

**IMPLEMENTASI ALGORITME SMOTE DAN KLASIFIKASI  
RANDOM FOREST PADA IMBALANCED DATA METILASI  
SEQUENCE PROTEIN LISIN**

(Skripsi)

Oleh

**ANNISA NURWALIKADANI  
1817051037**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2022**

**IMPLEMENTASI ALGORITME SMOTE DAN KLASIFIKASI  
RANDOM FOREST PADA IMBALANCED DATA METILASI  
SEQUENCE PROTEIN LISIN**

Oleh

**Annisa Nurwalikadani**

**Skripsi**

**Sebagai Salah Satu Syarat untuk Memperoleh Gelar  
SARJANA ILMU KOMPUTER**

**pada**

**Jurusan Ilmu Komputer  
Fakultas Matematika dan Ilmu Pengetahuan Alam**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2022**

## **ABSTRACT**

### **IMPLEMENTATION SMOTE ALGORITHM AND RANDOMFOREST CLASSIFICATION IN IMBALANCED DATA LYSINE PROTEIN SEQUENCE METHYLATION**

**By**  
**Annisa Nurwalikadani**

One problem that is often encountered when processing data is an imbalance in the number of samples from each class in the data or what is commonly called imbalanced data. In recent years, there have been many studies regarding imbalanced data in the field of bioinformatics, especially Posttranslational Modification (PTM). The case study in this research is the problem of imbalanced data lysine protein sequence methylation. The problem of imbalanced data will certainly affect the classification results. Therefore, a method is needed to deal with this problem, one of which is the SMOTE algorithm that will be used in this research. The purpose of this research is to analyze the performance of the SMOTE algorithm and random forest classification in handling with imbalanced data problems. The lysine protein sequence methylation dataset was obtained from <http://www.uniprot.org/> by searching for the keyword "methylation", has 1000 positive data and 172 negative data and has a sequence length of 15 amino acids. The dataset separation into 2, 80% training data 20% test data and 90% training data 10% test data. The feature extraction used is AA Index, PseACC, Hydrophobicity, and CTD. Then, the extracted data was processed using a random forest classification with ntree parameters 500, 800, and 1000 and mtry 7, 9, and 14. The highest results were obtained in the separation of the dataset 80% training data 20% test data with mtry 14 ntree 500, resulting in 95.65% accuracy, 96.2% sensitivity, 95% specificity, and 91.25% MCC.

**Keywords :** Post-translational modification, methylation, imbalanced data, feature extraction, random forest, SMOTE

## **ABSTRAK**

### **IMPLEMENTASI ALGORITME SMOTE DAN KLASIFIKASI RANDOM FOREST PADA IMBALANCED DATA METILASI SEQUENCE PROTEIN LISIN**

**Oleh**  
**Annisa Nurwalikadani**

Salah satu masalah yang sering ditemukan pada saat pengolahan data adalah ketidakseimbangan jumlah sampel dari masing-masing kelas dalam data atau yang biasa disebut dengan *imbalanced data*. Dalam beberapa tahun terakhir, banyak penelitian mengenai ketidakseimbangan data di bidang bioinformatika, khususnya Post translational Modification (PTM). Studi kasus dalam penelitian ini adalah masalah ketidakseimbangan data metilasi *sequence* protein lisin. Masalah ketidakseimbangan data tentunya akan mempengaruhi hasil klasifikasi. Oleh karena itu diperlukan suatu metode untuk mengatasi masalah tersebut, salah satunya adalah algoritme SMOTE yang akan digunakan dalam penelitian ini. Tujuan dari penelitian ini adalah untuk menganalisis kinerja algoritma SMOTE dan klasifikasi *random forest* dalam menangani masalah ketidakseimbangan data. Dataset metilasi urutan protein lisin diperoleh dari <http://www.uniprot.org/> dengan mencari kata kunci "metilasi", memiliki 1000 data positif dan 172 data negatif serta memiliki panjang *sequence* 15 asam amino. Pemisahan *dataset* menjadi 2, yaitu 80% data latih 20% data uji dan 90% data latih 10% data uji. Fitur ekstraksi yang digunakan adalah AA Index, PseACC, Hydrophobicity, dan CTD. Kemudian, data hasil ekstraksi diolah menggunakan klasifikasi *random forest* dengan parameter ntree 500, 800, dan 1000 serta mtry 7, 9, dan 14. Hasil tertinggi diperoleh pada pemisahan dataset 80% data latih 20% data uji dengan mtry 14 ntree 500, menghasilkan akurasi 95,65%, sensitivitas 96,2%, spesifisitas 95%, dan MCC 91,25%.

