

# Evaluasi Tiga Jenis Algoritme Berbasis Pembelajaran Mesin untuk Klasifikasi Jenis Tumor Payudara

Annisa Handayani<sup>1</sup>, Ade Jamal<sup>2</sup>, Ali Akbar Septiandri<sup>3</sup>

**Abstract**—According to the World Health Organization (WHO), malignant tumor (cancer) is one of the leading causes of mortality worldwide. Among all types of malignant tumors, malignant breast tumor (breast cancer) is the most common malignant tumors found, especially in women. One of four ways to distinguish benign breast tumor from malignant ones is by doing Fine Needle Aspiration (FNA) test. This method is simple, quick, inexpensive, and can be performed either in outpatients or inpatients. Although FNA is much more preferred than the other methods, the accuracy of FNA varies widely. Therefore, this research is conducted to find the best model to classify malignant breast tumors and benign breast tumors, based on data from FNA test by evaluating value of Area Under the Curve (AUC) of three algorithms, Extreme Gradient Boosting (XGBoost), Support Vector Machine with Radial Basic Function kernel (SVM-RBF), and Multi-layer Perceptron (MLP). The breast cancer data set used in this research is obtained from Wisconsin Breast Cancer (Original) Data set available in the UCI Machine Learning Repository. The experiment shows that SVM with eliminating missing value data set achieved the best result, with AUC value 99.23 and cost \$2,740.20.

**Intisari**—Berdasarkan data *World Health Organization* (WHO), tumor ganas merupakan salah satu penyebab kematian yang tinggi di dunia. Di antara seluruh jenis tumor ganas (kanker), tumor ganas payudara merupakan tumor ganas yang paling sering ditemukan, khususnya pada wanita. Salah satu cara untuk membedakan tumor ganas payudara dan tumor jinak payudara adalah dengan melakukan tes *Fine Needle Aspiration* (FNA). Metode ini disukai karena mudah dilakukan, aman, sederhana, murah, serta dapat dilakukan pada pasien rawat jalan maupun rawat inap. Meskipun metode ini banyak disukai, tetapi FNA memiliki tingkat akurasi yang bervariasi. Oleh karena itu, makalah ini disusun untuk menemukan model terbaik dari data tes FNA untuk mengklasifikasikan tumor ganas dan tumor jinak payudara dengan melakukan evaluasi pada nilai *Area Under the Curve* (AUC) dari Algoritme *Extreme Gradient Boosting* (XGBoost), *Support Vector Machine Kernel Radial Basic Function* (SVM-RBF), dan *Multilayer Perceptron* (MLP). Data yang digunakan merupakan data *Wisconsin Breast Cancer* (WBC) yang tersedia di *UCI Machine Learning Repository*. Hasil terbaik berdasarkan nilai AUC dan *score cost* terendah diraih oleh algoritme SVM-RBF pada data set yang

dilakukan *eliminate missing value* dengan nilai AUC 99,23 dan *score cost* \$2.740,20.

**Kata Kunci**— *Klasifikasi, Tumor Payudara, Wisconsin Breast Cancer (WBC), Extreme Gradient Boosting (XGBoost), Support Vector Machine Kernel Radial Basic Function (SVM-RBF), Multilayer Perceptron (MLP)*

## I. PENDAHULUAN

Menurut Yayasan Kanker Indonesia, penyakit kanker merupakan penyakit yang timbul akibat sel jaringan tubuh yang tumbuh tidak normal dan membelah diri secara cepat dan tidak terkendali sehingga berubah menjadi sel kanker [1]. Dalam perkembangannya, sel tumor ganas (kanker) dapat menyebar ke jaringan atau bagian tubuh lainnya, sehingga dapat menyebabkan kematian. Banyak masyarakat yang menyebut kanker sebagai tumor. Padahal tumor adalah segala benjolan tidak normal atau abnormal. Tumor sendiri dibagi dalam dua golongan, yaitu tumor jinak (*benign*) dan tumor ganas (*malignant*). Kanker merupakan istilah umum untuk semua jenis tumor ganas.

Berdasarkan data *World Health Organization* (WHO), tumor ganas merupakan salah satu penyebab kematian yang tinggi di dunia [2]. Ditemukan 7,6 juta kasus tumor ganas yang terjadi pada tahun 2004 di seluruh dunia. Di antara seluruh jenis tumor ganas, tumor ganas payudara (kanker payudara) merupakan tumor ganas yang paling sering ditemukan, khususnya pada wanita. Prevalensi tumor ganas payudara mencapai 23% dari seluruh kejadian keganasan dengan 14% di antaranya menyebabkan kematian.

Hal tersebut didukung data statistik *American Cancer Society* yang menunjukkan bahwa tumor ganas payudara menempati urutan pertama dengan jumlah kasus terbanyak dari seluruh jenis kasus tumor ganas di seluruh dunia [3]. Pada tahun 2012, tercatat sebanyak 1,7 juta kasus baru (25% dari seluruh kasus kanker pada perempuan) didiagnosis, dan 53% dari kasus tersebut terjadi di negara berkembang. Namun, di Afrika dan Asia Tenggara, termasuk Indonesia, kanker payudara menempati urutan kedua setelah kanker leher rahim. Meskipun menempati urutan kedua, data statistik Kementerian Kesehatan RI menunjukkan, penderita kanker payudara meningkat setiap tahunnya [4]. Hal ini terjadi karena di negara berkembang seperti Indonesia, kanker payudara tidak didiagnosis sejak dini. Biasanya, kanker didiagnosis setelah memasuki stadium lanjut dan memiliki tingkat *survivability* rendah.

Terdapat empat metode untuk membedakan tumor ganas dan jinak, yaitu *Surgical Biopsy*, *Mammography*, *Magnetic Resonance Imaging* (MRI), dan *Fine Needle Aspiration* (FNA) [5]. Dari keempat metode tersebut, *surgical biopsy* merupakan

<sup>1,3</sup> Program Studi Teknik Informatika, Universitas Al Azhar Indonesia, Jl. Sisingamangaraja, Kebayoran Baru, Kota Jakarta Selatan, Daerah Khusus Ibukota Jakarta 12110 INDONESIA (telp: 021-727 92753; fax: 021-724 4767; e-mail: <sup>1</sup>annisa.handayani@if.uai.ac.id, <sup>3</sup>aliakbars@live.com)

<sup>2</sup> Corresponding Author, Program Studi Teknik Informatika, Universitas Al Azhar Indonesia, Jl. Sisingamangaraja, Kebayoran Baru, Kota Jakarta Selatan, Daerah Khusus Ibukota Jakarta 12110 INDONESIA (telp: 021-727 92753; fax: 021-724 4767; e-mail: <sup>2</sup>adja@uai.ac.id)

metode yang memiliki tingkat akurasi tertinggi (hampir 100%). Sedangkan metode *mammography* memiliki akurasi 68%--79% [6]. MRI memiliki tingkat akurasi 70% untuk mendeteksi *benign* (tumor jinak payudara) dan 92% untuk *malignant* (tumor ganas payudara) [7]. Sedangkan FNA memiliki *range* akurasi yang lebih luas, yaitu 65%--98% [8]. Meskipun *surgical biopsy* memiliki tingkat akurasi tertinggi, metode ini terbilang mahal dan membutuhkan waktu. Metode FNA merupakan metode yang paling terjangkau, tetapi memiliki tingkat akurasi yang bervariasi. Saat ini metode FNA lebih disukai, karena lebih mudah dilakukan, aman, sederhana, tanpa komplikasi, murah, serta dapat dilakukan pada pasien rawat jalan maupun rawat inap dengan resiko yang sangat kecil [9]. Oleh karena itu, dibutuhkan sebuah model yang dapat mendiagnosis hasil tes FNA dengan akurasi tinggi.

Terdapat beberapa penelitian yang sudah dilakukan sebelumnya dengan menggunakan data *Wisconsin Breast Cancer (WBC)* yang tersedia di *UCI Machine Learning Repository*, yang berisi 699 *clinical instances* mengenai data tumor ganas payudara dan tumor jinak payudara dari hasil diagnosis *breast masses* berdasarkan tes FNA untuk membedakan tumor payudara ganas dan tumor payudara jinak [5], [10]--[16]. Namun, sebagian besar penelitian tersebut fokus untuk mendapatkan tingkat akurasi yang tinggi. Padahal dalam kasus kesehatan, akurasi bukan satu-satunya tujuan utama dalam penelitian. Sensitivitas (*sensitivity*) dan kekhususan (*specificity*) juga harus diperhatikan agar model yang dibangun sesuai dengan kebutuhan di bidang kedokteran, khususnya dalam melakukan klasifikasi tumor ganas payudara dan tumor jinak payudara. Dari delapan penelitian tersebut, hanya satu penelitian yang mencantumkan sensitivitas dan kekhususan dalam penelitiannya [10].

