**Building of Informatics, Technology and Science (BITS)**
Volume 4, No 4, Maret 2023 Page: 1967−1974
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i4.3141

# Undersampling dan K-Fold Random Forest Untuk Klasifikasi Kelas Tidak Seimbang

**Laila Qadrini**

FMIPA, Program Studi Statistika, Universitas Sulawesi Barat, Majene, Indonesia
Email: laila.qadrini@unsulbar.ac.id
Email Penulis Korespondensi: laila.qadrini@unsulbar.ac.id
Submitted: **13/02/2023**; Accepted: **31/03/2023**; Published**: 31/03/2023**

**Abstrak**−Klasifikasi pada Data Mining adalah sebuah proses untuk menemukan sebuah model yang menjelaskan dan membedakan kelas data dengan tujuan memperkirakan kelas dari suatu objek yang kelasnya tidak diketahui. Klasifikasi dapat diterapkan dalam berbagai aspek sehingga seiring berjalannya waktu algoritma klasifikasi cukup banyak dikembangkan, namun terdapat permasalahan yang sering ditemui dalam klasifikasi yaitu masalah ketidakseimbangan kelas. Masalah kelas tidak seimbang memiliki efek buruk pada ketepatan prediksi data. Ada kenyataannya, sebagian besar dataset klasifikasi tidak memiliki jumlah yang sama persis pada setiap kelasnya. Namun ketidak seimbangan kelas tidak menjadi masalah ketika perbandingan antar kelasnya tidak berbeda jauh. Ketidakseimbangan kelas dapat mengakibatkan prediksi model yang dihasilkan akan cenderung kepada kelompok mayoritas sehingga kontribusi kelas minoritas terhadap model kecil. Salah satu algoritma yang kerap digunakan untuk menangani kelas tak seimbang adalah algoritma Resampling. Tujuan Penelitian ini untuk menerapkan Algoritma Resampling Undersampling Random Forest dan Random Forest K-Fold, Undersampling pada dataset Breast Cancer Diagnostic dari UCI Machine Learning. Undersampling dipilih karena menghasilkan Akurasi yang lebih baik daripada Oversampling. Hasil Akurasi Recall untuk Algoritma Random Forest K-Fold 10 sebesar 83% dan untuk Recall Undersampling Random Forest sebesar 65%.

**Kata Kunci:** An Imbalanced Class; Undersampling; Random Forest; K-Fold

**Abstract**−Classification in Data Mining is a process of finding a model that explains and differentiates data classes to estimate the category of an object whose type is unknown. Classification can be applied in various aspects, so quite a lot of classification algorithms have been developed over time. Still, classification often encounters some problems, namely the class imbalance problem. P The class unbalanced problem hurts the predictive Presisi of the data. Most classification datasets have a different number of classes. However, the class imbalance is acceptable when comparing similar types. Class imbalance can cause the predictions of the resulting model to tend to the majority group so that the contribution of the minority class to the model is small. One of the algorithms often used to handle unbalanced types is the resampling algorithm. This research aims to apply the Resampling Undersampling Random Forest and Random Forest K-Fold Undersampling Algorithms to the Breast Cancer Diagnostic dataset from UCI Machine Learning. Undersampling was chosen because it produces better accuracy than oversampling. Recall accuracy for the K-Fold 10 Random Forest Algorithm is 83%, and for Recall Undersampling Random Forest is 65%.

**Keywords**: An Imbalanced Class; Undersampling; Random Forest; K-Fold

## 1. PENDAHULUAN

Klasifikasi pada Data Mining adalah sebuah proses untuk menemukan sebuah model yang menjelaskan dan membedakan kelas data dengan tujuan memperkirakan kelas dari suatu objek yang kelasnya tidak diketahui [1]. Klasifikasi dapat diterapkan dalam berbagai aspek sehingga seiring berjalannya waktu, algoritma klasifikasi cukup banyak dikembangkan, namun terdapat permasalahan yang sering ditemui dalam klasifikasi yaitu masalah ketidakseimbangan kelas[2]. Masalah kelas tidak seimbang memiliki efek buruk pada ketepatan prediksi data. Penelitian tentang kelas tidak seimbang sudah menjadi salah satu tantangan dalam performa banyak algoritma klasifikasi [3]. Kelas tidak seimbang adalah suatu kondisi dimana terdapat sejumlah data yang jumlah kelasnya tidak seimbang atau terdapat perbedaan yang signifikan terhadap masing-masing jumlah kelas [4]. Dalam kenyataannya, sebagian besar dataset klasifikasi tidak memiliki jumlah yang sama persis pada setiap kelasnya. Namun tidak seimbang tidak menjadi masalah ketika perbandingan antar kelasnya tidak berbeda jauh. Tidak seimbang menjadi masalah ketika perbandingannya sangat besar, sehingga dapat memengaruhi hasil evaluasi [5]. Sehingga akan memberikan dampak yang kurang optimal terhadap hasil klasifikasi [6]. Masalah seperti ini ditemukan hampir disemua dataset. Ketidakseimbangan kelas dapat menyebabkan masalah jika tidak ditangani, karena prediksi model yang dihasilkan akan cenderung kepada kelompok mayoritas sehingga kontribusi kelas minoritas terhadap model kecil [7]. Salah satu algoritma yang kerap digunakan untuk menangani kelas tak seimbang adalah algoritma *Resampling* [8]. Teknik *Resampling* dilakukan dengan mencoba menyeimbangkan data asli berdasarkan serangkaian algoritma sampling dengan menyesuaikan jumlah sampel dalam kelas yang berbeda, kemudian melatih data "seimbang" baru dengan mengadopsi algoritma klasifikasi [9]. Teknik *Resampling* dibagi menjadi tiga kategori yaitu: teknik *Oversampling*, *Undersampling* dan *Hybrid*. Pada penelitian ini menerapkan Teknik *Resampling Undersampling*. Algoritma *Undersampling* dipilih karena menghasilkan Akurasi yang cukup baik daripada *Oversampling*. *Undersampling* merupakan algoritma yang berfokus pada kelas mayoritas. Dalam algoritma ini, data kelas mayoritas dikurangi untuk mendapatkan proporsi yang seimbang [10]. Pada umumnya pendekatan pengurangan sampel yang dilakukan adalah secara random ataupun secara informatif. Pendekatan dengan algoritma random *Undersampling* secara acak memilih observasi dari kelas mayoritas untuk dieliminasi hingga data set menjadi seimbang. Di sisi lain,

