Muflikhah dkk (2018), terdapat beberapa pendefinisian dari data mining, sebagai berikut:

- a. Pengurain (yang tidak mudah) dari beberapa data yang telah menjadi informasi dan mempunyai potensi yang jelas (implisit) dari yang sebelumnya belum diketahui.
- b. Analisis dan penggalian menggunakan peranti atau semi otomatis dari sekumpulan data yang bertujuan untuk menemukan pola yang mempunyai arti.
- c. Data mining merupakan bagian dari *knowledge discovery* dalam database (KDD).

Menurut Larose (2005), peranan data mining secara umum, sebagai berikut:

- a) Deskripsi, adalah cara yang digunakan untuk menggambarkan pola dan tren yang terdapat dalam data.
- b) Estimasi, seperti klasifikasi kecuali bahwa atribut targetnya numerik dan tidak kategorik. Model disusun menggunakan *record* yang memberikan nilai variabel target serta prediktor.
- c) Prediksi, seperti klasifikasi dan estimasi, kecuali jika prediksi hasilnya terletak pada masa depan. Setiap metode dan teknik yang digunakan untuk klasifikasi dan estimasi juga dapat digunakan dalam keadaan yang sesuai untuk prediksi.
- d) Klasifikasi, adanya atribut kategori target. Model data mining memeriksa sekumpulan *record* dalam jumlah besar, setiap *record* berisi informasi tentang atribut target serta sekumpulan input atau atribut prediksi.

- e) *Clustering*, mengacu pada pengelompokan *record*, pengamatan, atau kasus-kasus ke dalam kelas yang serupa. *Clustering* sering kali dilakukan sebagai tahap awal dalam proses data mining.
- f) Asosiasi, tugas asosiasi didalam data mining adalah menemukan atribut yang mana “bersama-sama”. Secara umum dikenal sebagai analisis afinitas atau *market basket analysis*, tugas asosiasi berusaha mengungkapkan aturan untuk mengukur hubungan antara dua atribut atau lebih. Aturan asosiasi berbentuk “*if antecedent, then consequent*” bersama-sama dengan ukuran *support* dan *confidence* yang terkait dengan aturan.

2.1.3. *Decision Tree*

Decision tree atau pohon keputusan adalah salah satu struktur *machine learning* yang paling sederhana dan paling berguna. *Decision tree*, seperti namanya berasal dari fakta bahwa algoritma terus membagi database menjadi bagian-bagian yang lebih kecil sampai data telah dibagi menjadi satu contoh, yang kemudian diklasifikasikan. Model dari keputunnya berbentuk seperti pohon. Dalam riset operasi terutama dalam analisis keputusan, pohon keputusan seringkali digunakan sebagai alat untuk membantu mengenali strategi yang tepat untuk mencapai tujuan. Tetapi juga merupakan alat yang populer dalam *machine learning* (Maung, 2020).

Menurut Prasetyo (2014), *decision tree* adalah pohon yang digunakan sebagai teknik penalaran untuk memperoleh jawaban atas suatu masukan masalah. Tidak selalu menghasilkan pohon

biner. Menghasikan pohon biner Ketika semua fungsi dalam database menggunakan dua jenis nilai kategorikal. Di sisi lain, jika fitur berisi lebih dari dua nilai kategori atau tipe numerik yang berbeda, pohon yang dihasilkan biasanya bukan pohon biner.

Menurut Muflikhah dkk (2018), beberapa konsep dasar *decision tree* adalah sebagai berikut:

- a. Proses klasifikasi selalu dimulai dengan mengajukan pertanyaan untuk mengidentifikasi kelompok.
- b. Dalam *decision tree* pertanyaan pertama berada pada level 0 atau *root node* dan pertanyaan berikutnya berada pada level I atau cabang dari *root node*.
- c. Setiap *node* yang ditanyakan adalah nilai suatu atribut.
- d. Jawaban untuk setiap pertanyaan diungkapkan di cabang-cabang *node*.
- e. Pertanyaan akan berakhir ketika sudah jelas kelas atau objek apa yang dicari.
- f. *Decision tree* digunakan untuk persoalan yang nilai keluarannya berupa data diskrit.

Menurut Prasetyo (2014), tiga pendekatan klasik dalam *decision tree*:

- 1) Pohon klasifikasi, digunakan untuk memprediksi data baru dengan label kelas yang tidak diketahui, adalah pendekatan yang sangat umum digunakan.

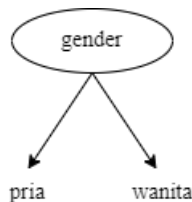
- 2) Pohon regresi digunakan jika hasil yang diprediksi dianggap sebagai nilai nyata yang dapat dicapai. Misalnya, kenaikan harga rumah, harga minyak, prakiraan inflasi tahunan, dan lainnya.
- 3) Menggunakan CART, ketika proses regresi dan klasifikasi digunakan secara bersama.

Beberapa pilihan algoritma untuk menghasilkan *decision tree*, seperti CART, C4.5, ID3, C5.0, CHAID, dan yang lainnya.

Menurut Prasetyo (2014), hal yang terpenting dalam *decision tree* adalah bagaimana cara menyatakan syarat pengujian pada *node*. Terdapat tiga kelompok penting dalam syarat pengujiannya:

(1) Fitur biner

Fitur biner adalah fitur yang memiliki dua nilai yang berbeda. Syarat pengujiannya ketika fitur ini menjadi *node* (akar ataupun internal) kecuali mempunyai dua pilihan cabang. Contoh pemecahannya dapat dilihat pada gambar 2.1.

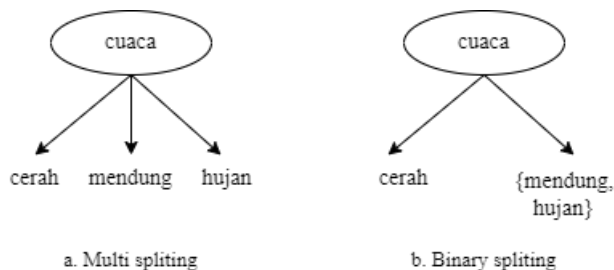


Gambar 2.1. Syarat pengujian fitur biner

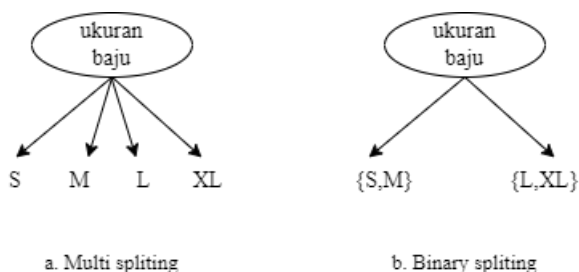
(2) Fitur bertipe kategorikal

Fitur bertipe kategorikal (ordinal atau nominal) adalah fitur yang memiliki kemungkinan beberapa nilai yang berbeda.

Secara umum ada 2 syarat pengujian pemecahan yaitu pemecahan (*binary spliting*) dan (*multi spliting*). Contoh pemecahannya disajikan pada gambar 2.2 dan 2.3.



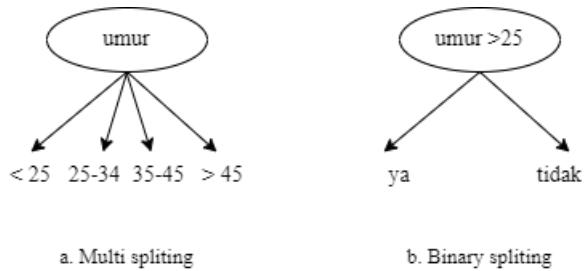
Gambar 2.2. Syarat Pengujian fitur bertipe nominal



Gambar 2.3. Syarat pengujian fitur bertipe ordinal

(3) Fitur bertipe numerik

Fitur bertipe numerik adalah syarat pengujian dalam *node* (akar ataupun internal). Kasus pemecahan biner, algoritma akan memeriksa semua pemecahan yang terbaik. Dalam kasus multiarah, algoritma harus memeriksa semua kemungkinan interval dari nilai interval kontinu. Contoh pemecahannya disajikan dalam gambar 2.4.



Gambar 2.4. Syarat pengujian bertipe numerik

Menurut Pramana dkk (2018), secara umum algoritma *decision tree* bekerja dengan cara *top-down*, dimulai dengan memilih atribut yang *best predictor* (prediksi terbaik) sebagai akar (*root*). Kemudian atribut selanjutnya yang *best splitting attribute* menjadi cabang dari *tree* yang terbentuk, dan seterusnya hingga semua database telah terbagi atau semua atribut menjadi cabang dari *tree*. Dalam proses pembentukan *decision tree*, terdapat beberapa metrik yang digunakan untuk menentukan atribut terbaik sebagai *best predictor* seperti *entropy* dan *gain ratio*, *information gain*, dan *gini index*.

Berikut adalah Algoritma dasar untuk menghasilkan *decision tree* dari *training tuple* (Han dan Kember, 2011):

Algorithm 1 Generate Decision Tree

Input:

- a. Data partition, D , which is a set of training tuples and their associated class labels;
- b. *attribute_list*, the set of candidate attributes;
- c. *Attribute_selection_method*, a procedure to determine the splitting criterion that "best" partitions the data tuples into individual classes. This criterion of a *splitting_attribute* and, possibly, either a *split_point* of splitting subset.

Output: A decision tree**Method:**

1. create a node N
 2. **if** tuple in D are all of the same class, C , **then**
 3. return N as a leaf node labeled with the class C
 4. **if** *attribute_list* is empty **then**
 5. return N as a leaf node labeled with the majority class in D ,
// majority voting
 6. apply *Attribute_selection_method* (D , *attribute_list*) to **find**
the "best" *splitting_criteria*s
 7. label node N with *splitting_criteria*s
 8. **if** *splitting_attribute* is discrete_valued **and**
multiway splits allowed **then** // not restricted to binary trees
 9. *attribute_list* $\leftarrow$ *attribute_list* $-$ *splitting_attribute*; //
remove *splitting_attribute*
 10. **for each** outcome j of *splitting_criterion*
// partition the tuples and grow subtrees for each partition
 11. let D_j be the set of data tuples in D satisfying outcome j ; //
a partition
 12. **if** D_j is empty **then**
 13. attach a leaf labeled with the majority class in D to node N ;
 14. **else** attach the node returned by **Generate_decision_tree**
(D_j , *attribute_list*) to node N ; **endfor**
 15. return N ;
-

1. Pohon dimulai sebagai sebuah *node* tunggal, N , yang mewakili *training tuple* di D (langkah 1).
2. Jika *tuple* di D berasal dari kelas yang sama, maka *node* N menjadi *leaf* dan diberi label dengan kelas tersebut (langkah 2 dan 3). Perhatikan bahwa langkah 4 dan 5 adalah kondisi terminasi. Semua kondisi penghentian dijelaskan di akhir algoritma.
3. Jika tidak, algoritma akan memanggil metode pemilihan atribut untuk menentukan *splitting criterion*. *Splitting criterion* memberi tahu tentang atribut mana yang akan diuji pada *node* N dengan menentukan cara "best" untuk memisahkan atau mempartisi *tuple* di D ke dalam kelas individual (langkah 6). *Splitting criterion* juga memberi tahu tentang cabang mana yang tumbuh dari *node* N sehubungan dengan hasil pengujian yang dipilih. Lebih khusus lagi, *splitting criterion* menunjukkan *splitting* atribut dan juga dapat menunjukkan *split-point* atau *splitting subset*. *Splitting criterion* ditentukan sedemikian rupa sehingga, idealnya, partisi yang dihasilkan pada setiap cabang adalah semurni mungkin. Sebuah partisi dikatakan murni jika *tuple* di dalamnya berasal dari kelas yang sama. dengan kata lain, jika membagi *tuple* di D berdasarkan hasil *splitting criterion* yang saling eksklusif, diharapkan partisi yang dihasilkan semurni mungkin.
4. *Node* N diberi label dengan *splitting criterion*, yang berfungsi sebagai pengujian pada *node* tersebut (langkah 7). Sebuah cabang ditumbuhkan dari *node* N untuk setiap hasil *splitting criterion*. *Tuple* di D di partisi sesuai (langkah 10 hingga

11). Ada tiga kemungkinan yaitu ketika A menjadi atribut *splitting*. A memiliki nilai v yang berbeda $[a_1, a_2, \dots, a_v]$, berdasarkan data *training*.

- (a) A bernilai diskrit: dalam hal ini, hasil pengujian pada *node N* berhubungan langsung dengan nilai A yang diketahui. Sebuah cabang dibuat untuk setiap nilai yang diketahui, a_j , dari A dan diberi label dengan nilai tersebut. Partisi D_j adalah subset dari *tuple* berlabel kelas di D yang memiliki nilai a_j dari A . Karena semua *tuple* dalam partisi tertentu mempunyai nilai A yang sama, A tidak perlu dipertimbangkan dalam setiap partisi *tuple* di masa mendatang. Oleh karena itu, dihapus dari daftar atribut (langkah 8 dan 9).
- (b) A bernilai kontinu: dalam hal ini, pengujian pada *node N* memiliki dua kemungkinan hasil, sesuai dengan kondisi A titik pisah dan $A \leq \textit{splitt - point}$ dan $A > \textit{splitt - point}$, dimana *splitt-point* adalah *splitt-point* yang dikembalikan oleh metode *splitting* atribut sebagai bagian bagian *splitting criterion*. (Dalam praktiknya, *splitt-point*, a , sering kali diambil sebagai titik tengah dari dua nilai A yang berdekatan dan oleh karena itu mungkin sebenarnya bukan nilai A yang sudah ada sebelumnya dari data *training*).
- (c) A bernilai diskrit dan pohon biner harus dihasilkan (seperti yang ditentukan oleh ukuran *splitting* atribut atau algoritma yang digunakan). Pengujian pada *node N* berbentuk " $A \in S_A?$ ", dimana S_A adalah subset *splitting* untuk A , dikembalikan dengan metode

pemilihan atribut sebagai bagian dari *splitting criterion* adalah subset dari nilai A yang diketahui. Jika *tuple* tertentu memiliki nilai a_j dari A dan jika $a_j \in S_A$, maka pengujian pada *node* N terpenuhi.

5. Algoritma ini menggunakan proses yang sama secara rekursif untuk membuat pohon keputusan untuk *tuple* di setiap partisi yang dihasilkan, D_j , dari D (langkah 14).
6. Partisi rekursif berhenti hanya ketika salah satu dari kondisi penghentian berikut ini benar:
 - (a) Semua *tuple* di partisi D (diwakili di *node* N) termasuk dalam kelas yang sama (langkah 2 dan 3).
 - (b) Tidak ada atribut yang tersisa dimana *tuple* dapat di partisi lebih lanjut (langkah 4). Dalam hal ini, pemungutan suara mayoritas digunakan (langkah 5). Hal ini konversi *node* N menjadi *leaf* dan memberi label dengan kelas yang paling umum di D . Alternatifnya, distribusi kelas dari *tuple node* dapat disimpan.
 - (c) Tidak ada *tuple* untuk cabang tertentu, yaitu partisi D_j kosong (langkah 12). Dalam hal ini, sebuah *leaf* dibuat dengan kelas mayoritas di D (langkah 13).
7. Pohon keputusan yang dihasilkan dikembalikan (langkah 15).

2.1.4. Algoritma C5.0

C5.0 adalah algoritma yang banyak digunakan sebagai metode pohon keputusan dalam *machine learning*, yang dikembangkan

oleh J. Ross Quinlan pada tahun 1994. C5.0 adalah algoritma penerus dari algoritma C4.5 yang merupakan perpanjangan dari ID3. Pohon keputusan yang diturunkan oleh C5.0 dapat digunakan untuk klasifikasi disebut juga sebagai pengklasifikasian statistik. Algoritma C5.0 menggunakan nilai *gain ratio* tertinggi untuk memilih atribut dalam proses menentukan model pohon keputusan. C5.0 menghasilkan *tree* dengan jumlah cabang per *node* bervariasi. C5.0 dapat dengan mudah menangani banyak data seperti data kategorikal, kontinu, tanggal, waktu, dan stempel waktu (Maung, 2020).

C4.5 membuat sejumlah perbaikan pada ID3 untuk fakta ini dapat menangani atribut diskrit dan kontinyu. Untuk menangani atribut kontinu, C4.5 membuat ambang batas dan kemudian membagi daftar menjadi atribut yang nilai atributnya kurang dari atau sama dengan ambang batas dan atribut yang lebih besar darinya. Itu juga dapat menangani kumpulan data *training* dengan *missing value*. *Missing value* tidak diperhitungkan dalam penghitungan *entropy* dan *information gain*. C5.0 menawarkan sejumlah perbaikan pada C4.5. Ini jelas lebih cepat dari C4.5 (beberapa kali lipat). C5.0 lebih hemat memori dibandingkan C4.5. Hasil yang diperoleh serupa dengan C4.5 dengan pohon keputusan yang jauh lebih kecil. Peningkatan akan meningkatkan kualitas pohon dan memberikan hasil yang lebih akurat. C5.0 memungkinkan Anda mempertimbangkan berbagai kasus dan jenis kesalahan klasifikasi. C5.0 secara otomatis menyaring atribut untuk menghapus atribut yang mungkin tidak membantu (Maung, 2020).

Menurut Wu Xidong dan Kumar (2008), C4.5 dirancang oleh J. R. Quinlan, dinamakan demikian karena merupakan turunan

dari pendekatan ID3 untuk mendorong pohon keputusan, yang pada gilirannya merupakan inkarnasi ketiga dalam serangkaian “*iterative dichotomizers*”. Pohon keputusan adalah serangkaian pertanyaan yang disusun secara sistematis sehingga setiap pertanyaan menanyakan suatu atribut dan cabang berdasarkan nilai atribut tersebut. Di *leaf tree* ditempatkan prediksi atribut kelas. Oleh karena itu, pohon keputusan tidak berbeda dengan serangkaian pertanyaan pemecahan masalah yang mungkin ditemukan.

Algorithm 2 C4.5 Decision Tree

Input: an attribute-predictor dataset D

```

1:  $Tree = \{\}$ 
2: if  $D$  is "pure" OR other stopping criteria met then
3:   Terminate
4: end if
5: for all attribute  $a \in D$  do
6:   compute information – theoretic criteria if we split on  $a$ 
7: end for
8:  $abest = \text{Best attribute according to above computed criteria}$ 
9:  $Tree = \text{Create a decision node that tests } abest \text{ in the root}$ 
10:  $Dv = \text{Induced sub-datasets from } D \text{ based on } abest$ 
11: for all  $Dv$  do
12:    $Tree_v = C4.5(Dv)$ 
13:   attach  $Tree_v$  to the corresponding branch of  $Tree$ 
14: end for
15: return  $Tree$ 

```

Algoritma 2 menjelaskan bagaimana algoritma C4.5 beroperasi menggunakan dataset " D " sebagai masukan. Secara ringkas, langkah kerja algoritma C4.5 adalah sebagai berikut (J. R. Quinlan, 1993):

1. Hitung frekuensi masing-masing kejadian pada setiap predictor dari atribut yang digunakan terhadap predictor dari atribut target.
2. Hitung nilai *entropy* total dan *entropy* untuk setiap atribut predictor. *Entropy* berfungsi sebagai penentu untuk penggunaan atribut dalam pembentukan pohon keputusan, karena tidak semua atribut akan digunakan dalam setiap percabangan pohon keputusan.
3. Hitung *information gain* untuk setiap atribut. *Information gain* akan digunakan untuk membuat pohon keputusan. Pada iterasi pertama, atribut yang memiliki *information gain* tertinggi akan dipilih sebagai *root tree*.
4. Bentuk *node* awal yang mengandung atribut dengan *information gain* tertinggi.
5. Ulangi langkah perhitungan *information gain* sampai semua data telah termasuk dalam kelas yang sama. Atribut yang dipilih tidak diikuti lagi dalam perhitungan nilai *information gain* pada iterasi selanjutnya.

Pembuatan pohon keputusan pada algoritma C5.0 mirip dengan algoritma C4.5 dalam hal perhitungan kemunculan kejadian, *entropy*, dan *information gain*. Namun perbedaannya adalah bahwa setelah menghitung *information gain*, algoritma C5.0 melanjutkan dengan menghitung *gain ratio*. *Gain Ratio* ini memanfaatkan *information gain* dan *entropy* yang telah dihitung sebelumnya untuk memilih atribut terbaik dalam menentukan *root node* nya.

2.1.5. Information Gain dan Gain Ratio

Rumusnya untuk menghitung nilai *entropy* seperti persamaan berikut (Han dan Kember, 2011) (Wu Xidong dan Kumar, 2008):

$$Entropy(S) = - \sum_{i=1}^m p_i \log_2(p_i) \quad (2.1)$$

dengan:

S = himpunan kasus

m = jumlah partisi S

p_i = proporsi dari S_i terhadap S

Untuk mencari nilai *information gain* menggunakan persamaan berikut (Han dan Kember, 2011) (Wu Xidong dan Kumar, 2008):

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^m \frac{|S_i|}{|S|} Entropy(S_i) \quad (2.2)$$

dengan:

S = himpunan kasus

A = atribut

m = jumlah partisi S

S_i = jumlah kasus pada partisi ke- i

S = jumlah kasus dalam S

Selanjutnya *split info* dapat dihitung dengan persamaan berikut (J. R. Quinlan, 1993):

$$SplitInfo(S, A) = - \sum_{i=1}^m \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|} \quad (2.3)$$

dengan:

S = himpunan kasus

A = atribut

m = jumlah partisi S

S_i = jumlah kasus pada partisi ke- i

S = jumlah kasus dalam S

Setelah mendapatkan nilai *entropy* dan nilai *gain*, selanjutnya menghitung nilai *gain rationya*. Untuk mencari nilai *gain ratio* menggunakan rumus dasar dari perhitungan *gain ratio*, sebagai berikut (J. R. Quinlan, 1993):

$$GainRatio(S, A) = \frac{Gain(S, A)}{SplitInfo(S, A)} \quad (2.4)$$

dengan:

$Gain(S, A)$ = nilai *gain* dari suatu atribut

$SplitInfo(S, A)$ = nilai *split info* dalam suatu atribut

Ulangi proses tersebut hingga kasus-kasus pada cabang mempunyai kelas yang sama. Disini, atribut yang telah menjadi cabang dikeluarkan terlebih dahulu, kemudian dilanjutkan hingga mendapatkan cabang yang baru.

Sebagai contoh perhitungan untuk mencari nilai *Entropy*, *Gain* dan *Gain Ratio* yaitu menggunakan dataset yang berisi informasi tentang pelanggan disebuah perusahaan *e-commerce*, sebagai berikut:

Tabel 2.2. Contoh Dataset Pelanggan Perusahaan *E-commerce*

Umur	Pendapatan	Pengeluaran Bulanan	Status Pelanggan (Label)
Muda	Rendah	Rendah	Tidak
Muda	Sedang	Sedang	Tidak
Muda	Sedang	Tinggi	Ya
Muda	Tinggi	Tinggi	Ya
Dewasa	Rendah	Rendah	Tidak
Dewasa	Sedang	Sedang	Tidak
Dewasa	Tinggi	Rendah	Ya
Tua	Rendah	Tinggi	Ya
Tua	Sedang	Rendah	Ya
Tua	Tinggi	Sedang	Ya

1. Menghitung $Entropy(S)$

$Entropy$ dari sebuah himpunan data S (label) dengan dua kelas (Ya dan Tidak).

$$Entropy(S) = -P_i \log_2(p_i) - p_i \log_2(p_i)$$

$$Entropy(S) = -0,6 \log_2(0,6) - 0,4 \log_2(0,4) = 0,971$$

2. Menghitung $Gain(S, A)$ untuk setiap atribut

$$Gain(S, Umur) = Entropy(S) - \left[\frac{4}{10} \cdot Entropy(Muda) + \frac{3}{10} \cdot Entropy(Dewasa) + \frac{3}{10} \cdot Entropy(Tua) \right]$$

Terlebih dahulu menghitung $Entropy(Muda, Dewasa, Tua)$ menggunakan rumus $Entropy$ yang sama di langkah 1, sehingga

$$Gain(S, Umur) = 0,971 - [0,4(1) + 0,3(0,918) + 0,3(0)] = -0,947$$

Lakukan hal yang sama untuk atribut pendapatan dan pengeluaran bulanan.

3. Menghitung $GainRatio(S, A)$ untuk setiap atribut

$$GainRatio(S, Umur) = \frac{Gain(S, Umur)}{SplitInfo(S, Umur)}$$

Terlebih dahulu menghitung nilai $SplitInfo(S, Umur)$ yaitu nilai $Entropy$ total dari setiap atribut, sehingga

$$GainRatio(S, Umur) = \frac{0,947}{[1 + 0,918 + 0]} = -0,494$$

Lakukan hal yang sama untuk atribut pendapatan dan pengeluaran bulanan.

2.1.6. Algoritma CART (*Classification and Regression Tree*)

Classification and regrestion (CART) diperkenalkan oleh sekelompok ahli statistik yaitu Leo Breiman, Jerome Freidman, Richard Olshen, Charles pada tahun 1984. Algoritma tersebut dapat menghasilkan pohon klasifikasi atau regresi, tergantung pada atribut dependennya apakah numerik atau kategorik. Jika atribut hasil bersifat kategorik, maka akan menghasilkan pohon klasifikasi (*classification tree*). Jika atribut kontinu, maka akan menghasilkan pohon regresi (*regression tree*). Sehingga dapat membangun *decision tree* menjadi dua nilai saja (pohon biner) (Maung, 2020).

Beberapa kelebihan algoritma CART dibandingkan algoritma lainnya, yaitu lebih akurat dan lebih cepat perhitungannya, hasilnya lebih mudah diinterpretasikan, selain itu algoritma CART digunakan sebagai himpunan data dalam jumlah besar, menggunakan banyak atribut dengan skala atribut campuran bisa melalui prosedur pemilihan biner. Pembentukan pohon klasifikasi

optimal menggunakan data *training* sedangkan validasi model yaitu seberapa besar kemampuan model dalam memprediksi data yang baru menggunakan data *testing* (Pratiwi, 2020).

Langkah-langkah membangun algoritma CART, sebagai berikut (Susanto dan Suryadi, 2010):

- a. Menyusun calon cabang dari semua atribut prediktor dengan lengkap. Daftar yang berisi calon cabang disebut daftar calon cabang muktahir.
- b. Menilai kinerja dari seluruhan calon cabang yang terdapat pada daftar calon cabang muktahir dengan menghitung nilai besaran kesesuaian $\varphi(s|t)$.
- c. Menentukan cabang dengan memilih calon cabang yang mempunyai nilai kesesuaian $\varphi(s|t)$ terbesar. Kemudian menggambarkan percabangan. Jika sudah tidak ada simpul keputusan yang tersisa, maka pembangunan algoritma CART akan berhenti. Namun, jika masih ada simpul keputusan yang tersisa, pembangunan algoritma akan dilanjutkan dengan kembali pada langkah kedua. Disini, calon cabang yang telah berhasil menjadi cabang akan dikeluarkan terlebih dahulu, kemudian dilanjutkan hingga mendapatkan daftar calon cabang muktahir yang baru.

