

Model Deteksi Penyimpangan Keuangan Medis Menggunakan *Gradient Boosted Tree* (GBT) Pada Rumah Sakit ABC

Aris Martono^{*1}, Padeli²

¹Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Raharja

²Program Studi Pendidikan Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Raharja

Email: ^{*1}aris.martono@raharja.info, ²padeli@raharja.info

Abstrak

Tujuan penelitian ini yaitu untuk mengetahui penyimpangan keuangan yang terjadi di lingkungan Rumah Sakit. Penyimpangan transaksi keuangan ini melibatkan aktivitas dokter, pembuatan resep dan apotik atau farmasi serta bagian keuangan Rumah Sakit. Setiap dokter yang mengeluarkan resep untuk pengobatan pasien, diharapkan pasien membeli obat di apotik Rumah Sakit itu sendiri sehingga transaksi keuangannya menjadi pemasukan bagi Rumah Sakit. Namun sebaliknya, hal ini bisa mempersulit mengetahui pemasukan kas yang diperoleh dari setiap dokter terkait resep yang dikeluarkan. Oleh karenanya penelitian ini dilakukan dengan membuat model untuk mengetahui penyimpangannya. Untuk mendapatkan model yang terbaik dilakukan evaluasi model terhadap algoritma *Gradient Boosted Tree*(GBT) dan *Random Forest*(RF). Hasilnya adalah AUC (*Area Under the Curve*) model GBT = 0.976 dan AUC model RF = 0.964 yang menunjukkan bahwa algoritma GBT pilihan terbaik untuk pemrosesan penyimpangan transaksi keuangan dataset medis di Rumah Sakit ABC.

Kata-kunci--Penyimpangan keuangan, *Gradient Boosted Tree*, *Random Forest*

Abstract

The aim of this research is to determine financial irregularities that occur in the hospital environment. This deviation in financial transactions involves the activities of doctors, prescriptions, pharmacies and hospital finance departments. Every doctor who issues a prescription for a patient's treatment is expected to have the patient buy the medicine at the hospital pharmacy itself so that the financial transaction becomes income for the hospital. However, on the contrary, this can make it difficult to know the cash income obtained from each doctor regarding the prescriptions issued. Therefore, this research was carried out by creating a model to determine the deviation. To get the best model, the model was evaluated using the Gradient Boosted Tree (GBT) and Random Forest (RF) algorithms. The results are AUC of the GBT model = 0.976 and AUC of the RF model = 0.964 which indicates that the GBT algorithm is the best choice for processing deviations in financial transactions in medical datasets at ABC Hospital.

Keyword-- *Financial irregularities, Gradient Boosted Tree, Random Forest*

I. PENDAHULUAN

Latar Belakang

Rumah sakit ABC dalam pengelolaan transaksi keuangan melibatkan aktivitas dokter, pembuatan resep dan apotik atau farmasi serta bagian keuangan relative cukup rumit untuk mengatasinya. Dalam pelayanan pasien Rumah Sakit biasanya, pasien mendaftarkan diri ke bagian pelayanan dan mengurus administrasi jenis pelayanan poli tertentu dan biaya konsultasi ke dokter serta jenis tindakan yang dilakukan oleh dokter. Hasil konsultasi dengan dokter biasanya dikeluarkan rekomendasi oleh dokter berupa resep ke bagian apotik atau farmasi dalam Rumah Sakit itu sendiri. Namun demikian kadang pasien mengambil obat berdasarkan rekomendasi dokter berupa resep di luar Rumah Sakit itu sendiri sehingga manajemen Rumah Sakit kesulitan monitoring transaksi keuangan yang melibatkan aktivitas dokter, resep yang direkomendasikan dokter dan pihak apotik atau farmasi serta bagian keuangan. Kesulitan terkait dengan jumlah setiap resep yang dikeluarkan oleh setiap dokter dan transaksi keuangan yang terjadi dimana nilai harganya sangat bervariasi tergantung hasil konsultasi dan Tindakan terhadap kondisi pasien dengan dokter. Oleh karena itu dipilih beberapa model yang mampu mendeteksi penyimpangan keuangan tersebut. Adapun dua model untuk mendeteksinya yaitu model *Gradient Boosted Tree* dan *Random Forest* dengan *number of trees* adalah 20 dan kedalaman levelnya yaitu 5.

