ANALISIS PERBANDINGAN KINERJA METODE KLASIFIKASI DALAM DATA MINING

Ari Wibowo
Jurusan Teknik Informatika Polteknik Negeri Batam
wibowo@polibatam.ac.id

Abstrak

Salah satu masalah yang menghambat perlindungan seseorang dari musibah sakit atau meninggal adalah terhentinya manfaat asuransi seseorang akibat lalai atau sengaja berhenti dari suatu program asuransi. Untuk mengetahui sebaran dan karakteristik nasabah yang putus di tengah jalan perlu dilakukan pengelompokan/klasifikasi sesuai dengan karakteristiknya. Model klasifikasi dibangun berdasarkan atribut yang sudah ada dan status polis nasabah yang sudah bergabung sebelumnya. Tujuan dari penelitian ini adalah melakukan prediksi terhadap calon nasabah perusahaan asuransi X dengan menggunakan metode klasifikasi CART dan menerapkan *bagging* untuk memperbaiki performansi hasil prediksi.

Metode lain yang diteliti adalah metode Random Forest dan Boosting. Adapun atribut yang dipakai adalah jenis kelamin, phone, kelas pekerjaan, status kawin, income, dan metode pembayaran. Metode dan atribut tersebut digunakan untuk memprediksi kelas status polis data nasabah asuransi. Berdasarkan hasil analisis pengujian didapatkan bahwa metode yang memberikan tingkat akurasi prediksi paling baik adalah metode *Bagging* CART. Dimana metode tersebut bisa melakukan prediksi dengan tingkat kebenaran/akurasi mencapai 90%, sementara metode yang lain hanya memiliki tingkat akurasi kurang dari 85%.

Kata kunci : klasifikasi, prediksi, *bagging*, akurasi.

Abstract

One of the problems that hamper a person's protection from calamities sick or dying person is the discontinuation of insurance benefits due to negligent or deliberately stopped from an insurance program. To determine the distribution and characteristics of clients who drop out in the middle of the road needs to be done grouping / classification according to their characteristics. Classification model is built based on the attributes of an existing policy and status of clients who have joined before. The purpose of this study is to make predictions of prospective insurance company X using the CART classification method and applying bagging to improve the performance prediction results.

Another method under study is a method of Random Forest and Boosting. The attributes used are gender, phone, job class, marital status, income, and payment methods. Methods and attributes are used to predict the class status of the insurance policy of customer data. Based on the results of test analysis found that the method gives the best prediction accuracy rate is Bagging CART method. Where such methods can make predictions with a level of truth / accuracy reached 90%, while other methods only have an accuracy rate of less than 85%.

Key words: classification, prediction, bagging, accuracy.

1. Pendahuluan

1.1 Latar Belakang

Kemajuan teknologi informasi telah menyebabkan banyak orang dapat memperoleh data dengan mudah bahkan cenderung berlebihan. Data tersebut semakin lama semakin banyak dan terakumulasi, akibatnya pemanfaatan data yang terakumulasi tersebut menjadi tidak optimal. Sebagai contoh perusahaan *retail* yang akan memberikan brosur penawaran barang-barang yang dijual ke pelanggan sesuai basis data pelanggan yang mereka punya. Jika perusahaan *retail* tersebut mempunyai satu juta data pelanggan dan masing-masing pelanggan tersebut dikirimkan sebuah brosur penawaran dimana biaya pengiriman brosur tersebut adalah dua ribu rupiah, maka biaya yang akan dikeluarkan oleh perusahaan tersebut adalah dua juta rupiah per bulan. Dari penggunaan dana tersebut mungkin hanya sepertiganya atau bahkan 8% saja yang secara efektif membeli penawaran tersebut (YUD 2003).

Maka dari itu perlu prediksi yang efektif terhadap calon pembeli supaya tujuan bisa tercapai. Disamping prediksi untuk pembelian suatu produk, ada juga perusahaan yang membutuhkan prediksi untuk kelangsungan dari produk yang dibeli, sebagai contoh bagaimana perusahaan asuransi menjaga agar status polis pada nasabah tidak *lapse*, sehingga nasabah yang bersangkutan mendapatkan manfaat yang maksimal dari produk yang dibeli.

