

**DEPARTEMEN ILMU KOMPUTER**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**INSTITUT PERTANIAN BOGOR**

**BOGOR**

**2018**

**GALIH SUKRON INSANI**

**KLASIFIKASI DATA GEMPA BUMI MENGGUNAKAN**

**ALGORITME POHON KEPUTUSAN C5.0**

**PERNYATAAN MENGENAI SKRIPSI DAN  
SUMBER INFORMASI SERTA PELIMPAHAN HAK CIPTA**

Dengan ini saya menyatakan bahwa skripsi berjudul Klasifikasi Data Gempa Bumi menggunakan Algoritme Pohon Keputusan C5.0 adalah benar karya saya dengan arahan dari komisi pembimbing dan belum diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka di bagian akhir skripsi ini.

Dengan ini saya melimpahkan hak cipta dari karya tulis saya kepada Institut Pertanian Bogor.

Bogor, Oktober 2018

*Galih Sukron Insani*

NIM G64154024

**ABSTRAK**

ANGELICA INDRAWAN. Klasifikasi Data Gempa Bumi menggunakan Algoritme Pohon Keputusan C5.0. Dibimbing oleh MUHAMMAD ASYHAR AGMALARO.

Penelitian ini merupakan pengembangan dan implementasi dari system electrical impedance tomography yang murah dana man digunakan. Mendesain sebuah system yang mampu melihat penampang dalam sebuah objek buah tropika secara 2 dimensi untuk mengetahui kualitas buah tersebut. Dengan menempatkan 16 elektroda melingkar diseluruh sisi sebuah tangka phantom yang saling terhubung. Arus 1mA dan 60 kHz gelombang sinusoida diinduksikan kepada elektrodapertama dari kedua secara konstan kemudian dilakukanpengukuran dan pengambilan data potensial tegangan untuk masing-masing elektroda yang saling bertetanggaan dengan menggunakan pin analog yang dimiliki oleh Arduino. Data ini akan direkontruksi oleh aplikasi open source yang dibangun menggunakan Bahasa pemrograman python.

Kata kunci: algoritme C5.0, data mining, gempa bumi, klasifikasi, pohon keputusan.

**ABSTRACT**

ANGELICA INDRAWAN. Earthquake Data Classification using C5.0 Decision Tree Algorithm. Supervised by MUHAMMAD ASYHAR AGMALARO.

An earthquake is a sudden movements of the Earth's plates caused by the sudden release of accumulated pressures along the Earth's geological fault. Indonesia becomes a country with geological conditions prone to earthquakes so that earthquake classification is needed to measure the intensity and level of earthquake risk to estimate the vulnerability of an area to earthquakes. This research was conducted to classify the earthquake through data mining using decision tree method. Data mining better known as the mining of information from the data. Classification is a form of data modeling analysis that describes important data classes. Decision tree is a tree structure similar to a flowchart consisting of internal nodes, branches, leaves, and roots. The C5.0 algorithm is one of the algorithms for creating decision trees.

Keywords: C5.0 algorithm, data mining, earthquake, classification, decision tree.

Skripsi  
sebagai salah satu syarat untuk memperoleh gelar  
Sarjana Komputer   
pada  
Departemen Ilmu Komputer

**GALIH SUKRON INSANI**

**DEPARTEMEN ILMU KOMPUTER**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**INSTITUT PERTANIAN BOGOR**

**BOGOR**

**2018**

**KLASIFIKASI DATA GEMPA BUMI MENGGUNAKAN**

**ALGORITME POHON KEPUTUSAN C5.0**

Penguji:

1. DrEng Wisnu Ananta Kusuma, ST MT
2. Husnul Khotimah, SKomp MKom

Judul Skripsi : Klasifikasi Data Gempa Bumi menggunakan Algoritme Pohon Keputusan C5.0

Nama : Galih Sukron Insani

NIM : G64154024

Disetujui oleh

|  |
| --- |
| Muhammad Asyhar Agmalaro, SSi, MKom  Pembimbing |

Diketahui oleh

Dr Ir Agus Buono, MSi, MKom

Ketua Departemen

Tanggal Lulus:

**PRAKATA**

Puji dan syukur penulis panjatkan kepada Allah SWT atas segala karunia-Nya sehingga karya ilmiah ini berhasil diselesaikan. Penelitian yang dilakukan sejak Juli 2017 memiliki tema klasifikasi gempa bumi dengan judul Klasifikasi Data Gempa Bumi menggunakan Algoritme Pohon Keputusan C5.0.

Pada kesempatan ini, penulis ingin menyampaikan ucapan terima kasih kepada:

1. Keluarga penulis yaitu kedua orang tua penulis bapak Suraji, ibu Siti Aisiyah dan juga Umi Pitri yang telah memberikan dukungan baik doa, moral maupun material kepada penulis.
2. Bapak Muhammad Asyhar Agmalaro, SSi, Mkom selaku dosen pembimbing, Bapak Drs M Nur Indro, Msc dan Bapak Drs Agah D Garnadi, Grad.Dip.Sci yang telah memberikan arahan, bimbingan, saran, serta bantuan lainnya dalam penyelesaian skripsi dan penelitian.
3. Bapak DrEng Wisnu Ananta Kusuma, ST, MT dan Ibu Husnul Khotimah, SKomp, MKom sebagai dosen penguji atas waktu yang telah diluangkan untuk menguji penelitian dan skripsi yang telah dilakukan.
4. Seluruh dosen yang telah membagikan ilmunya kepada penulis serta staf departemen Ilmu Komputer yang telah membantu penulis selama empat semester.
5. Teman-teman Ilmu Komputer Alih Jeni 10. Terima kasih atas pengalaman dan pelajaran terbaiknya sampai dengan tingkat akhir.

Penulis berharap agar tulisan ini dapat bermanfaat bagi pembaca.

Bogor, Oktober 2018

*Galih Sukron Insani*

**DAFTAR ISI**

DAFTAR TABEL vi

DAFTAR GAMBAR vi

DAFTAR LAMPIRAN vi

[PENDAHULUAN 1](#_Toc525039975)

[Latar Belakang 1](#_Toc525039976)

[Perumusan Masalah 1](#_Toc525039977)

[Tujuan Penelitian 1](#_Toc525039978)

[Manfaat Penelitian 2](#_Toc525039979)

[Ruang Lingkup Penelitian 2](#_Toc525039980)

[TINJAUAN PUSTAKA 2](#_Toc525039981)

[*Data Mining* 2](#_Toc525039982)

[Klasifikasi 2](#_Toc525039983)

[Algoritme Pohon Keputusan 3](#_Toc525039984)

[*Gain Ratio* 3](#_Toc525039985)

[Algoritme C5.0 4](#_Toc525039986)

[Penjelasan Istilah Data Gempa 4](#_Toc525039987)

[Penjelasan Istilah Seismik 5](#_Toc525039988)

[*K-fold Cross Validation* 5](#_Toc525039989)

[*Oversampling* 5](#_Toc525039990)

[METODE 6](#_Toc525039991)

[Data Penelitian 6](#_Toc525039992)

[Tahapan Penelitian 7](#_Toc525039993)

[Lingkungan Pengembangan 8](#_Toc525039994)

[HASIL DAN PEMBAHASAN 9](#_Toc525039995)

[Pengumpulan dan Praproses Data 9](#_Toc525039996)

[Penentuan Data Uji dan Data Latih 10](#_Toc525039997)

[Pemodelan dan Pengujian Pohon Keputusan 11](#_Toc525039998)

[Visualisasi dan Analisis Hasil Pemodelan 16](#_Toc525039999)

[SIMPULAN DAN SARAN 18](#_Toc525040000)

[Simpulan 18](#_Toc525040001)

[Saran 19](#_Toc525040002)

[DAFTAR PUSTAKA 19](#_Toc525040003)

[LAMPIRAN 21](#_Toc525040004)

[RIWAYAT HIDUP 57](#_Toc525040005)

**DAFTAR TABEL**

1. [Format data gempa 6](#_Toc525040135)
2. [Penentuan kategori atribut victim, damage, placeType, areaClass, dan terrainLevel. 10](#_Toc525040136)
3. [Perbandingan akurasi algoritme C5.0 dengan SVM. 19](#_Toc525040137)

**DAFTAR GAMBAR**

1. [Bentuk data gempa 6](#_Toc525038592)
2. [Tahapan penelitian 7](file:///D:\S.K.R.I.P.S.I\Klasifikasi%20Gempa\Tulisan\%5bRevisi-Seminar-1-3%5dG64154040_Angelica%20Indrawan.docx#_Toc525038593)
3. [Pseudocode penentuan Class 9](#_Toc525038594)
4. [Langkah oversampling dan k-fold cross validation 11](#_Toc525038595)
5. [Potongan kode pembacaan data 11](#_Toc525038596)
6. [Potongan kode konversi data 11](#_Toc525038597)
7. [Potongan kode pemodelan C5.0 11](#_Toc525038598)
8. [Ringkasan hasil pemodelan C5.0 12](#_Toc525038599)
9. [Pohon keputusan hasil pemodelan 13](#_Toc525038600)
10. [Potongan kode klasifikasi C5.0 13](#_Toc525038601)
11. [Hasil klasifikasi C5.0 pada data uji 13](#_Toc525038602)
12. [Potongan kode evaluasi hasil pemodelan dan klasifikasi 13](#_Toc525038603)
13. [Hasil evaluasi pemodelan dan klasifikasi 14](#_Toc525038604)
14. [Potongan kode dan hasil SVM 16](#_Toc525038605)
15. [Kode visualisasi hasil klasifikasi data uji 17](#_Toc525038606)
16. [Hasil visualisasi data uji. 18](#_Toc525038607)

**DAFTAR LAMPIRAN**

1. [Kode program pemodelan, klasifikasi, dan pengujian C5.0 21](#_Toc514627723)
2. [Kode program visualisasi dengan framework Shiny dan library Leaflet 23](#_Toc514627724)
3. [Pembagian data uji ditiap fold 24](#_Toc514627725)
4. [Pembagian data latih ditiap fold 25](#_Toc514627726)
5. [Hasil pemodelan ditiap fold 26](#_Toc514627727)
6. [Hasil prediksi kelas ditiap fold 38](#_Toc514627728)
7. [Hasil akurasi ditiap fold 39](#_Toc514627729)
8. [Confusion matrix hasil pengujian ditiap fold 40](#_Toc514627730)
9. [Pohon keputusan hasil pemodelan pada fold ke-1 51](#_Toc514627731)
10. [Pohon keputusan hasil pemodelan pada fold ke-2 52](#_Toc514627732)
11. [Pohon keputusan hasil pemodelan pada fold ke-3 52](#_Toc514627733)
12. [Pohon keputusan hasil pemodelan pada fold ke-4 53](#_Toc514627734)
13. [Pohon keputusan hasil pemodelan pada fold ke-5 53](#_Toc514627735)
14. [Pohon keputusan hasil pemodelan pada fold ke-6 54](#_Toc514627736)
15. [Pohon keputusan hasil pemodelan pada fold ke-7 54](#_Toc514627737)
16. [Pohon keputusan hasil pemodelan pada fold ke-8 55](#_Toc514627738)
17. [Pohon keputusan hasil pemodelan pada fold ke-9 55](#_Toc514627739)
18. [Pohon keputusan hasil pemodelan pada fold ke-10 56](#_Toc514627740)

# 

# PENDAHULUAN

## Latar Belakang

Salah satu permasalahan pasca panen buah tropika adalah adanya sejumlah hama serangga yang menghabiskan siklus hidup dari menetas hingga dewasa didalam buah, ketika sampai tahapan dewasa baru keluar dari buah saat buah sudah matang, Masalah ini akan menyebabkan kualitas buah rusak menjadi bahan buangan yang tak bernilai. Misalnya, pada buah manga, salah satu hama endemic antara lain adalah kutu Sternochetus (Fabricius) misalnya, yang tingkat kerusakannya baru diketahui saat buah matang. Begitu pula pada kasus buah manggis, misalnya penyakit getah kuning baru diketahui tingkat kerusakan pada buah manggis saat buah manggis dibelah.

Dengan demikian, perlu upaya secara dini untuk mengetahui kualitas buah tanpa merusak. Cara seperti demikian akan menjamin kualitas buah dari segi produsen/petani, serta menguntungkan konsumen atas keterjaminan kualitas buah.

Sehingga peristiwa yang disebutkan oleh Bapak Siswono Yudhohusodo mengenai penolakan 1 kontainer buah jika ditemukan 1 buah yang kualitasnya buruk atau busuk dapat dihindari. Karena menurutnya untuk negara-negara Eropa memasang standar pengemasan yang tinggi (Ramdani 2012).

EIT adalah alat yang memungkinkan rekonstruksi gambar dari objek apapun tanpa merusak bentuk dari objek tersebut. Ia merekonstruksi sebuah gambar berdasarkan pengukuran voltase listrik yang diperoleh dari elektroda yang terpasang secara merata di sekitar lingkar konduktif. Arus diinduksi dengan metode ketetanggaan dan pengukuran voltase dilakukan secara berdekatan seperti yang ditunkukkan pada gambar 1.

(Chitturi et al. 2014) sudah mengembangkan system EIT yang murah dan terjangkau untuk memodelkan dan mendiagnosis penyakit paru-paru. Namun mereka masih menggunakan aplikasi berbayar untuk melakukan prosessing dan rekonstruksi data yang didapat hingga menjadi sebuah citra yang dapat dianalisis.

## Perumusan Masalah

Rumusan masalah dari penelitian ini adalah:

1. Bagaimana hasil klasifikasi skala gempa bumi berdasarkan algoritme Pohon Keputusan C5.0?
2. Bagaimana hasil klasifikasi ukuran gempa bumi terhadap data dalam dua kelas risiko?
3. Bagaimana hasil pemetaan menggunakan OpenStreetMaps API terhadap titik-titik terjadinya gempa?
4. Bagaimana perbandingan akurasi klasifikasi data gempa antara algoritme C5.0 dengan algoritme klasifikasi lain (SVM)?

## Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Melakukan klasifikasi skala gempa bumi berdasarkan hasil klasifikasi dari data gempa bumi menggunakan Algoritme Pohon Keputusan C5.0.
2. Menentukan ukuran gempa bumi dalam dua kelas risiko (gempa yang memiliki risiko terjadinya kerusakan/korban, gempa nonrisiko).
3. Melakukan pemetaan menggunakan OpenStreetMaps API pada titik-titik terjadinya gempa.
4. Melakukan perbandingan akurasi klasifikasi data gempa antara algoritme C5.0 dengan algoritme klasifikasi lain (SVM).