II. METODE PENELITIAN

Literatur Review

Gradient boosting adalah teknik pembelajaran mesin yang didasarkan pada peningkatan dalam ruang fungsional, yang targetnya adalah residu semu, bukan residu biasa yang digunakan dalam peningkatan tradisional. Ini memberikan model prediksi dalam bentuk kumpulan model prediksi lemah, yaitu model yang membuat sangat sedikit asumsi tentang data, yang biasanya berupa pohon keputusan sederhana.[1][2] Jika pohon keputusan merupakan pembelajar yang lemah, algoritma yang dihasilkan disebut pohon yang ditingkatkan gradien; biasanya performanya mengungguli hutan acak.[1][2][3] Model pohon dengan peningkatan gradien dibuat secara bertahap seperti pada metode peningkatan lainnya, namun model ini menggeneralisasi metode lain dengan memungkinkan optimalisasi fungsi kerugian yang dapat dibedakan secara arbitrer.

Gradient Boosting adalah suatu teknik dalam machine learning yang digunakan untuk membangun model prediktif, terutama dalam konteks regresi dan klasifikasi. Ini adalah metode ensemble, yang berarti menggabungkan prediksi dari beberapa model lemah untuk membentuk model yang lebih kuat.

Proses dasar dari *Gradient Boosting* melibatkan pembuatan serangkaian model prediktif sederhana, yang biasanya disebut sebagai "pohon keputusan lemah" atau "weak learner". Setiap pohon dibangun untuk memperbaiki kesalahan prediksi model sebelumnya dalam rangkaian.

Berikut adalah langkah-langkah umum dalam algoritma *Gradient Boosting*:

1. Inisialisasi Model yaitu model pertama dibuat, dan prediksi awal diperoleh. Prediksi ini mungkin tidak akurat, dan itulah sebabnya kita menggunakan iterasi untuk memperbaikinya.
2. Menghitung Kesalahan Residual yaitu kesalahan residual (perbedaan antara prediksi aktual dan prediksi model saat ini) dihitung.
3. Membangun Model Lebih Baik untuk kesalahan residual ialah sebuah model lemah (biasanya pohon keputusan) dibangun untuk memprediksi kesalahan residual. Model ini ditambahkan ke model sebelumnya dengan bobot tertentu.

4. Penyusunan Bobot yaitu bobot diberikan pada model baru berdasarkan seberapa baiknya model tersebut memperbaiki kesalahan residual. Model yang lebih baik diberi bobot yang lebih tinggi.
5. Iterasi merupakan langkah-langkah 2-4 diulang sejumlah iterasi atau hingga mencapai kondisi berhenti yang ditentukan.
6. Membuat Prediksi Final ialah Prediksi akhir adalah hasil dari penjumlahan prediksi dari semua model yang telah dibangun.

Gradient Boosting memiliki beberapa varian, seperti *Gradient Boosting Machines* (GBM), XGBoost, LightGBM, dan CatBoost, yang memiliki peningkatan efisiensi dan fitur-fitur khusus. Meskipun *Gradient Boosting* bisa menjadi algoritma yang kuat, memerlukan tuning parameter yang hati-hati dan dapat menjadi komputasi yang mahal.

Inisialisasi

Inisialisasi model awal dengan prediksi awal seperti di bawah ini.

$$y_{\text{final}} = \text{argmix}_y \sum_{i=1}^N L(y_i, x)$$

dimana L merupakan fungsi kerugian yang mengukur kesalahan antara prediksi dan target.