Digunakan ukuran “*goodness*” $\varphi(s|t)$ dari calon cabang s pada *node* t , untuk mencari nilai besaran kesesuaian $\varphi(s|t)$ sebagaimana persamaan-persamaan berikut (Muflikhah dkk, 2018) (Larose, 2005):

$$Q(s|t) = \sum_{j=1}^{\neq \text{classes}} |P(j|t_L) - P(j|t_R)| \quad (2.5)$$

sehingga

$$\varphi(s|t) = 2P_L P_R Q(s|t) \quad (2.6)$$

dengan:

$$\begin{aligned} t_L &= \text{node anak kiri dari node } t \\ t_R &= \text{node anak kanan dari node } t \\ P_L &= \frac{\text{jumlah record pada } t_L}{\text{jumlah record dalam data training}} \\ P_R &= \frac{\text{jumlah record dalam data training}}{\text{jumlah record pada } t_R} \\ P(j|t_L) &= \frac{\text{jumlah record pada kelas } j \text{ dalam } t_L}{\text{jumlah record pada node } t} \\ P(j|t_R) &= \frac{\text{jumlah record pada kelas } j \text{ dalam } t_R}{\text{jumlah record pada node } t} \end{aligned}$$

Sebagai contoh perhitungan untuk mencari nilai besaran kesesuaian $\varphi(s|t)$ yaitu menggunakan dataset pada Tabel 2.2. Terlebih dahulu membagi calon cabang kiri dan calon cabang kanan, sebagai berikut:

Tabel 2.3. Contoh Calon Cabang

Candidat Split	Cabang Kiri	Cabang Kanan
1	Umur= Muda	Umur= Dewasa, Tua
2	Umur= Dewasa	Umur= Muda, Tua
3	Umur= Tua	Umur= Muda, Dewasa
4	Pendapatan= Rendah	Pendapatan= Sedang, Tinggi
5	Pendapatan= Sedang	Pendapatan= Rendah, Tinggi
6	Pendapatan= Tinggi	Pendapatan= Rendah, Sedang
7	Pengeluaran= Rendah	Pengeluaran= Sedang, Tinggi
8	Pengeluaran= Sedang	Pengeluaran= Rendah, Tinggi
9	Pengeluaran= Tinggi	Pengeluaran= Rendah, Sedang

1. Menghitung P_L dan P_R untuk *candidat split* ke-1

$$P_L = \frac{4}{10} = 0,4$$

$$P_R = \frac{6}{10} = 0,6$$

Lakukan hal yang sama untuk *candidat split* yang lainnya.

2. Menghitung $P(j|t_L)$ dan $P(j|t_R)$ untuk *candidat split* ke-1

$$P(j|t_L)(Ya) = \frac{2}{4} = 0,5$$

$$P(j|t_L)(Tidak) = \frac{2}{4} = 0,5$$

$$P(j|t_R)(Ya) = \frac{4}{6} = 0,67$$

$$P(j|t_R)(Tidak) = \frac{2}{6} = 0,33$$

Lakukan hal yang sama untuk *candidat split* yang lainnya.

3. Menghitung $Q(s|t)$ untuk *candidat split* ke-1

$$Q(s|t) = |P(j|t_L)(Ya) - P(j|t_L)(Tidak)| + |P(j|t_R)(Ya) -$$

$$P(j|t_L)(Tidak)|$$

$$Q(s|t) = |0,5 - 0,5| + |0,67 - 0,33| = 0,33$$

Lakukan hal yang sama untuk *candidate split* yang lainnya.

4. Menghitung $\varphi(s|t)$ untuk *candidate split* ke-1

$$\varphi(s|t) = 2P_L P_R Q(s|t)$$

$$\varphi(s|t) = 2(0,4)(0,6)(0,33) = 0,16$$

Lakukan hal yang sama untuk *candidate split* yang lainnya.

2.1.7. Perbedaan Algoritma ID3, C4.5, C5.0 dan CART

Karakteristik dasar dari keempat algoritma tersebut dijelaskan pada Tabel 2.4 dibawah ini (Kumar dan Kiruthika, 2015) (Rachmawati dan Ardani, 2021) (Maung, 2020).

Tabel 2.4. Perbandingan Algoritma ID3, C4.5, C5.0 dan CART

Algoritma	ID3	C4.5	C5.0	CART
Jenis Data	Kategorik	Kategorik Kontinu	Kategorik Kontinu Tanggal Waktu	Kontinu Nominal
Kecepatan	Rendah	Lebih cepat dari ID3	Paling tinggi	Rata-rata
Pruning	Tidak	Pre peruning	Pre peruning	Post peruning
Boosting	Tidak didukung	Tidak didukung	Didukung	Didukung
Rumus	Information gain dan entropy	Split info dan gain ratio	Split info dan gain ratio	Gini index /Ukuran goodness

2.1.8. Data *Training* dan Data *Testing*

Menurut Prasetyo (2014), data *Training* adalah data atau vektor yang telah diketahui sebelumnya untuk label kelas dan digunakan untuk membangun model klasifikator. Sedangkan data *Testing* adalah data atau vektor dengan label kelas yang tidak diketahui (dianggap tidak diketahui), label kelas diprediksi menggunakan model klasifikasi yang dibangun.

Model klasifikasi dikembangkan menggunakan data *Training* dan evaluasinya dilakukan dengan menggunakan data *Testing*. Biasanya, pembagian proporsi antara data *Training* dan data *Testing* adalah diskrit, misalnya 50:50 (50% data *training* dan 50% data *testing*), 60:40 (60% data *training* dan 40% data *testing*), 70:30 (70% data *training* dan 30% data *testing*), 80:20 (80% data *training* dan 20% data *testing*), serta 90:10 (90% data *training* dan 10% data *testing*). Menentukan jumlah data *training* dan data *testing* dapat menggunakan persamaan (2.7) dan (2.8) dengan N merupakan total jumlah dari data sampel (Pratiwi, 2020).

$$\text{Jumlah data training} = \text{proporsi data training} \times N \quad (2.7)$$

$$\text{Jumlah data testing} = N - \text{jumlah data training} \quad (2.8)$$

2.1.9. Mengukur Ketepatan Klasifikasi

Evaluasi model adalah langkah penting dalam pengembangan model pada proses klasifikasi untuk memperoleh pemahaman yang mendalam tentang seberapa baik model tersebut dalam memprediksi data baru. Metode yang digunakan untuk evaluasi model dalam penelitian ini adalah *confusion matrix* dan nilai

kurva AUC-ROC. *Confusion matrix* adalah alat yang berguna untuk mengukur kinerja dengan membandingkan kategori kelas yang sesungguhnya dengan hasil prediksi yang dikeluarkan oleh model. Sedangkan kurva AUC-ROC merupakan metrik evaluasi kinerja yang digunakan untuk memeriksa seberapa baik model klasifikasi bekerja dalam ambang batas.

2.1.10. *Confusion Matrix*

Confusion matrix adalah alat yang sangat berguna untuk menganalisis atau membandingkan seberapa baik pengklasifikasian *bias* mengenali *tuple* dari *class* yang berbeda. *Confusion matrix* adalah tabel matrik yang terbagi menjadi dua kelas, yaitu kelas yang diidentifikasi sebagai positif dan kelas yang diidentifikasi sebagai negatif. Evaluasi menggunakan *Confusion matrix* menghasilkan nilai *accuracy*, *precision*, *recall* (*sensitivity*) dan *specificity* (Fitriana, 2020). Tabel 2.5 merupakan format umum dari *Confusion matrix*. Tabel 2.5 Terdiri dari beberapa *class*, dimana setiap *class* berisi angka yang menunjukkan beberapa kasus aktual dari *class* yang digunakan untuk prediksi (Han dan Kember, 2011).

Tabel 2.5. *Confusion matrix*

<i>Actual Class</i>	<i>Predicted Class</i>	
	<i>C1</i>	<i>–C1</i>
<i>C1</i>	TP (<i>true positive</i>)	FN (<i>false negative</i>)
<i>–C1</i>	FP (<i>false positive</i>)	TN (<i>true negative</i>)

Actual class adalah kelas yang sebenarnya dalam *test set*. *Predicted class* adalah kelas hasil prediksi dari model yang dihasilkan oleh klasifikator. Menurut Pramana dkk (2018), *True*

Positive (TP) adalah jumlah baris yang sebenarnya memiliki label kelas *C1* sebagai klasifikator. *False Positive* (FP) adalah jumlah baris yang sebenarnya memiliki label kelas *C1* pada *test set*, namun diklasifikasikan bukan sebagai kelas *C1* oleh klasifikator. *False Positive* (FP) adalah jumlah baris yang sebenarnya memiliki label kelas bukan *C1* pada *test set*, namun diklasifikasikan sebagai kelas *C1* oleh klasifikator. *True Negative* (TN) adalah jumlah baris yang sebenarnya memiliki label bukan kelas *C1* pada *test set* dan benar diklasifikasikan sebagai kelas bukan *C1* oleh klasifikator (Pratiwi, 2020).

Menurut Han dan Kember (2011), rumus untuk mencari *accuracy*, *sensitivity*, *specificity*, dan *precision* adalah sebagai berikut:

$$Accuracy = \frac{TP + TN}{P + N} \quad (2.9)$$

$$Sensitivity = \frac{TP}{P} \quad (2.10)$$

$$Specificity = \frac{TN}{N} \quad (2.11)$$

$$Precision = \frac{TP}{TP + FP} \quad (2.12)$$

Evaluasi menggunakan kurva ROC (*Receiver Operating Character Curve*), secara umum menggambarkan grafik dua dimensi, dimana letak dari garis sumbu Y adalah tingkat *True Positive* (TP), sedangkan letak dari garis sumbu X adalah tingkat *False Positive* (FP), dengan ini kurva ROC menggambarkan *tradeoff* antara TP dan FP. Pencatatan di ROC dinyatakan dalam sebuah klausa artinya, semakin rendah titik ke kiri (0.0), maka klasifikasi

prediksi semakin dekat/negatif, sedangkan semakin tinggi titik kekanan (1.1), maka klasifikasi prediksi semakin dekat/positif. Titik dengan nilai 1 ditetapkan sebagai tingkat *True Positive* (TP), sedangkan titik dengan nilai 0 ditetapkan sebagai tingkat *False Positive* (FP). Pada titik (0.1) seluruh kasus baik positif maupun negatif dinyatakan dengan benar (*True*) sehingga klasifikasi prediksinya sempurna. Sebaliknya untuk (0.1) klasifikasi prediksi semuanya dinyatakan tidak benar (*False*) (Bisri dan Romi, 2015).

Kurva AUC-ROC merupakan metrik evaluasi kinerja yang digunakan untuk memeriksa seberapa baik model klasifikasi bekerja pada nilai ambang batas. ROC adalah grafik yang menggambarkan hubungan antara *sensitivity* dan *specificity*, sedangkan AUC menunjukkan seberapa baik model dapat memisahkan kelas-kelas yang berbeda. Semakin tinggi AUC, semakin baik model dalam memprediksi. Perhitungan AUC didefinisikan sebagai berikut (Sari dkk, 2020):

$$AUC = \frac{1}{2} \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right) \quad (2.13)$$

Performance klasifikasi keakuratan dengan nilai AUC dibagi menjadi beberapa kelompok (Gorunescu, 2011):

1. 0,90 - 1,00 = Klasifikasi sangat baik
2. 0,80 - 0,90 = Klasifikasi baik
3. 0,70 - 0,80 = Klasifikasi cukup
4. 0,60 - 0,70 = Klasifikasi buruk
5. 0,50 - 0,60 = Kegagalan

2.1.11. SMOTE (*Synthetic Minority Oversampling Technique*)

Ketidakseimbangan data terjadi ketika satu kelas data memiliki lebih banyak objek secara signifikan dibandingkan kelas lainnya. Kelas dengan jumlah objeknya terbanyak disebut kelas mayor, dan kelas dengan jumlah objeknya sedikit disebut kelas minor. Ketidakseimbangan ini dapat berdampak signifikan pada hasil model. Hal ini dikarenakan ketika digunakan tanpa penyesuaian, algoritma cenderung fokus pada kelas mayor dan kurang fokus pada kelas minor (Azmatul dkk, 2013).

Pada tahun 2002, Nithes V. Chawla memperkenalkan metode SMOTE sebagai salah satu alternative mengatasi ketidakseimbangan data dengan pendekatan yang berbeda dari teknik oversampling yang sudah ada sebelumnya. Konsep metode oversampling adalah menambahkan observasi secara acak, sedangkan pendekatan SMOTE adalah membuat data sintetik atau buatan untuk menambah jumlah data pada kelas minor sehingga dapat dibandingkan dengan kelas mayor (Sofyan dan Prasetyo, 2021).

2.2. Kajian Pustaka

Dalam penyusunan penelitian ini, penulis telah melakukan analisis terhadap penelitian-penelitian sebelumnya sebagai bahan perbandingan dan mengkaji karya-karya ilmiah yang mendukung dalam penelitian tugas akhir. Berikut beberapa contoh penelitian yang telah dilakukan:

1. Herman Santoso Pakpahan, Fenny Indar, dan Masna Wati (2018) melakukan penelitian yang berjudul “Penerapan Algoritma *CART Decision Tree* Pada Penentuan Penerima

Program Bantuan Pemerintah Daerah Kabupaten Kutai Kartanegara". Penelitian ini dilakukan untuk memudahkan pengelompokan calon penerima bantuan dan mengantisipasi dan mencegah kesalahan dalam menyalurkan dana bantuan. Menggunakan lima model percobaan untuk memperoleh hasil yang diinginkan dengan model percobaan yaitu 55%, 65%, 75%, 85%, dan 95%. Saat dilakukan model percobaan dari 55%, 65%, 75%, 85% tingkat akurasi mengalami kenaikan, namun ketika berada pada 95% data *training*, mengalami penurunan tingkat akurasi menjadi 97,35%. Maka dipilih model percobaan pada rasio data *training* sebesar 85% dengan 105 data memperoleh tingkat akurasi hasil klasifikasi sebesar 98,18%.

2. Kailash D. Kharat, dan Anil V. Turukmane (2014) melakukan penelitian yang berjudul "*Classification of Brain Tumor By Feature Vektor Using CART and C5.0 Algorithms*". Penelitian ini dilakukan untuk memperoleh informasi yang berguna untuk pengambilan keputusan klasifikasi tumor otak, tingkat tumor, dan perilaku tumor. Bukti empiris menunjukkan bahwa metode ini sangat cepat dan sangat berhasil, mengingat sifat dari kedua metode klasifikasi yang sangat berbeda. Tetapi tidak memberikan informasi manakah algoritma yang lebih baik.
3. Pratiwi, R., M. Nor Hayati, & Surya P (2020) melakukan penelitian yang berjudul "Klasifikasi Algoritma C5.0 dan Classification and Regression Tree (Studi Kasus: Data Sosial Kepala Keluarga Masyarakat Teluk Baru Kecamatan

muara Anclong Tahun 2019)". Penelitian ini dilakukan untuk mendapatkan ketepatan hasil klasifikasi rata-rata pendapatan masyarakat Desa Teluk Baru Kecamatan Muara Anclong menggunakan metode C5.0 dan CART. Hasil pengklasifikasian menunjukkan bahwa metode CART lebih baik dalam mengklasifikasikan rata-rata pendapat masyarakat Desa Teluk Baru Kecamatan Muara Anclong tahun 2019 dibandingkan dengan metode C5.0.

4. Ariska Kurnia R., Minhayati S., M. Nur Ramdani (2021) melakukan penelitian yang berjudul "*Comparison and Prediction of Data Mining Models to Determine the Classification of Family Planning Program User Status*". Penelitian ini dilakukan untuk membandingkan empat algoritma klasifikasi data mining yang digunakan yaitu *Decision Tree (C4.5)*, *Naïve Bayes*, *Logistic Regression* dan *Gradient Boosted Trees*. Keempat algoritma ini digunakan untuk mengklasifikasikan status pengguna KB di Kelurahan Mangunharjo Kota Semarang. Hasil nilai akurasi algoritma *decision tree* lebih tinggi dibandingkan dengan ketiga algoritma lainnya.

Beberapa penelitian tersebut dilakukan untuk membandingkan algoritma manakah yang mempunyai tingkat akurasi tertinggi dengan menerapkannya diberbagai macam kasus. Sehubungan dengan hal tersebut peneliti akan menggunakan metode klasifikasi *decision tree* dengan menggunakan algoritma C5.0 dan Algoritma CART digunakan untuk melihat perbandingan hasil akurasi dan mengetahui model klasifikasi pada kasus penerimaan bantuan PKH. Karena melihat masih banyak bantuan PKH yang

tidak tepat sasaran dan belum banyak yang melakukan penelitian pada kasus penerimaan bantuan PKH dengan menggunakan algoritma C5.0 dan CART. Beberapa penelitian tersebut diharapkan akan membantu proses pengerjaan penelitian ini.

BAB 3

Metode Penelitian

3.1. Sumber Data

Data yang digunakan dalam penelitian ini menggunakan dua data yaitu data primer dan data sekunder. Data utama yang digunakan pada penelitian ini menggunakan data primer sedangkan data sekunder digunakan sebagai data pendukung atau pelengkap dari data primer. Sumber data primer dan data sekunder yaitu :

1. Data Primer

Sumber data primer dalam penelitian ini diperoleh dengan membagikan kuesioner kepada penerima bantuan PKH tahap 1 periode bulan Januari - Maret Desa Sembungharjo Kecamatan Pulokulon Kabupaten Grobogan Tahun 2023.

2. Data Sekunder

Sumber data sekunder dalam penelitian ini diperoleh dari Dinas Kabupaten Grobogan melalui petugas pendamping PKH.

3.2. Waktu dan Tempat Penelitian

Waktu yang digunakan peneliti untuk melaksanakan penelitian adalah selama 1 bulan yaitu mulai tanggal 1 Desember - 31 Desember 2023. Tempat penelitian ini dilaksanakan di Desa Sembungharjo Kecamatan Pulokulon Kabupaten Grobogan.

3.3. Populasi dan Sampel

3.3.1. Populasi

Populasi dalam penelitian ini adalah penerima bantuan PKH tahap 1 Desa Sembungharjo Kecamatan Pulokulon Kabupaten Grobogan Tahun 2023. Data tahap 1 digunakan sebagai data calon penerima PKH ditahap 4 (akhir tahun 2023). Berdasarkan data yang telah diperoleh dari petugas pendamping PKH tahap 1 periode bulan Januari - Maret sebanyak 565 penerima. Sedangkan pada tahap 4 periode bulan Oktober - November mengalami penurunan penerima bantuan PKH sebanyak 520 penerima.

3.3.2. Sampel

Sampel adalah bagian dari jumlah serta karakteristik yang dimiliki oleh populasi tersebut. Sehingga sampel harus betul-betul representatif (mewakili) populasinya (Sugiyono, 2014). Metode pengambilan sampel (teknik *sampling*) dalam penelitian ini adalah menggunakan *probability sampling* dengan menerapkan teknik *simple random sampling*. *simple random sampling* adalah sebuah proses pengambilan sampel secara acak dari keseluruhan populasi tanpa mempertimbangkan strata yang terdapat pada populasi (Sugiyono, 2018).

Sampel dalam penelitian ini adalah sebagian penerima bantuan PKH tahap 1 pada periode bulan Januari - Maret 2023. Jumlah yang digunakan untuk pengambilan sampel dalam penelitian ini menggunakan metode rumus slovin, untuk rumusnya sebagai berikut (Siregar, 2015):

$$n = \frac{N}{1 + N(e)^2} \quad (3.1)$$

Dimana:

n = ukuran sampel/jumlah responden

N = ukuran Populasi

e = erorr (tingkat kesalahan).

Pada penelitian ini menggunakan error (e) sebesar 5%, sehingga untuk perhitungannya:

$$n = \frac{565}{1 + 565(5\%)^2}$$

$$n = \frac{565}{1 + 565(0,005)}$$

$$n = \frac{565}{2,4125} = 234,1968/234 \text{ (dibulatkan)}$$

Sehingga sampel yang akan digunakan minimal sebanyak 234 sampel. Dalam pembagian keusioner pada responden diperoleh sampel sebanyak 236 sampel, maka dari 236 sampel yang diperoleh digunakan untuk pengujian.

3.4. Teknik Pengumpulan Data

Teknik pengumpulan data adalah tahapan penting dalam sebuah penelitian. Hal ini dimaksud untuk memperoleh data yang valid dari atribut-atribut yang akan diteliti dalam suatu penelitian. Berikut beberapa teknik pengumpulan data dalam penelitian ini:

1. Wawancara

Menurut Sugiyono (2014), wawancara dipilih sebagai metode pengumpulan informasi saat peneliti ingin studi pendahuluan terhadap permasalahan yang akan diteliti. Wawancara dalam penelitian ini adalah dimana peneliti menggunakan pedoman wawancara terstruktur untuk mengumpulkan informasi tentang alur penerimaan bantuan PKH, target sasaran PKH, hingga apa saja permasalahan yang dihadapi petugas pendamping PKH dalam proses penerimaan bantuan PKH. Wawancara dilakukan dengan 2 petugas pendamping PKH yang kemudian peneliti akan merekam selama proses wawancara berlangsung sebagai bukti penelitian.

2. Dokumentasi

Menurut Sugiyono (2014), dokumen adalah catatan peristiwa yang sudah terjadi sebelumnya atau peristiwa yang sudah berlalu. Adapun dokumen dalam penelitian ini adalah dataset penerima bantuan PKH tahap 1 pada periode bulan Januari - Maret 2023 dan Penerima bantuan PKH tahap 4 periode bulan Oktober - November.

3. Kuesioner

kuesioner dalam penelitian ini adalah berupa pertanyaan singkat yang berkaitan dengan informasi pribadi untuk memperoleh informasi mengenai atribut-atribut yang digunakan dalam penelitian ini.

3.5. Atribut Penelitian

Atribut yang digunakan dalam penelitian ini meliputi:

1. Atribut terikat (Y) yang digunakan yaitu status (diterima atau ditolak). Dikategorikan sebagai berikut:

Y = 1, untuk kategori diterima

Y = 2, untuk kategori ditolak

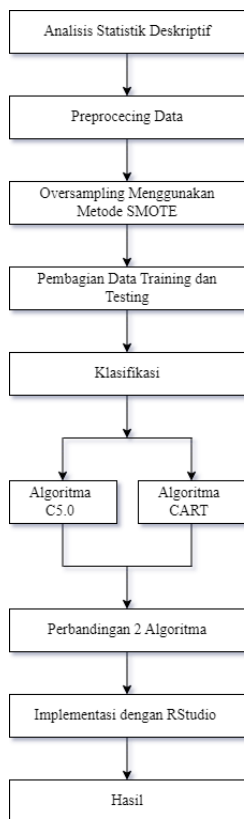
2. Atribut bebas (X) dikategorikan sebagai berikut:

Tabel 3.1. Atribut Bebas

No	Atribut		Tipe Data	Kategori
1.	USIA	X1	Numerik	
2.	PENDIDIKAN	X2	Ordinal	1 : SMA 2 : SMP 3 : SD 4 : Tidak Sekolah
3.	PEKERJAAN	X3	Nominal	1 : Lainnya 2 : Buruh Harian 3 : Pedagang 4 : Petani 5 : Tidak Bekerja
4.	PENGHASILAN	X4	Nominal	1 : > 2.000.000 2 : 1.000.000 - 2.000.000 3 : < 1.000.000
5.	ASET	X5	Nominal	1 : Ya 2 : Tidak
6.	LUAS LANTAI	X6	Nominal	1 : > 100m ² 2 : 50m ² - 100m ² 3 : < 50m ²
7.	JENIS LANTAI	X7	Nominal	1 : Keramik 2 : Semen 3 : Tanah
8.	DINDNG	X8	Nominal	1 : Bambu 2 : Kayu 3 : Tembok
9.	PENERANGAN	X9	Nominal	1 : > 450 Watt 2 : 450 Watt 3: Bukan Listrik Pribadi

3.6. Teknik Analisis Data

Metode analisis data yang digunakan dalam penelitian ini adalah menggunakan metode algoritma C5.0 dan algoritma CART dalam mengklasifikasi penerimaan program keluarga harapan. Menggunakan *Software* komputer yaitu *Microsoft Excel 2013* dan *Software* RStudio. Langkah-langkah analisis data dalam penelitian ini, sebagai berikut:



Gambar 3.1. Diagram Alir Analisis Penelitian

3.6.1. *preprocessing* Data

preprocessing adalah langkah pertama dalam proses KDD yaitu membersihkan data dari *noise* seperti *missing value*, *redundant* data, dan *inkonsisten* data. Semua atribut dipilih dengan cermat untuk memastikan hanya atribut yang relevan dan tidak mengandung *missing value* atau *redundant* yang dipertahankan. Atribut yang memiliki nilai kosong akan dihapus atau dihilangkan (Rachmawati dan Ardani, 2021).

Data yang akan digunakan untuk *preprocessing* dalam penelitian ini adalah data Responden penerima bantuan PKH tahap 1 Desa Sembungharjo Kecamatan Pulokulon Kabupaten Grobogan tahun 2023 yang berjumlah 236 sampel data. Dengan mengambil atribut usia, pendidikan, pekerjaan, penghasilan, aset, luas lantai, jenis lantai, dinding, dan penerangan. Proses untuk *preprocessing* data dilakukan dengan bantuan *software* RStudio, untuk *script software* RStudio tercantum di Lampiran 4.

3.6.2. Oversampling Menggunakan SMOTE

Oversampling menggunakan metode SMOTE dilakukan dengan menggunakan bantuan *Software* RStudio untuk *script* RStudionya sebagai berikut:

```
#Oversampling dengan metode SMOTE
library(smotefamily)
#Mengubah tipe data character menjadi factor (tranformasi data)
data$PENDIDIKAN <- as.factor(data$PENDIDIKAN)
data$PEKERJAAN <- as.factor(data$PEKERJAAN)
data$PENGHASILAN <- as.factor(data$PENGHASILAN)
data$ASET <- as.factor(data$ASET)
```

```

data$LUAS.LANTAI <- as.factor(data$LUAS.LANTAI)
data$JENIS.LANTAI <- as.factor(data$JENIS.LANTAI)
data$DINDING <- as.factor(data$DINDING)
data$PENERANGAN <- as.factor(data$PENERANGAN)
data$STATUS <- as.factor(data$STATUS)
str(data)
table(data$STATUS)
prop.table(table(data$STATUS))
target_variabel <- "STATUS"
predictor_variables <- setdiff(names(data), target_variabel)
table(data$STATUS)
barplot(table(data$STATUS))
#Mengubah tipe data factor menjadi numerik
data$PENDIDIKAN <- as.numeric(data$PENDIDIKAN)
data$PEKERJAAN <- as.numeric(data$PEKERJAAN)
data$PENGHASILAN <- as.numeric(data$PENGHASILAN)
data$ASET <- as.numeric(data$ASET)
data$LUAS.LANTAI <- as.numeric(data$LUAS.LANTAI)
data$JENIS.LANTAI <- as.numeric(data$JENIS.LANTAI)
data$DINDING <- as.numeric(data$DINDING)
data$PENERANGAN <- as.numeric(data$PENERANGAN)
data$STATUS <- as.numeric(data$STATUS)
smote=SMOTE(data[,-10],data$STATUS)
smote=smote$data
table(smote$class)
prop.table(table(smote$class))
barplot(table(smote$class))
View(smote)

```

3.6.3. Algoritma C5.0

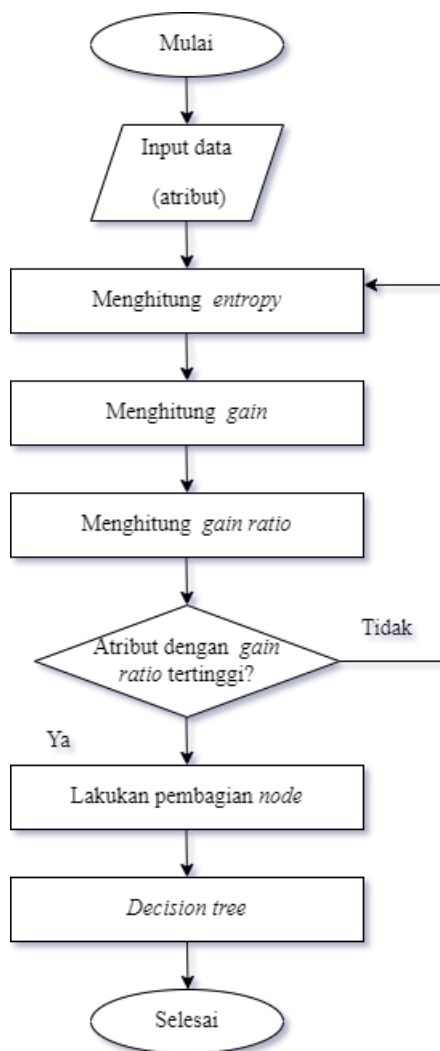
Algorithm 3 C5.0 Decision Tree

Input:

- a. Sekumpulan data atribut pada data *training*
- b. Metode seleksi atribut, suatu prosedur untuk menentukan kriteria pemisahan mempartisi data *tuple* ke alam kelas-kelas tertentu. Kriteria ini terdiri dari *splitting attribute* dan *splitting subset*.