## Manfaat Penelitian

Hasil penelitian diharapkan dapat membantu mengukur tingkat kerentanan suatu daerah terhadap gempa bumi sehingga instansi terkait dapat melakukan persiapan sesuai intensitas risiko yang ada. Selain itu, hasil klasifikasi dari data gempa bumi juga diharapkan mampu membantu instansi terkait dalam hal pendistribusian bantuan dan mitigasi bencana.

## Ruang Lingkup Penelitian

Penelitian ini dilakukan terhadap data gempa bumi dari repositori data gempa BMKG bukan *real-time*. Klasifikasi dilakukan untuk mengelompokkan data berdasarkan kedalaman, skala kegempaan (magnitudo), ada/tidaknya korban atau kerusakan, letak dan karakteristik titik pusat gempa, dan kelas risiko. *Pruning* dilakukan secara otomatis oleh *library* C50 di R Pemetaan hanya dilakukan pada titik-titik pusat gempa. Visualisasi dilakukan hanya untuk pemetaan data uji saja, tidak menggunakan fitur manipulasi visualisasi lain.

# TINJAUAN PUSTAKA

## *Data Mining*

*Data mining* lebih dikenal sebagai penambangan informasi dari data, akan tetapi penambangan pengetahuan mungkin tidak terlalu mencerminkan penambangan dari data yang jumlahnya besar. Banyak istilah lain yang memiliki makna yang sama dengan data mining seperti: penambangan pengetahuan dari data, ekstraksi pengetahuan, analisis pola/data, arkeologi data, dan pengerukan data. *Data mining* sering disamakan dengan istilah populer lainnya, yaitu *Knowledge Discovery from Data* (KDD). Selain itu, *data mining* juga sering dipandang hanya sebagai langkah penting dalam proses penemuan pengetahuan (Han *et al.* 2012).

## Klasifikasi

Klasifikasi adalah sebuah bentuk analisis data pembuatan model yang mendeskripsikan kelas-kelas data yang penting. Model-model tersebut memprediksi label-label kelas kategorik (diskret dan tak terurut), disebut juga sebagai *classifier*. Analisis tersebut membantu kita untuk memahami data berjumlah besar. Beberapa penerapan klasifikasi yang dapat dilakukan adalah deteksi *fraud*, target pemasaran, prediksi performa, manufaktur, dan diagnosis medis. Salah satu metode klasifikasi adalah pohon keputusan. Pohon keputusan adalah struktur pohon mirip diagram alur yang terdiri dari node internal, cabang, daun, dan akar. Node internal melambangkan pengujian atribut dengan setiap cabangnya merepresentasikan hasil pengujiannya. Daun menandakan label kelas sedangkan akar adalah node teratas pada pohon keputusan (Han *et al*. 2012).

## Algoritme Pohon Keputusan

Algoritme untuk membuat pohon keputusan dari *tuple* latih partisi data, *D*.

Input: Partisi data, *D*, yang merupakan himpunan *tuple* latih dan label kelasnya masing-masing; Daftar atribut, himpunan kandidat atribut; Metode seleksi atribut, prosedur penentuan kriteria terbaik untuk memisahkan *tuple* data kedalam kelasnya masing-masing. Kriteria ini terdiri dari atribut pemisah dan mungkin titip pemisah atau subhimpunannya.

Output: Pohon Keputusan.

1. Buat node *N*;
2. jika tuple-tuple di *D* berkelas sama, *C*, maka
3. kembalikan *N* sebagai node daun berlabel kelas *C*;
4. jika daftar atribut kosong maka
5. kembalikan *N* sebagai node daun yang dilabeli dengan kelas mayoritas di *D*; (voting mayoritas)
6. terapkan metode seleksi atribut(*D*, daftar atribut) untuk menemukan kriteria pemisah terbaik;
7. tandai node *N* sebagai kriteria pemisah;
8. jika atribut pemisah bernilai diskret dan terpisah menjadi banyak cabang yang dimungkinkan maka (tidak dibatasi menjadi pohon biner)
9. daftar atribut = daftar atribut – atribut pemisah; (hilangkan atribut pemisah)
10. untuk setiap hasil *j* dari kriteria pemisah (partisi tuple dan tumbuhkan subpohon untuk setiap partisi)
11. biarkan *Dj* menjadi himpunan tuple data di *D* yang memenuhi hasil *j*; (satu partisi)
12. jika *Dj* kosong maka
13. lampirkan daun yang dilabeli dengan kelas mayoritas di *D* untuk node *N*;
14. selain itu lampirkan node yang dikembalikan oleh Membuat pohon keputusan(*Dj*, daftar atribut) untuk node *N*; akhiri iterasi
15. kembalikan nilai *N*; (Han *et al.* 2012).

## *Gain Ratio*

Algoritme C4.5 menggunakan *gain ratio* yang merupakan perluasan dari *information gain*. *Gain ratio* berupaya untuk mengatasi bias dengan melakukan normalisasi terhadap *information gain* menggunakan nilai ”*split information*” yang didefinisikan secara analog terhadap *Info(D)* sebagai:

Informasi yang potensial dibuat dengan memisahkan himpunan data latih, *D*, ke dalam *v* partisi, bersesuaian dengan *v* hasil pengujian terhadap atribut *A*. Untuk setiap hasil, jumlah *tuple* yang memiliki hasil tersebut dipertimbangkan sesuai jumlah *tuple* di *D*. Hal ini membedakannya dari *information* *gain* yang mengukur informasi berdasarkan klasifikasi yang diperoleh berdasarkan partisi yang sama. *Gain ratio* didefinisikan sebagai:

Atribut dengan *gain ratio* maksimum dipilih sebagai atribut pemisah. Akan tetapi jika *split information* mendekati 0, rasionya menjadi tidak stabil. Untuk menghindarinya, diberikan sebuah batasan yaitu *information gain* dari pengujian yang terpilih harus lebih besar daripada hasil rataan seluruh pengujian yang diteliti (Han *et al.* 2012).

## Algoritme C5.0

Algoritme C5.0 adalah algoritme klasifikasi untuk *dataset* yang besar. Algoritme ini merupakan perluasan algoritme C4.5, lebih baik dalam hal efisiensi dan penggunaan memori. Model C5.0 memisahkan sampel berdasarkan *field* dengan *information gain* paling maksimum (Patil et al. 2012).

## Penjelasan Istilah Data Gempa

Penjelasan istilah-istilah yang ada pada data berdasarkan laman <http://inatews.bmkg.go.id/new/glossary.php> sebagai berikut:

1. Date : Tanggal terjadinya gempabumi.
2. Time : Waktu terjadinya gempa dalam GMT , WIB (GMT+7), WITA (GMT+8), WIT(GMT+9).
3. Latitude : Koordinat lintang geografi tempat terjadinya gempa, Lintang Selatan ((-) atau (S)) Lintang Utara (tanpa tanda atau (N)).
4. Longitude : Koordinat bujur geografi tempat terjadinya gempa.
5. Kedalaman (*Depth*) : Kedalaman gempa dalam kilometer.
6. Mag : Magnitudo, Kekuatan gempa dalam skala richter.
7. TypeMag : Tipe magnitudo atau jenis jenis magnitudo gempa.
   1. Mb : Magnitudo menggunakan gelombang badan perioda pendek.
   2. mB : Magnitudo menggunakan gelombang badan (*broad band*) perioda panjang.
   3. ML : Magnitudo menggunakan gelombang lokal dengan komponen horisontal.
   4. MLv : Magnitudo menggunakan gelombang lokal dengan komponen vertikal.
   5. Mw : Magnitudo menggunakan moment seismik.
   6. Mw(mB) : Magnitudo menggunakan integrasi ganda moment seismik dan gelombang badan *broadband*.
8. Smaj : Jarak stasiun terjauh dari episenter.
9. Smin : Jarak stasiun terdekat dari episenter.
10. Az : *Azimuthal gap*, adalah sudut *gap* antara stasiun-stasiun pencatat gempa terhadap episenter.
11. RMS : *Root Mean Square* atau kuadrat rata-rata, adalah nilai kesalahan perhitungan.
12. cPhase : *Count Phase*, adalah jumlah fase waktu tiba gelombang gempa yang digunakan.

## Penjelasan Istilah Seismik

Berikut penjelasan beberapa istilah seismik yang disadur dari jurnal yang ditulis Ikram dan Qamar (2017)

1. Latitude : Garis khayal yang memotong bumi secara horizontal. Prediksi latitude yang tepat dapat membantu menerka longitude yang prospektif.
2. Longitude : Garis khayal yang memotong bumi secara vertikal. Prediksi longitude yang tepat juga dapat membantu menerka latitude yang prospektif. Akan tetapi, prediksi longitude lebih sulit daripada latitude karena longitude mengerucut di kutub-kutub bumi namun melebar di ekuator.
3. Kedalaman (*Depth*) : Jarak di antara pusat gempa dengan titik pelepasan energi sebenarnya di dalam bumi.
4. Magnitudo : Skala logaritmik yang diasosiasikan dengan getaran bumi.

## *K-fold Cross Validation*

Sebuah mitos populer tentang perkiraan akurasi hasil dari *k-fold cross validation* adalah hubungan ketergantungan yang diakibatkan oleh himpunan data latih yang saling *overlap*. Penelitian ini menunjukkan bahwa himpunan data latih yang tingkat *overlap*-nya tinggi bias menghasilkan model dengan akurasi aktual yang mirip, akan tetapi ini tidak dapat menjadi bukti yang menyimpulkan bahwa prediksi dari dua instansiasi tersebut saling berkaitan. Sebuah metode berbasis estimasi dari ragam sampel kemudian diajukan untuk menguji independensi perkiraan akurasi yang dihasilkan dari *k-fold cross validation* (Wong dan Yang 2017).

Pendekatan *k-fold cross validation* terkadang digunakan untuk mengatasi masalah *overfitting*. Pada metode ini, *cross validation* dilakukan sebanyak *k* kali, setiap kali dilakukan partisi data yang digunakan pada data latih dan data uji berbeda, hasilnya kemudian dibuat rataannya. Pada salah satu versi pendekatan ini, *m* contoh yang tersedia dipisah menjadi *k* subhimpunan yang terpisah yang masing-masing berukuran *m/k*. Prosedur *cross validation* kemudian dilakukan *k* kali dengan subhimpunan yang berbeda untuk data uji dan latih di tiap iterasi. Setiap contoh digunakan sebagai data uji pada satu percobaan dan sebagai data latih pada percobaan *k-1* lain (Mitchell 1997).