Iterasi

Dalam iterasi ini terdapat 5 langkah yang harus dilalui untuk memperoleh pemutakhiran prediksi yang optimal antara-lain:

1. Iterasi $m=1$ hingga M , dimana M yaitu jumlah pohon atau model lemah yang ingin dibangun.
2. Penghitungan residual dengan rumus di bawah ini:

$$r_{im} = y_i - y_{im-1}$$

3. Pembangunan model lemah $h_m(x)$ untuk memprediksi residual r_{im}
4. Penentuan bobot γ_m (learning rate) dan gunakan untuk menghitung bobot prediksi yaitu:

$$y_{im} = y_{im-1} + \gamma_m \cdot h_m(x)$$

5. Pemutakhiran prediksi dengan rumus di bawah ini.

$$y_{im-1} = y_{im}$$

Prediksi Akhir

Prediksi akhir merupakan hasil dari penjumlahan prediksi semua model dengan rumus seperti di bawah ini.

$$y_{\text{final}} = \sum_{m=1}^M \gamma_m \cdot h_m(x)$$

Perlu diketahui bahwa terdapat variasi dalam implementasi *Gradient Boosting* tergantung pada algoritma spesifik yang digunakan seperti XGBoost atau LightGBM dan rumus-rumus di atas merupakan formulasi umum. Variasi ini dapat mencakup fungsi kerugian, kriteria pemisahan pada pohon dan penanganan khusus untuk klasifikasi atau regresi. Selain itu, parameter seperti learning rate dan jumlah iterasi juga perlu ditentukan dan dapat mempengaruhi kinerja model.

Random Forest (RF) adalah algoritma yang menggunakan metode pemisahan biner rekursif untuk mencapai node akhir dalam struktur pohon berdasarkan pada pohon klasifikasi dan regresi [3].

Breiman tahun 2001 memperkenalkan algoritma RF dengan menunjukkan beberapa kelebihan diantaranya mampu menghasilkan error yang relatif rendah, performa yang baik dalam klasifikasi, dapat mengatasi data pelatihan dalam jumlah besar secara efisien, serta metode yang efektif untuk mengestimasi missing data.

RF menghasilkan banyak pohon independen dengan subset yang dipilih secara acak melalui bootstrap dari sampel pelatihan dan dari variabel input di setiap node. RF melakukan klasifikasi dengan cara mengadopsi pendekatan ansambel. Dari berbagai pohon melalui kemunculan mayoritas untuk mencapai keputusan akhir[5].

Set data pelatihan pada algoritma RF diformulasikan sebagai $S = \{(x_i, y_j), i=1, 2, \dots, N; j = 1, 2, \dots, M\}$, dimana x adalah sampel dan y adalah variabel fitur S . N adalah jumlah sampel pelatihan, dan ada variabel fitur M di setiap sampel[4].

Adapun dalam pembangunan algoritma RF terdiri dari 3 langkah yaitu:

(1) Sampling himpunan bagian pelatihan, (2) Pembuatan setiap model pohon keputusan, dan (3) Pengumpulan k pohon ke dalam model RF. Penggunaan algoritma RF untuk klasifikasi dapat diterapkan pada data imbalance dalam jumlah besar dengan memberikan hasil performa yang baik dan waktu eksekusi yang cepat [7] [8]. Suatu algoritma yang menggunakan sejumlah besar pohon keputusan untuk melakukan prediksi (*Random Forest*) dimana setiap pohon dihasilkan secara acak untuk mengatasi overfitting dan meningkatkan keberagaman model.

Berikut ini adalah rumus-rumus dasar terkait dengan *Random Forest* [9][10] antara-lain:

1. Bagging (*Bootstrap Aggregating*)[11]

Setiap pohon dalam *Random Forest* dibangun pada dataset yang dihasilkan dengan pengambilan sample bootstrap(pengambilan sample dengan penggantian) dari dataset pelatihan asli. Hal ini memastikan keberagaman dataset yang digunakan untuk melatih setiap pohon.

2. Pohon Keputusan (*Decision Tree*) [12]

Setiap pohon keputusan dalam random forest dibangun dengan membagi dataset menjadi subset menggunakan kriteria pemisahan yang dipilih secara acak.

3. *Voting Average* [13]

Untuk masalah klasifikasi, prediksi akhir dari *Random Forest* yaitu hasil dari voting mayoritas semua pohon. Dengan demikian jika ada T pohon dan t pohon memprediksi kelas tertentu maka kelas tersebut dipilih sebagai prediksi akhir.

$$y_{\text{final}} = \operatorname{argmax}_k \sum_{t=1}^T I(y_t=k)$$

Untuk masalah regresi, prediksi akhir yaitu rata-rata dari semua pohon.

$$y_{\text{final}} = \operatorname{argmax}_k \frac{1}{T} \sum_{t=1}^T y_t$$

Dimana y_{final} merupakan prediksi akhir dari *Random Forest*, T yaitu jumlah pohon, y_t sebagai prediksi dari pohon ke- t , dan $I(-)$ merupakan fungsi indikator yang menghasilkan 1 jika kondisinya benar dan 0 sebaliknya.