Berdasarkan uraian di atas diperlukan analisis nasabah yang potensial membeli produk tertentu dan melakukan pengiriman brosur sesuai dengan potensi pembelian dari pelanggan. *Data mining* adalah salah satu solusi untuk permasalahan di atas. *Data mining* merupakan serangkaian proses untuk menggali suatu informasi terpendam dari suatu kumpulan data, yaitu berupa pengetahuan yang selama ini tidak diketahui secara manual. *Data mining* akan membentuk suatu pengetahuan dalam kelompok tertentu yang memiliki karakteristik masing-masing. Proses pembentukan pengetahuan ini biasa disebut dengan teknik *data mining.*

Terdapat beberapa teknik *data mining* yang telah dikembangkan, diantaranya klasifikasi, *clustering, association rule, neural network, decision tree,* dan lain-lain. Tapi bagaimana memilih teknik *data mining* yang tepat sehingga dihasilkan klasifikasi dan prediksi yang akurat? Karena dengan pemilihan metode yang tepat akan menghasilkan akurasi yang lebih baik, sehingga berguna dalam pengembangan, memperbaiki proses bisnis dan strategi dalam suatu perusahaan yang memanfaatkan teknologi *data mining*.

1.2 Tinjauan Pustaka

Metode CART

CART (*Classification and Regression Trees*) adalah salah satu metode atau algoritma dari salah satu teknik eksplorasi data *decision tree*. Metode ini dikembangkan oleh Leo Breiman, Jerome H. Friedman, Richard A. Olshen dan Charles J. Stone sekitar tahun 1980-an. CART merupakan metodologi statistik non-parametrik yang dikembangkan untuk topik analisis klasifikasi, baik untuk variabel respon kategorik maupun kontinu. CART menghasilkan suatu pohon klasifikasi jika variabel responnya kategorik, dan menghasilkan pohon regresi jika variabel responnya kontinu.

Langkah-langkah penerapan metode CART

1. Pembentukan pohon klasifikasi

Proses pembentukan pohon klasifikasi terdiri atas 3 tahapan yaitu:

a. Pemilihan Pemilah (*Classifier)*

Untuk membentuk pohon klasifikasi digunakan sampel data *Learning (L)* yang masih bersifat heterogen. Sampel tersebut akan dipilah berdasarkan aturan pemilahan. Pemilihan pemilah tergantung pada jenis *tree* atau lebih tepatnya tergantung pada jenis variabel responnya. Untuk mengukur tingkat keheterogenan suatu kelas dari suatu simpul tertentu dalam pohon klasifikasi dikenal dengan istilah *impurity measure i(t)*. Ukuran ini membantu menemukan fungsi pemilah yang optimal. Kualitas ukuran dari seberapa baik pemilah *s* dalam menyaring data menurut kelas merupakan ukuran penurunan keheterogenan dari suatu kelas.

$$\Delta i(s,t) = i(t) - p_L\, i(t_L) - p_R\, i(t_R) \quad \text{.....(1)}$$

Pemilah yang menghasilkan nilai $\Delta i(s,t)$ lebih tinggi merupakan pemilah yang lebih baik karena hal ini memungkinkan untuk mereduksi keheterogenan secara lebih signifikan. Karena t_R *u* $t_L \leq t$ maka nilai $\Delta i(s, t)$ merepresentasikan perubahan dari keheterogenan dalam simpul *t* yang semata-mata disebabkan oleh pemilah *s.* Jika simpul yang diperoleh merupakan kelas yang tidak homogen, prosedur yang sama diulangi sampai pohon klasifikasi menjadi suatu konfigurasi tertentu, dan memenuhi

$$\Delta i(s^*, t_1) = \max_{s \subset S} \Delta i(s, t_1) \quad \text{.....(2)}$$

b. Penentuan Simpul Terminal

Suatu simpul *t* akan menjadi simpul terminal atau tidak akan dipilah kembali apabila pada simpul *t* tidak terdapat penurunan keheterogenan secara berarti atau adanya batasan minimum *n* seperti hanya terdapat satu pengamatan pada tiap simpul anak. Menurut Breiman (1984), umumnya jumlah kasus minimum dalam suatu terminal akhir adalah

5, dan apabila hal itu terpenuhi maka pengembangan *tree* dihentikan. Sementara itu, menurut Steinberg dan Colla (1995) jumlah kasus yang terdapat dalam simpul terminal yang homogen adalah kurang dari 10 kasus.

c. Penandaan Label Kelas
Penandaan label kelas pada simpul terminal dilakukan berdasarkan aturan jumlah terbanyak.