**Building of Informatics, Technology and Science (BITS)**
Volume 4, No 4, Maret 2023 Page: 1967−1974
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i4.3141

pendekatan informatif *Undersampling* mengikuti kriteria seleksi yang ditentukan sebelumnya untuk menghapus pengamatan dari kelas mayoritas. Dalam banyak aplikasi, lebih penting untuk mengidentifikasi kelas minoritas *daripada* kelas mayoritas [11]. Penelitian sebelumnya yang meneliti tentang *Undersampling* adalah menggunakan *Synthetic Minority Oversampling Technique* (SMOTE), *Resampling Oversampling*, *Undersampling* dan SMOTE menghasilkan Akurasi yang baik dan efektif dalam menangani kelas yang tidak seimbang karena mengurangi *overfitting* [8]. SMOTE dan *Oversampling* dapat meningkatkan jumlah kelas mayoritas sehingga meningkatkan waktu prediksi. Sedangkan *Undersampling* memiliki waktu prediksi yang lebih cepat dibandingkan SMOTE dan *Oversampling* Adapun penelitian yang menghasilkan model klasifikasi yang baik untuk melakukan prediksi kebangkrutan, *Resampling* diterapkan pada data latih agar menghasilkan model klasifikasi yang lebih optimal. Algoritma *Resampling* yang digunakan adalah kombinasi SMOTE dan *Undersampling*. Algoritma klasifikasi yang digunakan untuk prediksi adalah *multilayer perceptron* dan *complement naïve bayes*. Performa prediksi dihitung menggunakan skor *Recall*, ROC AUC, dan PR AUC. Berdasarkan hasil pengujian, penggunaan SMOTE dan *Undersampling* cukup signifikan dalam memperbaiki model klasifikasi pada *multilayer perceptron* [12]. Selanjutnya penelitian proses klasifikasi bakteri E. Coli dilakukan menggunakan tiga model algoritma sebagai perbandingan yaitu algoritma *Oversampling Random* Forest, *Oversampling Naïve Bayes* dan *Oversampling Decision tree* dengan Data latih akan menggunakan K-*Fold Cross Validation* dengan nilai $k$ = 10. Algoritma *Random* Forest yang terbaik dengan menghasilkan Akurasi sebesar 84% [13]. Berbagai penelitian sebelumnya menjadi referensi penelitian saat ini, dan adapun penelitian ini memiliki tujuan untuk menerapkan Algoritma *Resampling Undersampling Random* Forest dan *Random* Forest K-*Fold*.untuk klasifikasi dataset *Breast Cancer Diagnostic* dari UCI *Machine Learning* yang kelasnya tidak seimbang, dan membandingkan hasil akurasi Algoritma *Resampling Undersampling Random* Forest dan *Random* Forest K-*Fold*.

## 2. METODOLOGI PENELITIAN

### 2.1 *Undersampling*

Mengatasi ketidakseimbangan kelas, beberapa algoritma dapat dibagi menjadi tiga kategori. Pertama yaitu dengan teknik tingkat data yang berusaha menyeimbangkan distribusi data dengan algoritma *Oversampling* dan *Undersampling*. Kedua yaitu pendekatan tingkat algoritma yaitu memodifikasi algoritma yang ada untuk memperhitungkan arti dari kelas minor atau dengan pengembangkan algoritma baru. Ketiga yaitu dengan mengkombinasikan pendekatan algoritma dan pendekatan level data [14]. *Undersampling* yaitu menghasilkan sub sampel acak dari instance kelas mayoritas [15]. *Undersampling* merupakan algoritma sampling secara acak memilih sampel di kelas mayoritas dan menambahkannya ke kelas minoritas, membentuk sebuah dataset pelatihan baru [16].

### 2.2 *Random* Forest K-*Fold*

*Random* Forest adalah classifer yang termasuk dalam golongan ensemble. Cara kerja algoritma *Random* Forest sebagai berikut: anggap setiap *Classifier* dalam ensemble berbentuk sebuah decision tree (pohon keputusan) maka dapat dikatakan bahwa kumpulan *Classifier* tersebut adalah Forest (hutan). Dalam Forest tersebut dibuat banyak decision tree, kemudian anggap sebuah test data dimasukan kedalam Forest. Setiap decision tree dalam Forest akan memberi keputusan dan keputusan akhir (keputusan dari Forest) adalah merupakan keputusan mayoritas dari setiap *decision tree* dalam Forest [17]. Untuk memberikan pemahaman mengenai cara kerja *Random* Forest maka diberikan analogi berikut: anggap perwakilan pemerintah bertanya kepada 300 orang anggota Dewan Perwakilan Rakyat (DPR) mengenai apakah ibukota negara harus pindah ke Kalimantan Timur atau tetap di Jakarta. Setiap anggota DPR kemudian memberikan keputusannya dan keputusan-keputusan tersebut bisa saja sama atau berbeda. Keputusan akhir dari DPR adalah merupakan keputusan mayoritas dari anggota DPR. Pada kasus ini, DPR merepresentasikan Forest, anggota DPR merepresentasikan decision tree dan pertanyaan pemerintah merepresentasikan test data [17]. Algoritma K-*Fold Cross-Validation* mensegmentasi data ke dalam $k$ partisi berukuran sama. Selama proses, salah satu dari partisi dipilih untuk latih, sedangkan sisanya digunakan untuk latih. Prosedur ini diulangi $k$ kali sedemikian sehingga setiap partisi digunakan untuk latih tepat satu kali. *Total error* ditentukan dengan menjumlahkan error untuk semua $k$ proses tersebut.