III. HASIL DAN PEMBAHASAN

Penelitian ini membahas tentang prediksi dengan menggunakan klasifikasi untuk mendeteksi penyimpangan transaksi keuangan pada suatu Rumah Sakit. Pendeteksian ini dilakukan dengan menerapkan dua model prediksi klasifikasi model *Gradient Boosted Tree* dan model *Random Forest* untuk menentukan model yang terbaik dengan kriteria seperti akurasi, presisi, recall dan AUC(*Area Under The Curve*)[14] serta klasifikasi error dari kedua model tersebut. Kriteria akurasi yaitu menggambarkan prosentase klasifikasi, sedangkan presisi menjelaskan persentase akurasi data terhadap hasil prediksi kedua model tersebut. Demikian pula kriteria recall yaitu menjelaskan persentase model yang menghasilkan informasi yang optimal dengan tingkat klasifikasi *error* yang rendah. Sedangkan AUC ini menjelaskan *threshold*—ambang batas--yang terbaik dalam bentuk kurva dari suatu model.

Data-Set

Dataset medis merupakan kumpulan data transaksi keuangan medis suatu Rumah Sakit ABC selama satu tahun dengan karakteristik antara-lain: jumlah_pembayaran_per_tgl dimana nilai nya kisaran:100.000 sd 300.000; jumlah_resep_per_tgl dimana nilai nya kisaran :0 sd 85, maks_resep_pertgl dimana nilai nya kisaran: 0 sd 2; maks_resep_perdokter dimana nilai nya kisaran: 0 sd 3; maks_resep_perRS dimana nilai nya kisaran: 0 sd 1300; maks_resep_perthn dimana nilai nya kisaran: 0 sd 1200; jlh_pembayaran_perthn dimana nilai nya kisaran: 0 sd 450.000 (10.000); jlh_pembayaran_perRS dimana nilai nya kisaran: 27.000 sd 148.500 (10.000); jlh_pembayaran_perdokter dimana nilai nya kisaran: 17.000 sd 147.000(10.000); jlh_pembayaran_resep dimana nilai nya kisaran:17.500 sd 141.000(10.000); total_pembayaran dimana nilai nya kisaran: 400.000 sd 1200.000(10.000); jlh_resep_perthn dimana nilai nya kisaran: 102.000 sd 495.000; jlh_resep_perRS dimana nilai nya kisaran: 25.000 sd 167.000; jlh_resep_perdokter dimana nilai nya kisaran:17 sd 104; no_resep_keresep dimana nilai nya kisaran: 17 sd 143; dan total_resep dimana nilai nya kisaran: 400 sd 1200; dimana setiap karakteristik ini memiliki id data medis dan diberi label bernilai true atau false untuk proses klasifikasi prediksi transaksi keuangan ini. Tabel 1 menunjukkan karakteristik dari dataset medis seperti di bawah ini.

Tabel 1. Karakteristik Dataset Medis

No.	Atribut	Deskripsi
1	id	id dataset medis
2	label_penyimpangan_keuangan_medis	Pernyataan penyimpangan medis yaitu true atau false.
3	jlh_pembayaran_pertgl	Jumlah pembayaran setiap tanggal tertentu.
4	jlh_resep_pertgl	Jumlah resep yang dikeluarkan dokter setiap tanggal tertentu.
5	maks_resep_pertgl	Jumlah maksimal resep yang dikeluarkan dokter setiap tanggal tertentu.
6	maks_resep_perdokter	Jumlah maksimal resep yang dikeluarkan setiap dokter.
7	maks_resep_perRS	Jumlah maksimal resep yang dikeluarkan setiap Rumah Sakit.