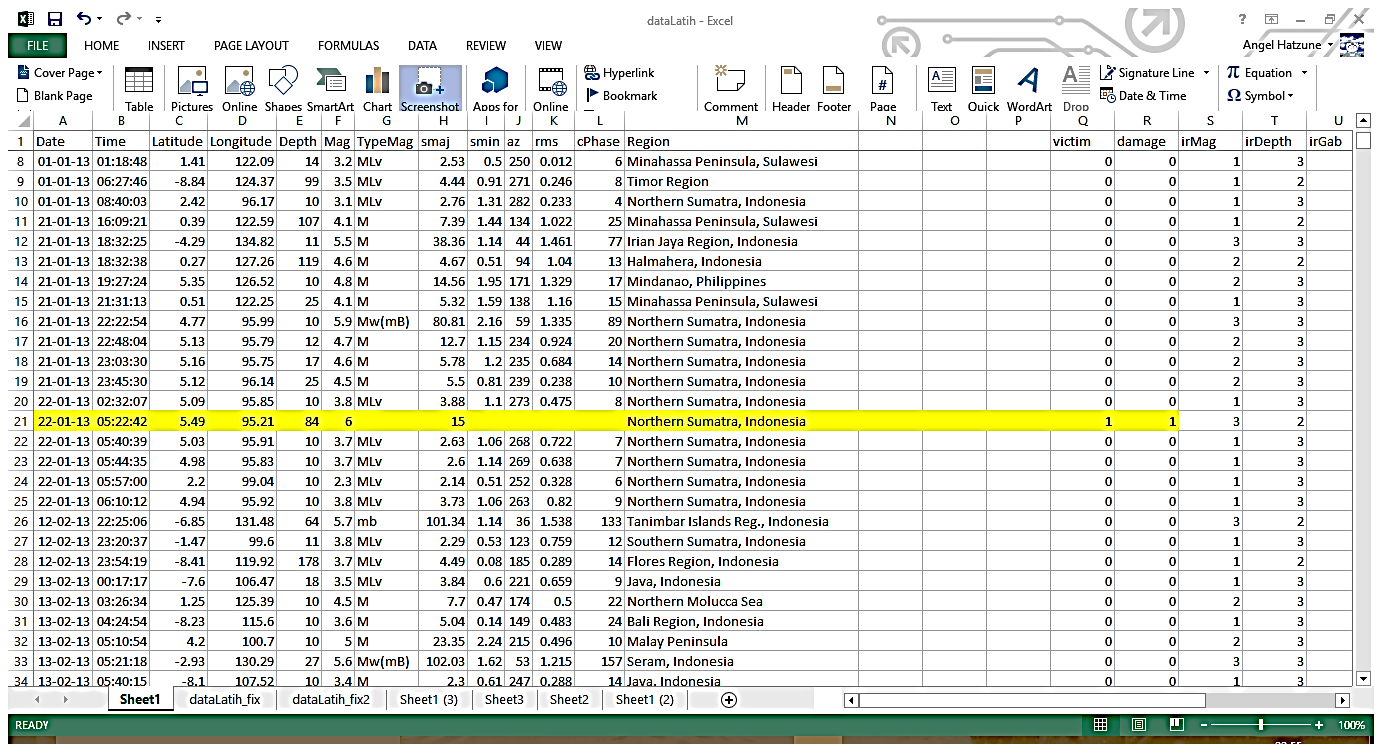
## *Oversampling*

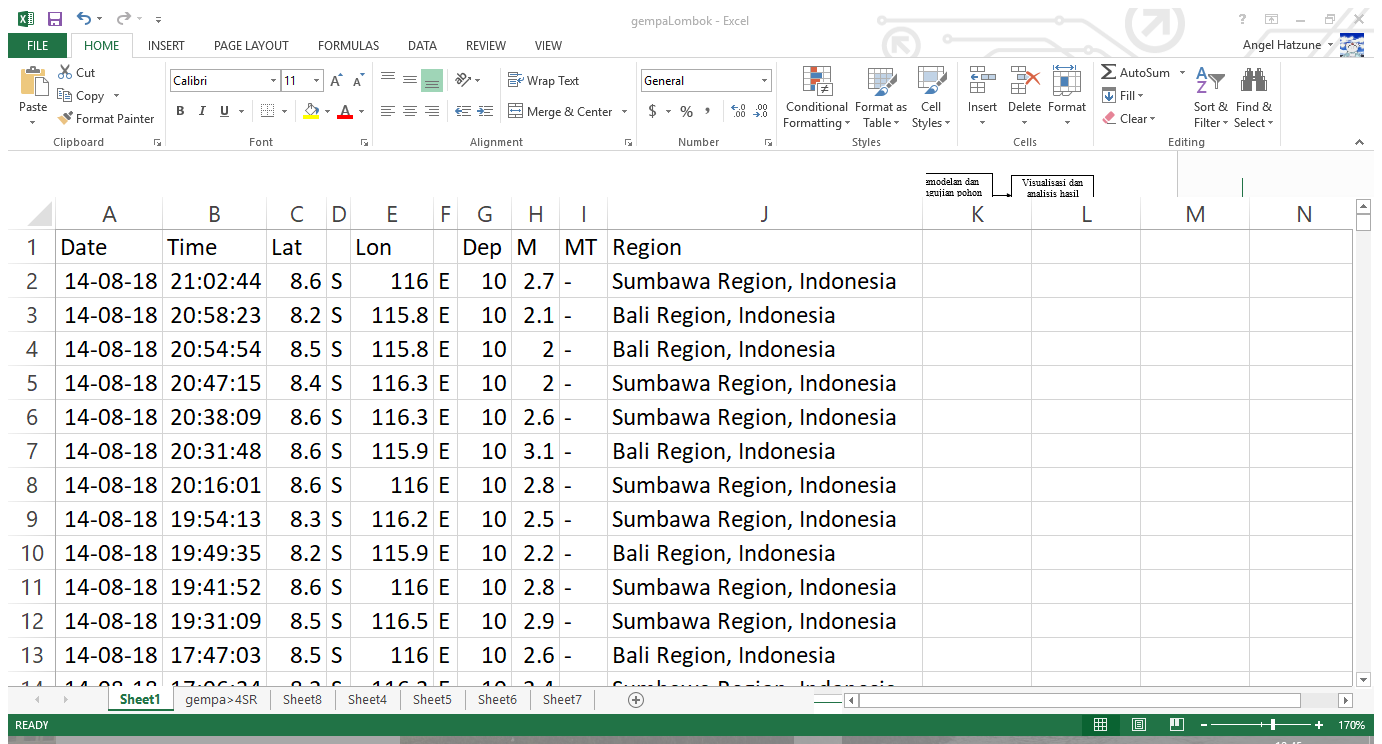
Algoritme klasifikasi tradisional bertujuan untuk meminimalisir jumlah kesalahan yang terjadi selama proses klasifikasi. Algoritme-algoritme tersebut mengasumsikan bahwa kesalahan dari *false positive* dan *negative* berjumlah sama sehingga algoritme tersebut tidak cocok untuk data dengan kelas yang tidak seimbang. *Oversampling* merupakan salah satu pendekatan yang digunakan untuk meningkatkan akurasi klasifikasi dari data yang tidak seimbang. *Oversampling* sendiri adalah proses yang menghasilkan jumlah baris data positif dan negatif yang sama dalam satu himpunan data latih dengan melakukan *resample* baris data positif. SMOTE adalah salah satu algoritme *oversampling* yang menambahkan baris data sintetis yang menyerupai baris data positif yang ada pada keseluruhan data (Han *et al.* 2012).

# METODE

## Data Penelitian

Data yang digunakan adalah data gempa bumi di Indonesia pada tahun 2012, 2013, dan 2018 yang dapat diakses gratis untuk tujuan penelitian dari situs <http://repogempa.bmkg.go.id> milik Badan Meteorologi, Klimatologi, dan Geofisika. Bentuk data dapat dilihat pada Gambar 1 dan keterangan mengenai format data dapat dilihat pada Tabel 1.





Gambar 1 Bentuk data gempa

Tabel 1 Format data gempa

| Nama Kolom | Tipe Data | Keterangan |
| --- | --- | --- |
| Date | *Character* | dd/mm/yyyy |
| Time | *Double* | hh:mm:ss.ms |
| Latitude/ Lat | *Double* | -90 s.d 90° |
| Longitude/ Lon | *Double* | -180 s.d 180° |
| Depth/ Dep | *Integer* | 1 = Dangkal (1 - 60 KM) |
|  |  | 2 = Sedang (61 - 300 KM) |
|  |  | 3 = Dalam (> 300 KM) |
| Mag/ M | *Double* | 1 = Kecil (<= 4,5) |
|  |  | 2 = Sedang (<= 5,5) |
|  |  | 3 = Besar (> 5,5) |
| TypeMag | *Character* | Mb, mB, ML, MLv, Mw, Mw(mB) |
| smaj | *Double* | Kilometer |
| smin | *Double* | Kilometer |
| Az | *Integer* | °sudut celah |
| rms | *Double* |  |
| cPhase | *Integer* |  |
| MT | *Character* | Potensi tsunami |
| region | *Character* | Wilayah gempa |

## Tahapan Penelitian

Penelitian ini dilakukan dalam beberapa tahap yaitu: pengumpulan dan praproses data, pembagian data uji dan data latih, pemodelan dan pengujian pohon keputusan, serta analisis dan evaluasi model pohon keputusan.

Mulai

Pengumpulan dan praproses data

Penentuan data uji dan data latih

Pemodelan dan pengujian pohon keputusan

Visualisasi dan analisis hasil pemodelan

Selesai

Gambar Tahapan penelitian

### Pengumpulan dan praproses data

Data yang digunakan pada penelitian ini diperoleh dari <http://repogempa.bmkg.go.id>. Data yang digunakan merupakan data gempa tahun 2012, 2013, dan 2018 sebanyak 167 data.

Praproses data dilakukan untuk menyesuaikan data yang akan diolah agar sesuai dengan penelitian yang akan dilakukan dengan menghilangkan atau menambahkan atribut data. Praproses juga dilakukan untuk meningkatkan akurasi hasil pengolahan dengan memperbaiki kualitas data. Praproses yang dilakukan adalah pemilihan atribut yang akan digunakan, penentuan kategori kelas, dan pemberian label kelas. Pemilihan atribut yang digunakan dilakukan dengan membuang atribut data yang tidak diperlukan yaitu: time, typeMag, az, rms, MT, dan cPhase. Penambahan atribut juga dilakukan dengan menambahkan kolom Class untuk menampung hasil label kelas, kolom victim untuk menandakan ada atau tidaknya korban, kolom damage untuk menandakan ada tidaknya kerusakan infrastruktur dan bangunan, kolom placeType untuk jenis dari tempat titik pusat gempa, areaClass untuk menampung kelas lokasi titik gempa, dan terrainLevel untuk menampung kelas ketinggian titik pusat gempa. Penentuan kategori kelas dilakukan dengan menggunakan prasyarat berikut:

* Gempa yang memiliki risiko terjadinya kerusakan/korban,
  + terdapat kerusakan bangunan, atau
  + terdapat korban jiwa (yang disebabkan gempa), atau
  + radius antara titik pusat gempa dengan stasiun pengukur gempa terjauh lebih dari 26.27 km (diambil dari nilai yang berada di atas kuartil dua data yang digunakan pada penelitian ini) dan berpotensi tsunami
    - kedalaman dibawah 60 km dan
    - nilai magnitudo diatas 5.5 sR
* Gempa nonrisiko

### Penentuan data uji dan data latih

Data latih dan uji yang digunakan merupakan data gempa yang diperoleh saat tahap pengumpulan dan praproses data. Data latih dan uji disatukan dalam satu fail csv. Jumlah data yang timpang antara kelas risiko dan nonrisiko diatasi dengan menggunakan metode *oversampling* menggunakan *library* Imbalance pada bahasa pemrograman R. Pembagian data dilakukan menggunakan metode *k-fold cross validation* menggunakan *library* Caret pada bahasa pemrograman R.

### Pemodelan dan pengujian pohon keputusan

Model klasifikasi yang digunakan pada penelitian ini adalah pohon keputusan. Algoritme yang digunakan untuk pohon keputusan adalah algoritme C5.0 yang memakai nilai *Information Gain* dengan *Gain Ratio* dalam pemodelannya. Pemodelan dan pengujian hasil pohon keputusan dilakukan menggunakan *library* c50 dengan bahasa pemrograman R. Pengujian model pohon keputusan dilakukan dengan menggunakan *Confusion Matrix*. Pengujian juga dilakukan menggunakan bahasa pemrograman R dengan *library* caret. Selain itu, perbandingan akurasi dilakukan terhadap hasil akurasi C5.0 dengan algoritme klasifikasi lain. Algoritme yang digunakan adalah *Support Vector Machine* (SVM) memakai bahasa pemrograman R dengan *library* e1071.

### Visualisasi dan analisis hasil pemodelan

Visualisasi hasil pemodelan dilakukan dengan memetakan titik-titik gempa ke dalam peta dijital. Pada penelitian ini, visualisasi dilakukan menggunakan OpenStreetMaps API melalui *library* Leaflet pada bahasa pemrograman R. Hasil visualisasi akan dianalisis untuk menentukan luas area yang terdampak gempa pada data gempa.

## Lingkungan Pengembangan

Spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam penelitian yaitu sebagai berikut :

1. Perangkat Keras :
   1. *Processor* Intel Core i3.
   2. RAM 4 GB.
   3. *Harddisk* 500 GB, 1 TB.
   4. Layar *monitor* 14 inci.
   5. *Graphic card* NVIDIA GeForce 920MX.
   6. *Mouse* dan *keyboard*.
2. Perangkat Lunak :
   1. Sistem operasi Windows 7 64-bit, Windows 10 64-bit.
   2. Microsoft Excel 2013 sebagai lembar pengolahan data tambahan dan praproses data.
   3. RStudio versi 1.1.383 untuk melakukan proses *data mining* dan pembuatan model pohon keputusan menggunakan *library* C50, Imbalance, Caret, e1071, readr.
   4. *Framework* Shiny untuk visualisasi data.
   5. OpenStreetMaps API dengan *library* Leaflet untuk melakukan pemetaan titik-titik terjadinya gempa bumi.
   6. Google Earth untuk penentuan atribut placeType, areaClass, dan terrainLevel.

# HASIL DAN PEMBAHASAN

## Pengumpulan dan Praproses Data

Data yang digunakan adalah data gempa bumi di Indonesia pada tahun 2012, 2013, 2018 yang dapat diakses gratis untuk tujuan penelitian dari situs <http://repogempa.bmkg.go.id> milik Badan Meteorologi, Klimatologi, dan Geofisika. Data gempa yang akan diolah tersebut merupakan data gempa khusus di wilayah Indonesia pada latitude (lintang) 6° utara hingga 11° selatan dan longitude (bujur) 142° timur hingga 94° timur dengan jangkauan magnitudo 1 hingga 9,5 skala Richter dan kedalaman 1 hingga 1000 KM.

Data yang dikumpulkan kemudian disimpan di dalam lembar sebar dengan format .xlsx. Setelah terkumpul, dilakukan penentuan kelas dengan menambahkan kolom Class di kolom pertama data. Penentuan data dari kategori Class dilakukan dengan menggunakan prasyarat berikut:

* Gempa yang memiliki risiko terjadinya kerusakan/korban,
  + terdapat kerusakan bangunan, atau
  + terdapat korban jiwa (yang disebabkan gempa), atau
  + radius antara titik pusat gempa dengan stasiun pengukur gempa terjauh lebih dari 8.82 km (diambil dari nilai kuartil tiga data yang digunakan pada penelitian ini) dan berpotensi tsunami
    - kedalaman dibawah 60 km dan
    - nilai magnitudo diatas 5.5 km
* Gempa nonrisiko

Penentuan kategori tersebut dilakukan dengan menggunakan fungsi logika di Microsoft Excel yang dapat dilihat pada Gambar 3. Penentuan kategori untuk kolom victim, damage, placeType, areaClass, dan terrainLevel dapat dilihat pada Tabel 2.

|  |
| --- |
| IF  (Damage == True OR Victim == True) OR (Smaj >= 8.82 AND Mag >5.5 AND Depth >60) THEN Class == Risk  ELSE  Class == Nonrisk  END IF |

Gambar 3 *Pseudocode* penentuan Class

Tabel Penentuan kategori atribut victim, damage, placeType, areaClass, dan terrainLevel.

| Kolom/ Atribut | Keterangan |
| --- | --- |
| victim | 0 = ada korban  1 = tidak ada korban |
| damage | 0 = tidak terjadi kerusakan  1 = terjadi kerusakan |
| placeType | 1 = air  2 = daratan |
| areaClass | 1 = laut  2 = hutan  3 = pemukiman penduduk/kota  4 = area pantai (laut)  5 = perkebunan  6 = persawahan |
| terrainLevel | 1 = laut  2 = pegunungan/perbukitan  3 = lembah/ dataran rendah |

## Penentuan Data Uji dan Data Latih

Data latih dan uji yang digunakan merupakan data hasil praproses. Data latih dan uji disatukan dalam satu fail csv. Jumlah data yang timpang antara kelas risiko dan nonrisiko diatasi dengan menggunakan metode *oversampling* menggunakan *library* imbalance pada bahasa pemrograman R. Pembagian data dilakukan menggunakan metode *k-fold cross validation* menggunakan *library* caret pada bahasa pemrograman R. Langkah *oversampling* dan *k-fold cross validation* dapat dilihat pada Gambar 4.

|  |
| --- |
| * library(imbalance) * imbalanceRatio(gempaDF) * gempaOver <- oversample(gempaDF, ratio = 0.5, method = "SMOTE", classAttr = "Class") * #dividing into train data and test data * gempaTrain <- gempaOver * library(caret)      * # actual cross validation * folds <- sample(rep(1:10, length.out = 195))     > for(k in 1:10) {  + # actual split of the data  + fold <- which(folds == k)  + data.train <- gempaTrain[-fold,]  + data.test <- gempaTrain[fold,]  + trainX <- data.train[,2:9]  + testX <- data.test[,2:9]  + trainY <-data.train[,1]  + testY <- data.test[,1]  + class.svm <- gempaTrain[-fold,1]  + data\_class.svm <- as.data.frame(class.svm)  + train.svm <- gempaTrain[-fold,2:9]  + ...  + } |

Gambar 4 Langkah *oversampling* dan *k-fold cross validation*

## Pemodelan dan Pengujian Pohon Keputusan

Pemodelan pohon keputusan dilakukan menggunakan algoritme C5.0. Algoritme C5.0 diimplementasikan menggunakan *package* c50 dengan bahasa pemrograman R pada perangkat lunak RStudio. C5.0 digunakan untuk melakukan klasifikasi berbasis aturan dengan model pohon keputusan. Kondisi dari tiap aturan yang dihasilkan oleh algoritme ini tidak sama. Aturan dari hasil algoritme maupun cabang pohon keputusan yang dihasilkan dapat dipangkas (*pruning*). *Library* c50 yang digunakan sudah otomatis melakukan pemangkasan aturan dan cabang pohon keputusan. Langkah-langkah klasifikasi dengan algoritme C5.0 yang dilakukan adalah:

1. Membaca data yang akan digunakan

|  |
| --- |
| * #Setting up workspace * setwd("D:/S.K.R.I.P.S.I/Klasifikasi Gempa/Data/110918/") * library(readr) * gempa1 <- read\_csv("dataLatih\_model.csv") * gempa <- gempa1 |

Gambar 5 Potongan kode pembacaan data

Data dengan format .csv yang akan digunakan dibaca menggunakan *library* readr di R dengan separator titik koma (;). Hasil pembacaan data dimasukkan ke dalam variabel-variabel yang akan digunakan selama pemrosesan.