Uraian di atas melatarbelakangi disusunnya makalah yang berjudul Evaluasi Tiga Algoritme Berbasis Pembelajaran Mesin untuk Klasifikasi Jenis Tumor Payudara.

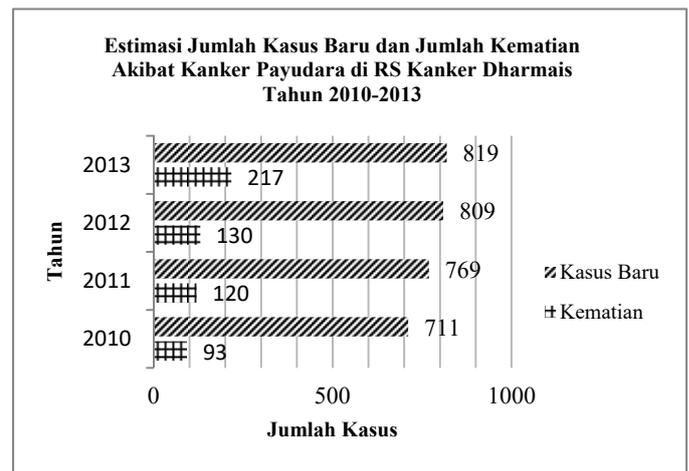
## II. PENYAKIT TUMOR PAYUDARA

Dalam sebuah literatur dijelaskan bahwa tumor dapat dibedakan menjadi dua, yaitu tumor jinak dan tumor ganas [17]. Tumor ganas lebih sering disebut dengan kanker. Tumor jinak tidak memiliki kemampuan untuk menginfiltrasi jaringan sekitar, tidak bermetastasis ke organ lain, serta sebagian besar di antaranya tumbuh perlahan. Sedangkan tumor ganas atau kanker memiliki laju pertumbuhan yang relatif lebih cepat jika dibandingkan dengan tumor jinak, serta biasanya tumbuh dengan cara infiltrasi, invasi, destruksi, hingga metastasis ke jaringan sekitar. Tumor payudara merupakan sekelompok sel yang tidak normal pada payudara yang terus tumbuh berlipat ganda. Sel-sel ini pada akhirnya menjadi bentuk benjolan di payudara.

Data statistik *American Cancer Society* menunjukkan bahwa tumor ganas payudara menempati urutan pertama dengan jumlah kasus terbanyak dari seluruh jenis kasus tumor ganas di seluruh dunia [3]. Pada tahun 2012, tercatat sebanyak 1,7 juta kasus baru (25% dari seluruh kasus kanker pada perempuan) didiagnosis, dan 53% dari kasus tersebut terjadi

di negara berkembang. Namun, di Afrika dan Asia Tenggara, termasuk Indonesia, kanker payudara menempati urutan kedua setelah kanker leher rahim. Meskipun menempati urutan kedua, data statistik Kementerian Kesehatan RI menunjukkan, penderita kanker payudara meningkat setiap tahunnya [4].

Grafik pertumbuhan penderita kanker payudara ditunjukkan pada Gbr. 1. Meningkatnya jumlah penderita kanker di negara berkembang seperti Indonesia dikarenakan kanker payudara tidak didiagnosis sejak dini. Biasanya kanker didiagnosis setelah memasuki stadium lanjut dan memiliki tingkat *survivality* rendah. Di negara maju, kanker biasanya didiagnosis sejak dini, sehingga tingkat *survivality* menjadi relatif lebih tinggi daripada di negara berkembang.



Gbr. 1 Estimasi jumlah kasus baru dan jumlah kematian akibat kanker payudara di RS Kanker Dharmais tahun 2010-2013. [4]

Sebelum penelitian dimulai, dilakukan studi literatur terhadap beberapa penelitian yang menggunakan data WBC. Sebuah penelitian mengimplementasikan metode *Back-propagation Neural Network* dalam mengklasifikasi WBC menjadi kelas *benign* dan *malignant* [5]. Model *Multi Layer Perceptron (MLP)* yang digunakan menggunakan dua *hidden layer*. Penelitian ini mengusulkan untuk melatih model menggunakan proses *iterative training*. Proses tersebut diulang sampai mendapatkan *sum-squared of error* yang diinginkan. Untuk penelitian selanjutnya, disarankan untuk meningkatkan akurasi model dengan menambah jumlah *hidden neuron*. Penelitian lain mengimplementasi *Fuzzy c-means* untuk mengelompokkan data WBC menjadi dua *cluster*, yaitu *benign* dan *malignant* [11]. Dari percobaan tersebut diperoleh hasil berupa *True Positive* 100%, *True Negative* 87%, *False Positive* 0%, dan *False Negative* 13%.

*Statistical neural networks* telah digunakan dalam melakukan diagnosis kanker payudara [12]. Penelitian tersebut membandingkan *MLP*, *Radial Basic Function (RBF)*, *General Regression Neural Networks (GRNN)*, dan *Probabilistic Neural Network (PNN)*. Algoritme *GRNN* merupakan algoritme yang menghasilkan akurasi tertinggi. Pada penelitian lain, algoritme *Bayesian Network*, *Naive Bayes*, *Decision Tree J4.8*, *ADTree*, dan *Multi-layer Neural Network (back-propagation)* dibandingkan [13]. Penelitian tersebut menyarankan untuk melakukan penelitian di tingkat

genom untuk mendapatkan akurasi yang lebih baik. Hal ini dapat membantu dalam diagnosis kanker payudara karena kelainan yang terjadi pada gen.

Berbeda dengan penelitian sebelumnya yang menjadikan akurasi sebagai pengukur kinerja *classifier*, sebuah penelitian menggunakan *metrics* tambahan, yaitu sensitivitas dan kekhususan [10]. Penelitian tersebut membandingkan *Extreme Learning Machine Neural Network* (ELM ANN) dan *Back Propagation Neural Network* (BP ANN). Algoritme ELM ANN unggul dalam akurasi dan kekhususan, tetapi dalam sensitivitas, algoritme BP ANN memiliki persentase yang lebih baik dibandingkan ELM ANN. Penelitian yang lain lagi membandingkan kombinasi algoritme *Decision Tree* dengan *Support Vector Machine* (DT-SVM), *Instance-based Learning* (IBL), *Sequential Minimal Optimization*, dan *Naïve Bayes* [14]. Masukan parameter untuk SVM dioptimalkan dengan algoritme DT. Hasil yang didapatkan dari percobaan yang dilakukan adalah metode DT-SVM memiliki akurasi tertinggi, yaitu 91%.

Metode SVM dan *Genetic Programming* juga telah digunakan [15]. Sebanyak 80% dari total 683 *instance* data digunakan sebagai data latih, dan sisa 20% digunakan sebagai data uji. Data latih dilatih dengan *Kernel* Linear, Polinomial, dan RBF dari SVM dengan parameter yang berbeda. Dari ketiga metode tersebut, RBF menghasilkan akurasi tertinggi, yaitu 97,37%, sedangkan Linear 94,49%, dan Polinomial 95,72%.

Sebuah penelitian lain menggunakan *Information Gain and Adaptive Neuro Fuzzy Inference System* (IGANFIS) untuk melakukan klasifikasi [16]. Hasil dari metode IGANFIS memiliki akurasi 98,24%. Penelitian ini juga menggunakan *Iterative Weighted k-NN* untuk menggantikan nilai fitur yang hilang. Hasil dari penelitian menggunakan metode *Iterative Weighted k-NN* memiliki maksimum akurasi sebesar 96,98%. Penelitian ini menyarankan untuk mengombinasikan metode *machine learning* agar diperoleh hasil akurasi yang lebih tinggi untuk diagnosis medis. Penelitian ini menjelaskan untuk pengembangan selanjutnya tidak hanya berfokus pada hasil akurasi yang tinggi, tetapi juga kecepatan dalam melakukan klasifikasi dan biaya komputasi.