8	maks_resep_perthn	Jumlah maksimal resep yang dikeluarkan setiap tahun.
9	jlh_pembayaran_perthn	Jumlah pembayaran setiap tahun.
10	jlh_pembayaran_perRS	Jumlah pembayaran setiap Rumah Sakit.
11	jlh_pembayaran_perdokter	Jumlah pembayaran setiap dokter
12	jlh_pembayaran_resep	Jumlah pembayaran dari resep
13	total_pembayaran	Total pembayaran
14	jlh_resep_perthn	Jumlah resep setiap tahun
15	jlh_resep_perRS	Jumlah resep setiap Rumah Sakit
16	jlh_resep_perdokter	Jumlah resep setiap dokter
17	no_resep_keresepe	No resep dalam resep
18	total_resep	Total resep

Data medis ini memiliki 1 atribut special dan 17 atribut lain dengan volume data mencapai 200 record. Data medis ini disimpan di file “DataMedis-Jan2024-CSV.CSV” seperti gambar 1 di bawah ini.

Row No.	nama	jlh_pemb...	jlh_resep_p...	maks_rese...	maks_rese...	maks_rese...	maks_rese...	no	jlh_pemb...	jlh_resep...
1	nama	500	0	1	1	0	0	21	888.800	71.100
2	nama	500.000	0	1	1	0	0	38	0.000	88.300
3	nama	500.000	0	1	1	0	0	27	100.000	04
4	nama	500	3	1	1	0	0	38	58.200	43.800
5	nama	500	22	1	1	0	22	24	05.000	38.200
6	nama	500.000	1	0	1	0	0	31	43.100	53
7	nama	500.000	1	0	1	0	0	134	49.800	42.800
8	nama	500.000	0	0	1	0	0	420	100.000	00
9	nama	500.000	4	0	1	0	0	13	127.200	78.800
10	nama	500.000	28	1	0	28	21	228	152.200	39.800
11	nama	500	0	0	1	0	0	320	0.700	28.500
12	nama	500.000	11	0	1	0	0	3	41.500	43.400
13	nama	500.000	11	0	1	0	0	87	107.000	78
14	nama	500.000	0	0	1	0	0	181	04.800	72.300

Gambar 1. Dataset Medis

Data Preprocessing

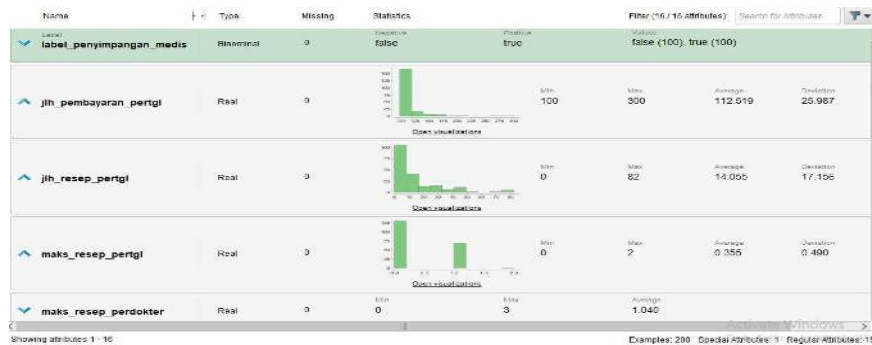
Tahap pertama dalam proses model pembelajaran prediksi klasifikasi adalah data preprocessing yaitu mempersiapkan dataset medis ini meliputi:

1. Distribusi kelas terhadap data ini dalam kondisi normal.
2. Tidak ada data yang hilang dataset medis ini.
3. Tidak ada data duplikasi dataset medis ini.

Jika terdapat duplikasi data maka dapat digunakan *remove duplicates* (operator Rapidminer) untuk menghapus data duplikasi.

Data-set ini dibagi menjadi data training dan testing dengan ukuran rasio 70:30.