### 2.3 Akurasi, Presisi, dan *Recall*

Akurasi dapat didefinisikan sebagai tingkat kedekatan antara nilai prediksi dengan nilai aktual. Presisi menunjukan tingkat ketepatan atau ketelitian dalam pengklasifikasian. Sedangkan *Recall* berfungsi untuk mengukur proporsi positif aktual yang benar diidentifikasi. Untuk mengukur Akurasi, Presisi, dan *Recall* biasanya digunakan *Confusion Matrix*. *Confusion Matrix* adalah alat ukur berbentuk matrix yang digunakan untuk mendapatkan jumlah ketepatan klasifikasi terhadap kelas dengan algoritma yang dipakai [18]. Berikut akan disajikan bentuk *Confusion Matrix* pada Tabel 1.

**Tabel 1**. Bentuk *Confusion Matrix* Dua Kelas

| **Confusion Matrix** | **Nilai Aktual** |
|---|---|

**Building of Informatics, Technology and Science (BITS)**
Volume 4, No 4, Maret 2023 Page: 1967−1974
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i4.3141

| | | TRUE | FALSE |
|---|---|---|---|
| **Nilai Prediksi** | **TRUE** | TP (True Positive) Correct result | FP (False Positive) Unexpected result |
| | **FALSE** | FN (*False Negative*) Missing result | TN (True Negative) Correct absence of result |

Pada Tabel 1 nilai TP (*True Positive*) dan TN (*True Negative*) menunjukan tingkat ketepatan klasifikasi. Umumnya semakin tinggi nilai TP dan TN semakin baik pula tingkat klasifikasi dari Akurasi, Presisi, dan *Recall*. Jika label prediksi keluaran bernilai benar (*true*) dan nilai sebenarnya bernilai salah (*false*) disebut sebagai *False Positive* (FP). Sedangkan jika prediksi label keluaran bernilai salah (*false*) dan nilai sebenarnya bernilai benar (*true*) maka hal ini disebut sebagai *False Negative* (FN) [19]. Berikut formulasi untuk menghitung Akurasi, Presisi, dan *Recall* pada pembentukan model klasifikasi ditunjukan pada Persamaan (1), Persamaan (2), dan Persamaan (3).

$$\text{Akurasi} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \times 100\% \quad (1)$$

$$\text{Presisi} = \frac{\text{TP}}{\text{TP+FP}} \times 100\% \quad (2)$$

$$\textit{Recall} = \frac{\text{TP}}{\text{TP+FN}} \times 100\% \quad (3)$$

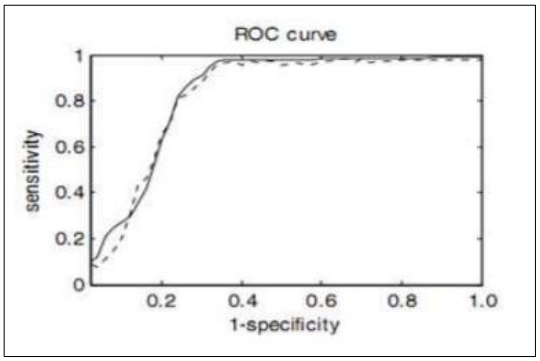
### 2.4 Kurva ROC

Kurva ROC (*Receiver Operating Characteristic*) adalah salah satu alat ukur untuk menilai kemampuan sistem klasifikasi. Kurva ROC sering digunakan untuk mengevaluasi pengklasifikasian karena memiliki kemampuan evaluasi algoritma dengan cukup baik. Kurva ROC merupakan grafik perbandingan antara *Sensitivity* (*True Positive rate* (TPR)) yang diterjemahkan kedalam sumbu vertikal atau sumbu koordinat *y* dengan *Specificity* (*False Positive rate* (FPR)) yang diterjemahkan dalam bentuk kurva [18]. Berikut formulasi dari *Sensitivity* dan *Specificity* dipaparkan pada Persamaan (4), dan Persamaan (5) .

$$\text{Sentivity} = \frac{\text{TP}}{\text{TP+FN}} \times 100\% \quad (4)$$

$$\text{Specificity} = \frac{\text{TP}}{\text{TP+FN}} \times 100\% \quad (5)$$

Kurva ROC dapat digunakan sebagai komparasi beberapa algoritma (*Classifier*) ataupun model *Classifier* yang memiliki perbedaan parameter guna mendapatkan model yang paling baik. Berikut adalah contoh penerapan komparasi performansi dari dua *Classifier* yang berbeda pada Gambar 1.




**Gambar 1**. Perbandingan *Classifier* dengan Kurva ROC

Pada Gambar 1 dapat dilihat bahwa terdapat dua buah *Classifier* yang disimbolkan dengan garis putus-putus dan garis utuh. Jika pada Gambar 1 menunjukan letak koordinat (0,1) hal tersebut mewakili *Sensitivity* dan *Specificity* sebesar 100%. Untuk menghitung dan memastikan *Classifier* mana yang lebih unggul maka digunakan penghitungan AUC (*Area Under Curve*). AUC (*Area Under Curve*) adalah luas area dibawah kurva. Luas dari AUC selalu berada diantara nilai 0 hingga 1. AUC dihitung berdasarkan gabungan luas trapesium titik-titik (*Sensitivity* dan *Specificity*). Pada Gambar 1 memperlihatkan bahwa garis yang utuh memiliki area dibawah kurva yang lebih besar dibandingkan garis yang putus-putus, hal ini berarti bahwa tingkat performansi klasifikasi dari *Classifier* yang dilambangkan dengan garis utuh lebih baik dibandingkan tingkat performansi klasifikasi dari *Classifier* yang dilambangkan dengan garis putus-putus [20]. Berikut adalah standar tabel kategori pengklasifikasian berdasarkan nilai AUC pada Tabel 2.