Output: C5.0 decision tree**Method:**

1. Menentukan atribut
 2. Pemilihan *root node* diawali dengan mencari nilai *Entropy* menggunakan persamaan (2.1). Selanjutnya, menghitung nilai *Gain* menggunakan persamaan (2.2). Setelah itu, menghitung nilai *Gain Ratio* menggunakan persamaan (2.4)
 3. Jika perhitungan selesai maka diperoleh nilai *Gain Ratio* tertinggi. Atribut yang memiliki *Gain Ratio* tertinggi akan dipilih sebagai *root node*
 4. Bentuk *node* awal yang mengandung atribut dengan *Gain Ratio* tertinggi
 5. Ulangi langkah kedua sampai semua data telah termasuk dalam kelas yang sama. Atribut yang dipilih tidak diikuti lagi dalam perhitungan pada iterasi selanjutnya.
-



Gambar 3.2. Diagram Alir Algoritma C5.0 (Pratiwi, 2020)

3.6.4. Algoritma CART

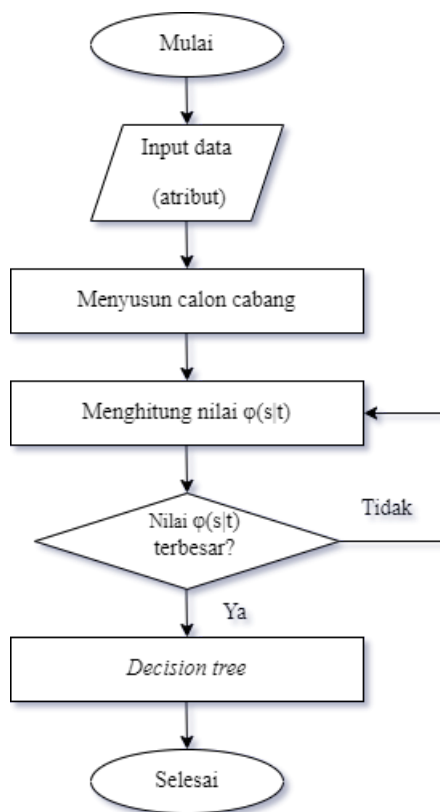
Algorithm 4 CART Decision Tree

Input:

- a. Sekumpulan data atribut pada data *training*
- b. Metode seleksi atribut, suatu prosedur untuk menentukan kriteria pemisahan data pada setiap *node* dalam pohon keputusan. Dengan kriteria berupa besaran kesesuaian $\varphi(s|t)$.

Output: CART decision tree**Method:**

1. Menentukan atribut
 2. Menyusun calon cabang muktahir
 3. Menghitung nilai besaran kesesuaian $\varphi(s|t)$ untuk calon cabang menggunakan persamaan (2.6)
 4. Jika perhitungan selesai maka diperoleh nilai kesesuaian $\varphi(s|t)$ terbesar. Dari nilai tersebut bisa digambarkan pohon sementara atau *decision tree*
 5. Jika sudah tidak ada simpul keputusan yang tersisa, maka pembangunan algoritma CART akan berhenti
 6. Jika masih ada simpul keputusan yang tersisa, pembangunan algoritma akan dilanjutkan dengan kembali kelangkah ketiga hingga menghasilkan *decision tree*. Calon cabang yang telah berhasil menjadi cabang tidak diikuti lagi dalam perhitungan pada iterasi selanjutnya.
-



Gambar 3.3. Diagram Alir Algoritma CART

3.6.5. Perbandingan 2 Algoritma

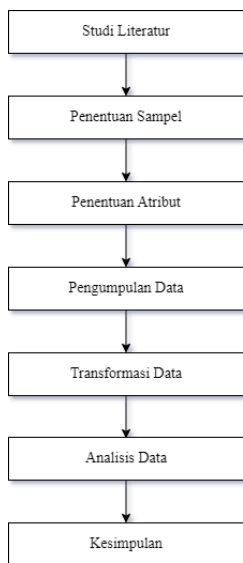
Setelah dilakukan perhitungan menggunakan algoritma C5.0 dan algoritma CART selanjutnya membandingkan kedua algoritma tersebut dengan menggunakan *confusion matrix* yaitu dengan melihat tingkat akurasi dan melihat nilai kurva AUC-ROC, untuk rumus akurasi sebagai berikut (Ramdani dkk, 2022):

$$Accuracy = \frac{TP + TN}{(TP + FP + TN + FN)} \times 100\% \quad (3.2)$$

Rumus perhitungan AUC sebagai berikut (Sari dkk, 2020):

$$AUC = \frac{1}{2} \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right) \quad (3.3)$$

3.7. Diagram Alur Penelitian



Gambar 3.4. Diagram Alur Penelitian

BAB 4

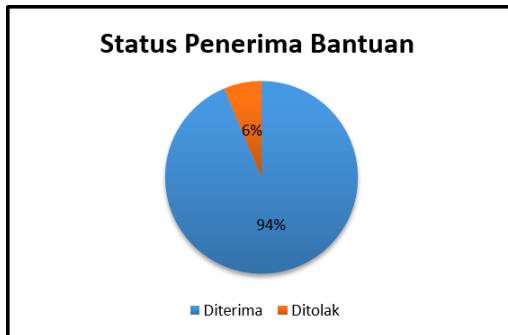
Hasil dan Pembahasan

4.1. Analisis Statistik Deskriptif

Analisis statistik deskriptif dilakukan untuk mengetahui karakteristik data yang akan diteliti. Bagian ini mendeskripsikan setiap atribut yang akan diteliti. Atribut yang digunakan dalam penelitian ini, sebagai berikut :

4.1.1. Atribut Status Penerima Bantuan

Iterpretasi data dari atribut Status tersaji pada gambar dibawah ini,

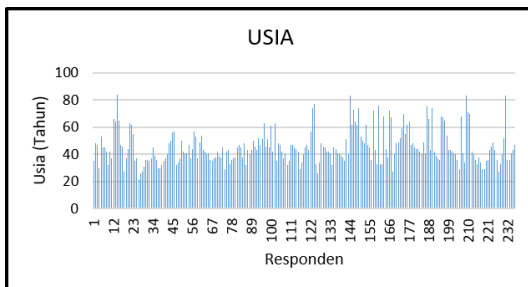


Gambar 4.1. Diagram Lingkaran untuk Atribut Status

Berdasarkan Gambar 4.1 diketahui bahwa status penerima bantuan PKH pada tahap 4 tahun 2023 Desa Sembungharjo, tercatat dari 236 sampel data terdapat 94% atau sebanyak 221 penerima bantuan yang diterima. Sedangkan, 6% atau 15 penerima bantuan yang ditolak.

4.1.2. Atribut Usia

Usia merupakan salah satu faktor yang mempengaruhi kriteria penerima bantuan PKH. Salah satunya yaitu usia ≥ 60 tahun keatas termasuk kriteria penerima bantuan PKH yaitu lanjut usia. Berikut adalah gambar usia penerima bantuan PKH tahap 1 dari 236 sampel,



Gambar 4.2. Atribut Usia

Gambar 4.2 memperlihatkan bahwa usia penerima bantuan PKH berkisar dari 22 hingga 84 tahun, dengan usia minimum 22 tahun dan usia maksimum 84 tahun. Rata-rata usia penerima bantuan PKH adalah 45 tahun. Dengan mengambil rata-rata usia tersebut yaitu 45 tahun. Maka, dengan menggunakan *Cross Tabulation* diperoleh hasil analisis deskriptif untuk atribut Usia berdasarkan status penerima bantuan sebagai berikut :

Tabel 4.1. Usia berdasarkan Status Penerima Bantuan

Status	Usia		Jumlah
	≤ 45	> 45	
Diterima	141	80	221
Ditolak	7	8	15
Jumlah	148	88	236

Berdasarkan Tabel 4.1 didapatkan informasi bahwa pada kategori usia ≤ 45 terdapat 141 orang dikategorikan diterima dan 7 orang dikategorikan ditolak. Sedangkan pada 88 data kategori usia 45 terdapat 80 orang dikategorikan diterima dan 8 orang dikategorikan ditolak.

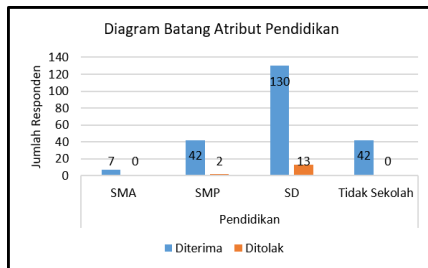
4.1.3. Atribut Pendidikan

Hasil analisis deskriptif atribut Pendidikan berdasarkan status penerima bantuan dengan menggunakan *Cross Tabulation* sebagai berikut :

Tabel 4.2. Pendidikan berdasarkan Status Penerima Bantuan

Status	Pendidikan				Jumlah
	SMA	SMP	SD	Tidak Sekolah	
Diterima	7	42	130	42	221
Ditolak	0	2	13	0	15
Jumlah	7	44	143	42	236

Berdasarkan Tabel 4.2 didapatkan informasi bahwa pendidikan terakhir penerima bantuan kebanyakan tingkat SD yaitu sebanyak 130 orang dikategorikan diterima dan 13 orang dikategorikan ditolak. Penerima bantuan PKH yang tidak sekolah dalam kategori diterima ada 42 orang dan 0 (nol) orang dikategorikan ditolak. Tingkat pendidikan SMP dikategorikan diterima sebanyak 42 orang dan 2 orang dikategorikan ditolak. Sedangkan Pada tingkat SMA kategori diterima bantuan PKH paling sedikit diantara tingkatan pendidikan yang lain yaitu 7 orang dan 0 (nol) dikategorikan ditolak. Data yang tercantum dalam Tabel 4.2 disajikan dalam bentuk diagram batang berikut:



Gambar 4.3. Diagram Batang untuk Atribut Pendidikan

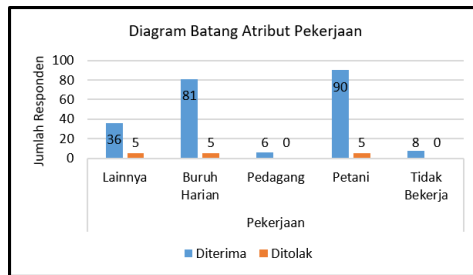
4.1.4. Atribut Pekerjaan

Hasil analisis deskriptif atribut Pekerjaan berdasarkan status penerima bantuan dengan menggunakan *Cross Tabulation* sebagai berikut :

Tabel 4.3. Pekerjaan berdasarkan Status Penerima Bantuan

Pekerjaan	Status		Jumlah
	Diterima	Ditolak	
Lainnya	36	5	41
Buruh Harian	81	5	86
Pedagang	6	0	6
Petani	90	5	95
Tidak Bekerja	8	0	8
Jumlah	221	15	236

Berdasarkan Tabel 4.3 didapatkan informasi bahwa pekerjaan penerima bantuan kebanyakan petani yaitu sebanyak 90 orang dikategorikan diterima dan 5 orang dikategorikan ditolak. Data yang tercantum dalam Tabel 4.3 disajikan dalam bentuk diagram batang berikut:



Gambar 4.4. Diagram Batang untuk Atribut Pekerjaan

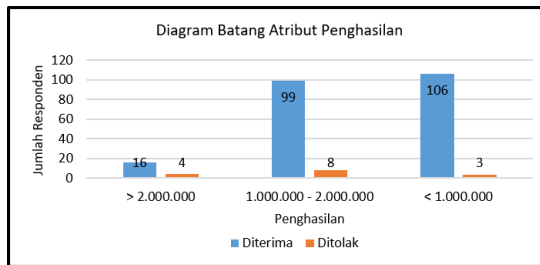
4.1.5. Atribut Penghasilan

Hasil analisis deskriptif atribut Penghasilan berdasarkan status penerima bantuan dengan menggunakan *Cross Tabulation* sebagai berikut :

Tabel 4.4. Penghasilan berdasarkan Status Penerima Bantuan

Penghasilan	Status		Jumlah
	Diterima	Ditolak	
>2.000.000	16	4	20
1.000.000 – 2.000.000	99	8	107
<1.000.000	106	3	109
Jumlah	221	15	236

Berdasarkan Tabel 4.4 didapatkan informasi bahwa penghasilan yang perbulan penerima bantuan kebanyakan adalah < 1.000.000 yaitu sebanyak 106 orang dikategorikan diterima dan 3 orang dikategorikan ditolak. Data yang tercantum dalam Tabel 4.4 disajikan dalam bentuk diagram batang berikut:



Gambar 4.5. Diagram Batang untuk Atribut Penghasilan

4.1.6. Atribut Aset

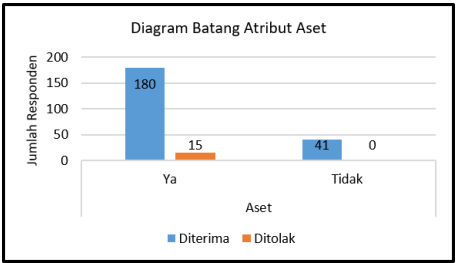
Aset merupakan tabungan yang dimiliki oleh calon penerima bantuan PKH. Badan pusat statistika menyebutkan seseorang yang layak untuk mendapatkan bantuan yaitu ketika calon penerima tidak memiliki aset (tabungan) atau mempunyai barang yang bisa dijual dengan minimal penjualan sebesar RP. 500.000,-. Hasil analisis deskriptif atribut Aset berdasarkan status penerima bantuan dengan menggunakan *Cross Tabulation* sebagai berikut :

Tabel 4.5. Aset berdasarkan Status Penerima Bantuan

Status	Aset		Jumlah
	Ya	Tidak	
Diterima	180	41	221
Ditolak	15	0	15
Jumlah	195	41	236

Berdasarkan Tabel 4.5 didapatkan informasi bahwa kebanyakan penerima bantuan memiliki aset atau tabungan sebanyak 180 orang dikategorikan diterima dan 15 orang dikategorikan ditolak. Sedangkan yang tidak memiliki aset ada 41 orang dikategorikan diterima dan 0 (nol) orang dikategorikan

ditolak. Data yang tercantum dalam Tabel 4.5 disajikan dalam bentuk diagram batang berikut:



Gambar 4.6. Diagram Batang untuk Atribut Aset

4.1.7. Atribut Luas Lantai

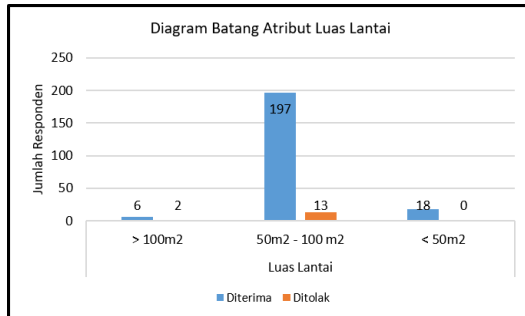
Hasil analisis deskriptif atribut Luas Lantai berdasarkan status penerima bantuan dengan menggunakan *Cross Tabulation* sebagai berikut :

Tabel 4.6. Luas Lantai berdasarkan Status Penerima Bantuan

Status	Luas Lantai			Jumlah
	$> 100m^2$	$50m^2 - 100m^2$	$< 50m^2$	
Diterima	6	197	18	221
Ditolak	2	13	0	15
Jumlah	8	210	18	236

Berdasarkan Tabel 4.6 didapatkan informasi bahwa luas lantai yang dimiliki penerima bantuan kebanyakan adalah $50m^2 - 100m^2$ yaitu sebanyak 197 orang dikategorikan diterima dan 13 orang dikategorikan ditolak. Sedangkan Luas lantai penerima bantuan paling sedikit yaitu $> 100m^2$ dalam kategori diterima ada 6 orang dan 2 orang dikategorikan ditolak. Data yang tercantum dalam

Tabel 4.6 disajikan dalam bentuk diagram batang berikut:



Gambar 4.7. Diagram Batang untuk Atribut Luas Lantai

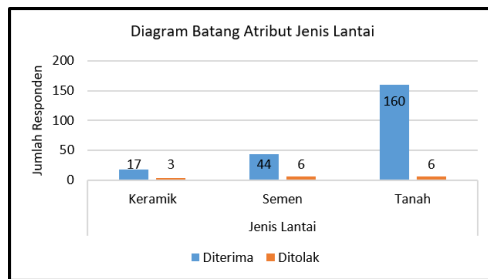
4.1.8. Atribut Jenis Lantai

Hasil analisis deskriptif atribut Jenis Lantai berdasarkan status penerima bantuan dengan menggunakan *Cross Tabulation* sebagai berikut :

Tabel 4.7. Jenis Lantai berdasarkan Status Penerima Bantuan

Status	Jenis Lantai			Jumlah
	Keramik	Semen	Tanah	
Diterima	17	44	160	221
Ditolak	3	6	6	15
Jumlah	20	50	166	236

Berdasarkan Tabel 4.7 didapatkan informasi bahwa jenis lantai penerima bantuan kebanyakan adalah tanah yaitu sebanyak 160 orang dikategorikan diterima dan 15 orang dikategorikan ditolak. Data yang tercantum dalam Tabel 4.7 disajikan dalam bentuk diagram batang berikut:



Gambar 4.8. Diagram Batang untuk Atribut Jenis Lantai

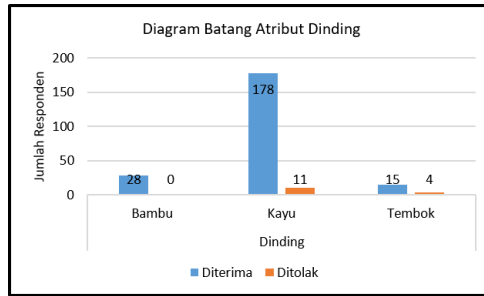
4.1.9. Atribut Dinding

Hasil analisis deskriptif atribut Dinding berdasarkan status penerima bantuan dengan menggunakan *Cross Tabulation* sebagai berikut :

Tabel 4.8. Dinding berdasarkan Status Penerima Bantuan

Status	Dinding			Jumlah
	Bambu	Kayu	Tembok	
Diterima	28	178	15	221
Ditolak	0	11	4	15
Jumlah	28	189	19	236

Berdasarkan Tabel 4.8 didapatkan informasi bahwa dinding yang dimiliki penerima bantuan kebanyakan adalah Kayu yaitu sebanyak 178 orang dikategorikan diterima dan 11 orang dikategorikan ditolak. Untuk dinding yang terbuat dari bambu ada sebanyak 28 orang yang dikategorikan diterima dan 0 (nol) orang dikategorikan ditolak. Sedangkan, untuk dinding yang terbuat dari tembok ada sebanyak 15 orang dikategorikan diterima dan 4 orang dikategorikan ditolak. Data yang tercantum dalam Tabel 4.8 disajikan dalam bentuk diagram batang berikut:



Gambar 4.9. Diagram Batang untuk Atribut Dinding

4.1.10. Atribut Penerangan

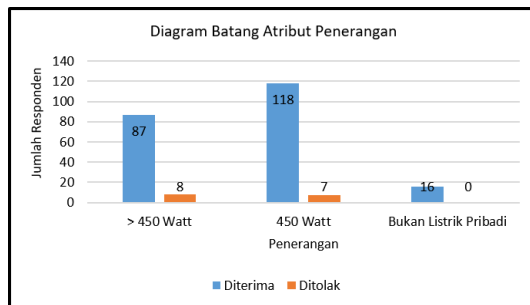
Hasil analisis deskriptif atribut Penerangan berdasarkan status penerima bantuan dengan menggunakan *Cross Tabulation* sebagai berikut :

Tabel 4.9. Penerangan berdasarkan Status Penerima Bantuan

Status	Penerangan			Jumlah
	>450 Watt	450 Watt	Bukan Listrik Pribadi	
Diterima	87	118	16	221
Ditolak	8	7	0	15
Jumlah	95	125	16	236

Berdasarkan Tabel 4.9 didapatkan informasi bahwa penerangan yang dimiliki penerima bantuan kebanyakan menggunakan listrik 450 Watt yaitu sebanyak 118 orang dikategorikan diterima dan 0 (nol) orang dikategorikan ditolak. Sedangkan masih ada penerima bantuan PKH yang tidak memiliki listrik pribadi, mereka menyalurkan listrik kerumahnya dengan cara ikut gabung ke listrik keluarganya. Terdapat 16 orang yang tidak memiliki listrik pribadi dengan kategori diterima dan 0 (nol)

orang diategorikan ditolak. Data yang tercantum dalam Tabel 4.9 disajikan dalam bentuk diagram batang berikut :



Gambar 4.10. Diagram Batang untuk Atribut Penerangan

4.2. Preprocecing data

Preprocecing data merupakan tahap awal dalam mengolah data yang bertujuan untuk menghilangkan *noise* atau inkonsistensi dalam data yang akan diolah. *Noise* yang biasanya ditemukan mencakup *missing value*, *redudant* data, dan *inkosisten* data. Semua atribut dipilih dengan cermat untuk memastikan hanya atribut yang relevan dan tidak mengandung *missing value* atau *redudant* yang dipertahankan. Atribut yang memiliki nilai kosong akan dihapus atau dihilangkan. Berikut ini *preprocecing* data menggunakan bantuan *software* RStudio,

4.2.1. Missing Value

Ketika bekerja dengan data, sering kali menemukan bahwa tidak semua nilai dalam setiap atribut tersedia dalam artian terdapat *missing value*. *Missing value* dapat disebabkan oleh berbagai alasan, seperti kesalahan pengukuran, ketidakhadiran

informasi, atau kegagalan dalam proses input data. Penting untuk mengidentifikasi dan menangani *missing value* dengan tepat karena mereka dapat mempengaruhi hasil analisis secara signifikan. Oleh sebab itu, sebelum melakukan analisis baiknya memeriksa terlebih dahulu apakah terdapat *missing value* atau tidak.

Pada data yang sudah diseleksi ternyata tidak terdapat *record* atau data yang tidak sempurna. Pada gambar dibawah ini menunjukkan bahwa semua atribut yang akan digunakan tidak menunjukkan adanya *missing value*,

```
> #Melihat missing value
> colsums(is.na(data)) #mengetahui jumlah missing value
```

USIA	PENDIDIKAN	PEKERJAAN	PENGHASILAN	ASET
0	0	0	0	0
LUAS.LANTAI	JENIS.LANTAI	DINDING	PENERANGAN	STATUS
0	0	0	0	0

Gambar 4.11. Output Missing Value

Sumber: Output RStudio, 2024

4.2.2. Redundant Data

Redundant data mengacu pada keberadaan informasi yang berlebihan atau tidak perlu dalam sebuah data, yang sering kali dapat memengaruhi performa model dan efisiensi analisis. Salah satu cara untuk mengidentifikasi data yang *redundant* adalah dengan melakukan analisis korelasi antar atribut. Korelasi mengukur sejauh mana dua atribut berhubungan satu sama lain. Jika dua atribut memiliki korelasi yang sangat tinggi, maka salah satunya dapat dianggap *redundant* karena keduanya menyimpan informasi yang serupa. Dengan demikian, dengan memeriksa analisis korelasi antar atribut, kita dapat mengidentifikasi dan menghilangkan atribut yang tidak memberikan kontribusi

tambahan terhadap pemahaman data atau pengambilan keputusan. Dengan pengujian koefisien korelasi antara dua atribut secara bersama-sama untuk mendeteksi apakah terdapat *redundant* dalam data, diperoleh output sebagai berikut :

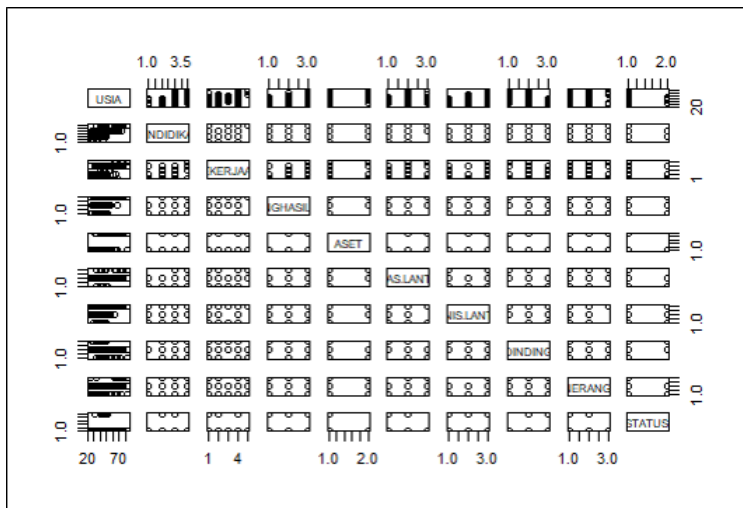
	USIA	PENDIDIKAN	PEKERJAAN	PENGHASILAN	
USIA	1.000000000	0.61773909	0.53033415	0.4522187	
PENDIDIKAN	0.617739092	1.000000000	0.41544236	0.4049172	
PEKERJAAN	0.530334147	0.41544236	1.000000000	0.4430797	
PENGHASILAN	0.452218663	0.40491722	0.44307969	1.000000000	
ASET	0.624749091	0.48149326	0.32365274	0.4139632	
LUAS. LANTAI	0.351760514	0.32895567	0.19105853	0.2475190	
JENIS. LANTAI	0.231131902	0.26840805	0.28034531	0.2400729	
DINDING	-0.197849973	-0.15995760	-0.24714472	-0.2488564	
PENERANGAN	0.324987959	0.25221228	0.20495930	0.2648566	
STATUS	-0.004452837	-0.02468391	-0.08927833	-0.1818115	
	ASET	LUAS. LANTAI	JENIS. LANTAI	DINDING	PENERANGAN
USIA	0.6247491	0.3517605	0.2311319	-0.1978500	0.32498796
PENDIDIKAN	0.4814933	0.3289557	0.2684081	-0.1599576	0.25221228
PEKERJAAN	0.3236527	0.1910585	0.2803453	-0.2471447	0.20495930
PENGHASILAN	0.4139632	0.2475190	0.2400729	-0.2488564	0.26485659
ASET	1.0000000	0.4165956	0.2219404	-0.2373546	0.29380275
LUAS. LANTAI	0.4165956	1.0000000	0.1579517	-0.2494956	0.15799692
JENIS. LANTAI	0.2219404	0.1579517	1.0000000	-0.3058101	0.24318129
DINDING	-0.2373546	-0.2494956	-0.3058101	1.0000000	-0.22309763
PENERANGAN	0.2938027	0.1579969	0.2431813	-0.2230976	1.00000000
STATUS	-0.1194604	-0.1390510	-0.1712949	0.1785938	-0.08643461
	STATUS				
USIA	-0.004452837				
PENDIDIKAN	-0.024683907				
PEKERJAAN	-0.089278329				
PENGHASILAN	-0.181811479				
ASET	-0.119460441				
LUAS. LANTAI	-0.139050964				
JENIS. LANTAI	-0.171294914				
DINDING	0.178593767				
PENERANGAN	-0.086434613				
STATUS	1.000000000				

Gambar 4.12. Output Koefisien Korelasi

Sumber: Output RStudio, 2024

Berdasarkan Gambar 4.12 adalah output yang dihasilkan oleh koefisien korelasi untuk setiap pasangan dua atribut pada data tersebut. Output tersebut menunjukkan bahwa koefisien korelasi antar pasangan dua atribut yang dilakukan secara bersama-sama tidak menunjukkan apakah mendekati -1 atau 1, jadi tidak terdapat *redundant* pada data tersebut untuk masing-masing atribut. Lebih jelasnya dapat dilihat dari *scatterplot* untuk mengetahui pola

koefisien korelasi dari setiap atribut, sebagai berikut :



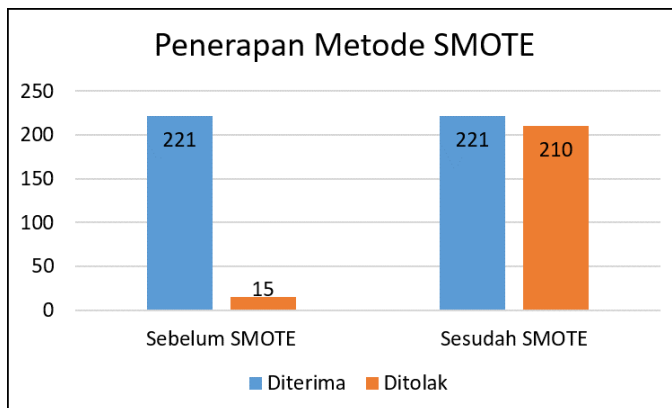
Gambar 4.13. *Scatterplot* Koefisien Korelasi

Sumber: Output RStudio, 2024

4.3. SMOTE (*Synthetic Minority Oversampling Technique*)

Metode SMOTE digunakan untuk mengatasi data yang tidak seimbang pada dataset yang digunakan, dimana data yang tidak seimbang ini dilihat dari status penerima bantuan. Proses penyeimbangan data merupakan langkah penting dalam analisis data, terutama dalam konteks klasifikasi atau prediksi di mana kelas-kelas yang berbeda memiliki distribusi yang tidak seimbang. Salah satu metode yang sering digunakan adalah dengan meningkatkan jumlah data pada kelas minoritas. Dalam penelitian ini diperoleh data sebanyak 236 responden. Tercatat dari 236 data, terdapat 221 data kategori diterima dan 15 data kategori ditolak, sehingga pada data tersebut sangatlah tidak

seimbang. Proses penyeimbangan data dilakukan dengan bantuan *software* RStudio dengan meningkatkan jumlah data minoritas. Dengan demikian data tersebut menjadi seimbang. Berikut Perbedaan data mayoritas dan minoritas sebelum dan sesudah dilakukan oversampling menggunakan metode SMOTE,



Gambar 4.14. Penerapan Metode SMOTE

Pada Gambar 4.14 diketahui bahwa sebelum dilakukan SMOTE, data pada kategori diterima dan ditolak sangatlah tidak seimbang dan sesudah dilakukan SMOTE data menjadi seimbang. Pada bagan diatas diperoleh informasi bahwa data pada kategori diterima yang awalnya 221 data setelah dilakukan SMOTE menjadi 221 data. Data pada kategori ditolak yang awalnya 15 data setelah dilakukan SMOTE menjadi 210 data. Tercatat dari data 236 menjadi 431 data yang sudah dioversampling menggunakan metode SMOTE. Sehingga dalam pengujian akan menggunakan data yang sudah seimbang atau data yang sudah dilakukan menggunakan metode SMOTE.