2. Pemangkasan pohon klasifikasi
Pemangkasan dilakukan dengan jalan memangkas bagian *tree* yang kurang penting sehingga didapatkan pohon optimal. Ukuran pemangkasan yang digunakan untuk memperoleh ukuran *tree* yang layak adalah *cost complexity minimum* (Breiman *et. Al*., 1984). Sebagai ilustrasi, untuk sembarang *tree T* yang merupakan sub *tree* dari *tree* terbesar Tmax ukuran *cost complexity* yaitu.

$$R_\alpha(T) = R(T) + \alpha\left|\tilde{T}\right| \quad \text{.....(3)}$$

dimana
R(*T*) = *Resubtitusion Estimate* (Proporsi kesalahan pada sub *tree*)
α = kompleksitas parameter (*complexity parameter)*
~|T|= ukuran banyaknya simpul terminal *tree T*

*R*α(*T*) merupakan kombinasi linear biaya dan kompleksitas *tree* yang dibentuk dengan menambahkan *cost penalty* bagi kompleksitas terhadap biaya kesalahan klasifikasi *tree*. *Cost complexity prunning* menentukan suatu pohon bagian *T*(α) yang meminimumkan *R*α(*T*) pada seluruh pohon bagian. Atau untuk setiap nilai α, dicari pohon bagian max *T* yang meminimumkan *R*α(*T*) yaitu.

$$R_\alpha(T(\alpha)) = \min_{T<T_{\max}} R_\alpha(T) \quad \text{.....(4)}$$

Jika *R*(*T*) digunakan sebagai kriteria penentuan *tree* optimal maka akan cenderung *tree* terbesar adalah T1, sebab semakin besar *tree*, maka semakin kecil nilai *R*(*T*) nya.

3. Penentuan Pohon Klasifikasi Optimal
Pohon klasifikasi yang berukuran besar akan memberikan nilai penduga pengganti paling kecil, sehingga *tree* ini cenderung dipilih untuk menduga nilai respon. Tetapi ukuran *tree* yang besar akan menyebabkan nilai kompleksitas yang tinggi karena struktur data yang digambarkan cenderung kompleks, sehingga perlu dipilih pohon optimal yang berukuran sederhana tetapi memberikan nilai penduga pengganti cukup kecil.

Metode Bagging CART

Metode Bagging merupakan penyempurnaan metode CART yaitu menggabungkan banyak nilai dugaan menjadi satu nilai dugaan. Dengan demikian proses pembuatan dugaan secara *bagging* menggunakan *tree* adalah sebagai berikut:

1. Pembuatan *tree*
   a. tahapan *bootstrap*-tarik pengamatan acak berukuran n dari gugus data training
   b. susun *tree* terbaik berdasarkan data tersebut
   c. ulangi langkah a-b sebanyak *k* kali sehingga diperoleh *k* buah *tree* acak
2. Lakukan pendugaan gabungan berdasarkan *k* buah *tree* tersebut (misal menggunakan *majority vote* untuk kasus klasifikasi, atau rata-rata untuk kasus regresi)

Penggunaan *bagging* ini sangat membantu terutama mengatasi sifat ketidakstabilan *tree* klasifikasi dan regresi tunggal seperti yang telah disinggung sebelumnya. Hastie et al. (2008) menyatakan bahwa proses *bagging* dapat mengurangi galat baku dugaan yang dihasilkan oleh *tree* tunggal. Hal ini dapat jelas terlihat karena dengan melakukan rata-rata misalnya maka ragam dugaan akan mengecil sedangkan tingkat bias dugaan tidak terpengaruh. Selain itu Breiman (1996) mencatat bahwa pada banyak gugus data yang dicoba, *bagging* mampu mengurangi tingkat kesalahan klasifikasi pada kasus klasifikasi. Hal ini tentu tidak berlaku secara keseluruhan. Berk (2008) mencatat beberapa kasus yang mungkin menyebabkan dugaan *bagging* memiliki ragam dugaan yang lebih besar atau juga bias yang lebih besar pula. Ini terjadi antara lain pada kasus dengan kategori peubah respon yang sangat tidak seimbang.