1. Melakukan konversi ke *data frame* dan mengubah kolom kelas Class menjadi faktor

|  |
| --- |
| * #converting to data frame * gempaDF <- gempa[order(runif(nrow(gempa))),] * gempaDF$Class <- as.factor(gempaDF$Class) * gempaDF = as.data.frame(gempaDF) |

Gambar 6 Potongan kode konversi data

Sebelum melakukan pemodelan dan klasifikasi data dengan algoritme C5.0, data yang ada dikonversi dulu menjadi *data frame* agar dapat diolah. Kolom kelas Class juga dikonversi menjadi faktor untuk penentuan kelas hasil klasifikasi pada data uji.

1. Melakukan pemodelan data latih menggunakan algoritme C5.0.

|  |
| --- |
| * library(C50) * model <- C5.0(trainX, trainY) * plot\_all[[k]] <- plot(model) * p <- predict.C5.0(model, testX) |

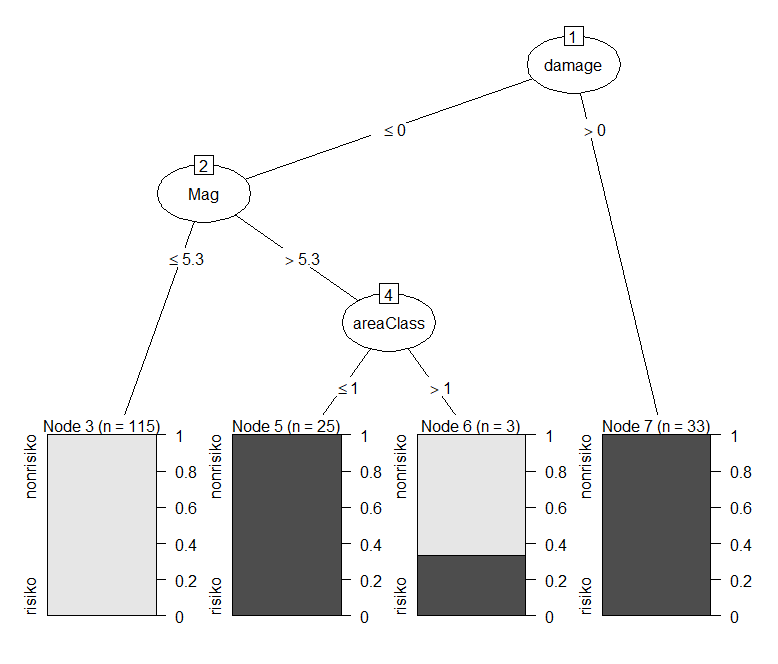
Gambar 7 Potongan kode pemodelan C5.0

Pemodelan dengan menggunakan algoritme C5.0 kemudian dilakukan terhadap data latih hasil konversi. Iterasi pemodelan C5.0 dilakukan sebanyak 10 kali berdasarkan jumlah *fold*. Ringkasan hasil pemodelan pada *fold* ke-7 dapat dilihat pada Gambar 8 dan pohon keputusan hasil pemodelan pada *fold* ke-7 dapat dilihat pada Gambar 9

|  |
| --- |
| Decision tree:  damage > 0: risiko (33)  damage <= 0:  :...Mag <= 5.3: nonrisiko (115)  Mag > 5.3:  :...areaClass <= 1: risiko (25)  areaClass > 1: nonrisiko (3/1)  Evaluation on training data (176 cases):  Decision Tree  ----------------  Size Errors  4 1( 0.6%) <<  (a) (b) <-classified as  ---- ----  117 (a): class nonrisiko  1 58 (b): class risiko  Attribute usage:  100.00% damage  81.25% Mag  15.91% areaClass  Time: 0.0 secs |

Gambar 8 Ringkasan hasil pemodelan C5.0

Hasil ringkasan menunjukkan damage dijadikan acuan pertama untuk memisahkan kelas gempa, diikut variabel lain yaitu mag, dan areaClass.



Gambar 9 Pohon keputusan hasil pemodelan

1. Melakukan klasifikasi data uji dengan model yang dihasilkan

|  |
| --- |
| > p <- predict.C5.0(model, testX)  > p\_all[[k]] <- summary(p)  > p\_all |

Gambar 10 Potongan kode klasifikasi C5.0

Setelah dilakukan pemodelan, data uji diklasifikasi menggunakan hasil pemodelan tersebut. Ringkasan hasil klasifikasi pada *fold* ke-7 dapat dilihat pada Gambar 11.

|  |
| --- |
| > p\_all[7]  nonrisiko risiko  13 6 |

Gambar 11 Hasil klasifikasi C5.0 pada data uji

1. Melakukan evaluasi hasil pemodelan dengan *confusion matrix*

|  |
| --- |
| > akurasi<- sum( p == testY ) / length( p )  > akurasi\_all[k] <- akurasi  > conf\_mat\_all[[k]] <- confusionMatrix(p, reference = testY)  > conf\_mat\_all |

Gambar 12 Potongan kode evaluasi hasil pemodelan dan klasifikasi

Evaluasi dilakukan terhadap hasil pemodelan dan klasifikasi yang dilakukan menggunakan *confusion matrix*. Hasil evaluasi pemodelan pada *fold* ke-7 dapat dilihat pada Gambar 13.

|  |
| --- |
| > akurasi\_all[7]  [[1]]  [1] 0.8947368  > conf\_mat\_all[7]  [[1]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 12 1  risiko 1 5    Accuracy : 0.8947  95% CI : (0.6686, 0.987)  No Information Rate : 0.6842  P-Value [Acc > NIR] : 0.03414    Kappa : 0.7564  Mcnemar's Test P-Value : 1.00000    Sensitivity : 0.9231  Specificity : 0.8333  Pos Pred Value : 0.9231  Neg Pred Value : 0.8333  Prevalence : 0.6842  Detection Rate : 0.6316  Detection Prevalence : 0.6842  Balanced Accuracy : 0.8782    'Positive' Class : nonrisiko |

Gambar 13 Hasil evaluasi pemodelan dan klasifikasi

Setelah itu, klasifikasi dengan menggunakan SVM juga dilakukan untuk membandingkan hasil akurasinya dengan algoritme C5.0. Potongan kode dan hasil klasifikasi serta akurasi SVM yang dibuat dapat dilihat pada Gambar 14.

|  |
| --- |
| * library(e1071) * model.svm <- svm(class.svm ~ ., data = train.svm) * predict.svm <- predict(model.svm, data.test) * akurasiS<- sum( predict.svm == testY ) / length( predict.svm ) * model\_allS[[k]] <- summary(model.svm) * p\_allS[[k]] <- summary(predict.svm) * akurasi\_allS[k] <- akurasiS * conf\_mat\_allS[[k]] <- confusionMatrix(predict.svm, reference = testY) * > p\_alls   [[1]]  nonrisiko risiko  15 5  [[2]]  nonrisiko risiko  14 6  [[3]]  nonrisiko risiko  14 6  [[4]]  nonrisiko risiko  12 8  [[5]]  nonrisiko risiko  13 7  [[6]]  nonrisiko risiko  12 7  [[7]]  nonrisiko risiko  13 6  [[8]]  nonrisiko risiko  11 8  [[9]]  nonrisiko risiko  13 6  [[10]]  nonrisiko risiko  13 6  > akurasi\_alls  [[1]]  [1] 0.9  [[2]]  [1] 0.95  [[3]]  [1] 1  [[4]]  [1] 1  [[5]]  [1] 1  [[6]]  [1] 0.9473684  [[7]]  [1] 1  [[8]]  [1] 1  [[9]]  [1] 1  [[10]]  [1] 0.8947368 |

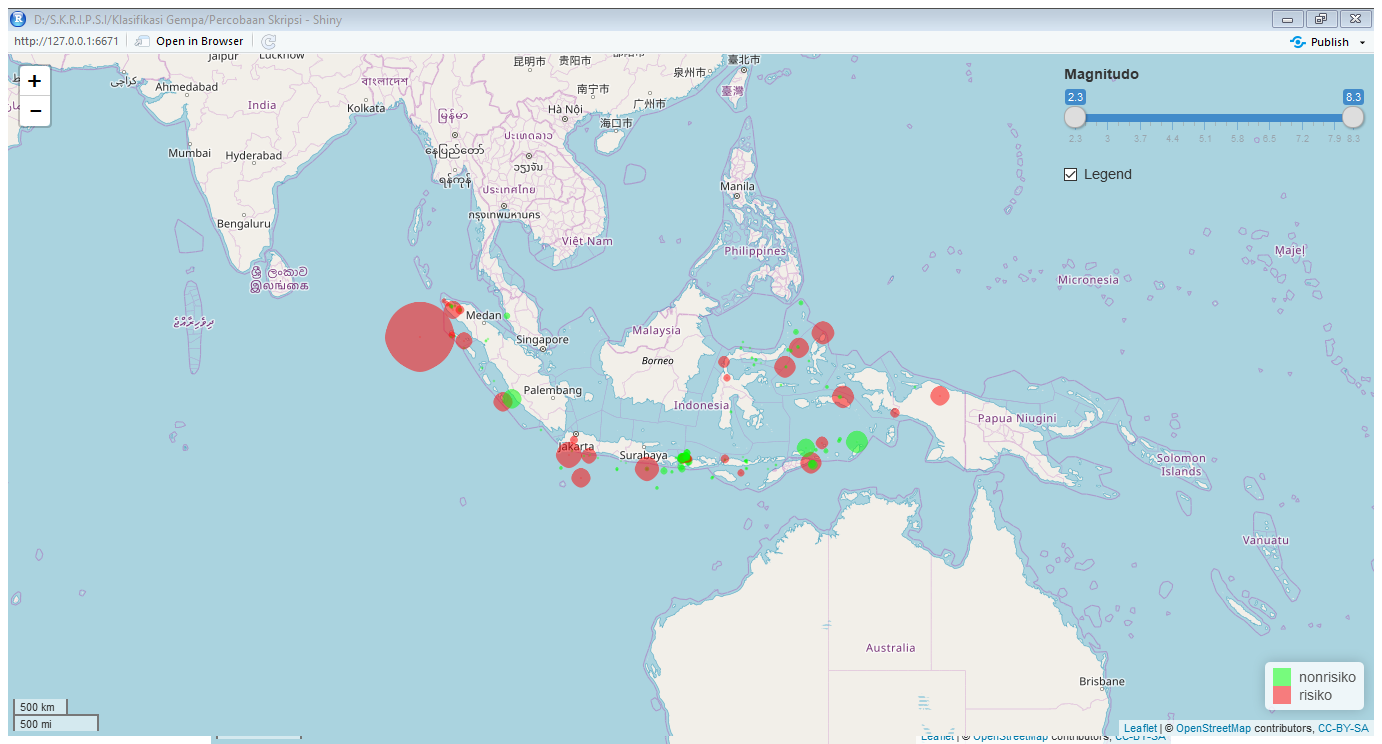
Gambar Potongan kode dan hasil SVM

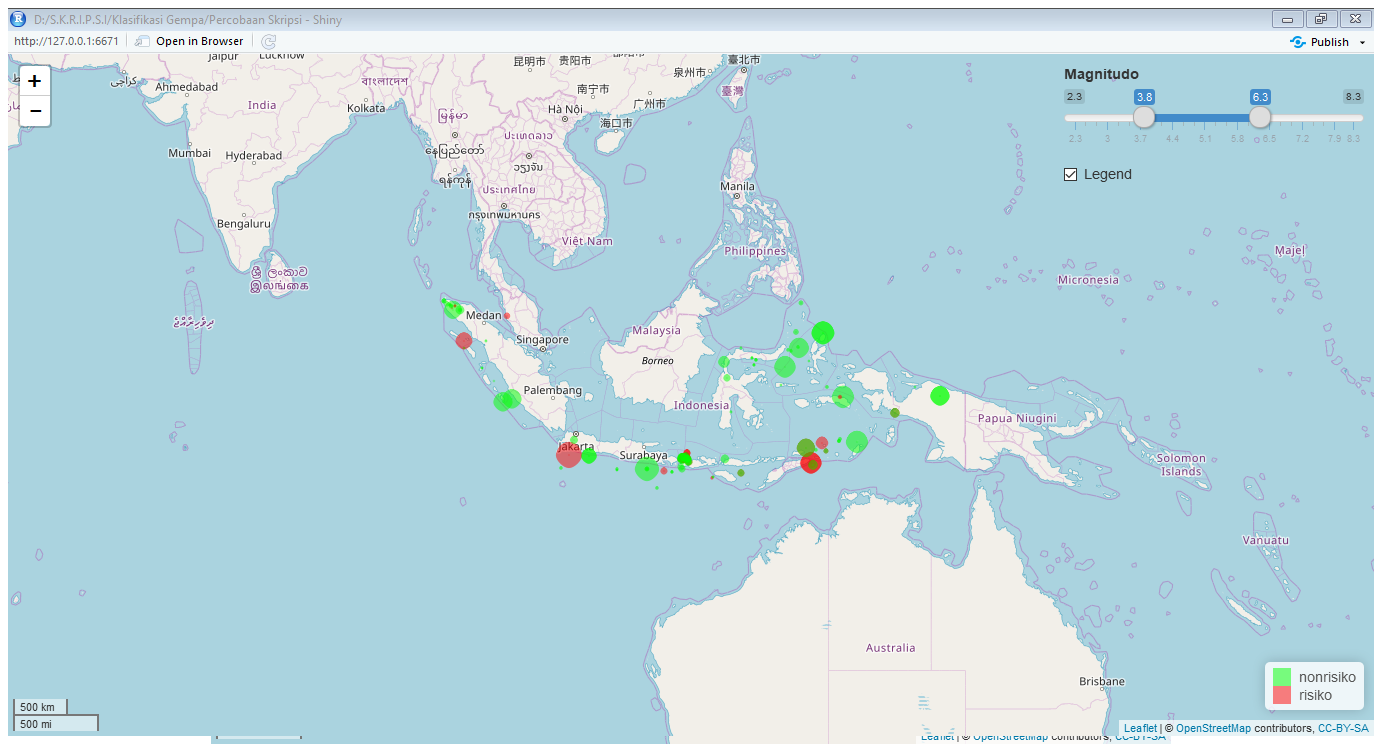
## Visualisasi dan Analisis Hasil Pemodelan