4.4. Pembagian Data *Training* dan Data *Testing*

Sebelum melakukan klasifikasi, langkah pertama adalah memisahkan dataset menjadi dua bagian yaitu data *training* dan data *testing*. Hal ini dilakukan untuk memastikan model yang dikembangkan dapat diuji secara akurat sebelum proses klasifikasi. Pada penelitian ini, pembagian presentase data *training* dan data *testing* menggunakan 5 macam proporsi pembagian yaitu (50%:50%), (60%:40%), (70%:30%), (80%:20%), dan (90%:10%), dimana pembagian data *training* dan data *testing* ini menggunakan data yang sudah di oversampling menggunakan metode SMOTE. Data yang sudah di oversampling menggunakan metode SMOTE diperoleh data sebanyak 431 sampel data. Selanjutnya membagi data tersebut menjadi dua bagian yaitu data *training* dan data *testing* dengan menggunakan 5 macam proporsi pembagian. Berikut adalah jumlah perhitungan untuk data *training* dan data *testing* yang telah dilakukan pengacakan menggunakan bantuan *software* RStudio dari 5 macam proporsi pembagian, diperoleh jumlah perhitungan sebagai berikut:

Tabel 4.10. Jumlah Data *Training* dan Data *Testing*

Proporsi	Training	Testing
50:50	216	215
60:40	259	172
70:30	302	129
80:20	345	86
90:10	388	43

Berdasarkan Tabel 4.10 didapatkan informasi bahwa jumlah dari data *training* dan data *testing* untuk proporsi 50:50 yaitu 216 menjadi data *training* dan 215 data menjadi data *testing*. Proporsi

60:40 yaitu 259 menjadi data *training* dan 172 data menjadi data *testing*. Proporsi 70:30 yaitu 302 menjadi data *training* dan 129 data menjadi data *testing*. Proporsi 80:20 yaitu 345 menjadi data *training* dan 86 data menjadi data *testing*. Sedangkan untuk Proporsi 90:10 yaitu 388 menjadi data *training* dan 43 data menjadi data *testing*. Berikut ini data *training* yang digunakan dalam proses pengujian secara manual dengan mengambil contoh proporsi data menggunakan proporsi (90%:10%), tersaji pada tabel sebagai berikut:

Tabel 4.11. Data *Training*

NO	USIA	PENDIDIKAN	PEKERJAAN	...	DINDING	PENERANGAN	class
1	52	3	4	...	2	2	1
2	43	3	1	...	3	1	2
3	57	3	4	...	1	2	1
4	41	3	4	...	2	2	1
5	45	1	2	...	2	2	1
6	34	3	2	...	2	1	1
7	49	3	4	...	2	1	1
8	50	3	4	...	2	2	2
9	41	3	2	...	2	2	2
10	41	3	4	...	2	2	1
11	43	3	1	...	2	1	1
12	52	3	1	...	2	2	2
13	32	2	1	...	2	1	1
14	41	3	2	...	2	2	2
15	33	3	4	...	2	1	1
16	38	3	4	...	2	2	1
17	45	3	2	...	3	2	2
18	40	2	3	...	3	2	1
19	48	3	4	...	2	1	2
20	37	2	4	...	2	2	1
21	40	3	4	...	2	1	1
22	62	4	4	...	1	3	1
23	34	3	2	...	2	1	1
24	42	3	1	...	2	1	2
25	48	3	4	...	2	1	2
26	34	3	2	...	2	2	1
27	26	2	4	...	2	1	1
28	44	3	2	...	3	2	2
...	...	...	...	...	...	...	...
380	47	2	2	...	2	2	1
381	42	3	2	...	2	2	1
382	43	3	1	...	3	1	2
383	35	2	2	...	2	1	2
384	35	3	1	...	2	2	1
385	47	3	2	...	2	1	2
386	48	3	4	...	2	1	2
387	46	3	2	...	3	2	2
388	45	3	2	...	3	2	2

Pada Tabel 4.11 merupakan data *training* yang telah dilakukan pengacakan menggunakan bantuan *software* RStudio dan telah di transformasi. Untuk atribut *class* merupakan atribut status yang diganti nama secara otomatis oleh *software* RStudio.

4.5. Teknik Data Mining

4.5.1. Analisis Algoritma C5.0

Algoritma C5.0 adalah metode data mining yang digunakan untuk mengklasifikasi data menghasilkan model pohon keputusan (*decision tree*). Langkah awal dalam pembentukan model pohon keputusan yaitu menghitung jumlah kejadian setiap atribut. Algoritma C5.0 bekerja dengan cara memilih atribut berdasarkan nilai dari *gain ratio* tertinggi. Sebagai contoh, dalam proses pembentukan pohon keputusan yaitu menggunakan proporsi 90:10, 90% dari total data yaitu 388 sampel digunakan sebagai data *training*, sementara 10% sisanya yaitu 34 sampel digunakan sebagai data *testing*.

Langkah-langkah dalam proses pembuatan *decision tree* menggunakan algoritma C5.0 sebagai berikut:

1. Menyiapkan data *training*. Data *training* yang dipakai sesuai dengan yang ditampilkan dalam Tabel 4.11.
2. Menghitung nilai *entropy* total dari semua data berdasarkan kategori label kelasnya. Menggunakan rumus dari persamaan (2.1), nilai *entropy* untuk seluruh data sampel dapat dihitung sebagai berikut:

$$Entropy(S) = \sum_{i=1}^m -p_i \log_2(p_i)$$

$$Entropy(total) = \left[-\frac{199}{388} \log_2 \frac{199}{388} + -\frac{189}{388} \log_2 \frac{189}{388} \right] = 0,9995$$

3. Menghitung nilai *entropy* untuk masing-masing atribut berdasarkan label kelasnya. Atribut yang digunakan dalam penelitian ini terdiri dari atribut numerik dan atribut kategorik. Dalam menentukan *entropy* dari atribut numerik dikelompokkan berdasarkan nilai rata-ratanya yaitu 45 tahun dan mediannya yaitu 44 tahun. Perhitungan *entropy* dari setiap atribut disajikan dalam Tabel 4.12.

Tabel 4.12. Perhitungan *Entropy* Semua Atribut

Atribut	Kategori	Jumlah	Diterima	Ditolak	Entropy
Usia	≤ 45	222	128	94	0,9830
	> 45	166	71	95	0,9849
	≤ 44	195	117	78	0,9710
	> 44	193	82	111	0,9837
Pendidikan	1	6	6	0	0
	2	61	39	22	0,9432
	3	284	117	167	0,9775
	4	37	37	0	0
Pekerjaan	1	89	33	56	0,9513
	2	143	74	69	0,9991
	3	29	5	24	0,6632
	4	121	81	40	0,9155
	5	6	6	0	0
Penghasilan	1	54	15	39	0,8524
	2	205	90	115	0,9892
	3	129	94	35	0,8433
Aset	1	353	164	189	0,9964
	2	35	35	0	0

Atribut	Kategori	Jumlah	Diterima	Ditolak	Entropy
Luas Lantai	1	29	6	23	0,7355
	2	342	176	166	0,9994
	3	17	17	0	0
Jenis Lantai	1	59	15	44	0,8179
	2	125	42	83	0,9209
	3	204	142	62	0,8860
Dinding	1	26	26	0	0
	2	285	158	127	0,9914
	3	77	15	62	0,7114
Penerangan	1	181	80	101	0,9903
	2	194	106	88	0,9938
	3	13	13	0	0

4. Menentukan *root node* atau *node* akar. Untuk menentukan *node* akar dari sebuah pohon keputusan, langkah pertama adalah memilih atribut yang paling berpengaruh dengan menghitung nilai *gain* dari setiap atribut menggunakan persamaan (2.2). Langkah selanjutnya adalah menghitung nilai *gain ratio* menggunakan persamaan (2.3). Nilai dari *gain ratio* digunakan sebagai *node* akar dari pohon keputusan. Sebagai contoh, diambil atribut pendidikan, maka persamaan (2.2) didefinisikan sebagai berikut:

$$Gain(Total, Pendidikan) =$$

$$Entropy(Total) - \sum_{i=1}^m \frac{|Pendidikan_i|}{|Total|} Entropy(Pendidikan_i)$$

Selanjutnya persamaan (2.4) yang digunakan untuk menghitung nilai dari *gain ratio* didefinisikan sebagai

berikut:

$$GainRatio = \frac{Gain(Total, Pendidikan)}{SplitInfo(Total, Pendidikan_i)}$$

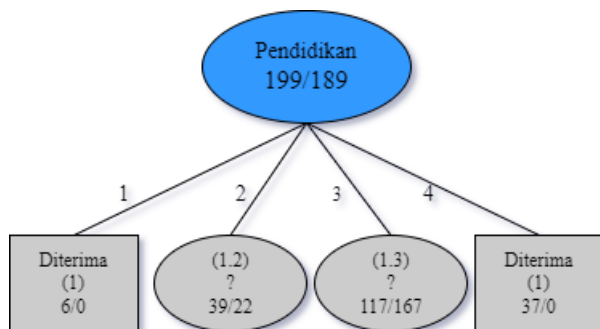
Perhitungan dari *gain ratio* digunakan untuk menentukan *node* akar ke-1 dalam algoritma C5.0. Diperoleh hasil perhitungan dari *gain* dan *gain ratio* untuk semua atribut tersaji pada Tabel 4.13.

Tabel 4.13. Perhitungan *Gain*, dan
Gain Ratio *Node* akar ke-1

Atribut	Kategori	Entropy	Gain	Gain Ratio
Usia	≤ 45	0,9830	-0,9616	-0,2451
	> 45	0,9849		
	≤ 44	0,9710		
	> 44	0,9837		
Pendidikan	1	0	1,8633	0,9701
	2	0,9432		
	3	0,9775		
	4	0		
Pekerjaan	1	0,9513	0,0780	0,0221
	2	0,9991		
	3	0,6632		
	4	0,9155		
	5	0		
Penghasilan	1	0,8524	0,0778	0,0290
	2	0,9892		
	3	0,8433		
Aset	1	0,9964	0,0930	0,0934
	2	0		

Atribut	Kategori	Entropy	Gain	Gain Ratio
Luas Lantai	1	0,7355	0,0636	0,0367
	2	0,9994		
	3	0		
Jenis Lantai	1	0,8179	0,1126	0,0429
	2	0,9209		
	3	0,8860		
Dinding	1	0	0,1301	0,0764
	2	0,9914		
	3	0,7114		
Penerangan	1	0,9903	0,0407	0,0205
	2	0,9938		
	3	0		

Tahap menentukan *node* akar ke-1 memakai atribut yang mempunyai nilai *gain ratio* tertinggi. Dari Tabel 4.13 dapat dilihat bahwa nilai dari *gain ratio* tertinggi diperoleh atribut Pendidikan sebesar 0,9701. Pengambilan keputusan disetiap cabang dilihat dari nilai *entropy*. Jika nilai *entropy* bernilai 0 (nol), maka cabang tersebut akan langsung diberi keputusan.



Gambar 4.15. *Node* Akar ke-1 Algoritma C5.0

Gambar 4.15 merupakan pohon keputusan untuk

menentukan *node* akar ke-1. Pohon keputusan tersebut memperlihatkan bahwa dalam mengambil keputusan dari setiap cabang dilihat dari nilai *entropy*. Pada atribut Pendidikan terdapat cabang yang bernilai 0 yaitu kategori 1 (SMA) dan 4 (Tidak Sekolah). Sedangkan untuk kategori 2 (SMP) dan 3 (SD) akan dilanjutkan perhitungan *node* akar 1.2 dan 1.3, karena keputusan masih belum jelas antara diterima atau ditolak. Perhitungan dari *gain ratio* akan berhenti ketika sudah tidak ada lagi atribut yang dapat dibagi menjadi bagian yang lebih kecil.

5. Ulangi langkah ke-3 dengan membagi kelas menjadi cabang dan jika ada cabang yang mempunyai dua kelas, pilihlah kelas yang memiliki jumlah terbanyak. Proses ini diulang untuk setiap cabang hingga setiap kelas pada cabang mempunyai kelas masing-masing. Disini, atribut yang telah menjadi cabang tidak diikuti lagi dalam perhitungan. Dengan bantuan *software* RStudio, diperoleh pohon keputusan untuk algoritma C5.0, dapat dilihat pada Gambar 4.16.

Dari Gambar 4.16 dapat disimpulkan bahwa:

- (a) Apabila penerima yang pendidikannya SMA(1) dan Tidak Sekolah(4) maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH.
- (b) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia ≤ 40 tahun, selanjutnya memiliki penghasilan 1.000.000 – 2.000.000(2) dan $< 1.000.000$ (3) maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH.
- (c) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia ≤ 40 tahun, selanjutnya memiliki penghasilan $> 2.000.000$ (1) dengan pendidikan SD(3) maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH.
- (d) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia ≤ 40 tahun, selanjutnya memiliki penghasilan $> 2.000.000$ (1) dengan pendidikan SMP(2) dan jenis lantainya adalah keramik(1) dan semen(2) maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH. Apabila jenis lantainya adalah tanah(3) maka diprediksi penerima tersebut ditolak(2) mendapatkan bantuan PKH tahap selanjutnya.
- (e) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia > 40 tahun, selanjutnya memiliki dinding bambu(1) maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH.
- (f) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia > 40 tahun, selanjutnya memiliki

dinding Kayu(2) dan tembok(3) dan tidak memiliki aset(2) maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH.

- (g) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia > 40 tahun, selanjutnya memiliki dinding Kayu(2) dan tembok(3) dan memiliki aset(1) dengan jenis lantai keramik(1) dan semen(2) dan juga dindingnya dari kayu(2) maka diprediksi penerima tersebut ditolak(2) mendapatkan bantuan PKH tahap selanjutnya.
- (h) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia > 40 tahun, selanjutnya memiliki dinding Kayu(2) dan tembok(3) dan memiliki aset(1) dengan jenis lantai keramik(1) dan semen(2) dan juga dindingnya dari tembok(3) dengan penghasilan $> 2.000.000(1)$ dan $1.000.000 - 2.000.000(2)$ maka diprediksi penerima tersebut ditolak(2) mendapatkan bantuan PKH tahap selanjutnya. Apabila penghasilan $< 1.000.000(3)$ maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH.
- (i) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia > 40 tahun, selanjutnya memiliki dinding Kayu(2) dan tembok(3) dan memiliki aset(1) dengan jenis lantai tanah(3) dengan usia ≤ 46 maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH.
- (j) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia > 40 tahun, selanjutnya memiliki dinding Kayu(2) dan tembok(3) dan memiliki aset(1)

dengan jenis lantai tanah(3) dengan usia > 46 dan usia > 52 maka diprediksi penerima tersebut tetap diterima bantuan PKH.

- (k) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia > 40 tahun, selanjutnya memiliki dinding Kayu(2) dan tembok(3) dan memiliki aset(1) dengan jenis lantai tanah(3) dengan usia > 46 dan usia ≤ 52 dengan dinding tembok(3) maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH.
- (l) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia > 40 tahun, selanjutnya memiliki dinding Kayu(2) dan tembok(3) dan memiliki aset(1) dengan jenis lantai tanah(3) dengan usia > 46 dan usia ≤ 52 dengan dinding kayu(2) dan luas lantai $> 100m^2(1)$ maka diprediksi penerima tersebut ditolak(2) mendapatkan bantuan PKH tahap selanjutnya. Apabila luas lantai $< 50m^2(3)$ maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH.
- (m) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia > 40 tahun, selanjutnya memiliki dinding Kayu(2) dan tembok(3) dan memiliki aset(1) dengan jenis lantai tanah(3) dengan usia > 46 dan usia ≤ 52 dengan dinding kayu(2) dan luas lantai $50m^2 - 100m^2(2)$, selanjutnya dengan penerangan > 450 Watt dan bukan listrik pribadi maka diprediksi penerima tersebut ditolak(2) mendapatkan bantuan PKH tahap selanjutnya.
- (n) Apabila penerima yang pendidikannya SMP(2) dan

SD(3) dengan usia > 40 tahun, selanjutnya memiliki dinding Kayu(2) dan tembok(3) dan memiliki aset(1) dengan jenis lantai tanah(3) dengan usia > 46 dan usia ≤ 52 dengan dinding kayu(2) dan luas lantai $50m^2 - 100m^2$ (2), selanjutnya dengan penerangan 450 Watt dan memiliki penghasilan $> 2.000.000$ (1) dan $< 1.000.000$ (3) maka diprediksi penerima tersebut tetap diterima bantuan PKH.

- (o) Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia > 40 tahun, selanjutnya memiliki dinding Kayu(2) dan tembok(3) dan memiliki aset(1) dengan jenis lantai tanah(3) dengan usia > 46 dan usia ≤ 52 dengan dinding kayu(2) dan luas lantai $50m^2 - 100m^2$ (2), selanjutnya dengan penerangan 450 Watt dan memiliki penghasilan $1.000.000 - 2.000.000$ (2) dengan usia ≤ 51 maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH. Apabila dengan usia > 51 maka diprediksi penerima tersebut ditolak(2) mendapatkan bantuan PHK tahap selanjutnya.

4.5.2. Analisis Algoritma CART

CART (*classification and regression tree*) merupakan algoritma yang menghasilkan pohon keputusan yang berbentuk pohon biner. Langkah pertama dalam pembentukan model dari pohon keputusan yaitu menyusun calon cabang dari semua atribut bebas dengan lengkap yaitu daftar calon cabang muktahir. Algoritma CART bekerja dengan cara memilih atribut berdasarkan nilai kesesuaian $\varphi(s|t)$ terbesar. Sebagai contoh, dalam proses pembentukan pohon keputusan yaitu juga menggunakan proporsi

90:10.

Langkah-langkah dalam proses pembuatan *decision tree* menggunakan algoritma CART sebagai berikut:

1. Menyiapkan data *training*. Data *training* yang dipakai sesuai dengan yang ditampilkan dalam Tabel 4.11.
2. Menyusun calon cabang dari semua atribut bebas dengan lengkap yaitu daftar calon cabang muktahir. Langkah awal yaitu menentukan calon cabang kiri dan kanan yang digunakan untuk menentukan *candidat split* untuk $t=root$ node sehingga membentuk pohon biner. Untuk atribut numerik dilakukan pemecahan dengan mengambil nilai dari *tree-based* model yang diperoleh dari output RStudio, untuk calon cabang iterasi ke-1 disajikan pada Tabel 4.14.

Tabel 4.14. Calon Cabang Iterasi ke-1

Candidat Split	Cabang Kiri	Cabang Kanan
1	Usia= $\geq 46,5$	Usia= $< 46,5$
2	Usia= $< 46,5$	Usia= $\geq 46,5$
3	Usia= $\geq 52,5$	Usia= $52,5$
4	Usia= $52,5$	Usia= $\geq 52,5$
5	Usia= $\geq 40,5$	Usia= $40,5$
6	Usia= $40,5$	Usia= $\geq 40,5$
7	Pendidikan= 1	Pendidikan= (2,3,4)
8	Pendidikan= 2	Pendidikan= (1,3,4)
9	Pendidikan= 3	Pendidikan= (1,2,4)
10	Pendidikan= 4	Pendidikan= (1,2,3)
11	Pekerjaan= 1	Pekerjaan= (2,3,4,5)
12	Pekerjaan= 2	Pekerjaan= (1,3,4,5)
13	Pekerjaan= 3	Pekerjaan= (1,2,4,5)
14	Pekerjaan= 4	Pekerjaan= (1,2,3,5)

Candidat Split	Cabang Kiri	Cabang Kanan
15	Pekerjaan= 5	Pekerjaan= (1,2,3,4)
16	Penghasilan= 1	Penghasilan= (2,3)
17	Penghasilan= 2	Penghasilan= (1,3)
18	Penghasilan= 3	Penghasilan= (1,2)
19	Aset= 1	Aset= 2
20	Aset= 2	Aset= 1
21	Luas Lantai= 1	Luas Lantai= (2,3)
22	Luas Lantai= 2	Luas Lantai= (1,3)
23	Luas Lantai= 3	Luas Lantai= (1,2)
24	Jenis Lantai= 1	Jenis Lantai= (2,3)
25	Jenis Lantai= 2	Jenis Lantai= (1,3)
26	Jenis Lantai= 3	Jenis Lantai= (1,2)
27	Dinding= 1	Dinding= (2,3)
28	Dinding= 2	Dinding= (1,3)
29	Dinding= 3	Dinding= (1,2)
30	Penerangan= 1	Penerangan= (2,3)
31	Penerangan= 2	Penerangan= (1,3)
32	Penerangan= 3	Penerangan= (1,2)

3. Menghitung nilai besaran kesuaian $\varphi(s|t)$ untuk masing-masing *candidat split*. Langkah pertama yaitu menentukan *root node* dari pohon keputusan. Pada data *training*, untuk menentukan *root node* dari pohon keputusan, diambil 32 atribut utama yaitu Usia, Pendidikan, Pekerjaan, Penghasilan, Aset, Luas Lantai, Jenis Lantai, Dinding, Penerangan. Kemudian dari data tersebut dihitung menjadi calon cabang dan menghitung PL (*candidat split purity left*) dan PR (*purity rigtt*). Sebagai contoh, perhitungannya untuk

candidat split ke-1, yaitu:

$$P_L = \frac{149}{388} = 0,38402$$

$$P_R = \frac{239}{388} = 0,61598$$

Hasil dari perhitungan untuk semua *candidat split* disajikan pada Tabel 4.15.

Tabel 4.15. Hasil Perhitungan PL dan PR Iterasi ke-1

Candidat Split	Jumlah Cabang Kiri	Jumlah Cabang Kanan	PL	PR
1	149	239	0,38402	0,61598
2	239	149	0,61598	0,38402
3	42	346	0,10825	0,89175
4	346	42	0,89175	0,10825
5	296	92	0,76289	0,23711
6	92	296	0,23711	0,76289
7	6	382	0,01546	0,98454
8	61	327	0,15722	0,84278
9	284	104	0,73196	0,26804
10	37	351	0,09536	0,90464
11	89	299	0,22938	0,77062
12	143	245	0,36856	0,63144
13	29	359	0,07474	0,92526
14	121	267	0,31186	0,68814
15	6	382	0,01546	0,98454
16	54	334	0,13918	0,86082
17	205	183	0,52835	0,47165
18	129	259	0,33247	0,66753
19	353	35	0,90979	0,09021

Candidat Split	Jumlah Cabang Kiri	Jumlah Cabang Kanan	PL	PR
20	35	353	0,09021	0,90979
21	29	359	0,07474	0,92526
22	342	46	0,88144	0,11856
23	17	371	0,04381	0,95619
24	59	329	0,15206	0,84794
25	125	263	0,32216	0,67784
26	204	184	0,52577	0,47423
27	26	362	0,06701	0,93299
28	285	103	0,73454	0,26546
29	77	311	0,19845	0,80155
30	181	207	0,46649	0,53351
31	194	194	0,50000	0,500000
32	13	375	0,3351	0,96649

Selanjutnya perhitungan $P(j|t_L)$ dan $P(j|t_R)$. Sebagai contoh, *candidat split* ke-1, yaitu:

$$P(j|t_L)(1) = \frac{68}{149} = 0,45638$$

$$P(j|t_L)(2) = \frac{81}{149} = 0,54362$$

$$P(j|t_R)(1) = \frac{131}{239} = 0,54812$$

$$P(j|t_R)(2) = \frac{108}{239} = 0,45188$$

Hasil dari perhitungan untuk semua *candidat split* disajikan pada Tabel 4.16.