**Kata kunci :** *Post-translational modification*, metilasi, data tidak seimbang, fitur ekstraksi, *random forest*, SMOT

Judul Skripsi

**: IMPLEMENTASI ALGORITME SMOTE  
DAN KLASIFIKASI RANDOM FOREST  
PADA IMBALANCED DATA METILASI  
SEQUENCE PROTEIN LISIN**

Nama Mahasiswa

**: Annisa Nurwasikadani**

Nomor Pokok Mahasiswa

**: 1817051037**

Jurusan

**: Ilmu Komputer**

Fakultas

**: Matematika dan Ilmu Pengetahuan Alam**

**MENYETUJUI**

**1. Komisi Pembimbing**

**Favorisen R. Lumbanraja, Ph.D.**  
NIP 19830110 200812 1 002

**2. Ketua Jurusan Ilmu Komputer**

**Didik Kurniawan, S.Si., M.T.**  
NIP 19800419 200501 1 004

## **MENGESAHKAN**

**1. Tim Pengaji**

Ketua

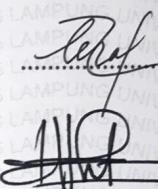
: **Favorisen R. Lumbanraja, Ph.D.**



Pengaji I

Pengaji Pembahas

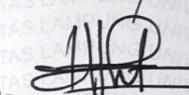
: **M. Reza Faisal, S.T., M.T., Ph.D.**



Pengaji II

Pengaji Pembahas

: **Dr. rer. nat. Akmal Junaidi, M.Sc.**



**2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam**



  
**Dr. Eng. Suryipto Dwi Yuwono, S.Si., M.T.**

NIP 19740705 200003 1 001

**Tanggal Lulus Ujian Skripsi : 24 November 2022**

## **PERNYATAAN**

Saya yang bertanda tangan di bawah ini:

Nama : Annisa Nurwalikadani  
NPM : 1817051037

Dengan ini menyatakan bahwa skripsi saya yang berjudul “IMPLEMENTASI ALGORITME SMOTE DAN KLASIFIKASI RANDOM FOREST PADA IMBALANCED DATA METILASI SEQUENCE PROTEIN LISIN” adalah benar hasil karya sendiridan bukan orang lain. Seluruh tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Jika di kemudian hari terbukti skripsi saya adalah hasil penjiplakan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.

Bandar Lampung, 14 Desember 2022

Penulis



Annisa Nurwalikadani

1817051037

## **RIWAYAT HIDUP**



Penulis bernama lengkap Annisa Nurwalikadani, dilahirkan di Kabupaten Way Kanan pada tanggal 7 Oktober 2000. Penulis merupakan putri sulung dari Bapak Ahmad Rosani dan Ibu Meda Sari Ira. Penulis menyelesaikan Pendidikan Sekolah Dasar (SD) pada tahun 2012 di SD N 1 Sukaramo. Kemudian, melanjutkan Pendidikan Sekolah Pertama (SMP) di SMPAI Kautsar Bandar Lampung pada tahun 2012 dan selesai pada tahun 2015. Pada tahun itu juga, penulis melanjutkan Pendidikan Sekolah Menengah Atas (SMA) di SMA Al Kautsar Bandar Lampung dan lulus pada tahun 2018. Penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung pada tahun 2018 melalui jalur Seleksi Bersama Masuk Perguruan Tinggi Negeri (SBMPTN). Selama menjadi mahasiswa, penulismelakukan beberapa kegiatan antara lain.