**Tabel 2.** Kategori Pengklasifikasian Berdasarkan Nilai AUC

| Nilai AUC | Kategori Pengklasifikasian |
|---|---|
| 0.90 - 1.00 | *Excellent* |
| 0.80 - 0.90 | *Good* |
| 0.70 - 0.80 | *Fair* |
| 0.60 - 0.70 | *Poor* |

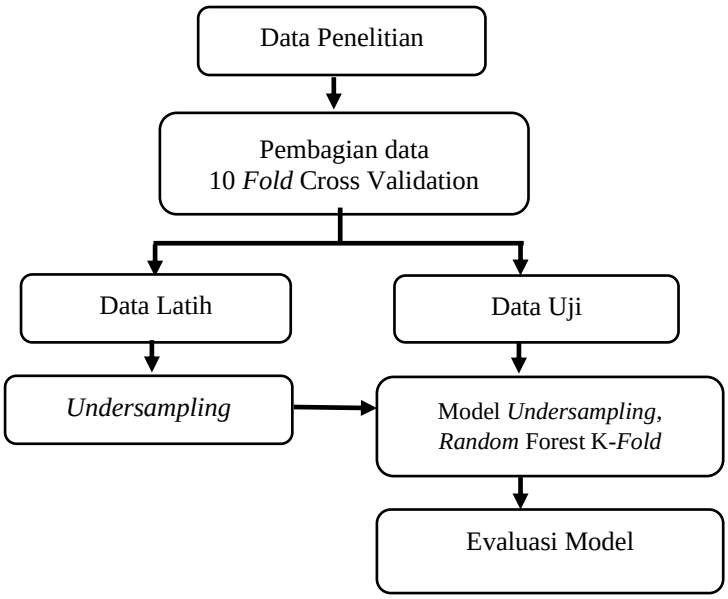
**Building of Informatics, Technology and Science (BITS)**
Volume 4, No 4, Maret 2023 Page: 1967−1974
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i4.3141

| | |
|---|---|
| 0.50 - 0.60 | *Fail* |

Perolehan nilai AUC sesuai hasil pengklasifikasian ada pada Tabel 2, nilai AUC yang semakin mendekati 1 maka hasil klasifikasinya *Excellent* dan nilai AUC yang dibawah 0,5 memberikan hasil klasifikasi buruk.

### 2.5 Tahapan Penelitian

Pada penelitian ini akan dilakukan analisis menggunakan algoritma klasifikasi *Undersampling Random* Forest dan *Random* Forest K-*Fold.* data dihitung dengan algoritma ini sesuai dengan algoritmanya kemudian dicari hasil akurasinya. Dalam tahapan ini akan dilakukan beberapa langkah pengujian data yaitu seperti berikut.



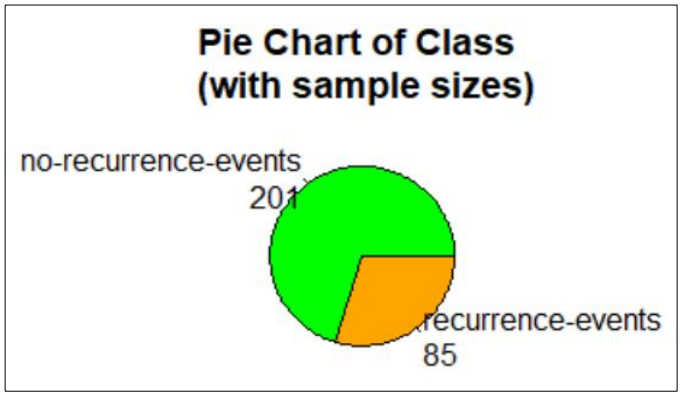

**Gambar 2**. Implementasi *Undersampling* dan K-*Fold Random* Forest

Gambar 2 adalah tahapan penelitian yang dimulai dengan pembagian data dengan *Resampling* 10 *Fold Cross Validation*, data dibagi menjadi 80% data latih dan 20% data uji. Pada data latih diterapkan *Undersampling* dan pada data uji diterapkan *Undersampling Random Forest* K-*Fold*, dalam hal *k* = 10 *Cross Validation*, Hasil klasifikasi diperoleh dan dievaluasi dengan ukuran kebaikan algoritma yaitu *Confussion Matrix*.

## 3. HASIL DAN PEMBAHASAN

### 3.1 Sumber Data

Sumber Data yang digunakan dalam penelitian ini adalah data pada UCI Machine Learning yaitu Breast Cancer Diagnostic (BDC). Variabel yang digunakan adalah numerik dan kategorik. Total data pada penelitian ini ada 286, terdiri dari 201 Kelas Mayoritas yaitu kelas *No Recurrence*, dan 85 untuk kelas minoritas *Recurrence*. Data ini bisa diakses melalui https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29.




**Gambar 3**. Visualisasi Kelas Data

Gambar 3 memvisualisasikan dataset kelas *No Recurrance* yang merupakan kelas pasien tidak terdiagnosis kanker payudara, terdapat 70% prevalensi pasien *No Recurrance*, dan kelas *Recurrance* adalah kelas pasien terdiagnosis kanker payudara, dan terdapat 30% preavalensi pasien *Recurrance*. Kelas tidak seimbang dapat terdeteksi

**Building of Informatics, Technology and Science (BITS)**
Volume 4, No 4, Maret 2023 Page: 1967−1974
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i4.3141

pada Gambar 3.