Metode Random Forest

Metode *Random Forest* berupaya untuk memperbaiki proses pendugaan yang dilakukan menggunakan metode *bagging*. Perbedaan utama dari kedua metode ini terletak pada penambahan tahapan *random sub-setting* sebelum di setiap kali pembentukan *tree*. Tahapan penyusunan dan pendugaan menggunakan RF adalah:

1. Tahap I
   a. Tahapan *bootstrap* : tarik contoh acak dengan permulihan berukuran n dari gugus data training
   b.Tahapan *random sub-setting* : susun *tree* berdasarkan data tersebut, namun pada setiap proses pemisahan pilih secara acak $m < d$ peubah penjelas, dan lakukan pemisahan terbaik.
   c. Ulangi langkah a-b sebanyak *k* kali sehingga diperoleh *k* buah *tree* acak

2.Lakukan pendugaan gabungan berdasarkan *k* buah *tree* tersebut (misal menggunakan *majority vote* untuk kasus klasifikasi, atau rata- rata untuk kasus regresi)

Proses penggabungan nilai dugaan dari banyak *tree* yang dihasilkan serupa dengan yang dilakukan pada metode *bagging*. Perhatikan bahwa pada setiap kali pembentukan *tree*, kandidat peubah penjelas yang digunakan untuk melakukan pemisahan bukanlah seluruh peubah yang terlibat namun hanya sebagian saja hasil pemilihan secara acak. Bisa dibayangkan bahwa proses ini menghasilkan kumpulan *tree* tunggal dengan ukuran dan bentuk yang berbeda-beda. Hasil yang diharapkan adalah kumpulan *tree* tunggal memiliki korelasi yang kecil antar *tree*-nya. Korelasi kecil ini mengakibatkan ragam dugaan hasil RF menjadi kecil (Hastie *et al*, 2008) dan lebih kecil dibandingkan ragam dugaan hasil *bagging* (Zu, 2008). Lebih jauh Zu (2008) menjelaskan bahwa dalam Breiman (2001) telah dibuktikan batasan besarnya kesalahan prediksi oleh *Random Forest* adalah

$$\mathcal{E} \leq \frac{r(1-s^2)}{s^2} \quad \text{.....(5)}$$

dengan r adalah rata-rata korelasi antar pasangan dugaan dari dua *tree* tunggal dan *s* adalah adalah rata-rata ukuran kekuatan (*strength*) akurasi *tree* tunggal. Nilai s yang semakin besar menunjukkan bahwa akurasi prediksinya semakin baik. Definisi formal mengenai *s* dapat dilihat di Breiman (2001). Pertidaksamaan tersebut mengarahkan bahwa jika ingin memiliki RF yang memuaskan maka haruslah diperoleh banyak *tree* tunggal dengan r yang kecil dan *s* yang besar.

2. Analisis Metode

Pada bagian ini diterangkan penjelasan pembentukan pohon klasifikasi dengan metode CART dan *Bagging* CART. Misal disediakan data sampel seperti ditunjukkan pada Tabel III-5. Pada data sampel ini menggunakan tiga atribut *predictor* dan satu atribut *target*, yang menjadi atribut target adalah atribut *status*.

Tabel 1 - Data Nasabah Pembentuk Pohon

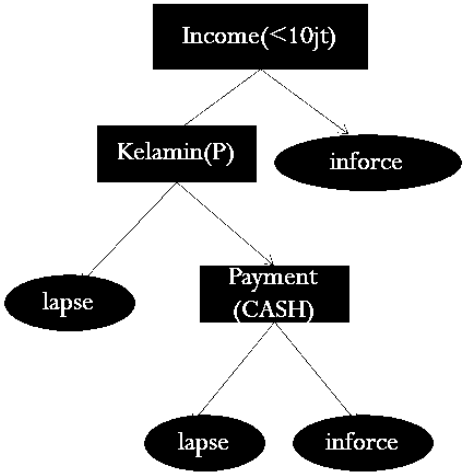
Nama	Payment	Income	Kelamin	Status
A	CASH	<10jt	L	Lapse
B	CASH	>50jt	L	Inforce
C	CASH	10-50jt	P	Inforce
D	DEBET	<10jt	L	Inforce
E	DEBET	<10jt	L	Lapse
F	CASH	>50jt	L	Inforce
G	DEBET	<10jt	P	Lapse
H	DEBET	10-50jt	L	Lapse

Metode CART

Untuk membangun suatu pohon dengan metode CART, pertama-tama yang harus dihitung adalah nilai indeks *gini* untuk setiap atribut. Berdasarkan perhitungan rumus indeks *gini* didapatkan hasil perhitungan nilai indeks *gini*. Setelah indeks *gini* diketahui baru kemudian disusun pohon seperti pada Gambar 1.