Visualisasi dilakukan untuk pemetaan titik-titik hasil klasifikasi gempa terhadap data uji. Visualisasi dibuat menggunakan *library* Leaflet pada *framework* Shiny menggunakan bahasa pemrograman R. Potongan kode visualisasi dapat dilihat pada Gambar 15 dan hasil visualisasi data uji dapat dilihat pada Gambar 16.

|  |
| --- |
| library(shiny)  library(leaflet)  ui <- bootstrapPage(  tags$style(type = "text/css", "html, body {width:100%;height:100%}"),  leafletOutput("mymap", width = "100%", height = "100%"),  absolutePanel(top = 10, right = 10,  sliderInput("range", "Magnitudo", min(gempaUjiVis$Mag), max(gempaUjiVis$Mag),  value = range(gempaUjiVis$Mag), step = 0.1  ),  checkboxInput("legend", "Legend", TRUE)  )  )  server <- function(input, output, session) {  filteredData <- reactive({  gempaUjiVis[(gempaUjiVis$Mag >= input$range[1] & gempaUjiVis$Mag <= input$range[2]),]  })    output$mymap <- renderLeaflet({  pal <- colorFactor(c("green", "red"), domain = c("nonrisiko","risiko"))    x <- 1  y <- 310    m <- leaflet(gempaUjiVis[,]) %>% fitBounds(90.79, -15.69, 143.00 ,10.40)  m %>% addTiles() %>% addScaleBar(position = "bottomleft", options = scaleBarOptions(maxWidth = 100, metric = TRUE))  })    observe({  pal <- colorFactor(c("green", "red"), domain = c("nonrisiko","risiko"))    leafletProxy("mymap", data = filteredData()) %>%  clearShapes() %>%  addCircles(lng = ~Longitude, lat = ~Latitude, weight = ~(smaj\*0.2), color = ~pal(gempaUjiVis$Class), popup =  paste("Tanggal Kejadian : ",gempaUjiVis$Date,"-",gempaUjiVis$Month,"-",gempaUjiVis$Year  ,"<br/>Magnitudo : ", gempaUjiVis$Mag, " sR"  ,"<br/>Kedalaman : ", gempaUjiVis$Depth, " km"  ,"<br/>Tingkat Resiko : ", gempaUjiVis$Class  ,"<br/>Lokasi (Longitude;Latitude) : ", gempaUjiVis$Longitude,"; ", gempaUjiVis$Latitude  )  )  }) |

Gambar 15 Kode visualisasi hasil klasifikasi data uji





Gambar 16 Hasil visualisasi data uji.

# SIMPULAN DAN SARAN

## Simpulan

Hasil klasifikasi skala gempa bumi berdasarkan algoritme Pohon Keputusan C5.0 telah dilakukan menggunakan *library* C5.0 pada bahasa pemrograman R dengan akurasi 89 hingga 100 persen. Hasil klasifikasi skala intensitas risiko gempa dibagi dalam dua kelas yaitu kelas 1 (Gempa yang memiliki risiko terjadinya kerusakan/korban) dan 0 (gempa nonrisiko) dengan jumlah yang bervariasi ditiap *fold*-nya. Hasil klasifikasi kebanyakan dibagi oleh variabel damage diawal dan dipengaruhi variabel lain seperti mag, smaj, areaClass, dan depth. Hasil pemetaan menggunakan OpenStreetMaps API terhadap titik-titik terjadinya gempa telah dilakukan dengan *library* Leaflet pada *framework* Shiny. Hasil visualisasi menggambarkan skala intensitas risiko tiap kejadian gempa dengan radius wilayah hingga stasiun pengukuran terjauh (smaj). Perbandingan akurasi antara algoritme C5.0 dengan SVM dapat dilihat pada Tabel 3. Rataan akurasi klasifikasi data antara C5.0 dengan SVM tidak berbeda jauh.

Tabel Perbandingan akurasi algoritme C5.0 dengan SVM.

| *fold* | C5.0 | SVM |
| --- | --- | --- |
| 1 | 0.9 | 0.9 |
| 2 | 1 | 0.95 |
| 3 | 1 | 1 |
| 4 | 1 | 1 |
| 5 | 1 | 1 |
| 6 | 0.9473684 | 0.9473684 |
| 7 | 0.8947368 | 1 |
| 8 | 0.9473684 | 1 |
| 9 | 1 | 1 |
| 10 | 0.9473684 | 0.8947368 |
| Rataan | 0.9636842 | 0.9692105 |

## Saran

Penelitian selanjutnya diharapkan dapat menambahkan klasifikasi tingkat risiko terjadinya tsunami pada titik kejadian gempa berdasarkan data yang ada. Selain itu, visualisasi yang dilakukan dapat ditambahkan gradasi warna untuk menandakan kedalaman titik pusat gempa maupun magnitudonya. Penelitian selanjutnya juga diharapkan dapat menambahkan *filter* data untuk menyaring data berdasarkan kedalaman dan tanggal kejadian. Pengambilan data *real-time* dari API BMKG juga dapat ditambahkan untuk penelitian selanjutnya. Selain itu, prediksi kemungkinan terjadinya gempa mungkin dapat dilakukan melalui analisis pola kejadian dari data gempa.

# DAFTAR PUSTAKA

Aprita D. 2016. Klasifikasi Kemunculan Titik Panas pada Lahan Gambut di Sumatera dan Kalimantan dengan Menggunakan Algoritme Pohon Keputusan C5.0 [Skripsi]. Bogor (ID): Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Pertanian Bogor.

Han J, Kamber M, Pei J. 2012. *Data Mining: Concepts and Techniques 3rd ed*. Massachusetts (US): Morgan Kaufmann Publishers.

Ikram A, Qamar U. 2017. Seismic Data Analysis for Earthquake Prediction: A Machine Learning Approach [Internet]. [Nama editor tidak diketahui]. 25th GISRUK Conference; 2017 April 18-21; Manchester, Inggris. Manchester (GB): University of Manchester. Chap. Posters 12. [Diunduh 2017 Juli 28]. Tersedia pada: <http://huckg.is/gisruk2017/GISRUK_2017_paper_12.pdf>.

Mitchell T M. 1997. *Machine Learning 1st ed*. New York (US): McGraw Hill.

Patil, N, Lathi R, Chitre V. 2012. Comparison of C5.0 and CART Classification algorithms using pruning technique. International Journal of Engineering Research and Technology [Internet]. [Waktu pembaharuan tidak diketahui]; [Diunduh 2017 Agustus 22]. 1 (4): [Lokasi tidak diketahui]. ISSN: 2278 – 0181. Tersedia pada: <http://www.ijert.org/download/204/comparison-of-c50-a-cart-classification-algorithms-using-pruning-technique>.

Santika M. 2017. Model Klasifikasi Nilai Akhir Mata Kuliah Data Mining berdasarkan Aktivitas Mahasiswa pada LMS menggunakan Pohon Keputusan [Skripsi]. Bogor (ID): Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Pertanian Bogor.

Wong T T, Yang N Y. 2017. Dependency Analysis of Accuracy Estimates in k-Fold Cross Validation [Internet]. *IEEE Transactions on Knowledge and Data Engineering*. 17 Agustus 2017; [Diunduh pada 2018 Mei 17]; 29(11):Piscataway(US). ISSN:1041-4347. Tersedia pada: <https://ieeexplore.ieee.org/document/8012491/>.

# LAMPIRAN

Lampiran 1 Kode program pemodelan, klasifikasi, dan pengujian C5.0

|  |
| --- |
| #Setting up workspace  setwd("D:/S.K.R.I.P.S.I/Klasifikasi Gempa/Data/110918/")  library(readr)  gempa1 <- read\_csv("dataLatih\_model.csv")  gempa <- gempa1  #converting to data frame  gempaDF <- gempa[order(runif(nrow(gempa))),]  gempaDF$Class <- as.factor(gempaDF$Class)  gempaDF = as.data.frame(gempaDF)  #oversampling  library(imbalance)  imbalanceRatio(gempaDF)  gempaOver <- oversample(gempaDF, ratio = 0.5, method = "SMOTE", classAttr = "Class")  str(gempaOver)  tail(gempaOver,10)  #dividing into train data and test data  gempaTrain <- gempaOver  library(caret)  library(C50)  library(e1071) #for SVM comparation  # actual cross validation  folds <- sample(rep(1:10, length.out = 195))  model\_all <- list()  trainX\_all <- list()  trainY\_all <- list()  testX\_all <- list()  testY\_all <- list()  p\_all <- list()  akurasi\_all <- list()  conf\_mat\_all <- list()  plot\_all <- list()  #SVM list  model\_alls <- list()  trainX\_alls <- list()  trainY\_alls <- list()  testX\_alls <- list()  testY\_alls <- list()  p\_alls <- list()  akurasi\_alls <- list()  conf\_mat\_alls <- list()  plot\_alls <- list()  for(k in 1:10) {  # actual split of the data  fold <- which(folds == k)  data.train <- gempaTrain[-fold,]  data.test <- gempaTrain[fold,]  trainX <- data.train[,2:9]  testX <- data.test[,2:9]  trainY <-data.train[,1]  testY <- data.test[,1]  class.svm <- gempaTrain[-fold,1]  data\_class.svm <- as.data.frame(class.svm)  train.svm <- gempaTrain[-fold,2:6]    model <- C5.0(trainX, trainY)  plot\_all[[k]] <- plot(model)  p <- predict.C5.0(model, testX)  akurasi<- sum( p == testY ) / length( p )  model\_all[[k]] <- summary(model)  p\_all[[k]] <- summary(p)  akurasi\_all[k] <- akurasi  conf\_mat\_all[[k]] <- confusionMatrix(p, reference = testY)  trainX\_all[[k]] <- trainX  trainY\_all[[k]] <- trainY    model.svm <- svm(class.svm ~ ., data = train.svm)  predict.svm <- predict(model.svm, data.test)    akurasis<- sum( predict.svm == testY ) / length( predict.svm )  model\_alls[[k]] <- summary(model.svm)  p\_alls[[k]] <- summary(predict.svm)  akurasi\_alls[k] <- akurasis  conf\_mat\_alls[[k]] <- confusionMatrix(predict.svm, reference = testY)    testX$Longitude <- data.test$Longitude  testX$Latitude <- data.test$Latitude  testX$Region <- data.test$Region  testX$Date <- data.test$Date  testX$Month <- data.test$Month  testX$Year <- data.test$Year  testX\_all[[k]] <- testX  testX\_all[[k]] <- testX  testY\_all[[k]] <- testY  }  for(k in 1:10){  print(paste0("fold",k))  print(table(testY\_all[[k]]))  }  for(k in 1:10){  print(paste0("fold",k))  print(table(trainY\_all[[k]]))  }  #showing result  model\_all  p\_all  conf\_mat\_all  akurasi\_all  model\_alls  p\_alls  conf\_mat\_alls  akurasi\_alls  gempaUjiVis <- gempaDF |

Lampiran 2 Kode program visualisasi dengan *framework* Shiny dan *library* Leaflet

|  |
| --- |
| library(shiny)  library(leaflet)  ui <- bootstrapPage(  tags$style(type = "text/css", "html, body {width:100%;height:100%}"),  leafletOutput("mymap", width = "100%", height = "100%"),  absolutePanel(top = 10, right = 10,  sliderInput("range", "Magnitudo", min(gempaUjiVis$Mag), max(gempaUjiVis$Mag),  value = range(gempaUjiVis$Mag), step = 0.1  ),  checkboxInput("legend", "Legend", TRUE)  )  )  server <- function(input, output, session) {  filteredData <- reactive({  gempaUjiVis[(gempaUjiVis$Mag >= input$range[1] & gempaUjiVis$Mag <= input$range[2]),]  })    output$mymap <- renderLeaflet({  pal <- colorFactor(c("green", "red"), domain = c("nonrisiko","risiko"))    x <- 1  y <- 310    m <- leaflet(gempaUjiVis[,]) %>% fitBounds(90.79, -15.69, 143.00 ,10.40)  m %>% addTiles() %>% addScaleBar(position = "bottomleft", options = scaleBarOptions(maxWidth = 100, metric = TRUE))  })    observe({  pal <- colorFactor(c("green", "red"), domain = c("nonrisiko","risiko"))    leafletProxy("mymap", data = filteredData()) %>%  clearShapes() %>%  addCircles(lng = ~Longitude, lat = ~Latitude, weight = ~(smaj\*0.2), color = ~pal(gempaUjiVis$Class), popup =  paste("Tanggal Kejadian : ",gempaUjiVis$Date,"-",gempaUjiVis$Month,"-",gempaUjiVis$Year  ,"<br/>Magnitudo : ", gempaUjiVis$Mag, " sR"  ,"<br/>Kedalaman : ", gempaUjiVis$Depth, " km"  ,"<br/>Tingkat Resiko : ", gempaUjiVis$Class  ,"<br/>Lokasi (Longitude;Latitude) : ", gempaUjiVis$Longitude,"; ", gempaUjiVis$Latitude  )  )  })    observe({  proxy <- leafletProxy("mymap", data = gempaUjiVis)    if (input$legend) {  pal <- colorFactor(c("green","red"), domain = c("nonrisiko","risiko"))  proxy %>% addLegend(position = "bottomright",  pal = pal, values = c("nonrisiko","risiko")  )  }  })    }  shinyApp(ui, server) |