### 3.2 Variabel Penelitian

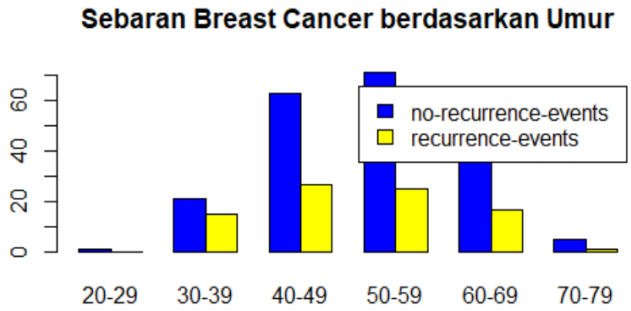
Terdapat 10 variabel pada penelitian ini, 9 atribut dan 1 kelas, Adapun deskripsi variabel sebagai berikut:
*Class: no-recurrence-events, recurrence-events*
*age*: 10-19, 20-29, 30-39, 40-49, 50-59, 60-69, 70-79, 80-89, 90-99.
*menopause: lt40, ge40, premeno*.
*tumor-size*: 0-4, 5-9, 10-14, 15-19, 20-24, 25-29, 30-34, 35-39, 40-44, 45-49, 50-54, 55-59.
*inv-nodes*: 0-2, 3-5, 6-8, 9-11, 12-14, 15-17, 18-20, 21-23, 24-26, 27-29, 30-32, 33-35, 36-39.
*node-caps*: yes, no.
*deg-malig*: 1, 2, 3.
*breast*: *left, right*.
*breast-quad: left-up, left-low, right-up, right-low, central*.
*irradiat: yes, no*.

### 3.3 Eksplorasi dan Analisis Data

Eksplorasi dan Analisis Data (EDA) merupakan bagian dari proses data mining [21]. EDA menjadi sangat penting sebelum melakukan modeling karena dalam tahap ini dibutuhkan pemahaman tentang data dan strukturnya terlebih dahulu. Eksplorasi Data adalah serangkaian pengecekan awal pada data untuk menemukan pola, memeriksa adanya anomali, menguji hipotesis dan untuk memeriksa asumsi dengan bantuan Summary statistik dan representasi grafis. Adapun Tahapan EDA pada penelitian ini adalah Mengecek Dimensi data, Mengecek Struktur data, Membuat Proporsi Tabel Kelas, Menampilkan Visualisasi kelas dengan pie chart, Mendeteksi Outlier, Membuat plot korelasi data, Mengubah variabel character menjadi numerik dan kategorik, Mengecek statistik data

### 3.4 Preprocessing Data

*Preprocessing* data dilakukan setelah proses EDA, prapemrosesan data adalah kegiatan mengeksplor dan memahami data yang akan diolah hingga data tersebut layak untuk melaju ketahap pemodelan, Hal ini merupakan langkah awal pada proses klasifikasi data. Penelitian ini menggunakan algoritma klasifikasi *Random Forest*. Pertama-tama dilakukan *Cleaning* data pada Seleksi Fitur. Seleksi Fitur digunakan untuk menyeleksi data yang rusak/tidak lengkap menggunakan fitur "*Input Missing Value*" dan membuang variabel yang tidak dibutuhkan untuk pemodelan, contoh misalnya *Customer* ID, selanjutnya ada ekstraksi fitur yang mana data begitu selanjutnya hingga didapatkan data set murni yang siap untuk diolah. Mentranformasikan tipe data dan melakukan normalisasi data numerik menentukan bentuk data yang paling tepat. Selanjutnya membagi data latih dan data uji dengan perbandingan 80%:20%. Berikut adalah visualisasi sebaran *Breast Cancer* berdasarkan umur pasien.




**Gambar 4.** Visualisasi Sebaran *Breast Cancer*

Gambar 4 adalah sebaran diagnosis *Breast Cancer* berdasarkan umur, kelompok umur 40-49 Tahun dan kelompok umur 50-59 Tahun adalah yang paling tinggi terdiagnosis kanker payudara, kelompok Umur 20-29 Tahun dan kelompok umur 70-79 Tahun adalah yang paling rendah terdiagnosis kanker payudara.

### 3.4 Pembagian data latih dan data uji

Agar dapat melakukan prediksi, data harus dibagi terlebih dahulu ke dalam data latih dan data uji, Pada penelitian ini keseluruhan data dibagi menjadi 80%:20% untuk 80% data latih dan 20% data uji. Tabel 2 menunjukkan jumlah data latih dan data uji.

**Tabel 1**. Pembagian Data Latih Dan Data Uji

| Jenis Data | Jumlah data | | Total |
|---|---|---|---|
| | Kelas 1 | Kelas 0 | |
| Latih | 160 | 69 | 229 |
| Uji | 41 | 16 | 57 |

**Building of Informatics, Technology and Science (BITS)**
Volume 4, No 4, Maret 2023 Page: 1967−1974
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i4.3141

### 3.5 Hasil Prediksi *Random* Forest

**Tabel 2**. *Confusion Matrix Undersampling Random* Forest

| | *Actual* | |
|---|---|---|
| *Prediction* | *No-Recurrence* | *Recurrence* |
| *No-Recurrence* | 27 | 7 |
| *Recurrence* | 14 | 9 |

**Tabel 3**. *Confusion Matricx Random* Forest K-*Fold* 10

| | *Actual* | |
|---|---|---|
| *Prediction* | *No-Recurrence* | *Recurrence* |
| *No-Recurrence* | 34 | 11 |
| *Recurrence* | 7 | 5 |