Hasil penghitungan indeks gini didapatkan hasil sebagai berikut:

Indeks Gini (payment) = 0.19
Indeks Gini (income = 0.59
Indeks Gini (kelamin) = 0.06



Gambar 1 - Pembentukan Pohon Menggunakan Metode CART

Dengan menggunakan aturan yang dibentuk dari pohon pada Gambar 1, maka hasil prediksi dengan metode CART ditunjukkan pada Tabel 2.

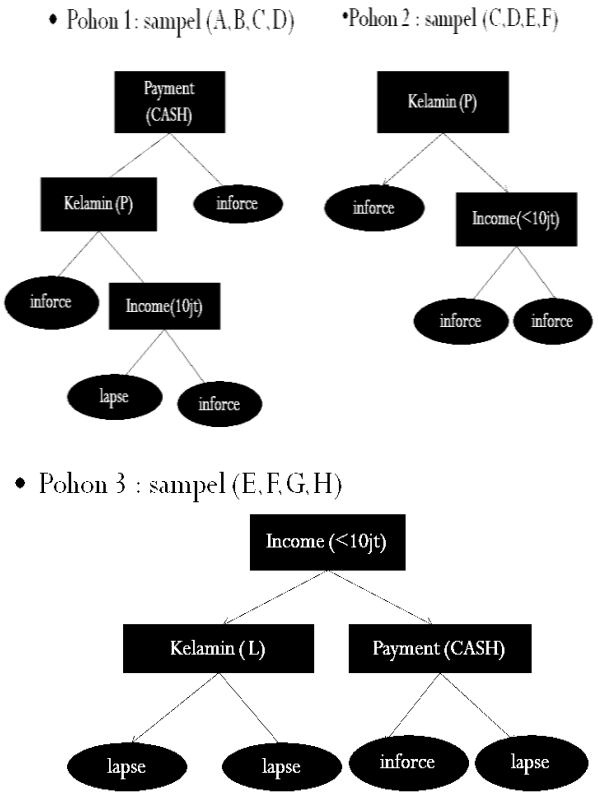
Tabel 2- Hasil Prediksi Menggunakan Metode CART

Nama	Payment	Income	Kelamin	Status
A	CASH	<10jt	L	Lapse
B	CASH	>50jt	L	Inforce
C	CASH	10-50jt	P	Inforce
D	DEBET	<10jt	L	Inforce
E *(error)*	DEBET	<10jt	L	Inforce
F	CASH	>50jt	L	Inforce
G	DEBET	<10jt	P	Lapse
H *(error)*	DEBET	10-50jt	L	Inforce

Akurasi=6/8*100%=75%

Bagging CART

Pembentukan pohon dengan metode *Bagging* CART ditunjukkan pada Gambar 2. Pada contoh yang dibuat ini misalkan jumlah *bootstrap* yang diinginkan sebanyak tiga, jadi ada 3 pohon yang terbentuk.



Gambar 2 - Pembentukan Pohon Menggunakan Bagging CART

Dengan menggunakan aturan yang dibentuk dari pohon pada Gambar 2 ditambah dengan aturan *majority voting*, maka hasil prediksi dengan metode *Bagging* CART ditunjukkan pada Tabel 3.

Tabel 3 - Hasil Prediksi Menggunakan Metode Bagging CART

Nama	Payment	Income	Kelamin	Status
A	CASH	<10jt	L	Lapse
B	CASH	>50jt	L	Inforce
C	CASH	10-50jt	W	Inforce
D	DEBET	<10jt	L	Inforce
E *(error)*	DEBET	<10jt	L	Inforce
F	CASH	>50jt	L	Inforce
G	DEBET	<10jt	W	Lapse
H	DEBET	10-50jt	L	Lapse

Akurasi=7/8*100%=87.5%

3. Pembahasan

3.1 Data dan Metode

Pada model ini ada dua belas atribut yang dipakai dimana sebelas diantaranya sebagai *predictor* dan satu atribut sebagai *target*. Pada atribut yang menjadi *target* ada tiga kelas yang menjadi tujuan/*respon*/*target* dari hasil klasifikasi yang terdapat pada atribut *status*, yaitu *inforce*, *lapse*, dan *surrender*. Atribut-atribut yang menyertai data calon nasabah dapat dilihat pada Tabel 1.