Tabel 4.16. Hasil Perhitungan $P(j|t_L)$ dan $P(j|t_R)$ Iterasi ke-1

Candidat Split	Class	Jumlah Cab. Kiri	Jumlah Cab. Kanan	$P(j t_L)$	$P(j t_R)$
1	1	68	131	0,45638	0,54812
	2	81	108	0,54362	0,45188
2	1	131	68	0,54812	0,45638
	2	108	81	0,45188	0,54362
3	1	42	157	1,00000	0,45376
	2	0	189	0,00000	0,54624
4	1	157	42	0,45376	1,00000
	2	189	0	0,54624	0,00000
5	1	120	79	0,40541	0,85870
	2	176	13	0,59459	0,14130
6	1	79	120	0,85870	0,40541
	2	13	176	0,14130	0,59459
7	1	6	193	1,00000	0,05024
	2	0	189	0,00000	0,49476
8	1	39	160	0,63934	0,48930
	2	22	167	0,36066	0,51070
9	1	117	82	0,41197	0,78846
	2	167	22	0,58803	0,21154
10	1	37	162	1,00000	0,46154
	2	0	189	0,00000	0,53846
11	1	33	166	0,37079	0,57191
	2	56	133	0,62921	0,44482
12	1	74	125	0,51748	0,51020
	2	69	120	0,48252	0,48980
13	1	5	194	0,17241	0,54039
	2	24	165	0,82759	0,45961
14	1	81	118	0,66942	0,44195
	2	40	149	0,33058	0,55805
15	1	6	193	1,00000	0,50524

Candidat Split	Class	Jumlah Cab. Kiri	Jumlah Cab. Kanan	$P(j t_L)$	$P(j t_R)$
	2	0	189	0,00000	0,49476
16	1	15	184	0,27778	0,55090
	2	56	93	0,72222	0,27844
17	1	90	109	0,43902	0,59563
	2	69	80	0,56098	0,43716
18	1	94	105	0,72868	0,40541
	2	24	125	0,27132	0,48263
...	...	...	...	...	...
25	1	42	157	0,33600	0,59696
	2	83	106	0,66400	0,40304
26	1	142	57	0,69608	0,30978
	2	62	127	0,30392	0,69022
27	1	26	173	1,00000	0,47790
	2	0	189	0,00000	0,52210
28	1	158	41	0,55439	0,39806
	2	127	62	0,44561	0,60194
29	1	15	184	0,19841	0,59164
	2	62	127	0,80159	0,40836
30	1	80	119	0,44199	0,57488
	2	101	88	0,55801	0,42512
31	1	106	93	0,54639	0,47938
	2	88	101	0,45361	0,52062
32	1	13	189	1,00000	0,49600
	2	0	189	0,00000	0,50400

Dari Tabel 4.16 kemudian barulah dihitung nilai kesesuaian dari calon cabang s pada keputusan t yang dilambangkan $\varphi(s|t)$. Sebagai contoh, *candidat split* ke-1, yaitu:

Menghitung nilai dari $Q(s|t)$ terlebih dahulu menggunakan persamaan (2.5).

$$Q(s|t) = \sum_{j=1}^{\neq \text{classes}} |P(j|t_L) - P(j|t_R)|$$

$$\begin{aligned}
&= |0,45638 - 0,54812| + |0,54362 - 0,45188| \\
&= |0,09174| + |0,09174| = 0,18348
\end{aligned}$$

Selanjutnya menghitung nilai kesesuaian $\varphi(s|t)$ menggunakan persamaan (2.6).

$$2P_L P_R = 2(0,38402)(0,61598) = 0,47310$$

$$\varphi(s|t) = 2P_L P_R Q(s|t)$$

$$\varphi(s|t) = 0,47310(0,18348) = 0,08681$$

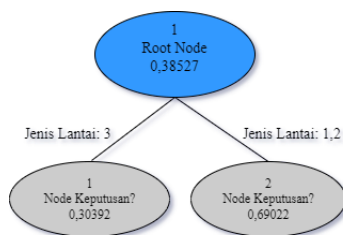
Hasil dari perhitungan untuk semua *candidat split* dapat dilihat pada Tabel 4.17.

Tabel 4.17. Hasil Perhitungan $\varphi(s|t)$ Iterasi ke-1

Candidat Split	$2P_L P_R$	$Q(s t)$	$\varphi(s t)$
1	0,47310	0,18348	0,08681
2	0,47310	0,18348	0,08681
3	0,19306	1,09249	0,21092
4	0,19306	1,09249	0,21092
5	0,36178	0,90658	0,32798
6	0,36178	0,90658	0,32798
7	0,03045	0,98953	0,03013
8	0,26500	0,30010	0,07952
9	0,39239	0,75298	0,29546
10	0,17253	1,07692	0,18581
11	0,35353	0,38552	0,13629
12	0,46545	0,01456	0,00678
13	0,13831	0,73595	0,10179
14	42920	0,45495	0,19527
15	0,03045	0,98953	0,03013
16	0,23961	0,71690	0,17178
17	0,49839	0,28042	0,13976

Candidat Split	$2P_L P_R$	$Q(s t)$	$\varphi(s t)$
18	0,44387	0,53458	0,23729
19	0,16414	1,07082	0,17576
20	0,16414	1,07082	0,17576
21	0,13831	0,66142	0,09148
22	0,20900	0,02924	0,00611
23	0,08379	1,01887	0,08537
24	0,25788	0,61007	0,15732
25	0,43675	0,52192	0,22795
26	0,49867	0,77259	0,38527
27	0,12504	1,04420	0,13057
28	0,38999	0,31266	0,12193
29	0,31814	0,79367	0,25250
30	0,49775	0,26578	0,13229
31	0,50000	0,13402	0,06701
32	0,96649	1,00800	0,06528

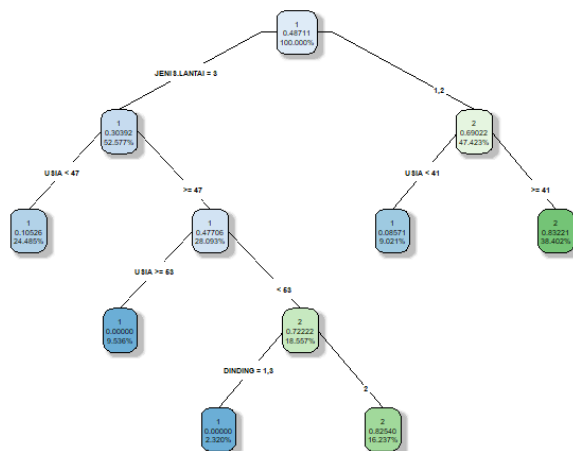
Pada tahap menentukan *root node* dari pohon keputusan akan diambil *candidat split* yang mempunyai nilai kesesuaian $\varphi(s|t)$ terbesar. Hasil perhitungan dari Tabel 4.17, diperoleh nilai $\varphi(s|t)$ terbesar adalah *candidat split* ke-26 sebesar 0,38527. oleh karena itu, pohon keputusan untuk iterasi ke-1 dari algoritma CART terbentuk seperti yang disajikan dalam Gambar 4.17.



Gambar 4.17. *Root Node* Iterasi ke-1 Algoritma CART

Gambar 4.17 merupakan pohon keputusan untuk menentukan *root node* itersai ke-1. Pohon keputusan tersebut menunjukkan bahwa dari kedua cabang masih berupa *node* keputusan, sehingga mengulangi iterasi ke-2 dan tidak lagi menggunakan cabang ke-26. Lakukan perhitungan hingga tidak ada lagi *node* keputusan, jika masih ada *node* keputusan maka pelaksanaan algoritma akan dilanjutkan dengan kembali ke iterasi berikutnya, dengan mengeluarkan *candidat split* yang sudah berhasil menjadi cabang terlebih dahulu hingga mendapat daftar dari *candidat split* muktahir yang baru.

4. Ulangi langkah ke-3 hingga menghasilkan pohon keputusan atau *decision tree*. Dengan bantuan *software* RStudio, diperoleh pohon keputusan dari algoritma CART dapat dilihat dalam Gambar 4.18.



Gambar 4.18. Model Pohon Keputusan Algoritma CART

Dari Gambar 4.18 dapat disimpulkan bahwa:

- (a) Apabila penerima yang memiliki jenis lantai dari tanah(3) dengan usia < 47 tahun maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH. Apabila dengan usia ≥ 47 tahun dan dengan usia ≥ 53 tahun maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH. Sedangkan Apabila dengan usia < 53 tahun dan memiliki dinding dari bambu(1) dan tembok(3) maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH. Tetapi Apabila memiliki dinding dari kayu(2) maka diprediksi penerima tersebut ditolak(2) mendapatkan bantuan PKH tahap selanjutnya.
- (b) Apabila penerima yang memiliki jenis lantai dari keramik(1) dan semen(2) dengan usia < 41 tahun maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH. Sedangkan apabila dengan usia ≥ 41 maka diprediksi penerima tersebut ditolak(2) mendapatkan bantuan PKH tahap selanjutnya.

4.5.3. Mengukur Ketepatan Klasifikasi

Setelah mendapatkan model pohon keputusan, langkah selanjutnya adalah menguji tingkat ketepatan klasifikasinya dengan menggunakan tabel *confusion matrix* untuk algoritma C5.0 dan CART. Sebagai contoh, tetap menggunakan proporsi data 90:10, dimana pengujiannya menggunakan data testing sebanyak 43 sampel, seperti yang tercantum di Lampiran 3.

4.5.3.1. Ketepatan Klasifikasi Algoritma C5.0

Tabel berikut merupakan *confussion matrix* yang diperoleh dari *output* RStudio, proporsi 90:10 digunakan untuk mengukur tingkat akurasi hasil klasifikasi dari algoritma C5.0.

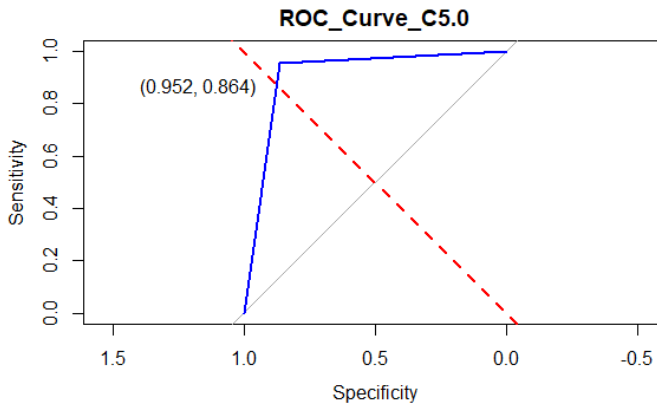
Tabel 4.18. *Confussion Matrix* Algoritma C5.0

Actual Class	Predicted Class	
	Diterima	Ditolak
Diterima	19	3
Ditolak	1	20

Berdasarkan Tabel 4.18 diketahui bahwa data yang diprediksi diterima ada 19 sampel yang diterima dan 1 sampel yang ditolak. Sedangkan data yang diprediksi ditolak ada 3 sampel yang diterima dan 20 sampel yang ditolak. Dari Tabel 4.18 dapat melakukan perhitungan untuk mencari nilai *accuracy* menggunakan persamaan (3.2), sebagai berikut:

$$\begin{aligned}
 Accuracy &= \frac{TP + TN}{TP + FP + TN + FN} \times 100\% \\
 &= \frac{19 + 20}{19 + 1 + 20 + 3} \times 100\% \\
 &= \frac{39}{43} \times 100\% = 90,70\%
 \end{aligned}$$

Berdasarkan perhitungan diatas diperoleh nilai dari *accuracy* sebesar 90,70%. Selanjutnya dilakukan pengujian dengan menggunakan AUC-ROC. Pengujian menggunakan AUC-ROC dilakukan dengan bantuan *software* RStudio dengan diperoleh kurva ROC sebagai berikut:



Gambar 4.19. Kurva ROC Algoritma C5.0

Dari Gambar 4.19 didapatkan informasi bahwa nilai dari *specivicity* sebesar 0,952 dan nilai dari *sentivity* sebesar 0,864. Selanjutnya dilakukan perhitungan untuk mencari nilai dari AUC menggunakan persamaan (3.3).

$$\begin{aligned}
 AUC &= \frac{1}{2} \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right) \\
 &= \frac{1}{2} \left(\text{Sentivity} + \text{Specivicity} \right) \\
 &= \frac{1}{2} (0,864 + 0,952) = 0,9080
 \end{aligned}$$

Berdasarkan perhitungan diatas diperoleh nilai AUC sebesar 0,9080. *Output* komputasi *software* RStudio juga menghasilkan nilai *accuracy*, *sentivity*, *specivicity* dan AUC dari semua proporsi data, tersaji dalam Tabel 4.19.

Tabel 4.19. Nilai *Accuracy*, *Sentivity*, *Specificity* dan *AUC* untuk Semua Proporsi Algoritma C5.0

Proporsi	Accuracy	Sentivity	Specificity	AUC
50:50	82,79%	80,20%	85,09%	0,8264
60:40	91,86%	88,31%	94,74%	0,9152
70:30	92,25%	91,53%	92,86%	0,9219
80:20	93,02%	89,74%	95,74%	0,9274
90:10	90,70%	86,36%	95,24%	0,9080
Rata-rata	90,12%	87,23%	92,73%	0,8998

Berdasarkan Tabel 4.19 diperoleh informasi bahwa tingkat akurasi tertinggi diperoleh pada proporsi 80:20 sebesar 93,02% sedangkan nilai AUC tertinggi diperoleh pada proporsi 80:20 sebesar 0,9274. Sehingga penerapan algoritma C5.0 dalam klasifikasi penerimaan bantuan PKH lebih baik menggunakan proporsi data 80:20.

4.5.3.2. Ketepatan Klasifikasi Algoritma CART

Tabel berikut merupakan *confussion matrix* yang diperoleh dari *output* RStudio, proporsi 90:10 digunakan untuk mengukur tingkat akurasi hasil klasifikasi dari algoritma CART.

Tabel 4.20. *Confussion Matrix* Algoritma CART

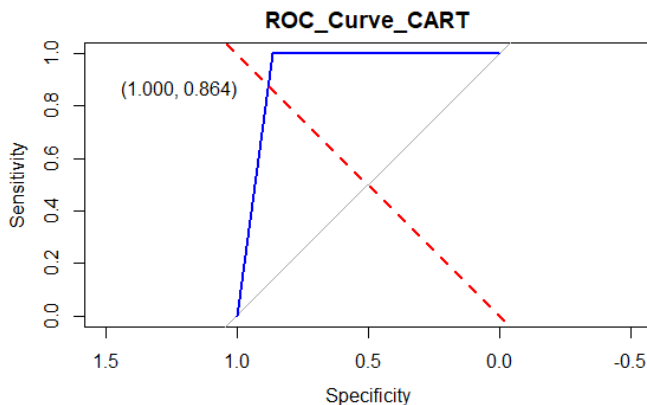
Actual Class	Predicted Class	
	Diterima	Ditolak
Diterima	19	0
Ditolak	3	21

Berdasarkan Tabel 4.20 diketahui bahwa data yang diprediksi diterima ada 19 sampel yang diterima dan 3 sampel yang

ditolak. Sedangkan data yang diprediksi ditolak ada 0 sampel yang diterima dan 21 sampel yang ditolak. Dari Tabel 4.18 dapat melakukan perhitungan untuk mencari nilai *accuracy* menggunakan persamaan (3.2), sebagai berikut:

$$\begin{aligned}
 Accuracy &= \frac{TP + TN}{TP + FP + TN + FN} \times 100\% \\
 &= \frac{19 + 21}{19 + 3 + 21 + 0} \times 100\% \\
 &= \frac{40}{43} \times 100\% = 93,02\%
 \end{aligned}$$

Berdasarkan perhitungan diatas diperoleh nilai dari *accuracy* sebesar 93,02%. Selanjutnya dilakukan pengujian dengan menggunakan AUC-ROC. Pengujian menggunakan AUC-ROC dilakukan dengan bantuan *software* RStudio dengan diperoleh kurva ROC sebagai berikut:



Gambar 4.20. Kurva ROC Algoritma CART

Dari Gambar 4.20 didapatkan informasi bahwa nilai dari *specivicity* sebesar 1,0000 dan nilai dari *sentivity* sebesar 0,8636. Selanjutnya dilakukan perhitungan untuk mencari nilai dari AUC menggunakan persamaan (3.3).

$$\begin{aligned}
 AUC &= \frac{1}{2} \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right) \\
 &= \frac{1}{2} \left(\text{Sentivity} + \text{Specivicity} \right) \\
 &= \frac{1}{2} (0,8636 + 1,000) = 0,9318
 \end{aligned}$$

Berdasarkan perhitungan diatas diperoleh nilai AUC sebesar 0,9318. *Output* komputasi *software* RStudio juga menghasilkan nilai *accuracy*, *sentivity*, *specivicity* dan AUC dari semua proporsi data, tersaji dalam Tabel 4.21.

Tabel 4.21. Nilai *Accuracy*, *Sentivity*, *Specivicity* dan AUC untuk Semua Proporsi Algoritma CART

Proporsi	Accuracy	Sentivity	Specivicity	AUC
50:50	87,91%	82,18%	92,98%	0,8758
60:40	86,05%	77,92%	92,63%	0,8528
70:30	86,82%	91,53%	82,86%	0,8719
80:20	87,21%	89,74%	85,11%	0,8742
90:10	93,02%	86,36%	100%	0,9318
Rata-rata	88,20%	85,55%	90,72%	0,8813

Berdasarkan Tabel 4.21 diperoleh informasi bahwa tingkat akurasi tertinggi diperoleh dari proporsi 90:10 sebesar 93,02% sedangkan nilai AUC tertinggi diperoleh dari proporsi 90:10 sebesar 0,9318. Sehingga penerapan algoritma CART dalam

klasifikasi penerimaan bantuan PKH lebih baik menggunakan proporsi data 90:10.

4.5.3.3. Perbandingan Algoritma C5.0 dan CART

Setelah dilakukan analisis untuk kedua algoritma tersebut, selanjutnya dilakukan perbandingan antara kedua algoritma dengan melihat tingkat akurasi dan nilai AUC yang dipilih dari hasil terbaik dari masing-masing proporsi kedua metode tersebut:

Tabel 4.22. Perbandingan Tingkat Akurasi dan Nilai AUC untuk Kedua Algoritma

Rata-rata	Algoritma C5.0	Algoritma CART
Tingkat Akurasi	90,12%	88,20%
Nilai AUC	0,8998	0,8813

Tabel 4.22 menunjukkan perbandingan tingkat akurasi dan nilai AUC dari kedua algoritma yang diperoleh dari komputasi *software* RStudio. Dapat dilihat dari tabel diatas rata-rata tingkat akurasi dari algoritma C5.0 sebesar 90,12% dengan rata-rata nilai AUC sebesar 0,8998 dan rata-rata tingkat akurasi dari algoritma CART sebesar 88,20% dengan nilai AUC sebesar 0,8813. Sehingga algoritma C5.0 yang memiliki tingkat akurasi dan nilai AUC tertinggi dibandingkan algoritma CART. Hal ini menunjukkan bahwa algoritma C5.0 menghasilkan model klasifikasi yang lebih baik dibandingkan algoritma CART.

4.5.3.4. Pengujian Data *Testing* kedalam Model Algoritma Terbaik

Setelah dilakukan perbandingan antara algoritma C5.0 dan CART diperoleh hasil bahwa algoritma C5.0 menghasilkan model klasifikasi yang lebih baik dibandingkan algoritma CART. Sehingga untuk pengujian data *testing* menggunakan algoritma C5.0. Berikut hasil prediksi pada data *testing* kedalam model algoritma C5.0 dengan menggunakan proporsi 90:10, tersaji pada Tabel 4.23.

Tabel 4.23. Hasil Prediksi pada Data *Testing*

NO	USIA	PENDIDIKAN	PEKERJAAN	...	PENERANGAN	class	PREDIKSI
1	49	2	1	...	1	2	2
2	51	3	4	...	2	2	2
3	41	3	2	...	2	2	2
4	47	3	4	...	1	2	2
5	50	2	2	...	1	2	2
6	51	3	4	...	2	2	2
7	48	3	4	...	1	2	2
8	43	3	3	...	1	2	2
9	50	3	4	...	2	2	2
10	42	3	1	...	1	2	2
11	43	3	1	...	1	2	2
12	46	3	2	...	1	2	2
13	46	3	2	...	2	2	2
14	43	3	2	...	2	2	2
15	51	3	4	...	2	2	2
16	51	3	2	...	2	2	1
17	45	3	2	...	2	2	2
18	42	3	3	...	1	2	2
19	45	3	2	...	2	2	2
20	46	3	2	...	2	2	2
21	47	3	4	...	1	2	2
22	64	3	4	...	2	1	1
23	55	3	4	...	1	1	1
24	37	2	1	...	1	1	1
25	36	2	2	...	2	1	1
26	36	3	1	...	2	1	1
27	35	3	2	...	2	1	1
28	30	1	2	...	2	1	1
29	42	3	4	...	2	1	1
30	36	3	2	...	2	1	1
31	45	3	3	...	2	1	1
32	46	3	4	...	1	1	1
33	63	4	5	...	3	1	1
34	41	3	2	...	2	1	2
35	32	3	2	...	1	1	1
36	77	4	5	...	2	1	1
37	34	2	1	...	1	1	2
38	48	4	4	...	2	1	1
39	42	3	4	...	1	1	1
40	68	3	4	...	2	1	1
41	43	3	2	...	1	1	2
42	71	4	4	...	3	1	1
43	70	4	4	...	3	1	1

Tabel 4.23 menunjukkan bahwa ada 1 sampel yang aktual ditolak tetapi diprediksi diterima yang ditandai dengan warna kuning. Sedangkan ada 3 sampel yang aktual diterima tetapi diprediksi ditolak yang ditandai dengan warna hijau. Dimana hasil

prediksi tersebut dianalisis menggunakan *rule* dari model pohon keputusan algoritma C5.0 dengan bantuan *software* RStudio. Diambil contoh analisis untuk 1 sampel yang aktual ditolak tetapi diprediksi diterima dengan melihat kesimpulan model pohon keputusan C5.0 yaitu Apabila penerima yang pendidikannya SMP(2) dan SD(3) dengan usia > 40 tahun, selanjutnya memiliki dinding Kayu(2) dan tembok(3) dan memiliki aset(1) dengan jenis lantai tanah(3) dengan usia > 46 dan usia ≤ 52 dengan dinding kayu(2) dan luas lantai $50m^2 - 100m^2$ (2), selanjutnya dengan penerangan 450 Watt dan memiliki penghasilan 1.000.000 – 2.000.000(2) dengan usia ≤ 51 maka diprediksi penerima tersebut tetap diterima(1) bantuan PKH. Untuk data testingnya secara lengkap disajikan pada Lampiran 3.

BAB 5

PENUTUP

5.1. Kesimpulan

Dari hasil analisis dan pembahasan yang telah dilakukan, dapat disimpulkan sebagai berikut:

1. Model klasifikasi dari algoritma C5.0 membentuk pohon keputusan (*decision tree*) untuk memprediksi status penerima bantuan PKH Desa Sembungharjo Kecamatan Pulokulon. Pohon keputusan terbentuk berdasarkan nilai dari *entropy*, *gain* dan *gain ratio*. Atribut yang paling berpengaruh dari algoritma C5.0 yaitu atribut pendidikan, usia, dinding, jenis lantai, aset, penghasilan, luas lantai, dan penerangan. Hasil ketepatan klasifikasi penerima bantuan PKH menggunakan algoritma C5.0 untuk proporsi 80:20 menghasilkan tingkat akurasi tertinggi yaitu 93,02% dan memperoleh nilai AUC tertinggi yaitu 0,9274.
2. Model klasifikasi CART membentuk pohon keputusan (*decision tree*) untuk memprediksi status penerima bantuan PKH Desa Sembungharjo Kecamatan Pulokulon. Pohon keputusan yang terbentuk berdasarkan nilai kesesuaian $\varphi(s|t)$ terbesar. Pada model algoritma CART, atribut yang paling berpengaruh yaitu atribut jenis lantai, usia, dan dinding. Hasil ketepatan klasifikasi penerima bantuan PKH menggunakan algoritma CART untuk proporsi 90:10 memperoleh akurasi tertinggi yaitu 93,02% dan memperoleh nilai AUC tertinggi yaitu 0,9318.

3. *Output* pada penelitian ini adalah status penerima bantuan PKH yang diprediksi menggunakan teknik klasifikasi data mining dengan algoritma C5.0 dan CART dengan bantuan *software* RStudio. Hasil perhitungan memperlihatkan bahwa algoritma C5.0 menghasilkan model yang lebih baik dibandingkan algoritma CART, ditunjukkan dari rata-rata tingkat akurasi sebesar 90,12% dan rata-rata nilai AUC sebesar 0,8998. Tingkat Akurasi dan nilai AUC yang paling baik jika menggunakan kedua algoritma tersebut terdapat pada proporsi 90:10 dimana nilai AUC dari kedua algoritma tersebut berada pada kategori klasifikasi sangat baik. Kondisi ini menunjukkan bahwa 90:10 merupakan proporsi yang paling efektif untuk menjalankan algoritma klasifikasi dalam penerimaan bantuan PKH Desa Sembungharjo Kecamatan Pulokulon.

5.2. Saran

Berikut beberapa saran yang dapat penulis berikan untuk penelitian ini:

1. Hasil penelitian menyimpulkan bahwa dalam memprediksi status penerima bantuan program keluarga harapan dapat menggunakan metode klasifikasi yang lain agar memperoleh metode yang terbaik.
2. Saran bagi penelitian selanjutnya ketika menggunakan data yang sama sebaiknya mengambil sampel dari calon penerima bantuan program keluarga harapan agar data lebih real atau mendekati hasil yang sebenarnya.

DAFTAR PUSTAKA

- Azmatul B., R., Dina S., I., M. Afendi, Farit. 2013. Penerapan Synthetic Minority Oversampling Technique (SMOTE) terhadap data tidak seimbang pada Pembuatan Model Komposisi Jamu. *Xplore*, 1(1): 1-6.
- Bisri, Achmad, dan Romi S. Wahono. 2015. Penerapan Adaboost untuk Penyelesaian Ketidakseimbangan Kelas pada Penentuan Kelulusan Mahasiswa dengan Metode Decision Tree. *Journal of Intelligent Systems*, 1 (1): 27-32.
- Fitriana, Eka. 2020. Perbandingan Algoritma C4.5 dan Naive Bayes Untuk Menentukan Kelayakan Penerima Bantuan Program Keluarga Harapan. *SISTEMASI: Jurnal Sistem Informasi*, 9(1): 103-115.
- Gorunescu, Florin. 2011. *Data Mining: Concept, Model Technique*. Springer-Verlag Berlin Heidelberg.
- Han, Jiawei, dan Kember, Micheline. 2011. *Data Mining: Concepts and Techniques, 3rd Edition*. Burlington: Morgan Kaufmann.
- Hanum, E.Nst., Rismayanti, dan Lestari Y.D. 2021. Penerapan Data Mining Untuk Memprediksi Hasil Produksi Buah Kelapa Sawit Pada PT. LNK Kebun Basilam Menggunakan Algoritma C5.0. *Seminar Nasional Teknologi Informasi & Komunikasi*, 978-623-91911-2-2.
- Kemenkeu, Dirjen Anggaran. 2015. *Kajian Program Keluarga Harapan*. Diunduh di <https://anggaran.kemenkeu.go.id/> tanggal 28 April 2023.

- Kemensos RI. 2019. *Program Keluarga Harapan (PKH)*. Diakses pada tanggal 30 April 2024 dari <https://kemensos.go.id/program-keluarga-harapan-pkh>
- Kemensos RI. 2020. *Rencana Strategis Kementerian Sosial*. Diunduh tanggal 23 November 2023 di <https://kemensos.go.id/uploads/topics/15995730687808.pdf>
- Kharat, Kailash D., dan Anil V. Turukmane. 2014. *Classification of Brain Tumor By Feature Vektor Using CART and C5.0 Algorithms*. *JAAST: Material Science*, 1(2): 173-178.
- Larose, Danie T. 2005. *Discovering Knowledge in Data: An Introduction to Data Mining*. John Wiley & Sons, inc.
- Maung, Ei T.W. 2020. *Comparasion of Data Mining Classification Algorithms: C5.0 and CART for Car Evaluation and Credit Card Information Datasets*. Thesis. Yangon: University of Computer Studies, Yangon.
- Muflikhah, L., Ratnawati, D.E, Putri, Rekyan R.M. 2018. *Data Mining*. Malang: UB Press.
- Pakpahan, H. S., F. Indar, M. Wati. 2018. *Penerapan Algoritma CART Decision Tree pada Penentuan Penerima Program Bantuan Pemerintah Daerah Kabupaten Kutai Kartanegara*. *JURTI*, 2(1): 2579-8790.
- Patil, N., R. Lathi, V. Chitre. 2012. *Customer Card Classification Based on C5.0 & CART Algorithms*. *International Journal of Engineering Research and Aplication (IJERA)*, 2(4): 164-167.