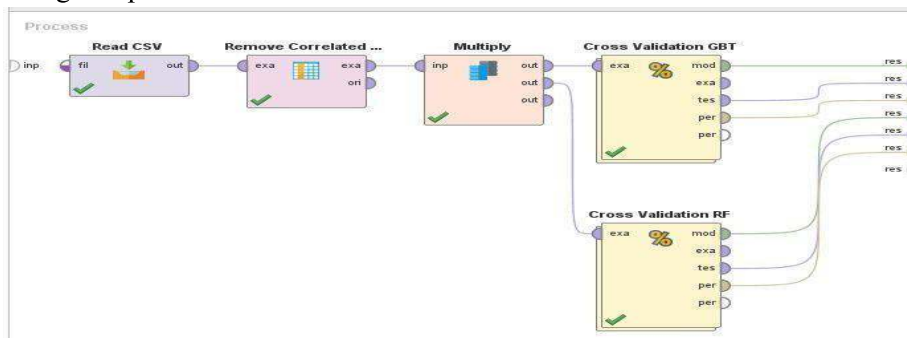
Dalam preprocessing dataset medis ini dengan menggunakan aplikasi *data mining* Rapidminer versi 9.10 dimana hasil statistik dataset medis ini seperti gambar 2 di bawah ini.



Gambar 2. Statistik Dataset Medis

Algoritma Model Pembelajaran Yang Digunakan

Penelitian ini menggunakan algoritma *Gradient Boosted Tree(GBT)* dan *Random Forest(RF)* dengan penerapan *Cross Validation* (lihat gambar 3) dimana *number of fold* adalah 10 bagian dari *data-set* medis ini dan berfungsi untuk membuat subproses training dan subproses testing dan performa model.

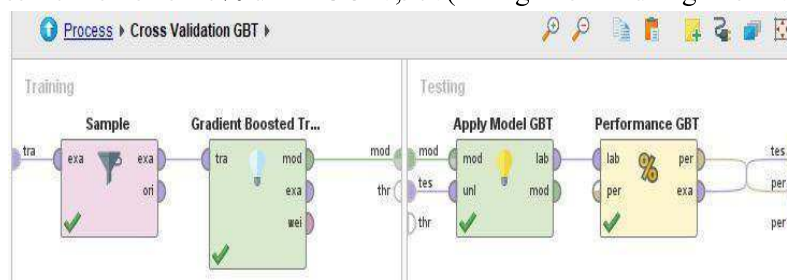


Gambar 3. Cross Validation Model GBT dan Model RF

Perfoma prediksi klasifikasi ini, menggunakan proses training dan testing terhadap dataset ini dengan rasio 70:30.

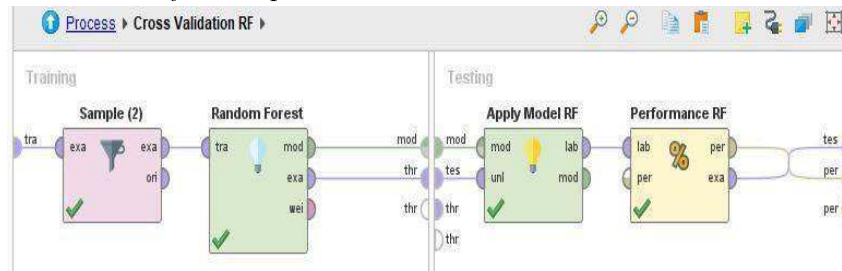
Penghitungan Performa Algoritma Model Pembelajaran

Proses penghitungan performa algoritma kedua model tersebut yaitu mengeksekusi model GBT dan RF pada rapidminer. Sebelum dilakukan proses training dan testing pada model GBT ini dilakukan penentuan beberapa parameter antara-lain: Number of trees = 20; Maximal dept level=5; Minimum rows=10 dan Number of bins=20. Kemudian dilakukan eksekusi model GBT ini dan menghasilkan accuracy = 93%, precision = 90, 57%, recall = 96% dengan classification error=7% dan AUC=0,976(lihat gambar 4 dan gambar 6).



Gambar 4. Proses Training dan Testing Model GBT

Sedangkan proses training dan testing pada model RF dengan menentukan parameter yaitu Number of trees = 20; Maximal dept level=5 dan Criterion = gain ratio. Hasil eksekusi model RF dengan beberapa parameter tersebut menghasilkan accuracy=93%, presision=79,84%, recall 99% dengan classification error=13% dan AUC=0,964 (lihat gambar 5 dan gambar 7) dimana AUC merupakan luas area di bawah curve ROC[15], atau integral dari fungsi ROC. Sedangkan ROC(Receiver Operating Characteristics) adalah alat ukur performa untuk *classification problem* dalam menentukan *threshold* dari suatu model.