1. Menjadi anggota Adapter Himpunan Mahasiswa Jurusan Ilmu Komputer padaperiode 2018/2019.
2. Menjadi bendahara bidang Kaderisasi Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2019/2020.
3. Melaksanakan Kerja Praktik di Gink Technology pada tahun 2021.
4. Melaksanakan Kuliah Kerja Nyata (KKN) pada tahun ajaran 2021/2022 di Kelurahan Kaliawi, Kecamatan Tanjung Karang Pusat, Kota Bandar Lampung.

## **MOTTO**

”Be patient. Verily Allah is with those who are patient.“

**(Surah Al-Anfaal 46)**

“It is possible that you hate something even though it is very good for you, and it is also possible that you like something even though it is very bad for you, Allah knows while you do not know.“

**(Surah Al-Baqarah 216)**

“To achieve what you want, you must continue to chase and fight, but at the same time keep yourself in good condition.”

**(Chanyeol Park)**

## **PERSEMBAHAN**

*Alhamdulillahirobbilalamin*

Puji syukur kepada Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga dapat menyelesaikan penulisan skripsi ini. Sholawat dan salam snantiasa saya haturkan kepada Nabi Muhammad SAW.

Ku persembahkan karya ini kepada:

### **Orang tuaku Tercinta**

Sebagai tanda terima kasihku kepada orang tuaku tercinta dan tersayang. Terima kasih telah mendidik dan membeskarkanku dengan penuh kasih sayang. Terima kasih atas semua pengorbanan, dukungan, dan doa yang tiada henti.

### **Keluargaku Tercinta**

Terima kasih telah memberikan semangat, dukungan, dan doa.

**Sahabat dan Teman-teman** yang selalu memberikan semangat dan dukungan.

**Almamater Tercinta, Universitas Lampung**

## **SANWACANA**

Puji syukur kehadirat Allah SWT, karena telah memberikan rahmat dan hidayah-Nyakepada saya sehingga saya dapat menyelesaikan skripsi dengan judul “Implementasi Algoritme SMOTE Dan Klasifikasi Random Forest pada Imbalanced Data Metilasi Sequence Protein Lisin”. Saya berharap skripsi ini dapat menambah pengetahuan bagipembaca tentang metilasi protein, fitur ekstraksi, algoritme SMOTE dan klasifikasi random forest.

Proses penulisan skripsi ini tidak terlepas dari dukungan banyak pihak yang telah membimbing, membantu, dan mendukung, sehingga pada kesempatan ini saya ingin menyampaikan ungkapan terima kasih kepada:

1. Ayah, bunda, adik-adik, dan keluarga yang selalu mendoakan, member dukungan, kasihsayang, dan semangat baik secara moral maupun material dalam menyelesaikanskripsi ini.
2. Bapak Favorisen R. Lumbanraja, Ph.D. sebagai dosen pembimbing sekaligus pembimbing akademik yang telah membimbing, memberikan kritik dan saran dalam menyelesaikan skripsi ini sehingga dapat diselesaikan dengan baik.
3. Bapak M. Reza Faisal, S.T., M.T., Ph.D. sebagai pembahas pertama yang telah membimbing dalam memberikan ide, kritik, saran sehingga penulisan skripsi ini dapat diselesaikan dengan baik.
4. Bapak Dr. rer. nat. Akmal Junaidi, M.Sc. sebagai pembahas kedua sekaligus sekretaris Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah membimbing dalam memberikan ide, kritik, saran sehingga penulisan skripsi ini dapat selesai dengan baik.
5. Bapak Dr. Eng. Suripto Dwi Yuwono, S.Si., M.T. selaku dekan FMIPA Universitas Lampung.