- Pramana, S, Yuniarto, B, Mariyah, Siti, Santoso, Ibnu dan Nooraeni. 2018. *Data Mining dengan R. Konsep Serta Implementasi*. Bogor: IN MEDIA.
- Prasetyo, E. 2014. *Data Mining-Mengolah Data Menjadi Informasi Menggunakan Matlab*. Yogyakarta: Penerbit Andi.
- Pratiwi, R., M. Nor Hayati, Surya P. 2020. Perbandingan Klasifikasi Algoritma C5.0 dan Classification and Regression Tree (Studi Kasus: Data Sosial Kepala Keluarga Masyarakat Teluk Baru Kecamatan muara Anclong Tahun 2019). *Jurnal Ilmu Matematika dan Terapan*, 14(2): 267-278.
- Pratiwi, Reni. 2020. *Perbandingan Klasifikasi Algoritma C5.0 dan Classification and Regression Tree (Studi Kasus: Data Sosial Kepala Keluarga Masyarakat Teluk Baru Kecamatan muara Anclong Tahun 2019)*. Skripsi. Samarinda: Universitas Mulawarman Samarinda.
- Priyam, Anuja, Abijheet, Ghupta R., A. Rathee, Srivastava S. 2013. *Comparative Analysis of Decision Tree Classification Algorithms. International Journal of Current Engineering and Technology*, 3(2): 227-4106.
- Putri, Yolanda R. 2016. *PREDIKSI POLA KECELAKAAN KERJA PADA PERUSAHAAN NON EKSTRAKTIF MENGGUNAKAN ALGORITMA DECISION TREE: C4.5 DAN C5.0*. Thesis. Surabaya: Institut Teknologi Sepulun November.
- Quinlan, J. R. 1993. *C4.5: Programs for Machine Learning*. San Fransisco: Morgan Kaufmann.

- Rachmawati, Ariska K., dan Ardani, Riska A. 2021. *Analisis Kinerja Algoritma C5.0 dan Naïve Bayes untuk Pengenalan Pola Lulusan Mahasiswa*. Diunduh di <https://eprints.walisongo.ac.id/id/eprint/19884/> tanggal 27 November 2023.
- Rachmawati, Ariska Kurnia, Minhayati Saleh, M. Nur Ramdani. 2021. *Comparasion and Prediction of Data Mining Models to Determine the Clasification of Family Planning Program User Status*. *Indonesian Journal of Mathematics Education*, 4(2): 66-73.
- Ramdani, A., Sofyan, C.D., Ramdani, F., Tama, M.F.A., Rachmatsyah, M.A. 2022. Algoritma Klasifikasi Data Mining Untuk Memprediksi Masyarakat dalam Menerima Bantuan Sosial. *Jurnal Ilmiah Sistem Informasi (JUSI)*, 1(2): 39-47.
- Sari, Veronica Retno, Feranandah, Firdausi, Yufis, Azhar. 2020. Perbandingan Prediksi Kualitas Kopi Arabika dengan Menggunakan Algoritma SGD, Random Forest dan Naïve Bayes. *Jurnal Pendidikan Informatika*, 4 (2): 1-9.
- Shihab, M. Quraish. 2002. *TAFSIR AL-MISBAH pesan, kesan dan keserasian al-Qur'an*. Jakarta : Lentera Hati.
- Siregar, Syofian. 2015. *Metode Penelitian Kuantitatif: Dilengkapi Perbandingan Perhitungan Manual & SPSS Edisi Pertama* Jakarta: Prenadamedia Group.
- Sofyan, Sabiq, dan A. Prasetyo. 2021. Penerapan Synthetic Minority Oversampling Technique (SMOTE) Terhadap Data Tidak Seimbang Pada Tingkat Pendapatan Pekerja Informal

- Di Provinsi D.I. Yogyakarta Tahun 2019. *Seminar Nasional Official Statistics*.
- Sugiyono. 2014. *Statistika untuk Penelitian*. Bandung: Alfabeta.
- Sugiyono. 2018. *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*. Bandung: Alfabeta.
- Susanto, S., dan Suryadi, D. 2010. *Pengantar Data Mining*. Yogyakarta: Andi Offset.
- Susanti, Y., Respatiwan, Handajani, S.S., dkk. 2019. *Clasificattion of teak wood production in Central Java using the C5.0 algorithm. International Conference on Aplied Science (ICSAS)*, 020094-1-020094.
- Wijaya, A.C., Hasibuan, N.A, Ramdani, P. 2018. Implementasi Algoritma C5.0 dalam Klasifikasi Pendapatan Masyarakat (Studi Kasus: Kelurahan Mesjid Kecamatan Medan Kota). *Majalah Ilmian INTI*, 5(3), 2339-210X.
- Wu Xidong, dan Kumar, V. 2008. *The Top Ten Algorithms in Data Mining*. London: CRC Press.
- Yusuf, Y.W. 2007. Perbandingan Performansi Algoritma Decision Tree C5.0, CART, dan CHAID: Kasus Prediksi Status Resiko Kredit di Bank X. *Seminar Nasional Aplikasi Teknologi Informasi*, 1907-5022.

Lampiran 1. Kuesioner Penelitian

KUESIONER PENELITIAN

Saya Rizka Zuliyantri, mahasiswa UIN Walisongo Prodi Matematika. Saat ini saya sedang melakukan penulisan skripsi dengan topik penelitian "Analisis Perbandingan Klasifikasi Data Mining Menggunakan Algoritma C5.0 dan CART pada Penerimaan Program Keluarga Harapan". Sehubungan dengan hal tersebut, saya bermaksud untuk meminta kesediaan Bapak/Ibu untuk menjadi responden dengan mengisi kuesioner ini. Seluruh data yang Bapak/Ibu berikan akan bersifat rahasia dan hanya akan di gunakan untuk kepentingan penelitian. Atas kesediaan dan partisipasi Bapak/Ibu saya ucapkan terimakasih.

A. Identitas Responden

Nama :
Usia :
Jenis Kelamin :
Alamat :

B. Daftar Pertanyaan

Petunjuk pengisian: centanglah (✓) salah satu pilihan jawaban pertanyaan dibawah ini sesuai dengan kejadian yang sebenarnya.

- Pendidikan terakhir kepala keluarga
☐ Tidak sekolah ☐ SMP
☐ SD ☐ SMA
- Pekerjaan kepala keluarga
☐ Tidak bekerja ☐ Buruh harian ☐ Lainnya
☐ Petani ☐ Pedagang
- Penghasilan kepala keluarga
☐ < 1.000.000
☐ 1.000.000 – 2.000.000
☐ > 2.000.000
- Tabungan/aset senilai minimal Rp. 500.000
☐ Ya ☐ Tidak
- Luas lantai tempat tinggal
☐ < 50 m²
☐ 50 m² – 100 m²
☐ > 100 m²
- Jenis lantai tempat tinggal
☐ Tanah
☐ Semen
☐ Keramik
- Jenis dinding tempat tinggal
☐ Bambu
☐ Kayu
☐ Tembok
- Sumber penerangan
☐ Bukan listrik
☐ Listrik 450 watt
☐ > Listrik 450 watt

Lampiran 2. Data Penerima Bantuan PKH Tahap I

NO	USIA	PENDIDIKAN	PEKERJAAN	PENGHASILAN	ASET	LUAS LANTAI	JENIS LANTAI	DINDING	PENERANGAN	STATUS
1	35	SMA	Buruh Harian	1.000.000 - 2.000.000	Ya	> 100m2	Semen	Kayu	450 Watt	Diterima
2	48	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Ditolak
3	47	SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
4	30	SMP	Buruh Harian	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
5	53	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
6	45	SMP	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
7	45	SMP	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
8	42	SD	Lainnya	>2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
9	32	SMP	Lainnya	>2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Ditolak
10	42	SD	Petani	>2.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Ditolak
11	37	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
12	66	Tidak Sekolah	Tidak Bekerja	<1.000.000	Tidak	<50m2	Tanah	Kayu	450 Watt	Diterima
13	64	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
14	84	Tidak Sekolah	Tidak Bekerja	<1.000.000	Tidak	<50m2	Tanah	Bambu	Bukan Listrik Pribadi	Diterima
15	65	SD	Buruh Harian	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
16	47	SD	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
17	46	SMA	Lainnya	>2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
18	27	SD	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
19	37	SD	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
20	44	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Bambu	>450 Watt	Diterima
21	63	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
22	62	Tidak Sekolah	Tidak Bekerja	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	Bukan Listrik Pribadi	Diterima
23	55	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
24	35	SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima
25	37	SMP	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
26	22	SMP	Buruh Harian	<1.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
27	26	SMP	Lainnya	>2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	Bukan Listrik Pribadi	Diterima
28	27	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
29	31	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
30	36	SMP	Buruh Harian	<1.000.000	Ya	50m2 - 100m2	Keramik	Kayu	450 Watt	Diterima
31	36	SD	Lainnya	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
32	35	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Bambu	450 Watt	Diterima
33	37	SD	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
34	45	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Keramik	Tembok	450 Watt	Ditolak
35	39	SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
36	36	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
37	30	SMA	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
38	30	SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
39	32	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
40	35	Tidak Sekolah	Lainnya	<1.000.000	Tidak	<50m2	Tanah	Kayu	>450 Watt	Diterima
41	37	SD	Buruh Harian	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
42	40	Tidak Sekolah	Petani	<1.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
43	48	Tidak Sekolah	Lainnya	<1.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
44	50	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Keramik	Tembok	>450 Watt	Diterima
45	56	SD	Pedagang	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
46	57	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima
47	32	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
48	34	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
49	37	SD	Buruh Harian	>2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
50	50	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
51	42	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
52	41	SD	Buruh Harian	<1.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Ditolak
53	41	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
54	47	SD	Lainnya	>2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
55	37	SMP	Buruh Harian	>2.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
56	44	Tidak Sekolah	Petani	<1.000.000	Ya	>100m2	Tanah	Kayu	450 Watt	Diterima
57	57	SD	Petani	1.000.000 - 2.000.000	Tidak	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima
58	53	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Semen	Bambu	450 Watt	Diterima
59	37	SMP	Petani	>2.000.000	Ya	50m2 - 100m2	Semen	Tembok	>450 Watt	Diterima
60	49	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Bambu	450 Watt	Diterima
61	54	SD	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
62	43	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Bambu	450 Watt	Diterima
63	42	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
64	40	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima
65	41	SD	Petani	<1.000.000	Ya	>100m2	Tanah	Kayu	450 Watt	Diterima
66	36	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
67	35	SMP	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
68	37	SD	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima

69	38 SD	Petani	1.000.000 - 2.000.000	Ya	30m2 - 100m2	Tanah	Kayu	> 450 Watt	Diterima
70	42 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	< 50m2	Tanah	Bambu	450 Watt	Diterima
71	39 SMP	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Bambu	450 Watt	Diterima
72	38 SD	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	> 450 Watt	Diterima
73	45 SD	Pedagang	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
74	29 SMP	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Saman	Tembok	450 Watt	Diterima
75	42 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
76	43 SD	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
77	39 SD	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	> 450 Watt	Diterima
78	36 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Saman	Kayu	> 450 Watt	Diterima
79	27 SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Saman	Kayu	450 Watt	Diterima
80	38 SD	Petani	< 1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
81	45 SD	Petani	< 1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
82	47 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Saman	Kayu	450 Watt	Diterima
83	45 SD	Lainnya	< 1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
84	38 SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	> 450 Watt	Diterima
85	48 SD	Petani	< 1.000.000	Ya	< 50m2	Tanah	Bambu	450 Watt	Diterima
86	32 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	> 450 Watt	Diterima
87	43 SD	Petani	> 2.000.000	Ya	50m2 - 100m2	Saman	Kayu	> 450 Watt	Diterima
88	40 SD	Lainnya	1.000.000 - 2.000.000	Ya	< 50m2	Keramik	Kayu	> 450 Watt	Diterima
89	43 SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Bambu	450 Watt	Diterima
90	50 SD	Petani	< 1.000.000	Ya	50m2 - 100m2	Tanah	Bambu	450 Watt	Diterima
91	46 SD	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	> 450 Watt	Diterima
92	43 SD	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Tembok	> 450 Watt	Ditolak
93	52 SD	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Ditolak
94	47 SD	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	> 450 Watt	Ditolak
95	54 SD	Petani	> 2.000.000	Ya	> 100m2	Tanah	Kayu	450 Watt	Ditolak
96	63 Tidak Sekolah	Petani	< 1.000.000	Tidak	< 50m2	Tanah	Kayu	> 450 Watt	Diterima
97	46 SD	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
98	51 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
99	45 SMA	Buruh Harian	> 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
100	61 SD	Buruh Harian	< 1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
101	42 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	> 450 Watt	Diterima
102	63 Tidak Sekolah	Tidak Bekerja	< 1.000.000	Tidak	< 50m2	Tanah	Bambu	Bukan Listrik Pribadi	Diterima
103	35 SD	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Saman	Kayu	450 Watt	Diterima
104	48 SD	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Keramik	Kayu	> 450 Watt	Diterima
105	47 SMP	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
106	42 SD	Lainnya	< 1.000.000	Tidak	50m2 - 100m2	Tanah	Bambu	> 450 Watt	Diterima
107	37 SMP	Petani	< 1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
108	41 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
109	32 SD	Buruh Harian	1.000.000 - 2.000.000	Tidak	50m2 - 100m2	Semen	Kayu	> 450 Watt	Diterima
110	35 SMA	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Keramik	Kayu	> 450 Watt	Diterima
111	47 SD	Lainnya	< 1.000.000	Ya	50m2 - 100m2	Tanah	Bambu	450 Watt	Diterima
112	47 SD	Petani	< 1.000.000	Ya	50m2 - 100m2	Tanah	Bambu	> 450 Watt	Diterima
113	45 SD	Buruh Harian	< 1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
114	44 SD	Lainnya	< 1.000.000	Tidak	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
115	42 SD	Buruh Harian	< 1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
116	29 SMP	Buruh Harian	< 1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	> 450 Watt	Diterima
117	34 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
118	40 SMP	Pedagang	1.000.000 - 2.000.000	Ya	> 100m2	Keramik	Tembok	450 Watt	Diterima
119	45 Tidak Sekolah	Petani	> 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	> 450 Watt	Diterima
120	47 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	> 450 Watt	Ditolak
121	43 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Ditolak
122	57 SD	Petani	< 1.000.000	Ya	50m2 - 100m2	Tanah	Bambu	450 Watt	Diterima
123	74 SD	Petani	< 1.000.000	Tidak	< 50m2	Tanah	Kayu	Bukan Listrik Pribadi	Diterima
124	77 Tidak Sekolah	Tidak Bekerja	< 1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
125	39 SMP	Lainnya	1.000.000 - 2.000.000	Ya	> 100m2	Tanah	Tembok	> 450 Watt	Diterima
126	26 SMP	Petani	< 1.000.000	Ya	50m2 - 100m2	Semen	Kayu	> 450 Watt	Diterima
127	34 SMP	Lainnya	> 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	> 450 Watt	Diterima
128	48 Tidak Sekolah	Petani	< 1.000.000	Ya	50m2 - 100m2	Tanah	Tembok	> 450 Watt	Diterima
129	46 SD	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Bambu	450 Watt	Diterima
130	45 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
131	42 SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
132	42 SD	Petani	< 1.000.000	Ya	50m2 - 100m2	Tanah	Bambu	> 450 Watt	Diterima
133	41 SD	Buruh Harian	< 1.000.000	Ya	50m2 - 100m2	Keramik	Tembok	> 450 Watt	Diterima
134	32 SMP	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	> 450 Watt	Diterima
135	45 SD	Petani	< 1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
136	44 SMP	Lainnya	< 1.000.000	Ya	50m2 - 100m2	Keramik	Tembok	> 450 Watt	Diterima
137	41 SD	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
138	41 SD	Petani	< 1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima

139	39	SMP	Buruh Harian	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
140	38	SMP	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
141	35	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Tembok	Bukan Listrik Pribadi	Diterima
142	51	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Ditolak
143	40	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
144	83	Tidak Sekolah	Tidak Bekerja	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
145	62	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
146	73	Tidak Sekolah	Tidak Bekerja	<1.000.000	Tidak	<50m2	Tanah	Kayu	Bukan Listrik Pribadi	Diterima
147	64	Tidak Sekolah	Petani	<1.000.000	Tidak	<50m2	Tanah	Bambu	Bukan Listrik Pribadi	Diterima
148	62	Tidak Sekolah	Petani	<1.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
149	74	Tidak Sekolah	Tidak Bekerja	<1.000.000	Tidak	<50m2	Tanah	Kayu	450 Watt	Diterima
150	53	Tidak Sekolah	Pedagang	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
151	50	SD	Buruh Harian	<1.000.000	Ya	50m2 - 100m2	Semen	Tembok	450 Watt	Diterima
152	48	SMP	Buruh Harian	<1.000.000	Tidak	50m2 - 100m2	Tanah	Bambu	>450 Watt	Diterima
153	62	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	Bukan Listrik Pribadi	Diterima
154	47	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
155	45	SD	Pedagang	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
156	36	SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima
157	72	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Bambu	450 Watt	Diterima
158	43	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
159	33	SMP	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
160	76	Tidak Sekolah	Petani	<1.000.000	Tidak	<50m2	Tanah	Kayu	Bukan Listrik Pribadi	Diterima
161	33	SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	Bukan Listrik Pribadi	Diterima
162	33	SMP	Lainnya	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
163	68	Tidak Sekolah	Petani	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
164	44	SD	Lainnya	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
165	38	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
166	72	Tidak Sekolah	Petani	<1.000.000	Tidak	<50m2	Tanah	Bambu	450 Watt	Diterima
167	67	SD	Lainnya	<1.000.000	Ya	50m2 - 100m2	Tanah	Bambu	Bukan Listrik Pribadi	Diterima
168	27	SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Tembok	>450 Watt	Diterima
169	40	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima
170	48	SD	Pedagang	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Tembok	>450 Watt	Diterima
171	49	SD	Petani	<1.000.000	Ya	<50m2	Tanah	Kayu	>450 Watt	Diterima
172	52	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
173	59	Tidak Sekolah	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
174	70	SD	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
175	55	Tidak Sekolah	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
176	62	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Bambu	Bukan Listrik Pribadi	Diterima
177	64	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Bambu	Bukan Listrik Pribadi	Diterima
178	47	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Bambu	450 Watt	Diterima
179	48	Tidak Sekolah	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
180	45	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Bambu	>450 Watt	Diterima
181	44	SD	Lainnya	<1.000.000	Ya	50m2 - 100m2	Keramik	Kayu	450 Watt	Diterima
182	44	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
183	43	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
184	43	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Bambu	450 Watt	Diterima
185	49	Tidak Sekolah	Lainnya	<1.000.000	Tidak	<50m2	Tanah	Kayu	450 Watt	Diterima
186	41	Tidak Sekolah	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Tembok	450 Watt	Diterima
187	75	Tidak Sekolah	Petani	<1.000.000	Tidak	<50m2	Tanah	Kayu	>450 Watt	Diterima
188	66	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
189	43	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Keramik	Tembok	>450 Watt	Diterima
190	74	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
191	42	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
192	39	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
193	37	SMP	Buruh Harian	>2.000.000	Ya	50m2 - 100m2	Keramik	Kayu	>450 Watt	Diterima
194	36	SD	Lainnya	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
195	68	SD	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
196	67	SD	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
197	65	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
198	54	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
199	43	SD	Lainnya	1.000.000 - 2.000.000	Ya	>100m2	Semen	Kayu	>450 Watt	Diterima
200	43	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Keramik	Kayu	>450 Watt	Diterima
201	42	SD	Lainnya	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Keramik	Kayu	>450 Watt	Ditolak
202	41	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
203	40	SD	Petani	>2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
204	36	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
205	29	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
206	68	Tidak Sekolah	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
207	41	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
208	34	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima

209	83	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
210	71	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	Bukan Listrik Pribadi	Diterima
211	70	Tidak Sekolah	Petani	<1.000.000	Tidak	50m2 - 100m2	Tanah	Kayu	Bukan Listrik Pribadi	Diterima
212	42	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
213	40	SD	Lainnya	>2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
214	36	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
215	33	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
216	38	SD	Petani	>2.000.000	Ya	50m2 - 100m2	Keramik	Kayu	>450 Watt	Diterima
217	34	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Keramik	Kayu	>450 Watt	Diterima
218	29	SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
219	29	SMA	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Keramik	Kayu	>450 Watt	Diterima
220	35	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima
221	36	SMP	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
222	43	SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima
223	46	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Tembok	450 Watt	Ditolak
224	49	SMP	Lainnya	>2.000.000	Ya	>100m2	Keramik	Tembok	>450 Watt	Ditolak
225	43	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Keramik	Kayu	>450 Watt	Diterima
226	36	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Tanah	Kayu	>450 Watt	Diterima
227	27	SMA	Lainnya	>2.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima
228	33	SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Keramik	Kayu	450 Watt	Diterima
229	40	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Tembok	>450 Watt	Diterima
230	52	SD	Petani	<1.000.000	Ya	50m2 - 100m2	Tanah	Kayu	450 Watt	Diterima
231	82	Tidak Sekolah	Petani	<1.000.000	Tidak	<50m2	Tanah	Kayu	450 Watt	Diterima
232	36	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
233	36	SMP	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima
234	41	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	>450 Watt	Diterima
235	42	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima
236	47	SD	Buruh Harian	1.000.000 - 2.000.000	Ya	50m2 - 100m2	Semen	Kayu	450 Watt	Diterima

Lampiran 3. Data Testing Proporsi 90:10

USIA	PENDIDIKAN	PEKERJAAN	PENGHASILAN	ASET	LUAS, LANTAI	JENIS, LANTAI	DINDING	PENERANGAN	class
49	2	1	1	1	1	1	3	1	2
51	3	4	1	1	1	3	2	2	2
41	3	2	3	1	2	2	2	2	2
47	3	4	2	1	2	3	2	1	2
50	2	2	1	1	1	2	3	1	2
51	3	4	3	1	2	2	2	2	2
48	3	4	3	1	2	3	2	1	2
43	3	3	1	1	2	2	2	1	2
50	3	4	2	1	1	3	2	2	2
42	3	1	2	1	2	1	2	1	2
43	3	1	2	1	2	2	3	1	2
46	3	2	2	1	2	2	2	1	2
46	3	2	2	1	2	2	3	2	2
43	3	2	2	1	2	2	2	2	2
51	3	4	1	1	1	3	2	2	2
51	3	2	2	1	2	3	2	2	2
45	3	2	2	1	2	1	3	2	2
42	3	3	1	1	2	2	2	1	2
45	3	2	2	1	2	2	3	2	2
46	3	2	2	1	2	2	3	2	2
47	3	4	2	1	2	3	2	1	2
64	3	4	3	1	2	3	2	2	1
55	3	4	3	1	2	3	2	1	1
37	2	1	2	1	2	3	2	1	1
36	2	2	3	1	2	1	2	2	1
36	3	1	3	1	2	3	2	2	1
35	3	2	2	1	2	3	1	2	1
30	1	2	2	1	2	3	2	2	1
42	3	4	3	1	2	3	2	2	1
36	3	2	2	1	2	3	2	2	1
45	3	3	2	1	2	3	2	2	1
46	3	4	2	1	2	3	2	1	1
63	4	5	3	2	3	3	1	3	1
41	3	2	2	1	2	2	2	2	1
32	3	2	2	2	2	2	2	1	1
77	4	5	3	2	2	3	2	2	1
34	2	1	1	1	2	3	2	1	1
48	4	4	3	1	2	3	2	2	1
42	3	4	3	1	2	3	2	1	1
68	3	4	3	2	2	3	2	2	1
43	3	2	2	1	2	1	2	1	1
71	4	4	3	2	2	3	2	3	1
70	4	4	3	2	2	3	2	3	1

Lampiran 4. Script Software RStudio

0.1. Preprocecing Data

```

1 #Memanggil data
2 data = read.delim("clipboard")
3 view(data)
4 head(data)
5
6 #Summary data
7 summary(data)
8
9 #Structure dari data
10 str(data)
11
12 #Melihat missing value
13 colsums(is.na(data)) #mengetahui jumlah missing value
14
15
16 #Melihat Redudant data dengan analisis korelasi
17 #Pendeteksian redundant
18 #Melihat koef. korelasi antara dua variabel secara bersama-sama
19 require(graphics)
20 pairs(data)
21 cor(data)

```

0.2. SMOTE

```

#oversampling dengan metode SMOTE
library(smotefamily)

#mengubah tipe data character menjadi factor (transformasi data)
data$PENDIDIKAN <- as.factor(data$PENDIDIKAN)
data$PEKERJAAN <- as.factor(data$PEKERJAAN)
data$PENGHASILAN <- as.factor(data$PENGHASILAN)
data$ASET <- as.factor(data$ASET)
data$LUAS.LANTAI <- as.factor(data$LUAS.LANTAI)
data$JENIS.LANTAI <- as.factor(data$JENIS.LANTAI)
data$DINDING <- as.factor(data$DINDING)
data$PENERANGAN <- as.factor(data$PENERANGAN)
data$STATUS <- as.factor(data$STATUS)
str(data)

> table(data$STATUS)

Diterima Ditolak
  221      15

> prop.table(table(data$STATUS))

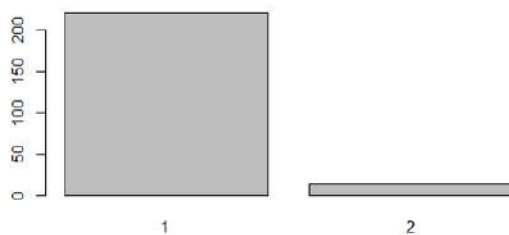
Diterima Ditolak
0.93644068 0.06355932

> target_variabel <- "STATUS"
> predictor_variables <- setdiff(names(data), target_variabel)
> table(data$STATUS)

Diterima Ditolak
  221      15

> barplot(table(data$STATUS))

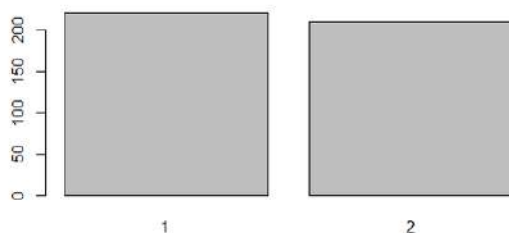
```



```
> #Mengubah tipe data factor menjadi numerik
> data$PENDIDIKAN <- as.numeric(data$PENDIDIKAN)
> data$PEKERJAAN <- as.numeric(data$PEKERJAAN)
> data$PENGHASILAN <- as.numeric(data$PENGHASILAN)
> data$ASET <- as.numeric(data$ASET)
> data$LUAS.LANTAI <- as.numeric(data$LUAS.LANTAI)
> data$JENIS.LANTAI <- as.numeric(data$JENIS.LANTAI)
> data$DINDING <- as.numeric(data$DINDING)
> data$PENERANGAN <- as.numeric(data$PENERANGAN)
> data$STATUS <- as.numeric(data$STATUS)
> smote=SMOTE(data[,-10],data$STATUS)
> smote=smote$data
> table(smote$class)

 1  2
221 210
> prop.table(table(smote$class))

      1      2
0.512761 0.487239
> barplot(table(smote$class))
> View(smote)
```