True Positive (TP) : kasus dimana pasien diprediksi (Positif) Tidak terkena kanker payudara dan memang benar(True) tidak terkena kanker payudara payudara. Pada Tabel 2 dan 3 di atas sebanyak 69 pasien data uji terdapat 27 pasien dan 34 pasien yang diprediksi tidak terkena kanker payudara dan kenyataanya memang tidak terkena kanker payudara

True Negative (TN) : kasus dimana pasien diprediksi (Negatif) Terkena kanker payudara dan memang benar(True) terkena kanker payudara. Pada Tabel 2 dan 3 di atas sebanyak 69 pasien data uji terdapat 9 pasien dan 5 pasien yang diprediksi terkena kanker payudara dan kenyataanya memang terkena kanker payudara

False Positve (FP) : kasus dimana pasien diprediksi (Positif) Tidak terkena kanker payudara namun kenyataan terkena kanker payudara. Pada Tabel 2 dan 3 di atas sebanyak 69 pasien data uji terdapat 7 pasien dan 11 pasien yang diprediksi tidak terkena kanker payudara dan kenyataanya terkena kanker payudara

False Negatif (FN): kasus dimana pasien diprediksi (Negatif) Terkena kanker payudara dan kenyataanya (Positif) tidak terkena payudara. Pada Tabel 2 di atas sebanyak 69 pasien data uji terdapat 9 pasien dan 7 pasien yang diprediksi terkena kanker payudara dan kenyataanya tidak terkena kanker payudara

### 3.6 Akurasi, Presisi dan *Recall* Algoritma K-*Fold* 10 dan *Undersampling* K-*Fold*

Terdapat hasil Akurasi, Presisi dan *Recall* serta berbagai pengukuran lainnya pada *Confusion Matrix*, berikut adalah rangkuman hasil Akurasi, Presisi dan *Recall* untuk Algoritma K-*Fold* 10 dan *Undersampling* K-*Fold*

**Tabel 4**. Hasil Pengukuran *Random* Forest K *Fold* 10 dan *Undersampling* K-*Fold*

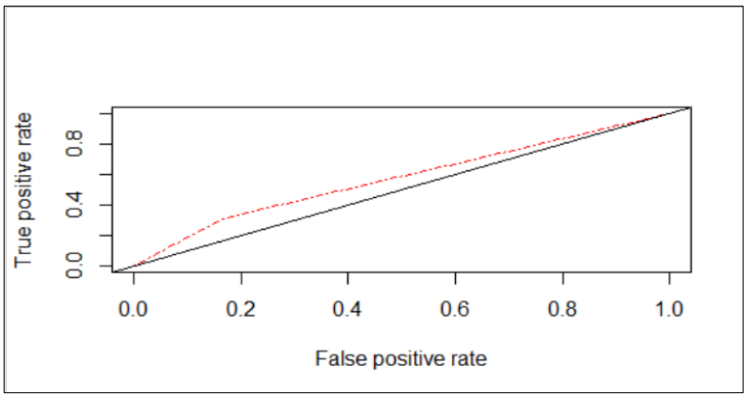
| Pengukuran | K-*Fold* 10 | *Undersampling* |
|---|---|---|
| Akurasi | 0.68 | 0.63 |
| Presisi | 0.31 | 0.56 |
| *Recall* | 0.83 | 0.65 |
| AUC | 0.57 | 0.61 |

Akurasi menjawab pertanyaan “Berapa persen pasien yang benar diprediksi tidak terkena kanker payudara dan terkena kanker payudara dari kesuluruhan pasien”. Akurasi model *Random* Forest K *Fold* 10 dipilih karena menghasilkan Akurasi yang baik dibandingkan K-*Fold* 5, pada Tabel 4 diatas Akurasi bernilai 0,68 artinya 68% yang berarti belum baik, secara umum dibidang data sains model dengan Akurasi diatas 70% dapat digolongkan sebagai model yang berkinerja cukup baik [8]. Akurasi pada model *Undersampling Random* Forest ini memiliki Akurasi 0,63 artinya 63%, Akurasi ini mengukur keseluruhan Akurasi model, tanpa membedakan error FP ataupun FN. Informasi Akurasi ini sebenarnya kurang informatif terutama pada penerapan model yang lebih difokuskan pada mendeteksi hal-hal yang sangat peka pada *False Positive* atau *False Negative* saja. Presisi memberi petunjuk seberapa baik model dapat memprediksi yang positif. Presisi menjawab pertanyaan “Berapa persen pasien yang kenyatannya tidak terkena kanker dari keseluruhan pasien yang diprediksi tidak terkena kanker. Nilai Presisi *Undersampling Random* Forest sebesar 0,56 hal ini menunjukkan hanya 56% model berhasil memprediksi data yang positif, nilai Presisi *Random* Forest K-*Fold* 10 sebesar 0,31 atau 31%. Sensitifitas atau *Recall* mengukur banyaknya data yang yang sukses diprediksi sebagai positif dibandingkan dengan seluruh data yang pada kenyataannya positif. *Recall* menjawab pertanyaan “Berapa persen pasien yang diprediksi tidak terkena kanker dibandingkan keseluruhan pasien yang sebenarnya tidak terkena kanker”. Model menunjukkan angka *Sensitivity* untuk *Undersampling* sebesar adalah 65%, K-*Fold* 10 sebesar 83%. Nilai *Recall* Algoritma *Random* Forest dengan K-*Fold* 10 adalah sebesar 83% lebih tinggi daripada *Undersampling Random* Forest yang memiliki nilai *Recall* 65%, pada penelitian ini lebih memilih *False Positive* lebih baik terjadi daripada *False Negative*. maka kita mempertimbangkan *Recall* karena lebih baik algoritma kita memprediksi pasien positif tidak terkena kanker tetapi sebenarnya terkena kanker daripada algoritma salah memprediksi bahwa pasien diprediksi terkena kanker padahal sebenarnya pasien yang tidak terkena kanker payudara. Adapun nilai FP untuk algoritma *Undersampling* adalah 7 yang sebenarnya lebih baik daripada nilai FP algoritma *Random* Forest K-*Fold* yaitu 11. Namun nilai TN untuk algoritma *Random Forest* K-*Fold* masih lebih banyak yaitu 34 dibandingkan dengan nilai TN untuk algoritma *Undersampling Random* Forest sebesar 27.