Tabel 4 - Karakteristik Data Nasabah

No	Nama Atribut	Jenis	Value
1	*kelamin*	Kategorik	P, L
2	*phone*	Kategorik	*Yes*, *No*
3	*kelas pekerjaan*	Kategorik	1,2,3,4
4	*status kawin*	Kategorik	M, S
5	*income*	Kategorik	
6	*rawat inap*	Kategorik	pernah, tdk pernah
7	*payment*	Kategorik	*card*, *autodebet*, *cash*
8	*p.mode*	Kategorik	bulanan, kwartalan, semesteran, tahunan
9	*merokok*	Kategorik	Y, N
10	*tahun lahir*	Numeric	
11	*kode agen*	Kategorik	
12	*status*	Kategorik	*inforce*, *lapse*, *surrender*

Tabel nasabah berisi data nasabah antara tahun 2005-2010, dengan jumlah data 1287 record. Ada beberapa proses *preprocesing* agar data siap diolah oleh model, yaitu *data cleaning* dan *data*

transformasi. Untuk *data cleaning* dilakukan secara manual, yaitu adanya pembersihan tanda spasi yang tidak perlu pada *row* status. Pada proses *data transformasi* dilakukan secara otomatis dan *manual*.

Transformasi secara *manual* dilakukan pada saat perubahan tipe data pada atribut *tahun lahir*, pengurangan jumlah atribut, penambahan tanda *underscore*(_), dan pengisian data yang kosong (*missing value*), sedangkan proses transformasi secara otomatis dilakukan saat pembentukan file data nasabah menjadi file dengan format **.csv* dan **.arff* agar siap diolah oleh sistem.

Untuk menghitung akurasi, data asli dipartisi menjadi dua bagian yaitu *data training* dan *data testing*. Model klasifikasi kemudian dibangun berdasarkan *data training*, kemudian hasilnya dievaluasi dengan menggunakan *data testing*. Akurasi dari masing-masing metode klasifikasi dapat diestimasi berdasarkan akurasi yang diperoleh dari *data testing*. Akurasi dapat dihitung berdasarkan persentase *error* yang terjadi.
*Error=(prediksi salah/total prediksi)*x100% …..(6)

Akurasi dihitung berdasarkan rumus:
Akurasi=100%-*error* ….....(7)

Proporsi antara *data training* dan *data testing* tidak mengikat, tetapi agar variansi dalam model tidak terlalu besar maka dapat ditentukan bahwa proporsi *data training* lebih besar daripada *data testing*. Penentuan data yang masuk ke dalam *data training* dan *data testing* diusahakan dari kelompok yang berbeda sehingga diharapkan data yang masuk adalah data yang saling bebas.

3.2 Hasil dan Pembahasan

Pada sub bab ini dipaparkan pengujian untuk semua atribut yang dimasukkan sekaligus pada semua metode klasifikasi. Semua atribut diinput ke dalam model, kemudian dihitung tingkat akurasinya, selanjutnya dilakukan analisa terhadap hasil pengujian. Ada 11 atribut yang diuji yaitu atribut *income, kelamin, kelas pekerjaan, kode agen, merokok, p_mode, payment, phone, rawat inap, status kawin, dan tahun lahir*. Perhitungan akurasi dihitung berdasarkan rata-rata dari masing-masing kelas dan ketepatan prediksi untuk semua kelas.

Tujuan Pengujian

Pengujian ini bertujuan untuk mengetahui metode apa yang memberikan tingkat akurasi yang paling tinggi. Semua atribut dijadikan input pada masing-masing metode kemudian dihitung tingkat akurasinya. Berdasarkan hasil akurasi yang didapat akan ditentukan metode apa yang memberikan tingkat akurasi paling baik. Hasil akurasi yang didapat akan digunakan untuk proses analisa selanjutnya.

Hasil Pengujian

Setelah dilakukan pengujian didapatkan hasil pengujian untuk semua atribut pada setiap metode yang diuji coba, hasil untuk masing-masing metode dapat dilihat pada Tabel 5 sampai Tabel 7 sebagai berikut.