6. Bapak Didik Kurniawan, S.Si., M.T., selaku Ketua Jurusan Ilmu Komputer Universitas Lampung.
7. Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan ilmu dan pengalaman selama di perkuliahan.
8. Teman-teman saya, Syela Septania, Rika Ningtias Azhari, Abie Perdana Kusuma, Arbi Hidayatullah, Yuan Ferdinand, dan Pandi Barep Arianza yang telahmenemani, membantu, dan memotivasi selama masa perkuliahan.
9. Teman seperbimbingan saya, Intania Rahmadhilla, Ridho Alrafi, M. Fajru Ramadhan, M. Sepryan Astrayesa, dan Suci Hikmawati yang telah membantu, menemani, dan memberi semangat satu sama lain.
10. Rekan kerja sekaligus keluarga baru saya di perantauan, Alfina Lailil Izza, Elysia Muchris, Rhaistu Nurmatalita, Haykal Ramadhani Irwan, Muhammad Nur, Nilam Amalia Hidayah, dan Nia Winiati yang selalu memotivasi saya dalam menyelesaikan penulisan skripsi ini.
11. Seluruh member EXO yang perjalanan hidup dan karyanya telah memotivasi dan memberi semangat selama proses penggeraan skripsi.
12. Teman-teman Ilmu Komputer 2018 yang telah memberikan pengalaman tak ternilai semasa duduk di bangku kuliah.
13. Semua pihak yang telah berpartisipasi baik secara langsung maupun tidak langsung dalam membantu penyusunan skripsi ini.

Saya menyadari bahwa dalam penulisan skripsi ini masih terdapat banyakkekurangan. Oleh karena itu, saran dan kritik yang membangun sangat diharapkansebagai bahan evaluasi untuk kedepannya. Semoga skripsi ini dapat bermanfaat bagi semua pihak.

Bandar Lampung, 14 Desember 2022

Annisa Nurwalikadani

## DAFTAR ISI

<b>DAFTAR ISI .....</b>	<b>xiii</b>
<b>DAFTAR TABEL .....</b>	<b>xv</b>
<b>DAFTAR GAMBAR .....</b>	<b>xvi</b>
<b>DAFTAR <i>PSEUDOCODE</i> .....</b>	<b>xvii</b>
<b>I. PENDAHULUAN .....</b>	<b>1</b>
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah .....	3
1.4. Tujuan Penelitian.....	4
1.5. Manfaat Penelitian .....	4
<b>II. TINJAUAN PUSTAKA .....</b>	<b>5</b>
2.1. Penelitian Terdahulu .....	5
2.2. Protein .....	8
2.3. Post-translational Modification .....	9
2.4. Metilasi.....	12
2.5. Imbalanced Data .....	12
2.6. Synthetic Minority Oversampling Technique .....	13
2.7. Random Forest.....	16
2.8. Feature Extraction.....	19
2.9. Cross-validation.....	24
2.10. Performance Classification.....	26

III. DATA DAN METODOLOGI .....	29
3.1. Waktu dan Tempat.....	29
3.1.1. Tempat Penelitian .....	29
3.1.2. Waktu dan Jadwal (Rencana) Penelitian.....	29
3.2. Data dan Alat.....	32
3.2.1. Data.....	32
3.2.2. Alat .....	32
3.3. Metodologi .....	34
IV. HASIL DAN PEMBAHASAN .....	36
4.1. Pra Pemrosesan Data .....	36
4.2. Ekstraksi Fitur .....	37
4.3. Pemrosesan Data.....	39
4.4. Import Data Setelah Ekstraksi Fitur .....	40
4.5. Pembagian Data Menggunakan <i>5-Fold Cross Validation</i> .....	41
4.6. Klasifikasi Random Forest.....	41
4.7. Pemodelan SMOTE .....	42
4.8. Pengujian Hasil Klasifikasi .....	43
4.9. Pembahasan .....	50
4.10. Perbandingan dengan Penelitian Sebelumnya .....	53
V. PENUTUP .....	55
5.1. Simpulan .....	55
5.2. Saran .....	56
DAFTAR PUSTAKA .....	57