Lampiran 5. *Script dan Output Software RStudio Algoritma C5.0*

0.3. Proporsi Data 50:50

```
> #Pembagian data training dan testing
> n <- round(nrow(smote)*0.50);n
[1] 216
> set.seed(99191)
> samp=sample(1:nrow(smote),n)
> data.train=smote[samp,]
> dim(data.train)
[1] 216 10
> data.test=smote[~samp,]
> dim(data.test)
[1] 215 10
> str(data.test$class)
Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 ...
> #Create decision tree c5.0 tree-based
> library(C50)
> model <- C5.0(class~, data=data.train)
> model

Call:
C5.0.formula(formula = class ~ ., data = data.train)

Classification Tree
Number of samples: 216
Number of predictors: 9

Tree size: 14

Non-standard options: attempt to group attributes

> plot(model)
> summary(model)

Call:
C5.0.formula(formula = class ~ ., data = data.train)

Classification Tree
Number of samples: 216
Number of predictors: 9

Tree size: 14

Non-standard options: attempt to group attributes

> plot(model)
> summary(model)

Call:
C5.0.formula(formula = class ~ ., data = data.train)

C5.0 [Release 2.07 GPL Edition] Tue May 21 11:31:06 2024
-----

Class specified by attribute 'outcome'

Read 216 cases (10 attributes) from undefined.data

Decision tree:
```



```

        Accuracy : 0.8279
        95% CI : (0.7707, 0.8758)
        No Information Rate : 0.5302
        P-value [Acc > NIR] : <2e-16

        Kappa : 0.654

        Mcnemar's Test P-value : 0.7423

        Sensitivity : 0.8020
        Specificity : 0.8509
        Pos Pred Value : 0.8265
        Neg Pred Value : 0.8291
        Prevalence : 0.4698
        Detection Rate : 0.3767
        Detection Prevalence : 0.4558
        Balanced Accuracy : 0.8264

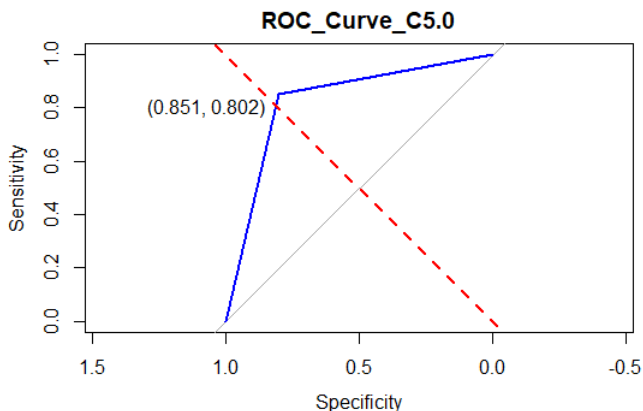
        'Positive' Class : 1

> ##Membuat kurva ROC algoritma C5.0
> #konversi variabel target ke format biner
> binary_truth <- ifelse(data.test$class == "2",1,0)
> binary_prediction <- ifelse(class_predict == "2",1,0)
> length(binary_prediction)
[1] 215
> length(binary_truth)
[1] 215
> library(ROC)
> #Hitung ROC
> roc_curve <- roc(binary_truth, binary_prediction)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> roc_curve

Call:
roc.default(response = binary_truth, predictor = binary_prediction)

Data: binary_prediction in 101 controls (binary_truth 0) < 114 cases (binary_truth 1).
Area under the curve: 0.8264
> #Hitung AUC
> auc_value <- auc(roc_curve)
> auc_value
Area under the curve: 0.8264
> #Gambar kurva ROC algoritma C5.0
> plot(roc_curve, main = "ROC_curve_C5.0", col = "blue", lwd = 2)
> abline(a=0, b=1, col = "red", lwd = 2, lty = 2)
> text(0.851, 0.802, "(0.851, 0.802)", col = "black", cex=1.1)

```



```
#Pembagian data training dan testing
> n <- round(nrow(smote)*0.60);n
[1] 259
> set.seed(99191)
> samp=sample(1:nrow(smote),n)
> data.train=smote[samp,]
> dim(data.train)
[1] 259 10
> View(data.train)
> data.test=smote[-samp,]
> dim(data.test)
[1] 172 10
> View(data.test)
> str(data.test$class)
str(data.test$class)
 factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 ...
> #Create decision tree c5.0 tree-based
> library(C50)
warning message:
package 'C50' was built under R version 4.3.3
> model <- C5.0(class ~., data=data.train)
> model

Call:
C5.0.formula(formula = class ~., data = data.train)

Classification Tree
Number of samples: 259
Number of predictors: 9

Tree size: 18

Non-standard options: attempt to group attributes

> plot(model)
> summary(model)

Call:
C5.0.formula(formula = class ~., data = data.train)

C5.0 [Release 2.07 GPL Edition] Fri Mar 29 2
-----
class specified by attribute 'outcome'

Read 259 cases (10 attributes) from undefined.data

Decision tree:
```

```

PENDIDIKAN in {1,4}: 1 (33)
PENDIDIKAN in {2,3}:
...USIA <= 40:
...PEKERJAAN in {2,3,4,5}: 1 (47/1)
...PEKERJAAN = 1:
...: ...PENGHASILAN = 1: 2 (3)
...: ...PENGHASILAN in {2,3}: 1 (8/2)
USIA > 40:
...DINDING = 1: 1 (14)
DINDING in {2,3}:
...ASET = 2: 1 (7)
ASET = 1:
...LUAS.LANTAI = 3: 1 (1)
LUAS.LANTAI = 1:
...USIA <= 45: 1 (2)
...USIA > 45: 2 (17)
LUAS.LANTAI = 2:
...PENDIDIKAN = 2: 1 (3)
PENDIDIKAN = 3:
...JENIS.LANTAI in {1,2}: 2 (72/12)
JENIS.LANTAI = 3:
...USIA <= 46: 1 (10)
USIA > 46:
...USIA > 52: 1 (3)
USIA <= 52:
...DINDING = 3: 1 (1)
DINDING = 2:
...PENERANGAN in {1,3}: 2 (27/1)
PENERANGAN = 2:
...PENGHASILAN in {1,3}: 1 (3)
PENGHASILAN = 2:
...USIA <= 51: 1 (2)
USIA > 51: 2 (6)

```

```

        Accuracy : 0.8279
        95% CI : (0.7707, 0.8758)
    No Information Rate : 0.5302
    P-value [Acc > NIR] : <2e-16

    Kappa : 0.654

    Mcnemar's Test P-value : 0.7423

    Sensitivity : 0.8020
    Specificity : 0.8509
    Pos Pred Value : 0.8265
    Neg Pred Value : 0.8291
    Prevalence : 0.4698
    Detection Rate : 0.3767
    Detection Prevalence : 0.4558
    Balanced Accuracy : 0.8264

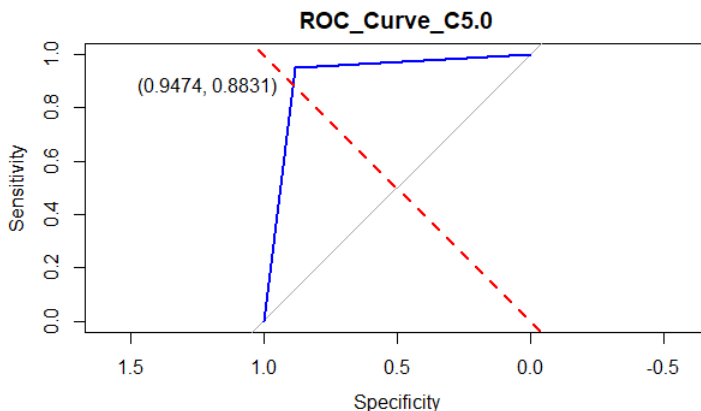
    'Positive' Class : 1

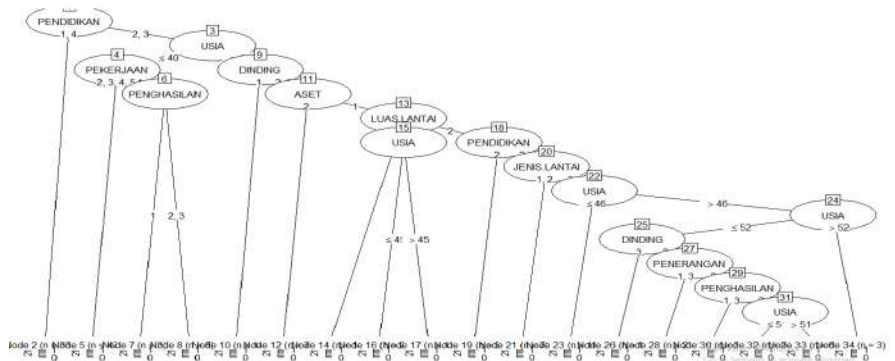
> #Membuat kurva ROC algoritma C5.0
> #konversi variabel target ke format biner
> binary_truth <- ifelse(data.test$class == "2",1,0)
> binary_prediction <- ifelse(class_predict == "2",1,0)
> length(binary_prediction)
[1] 215
> length(binary_truth)
[1] 215
> library(ROC)
> #Hitung ROC
> roc_curve <- roc(binary_truth, binary_prediction)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> roc_curve

Call:
roc.default(response = binary_truth, predictor = binary_prediction)

Data: binary_prediction in 101 controls (binary_truth 0) < 114 cases (binary_truth 1).
Area under the curve: 0.8264
> #Hitung AUC
> auc_value <- auc(roc_curve)
> auc_value
Area under the curve: 0.8264
> #Gambar kurva ROC algoritma C5.0
> plot(roc_curve, main = "ROC_curve_C5.0", col = "blue", lwd = 2)
> abline(a=0, b=1, col = "red", lwd = 2, lty = 2)
> text(0.851, 0.802, "(0.851, 0.802)", col = "black", cex=1.1)

```





0.5. Proporsi Data 70:30

```
> #Pembagian data training dan testing
> n <- round(nrow(smote)*0.70);n
[1] 302
> set.seed(99191)
> samp=sample(1:nrow(smote),n)
> data.train=smote[samp,]
> dim(data.train)
[1] 302 10
> view(data.train)
> data.test=smote[-samp,]
> dim(data.test)
[1] 129 10
> view(data.test)
> str(data.test$class)
Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 ...
> #create decision tree c5.0 tree-based
> library(c50)
> model <- c5.0(class=., data=data.train)
> model

call:
c5.0(formula = class ~., data = data.train)

Classification tree
Number of samples: 302
Number of predictors: 9

Tree size: 20

Non-standard options: attempt to group attributes

> plot(model)
> summary(model)

call:
c5.0(formula = class ~., data = data.train)

C5.0 [release 2.07 GPL Edition] Tue May 21 19:43:20 2024

Class specified by attribute 'outcome'

Read 302 cases (10 attributes) from undefined.data

decision tree:

PENDIDIKAN in {1,4}: 1 (37)
PENDIDIKAN in {2,3}:
...:USIA <= 40:
...:PEKERJAAN in {2,3,4,5}: 1 (50/1)
: PEKERJAAN = 1:
: ...:PENGHASILAN in {2,3}: 1 (9/2)
: PENGHASILAN = 1:
: ...:PENERANGAN in {1,2}: 2 (4)
: PENERANGAN = 3: 1 (1)
```

```

USIA > 40:
:...DINDING = 1: 1 (14)
  DINDING in {2,3}:
:...ASET = 2: 1 (7)
    ASET = 1:
      :...LUAS.LANTAI = 3: 1 (1)
        LUAS.LANTAI = 1:
          :...USIA <= 45: 1 (2)
            : USIA > 45: 2 (19)
              LUAS.LANTAI = 2:
                :...PENDIDIKAN = 2: 1 (3)
                  PENDIDIKAN = 3:
                    :...JENIS.LANTAI in {1,2}: 2 (94/15)
                      JENIS.LANTAI = 3:
                        :...USIA <= 46: 1 (15)
                          USIA > 46:
                            :...USIA > 52: 1 (3)
                              USIA <= 52:
                                :...PENERANGAN = 3: 2 (0)
                                  PENERANGAN = 1:
                                    :...DINDING = 2: 2 (30/1)
                                      : DINDING = 3: 1 (1)
                                        PENERANGAN = 2:
                                          :...USIA <= 51: 1 (4)
                                            USIA > 51:
                                              :...PENGHASILAN in {1,2}: 2 (6)
                                                PENGHASILAN = 3: 1 (2)

```

Evaluation on training data (302 cases):

```

Decision Tree
-----
size      errors

  19    19( 6.3%)  <<

(a)  (b)  <-classified as
----  ----
146    16  (a): class 1
 3    137  (b): class 2

Attribute usage:

100.00% PENDIDIKAN
 87.75% USIA
 66.56% DINDING
 61.92% ASET
 59.60% LUAS.LANTAI
 51.32% JENIS.LANTAI
 21.19% PEKERJAAN
 15.89% PENERANGAN
  7.28% PENGHASILAN

Time: 0.0 secs

> #Melakukan prediksi pada data pengujian
> predictions <- predict(model, data.test)
> #Akurasi model
> library(caret)
> confusion_matrix <- table(Actual=data.test$class, predicted=predictions)
> print(confusion_matrix)
      predicted
Actual 1  2
      1 54  5
      2  5 65

> class_predict <- predict(model, newdata = data.test, type = "class")
> class_predict
[1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2
[41] 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1
[81] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 2 1 1 1 1 2 1 1 1 1 1 1 1
[121] 1 2 1 1 1 2 1 1 1
Levels: 1 2
> confusionMatrix(as.factor(class_predict), as.factor(data.test$class), positive='1')
Confusion Matrix and Statistics

      Reference
Prediction 1  2
      1 54  5
      2  5 65

```

```

      Accuracy : 0.9225
      95% CI   : (0.8621, 0.9622)
No Information Rate : 0.5426
P-value [Acc > NIR] : <2e-16

      Kappa : 0.8438

McNemar's Test P-value : 1

      Sensitivity : 0.9153
      specificity : 0.9286
      Pos Pred Value : 0.9153
      Neg Pred Value : 0.9286
      Prevalence : 0.4574
      Detection Rate : 0.4186
      Detection Prevalence : 0.4574
      Balanced Accuracy : 0.9219

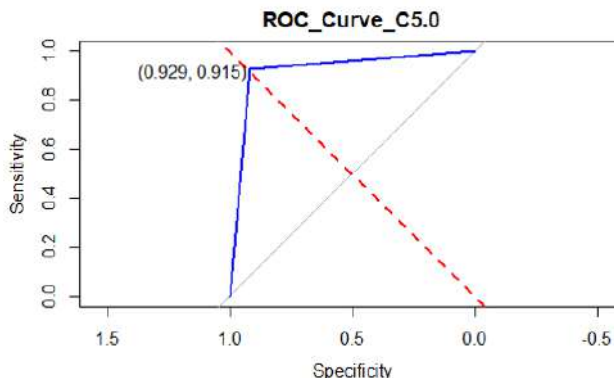
'Positive' Class : 1

> #membuat kurva ROC algoritma C5.0
> #konversi variabel target ke format biner
> binary_truth <- ifelse(data.test$class == "2",1,0)
> binary_prediction <- ifelse(class_predict == "2",1,0)
> length(binary_prediction)
[1] 129
> length(binary_truth)
[1] 129
> library(pROC)
> #hitung ROC
> roc_curve <- roc(binary_truth, binary_prediction)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> roc_curve

Call:
roc.default(response = binary_truth, predictor = binary_prediction)

data: binary_prediction in 59 controls (binary_truth 0) < 70 cases (binary_truth 1).
Area under the curve: 0.9219
> #hitung AUC
> auc_value <- auc(roc_curve)
> auc_value
Area under the curve: 0.9219
> #gambar kurva ROC algoritma C5.0
> plot(roc_curve, main = "ROC_Curve_C5.0", col = "blue", lwd = 2)
> abline(a=0, b=1, col = "red", lwd = 2, lty = 2)
> text(0.9228, 0.9153, "(0.9228, 0.9153)", col = "black", cex=1.1)

```



```
#Pembagian data training dan testing
> n <- round(nrow(smote)*0.80);n
[1] 345
> set.seed(99191)
> samp=sample(1:nrow(smote),n)
> data.train=smote[samp,]
> dim(data.train)
[1] 345 10
> View(data.train)
> data.test=smote[-samp,]
> dim(data.test)
[1] 86 10
> View(data.test)
> str(data.test$class)
Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...
> #Create decision tree c5.0 tree-based
> library(C50)
> model <- C5.0(class=., data=data.train)
> model

call:
C5.0.formula(formula = class ~ ., data = data.train)

Classification Tree
Number of samples: 345
Number of predictors: 9

Tree size: 16

Non-standard options: attempt to group attributes

> plot(model)
> summary(model)

Call:
C5.0.formula(formula = class ~ ., data = data.train)
```

Decision tree:

```

PENDIDIKAN in {1,4}: 1 (40)
PENDIDIKAN in {2,3}:
...USIA <= 40:
...PEKERJAAN in {2,3,4,5}: 1 (57/1)
: PEKERJAAN = 1:
: ...PENERANGAN in {2,3}: 1 (4)
: PENERANGAN = 1:
: ...PENGHASILAN = 1: 2 (6)
: PENGHASILAN in {2,3}: 1 (9/3)
USIA > 40:
...DINDING = 1: 1 (16)
DINDING in {2,3}:
...ASET = 2: 1 (7)
ASET = 1:
...JENIS.LANTAI in {1,2}: 2 (125/18)
JENIS.LANTAI = 3:
...USIA <= 46: 1 (22)
USIA > 46:
...USIA > 52: 1 (3)
USIA <= 52:
...LUAS.LANTAI = 1: 2 (8)
LUAS.LANTAI = 3: 1 (1)
LUAS.LANTAI = 2:
...PENERANGAN in {1,3}: 2 (32/2)
PENERANGAN = 2:
...USIA <= 48: 1 (4)
USIA > 48:
...PENGHASILAN in {1,2}: 2 (9/1)
PENGHASILAN = 3: 1 (2)

```

```

      Accuracy : 0.9302
      95% CI   : (0.8543, 0.974)
    No Information Rate : 0.5465
    P-Value [Acc > NIR] : 4.566e-15

      Kappa : 0.8586

  Mcnemar's Test P-Value : 0.6831

      Sensitivity : 0.8974
      Specificity : 0.9574
      Pos Pred Value : 0.9459
      Neg Pred Value : 0.9184
      Prevalence : 0.4535
      Detection Rate : 0.4070
      Detection Prevalence : 0.4302
      Balanced Accuracy : 0.9274

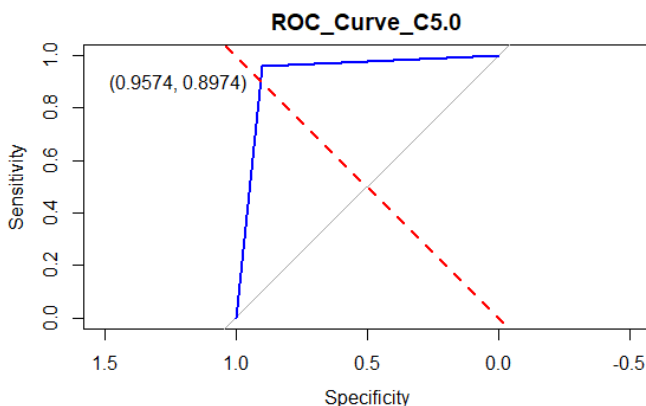
'Positive' Class : 1

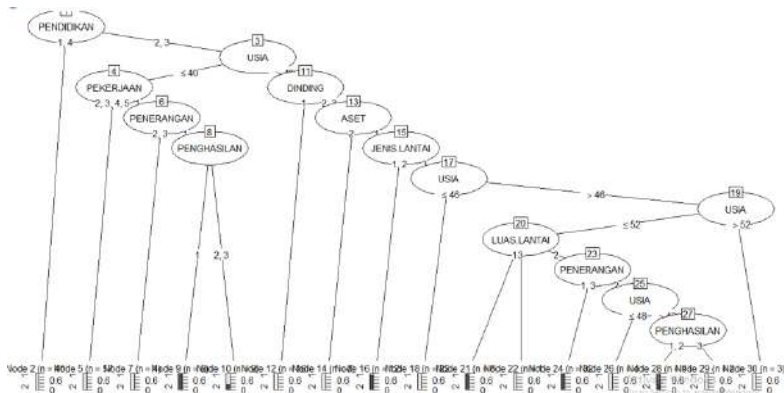
> #membuat kurva ROC algoritma C5.0
> #konversi variabel target ke format biner
> binary_truth <- ifelse(data.test$class == "2",1,0)
> binary_prediction <- ifelse(class_predict == "2",1,0)
> length(binary_prediction)
[1] 86
> length(binary_truth)
[1] 86
> library(roc)
> #hitung ROC
> roc_curve <- roc(binary_truth, binary_prediction)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> roc_curve

Call:
roc.default(response = binary_truth, predictor = binary_prediction)

Data: binary_prediction in 39 controls (binary_truth 0) < 47 cases (binary_truth 1).
Area under the curve: 0.9274
> #hitung AUC
> auc_value <- auc(roc_curve)
> auc_value
Area under the curve: 0.9274
> #gambar kurva ROC algoritma C5.0
> plot(roc_curve, main = "ROC_Curve_C5.0", col = "blue", lwd = 2)
> abline(a=0, b=1, col = "red", lwd = 2, lty = 2)
> text(0.9574, 0.8974, "(0.9574, 0.8974)", col = "black", cex=1.1)

```





0.7. Proporsi Data 90:10

```
> #Pembagian data training dan testing
> n <- round(nrow(smote)*0.90);n
[1] 388
> set.seed(99191)
> samp=sample(1:nrow(smote),n)
> data.train=smote[samp,]
> dim(data.train)
[1] 388 10
> view(data.train)
> data.test=smote[-samp,]
> dim(data.test)
[1] 43 10
> view(data.test)
> str(data.test$class)
Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 ...
> #Create decision tree c5.0 tree-based
> library(c50)
> model <- C5.0(class=., data=data.train)
> model

call:
C5.0.formula(formula = class ~ ., data = data.train)

Classification Tree
Number of samples: 388
Number of predictors: 9

Tree size: 19

Non-standard options: attempt to group attributes

> plot(model)
> summary(model)

call:
C5.0.formula(formula = class ~ ., data = data.train)

C5.0 [Release 2.07 GPL Edition] Tue May 21 10:10:36 2024

-----

Class specified by attribute 'outcome'

Read 388 cases (10 attributes) from undefined.data

Decision tree:
```

```

PENDIDIKAN in {1,4}: 1 (43)
PENDIDIKAN in {2,3}:
...USIA <= 40:
...PENGHASILAN in {2,3}: 1 (69/4)
:   PENGHASILAN = 1:
:   ...PENDIDIKAN = 3: 1 (4)
:   PENDIDIKAN = 2:
:   ...JENIS.LANTAI in {1,2}: 1 (3)
:   JENIS.LANTAI = 3: 2 (10/1)
USIA > 40:
...DINDING = 1: 1 (17)
DINDING in {2,3}:
...ASET = 2: 1 (9)
ASET = 1:
...JENIS.LANTAI in {1,2}:
...DINDING = 2: 2 (78/14)
:   DINDING = 3:
:   ...PENGHASILAN in {1,2}: 2 (62/2)
:   PENGHASILAN = 3: 1 (3)
JENIS.LANTAI = 3:
...USIA <= 46: 1 (23)
USIA > 46:
...USIA > 52: 1 (4)
USIA <= 52:
...DINDING = 3: 1 (1)
DINDING = 2:
...LUAS.LANTAI = 1: 2 (9)
LUAS.LANTAI = 3: 1 (1)
LUAS.LANTAI = 2:
...PENERANGAN in {1,3}: 2 (34/1)
PENERANGAN = 2:
...PENGHASILAN in {1,3}: 1 (4)
PENGHASILAN = 2:
...USIA <= 51: 1 (5/1)
USIA > 51: 2 (9)

```

```

                Accuracy : 0.907
                95% CI : (0.7786, 0.9741)
    No Information Rate : 0.5116
    P-Value [Acc > NIR] : 3.485e-08

                Kappa : 0.8143

McNemar's Test P-Value : 0.6171

                Sensitivity : 0.8636
                Specificity : 0.9524
                Pos Pred Value : 0.9500
                Neg Pred Value : 0.8696
                Prevalence : 0.5116
                Detection Rate : 0.4419
    Detection Prevalence : 0.4651
    Balanced Accuracy : 0.9080

'Positive' Class : 1

> #Membuat kurva ROC algoritma C5.0
> #konversi variabel target ke format biner
> binary_truth <- ifelse(data.test$class == "2",1,0)
> binary_prediction <- ifelse(class_predict == "2",1,0)
> length(binary_prediction)
[1] 43
> length(binary_truth)
[1] 43
> library(ROC)
> #Hitung ROC
> roc_curve <- roc(binary_truth, binary_prediction)
> roc_curve

Call:
roc.default(response = binary_truth, predictor = binary_prediction)

Data: binary_prediction in 22 controls (binary_truth 0) < 21 cases (binary_truth 1),
Area under the curve: 0.908
> #Hitung AUC
> auc_value <- auc(roc_curve)
> auc_value
Area under the curve: 0.908
> #Gambar kurva ROC algoritma C5.0
> plot(roc_curve, main = "ROC_curve_C5.0", col = "blue", lwd = 2)
> abline(a=0, b=1, col = "red", lwd = 2, lty = 2)
> text(0.952, 0.864, "(0.952, 0.864)", col = "black", cex=1.1)