### III. ALGORITME PEMBELAJARAN MESIN

#### A. Tiga Jenis Algoritme

Klasifikasi merupakan teknik *machine learning* yang digunakan untuk melakukan pembelajaran terhadap fungsi target  $f$  yang memetakan setiap atribut  $x$  ke satu dari sejumlah label kelas yang tersedia. *Machine learning* yang dimaksud mengacu pada sistem yang memiliki kemampuan untuk belajar dari pengalaman secara otomatis [18]. Proses klasifikasi meliputi dua langkah proses. Tahap pertama adalah *learning step*, yaitu tahap saat algoritme klasifikasi membangun *classifier* dengan menganalisis *training set* dari data latih yang diberikan. Kemudian, langkah selanjutnya menggunakan model *classifier* yang telah dibangun untuk melakukan klasifikasi [19].

1) *Support Vector Machine (SVM)*: Algoritme SVM menggunakan ruang hipotesis berupa fungsi linear dalam sebuah ruang fitur (*feature space*) berdimensi tinggi yang

dilatih dengan metode pembelajaran berdasarkan teori optimasi dan pembelajaran statistik [20]. SVM dikembangkan oleh Boser, Guyon, dan Vapnik dengan konsep mengombinasikan teori komputasi seperti *margin hyperplane*, *kernel*, dan konsep-konsep pendukung lain. Teori tersebut sebenarnya sudah ada puluhan tahun sebelumnya, tetapi belum ada upaya untuk merangkaikan komponen-komponen tersebut.

Konsep SVM dapat dijelaskan sebagai usaha mencari fungsi pemisah (*hyperplane*) terbaik yang berfungsi memisahkan dua kelas, yaitu kelas +1 (positif) dan kelas -1 (negatif). *Margin* merupakan jarak antara *hyperplane* dengan *pattern* terdekat dari masing-masing kelas. *Pattern* yang paling dekat ini disebut *support vector*. Karena itu, metodenya disebut sebagai *Support Vector Machine* [20].

Pada umumnya, sangat jarang ditemukan masalah dalam domain dunia nyata yang terpisah secara linear. Dalam menyelesaikan masalah nonlinear, SVM menggunakan fungsi *kernel*. Secara konseptual, ide dari fungsi *kernel* ini adalah untuk memetakan  $\vec{x}$  oleh fungsi  $\Phi(\vec{x})$  ke ruang vektor berdimensi lebih tinggi [21]. Transformasi  $\Phi$  tidak perlu diketahui dan sangat sulit untuk dimengerti, sehingga perhitungan *dot product* dapat digantikan dengan fungsi kernel  $K(\vec{x}_i, \vec{x}_j)$  yang disebut sebagai *Kernel Trick*.

Fungsi *Kernel* merupakan fungsi untuk menghadapi kasus yang tidak linear. *Kernel trick* memberikan berbagai kemudahan karena dalam proses pembelajaran SVM, untuk menentukan *support vector*, cukup dengan mengetahui fungsi *kernel* yang dipakai, dan tidak perlu mengetahui wujud dari fungsi nonlinear.

Fungsi *kernel* yang direkomendasikan untuk diuji pertama kali adalah fungsi *kernel* RBF, karena *kernel* ini memiliki kinerja yang sama dengan *kernel* linear pada parameter tertentu, serta memiliki perilaku seperti fungsi *kernel sigmoid* pada parameter tertentu dan rentang nilainya kecil [0,1] [22]. *Kernel* ini memetakan sampel ke dalam ruang dimensi yang lebih tinggi sehingga dapat menangani kasus ketika hubungan antara kelas dan atribut tidak linear.

Akurasi model yang dihasilkan dari proses pelatihan dengan menggunakan SVM sangat bergantung pada fungsi *kernel* serta parameter yang digunakan. Parameter yang diatur untuk algoritme ini adalah parameter  $C$  (*cost*) dan  $\gamma$  (*gamma*). Semakin besar nilai  $C$  akan memberikan *penalty* yang lebih besar terhadap klasifikasi tersebut. Sedangkan parameter  $\gamma$  (*gamma*) digunakan pada *kernel Radial Basic Function* untuk mentransformasikan data latih ke ruang fitur yang kemudian dioptimasi menggunakan metode *Lagrange Multipliers* sehingga menghasilkan nilai  $\alpha$  yang digunakan untuk menentukan *support vector* dan menduga koefisien  $w$  (bobot) maupun  $b$  (*bias*) pada model klasifikasi.

Untuk *kernel* RBF, nilai parameter yang sebaiknya digunakan setiap iterasi bertambah secara eksponensial, seperti  $C = 2^{-5}, 2^{-3}, \dots, 2^{15}$  serta  $\gamma = 2^{-15}, 2^{-13}, \dots, 2^3$  [22]. Kemudian, jika sudah ditemukan parameter yang baik, dilakukan pencarian parameter pada rentang nilai yang lebih kecil agar diperoleh parameter yang terbaik. Setelah menemukan parameter terbaik, pelatihan dilakukan dengan

menggunakan keseluruhan data. Dalam mengimplementasikan algoritme ini, digunakan *library scikit-learn*.

2) *Neural Networks*: Istilah *Neural Network* memiliki asal-usul dalam upaya untuk menemukan representasi matematik dari pengolahan informasi dalam sistem biologis. Jaringan saraf tiruan (*Artificial Neural Network/ANN*) merupakan konsep rekayasa pengetahuan dalam bidang kecerdasan buatan yang didesain dengan mengadopsi sistem saraf manusia, yang pemrosesan utamanya berada di otak [23].

Dijelaskan bahwa seringkali neuron divisualisasikan ke dalam bentuk lapisan (*layer*), dengan neuron-neuron yang berada pada lapisan yang sama memiliki perilaku yang sama. Fungsi aktivasi dan pola dari bobot menjadi faktor penentu perilaku dari neuron. Lapisan penyusun jaringan saraf tiruan terdiri atas *input layer*, *hidden layer*, serta *output layer*. Penyusunan neuron pada lapisan-lapisan dan pola koneksinya dalam dan antar lapisan disebut arsitektur jaringan. Adapun jenis arsitektur jaringan yang sering dipergunakan yaitu jaringan berlapis tunggal (*Single-Layer Feedforward Networks*), jaringan berlapis banyak (*Multilayer Feedforward Networks*), dan jaringan saraf berulang (*Recurrent Networks*).

Fungsi aktivasi merupakan salah satu parameter yang penting dalam jaringan saraf tiruan. Fungsi aktivasi,  $\varphi(\cdot)$ , digunakan untuk menentukan keluaran suatu neuron. Fungsi ini meliputi penjumlahan bobot sinyal masukan dan mengaplikasikannya pada keluaran. Untuk unit masukan, fungsi ini adalah fungsi identitas. Pada umumnya, fungsi aktivasi yang sama digunakan untuk semua neuron pada lapisan tertentu dari *NN* [24]. Pada makalah ini, digunakan fungsi aktivasi *Rectified Linier Units* (ReLU). Fungsi ini telah menjadi fungsi yang sangat populer dalam beberapa tahun terakhir, karena fungsi ini mudah untuk dioptimalkan [25]. Selain itu, ReLU merupakan fungsi aktivasi yang tidak cepat jenuh karena tidak *asymptotic*. Persamaan fungsi ini dinyatakan dalam (1).

$$h(x_i) = \max(0, x_i) \quad (1)$$

*Backpropagation Neural Network* (BPNN) merupakan metode untuk melatih *multi-layer neural network*. Metode ini menggunakan *gradient-descent* berdasarkan aturan *delta-learning* yang disebut aturan *backpropagation* [26]. *Gradient-descent* digunakan untuk melakukan *update* bobot dan *bias* untuk mendapatkan kesalahan terkecil dari keluaran. Setiap perubahan bobot yang terjadi dapat mengurangi *error*. Perubahan bobot dapat dilakukan dengan menggunakan fungsi aktivasi yang berbeda. Siklus perubahan bobot (*epoch*) dilakukan pada setiap set pelatihan hingga kondisi berhenti dicapai. Kondisi berhenti tersebut dapat berupa jumlah *epoch* yang diinginkan sudah tercapai atau setelah melampaui sebuah nilai ambang yang telah ditetapkan.

3) *Extreme Gradient Boosting*: Selain dikombinasikan dengan *tree*, *boosting* juga dapat dikombinasikan dengan *gradient descent*. Metode ini pertama kali diperkenalkan oleh Friedman [27]. Dalam penelitian tersebut digunakan hubungan antara *boosting* dan optimasi untuk membuat *Gradient Boosting Machine* (GBM). *Gradient Boosting*

merupakan teknik dalam *machine learning* untuk masalah regresi dan klasifikasi yang menghasilkan model prediksi dalam bentuk gabungan (*ensemble*) model prediksi yang lemah. Pembangunan model dilakukan dengan menggunakan metode *boosting*, yaitu dengan membuat model baru untuk memprediksi *error/residual* dari model sebelumnya. Model baru ditambahkan hingga tidak ada lagi perbaikan pada *error* yang dapat dilakukan. Algoritme ini dinamakan *gradient boosting* karena menggunakan *gradient descent* untuk memperkecil *error* saat membuat model baru.