Tabel 5 - Hasil Pengujian Metode CART

Actual Class	Total Class	Percent Correct	INFORCE N=79	LAPSE N=29	SURRENDERN=22
INFORCE	89	86,52	77	6	6
LAPSE	34	67,65	2	23	9
SURRENDER	7	100,00	0	0	7
Total:	130,00				
Average:		84,72			
Overall % Correct:		82,31			

Tabel 6 Hasil Pengujian Metode Bagging CART

Actual Class	Total Class	Percent Correct	INFORCE N=85	LAPSE N=32	SURRENDER N=13
INFORCE	89	93,26	83	1	5
LAPSE	34	85,29	2	29	3
SURRENDER	7	71,43	0	2	5
Total:	130,00				
Average:		75,58			
Overall % Correct:		90,00			

Tabel 7 - Hasil Prediksi

Kelas	CART (%)	Bagging CART (%)	Random Forest (%)
Inforce	86.52	93.26	78.65
Lapse	67.65	85.29	47.06
Surrender	100	71.43	100
Rata-rata	82.31	90.00	71.53

Analisa Hasil Pengujian

Berdasarkan data yang didapat dari hasil pengujian maka dilakukan analisis seperti dibawah ini.

- Berdasarkan tingkat akurasi yang dihasilkan metode *Bagging* CART memberikan tingkat akurasi yang paling baik, sedangkan metode *Random Forest* memberikan tingkat akurasi yang paling rendah.

- Pohon klasifikasi yang dihasilkan oleh algoritma *Bagging* CART merupakan pohon klasifikasi yang sangat kompleks karena *tree* ini dibentuk oleh semua variabel *predictor*. Proses pengklasifikasian data baru dengan pohon klasifikasi *Bagging* CART dijalankan secara paralel pada semua pohon klasifikasi tersebut sehingga akan diperoleh berbagai versi hasil prediksi, dimana hasil prediksi akhir

dari pohon klasifikasi ini merupakan hasil *voting* dari berbagai versi prediksi kelas yang paling banyak muncul.

- Pada *Random Forest* setiap kali pembentukan *tree*, kandidat *predictor* yang digunakan untuk melakukan pemisahan bukanlah seluruh peubah yang terlibat namun hanya sebagian saja hasil pemilihan secara acak. Bisa dibayangkan bahwa proses ini menghasilkan kumpulan *tree* tunggal dengan ukuran dan bentuk yang berbeda-beda. Sehingga ada kemungkinan menghasilkan akurasi lebih rendah bila dibandingkan dengan metode CART maupun *Bagging* CART.

- Pada metode CART, *error* yang terjadi karena adanya *noise* dan pada metode ini tidak ada penanganan masalah ini karena pohon yang dibangun hanya satu saja. *Noise* ini terjadi karena adanya anomali pada data tertentu dimana seharusnya suatu kelas diprediksi *inforce* tetapi diprediksi *lapse*. Penentuan prediksinya hanya berdasarkan nilai probabilitas terbesar pada *leaf node* yang bersangkutan.

- Pada metode *Bagging* CART ada penanganan *noise* jika terjadi anomali, penanganan prediksinya ditentukan dua hal, yaitu berdasarkan perhitungan nilai probabilitas dari suatu kelas pada *leaf node* dan hasil *voting* dari keseluruhan pohon yang terbentuk. Disamping itu juga kalau hasil *voting* ternyata sama, maka ditentukan bahwa untuk *record* yang bersangkutan dimasukkan ke dalam kelas *inforce*, karena sesuai prioritas berdasarkan jumlah kelas dalam suatu sampel dimana kelas *inforce* memiliki jumlah lebih banyak.

Pengujian 2 kelas

- Metode yang memberikan tingkat akurasi paling tinggi ada pada metode *Bagging* CART. Hal ini terjadi karena pada metode *Bagging* CART pemilahan data pada saat pembentukan *tree* dilakukan dengan lebih natural dan tidak dipaksakan. Sementara pada metode *Boosting* dan *Random Forest* proses pemilahan datanya agak sedikit dipaksakan.