**Building of Informatics, Technology and Science (BITS)**
Volume 4, No 4, Maret 2023 Page: 1967−1974
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i4.3141

### 3.6 Kurva ROC

Adapun nilai AUC dipakai sebagai ukuran baik buruknya suatu model [8]. AUC mendekati 1 menunjukkan model yang mendekati sempurna, sementara AUC mendekati 0,5 adalah model yang buruk. Nilai AUC untuk model *Undersampling* adalah 0,61, model K-*Fold* 10 adalah 0,57.

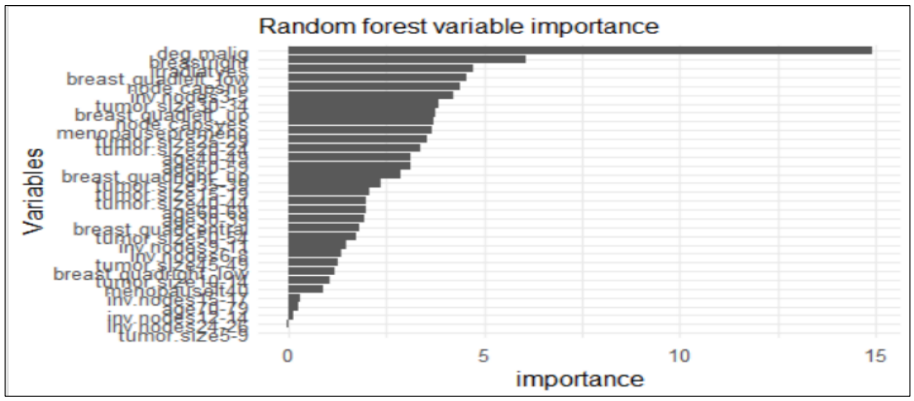



**Gambar 4**. Kurva ROC model *Random* Forest K-*Fold*

Sebagai referensi tambahan pengukuran kebaikan suatu algoritma, Berdasarkan gambar 4 dapat diketahui nilai AUC = 0,57 maka kategori pengklasifikasian dapat dikatakan baik.

### 3.7 Variabel Penting

Salah satu hal yang bisa diketahui dari hasil model *Random* Forest adalah tingkat kepentingan variabel yang digunakan [22].




**Gambar 5.** Variabel Penting pada data *Breast Cancer*

Gambar 5 menunjukkan prioritas dari 10 variabel pada penelitian ini, variabel deg malig terlihat bahwa deg.malig merupakan variabel yang paling berpengaruh terhadap kelas diagnosis kanker dibandingkan variabel lain.

## 4. KESIMPULAN

Penelitian ini merupakan salah satu penelitian Kesehatan yang mana lebih memilih *False Positif* lebih baik terjadi daripada *False Negatif*. maka kita mempertimbangkan *Recall* karena lebih baik algoritma kita memprediksi pasien positif tidak terkena kanker tetapi sebenarnya terkena kanker daripada algoritma salah memprediksi bahwa pasien diprediksi terkena kanker padahal sebenarnya pasien tidak terkena kanker payudara. *Recall* menjawab pertanyaan "Berapa persen pasien yang diprediksi tidak terkena kanker dibandingkan keseluruhan pasien yang sebenarnya tidak terkena kanker". Adapun nilai FP untuk algoritma *Undersampling* adalah 7 yang sebenarnya lebih baik daripada nilai FP algoritma *Random Forest* K-*Fold* yaitu 11. Namun nilai TN untuk algoritma Random Forest K-Fold masih lebih banyak yaitu 34 dibandingkan dengan nilai TN untuk algoritma Undersampling Random Forest sebesar 27. Hasil pengujian yang telah dilakukan dapat disimpulkan bahwa Algoritma *Random* Forest K-*Fold* dan *Undersampling Random* Forest sudah tepat diimplementasikan pada penelitian ini. Akurasi prediksi *Random* Forest K-*Fold* 10 yaitu sebesar 68%, nilai *Recall* 83% dan Nilai AUC sebesar 0,57, Akurasi untuk model *Undersampling Random Forest* adalah sebesar 63%, nilai *Recall* 65% dan Nilai AUC sebesar 0,61. Algoritma *Random Forest* K-*Fold* 10 memiliki hasil Akurasi yang baik daripada *Undersampling Random Forest* dari pengujian diatas maka penerapan algoritma *Undersampling* dan *Random Forest* K-*Fold* mempu menangani ketidakseimbangan kelas dataset pada prediksi diagnosis kanker dengan menghasilkan nilai AUC diatas 0,5.

**Building of Informatics, Technology and Science (BITS)**
Volume 4, No 4, Maret 2023 Page: 1967−1974
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i4.3141

# REFERENCES

[1] D. Pramadhana, R. Rendi, and R. Robiyanto, "Peningkatan Algoritma J48 Untuk Klasifikasi Hasil Prestasi Mahasiswa Selama Proses Pembelajaran Secara Daring Menggunakan CFS Dan Adaboost," J. Informatics Inf. Syst. Softw. Eng. Appl., vol. 5, no. 1, pp. 17–26, 2022.

[2] R. D. Fitriani, H. Yasin, and T. Tarno, "Penanganan Klasifikasi Kelas Data Tidak Seimbang Dengan Random *Oversampling* Pada Naive Bayes (Studi Kasus: Status Peserta KB IUD di Kabupaten Kendal)," J. Gaussian, vol. 10, no. 1, pp. 11–20, 2021.