Sebuah penelitian lain kemudian melakukan perubahan minor terhadap *regularized objective* pada GBM [28]. XGBoost merupakan versi GBM yang lebih efisien dan *scalable*. XGBoost mampu mengerjakan berbagai fungsi seperti regresi, klasifikasi, dan *ranking*. XGBoost merupakan *tree ensembles algorithm* yang terdiri atas kumpulan beberapa *classification and regression trees* (CART). Faktor yang paling penting di balik kesuksesan XGBoost adalah *scalability* dalam berbagai skenario. *Scalability* ini disebabkan karena optimasi dari algoritme sebelumnya. Inovasi yang diberikan meliputi *novel tree learning algorithm* untuk menangani *sparse data*. Kesuksesan tersebut terbukti saat XGBoost menjadi salah satu metode yang sedang banyak diterapkan untuk berbagai kasus pada *machine learning*.

XGBoost pertama kali dikenalkan dalam *Higgs Boson Competition*. Pada akhir kompetisi, XGBoost menjadi metode yang paling banyak digunakan oleh sebagian besar tim yang mengikuti kompetisi tersebut. Hal ini juga terjadi pada kompetisi *machine learning* yang diselenggarakan oleh situs Kaggle selama tahun 2015. Dari 29 *winning solution*, 17 di antaranya menggunakan algoritme XGBoost. Dari 17 solusi tersebut, sebanyak delapan solusi menggunakan XGBoost untuk model *training*, sedangkan sisanya mengombinasikan algoritme XGBoost dengan algoritme *NN*.

## B. Evaluasi Model

Sebuah sistem yang melakukan proses klasifikasi diharapkan dapat mengklasifikasikan semua data dengan benar. Namun, tidak dapat dipungkiri bahwa kinerja suatu sistem tidak bisa 100% benar, sehingga sebuah sistem klasifikasi juga harus diukur kinerjanya. Pada makalah ini, digunakan *confusion matrix* dan kurva *Receiver Operating Characteristic* (ROC) untuk mengukur kinerja *classifier*. *Confusion matrix* adalah tabel pencatat hasil klasifikasi.

Pengukuran kinerja standar untuk masalah klasifikasi yang biasanya digunakan adalah akurasi. Namun, untuk kasus yang memiliki distribusi kelas tidak seimbang, perlu digunakan kekhususan dan sensitivitas sebagai pengukuran tambahan. Dalam kasus ini, sensitivitas merupakan ukuran yang mengukur seberapa baik sebuah model dalam memprediksi pasien tumor ganas payudara (*malignant*) sebagai pasien tumor ganas payudara (*malignant*). Sedangkan kekhususan merupakan ukuran yang mengukur seberapa baik sebuah model dalam memprediksi pasien tumor jinak payudara (*benign*) sebagai pasien tumor jinak payudara (*benign*).

$$\text{akurasi} = \frac{TP + TN}{TP + FP + FN + TN} \quad (2)$$

$$\text{sensitivitas} = \frac{TP}{TP + FN} \quad (3)$$

$$\text{kekhususan} = \frac{TN}{FP + TN} \quad (4)$$

Sedangkan kurva ROC merupakan grafik antara sensitivitas (*true positive rate*) pada sumbu Y dengan 1-kekhususan pada sumbu X (*false positive rate*). Kurva ini seakan-akan menggambarkan tarik-menarik/tawar-menawar antara sumbu Y (sensitivitas) dengan sumbu X (1-kekhususan). Dari kurva ini, didapatkan titik potong, yaitu nilai batas normal dan abnormal atau batas hasil uji positif dan negatif [29]. Selain titik potong, kurva tersebut juga memperlihatkan *Area Under the Curve* (AUC) yaitu luas area di bawah kurva ROC.

#### IV. METODOLOGI

##### A. Sumber Data

Data yang digunakan merupakan data WBC (*Original*) yang tersedia di *UCI Machine learning Repository* [30]. *Data set* ini berisi informasi mengenai kasus klinis tumor payudara yang dibuat oleh Dr. William H. Wolberg dari University of Wisconsin Hospital. *Data set* ini terdiri atas 699 *instance* berupa data klinis hasil diagnosis *breast masses* dengan tes FNA. Dari 699 *instance*, 458 (65,52%) di antaranya dikategorikan sebagai *benign*, sedangkan 241 (34,47%) dikategorikan sebagai *malignant*.

TABEL I  
ATRIBUT *DATA SET* WBC

No.	Nama Atribut	Nilai
1.	ID	ID numeric
2.	Clump Thickness	1-10
3.	Uniformity of Cell Size	1-10
4.	Uniformity of Cell Shape	1-10
5.	Marginal Adhesion	1-10
6.	Single Epithelial Cell Size	1-10
7.	Bare Nuclei	1-10
8.	Bland Chromatin	1-10
9.	Normal Nucleoli	1-10
10.	Mitoses	1-10
11.	Class	2 ( <i>benign</i> ), 4 ( <i>malignant</i> )

Setiap *instance* terdiri atas sembilan atribut dengan nilai *integer* dengan *range* 1--10 serta satu kategori kelas dengan nilai 2 (*benign*) dan 4 (*malignant*). Dari 699 *instance* terdapat 16 *instance* yang berisi *missing value* yang dinotasikan dengan “?” pada atribut *bare nuclei*. Atribut data WBC diperlihatkan pada Tabel I. Nilai dengan *range* 1--10 untuk sembilan atribut di atas merupakan hasil interpretasi *cytopathologist* (dokter spesialis yang memeriksa sel dengan mikroskop) mengenai hasil diagnosis *breast masses* dari tes FNA. Karena nilai tersebut bukan hasil pengukuran, maka distribusi data untuk setiap atribut menjadi tidak mengikuti distribusi normal.

##### B. Data Preprocessing

Karena pada atribut *bare nuclei* terdapat 16 *instance* yang berisi *missing value*, maka dilakukan *data preprocessing* untuk membangun dua *data set*, yaitu *Data set A* dan *Data set B*. *Data set A* merupakan data WBC yang dikenai *replace missing value* dengan median (nilai tengah) dari kolom yang terdapat *missing value*. Pemilihan median untuk mengganti *missing value* karena data dengan atribut *bare nuclei* tidak mengikuti distribusi normal. Distribusi data dari atribut *bare nuclei* didominasi oleh nilai 1 dengan persentase 59%.

Sedangkan *Data set B* merupakan data WBC yang dikenai proses *eliminate missing value*, sehingga *Data set B* hanya terdiri atas 683 *instance*.

Adapun perbandingan data latih dan data uji untuk masing-masing *data set* adalah 67% dan 33%. Fungsi *train\_test\_split()* yang disediakan oleh *library scikit-learn* digunakan untuk membagi data latih dan data uji pada *Data set A* dan *Data set B*.

##### C. Fungsi Score Cost

Untuk menemukan estimator terbaik untuk setiap *classifier*, diperlukan fungsi *scoring* yang digunakan untuk menjadi indikator. Di sini diimplementasikan fungsi *detection cost false positive* dan *false negative*.

$$\text{Detection Cost} = ((\text{Cost FP} * \text{FP}) + (\text{Cost FN} * \text{FN})) \quad (5)$$

*Cost false positive* merupakan *cost* yang dikeluarkan untuk melakukan tes FNA. Biaya untuk melakukan tes FNA €210 (\$228,35) per lesi (*sample jaringan abnormal yang diambil*) [31]. Sedangkan *cost false negative* dihitung berdasarkan lama (tahun) potensi hidup yang hilang saat kematian akibat *breast cancer* yang dikalikan dengan *value of a year of life*. Digunakan *value of a year of life* yang sebelumnya sudah dipublikasikan, yaitu \$150.000 untuk pria dan wanita [32].