- Setelah dilakukan analisis hasil *data testing* antara metode CART dan *Bagging* CART terdapat 59 perbedaan pada *data testing*, pada metode *Bagging* CART ada 20 *error*, sementara pada metode CART ada 39 *error*. Kemudian perbandingan antara *Bagging* CART dan *Random Forest* ada 67 perbedaan, dimana pada metode *Bagging* CART ada 27 *error*, sementara pada *Random Forest* ada 43 error. Terakhir perbandingan antara *Bagging* CART dengan *Boosting*, di sini ditemukan 81 perbedaaan, dimana pada metode *Bagging* CART ada 31 *error*, sementara pada *Boosting* 52 *error*. Setelah dilakukan pengamatan lebih lanjut dari hasil keempat metode di atas ditemukan ada 13 *error* pada metode CART, *Random Forest*, dan *Boosting* yang diprediksi benar oleh metode *Bagging* CART.

Tabel 8 - Perbandingan Prediksi Salah - Benar

Case ID	CART	Random Forest	Boosting	Bagging CART
12	X	X	X	V
113	X	X	X	V
183	X	X	X	V
374	X	X	X	V
517	X	X	X	V
621	X	X	X	V
747	X	X	X	V
757	X	X	X	V
880	X	X	X	V
1125	X	X	X	V
1127	X	X	X	V
1254	X	X	X	V
1285	X	X	X	V

Keterangan
X : diprediksi salah
V : diprediksi benar

4. Kesimpulan

Setelah menyelesaikan penelitian ini, maka dapat ditarik beberapa kesimpulan seperti di bawah ini.

1. Berdasarkan tingkat akurasi yang dihasilkan metode *Bagging* CART memberikan tingkat akurasi yang paling baik bila dibandingkan dengan metode CART, dan *Random Forest*, sedangkan metode *Random Forest* memberikan tingkat akurasi yang paling rendah. Hal ini disebabkan karena pada metode *Bagging* CART ada pembangkitan *learning* sampel yang akan mereduksi variansi atribut *predictor*, sehingga ketika dikombinasikan hasilnya lebih baik bila dibandingkan dengan *predictor* tunggal yang dibangun untuk menyelesaikan masalah yang sama.

2. Berdasarkan hasil pengujian bisa disimpulkan bahwa calon nasabah potensial ditentukan oleh atribut *phone, payment*, dan *status kawin*.

5. Daftar Pustaka

1. Han, Jiawei., Kamber, Micheline (2000), *Data Mining Concepts and Technigues*, Morgan Kaufman Publishers
2. Piatetsky, Gregory (2006), *Data Mining and Knowledge Discovery in Business Databases*.

3. Breiman, L., Friedman, J., Olsen, R.A., dan Stone, C. (1984), *Classification and regression trees*, Wadsworth, Belmont, California.
4. Breiman, L (1996a). *Bagging Predictors*, *Machine Learning*, Vol. 24. 123-140
5. Bühlman, P. dan Yu, B. (2002), *Analyzing Bagging*, *The Annals of Statistics*, Vol. 30 no. 4, hal 927-961.
6. Breiman, L (1996a). *Bagging Predictors*, *Machine Learning*, Vol. 24. 123-140
7. Breiman, L. (1996b): *Heuristics of instability and stabilization in model selection*, *Annals of Statistics*, 24, hal. 2350–2383.
8. Bühlman, P. dan Yu, B. (2002), *Analyzing Bagging*, *The Annals of Statistics*, Vol. 30 no. 4, hal 927-961.
9. Clarke, R.T. dan Bintercourt, H. R (2003), *"Use of Classification And Regression Trees (CART) to Classify Remotely_Sensed Digital Images",* Research Report , Centro stadual de Pesquisas em Sensoriamento Remoto Universidade Federal do Rio Grande do Sul – UFRGS , Porto Alegre, Brazil
10. Efron, B. dan Tibshirani, R.J. (1993) "*An Introduction to the Bootstrap"* Chapman Hall, New York.
11. Sibaroni, Yuliant (2008) "*Analisis dan Penerapan Metode Klasifikasi untuk Pembangunan Perangkat Lunak Penerimaan Mahasiswa Baru Jalur Non-Tulis"* ITB, Bandung.
12. Wijanarko Bambang dan Sumarmi (2009) "Bagging CART pada Klasifikasi Anak Putus Sekolah" ITS, Surabaya.
13. Sri, Veronika (2007) "*Pengembangan Skalabilitas Algoritma Klasifikasi C4.5 dengan Pendekatan Konsep Operator Relasi*" ITB, Bandung.