## DAFTAR TABEL

<b>Tabel</b>	<b>Halaman</b>
1. Penelitian Terkait Klasifikasi Random Forest dan Algoritme SMOTE .....	5
2. Jenis PTM dan Penjelasannya (Spoel et.al., 2018) .....	10
3. Contoh Data Tidak Seimbang.....	15
4. <i>Output</i> penggunaan <i>featureCTD</i> .....	21
5. <i>Output</i> penggunaan <i>featureAAindex</i> .....	22
6. <i>Output</i> penggunaan <i>featurePseAAComp</i> .....	23
7. <i>Output</i> penggunaan <i>featureHydro</i> .....	24
8. Evaluasi Matriks Dua Kelas Data.....	26
9. Rencana Alur dan Waktu Penelitian .....	30
10. Bentuk Dataset Metilasi Lisin .....	32
11. Nilai Prediksi dan Nilai Aktual 20% Data Uji Tanpa SMOTE .....	43
12. Hasil Pengujian Klasifikasi 20% Data Uji Tanpa SMOTE.....	44
13. Nilai Prediksi dan Nilai Aktual 10% Data Uji Tanpa SMOTE .....	45
14. Hasil Pengujian Klasifikasi 10% Data Uji Tanpa SMOTE.....	45
15. Hasil Pengujian Menggunakan SMOTE dengan <i>ntree</i> 500 .....	46
16. Hasil Pengujian Menggunakan SMOTE dengan <i>ntree</i> 800 .....	47
17. Hasil Pengujian Menggunakan SMOTE dengan <i>ntree</i> 1000 .....	48
18. Hasil Pengujian Menggunakan SMOTE 10% Data Uji .....	49
19. Perbandingan Hasil Pengujian dengan Penelitian Sebelumnya.....	53

**DAFTAR GAMBAR**

<b>Gambar</b>	<b>Halaman</b>
1. Tingkatan Struktur Protein (Oliviera et.al., 2018) .....	9
2. <i>Post-translational Modification</i> (Spoel et.al., 2018) .....	10
3. <i>LysineMethylation</i> (Han et.al., 2019).....	12
4. Ilustrasi <i>oversampling</i> , <i>undersampling</i> , dan <i>hybridsteknik</i> (Le et.al., 2019)....	14
5. Ilustrasi alur kerja metode <i>random forest</i> (Yang et.al., 2019).....	17
6. <i>K-Fold Cross Validation</i> (Ren et.al., 2019) .....	25
7. <i>Leave-One-Out Cross Validation</i> .....	25
8. Diagram Alir Rencana Penelitian .....	34
9. Grafik Hasil Pengujian Data Metilasi Lisin 80% Training 20% Testing.....	50
10. Grafik ROC Data Lisin 80% Training 20% Testing .....	51
11. Grafik Hasil Pengujian Data Metilasi Lisin 90% Training 10% Testing.....	52
12. Grafik ROC Data Lisin 90% Training 10% Testing .....	53

## **DAFTAR PSEUDOCODE**

<b>Pseudocode</b>	<b>Halaman</b>
1. Kode program <i>featureCTD</i> .....	20
2. Kode program <i>featureAAindex</i> . .....	21
3. Kode program <i>featurePseudoAACComp</i> . .....	22
4. Kode program <i>featureHydro</i> . .....	23
5. Kode Program Import Data. ....	36
6. Kode Program Ekstraksi Fitur AAindex. ....	37
7. Kode Program Ekstraksi Fitur CTD.....	38
8. Kode Program Ekstraksi Fitur Hydrophobicity. ....	38
9. Kode Program Ekstraksi Fitur PseAAC.....	39
10. Kode Program Penggabungan Data Ekstraksi Fitur. ....	39
11. Kode Program Penggabungan Kelas Positif dan Negatif.....	40
12. Kode Program Pemberian Label Kelas Data. ....	40
13. Import Data Setelah Ekstraksi. ....	40
14. Kode Program Pembagian Data Uji dan Data Latih. ....	41
15. Kode Program Klasifikasi Random Forest.....	41
16. Kode Program Hasil Klasifikasi Random Forest.....	42
17. Kode Program Pemodelan SMOTE. ....	42

## I. PENDAHULUAN

### 1.1. Latar Belakang

*Post-translational Modification* (PTM) adalah peristiwa pemrosesan kovalen yang mengubah sifat protein dengan pembelahan *proteolytic* atau menambahkan gugus pengubah menjadi satu atau lebih asam amino(Martins et al., 2018). Peristiwa ini dapat menentukan status aktivitas protein, lokalisasi, pergantian, dan interaksi dengan protein lain(Xu et al., 2014).Salah satu faktor yang mempengaruhi PTM adalah urutan asam amino dalam struktur protein.Modifikasi *post-translation* seringterjadi pada metilasi urutan asam amino lisin pada protein(Lee et al., 2005). Metilasi lisin memiliki fungsi penting dalam banyak proses biologis yang mencakup pembentukan heterokromatin, inaktivasi kromosom X dan regulasi transkripsi(Martin & Zhang, 2005).