Sedangkan berdasarkan data *Canadian Breast Cancer Foundation Ontario*, rata-rata perempuan Amerika Serikat yang meninggal akibat *breast cancer* meninggal 19 tahun lebih awal dibanding jika tidak meninggal akibat *breast cancer* [33]. *Cost false negative* yang dihasilkan dengan mengalikan potensi hidup yang hilang akibat *breast cancer* dengan *value of a year of life* adalah \$2.850.000, sehingga fungsi *detection cost* sesuai (5) menjadi:

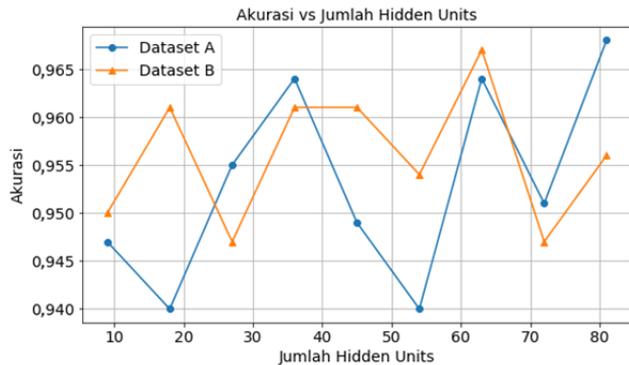
$$\text{Detection Cost} = (-1 * (228,35 * \text{FP}) + (2.850.000 * \text{FN})) \quad (6)$$

#### V. IMPLEMENTASI DAN PENGUJIAN

##### A. Penentuan Arsitektur Multi-layer Perceptron

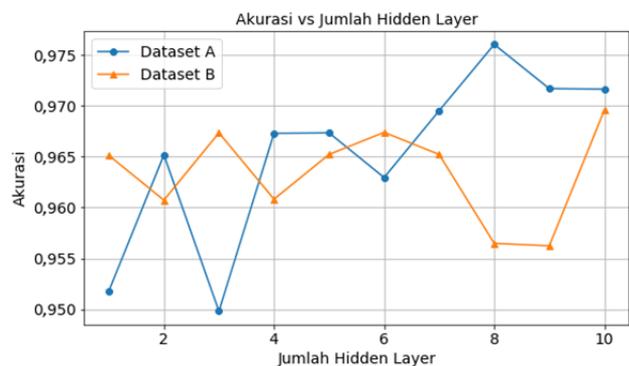
Untuk menemukan arsitektur terbaik pada algoritme MLP, dilakukan dua percobaan untuk menemukan jumlah *hidden layer* dan jumlah *hidden unit* setiap *layer* terbaik, sehingga menghasilkan model yang sesuai dengan kebutuhan. Percobaan pertama dilakukan dengan memvariasikan jumlah *hidden unit* dengan satu *hidden layer*. Jumlah *hidden unit* yang divariasikan merupakan angka kelipatan 9, yaitu 9, 18,

27, 36, 45, 54, 63, 72, dan 81. Sedangkan percobaan 2 dilakukan dengan memvariasikan jumlah *hidden layer*. Untuk jaringan saraf tiruan yang memiliki jumlah *hidden layer* lebih dari satu, perlu disamakan jumlah *trainable weights*-nya (jumlah garis yang menghubungkan setiap *layer*).



Gbr. 2 Akurasi vs jumlah *hidden unit*.

Gbr. 2 merupakan hasil percobaan pertama dengan *Data set* A dan B. Dari gambar tersebut dapat dilihat bahwa jumlah *hidden unit* tidak berbanding lurus dengan akurasi. Untuk *Data set* A, akurasi tertinggi dicapai oleh jumlah *hidden unit* 81, dengan nilai akurasi 0,968 (96,80%). Sedangkan akurasi tertinggi pada *Data set* B dicapai oleh jumlah *hidden unit* 63 dengan nilai akurasi 0,967 (96,70%). Hasil inilah yang kemudian digunakan pada percobaan 2 yang ditunjukkan pada Gbr. 3.



Gbr. 3 Akurasi vs jumlah *hidden layer*.

Berdasarkan Gbr. 3, pada percobaan dengan *Data set* A, jumlah *hidden layer* divariasikan dengan range 1-10. Dengan mempertimbangkan jumlah *hidden unit* yang menghasilkan akurasi terbaik yang didapatkan dari percobaan 1, percobaan 2 untuk *Data set* A dan B menghasilkan persentase akurasi terbaik dengan jumlah *hidden layer* 8 dan 10 dengan persentase akurasi 0,9760 (97,60%) dan 0,9695 (96,95%).

**B. Baseline**

*Dummy Classifier* merupakan *classifier* yang memberikan pengukuran dasar dengan melakukan prediksi dengan menggunakan aturan sederhana. *Classifier* ini berguna sebagai *baseline* sederhana untuk dibandingkan dengan *classifier* lainnya. Fungsi ini memiliki beberapa parameter, salah satunya adalah strategi yang digunakan untuk menghasilkan

prediksi. Dalam makalah ini digunakan strategi *most\_frequent*. *Most\_frequent* merupakan strategi yang selalu memberikan hasil prediksi berdasarkan label yang paling banyak dalam data uji. Hasil *dummy classifier* dengan menggunakan *Data set* A dan *Data set* B disajikan pada Tabel II.

TABEL II  
HASIL DUMMY CLASSIFIER

No.	Data set	Akurasi
1.	A	70,99%
2.	B	67,25%

**C. Mengatur Parameter**

Proses mengatur parameter menghasilkan *estimator* terbaik dengan tolok ukur berupa *score* yang dapat diubah sesuai dengan kebutuhan. Dalam hal ini, digunakan *cost false positive* dan *cost false negative* yang sudah dijelaskan sebelumnya. Sebelum mengatur parameter dengan metode *grid search*, data terlebih dahulu dibagi menjadi data latih dan data uji. *Grid search* melakukan pencarian secara menyeluruh (*exhaustive*) terhadap parameter spesifik. Daftar *hyperparameter* untuk setiap algoritme diperlihatkan pada Tabel III.

TABEL III  
DAFTAR HYPERPARAMETERS

Classifier	Hyper-parameters	Nilai
SVM	c	$[2^{-15} 2^{-13} 2^{-11} 2^{-9} 2^{-7} 2^{-5} 2^{-3} 2^{-1} 2^1 2^3]$
	gamma	$[2^{-5} 2^{-3} 2^{-1} 2^1 2^3 2^5 2^7 2^9 2^{11} 2^{13} 2^{15}]$
XGB	n_estimator	[10 20 30 50 100]
	subsample	[0,5 0,6 0,7 0,8 0,9 1,]
	colsample_bytree	[0,5 0,6 0,7 0,8 0,9 1,]
	lambda	[0, 0,001 0,005 0,01 0,05 0,1 0,5 1, 5, 10,]
	alpha	[0, 0,001 0,005 0,01 0,05 0,1 0,5 1, 5 10,]
MLP	ephocs	[10 20 50 100 200 500]
	class_weight	[2 4 8 16 32 64]

TABEL IV  
HASIL GRID SEARCH

Classifier	Hyper-parameters	Nilai	
		Data set A	Data set B
SVM	c	$2^{-1}$	$2^{-1}$
	gamma	$2^{-3}$	$2^{-3}$
XGB	n_estimator	10	10
	subsample	1,0	0,8
	colsample_bytree	0,5	0,6
	lambda	0,0	0,5
	alpha	0,0	1,0
MLP	ephocs	16	8
	class_weight	50	20

Hasil dari proses *grid search* untuk ketiga algoritme ditunjukkan pada Tabel IV. Parameter terbaik didapatkan

berdasarkan *score cost* terkecil yang didapatkan pada proses *grid search*.

#### D. Validasi

Pada tahap validasi, dilakukan evaluasi pada hasil *cross validation* data latih. Setelah diperoleh parameter terbaik dari setiap algoritme, dilakukan evaluasi terhadap parameter tersebut dengan menggunakan *cross validation* pada data latih. Hasil *score cost* sesuai (6) yang didapatkan dari hasil *cross validation* seharusnya tidak berbeda terlalu jauh dengan hasil *grid search*.

Pada algoritme SVM dan XGBoost, *score cost* hasil *grid search* dan hasil *cross validation* tidak memiliki perbedaan yang terlalu signifikan. Namun, pada algoritme MLP, *score cost* hasil *grid search* dan hasil *cross validation* memiliki perbedaan yang sangat signifikan. Hal ini terjadi karena pada algoritme MLP, proses inialisasi bobot dilakukan secara acak. Oleh karena itu, digunakan fungsi yang disediakan oleh *library keras* untuk menyimpan inialisasi bobot, sehingga dapat dipastikan proses inialisasi bobot sama untuk setiap iterasi.