```

Lampiran 6. Script dan Output Software RStudio Algoritma CART

0.8. Proporsi Data 50:50

```
> #classification using CART
> library(rpart) #model CART
> library(rpart.plot)
> #visualisasi decision tree
> library(MASS) #untuk data birchwt untuk variabel y kualitatif
> library(cppplot2) #untuk data msleep untuk variabel y kuantitatif
> #create decision tree cart tree-based
> cartTree <- rpart(class~., data=data.train, method = 'class') #method yang dipilih 'class'
karena bertipe faktor
> printcp(cartTree)

Classification tree:
rpart(formula = class ~ ., data = data.train, method = "class")

Variables actually used in tree construction:
[1] DINDING JENIS.LANTAI USIA

Root node error: 96/216 = 0.44444

n= 216

      cp nsplit rel error  xerror  xstd
1 0.244792      0 1.00000 1.00000 0.076073
2 0.104167      2 0.51042 0.55208 0.065877
3 0.072817      3 0.40625 0.48958 0.063168
4 0.010000      5 0.26042 0.34375 0.055079
> #menampilkan hasil tree-based model
> cartTree
n= 216

node), split, n, loss, yval, (yprob)
      * denotes terminal node
1) root 216 96 1 (0.5555556 0.4444444)
 2) USIA< 40.5 94 5 1 (0.9074071 0.0925929) *
 3) USIA>=40.5 162 71 2 (0.4382716 0.5617284)
   6) USIA>=52.5 27 0 1 (1.0000000 0.0000000) *
   7) USIA< 52.5 135 44 2 (0.3259259 0.6740741) *
    14) DINDING=1 10 0 1 (1.0000000 0.0000000) *
    15) DINDING=2,3 125 34 2 (0.2720000 0.7280000)
       30) JENIS.LANTAI=3 53 22 2 (0.4150943 0.5849057)
        60) USIA< 46 14 0 1 (1.0000000 0.0000000) *
        61) USIA>=46 39 8 2 (0.2051182 0.7948818) *
       31) JENIS.LANTAI=1,2 72 12 2 (0.1666667 0.8333333) *
> plot(cartTree)
> summary(cartTree)
> #visualisasi dengan rpart.plot
> rpart.plot(cartTree, type=4, digits=5, fallen.leaves=FALSE, shadow.col="gray", box.palette
="bugn")
> #akurasi model
> cartPred = predict(cartTree, data.test, type = 'class')
> table(cartPred, data.test$class) #confusion matrix dari data testing

cartPred      1      2
      1 83      8
      2 18 106

> confusionMatrix(as.factor(cartPred), as.factor(data.test$class), positive='1')
Confusion Matrix and Statistics

      Reference
Prediction      1      2
      1 83      8
      2 18 106
```

```

      Accuracy : 0.8791
      95% CI   : (0.8278, 0.9195)
      No Information Rate : 0.5302
      P-Value [Acc > NIR] : < 2e-16

      Kappa : 0.7559

      Mcnemar's Test P-Value : 0.07756

      Sensitivity : 0.8218
      Specificity : 0.9298
      Pos Pred Value : 0.9121
      Neg Pred Value : 0.8548
      Prevalence : 0.4698
      Detection Rate : 0.3860
      Detection Prevalence : 0.4233
      Balanced Accuracy : 0.8758

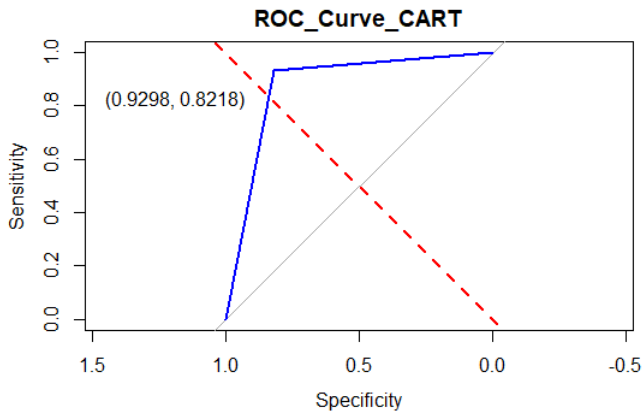
      'Positive' Class : 1

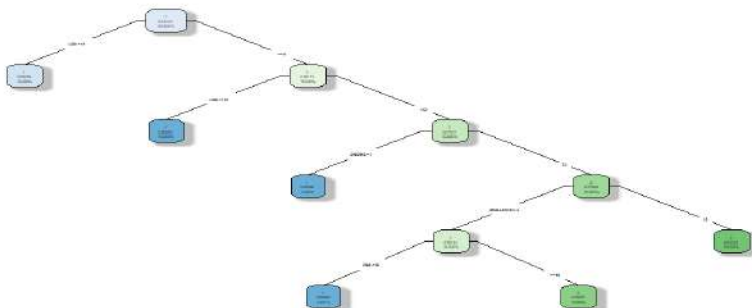
> #Membuat kurva ROC algoritma CART
> #konversi variabel target ke format biner
> binary_truth2 <- ifelse(data.test$class == "2",1,0)
> binary_prediction2 <- ifelse(cartPred == "2",1,0)
> length(binary_prediction2)
[1] 215
> length(binary_truth2)
[1] 215
> #Hitung ROC
> roc_curve <- roc(binary_truth2, binary_prediction2)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> roc_curve

Call:
roc.default(response = binary_truth2, predictor = binary_prediction2)

Data: binary_prediction2 in 101 controls (binary_truth2 0) < 114 cases (binary_truth2 1).
Area under the curve: 0.8758
> #Hitung AUC
> auc_value <- auc(roc_curve)
> auc_value
Area under the curve: 0.8758
> #Gambar kurva ROC algoritma CART
> plot(roc_curve, main = "ROC_Curve_CART", col = "blue", lwd = 2)
> abline(a=0, b=1, col = "red", lwd = 2, lty = 2)
> text(0.9298, 0.8218, "(0.9298, 0.8218)", col = "black", cex=1,1)

```





0.9. Proporsi Data 60:40

```
> #Classification using CART
> library(rpart) #model CART
> library(rpart.plot)
warning message:
package 'rpart.plot' was built under R version 4.2.2
> #visualisasi decision tree
> library(MASS) #untuk data birchwt untuk variabel y kualitatif
> library(ggplot2) #untuk data mslwp untuk variabel y kuantitatif
> #create decision tree cart tree-based
> cartTree <- rpart(class ~., data=data.train, method = "class") #method yang dipilih
#class' karena bertipe faktor
> printcp(cartTree)

Classification tree:
rpart(formula = class ~., data = data.train, method = "class")

Variables actually used in tree construction:
[1] DINDING JENIS.LANTAI PENDIDIKAN USIA

Root node error: 115/259 = 0.44402

n= 259

CP nsplit rel error xerror xstd
1 0.234783 0 1.00000 1.00000 0.069512
2 0.104348 2 0.53043 0.53043 0.059381
3 0.060879 3 0.42609 0.42609 0.054810
4 0.047826 4 0.38122 0.38113 0.050049
5 0.010000 6 0.26917 0.26965 0.047259
> #memanggilkan hasil tree-based model
> cartTree

n= 259

node), split, n, loss, yval, (yprob)
# denotes terminal node

1) root 259 115 1 (0.5559846 0.4440154)
2) USIA< 40.5 64 6 1 (0.9062500 0.0937500) *
3) USIA>=40.5 195 86 2 (0.4410256 0.5589744)
6) USIA>=52.5 31 0 1 (1.0000000 0.0000000) *
7) USIA< 52.5 164 55 2 (0.3353659 0.6646341)
14) DINDING=1 12 0 1 (1.0000000 0.0000000) *
15) DINDING=2,3 152 43 2 (0.2828947 0.7171053)
30) PENDIDIKAN=1,4 7 0 1 (1.0000000 0.0000000) *
31) PENDIDIKAN=2,3 145 36 2 (0.2482759 0.7517241)
62) USIA< 44.5 59 22 2 (0.3728814 0.6271186)
124) JENIS.LANTAI=3 11 0 1 (1.0000000 0.0000000) *
125) JENIS.LANTAI=1,2 48 11 2 (0.2291667 0.7708333) *
63) USIA>=44.5 86 14 2 (0.1627907 0.8372093) *
```

```

> #Visualisasi dengan rpart.plot
> rpart.plot(cartree, type=4, digits=5, fallen.leaves=FALSE, shadow.col="gray", bo
x.palette="bugn")
> #akurasi model
> cartPred = predict(cartree, data.test, type = 'class')
> table(cartPred, data.test$class) #confusion matrix dari data testing

cartPred 1 2
         1 60 7
         2 17 88
> confusionMatrix(as.factor(cartPred), as.factor(data.test$class), positive='1')
Confusion Matrix and Statistics

              Reference
Prediction 1 2
          1 60 7
          2 17 88

              Accuracy : 0.8605
              95% CI : (0.7995, 0.9085)
      No Information Rate : 0.5523
      P-Value [Acc > NIR] : < 2e-16

              Kappa : 0.7143

McNemar's Test P-value : 0.06619

      Sensitivity : 0.7792
      Specificity : 0.9263
      Pos Pred Value : 0.8955
      Neg Pred Value : 0.8381
      Prevalence : 0.4477
      Detection Rate : 0.3488
      Detection Prevalence : 0.3895
      Balanced Accuracy : 0.8528

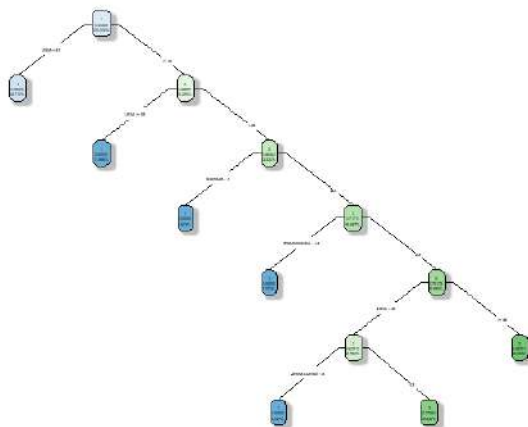
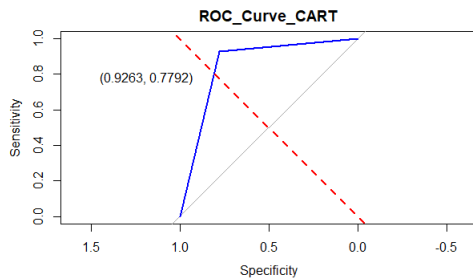
      'Positive' Class : 1

> #Membuat kurva ROC algoritma CART
> #Konversi variabel target ke format biner
> binary_truth2 <- ifelse(data.test$class == "2",1,0)
> binary_prediction2 <- ifelse(cartPred == "2",1,0)
> length(binary_prediction2)
[1] 172
> length(binary_truth2)
[1] 172
> #Hitung ROC
> roc_curve <- roc(binary_truth2, binary_prediction2)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> roc_curve

Call:
roc.default(response = binary_truth2, predictor = binary_prediction2)

Data: binary_prediction2 in 77 controls (binary_truth2 0) < 95 cases (binary_truth2
1).
Area under the curve: 0.8528
> #Hitung auc
> auc_value <- auc(roc_curve)
> auc_value
Area under the curve: 0.8528
> #Gambar kurva ROC algoritma CART
> plot(roc_curve, main = "ROC_curve_CART", col = "blue", lwd = 2)
> abline(a=0, b=1, col = "red", lwd = 2, lty = 2)
> text(0.9263, 0.7792, "(0.9263, 0.7792)", col = "black", cex=1,1)

```



0.10. Proporsi Data 70:30

```
## Classification using CART
> library(rpart) #model CART
> library(rpart.plot)
> #visualisasi decision tree
> library(MASS) #untuk data birthwt untuk variabel y kualitatif
> library(ggplot2) #untuk data mslsnp untuk variabel y kuantitatif
> #create decision tree cart tree-based
> cartree <- rpart(class ~., data=data.train, method = "class") #method yang dipilih 'class'
# karena bertipe faktor
> printcp(cartree)

Classification trees:
rpart(formula = class ~., data = data.train, method = "class")

Variables actually used in tree construction:
[1] DINDING    JENIS.LAKAI PEKERJAAN  PENDIDIKAN  USIA

Root node error: 140/302 = 0.46358

n= 302

CP nsplit rel error  xerror  xstd
1 0.246429 0 1.00000 1.00000 0.061900
2 0.085716 2 0.50714 0.65714 0.057131
3 0.039524 3 0.42143 0.41571 0.049816
4 0.011301 6 0.24586 0.20386 0.047519
5 0.010000 9 0.20714 0.30714 0.043376
> #menampilkan hasil tree-based model
> cartree
```

```

n= 302

node): split, n, loss, yval, (yprob)
      * denotes terminal node

1) root 302 140 1 (0.5364238 0.4635762)
  2) USIA< 40.5 70 7 1 (0.9000000 0.1000000) *
  3) USIA>=40.5 232 99 2 (0.4267241 0.5732759)
    6) USIA<=52.5 15 0 1 (1.0000000 0.0000000) *
    7) USIA< 52.5 197 64 2 (0.3248731 0.6751269) *
      14) DINDING=1 12 0 1 (1.0000000 0.0000000) *
      15) DINDING=2,3 185 52 2 (0.2810811 0.7189189)
        30) PENDIDIKAN=1,4 8 0 1 (1.0000000 0.0000000) *
        31) PENDIDIKAN=2,3 177 44 2 (0.2485876 0.7514124)
          62) JENIS.LANTAI=3 69 27 2 (0.3913043 0.6086957)
            124) USIA< 46 17 0 1 (1.0000000 0.0000000) *
            125) USIA>=46 52 10 2 (0.1923077 0.8076923) *
              63) JENIS.LANTAI=1,2 108 17 2 (0.1574074 0.8425926)
                126) PEKERJAAN=1,2 84 17 2 (0.2023810 0.7976190)
                  252) DINDING=2 35 12 2 (0.3428571 0.6571429)
                    504) USIA>=42.5 13 4 1 (0.6923077 0.3076923) *
                    505) USIA< 42.5 22 3 2 (0.1363636 0.8636364) *
                      253) DINDING=3 49 5 2 (0.1020408 0.8979592) *
                        127) PEKERJAAN=3,4 24 0 2 (0.0000000 1.0000000) *

> plot(cartTree)
> summary(cartTree)

      Accuracy : 0.8682
      95% CI : (0.7974, 0.9213)
      No Information Rate : 0.5426
      P-value [Acc > NIR] : 2.653e-15

      kappa : 0.737

McNemar's Test P-value : 0.1456

      Sensitivity : 0.9153
      Specificity : 0.8286
      Pos Pred Value : 0.8182
      Neg Pred Value : 0.9206
      Prevalence : 0.4574
      Detection Rate : 0.4186
      Detection Prevalence : 0.5116
      Balanced Accuracy : 0.8719

'Positive' Class : 1

> #Membuat kurva ROC algoritma CART
> #Konversi variabel target ke format biner
> binary_truth2 <- ifelse(data.test$class == "2",1,0)
> binary_prediction2 <- ifelse(cartPred == "2",1,0)
> length(binary_prediction2)
[1] 129
> length(binary_truth2)
[1] 129
> #Hitung ROC
> roc_curve <- roc(binary_truth2, binary_prediction2)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> roc_curve

> #Visualisasi dengan rpart.plot
> rpart.plot(cartTree, type=4, digits=5, fallen.leaves=FALSE, shadow.col="gray", box.pale
tte="Bugn")
> #akurasi model
> cartPred = predict(cartTree, data.test, type = 'class')
> table(cartPred, data.test$class) #confusion matrix dari data testing

cartPred 1 2
          1 54 12
          2 5 58

> confusionMatrix(as.factor(cartPred), as.factor(data.test$class), positive='1')
Confusion Matrix and Statistics

      Reference
Prediction 1 2
          1 54 12
          2 5 58

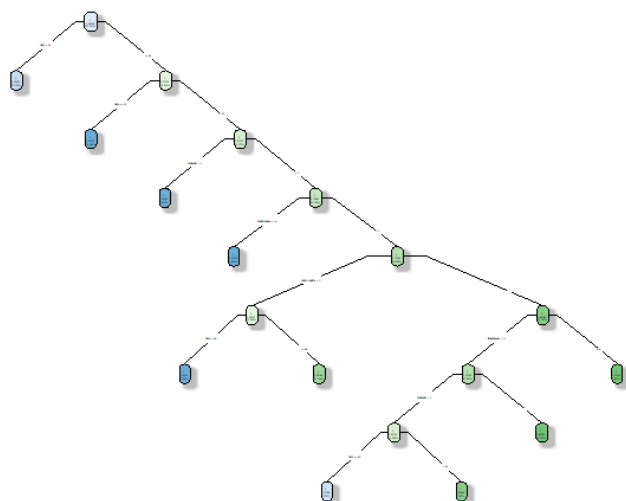
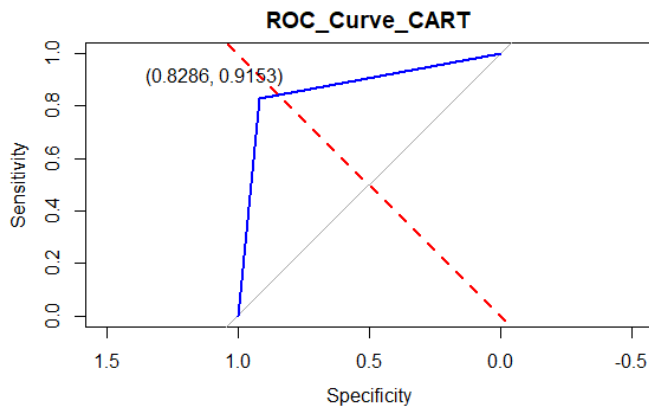
```

```

Call:
roc.default(response = binary_truth2, predictor = binary_prediction2)

Data: binary_prediction2 in 59 controls (binary_truth2 0) < 70 cases (binary_truth2 1).
Area under the curve: 0.8719
> #Hitung AUC
> auc_value <- auc(roc_curve)
> auc_value
Area under the curve: 0.8719
> #Gambar kurva ROC algoritma CART
> plot(roc_curve, main = "ROC_Curve_CART", col = "blue", lwd = 2)
> abline(a=0, b=1, col = "red", lwd = 2, lty = 2)
> text(0.8286, 0.9153, "(0.8286, 0.9153)", col = "black", cex=1,1)

```



0.11. Proporsi Data 80:20

```
> #Classification using CART
> library(rpart) #model CART
> library(rpart.plot)
> #visualisasi decision tree
> library(MASS) #untuk data birthwt untuk variabel y kualitatif
> library(ggplot2) #untuk data msleep untuk variabel y kuantitatif
> #Create decision tree cart tree-based
> cartTree <- rpart(class~., data=data.train, method = 'class') #method yang dipilih 'class' karena bertipe faktor
> printcp(cartTree)

Classification tree:
rpart(formula = class ~ ., data = data.train, method = "class")

Variables actually used in tree construction:
[1] DINDING      JENIS.LANTAI PEKERJAAN  USIA

Root node error: 163/345 = 0.47246

n= 345

      CP nsplit rel error  xerror   xstd
1 0.368098      0  1.00000 1.00000 0.056890
2 0.141104      1  0.63190 0.71779 0.053946
3 0.085890      2  0.49080 0.58282 0.050902
4 0.049080      4  0.31902 0.36810 0.043192
5 0.010225      5  0.26994 0.28834 0.039089
6 0.010000      8  0.23926 0.30675 0.040114
> #Menampilkan hasil tree-based model
> cartTree
> #Menampilkan hasil tree-based model
> cartTree
n= 345

node), split, n, loss, yval, (yprob)
* denotes terminal node

1) root 345 163 1 (0.52753623 0.47246377)
 2) JENIS.LANTAI=3 185 53 1 (0.71351351 0.28648649)
 4) USIA< 46.5 88 7 1 (0.92045455 0.07954545) *
 5) USIA>=46.5 97 46 1 (0.52577320 0.47422680)
 10) USIA>=52.5 33 0 1 (1.00000000 0.00000000) *
 11) USIA< 52.5 64 18 2 (0.28125000 0.71875000)
 22) DINDING=1,3 8 0 1 (1.00000000 0.00000000) *
 23) DINDING=2 56 10 2 (0.17857143 0.82142857) *
 3) JENIS.LANTAI=1,2 160 50 2 (0.31250000 0.68750000)
 6) USIA< 40.5 29 3 1 (0.89655172 0.10344828) *
 7) USIA>=40.5 131 24 2 (0.18320611 0.81679389)
 14) DINDING=1,2 72 19 2 (0.26388889 0.73611111)
 28) USIA>=42.5 37 16 2 (0.43243243 0.56756757)
 56) PEKERJAAN=1,2 17 6 1 (0.64705882 0.35294118) *
 57) PEKERJAAN=3,4 20 5 2 (0.25000000 0.75000000) *
 29) USIA< 42.5 35 3 2 (0.08571429 0.91428571) *
 15) DINDING=3 59 5 2 (0.08474576 0.91525424) *
> plot(cartTree)
> summary(cartTree)
```

```

> #Visualisasi dengan rpart.plot
> rpart.plot(cartTree, type=4, digits=5, fallen.leaves=FALSE, shadow.col="gray", box.pale
tte="BuGn")
> #akurasi model
> cartPred = predict(cartTree, data.test, type = 'class')
> table(cartPred, data.test$class) #confusion matrix dari data testing

cartPred 1 2
1 35 7
2 4 40
> confusionMatrix(as.factor(cartPred), as.factor(data.test$class), positive='1')
Confusion Matrix and Statistics

          Reference
Prediction 1 2
1 35 7
2 4 40

          Accuracy : 0.8721
          95% CI : (0.7827, 0.9344)
       No Information Rate : 0.5465
       P-Value [Acc > NIR] : 1.028e-10

          Kappa : 0.7436
  Mcnemar's Test P-value : 0.5465

       Sensitivity : 0.8974
       Specificity : 0.8511
       Pos Pred Value : 0.8333
       Neg Pred Value : 0.9091
       Prevalence : 0.4535
       Detection Rate : 0.4070
       Detection Prevalence : 0.4884
       Balanced Accuracy : 0.8742

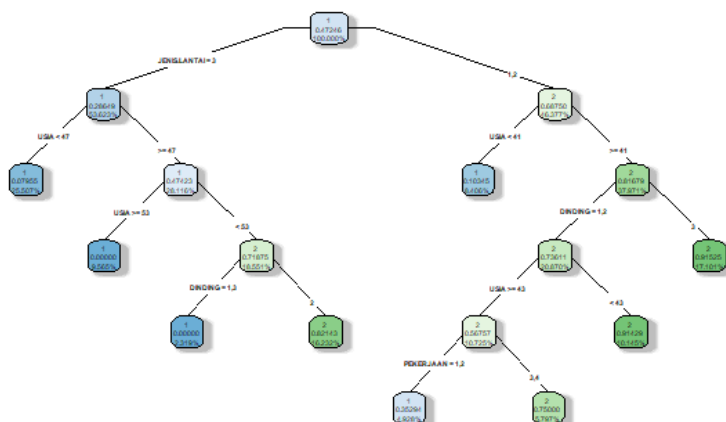
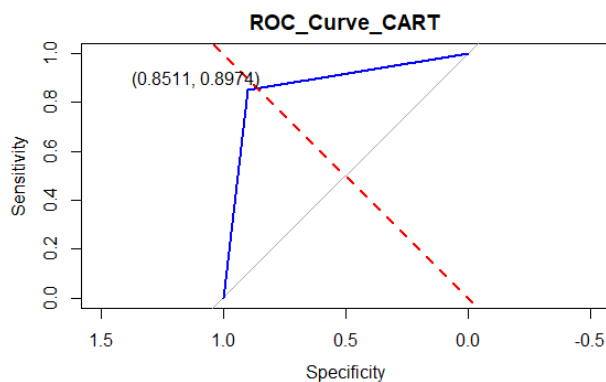
       'Positive' Class : 1

> #Membuat kurva ROC algoritma CART
> #Konversi variabel target ke format biner
> binary_truth2 <- ifelse(data.test$class == "2",1,0)
> binary_prediction2 <- ifelse(cartPred == "2",1,0)
> length(binary_prediction2)
[1] 86
> length(binary_truth2)
[1] 86
> #Hitung ROC
> roc_curve <- roc(binary_truth2, binary_prediction2)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> roc_curve

call:
roc.default(response = binary_truth2, predictor = binary_prediction2)

Data: binary_prediction2 in 39 controls (binary_truth2 0) < 47 cases (binary_truth2 1).
Area under the curve: 0.8742
> #Hitung AUC
> auc_value <- auc(roc_curve)
> auc_value
Area under the curve: 0.8742
> #Gambar kurva ROC algoritma CART
> plot(roc_curve, main = "ROC_Curve_CART", col = "blue", lwd = 2)
> abline(a=0, b=1, col = "red", lwd = 2, lty = 2)
> text(0.8511, 0.8974, "(0.8511, 0.8974)", col = "black", cex=1.1)

```



0.12. Proporsi Data 90:10

```

> #Clasification using CART
> library(rpart) #model CART
> library(rpart.plot)
> #visualisasi decision tree
> library(mass) #untuk data birthwt untuk variabel y kualitatif
> library(ggplot2) #untuk data msleep untuk variabel y kuantitatif
> #Create decision tree cart tree-based
> cartTree <- rpart(class~., data=data.train, method = 'class') #method yang dipilih 'class'
karena bertipe faktor
> printcp(cartTree)

Classification tree:
rpart(formula = class ~ ., data = data.train, method = "class")

Variables actually used in tree construction:
[1] DINDING JENIS.LANTAI USIA

Root node error: 189/388 = 0.48711

n= 388

      cp nsplit rel error xerror xstd
1 0.370370 0 1.00000 1.00000 0.052093
2 0.153439 1 0.62963 0.75661 0.050277
3 0.084656 2 0.47619 0.53439 0.045732
4 0.047619 4 0.30688 0.30139 0.036896
5 0.010000 5 0.25926 0.25397 0.034315
> #Menampilkan hasil tree-based model
> cartTree

n= 388

node), split, n, loss, yval, (yprob)
* denotes terminal node

1) root 388 189 1 (0.51288660 0.48711340)
 2) JENIS.LANTAI=3 204 62 1 (0.69607843 0.30392157)
   4) USIA< 46.5 95 10 1 (0.89473684 0.10526316) *
   5) USIA>=46.5 109 52 1 (0.52293578 0.47706422)
      10) USIA>=52.5 37 0 1 (1.00000000 0.00000000) *
      11) USIA< 52.5 72 20 2 (0.27777778 0.72222222)
          22) DINDING=1,3 9 0 1 (1.00000000 0.00000000) *
          23) DINDING=2 63 11 2 (0.17460317 0.82539683) *
 3) JENIS.LANTAI=1,2 184 57 2 (0.30978261 0.69021739)
   6) USIA< 40.5 35 3 1 (0.91428571 0.08571429) *
   7) USIA>=40.5 149 25 2 (0.16778523 0.83221477) *

> plot(cartTree)
> summary(cartTree)
> #Visualisasi dengan rpart.plot
> rpart.plot(cartTree, type="t", digits=5, fallen.leaves=FALSE, shadow.col="gray", box.palette
="BuGn")
> #akurasi model
> cartPred = predict(cartTree, data.test, type = 'class')
> table(cartPred, data.test$class) #confusion matrix dari data testing

cartPred 1 2
         1 19 0
         2 3 21

> confusionMatrix(as.factor(cartPred), as.factor(data.test$class), positive='1')
Confusion Matrix and Statistics

      Reference
Prediction 1 2
         1 19 0
         2 3 21

```

```

        Accuracy : 0.9302
        95% CI : (0.8094, 0.9854)
    No Information Rate : 0.5116
    P-Value [Acc > NIR] : 3.543e-09

        Kappa : 0.8608

McNemar's Test P-Value : 0.2482

        Sensitivity : 0.8636
        Specificity : 1.0000
        Pos Pred Value : 1.0000
        Neg Pred Value : 0.8750
        Prevalence : 0.5116
        Detection Rate : 0.4419
        Detection Prevalence : 0.4419
        Balanced Accuracy : 0.9318

'Positive' Class : 1

> #Membuat kurva ROC algoritma CART
> #konversi variabel target ke format biner
> binary_truth2 <- ifelse(data.test$class == "2",1,0)
> binary_prediction2 <- ifelse(cartPred == "2",1,0)
> length(binary_prediction2)
[1] 43
> length(binary_truth2)
[1] 43
> #Hitung ROC
> roc_curve <- roc(binary_truth2, binary_prediction2)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> roc_curve

Call:
roc.default(response = binary_truth2, predictor = binary_prediction2)

Data: binary_prediction2 in 22 controls (binary_truth2 0) < 21 cases (binary_truth2 1).
Area under the curve: 0.9318
> #Hitung AUC
> auc_value <- auc(roc_curve)
> auc_value
Area under the curve: 0.9318
> #Gambar kurva ROC algoritma CART
> plot(roc_curve, main = "ROC_Curve_CART", col = "blue", lwd = 2)
> abline(a=0, b=1, col = "red", lwd = 2, lty = 2)
> text(1.000, 0.864, "(0.8511, 0.8974)", col = "black", cex=1.1)
> #Gambar kurva ROC algoritma CART
> plot(roc_curve, main = "ROC_Curve_CART", col = "blue", lwd = 2)
> abline(a=0, b=1, col = "red", lwd = 2, lty = 2)
> text(1.000, 0.864, "(1.000, 0.864)", col = "black", cex=1.1)

```

Lampiran 7. Foto Dokumentasi Penelitian



Gambar 0.1. Wawancara dengan Pendamping PKH



Gambar 0.2. Pengisian Kuesioner dengan Responden



Gambar 0.3. Mengikuti Pendampingan dengan Penerima PKH



Gambar 0.4. Mengikuti Pendampingan dengan Penerima PKH

Lampiran 8. Daftar Riwayat Hidup**DAFTAR RIWAYAT HIDUP****A. Identitas Diri**

1. Nama : Rizka Zuliyanti
2. Tempat, Tanggal Lahir : Grobogan, 12 Juli 2002
3. Alamat : Ngemplak, RT.05/RW.07
Ds. Sembungharjo, Kec. Pulokulon,
Kab. Grobogan, Prov. Jawa Tengah
4. Email : rizkazuly127@gmail.com

B. Riwayat Pendidikan

1. TK Dharma Wanita Desa Sembungharjo
2. SD N 4 Sembungharjo
3. MTS Puteri Sunniyyah Selo
4. MA Sunniyyah Selo
5. Fakultas Sains dan Teknologi UIN Walisongo Semarang

Semarang, 28 Mei 2024

Pembuat Pernyataan,



Rizka Zuliyanti

NIM : 2008046041