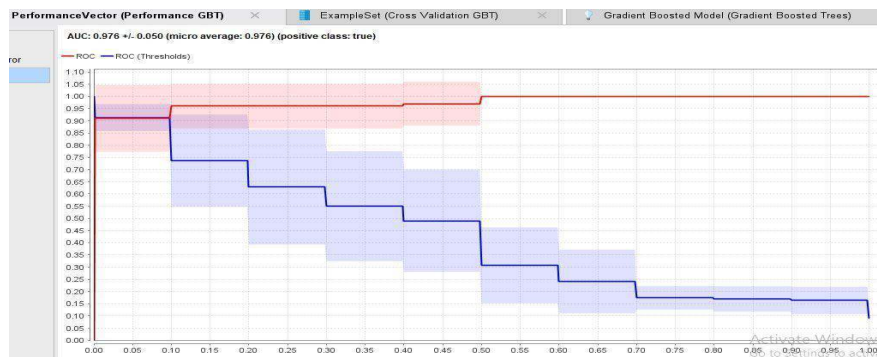


Gambar 5. Proses Training dan Testing Model RF

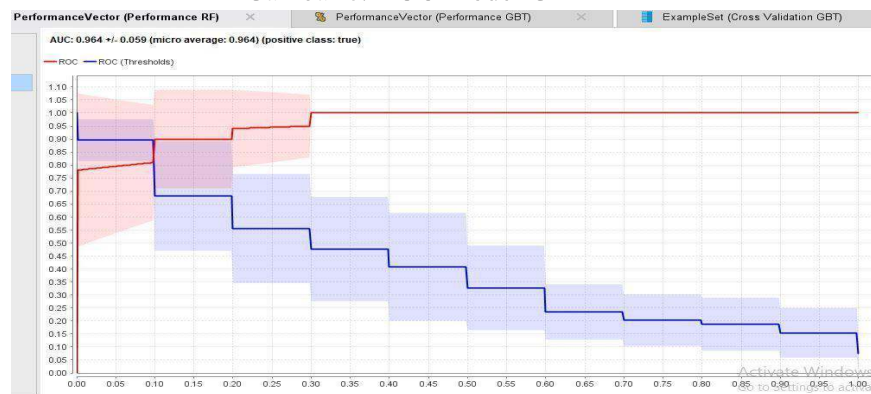
Hasil perbandingan evaluasi kedua model di atas dapat dilihat pada tabel 2 di bawah ini.

Tabel 2. Evaluasi Model GBT dan RF

Model	Accuracy	Precision	Recall	AUC	Classification Error	Kesimpulan
1.GBT	93 %	90,57%	96%	0,976	7%	Terbaik
2.RF	93%	79,84%	99%	0,964	13%	Baik



Gambar 6. AUC Model GBT



Gambar 7. AUC Model RF

IV. KESIMPULAN

Hasil evaluasi model GBT dan RF menunjukkan bahwa keakuratan kedua model tersebut dalam menggambarkan klasifikasi menghasilkan prosentase yang sama yaitu 93%. Sedangkan presisi yang menjelaskan akurasi data terhadap hasil prediksi bahwa model GBT mencapai 90,57% dibandingkan model RF hanya mencapai 79,84% meskipun prosentase recall yang menggambarkan model tersebut menghasilkan informasi untuk model RF lebih tinggi (99%) dengan classification error yaitu 13% dari pada model GBT (96%) dengan classification error lebih rendah (7%). Namun demikian AUC model GBT menunjukkan 0,976 dibandingkan dengan AUC model RF hanya 0,964. Maka hal ini dapat disimpulkan bahwa model GBT lebih baik proses prediksi dengan klasifikasi dibandingkan dengan model RF (lihat tabel 1 di atas).

V. SARAN

Bagaimana meningkatkan recall yang membuat model ini menghasilkan informasi yang lebih meningkatkan prosentasenya dibandingkan dengan model RF tersebut meskipun threshold AUC model GBT lebih baik dibandingkan dengan threshold AUC model RF.