Dalam makalah ini digunakan inialisasi bobot yang sebelumnya sudah disimpan sebelum melakukan *grid search* dan *cross validation*, dengan harapan dapat memperkecil perbedaan *score cost* yang dihasilkan. Dilakukan sepuluh kali *grid search* dan *cross validation* kemudian dihitung rata-rata perbedaan yang dihasilkan. Namun, rata-rata perbedaan *score cost* hasil *grid search* dan hasil *cross validation* masih terlampaui tinggi, yaitu \$777.179,117. Hal ini dikarenakan dalam proses *grid search*, inialisasi bobot akan tetap berubah untuk setiap *fold cross validation* yang dilakukan. Oleh sebab itu, diimplementasikan fungsi untuk melakukan *grid search* sederhana.

Dengan mengimplementasikan fungsi *grid search*, parameter terikat dan parameter bebas yang akan dilihat pengaruhnya terhadap model dapat dikontrol. Hasil *score cost* dari fungsi *grid search* sederhana yang diimplementasikan yaitu \$251,185 untuk parameter epochs : 200 dan class weight : 8 untuk *Data set A*, serta \$2.580,355 untuk parameter epochs : 50 dan class weight : 64 untuk *Data set B*. Hasil *cross validation final* dapat dilihat pada Tabel V.

Berdasarkan Tabel V, tampak bahwa hampir semua *metrics* untuk ketiga algoritme menghasilkan persentase di atas 90%. Namun, untuk algoritme MLP pada *Data set B*, kekhususan yang dihasilkan paling kecil jika dibandingkan dengan kekhususan algoritme lainnya, yaitu 74,02%. Karena nilai kekhususan dapat dikatakan cukup rendah, hal ini berarti jumlah *false positive* tinggi. Selain itu, *Data set B* merupakan *data set* yang dikenai proses *eliminate missing value*, sehingga jumlah data latih yang digunakan menjadi semakin sedikit. Kekhususan yang rendah diakibatkan oleh pembebanan yang dilakukan parameter *class weight* terhadap kelas *malignant* untuk meminimalkan jumlah *False Negative*. Hal ini didukung dengan percobaan yang dilakukan tanpa menyertakan parameter *class weight*, yang menghasilkan nilai kekhususan dan akurasi yang tinggi, yaitu 0,9691 ( $\pm 0,0294$ ) dan 0,9715

( $\pm 0,0140$ ). Namun, nilai sensitivitas yang dihasilkan adalah 0,8673 ( $\pm 0,2933$ ).

TABEL V  
HASIL CROSS VALIDATION

Metrics	Data set A			Data set B		
	SVM	XGB	MLP	SVM	XGB	MLP
Akurasi	0,944 ( $\pm 0,074$ )	<b>0,976</b> ( $\pm 0,0232$ )	0,963 ( $\pm 0,025$ )	0,949 ( $\pm 0,068$ )	<b>0,978</b> ( $\pm 0,034$ )	0,804 ( $\pm 0,254$ )
Spesifisitas	0,913 9 ( $\pm 0,116$ )	<b>0,970</b> ( $\pm 0,0359$ )	0,966 ( $\pm 0,029$ )	0,921 ( $\pm 0,106$ )	<b>0,969</b> ( $\pm 0,047$ )	0,740 ( $\pm 0,374$ )
Sensitivitas	<b>1,000</b> ( $\pm 0,000$ )	0,987 ( $\pm 0,0486$ )	0,971 ( $\pm 0,0254$ )	<b>1,000</b> ( $\pm 0,000$ )	0,993 ( $\pm 0,0375$ )	<b>1,000</b> ( $\pm 0,000$ )
Cost (\$)	593,7 ( $\pm 796,28$ )	570,2 05,5 ( $\pm 2.279.908,6$ )	<b>251,185</b> ( $\pm 238,4$ )	<b>525,2</b> ( $\pm 708,9$ )	285,2 05,5 ( $\pm 1.710.015,2$ )	2.580 ,3 ( $\pm 2.431,7$ )

#### E. Pengujian

Setelah melakukan validasi, proses selanjutnya adalah pengujian. Proses ini dilakukan untuk menguji model yang diperoleh menggunakan data uji. Hasil yang didapatkan berupa persentase akurasi, kekhususan, dan sensitivitas, sebagai hasil akhir yang dapat dilihat pada Tabel VI.

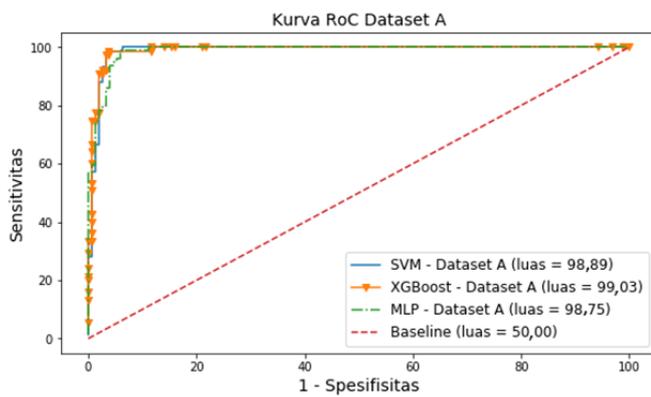
TABEL VI  
HASIL PENGUJIAN

Metrics	Data set A			Data set B		
	SVM	XGB	MLP	SVM	XGB	MLP
Akurasi	0,952	<b>0,961</b>	0,952	0,946	<b>0,977</b>	0,955
Spesifisitas	0,935	<b>0,967</b>	0,934	0,920	<b>0,980</b>	0,943
Sensitivitas	0,986	0,946	<b>0,987</b>	<b>1,000</b>	0,973	0,976

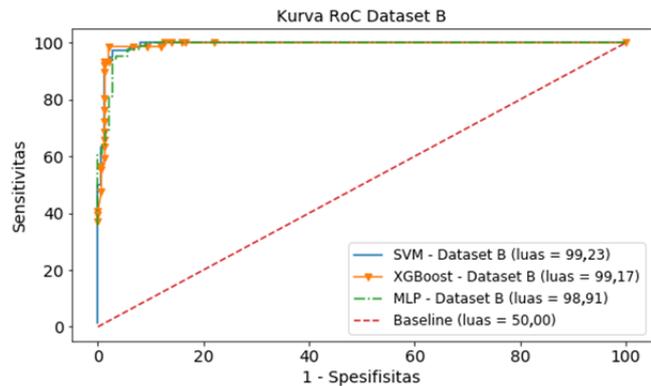
Dari ketiga *classifier*, SVM dan MLP menghasilkan sensitivitas yang relatif lebih tinggi dibanding persentase kekhususannya. Hal ini dikarenakan fungsi *scoring* yang digunakan memang dibuat untuk meminimalkan jumlah *False Negative*.

1) *Kurva RoC*: Berdasarkan Gbr. 4, baik algoritme SVM, XGBoost, maupun MLP memiliki nilai AUC di atas 98. Hal ini menunjukkan ketiga *classifier* untuk *Data set A* menghasilkan kinerja yang sangat baik.

Serupa dengan kurva RoC pada *Data set A*, berdasarkan Gbr. 5, kurva RoC pada *Data set B* juga memiliki nilai AUC diatas 98. Hal ini menunjukkan ketiga *classifier* untuk *Data set A* dan *Data set B* menghasilkan kinerja yang sangat baik. Semakin dekat kurva RoC dengan sumbu Y (0,100), maka semakin baik model dalam membedakan jenis tumor *benign* dan *malignant*. Nilai AUC tertinggi secara keseluruhan dihasilkan oleh model SVM dengan *Data set B* dengan nilai AUC 99,23.



Gbr. 4 Kurva RoC data set A.



Gbr. 5 Kurva RoC data set B.

2) *Confusion Matrix*: Selain menggunakan kurva RoC, juga digunakan *confusion matrix* untuk mengukur kinerja *classifier*. *Confusion matrix* hasil proses pengujian diperlihatkan pada Tabel VII.