Lampiran 3 Pembagian data uji ditiap *fold*

|  |
| --- |
| [1] "fold1"  nonrisiko risiko  13 7  [1] "fold2"  nonrisiko risiko  13 7  [1] "fold3"  nonrisiko risiko  14 6  [1] "fold4"  nonrisiko risiko  12 8  [1] "fold5"  nonrisiko risiko  13 7  [1] "fold6"  nonrisiko risiko  13 6  [1] "fold7"  nonrisiko risiko  13 6  [1] "fold8"  nonrisiko risiko  11 8  [1] "fold9"  nonrisiko risiko  13 6  [1] "fold10"  nonrisiko risiko  15 4 |

Lampiran 4 Pembagian data latih ditiap *fold*

|  |
| --- |
| [1] "fold1"  nonrisiko risiko  117 58  [1] "fold2"  nonrisiko risiko  117 58  [1] "fold3"  nonrisiko risiko  116 59  [1] "fold4"  nonrisiko risiko  118 57  [1] "fold5"  nonrisiko risiko  117 58  [1] "fold6"  nonrisiko risiko  117 59  [1] "fold7"  nonrisiko risiko  117 59  [1] "fold8"  nonrisiko risiko  119 57  [1] "fold9"  nonrisiko risiko  117 59  [1] "fold10"  nonrisiko risiko  115 61 |

Lampiran 5 Hasil pemodelan ditiap *fold*

|  |
| --- |
| > model\_all  [[1]]  Call:  C5.0.default(x = trainX, y = trainY)  C5.0 [Release 2.07 GPL Edition] Tue Sep 18 11:40:02 2018  -------------------------------  Class specified by attribute `outcome'  Read 175 cases (9 attributes) from undefined.data  Decision tree:  damage > 0: risiko (35)  damage <= 0:  :...smaj <= 39.41: nonrisiko (114)  smaj > 39.41:  :...Mag <= 5.3: nonrisiko (2)  Mag > 5.3: risiko (24/1)  Evaluation on training data (175 cases):  Decision Tree  ----------------  Size Errors  4 1( 0.6%) <<  (a) (b) <-classified as  ---- ----  116 1 (a): class nonrisiko  58 (b): class risiko  Attribute usage:  100.00% damage  80.00% smaj  14.86% Mag  Time: 0.0 secs  [[2]]  Call:  C5.0.default(x = trainX, y = trainY)  C5.0 [Release 2.07 GPL Edition] Tue Sep 18 11:40:02 2018  -------------------------------  Class specified by attribute `outcome'  Read 175 cases (9 attributes) from undefined.data  Decision tree:  damage > 0: risiko (33)  damage <= 0:  :...Mag <= 5.354007: nonrisiko (114)  Mag > 5.354007:  :...smaj <= 22.45929: nonrisiko (2)  smaj > 22.45929: risiko (26/1)  Evaluation on training data (175 cases):  Decision Tree  ----------------  Size Errors  4 1( 0.6%) <<  (a) (b) <-classified as  ---- ----  116 1 (a): class nonrisiko  58 (b): class risiko  Attribute usage:  100.00% damage  81.14% Mag  16.00% smaj  Time: 0.0 secs  [[3]]  Call:  C5.0.default(x = trainX, y = trainY)  C5.0 [Release 2.07 GPL Edition] Tue Sep 18 11:40:02 2018  -------------------------------  Class specified by attribute `outcome'  Read 175 cases (9 attributes) from undefined.data  Decision tree:  damage > 0: risiko (35)  damage <= 0:  :...Mag <= 5.354007: nonrisiko (113)  Mag > 5.354007:  :...smaj <= 24: nonrisiko (2)  smaj > 24: risiko (25/1)  Evaluation on training data (175 cases):  Decision Tree  ----------------  Size Errors  4 1( 0.6%) <<  (a) (b) <-classified as  ---- ----  115 1 (a): class nonrisiko  59 (b): class risiko  Attribute usage:  100.00% damage  80.00% Mag  15.43% smaj  Time: 0.0 secs  [[4]]  Call:  C5.0.default(x = trainX, y = trainY)  C5.0 [Release 2.07 GPL Edition] Tue Sep 18 11:40:02 2018  -------------------------------  Class specified by attribute `outcome'  Read 175 cases (9 attributes) from undefined.data  Decision tree:  damage > 0: risiko (35)  damage <= 0:  :...Mag <= 5.354007: nonrisiko (115)  Mag > 5.354007:  :...smaj <= 24: nonrisiko (2)  smaj > 24: risiko (23/1)  Evaluation on training data (175 cases):  Decision Tree  ----------------  Size Errors  4 1( 0.6%) <<  (a) (b) <-classified as  ---- ----  117 1 (a): class nonrisiko  57 (b): class risiko  Attribute usage:  100.00% damage  80.00% Mag  14.29% smaj  Time: 0.0 secs  [[5]]  Call:  C5.0.default(x = trainX, y = trainY)  C5.0 [Release 2.07 GPL Edition] Tue Sep 18 11:40:03 2018  -------------------------------  Class specified by attribute `outcome'  Read 175 cases (9 attributes) from undefined.data  Decision tree:  damage > 0: risiko (32)  damage <= 0:  :...Mag <= 5.354007: nonrisiko (114)  Mag > 5.354007:  :...smaj <= 24: nonrisiko (2)  smaj > 24: risiko (27/1)  Evaluation on training data (175 cases):  Decision Tree  ----------------  Size Errors  4 1( 0.6%) <<  (a) (b) <-classified as  ---- ----  116 1 (a): class nonrisiko  58 (b): class risiko  Attribute usage:  100.00% damage  81.71% Mag  16.57% smaj  Time: 0.0 secs  [[6]]  Call:  C5.0.default(x = trainX, y = trainY)  C5.0 [Release 2.07 GPL Edition] Tue Sep 18 11:40:03 2018  -------------------------------  Class specified by attribute `outcome'  Read 176 cases (9 attributes) from undefined.data  Decision tree:  damage > 0: risiko (35)  damage <= 0:  :...Mag <= 5.354007: nonrisiko (115)  Mag > 5.354007:  :...smaj <= 24: nonrisiko (2)  smaj > 24: risiko (24)  Evaluation on training data (176 cases):  Decision Tree  ----------------  Size Errors  4 0( 0.0%) <<  (a) (b) <-classified as  ---- ----  117 (a): class nonrisiko  59 (b): class risiko  Attribute usage:  100.00% damage  80.11% Mag  14.77% smaj  Time: 0.0 secs  [[7]]  Call:  C5.0.default(x = trainX, y = trainY)  C5.0 [Release 2.07 GPL Edition] Tue Sep 18 11:40:03 2018  -------------------------------  Class specified by attribute `outcome'  Read 176 cases (9 attributes) from undefined.data  Decision tree:  damage > 0: risiko (33)  damage <= 0:  :...Mag <= 5.3: nonrisiko (115)  Mag > 5.3:  :...areaClass <= 1: risiko (25)  areaClass > 1: nonrisiko (3/1)  Evaluation on training data (176 cases):  Decision Tree  ----------------  Size Errors  4 1( 0.6%) <<  (a) (b) <-classified as  ---- ----  117 (a): class nonrisiko  1 58 (b): class risiko  Attribute usage:  100.00% damage  81.25% Mag  15.91% areaClass  Time: 0.0 secs  [[8]]  Call:  C5.0.default(x = trainX, y = trainY)  C5.0 [Release 2.07 GPL Edition] Tue Sep 18 11:40:03 2018  -------------------------------  Class specified by attribute `outcome'  Read 176 cases (9 attributes) from undefined.data  Decision tree:  damage > 0: risiko (33)  damage <= 0:  :...Mag <= 5.354007: nonrisiko (117)  Mag > 5.354007: risiko (26/2)  Evaluation on training data (176 cases):  Decision Tree  ----------------  Size Errors  3 2( 1.1%) <<  (a) (b) <-classified as  ---- ----  117 2 (a): class nonrisiko  57 (b): class risiko  Attribute usage:  100.00% damage  81.25% Mag  Time: 0.0 secs  [[9]]  Call:  C5.0.default(x = trainX, y = trainY)  C5.0 [Release 2.07 GPL Edition] Tue Sep 18 11:40:03 2018  -------------------------------  Class specified by attribute `outcome'  Read 176 cases (9 attributes) from undefined.data  Decision tree:  damage > 0: risiko (36)  damage <= 0:  :...Mag <= 5.354007: nonrisiko (114)  Mag > 5.354007:  :...smaj <= 24: nonrisiko (2)  smaj > 24: risiko (24/1)  Evaluation on training data (176 cases):  Decision Tree  ----------------  Size Errors  4 1( 0.6%) <<  (a) (b) <-classified as  ---- ----  116 1 (a): class nonrisiko  59 (b): class risiko  Attribute usage:  100.00% damage  79.55% Mag  14.77% smaj  Time: 0.0 secs  [[10]]  Call:  C5.0.default(x = trainX, y = trainY)  C5.0 [Release 2.07 GPL Edition] Tue Sep 18 11:40:04 2018  -------------------------------  Class specified by attribute `outcome'  Read 176 cases (9 attributes) from undefined.data  Decision tree:  damage > 0: risiko (35)  damage <= 0:  :...smaj <= 34: nonrisiko (113)  smaj > 34:  :...Depth <= 58: risiko (25)  Depth > 58: nonrisiko (3/1)  Evaluation on training data (176 cases):  Decision Tree  ----------------  Size Errors  4 1( 0.6%) <<  (a) (b) <-classified as  ---- ----  115 (a): class nonrisiko  1 60 (b): class risiko  Attribute usage:  100.00% damage  80.11% smaj  15.91% Depth  Time: 0.0 secs  > model\_alls  [[1]]  Call:  svm(formula = class.svm ~ ., data = train.svm)  Parameters:  SVM-Type: C-classification  SVM-Kernel: radial  cost: 1  gamma: 0.2  Number of Support Vectors: 28  ( 13 15 )  Number of Classes: 2  Levels:  nonrisiko risiko  [[2]]  Call:  svm(formula = class.svm ~ ., data = train.svm)  Parameters:  SVM-Type: C-classification  SVM-Kernel: radial  cost: 1  gamma: 0.2  Number of Support Vectors: 31  ( 14 17 )  Number of Classes: 2  Levels:  nonrisiko risiko  [[3]]  Call:  svm(formula = class.svm ~ ., data = train.svm)  Parameters:  SVM-Type: C-classification  SVM-Kernel: radial  cost: 1  gamma: 0.2  Number of Support Vectors: 33  ( 16 17 )  Number of Classes: 2  Levels:  nonrisiko risiko  [[4]]  Call:  svm(formula = class.svm ~ ., data = train.svm)  Parameters:  SVM-Type: C-classification  SVM-Kernel: radial  cost: 1  gamma: 0.2  Number of Support Vectors: 32  ( 15 17 )  Number of Classes: 2  Levels:  nonrisiko risiko  [[5]]  Call:  svm(formula = class.svm ~ ., data = train.svm)  Parameters:  SVM-Type: C-classification  SVM-Kernel: radial  cost: 1  gamma: 0.2  Number of Support Vectors: 33  ( 16 17 )  Number of Classes: 2  Levels:  nonrisiko risiko  [[6]]  Call:  svm(formula = class.svm ~ ., data = train.svm)  Parameters:  SVM-Type: C-classification  SVM-Kernel: radial  cost: 1  gamma: 0.2  Number of Support Vectors: 30  ( 14 16 )  Number of Classes: 2  Levels:  nonrisiko risiko  [[7]]  Call:  svm(formula = class.svm ~ ., data = train.svm)  Parameters:  SVM-Type: C-classification  SVM-Kernel: radial  cost: 1  gamma: 0.2  Number of Support Vectors: 32  ( 15 17 )  Number of Classes: 2  Levels:  nonrisiko risiko  [[8]]  Call:  svm(formula = class.svm ~ ., data = train.svm)  Parameters:  SVM-Type: C-classification  SVM-Kernel: radial  cost: 1  gamma: 0.2  Number of Support Vectors: 30  ( 14 16 )  Number of Classes: 2  Levels:  nonrisiko risiko  [[9]]  Call:  svm(formula = class.svm ~ ., data = train.svm)  Parameters:  SVM-Type: C-classification  SVM-Kernel: radial  cost: 1  gamma: 0.2  Number of Support Vectors: 29  ( 14 15 )  Number of Classes: 2  Levels:  nonrisiko risiko  [[10]]  Call:  svm(formula = class.svm ~ ., data = train.svm)  Parameters:  SVM-Type: C-classification  SVM-Kernel: radial  cost: 1  gamma: 0.2  Number of Support Vectors: 30  ( 14 16 )  Number of Classes: 2  Levels:  nonrisiko risiko |

Lampiran 6 Hasil prediksi kelas ditiap *fold*

|  |
| --- |
| > p\_all  [[1]]  nonrisiko risiko  15 5  [[2]]  nonrisiko risiko  13 7  [[3]]  nonrisiko risiko  14 6  [[4]]  nonrisiko risiko  12 8  [[5]]  nonrisiko risiko  13 7  [[6]]  nonrisiko risiko  12 7  [[7]]  nonrisiko risiko  13 6  [[8]]  nonrisiko risiko  10 9  [[9]]  nonrisiko risiko  13 6  [[10]]  nonrisiko risiko  14 5  > p\_alls  [[1]]  nonrisiko risiko  15 5  [[2]]  nonrisiko risiko  14 6  [[3]]  nonrisiko risiko  14 6  [[4]]  nonrisiko risiko  12 8  [[5]]  nonrisiko risiko  13 7  [[6]]  nonrisiko risiko  12 7  [[7]]  nonrisiko risiko  13 6  [[8]]  nonrisiko risiko  11 8  [[9]]  nonrisiko risiko  13 6  [[10]]  nonrisiko risiko  13 6 |