Salah satu masalah yang sering dihadapi ketika memprediksi metilasi pada *sequence* protein adalah ketidakseimbangan jumlah sampel dari setiap kelas dalam data. *Imbalanced* data terjadi ketika jumlah data dalam satu kelas jauh lebih tinggi (*majority class*) atau lebih rendah (*minority class*) dibandingkan kelas lainnya (Indrawati et al., 2021). Kondisi ini mempengaruhi kinerja model dalam memprediksi interaksi, sehingga hasil prediksi menjadi bias. Oleh karena itu, diperlukan praproses data untuk meminimalkan hilangnya keseimbangan data.

Salah satu hal yang perlu diperhatikan dalam evaluasi model adalah tingkat akurasi sebuah model dalam memprediksi respon dengan benar. Kebaikan model dipengaruhi salah satunya oleh adanya keseimbangan

antara kelas mayor dengan kelas minor. Kelas mayor adalah data yang ukuran kelasnya lebih besar dari kelas minor berdasarkan peubah respon(Barro et al., 2013). Jika data yang digunakan untuk membuat model tidak seimbang maka akan meningkatkan salah klasifikasi kelas minor.

Dalam beberapa tahun terakhir, telah banyak studi mengenai data tidak seimbang dalam bidang bioinformatika. Salah satunya dari penelitian yang berjudul “Optimasi Data Tidak Seimbang pada Interaksi Drug Target dengan *Sampling* dan *Ensemble Support Vector Machine*” yang dilakukan oleh Sekar et al. (2020). Penelitian ini mengeksplorasi pengaruh *sampling* yang digabungkan dalam metode SVM pada data interaksi senyawa-protein. Pengujian ini dilakukan pada *dataset Nuclear Receptor*, *G-Protein Coupled Receptor* dan *Ion Channel* dengan rasio ketidakseimbangan sebesar 14.6%, 32.36%, dan 28.2%. Hasil pengujian dengan menggunakan ketiga *dataset* tersebut menunjukkan nilai *area under curve* (AUC) secara berturut-turut sebesar 63.4%, 71.4%, 61.3% dan *F-measure* sebesar 54%, 60.7%, dan 39%. Meskipun nilai akurasi dari metode yang diusulkan lebih kecil dari metode SVM tanpa perlakuan apapun, hasil AUC dan *F-measure* menunjukkan bahwa metode yang digunakan pada penelitian ini mampu menurunkan bias dari model yang menggunakan SVM tersebut. Hal ini ditunjukkan oleh peningkatan nilai AUC dan *f-measure* sekitar 5%-20%.

Pada penelitian ini, mengusung studi kasus ketidakseimbangan data pada metilasi *sequence* protein lisin. Seperti penjabaran pada paragraf sebelumnya, metilasi lisin merupakan hal penting dalam berbagai proses biologis. Dalam hal ini, ketidakseimbangan data menjadi masalah yang perlu diatasi untuk mendapatkan prediksi dan klasifikasi yang baik.Oleh karena itu, diperlukan alternatif untuk meningkatkan akurasi model pada *dataset* tersebut.

Penelitian ini dibangun dengan menggunakan algoritme SMOTE dan metode klasifikasi *random forest*. *Synthetic Minority Over-sampling Technique* (SMOTE) merupakan metode pembangkitan data minoritas sebanyak data mayoritas(Kasanah et al., 2019). *Random forest* adalah kumpulan dari pohon klasifikasi hasil dari sampling *bootstrap* data(Chairunisa et al., 2020). Langkah awal dalam membangun model *random forest* yaitu menentukan nilai N sebagai jumlah *decision tree* yang dibangun. Pemilihan data yang digunakan untuk pembangunan tree menggunakan teknik *bootstrapsample*. Teknik ini akan memilih sampel dari data secara acak dan dilakukan secara berulang hingga jumlah sampel pada *bootstrapsample* sama dengan jumlah data sebenarnya. Penelitian ini membandingkan model klasifikasi *random forest* dengan proses SMOTE dan tanpa proses SMOTE dalam studi kasus ketidakseimbangan data metilasi *sequenceprotein lisin*.