TABEL VII  
HASIL *CONFUSION MATRIX* DATA SET A DAN DATA SET B

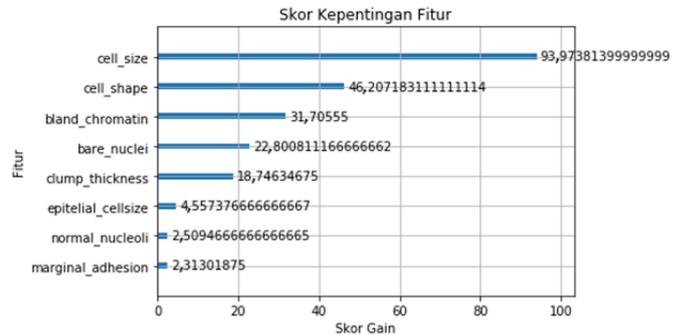
Algoritme	Data set	TP	TN	FP	FN	Cost
SVM	A	74	146	10	1	2.852.283,5
	B	76	138	12	0	2.740,20
XGBoost	A	71	151	5	4	11.401.141,75
	B	74	147	3	2	5.700.685,05
MLP	A	78	142	10	1	2.852.283,5
	B	82	134	8	2	5.701.826,80

Berdasarkan (5), dapat dihitung *cost* yang dihasilkan oleh setiap *classifier* berdasarkan jumlah *False Negative* dan *False Positive* yang dihasilkan. *Cost* terendah dihasilkan oleh model SVM dengan *Data set B*, yaitu \$2.740,20, sedangkan *cost* tertinggi dihasilkan oleh model XGBoost *Data set B*, yaitu \$11.401.141,75.

F. *Feature Importance* dan *Tree XGBoost*

Salah satu keuntungan menggunakan *gradient descent* adalah setelah *boosted trees* dibangun, *importance score* untuk setiap atribut mudah diambil. *Importance score* ini dapat digunakan untuk menunjukkan besarnya kegunaan atau pengaruh setiap fitur dalam pembuatan *boosted tree* dalam model. *Importance score* dihitung secara eksplisit untuk setiap

fitur dalam *data set*, sehingga memungkinkan fitur diberi peringkat dan dibandingkan satu sama lain. Digunakan fungsi `plot_importance()` yang telah disediakan oleh *library xgboost* untuk menampilkan *feature importance* setiap fitur.



Gbr. 6 *Feature importance* XGBoost.

Salah satu parameter fungsi `plot_importance()` adalah `importance_type`, yaitu parameter yang menentukan cara *importance score* dihitung. Di sini digunakan `gain` yang merupakan rata-rata *gain of splits* yang menggunakan fitur tersebut dalam *tree*. Hasil *feature importance* setiap atribut untuk *Data set B* ditunjukkan pada Gbr. 6.

Gbr. 6 menunjukkan bahwa fitur *cell\_size* merupakan fitur yang memiliki *gain* tertinggi dengan nilai 93,973. Hal ini menunjukkan bahwa fitur *cell\_size* memiliki pengaruh yang cukup tinggi terhadap model dalam melakukan klasifikasi jenis tumor payudara. Sedangkan atribut *marginal\_adhesion* merupakan fitur yang memiliki *gain* terkecil dengan nilai 2,31. Hal ini berarti atribut *marginal\_adhesion* kurang berpengaruh terhadap model dalam melakukan klasifikasi jenis tumor payudara.

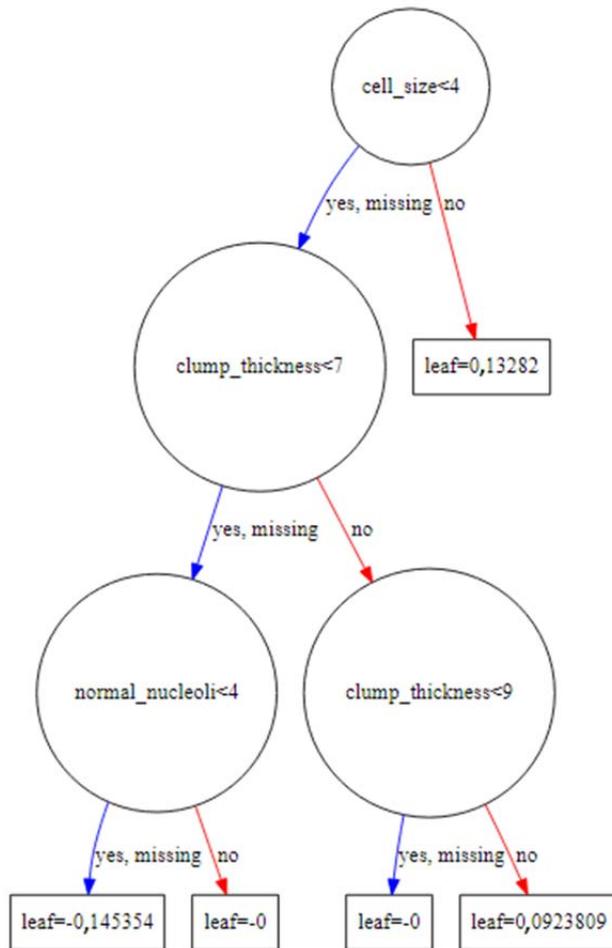
Dalam algoritme XGBoost, jumlah *tree* dalam model ditentukan oleh parameter `n_estimator`. Data latih dibagi menjadi sejumlah subset sesuai dengan jumlah `n_estimator` yang ditentukan. Setiap subset data menghasilkan satu *tree*, sehingga pada akhirnya *tree* yang terbentuk dalam model adalah sejumlah `n_estimator`.

Dalam makalah ini, jumlah `n_estimator` terbaik yang didapatkan dari hasil *hyperparameter tuning* adalah sepuluh untuk *Data set A* dan *Data set B*. Hal ini berarti data latih dibagi menjadi sepuluh subset berbeda yang akan menghasilkan sepuluh *tree* berbeda dalam model. Probabilitas kelas didapatkan dengan *logistics transformation* dari jumlah *leaf score* setiap *tree* dalam model.

Selain menyediakan fungsi untuk memetakan *feature importance*, *library xgboost* juga menyediakan fungsi untuk memetakan setiap *tree* yang dihasilkan dalam model. Hasil salah satu *tree* yang dihasilkan dalam model pada *Data set B* diperlihatkan pada Gbr. 7.

Setiap *tree* mungkin memiliki struktur yang berbeda. Hal ini dikarenakan setiap *tree* dibuat berdasarkan subset yang berbeda. *Rootnode* pada *tree* indeks ke-4 berbeda dengan *rootnode* pada *tree* indeks ke-0, dan seterusnya. Pemilihan *rootnode* setiap atribut didasarkan pada perhitungan *score*

gain. Berdasarkan Gbr. 7, pada *tree* indeks ke-4 fitur *cell\_size* memiliki *score gain* tertinggi, sehingga menjadi *rootnode*.



Gbr. 7 *Tree* indeks ke-4.

Setiap *tree* yang terbentuk tidak hanya mungkin memiliki struktur yang berbeda, tetapi juga memiliki *depth* (kedalaman) berbeda. Namun, setiap *tree* yang terbentuk memiliki *maximal depth* sejumlah nilai pada parameter *max\_depth*. Dalam makalah ini, *max\_depth* yang ditetapkan adalah 3. Parameter *max\_depth* merupakan parameter yang berfungsi untuk mengendalikan kompleksitas model untuk menghindari *overfitting*.

## VI. KESIMPULAN DAN SARAN

### A. Kesimpulan

Berdasarkan hasil dan analisis hasil uji coba yang telah dijelaskan, dapat disimpulkan bahwa berdasarkan kurva RoC secara keseluruhan, model terbaik dihasilkan oleh algoritme SVM pada *Data set B* dengan nilai AUC 99,23. Sedangkan berdasarkan *confusion matrix* secara keseluruhan, *score cost* terendah dihasilkan oleh algoritme SVM pada *Data set B*, yaitu \$2.740,20, sehingga dari ketiga algoritme ini, berdasarkan nilai AUC dan *score cost confusion matrix*, algoritme SVM merupakan algoritme yang paling tepat digunakan untuk kasus klasifikasi jenis tumor payudara

dengan jumlah data yang tersedia. Selain itu, kesimpulan lain yang dapat disampaikan adalah perbedaan perlakuan yang dilakukan terhadap data set A dan B tidak menghasilkan perubahan yang signifikan dari jumlah FP dan FN, walaupun menghasilkan *score cost* yang cukup jauh perbedaannya.