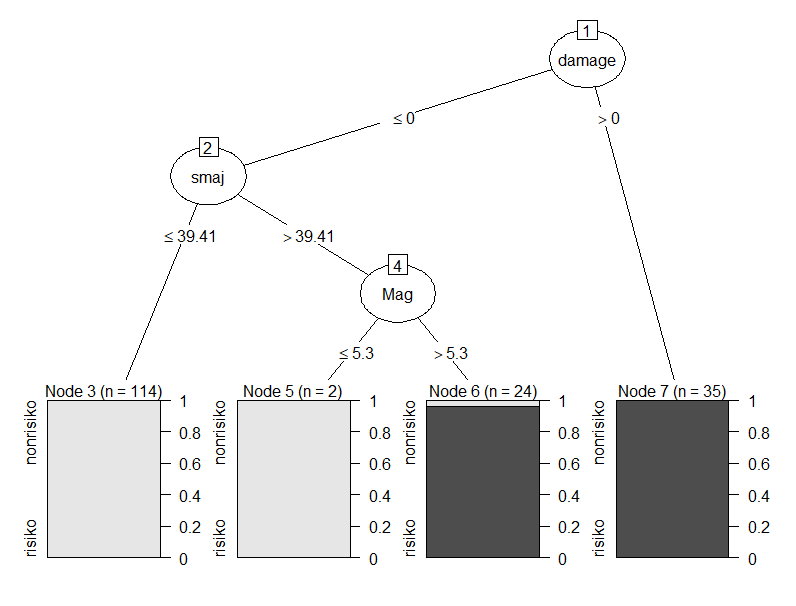
Lampiran 7 Hasil akurasi ditiap *fold*

|  |
| --- |
| > akurasi\_all  [[1]]  [1] 0.9  [[2]]  [1] 1  [[3]]  [1] 1  [[4]]  [1] 1  [[5]]  [1] 1  [[6]]  [1] 0.9473684  [[7]]  [1] 0.8947368  [[8]]  [1] 0.9473684  [[9]]  [1] 1  [[10]]  [1] 0.9473684  > akurasi\_alls  [[1]]  [1] 0.9  [[2]]  [1] 0.95  [[3]]  [1] 1  [[4]]  [1] 1  [[5]]  [1] 1  [[6]]  [1] 0.9473684  [[7]]  [1] 1  [[8]]  [1] 1  [[9]]  [1] 1  [[10]]  [1] 0.8947368 |

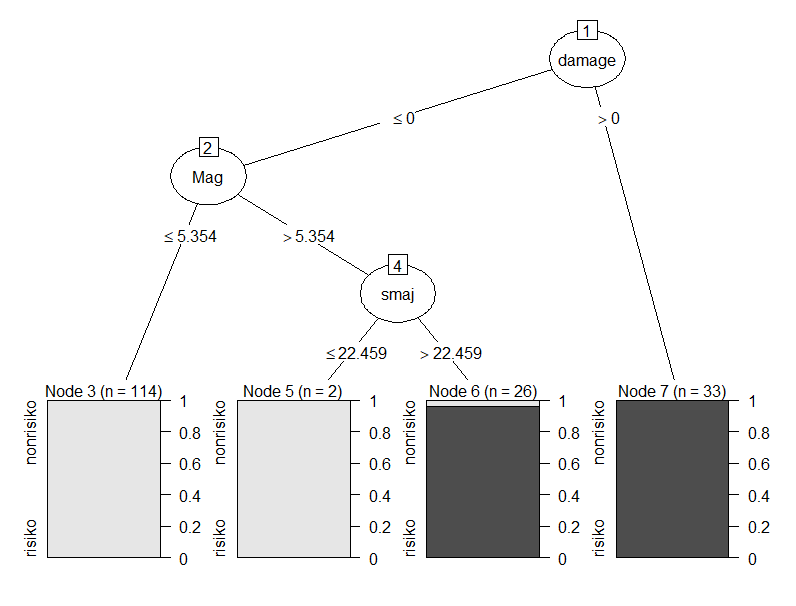
Lampiran 8 *Confusion matrix* hasil pengujian ditiap *fold*

|  |
| --- |
| > conf\_mat\_all  [[1]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 13 2  risiko 0 5    Accuracy : 0.9  95% CI : (0.683, 0.9877)  No Information Rate : 0.65  P-Value [Acc > NIR] : 0.01212    Kappa : 0.7647  Mcnemar's Test P-Value : 0.47950    Sensitivity : 1.0000  Specificity : 0.7143  Pos Pred Value : 0.8667  Neg Pred Value : 1.0000  Prevalence : 0.6500  Detection Rate : 0.6500  Detection Prevalence : 0.7500  Balanced Accuracy : 0.8571    'Positive' Class : nonrisiko    [[2]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 13 0  risiko 0 7    Accuracy : 1  95% CI : (0.8316, 1)  No Information Rate : 0.65  P-Value [Acc > NIR] : 0.0001812    Kappa : 1  Mcnemar's Test P-Value : NA    Sensitivity : 1.00  Specificity : 1.00  Pos Pred Value : 1.00  Neg Pred Value : 1.00  Prevalence : 0.65  Detection Rate : 0.65  Detection Prevalence : 0.65  Balanced Accuracy : 1.00    'Positive' Class : nonrisiko    [[3]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 14 0  risiko 0 6    Accuracy : 1  95% CI : (0.8316, 1)  No Information Rate : 0.7  P-Value [Acc > NIR] : 0.0007979    Kappa : 1  Mcnemar's Test P-Value : NA    Sensitivity : 1.0  Specificity : 1.0  Pos Pred Value : 1.0  Neg Pred Value : 1.0  Prevalence : 0.7  Detection Rate : 0.7  Detection Prevalence : 0.7  Balanced Accuracy : 1.0    'Positive' Class : nonrisiko    [[4]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 12 0  risiko 0 8    Accuracy : 1  95% CI : (0.8316, 1)  No Information Rate : 0.6  P-Value [Acc > NIR] : 3.656e-05    Kappa : 1  Mcnemar's Test P-Value : NA    Sensitivity : 1.0  Specificity : 1.0  Pos Pred Value : 1.0  Neg Pred Value : 1.0  Prevalence : 0.6  Detection Rate : 0.6  Detection Prevalence : 0.6  Balanced Accuracy : 1.0    'Positive' Class : nonrisiko    [[5]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 13 0  risiko 0 7    Accuracy : 1  95% CI : (0.8316, 1)  No Information Rate : 0.65  P-Value [Acc > NIR] : 0.0001812    Kappa : 1  Mcnemar's Test P-Value : NA    Sensitivity : 1.00  Specificity : 1.00  Pos Pred Value : 1.00  Neg Pred Value : 1.00  Prevalence : 0.65  Detection Rate : 0.65  Detection Prevalence : 0.65  Balanced Accuracy : 1.00    'Positive' Class : nonrisiko    [[6]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 12 0  risiko 1 6    Accuracy : 0.9474  95% CI : (0.7397, 0.9987)  No Information Rate : 0.6842  P-Value [Acc > NIR] : 0.007219    Kappa : 0.8834  Mcnemar's Test P-Value : 1.000000    Sensitivity : 0.9231  Specificity : 1.0000  Pos Pred Value : 1.0000  Neg Pred Value : 0.8571  Prevalence : 0.6842  Detection Rate : 0.6316  Detection Prevalence : 0.6316  Balanced Accuracy : 0.9615    'Positive' Class : nonrisiko    [[7]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 12 1  risiko 1 5    Accuracy : 0.8947  95% CI : (0.6686, 0.987)  No Information Rate : 0.6842  P-Value [Acc > NIR] : 0.03414    Kappa : 0.7564  Mcnemar's Test P-Value : 1.00000    Sensitivity : 0.9231  Specificity : 0.8333  Pos Pred Value : 0.9231  Neg Pred Value : 0.8333  Prevalence : 0.6842  Detection Rate : 0.6316  Detection Prevalence : 0.6842  Balanced Accuracy : 0.8782    'Positive' Class : nonrisiko    [[8]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 10 0  risiko 1 8    Accuracy : 0.9474  95% CI : (0.7397, 0.9987)  No Information Rate : 0.5789  P-Value [Acc > NIR] : 0.0004581    Kappa : 0.8939  Mcnemar's Test P-Value : 1.0000000    Sensitivity : 0.9091  Specificity : 1.0000  Pos Pred Value : 1.0000  Neg Pred Value : 0.8889  Prevalence : 0.5789  Detection Rate : 0.5263  Detection Prevalence : 0.5263  Balanced Accuracy : 0.9545    'Positive' Class : nonrisiko    [[9]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 13 0  risiko 0 6    Accuracy : 1  95% CI : (0.8235, 1)  No Information Rate : 0.6842  P-Value [Acc > NIR] : 0.0007389    Kappa : 1  Mcnemar's Test P-Value : NA    Sensitivity : 1.0000  Specificity : 1.0000  Pos Pred Value : 1.0000  Neg Pred Value : 1.0000  Prevalence : 0.6842  Detection Rate : 0.6842  Detection Prevalence : 0.6842  Balanced Accuracy : 1.0000    'Positive' Class : nonrisiko    [[10]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 14 0  risiko 1 4    Accuracy : 0.9474  95% CI : (0.7397, 0.9987)  No Information Rate : 0.7895  P-Value [Acc > NIR] : 0.06798    Kappa : 0.855  Mcnemar's Test P-Value : 1.00000    Sensitivity : 0.9333  Specificity : 1.0000  Pos Pred Value : 1.0000  Neg Pred Value : 0.8000  Prevalence : 0.7895  Detection Rate : 0.7368  Detection Prevalence : 0.7368  Balanced Accuracy : 0.9667    'Positive' Class : nonrisiko  > conf\_mat\_alls  [[1]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 13 2  risiko 0 5    Accuracy : 0.9  95% CI : (0.683, 0.9877)  No Information Rate : 0.65  P-Value [Acc > NIR] : 0.01212    Kappa : 0.7647  Mcnemar's Test P-Value : 0.47950    Sensitivity : 1.0000  Specificity : 0.7143  Pos Pred Value : 0.8667  Neg Pred Value : 1.0000  Prevalence : 0.6500  Detection Rate : 0.6500  Detection Prevalence : 0.7500  Balanced Accuracy : 0.8571    'Positive' Class : nonrisiko    [[2]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 13 1  risiko 0 6    Accuracy : 0.95  95% CI : (0.7513, 0.9987)  No Information Rate : 0.65  P-Value [Acc > NIR] : 0.002133    Kappa : 0.8864  Mcnemar's Test P-Value : 1.000000    Sensitivity : 1.0000  Specificity : 0.8571  Pos Pred Value : 0.9286  Neg Pred Value : 1.0000  Prevalence : 0.6500  Detection Rate : 0.6500  Detection Prevalence : 0.7000  Balanced Accuracy : 0.9286    'Positive' Class : nonrisiko    [[3]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 14 0  risiko 0 6    Accuracy : 1  95% CI : (0.8316, 1)  No Information Rate : 0.7  P-Value [Acc > NIR] : 0.0007979    Kappa : 1  Mcnemar's Test P-Value : NA    Sensitivity : 1.0  Specificity : 1.0  Pos Pred Value : 1.0  Neg Pred Value : 1.0  Prevalence : 0.7  Detection Rate : 0.7  Detection Prevalence : 0.7  Balanced Accuracy : 1.0    'Positive' Class : nonrisiko    [[4]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 12 0  risiko 0 8    Accuracy : 1  95% CI : (0.8316, 1)  No Information Rate : 0.6  P-Value [Acc > NIR] : 3.656e-05    Kappa : 1  Mcnemar's Test P-Value : NA    Sensitivity : 1.0  Specificity : 1.0  Pos Pred Value : 1.0  Neg Pred Value : 1.0  Prevalence : 0.6  Detection Rate : 0.6  Detection Prevalence : 0.6  Balanced Accuracy : 1.0    'Positive' Class : nonrisiko    [[5]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 13 0  risiko 0 7    Accuracy : 1  95% CI : (0.8316, 1)  No Information Rate : 0.65  P-Value [Acc > NIR] : 0.0001812    Kappa : 1  Mcnemar's Test P-Value : NA    Sensitivity : 1.00  Specificity : 1.00  Pos Pred Value : 1.00  Neg Pred Value : 1.00  Prevalence : 0.65  Detection Rate : 0.65  Detection Prevalence : 0.65  Balanced Accuracy : 1.00    'Positive' Class : nonrisiko    [[6]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 12 0  risiko 1 6    Accuracy : 0.9474  95% CI : (0.7397, 0.9987)  No Information Rate : 0.6842  P-Value [Acc > NIR] : 0.007219    Kappa : 0.8834  Mcnemar's Test P-Value : 1.000000    Sensitivity : 0.9231  Specificity : 1.0000  Pos Pred Value : 1.0000  Neg Pred Value : 0.8571  Prevalence : 0.6842  Detection Rate : 0.6316  Detection Prevalence : 0.6316  Balanced Accuracy : 0.9615    'Positive' Class : nonrisiko    [[7]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 13 0  risiko 0 6    Accuracy : 1  95% CI : (0.8235, 1)  No Information Rate : 0.6842  P-Value [Acc > NIR] : 0.0007389    Kappa : 1  Mcnemar's Test P-Value : NA    Sensitivity : 1.0000  Specificity : 1.0000  Pos Pred Value : 1.0000  Neg Pred Value : 1.0000  Prevalence : 0.6842  Detection Rate : 0.6842  Detection Prevalence : 0.6842  Balanced Accuracy : 1.0000    'Positive' Class : nonrisiko    [[8]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 11 0  risiko 0 8    Accuracy : 1  95% CI : (0.8235, 1)  No Information Rate : 0.5789  P-Value [Acc > NIR] : 3.091e-05    Kappa : 1  Mcnemar's Test P-Value : NA    Sensitivity : 1.0000  Specificity : 1.0000  Pos Pred Value : 1.0000  Neg Pred Value : 1.0000  Prevalence : 0.5789  Detection Rate : 0.5789  Detection Prevalence : 0.5789  Balanced Accuracy : 1.0000    'Positive' Class : nonrisiko    [[9]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 13 0  risiko 0 6    Accuracy : 1  95% CI : (0.8235, 1)  No Information Rate : 0.6842  P-Value [Acc > NIR] : 0.0007389    Kappa : 1  Mcnemar's Test P-Value : NA    Sensitivity : 1.0000  Specificity : 1.0000  Pos Pred Value : 1.0000  Neg Pred Value : 1.0000  Prevalence : 0.6842  Detection Rate : 0.6842  Detection Prevalence : 0.6842  Balanced Accuracy : 1.0000    'Positive' Class : nonrisiko    [[10]]  Confusion Matrix and Statistics  Reference  Prediction nonrisiko risiko  nonrisiko 13 0  risiko 2 4    Accuracy : 0.8947  95% CI : (0.6686, 0.987)  No Information Rate : 0.7895  P-Value [Acc > NIR] : 0.2042    Kappa : 0.7324  Mcnemar's Test P-Value : 0.4795    Sensitivity : 0.8667  Specificity : 1.0000  Pos Pred Value : 1.0000  Neg Pred Value : 0.6667  Prevalence : 0.7895  Detection Rate : 0.6842  Detection Prevalence : 0.6842  Balanced Accuracy : 0.9333    'Positive' Class : nonrisiko |