[3] R. T. Prasetio and P. Pratiwi, "Penerapan Teknik Bagging pada Algoritma Klasifikasi untuk Mengatasi Ketidakseimbangan Kelas Dataset Medis," J. Inform., vol. 2, no. 2, 2015.

[4] F. D. Astuti and F. N. Lenti, "Implementasi SMOTE untuk mengatasi Imbalance Class pada Klasifikasi Car Evolution menggunakan K-NN".

[5] R. Ihfa and T. Harsanti, "Komparasi Teknik *Resampling* Pada Pemodelan Regresi Logistik Biner," in Seminar Nasional Official Statistics, 2020, vol. 2020, no. 1, pp. 863–870.

[6] R. Siringoringo, "Klasifikasi data tidak seimbang menggunakan algoritma SMOTE dan k-nearest neighbor," J. Inf. Syst. Dev., vol. 3, no. 1, 2018.

[7] M. P. Pangestika, I. M. Sumertajaya, and A. Rizki, "Penerapan Synthetic Minority *Oversampling* Technique pada Pemodelan Regresi Logistik Biner terhadap Keberhasilan Studi Mahasiswa Program Magister IPB," Xplore J. Stat., vol. 10, no. 2, pp. 152–166, 2021.

[8] L. Qadrini, H. Hikmah, and M. Megasari, "*Oversampling*, *Undersampling*, Smote SVM dan *Random* Forest pada Klasifikasi Penerima Bidikmisi Sejawa Timur Tahun 2017," J. Comput. Syst. Informatics, vol. 3, no. 4, pp. 386–391, 2022.

[9] R. Wasono, "Perbandingan Algoritma *Random* Forest dan naive bayes untuk Klasifikasi Debitur Berdasarkan Kualitas Kredit," 2022.

[10] A. Lestari, E. Mariati, and W. Widiatry, "Model Klasifikasi Kepuasan Mahasiswa Teknik Terhadap Sarana Pembelajaran Menggunakan Data Mining," J. Teknol. Inf. J. Keilmuan dan Apl. Bid. Tek. Inform., vol. 14, no. 2, pp. 112–118, 2020.

[11] N. Sulistiyowati and M. Jajuli, "Integrasi Naive Bayes Dengan Teknik Sampling SMOTE Untuk Menangani Data Tidak Seimbang," Nuansa Inform., vol. 14, no. 1, pp. 34–37, 2020.

[12] W. I. Sabilla and C. B. Vista, "Implementasi SMOTE dan Under Sampling pada Imbalanced Dataset untuk Prediksi Kebangkrutan Perusahaan," J. Komput. Terap., vol. 7, no. 2, pp. 329–339, 2021.

[13] A. I. Kusumarini, P. A. Hogantara, M. Fadhlurohman, N. Chamidah, S. Kom, and M. Kom, "Perbandingan Algoritma Random Forest, Naive Bayes, Dan Decision Tree Dengan *Oversampling* Untuk Klasifikasi Bakteri E. Coli," in Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya, 2021, vol. 2, no. 1, pp. 792–799.

[14] E. Saputro and D. Rosiyadi, "Penerapan Algoritma Random Over-Under Sampling Pada Algoritma Klasifikasi Penentuan Penyakit Diabetes," Bianglala Inform., vol. 10, no. 1, pp. 42–47, 2022.

[15] O. Heranova, "Synthetic Minority *Oversampling* Technique pada Averaged One Dependence Estimators untuk Klasifikasi Credit Scoring," J. RESTI (Rekayasa Sist. Dan Teknol. Informasi), vol. 3, no. 3, pp. 443–450, 2019.

[16] R. Prasetyo, I. Nawawi, and A. Fauzi, "Komparasi Algoritma Logistic Regression dan *Random* Forest pada Prediksi Cacat Software," J. Tek. Inform. UNIKA St. Thomas, pp. 275–281, 2021.

[17] R. I. Arumnisaa and A. W. Wijayanto, "Comparison of Ensemble Learning Method: Random Forest, Support Vector Machine, AdaBoost for Classification Human Development Index (HDI)," Sist. J. Sist. Inf., vol. 12, no. 1, pp. 206–218, 2023.

[18] Qadrini L, Sepperwali A, and Aina A, "Decision Treedan Adaboostpada Klasifikasi Penerima Program Bantuan Sosial," Decis. Tree Dan Adab. Pada Klasifikasi Penerima Progr. Bantu. Sos., vol. 2, no. 7, pp. 1959–1966, 2021.

[19] O. Arifin and T. B. Sasongko, "Analisa perbandingan tingkat performansi algoritma support vector machine dan naïve bayes *Classifier* untuk klasifikasi jalur minat SMA," SEMNASTEKNOMEDIA ONLINE, vol. 6, no. 1, pp. 1–2, 2018.

[20] R. Ridwansyah, I. Ariyati, and S. Faizah, "PARTICLE SWARM OPTIMIZATION BERBASIS CO-EVOLUSIONER DALAM EVALUASI KINERJA ASISTEN DOSEN," J. Saintekom, vol. 9, no. 2, pp. 165–177, 2019.

[21] E. D. Wahyuni, A. A. Arifiyanti, and M. Kustyani, "Exploratory data analysis dalam konteks klasifikasi data mining," ReTII, pp. 263–269, 2019.

[22] E. Christy and K. Suryowati, "ANALISIS KLASIFIKASI STATUS BEKERJA PENDUDUK DAERAH ISTIMEWA YOGYAKARTA MENGGUNAKAN ALGORITMA RANDOM FOREST," J. Stat. Ind. dan Komputasi, vol. 6, no. 01, pp. 69–76, 2021.