Kesimpulan lainnya, algoritme MLP memberikan hasil yang baik berdasarkan *confusion matrix* maupun berdasarkan kurva RoC. Namun, kompleksitas algoritme MLP serta terbatasnya data yang tersedia menyebabkan sulitnya menemukan nilai parameter yang sesuai untuk membangun model terbaik. Oleh karena itu, walaupun pada akhirnya algoritme ini memiliki hasil yang sama maupun memiliki perbedaan yang tidak signifikan dengan kedua algoritme lainnya, penggunaan algoritme ini tidak disarankan. Hal ini didasarkan pada prinsip Occam's *Razor* mengenai kesederhanaan, bahwa jika ada dua model yang memiliki hasil yang sama maupun memiliki perbedaan yang tidak signifikan, maka model yang dipilih adalah model yang lebih sederhana. Dengan kata lain, jika algoritme SVM/XGBoost dengan algoritme MLP memiliki hasil yang hampir sama, maka algoritme yang dipilih adalah algoritme SVM/XGBoost.

### B. Saran

Untuk mendapatkan hasil yang optimal, pada penelitian selanjutnya disarankan dari setiap algoritme perlu dilakukan *grid search*. Namun, proses *grid search* memerlukan spesifikasi sumber daya yang tinggi agar waktu proses *grid search* yang dibutuhkan dapat diminimalkan. Salah satu pilihan yang dapat digunakan adalah dengan menggunakan *cluster computing*. Kemudian, untuk mengatasi keterbatasan data, penelitian selanjutnya diharapkan tidak hanya menggunakan data sekunder yang tersedia dalam *machine learning repository*, tetapi menggunakan data primer, yaitu data yang diperoleh langsung dari sumber asli dengan melakukan penelitian bersama pihak terkait seperti *cytopatologist*.

Dibutuhkan penelitian lebih lanjut terkait fungsi *cost* yang digunakan sebagai tolok ukur pemilihan model terbaik, terutama karena ditemukan perbedaan yang signifikan pada *score cost Data set A* dan B.

## UCAPAN TERIMA KASIH

Terima kasih disampaikan kepada seluruh dosen Program Studi Teknik Informatika Universitas Al Azhar Indonesia, keluarga, dan teman-teman penulis yang selalu memberi motivasi dan dukungan selama penelitian berlangsung.

## REFERENSI

- [1] Yayasan Kanker Indonesia. (2016) Yayasan Kanker Indonesia. [Online]. <http://yayasankankerindonesia.org/tentang-kanker/>
- [2] World Health Organization. (2008) Global Burden of Disease. [Online]. <http://www.who.int>
- [3] American Cancer Society, *Global Cancer Facts and Figures*, 3rd ed. Atlanta: American Cancer Society, 2015. [Online]. <https://www.cancer.org>
- [4] Kementerian Kesehatan RI. (2015) Pusat Data dan Informasi Kementerian Kesehatan RI. [Online]. <http://www.depkes.go.id>
- [5] G. J. Miao, K. H. Miao, and J. H. Miao, "Neural Pattern Recognition

- Model for Breast Cancer Diagnosis," *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Bioinformatics (JBIO)*, pp. 1-8, August 2012.
- [6] S. W. Fletcher, W. Black, R. Harris, B. K. Rimer, and S. Saphiro, "Report of The International Workshop on Screening Breast Cancer," *Journal of National Cancer Institute*, vol. 85, pp. 1644-1656, 1993.
- [7] H. Al-Khawari, R. Athyal, A. Kovacs, M. Al-Saleh, and J. P. Madda, "Accuracy of the Fischer Scoring System and Breast Imaging Reporting and Data System Identification of Malignant Breast Lesions," *Annals of Saudi Medicine*, vol. 29, no. 4, pp. 280-287, August 2009.
- [8] W. L. Street, W. N. Street, and W. H. Wolberg, "Breast Cancer Diagnosis and Prognosis via Linear Programming," *Mathematical Programming Technical Report*, pp. 1-10, December 1994.
- [9] Z. A. Prasetyo, "Uji Diagnostik FNAB (Fine Needle Aspiration Biopsy) dibandingkan dengan Biopsy Patologi Anatomi dalam Mendiagnosis Karsinoma Tiroid," Universitas Diponegoro, Semarang, Karya Tulis Ilmiah 2012. [Online]. epints.undip.ac.id
- [10] C. P. Utomo, A. Kardiana, and R. Yuliwulandari, "Breast Cancer Diagnosis using Artificial Neural Networks with Extreme learning Techniques," *International Journal of Advanced Research in Artificial Intelligence*, 2014.
- [11] I. Muhic, "Fuzzy Analysis of Breast Cancer Disease Using Fuzzy C-Means and Pattern recognition," *Southeast Europe Journal of Soft Computing*, pp. 50-55, 2013.
- [12] T. Kiyan and T. Yildirim, "Breast Cancer Diagnosis Using Statistical Neural Networks," *Journal of Electrical & Electronics Engineering*, p. Vol.4, 2004.
- [13] A. Aloraini, H. Mousanni, H. A. Moatassime, and N. Thomas, "Different Machine Learning Algorithms for Breast Cancer Diagnosis," *International Journal of Artificial Intelligence & Applications (IJAIA)*, 2012.
- [14] K. Sivakami and Nadar Saraswathi, "Mining Big Data: Breast Cancer Prediction using DT - SVM Hybrid Model," *International Journal of Scientific Engineering and Applied Science (IJSEAS)*, p. Vol.1, 2015.
- [15] K. Menaka and S. Karpagavalli, "Breast Cancer Classification using Support Vector Machine and Genetic Programming," *International Journal of Innovative Research in Computer and Communication Engineering*, p. Vol.1, 2013.
- [16] M. A. B. Ahmad, "Mining Health Data for Breast Cancer Diagnosis Using Machine Learning," University of Canberra, Canberra, PhD Thesis December 2013. [Online]. <http://www.canberra.edu.au>
- [17] V. Kumar, R. S. Cotran, and S. L. Robbins, *Buku Ajar Patologi*, 7th ed. Jakarta, Indonesia: EGC, 2007.
- [18] D. K. Roy and L. K. Sharma, "Genetic k-Means Clustering Algorithm for Mixed Numeric and Categorical Data Sets," *International Journal of Artificial Intelligence & Applications*, vol. 1, no. 2, pp. 23-38, 2010.
- [19] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2nd ed. San Fransisco, United States of America: Morgan Kaufmann, 2006.
- [20] N. Cristianini and J. S. Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. United Kingdom: Cambridge University Press, 2000.
- [21] M. J. Zaki and W. M. Jr., *Data Mining and Analysis Fundamental Concepts and Algorithms*. New York, USA: Cambridge University Press, 2014.
- [22] C. W. Hsu, C. C. Chang, and C. J. Lin, *A Practical Guide to Support Vector Classification*. Taipei, Taiwan: National Taiwan University, 2003.
- [23] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, USA: Springer, 2006.
- [24] L. Fausett, *Fundamental of Neural Networks: Architectures, Algorithms, and Applications*. New York, USA: Prentice-Hall, 1994.
- [25] X. Jin et al., "Deep Learning with S-Shaped Rectified Linear Activation Units," *CoRR*, vol. abs/1512.07030, December 2015.
- [26] S. N. Sivanandam, S. Sumathi, and S. N. Deepa, *Introduction to Neural Networks using Matlab 6.0*. New Delhi, India: Tata Mcgraw-Hill, December 2006.
- [27] J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting," *Annals of Statistics*, vol. 38, no. 4, pp. 367-378, 2001.
- [28] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *KDD'16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, California, March 2016, pp. 785-794.
- [29] F. Gorunescu, *Data Mining Concepts, Models and Techniques*, 12th ed. Berlin, Germany: Springer, 2011.
- [30] O. L. Mangasarian, "Cancer Diagnosis via Linear Programming," *SIAM News*, vol. 23, no. 5, pp. 1-18, 1990.
- [31] K. Hukkinen, L. Kivisaari, P. S. Heikkila, K. V. Smitten, and M. Leidenius, "Unsuccessful Preoperative Biopsies, Fine Needle Asspiration Cytology or Core Needle Biopsy, Leads to Increased Costs in The Diagnostic Workup in Breast Cancer," *Acta Oncologica*, vol. 477, no. 6, pp. 1037-1045, 2008.
- [32] K. R. Yabroff, C. J. Bradley, A. B. Mariotto, M. L. Brown, and E. J. Feuer, "Estimates and Projections of Value of Life Lost From Cancer Death in the United States," *Journal of the National Cancer Institute (JNCI)*, vol. 100, no. 24, pp. 1755-1762, 2008.
- [33] M. J. Yaffe, R. Jong, E. D. Pisano, K. I. Pritchard, and R. A. Smith, "Earlier Detection and Diagnosis of Breast Cancer," Canadian Breast Cancer Foundation Ontario, Toronto, Recommendation and Scientific Review from It's About Time! A Consensus Conference.