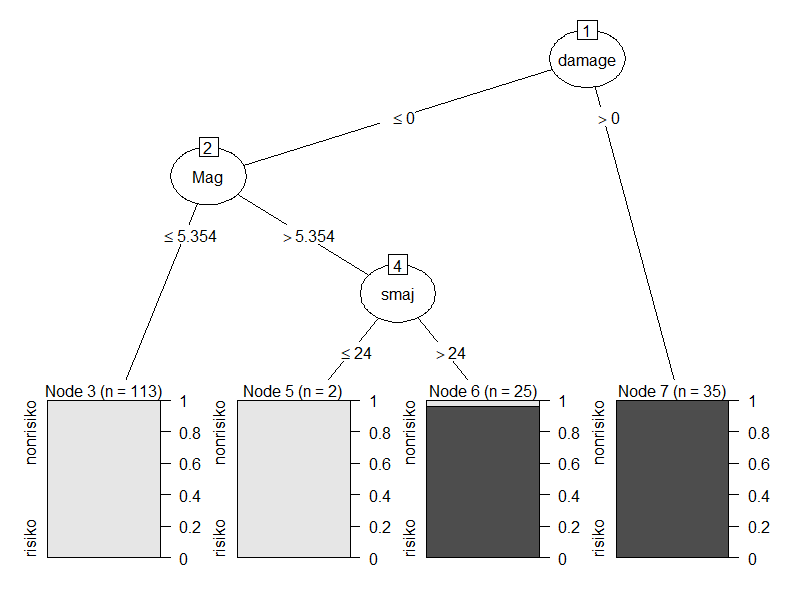
Lampiran 9 Pohon keputusan hasil pemodelan pada *fold* ke-1



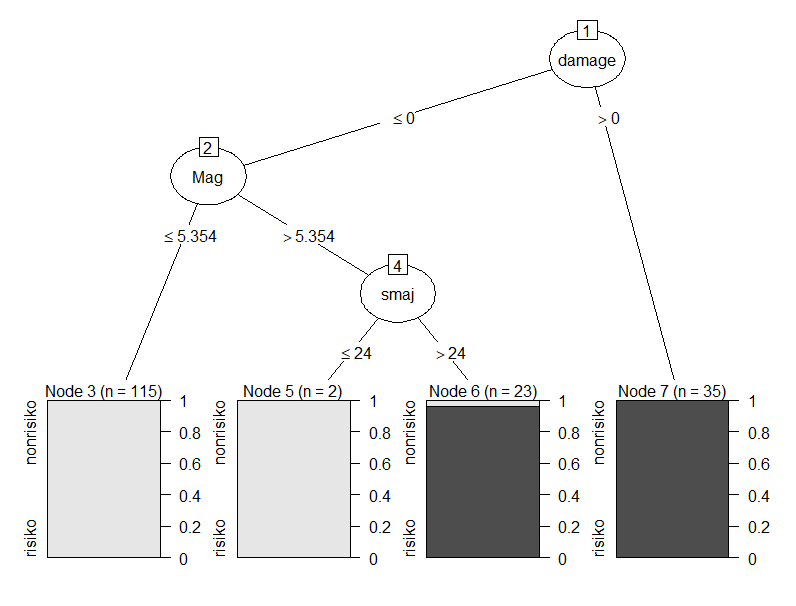
Lampiran 10 Pohon keputusan hasil pemodelan pada *fold* ke-2



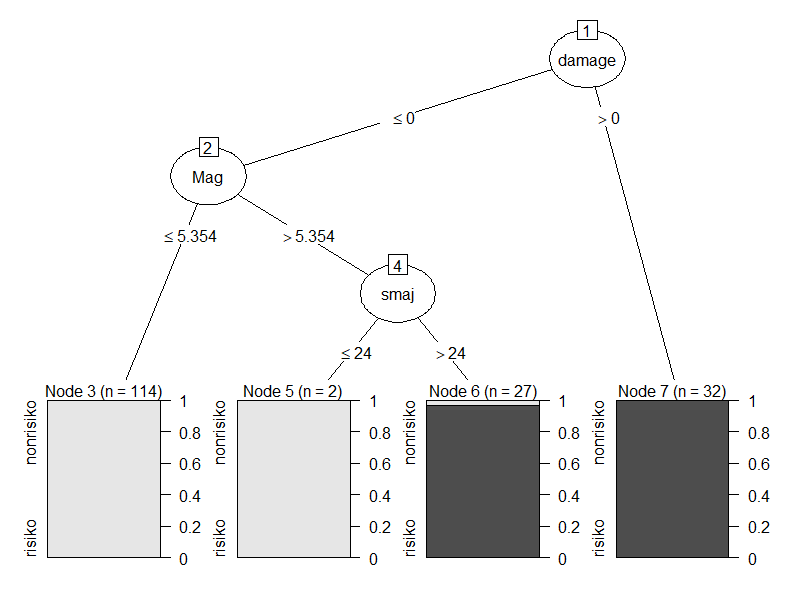
Lampiran 11 Pohon keputusan hasil pemodelan pada *fold* ke-3



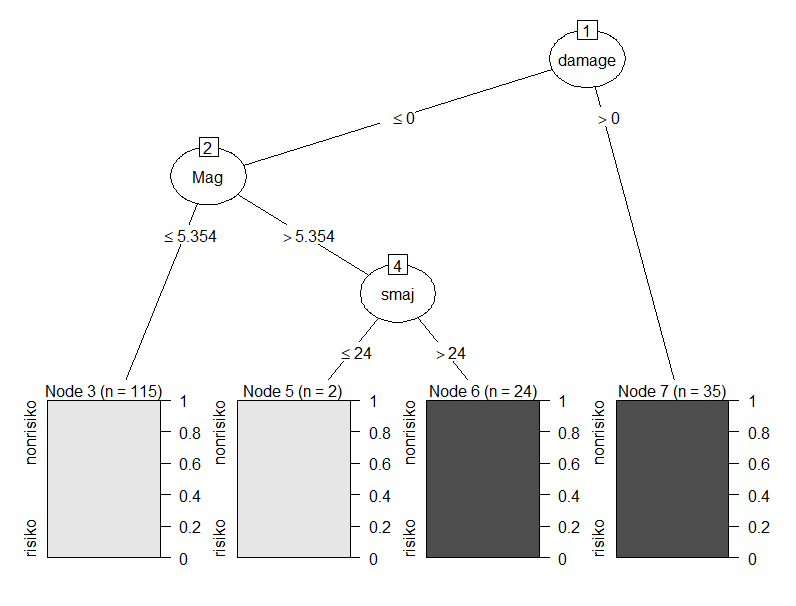
Lampiran 12 Pohon keputusan hasil pemodelan pada *fold* ke-4



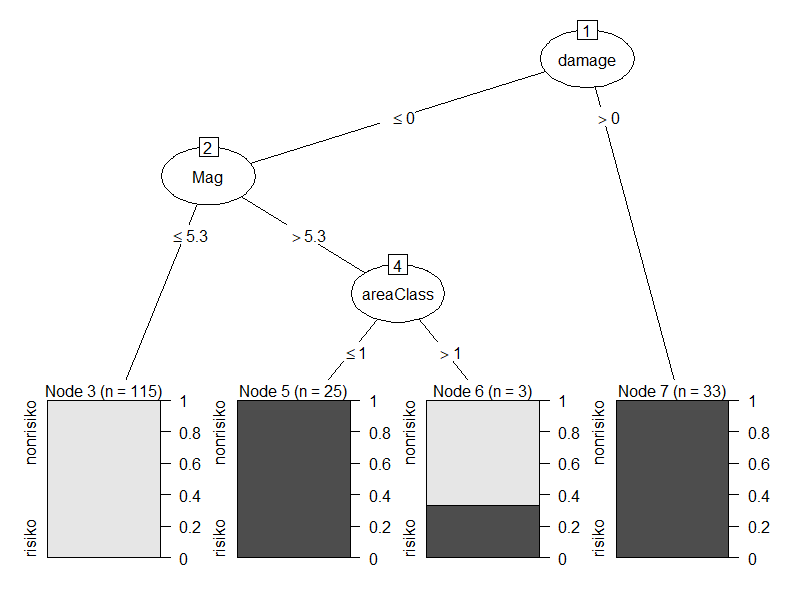
Lampiran 13 Pohon keputusan hasil pemodelan pada *fold* ke-5



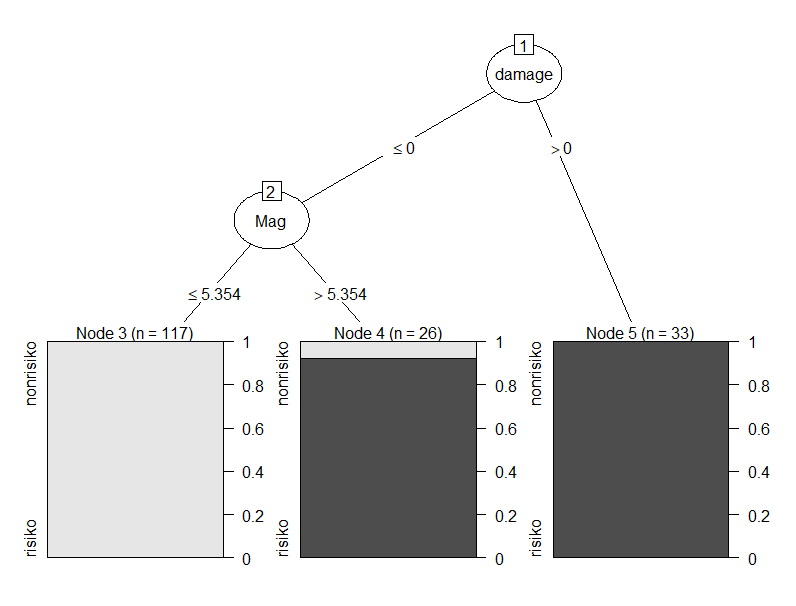
Lampiran 14 Pohon keputusan hasil pemodelan pada *fold* ke-6



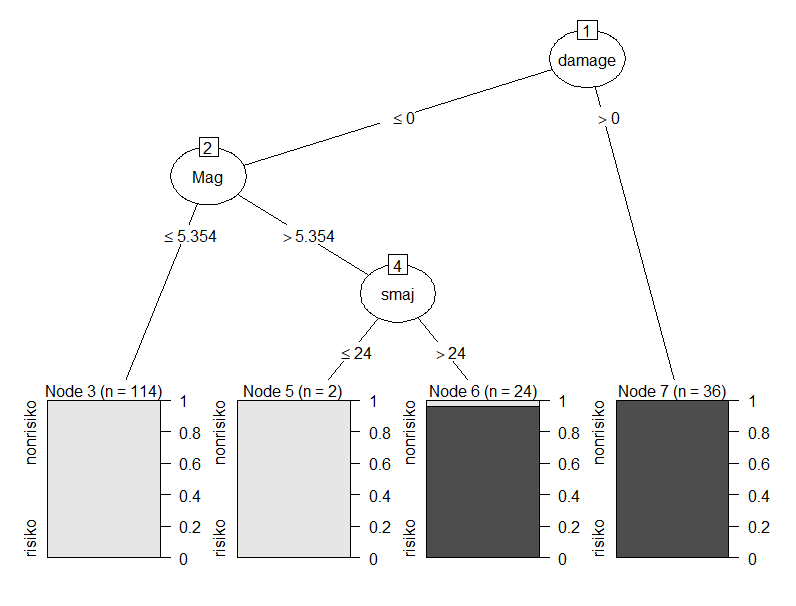
Lampiran 15 Pohon keputusan hasil pemodelan pada *fold* ke-7



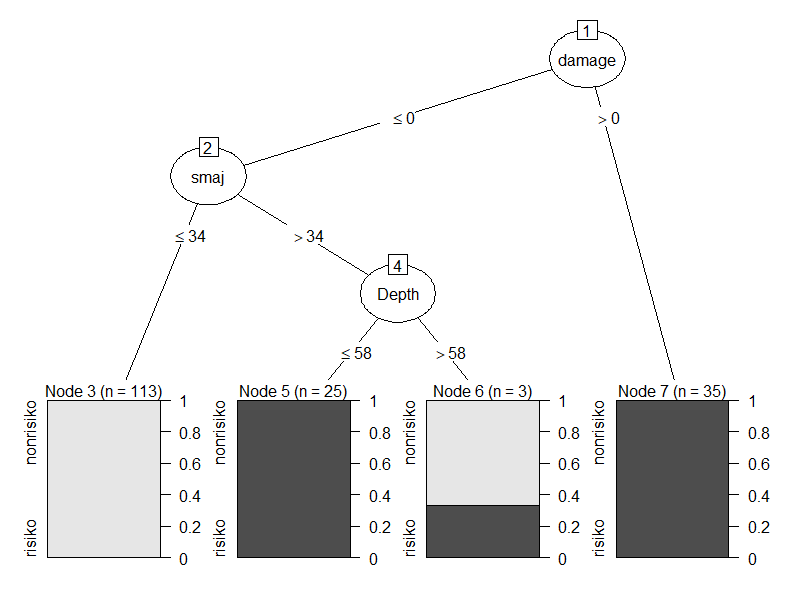
Lampiran 16 Pohon keputusan hasil pemodelan pada *fold* ke-8



Lampiran 17 Pohon keputusan hasil pemodelan pada *fold* ke-9



Lampiran 18 Pohon keputusan hasil pemodelan pada *fold* ke-10



# RIWAYAT HIDUP

Penulis dilahirkan di Jakarta pada tanggal 4 Agustus 1994. Penulis merupakan anak kelima dari tujuh bersaudara dari pasangan Indrawan Widjasena dan Sumarti.

Penulis menyelesaikan pendidikan di jenjang Sekolah Menengah Atas (SMA) pada tahun 2012 di SMA Mardi Yuana Kota Bogor. Penulis menyelesaikan jenjang pendidikan Diploma pada tahun 2015 di Program Diploma Institut Pertanian Bogor dengan program keahlian Manajemen Informatika. Penulis diterima sebagai mahasiswa Program Sarjana Alih Jenis (S1) Departemen Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam pada tahun 2015.