DAFTAR PUSTAKA

- [1] AMRA, IHSAN A ABU & ASHRAF Y.A. MAGHARI. Student Performance Prediction Using KNN and Naïve Bayesian. pp 909–913. 2017 7th International Conference on Information Technology (ICIT), 2017
- [2] TIAN, D., HE, G., WU, J., CHEN, H., & JIANG, Y. An Accurate Eye Pupil Localization Approach Based on Adaptive Gradient Boosting Decision Tree. pp 27–30. 2016. <https://doi.org/978-1-5090-5316-2/2016>
- [3] DEVIKA,R., AVILALA, S. V., & Subramaniaswamy,V. Comparative Study of Classifier for Chronic Kidney Disease prediction using Naive Bayes , KNN and Random Forest. pp 679–684. 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), (Iccmc), 2019.
- [4] L. Lumbaa, E. Mailoa, and ..., “Implementasi Metode SVM Dan Gradient Boost Dalam Klasifikasi Bahasa Daerah (Halmahera, Kalimantan, Toraja)”, J. Tek. Inform. dan Sist. Inf. , vol. 9, no. 2, pp. 908–915, 2022.
- [5] G. A. Mursianto, I. M. Falih, M. Irfan, T. Sakinah, and D. S. Prasvita, “Perbandingan Metode Klasifikasi Random Forest dan XGBoost Serta Implementasi Teknik SMOTE pada Kasus Prediksi Hujan”, J. Senamika, vol. 2, no. 2, pp. 41–50, 2021.
- [6] E. Firasari, U. Khultsum, M. N. Winnarto, and R. Risnandar, “Kombinasi K-NN dan Gradient Boosted Trees untuk Klasifikasi Penerima Program Bantuan Sosial”, J. Teknol. Inf. dan Ilmu Komput., vol. 7, no. 6, pp. 1231-1236, 2020.
- [7] A. C. Nugraha and M. I. Irawan, “Komparasi Deteksi Kecurangan pada Data Klaim Asuransi Pelayanan Kesehatan Menggunakan Metode Support Vector Machine

- (SVM) dan Extreme Gradient Boosting (XGBoost)”, *J. Sains dan Seni ITS*, vol. 12, no. 1, pp. 40–46, 2023.
- [8] S. E. Herni Yulianti, Oni Soesanto, and Yuana Sukmawaty, “Penerapan Metode Extreme Gradient Boosting (XGBOOST) pada Klasifikasi Nasabah Kartu Kredit”, *J. Math. Theory Appl.*, vol. 4, no. 1, pp. 21–26, 2022.
- [9] G. M. Momole, “Perbandingan Naïve Bayes dan Random Forest Dalam Klasifikasi Bahasa Daerah”, *J. Tek. Inform. dan Sist. Inf.*, vol. 9, no. 2, pp. 855–863, 2022.
- [10] E. Renata and M. Ayub, “Penerapan Metode Random forest untuk Analisis Risiko pada dataset Peer to peer lending”, *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 3, pp. 462–474, 2020.
- [11] TH Lee, A Ullah, R Wang, “Bootstrap Aggregating and Random Forest”, Vol 2, pp. 389–429, Springer Link: ASTA Book Series, 2019.
- [12] S Suthaharan, S Suthaharan, “Decision tree learning”, vol 36, pp 237–269, Springer Link: ISIS book series, 2016.
- [13] R Renault, A Trannoy, “Protecting minorities through the average voting rule”, Wiley Online Library: *Journal of Public Economic Theory*, 2005.
- [14] Chun-Tao Wai, Chee Leong Cheng, Aileen Wee, Yock-Young Dan, Edwin Chan, Winnie Chua, Belinda Mak, Aung Myat Oo, Seng Gee Lim, “Non-invasive models for predicting histology in patients with chronic hepatitis B”, National University of Singapore: Wiley, Online Library, 2006.
- [15] Miguel Angel Luque-Fernandez, Daniel Redondo-Sánchez, and Camille Maringe, “Cvauroc: Command to compute cross-validated area under the curve for ROC analysis after predictive modeling for binary outcomes”, Volume 19, Issue 3, STATA Press: Sage Journal, 2019.