### **1.2. Rumusan Masalah**

Berdasarkan pemaparan latar belakang, adapun masalah dalam penelitian ini adalah sebagai berikut :

1. Apakah ketidakseimbangan data metilasi *sequence protein lisin* pada manusia dapat mempengaruhi hasil prediksi?
2. Berapa akurasi kinerja metode klasifikasi *random forest* dengan dan tanpa proses algoritme SMOTE dalam mengatasi masalah ketidakseimbangan data pada studi kasus metilasi *sequence protein lisin*?

### **1.3. Batasan Masalah**

Batasan masalah dalam penelitian ini adalah sebagai berikut :

1. Penelitian dilakukan dengan hanya menggunakan metode klasifikasi *random forest* dengan algoritme *Synthetic Minority Oversampling*

*Technique* (SMOTE), serta *feature extraction* PseAAC, CTD, AA Index, dan *Hydrophobicity*.

2. Data yang digunakan diperoleh dari riset penelitian Hasan & Ahmad (2018) yang merupakan *dataset* metilasi *sequence* protein lisin pada manusia.

#### **1.4. Tujuan Penelitian**

Tujuan dari penelitian adalah sebagai berikut :

1. Menentukan akurasi performa algoritme SMOTE dan metode *random forest* dalam masalah ketidakseimbangan data metilasi *sequence* protein lisin.
2. Membandingkan hasil model klasifikasi dan nilai akurasi pada penelitian ini dengan penelitian Hasan & Ahmad(2018)dengan judul “*mLysPTMpred: Multiple Lysine PTM Site Prediction Using Combination of SVM with Resolving Data Imbalance Issue*”.

#### **1.5. Manfaat Penelitian**

Manfaat dari penelitian ini adalah sebagai berikut :

1. Menambah pengetahuan tentang ketidakseimbangan data metilasi protein lisin.
2. Menambah pengetahuan mengenai manfaat gugus metil pada protein lisin.
3. Menjadi informasi untuk penelitian selanjutnya dalam masalah ketidakseimbangan data metilasi protein lisin.

## II. TINJAUAN PUSTAKA

### 2.1. Penelitian Terdahulu

Dalam melakukan penelitian ini, tentunya dibutuhkan *paper* penelitian terdahulu terkait metode klasifikasi *random forest* dan algoritme SMOTE untuk menunjang berlangsungnya penelitian yang akan dilakukan. Berikut ini merupakan ringkasan penelitian terdahulu yang dapat dilihat pada Tabel 1.

Tabel 1. Penelitian Terkait Klasifikasi Random Forest dan Algoritme SMOTE

No	Penelitian	Data		Metode	Hasil		
		Data Latih					
		Negatif	Positif				
1	Optimasi Data Tidak Seimbang pada Interaksi <i>Drug Target</i> dengan <i>Sampling</i> dan <i>Ensemble Support Vector Machine</i> (Sekar &Kusuma, 2020)	<b>NR</b>	983	70	Akurasi : Metode klasifikasi: <i>Support Vector Machine</i>		
		<b>GPCR</b>	15 429	468			
		<b>IC</b>	31 012	1 118			
		Data Uji					
		Negatif	Positif				
		<b>NR</b>	331	20			
		<b>GPCR</b>	513	167			
		<b>IC</b>	10 352	358			
		Sumber : <i>Benchmark Dataset</i>					
2	<i>mLysPTMpred: Multiple Lysine PTM Site Prediction Using Combination of SVM with Resolving Data Imbalance Issue</i> (Hasan & Ahmad, 2018)	Negatif	6 267	Metode klasifikasi: <i>Support Vector Machine</i>	Akurasi : 83,73%		
			127				
		Sumber : <i>Uniprot Database</i>					
		Ekstraksi Fitur: <i>PseAAC</i>					
3	<i>A Hybrid Approach to Parkinson Disease</i>	<b>Parkinson Disease Dataset</b>		Metode klasifikasi: <i>Random Forest</i>	Akurasi: 87,03% Precision: 87,1%		
		Negatif	192				
		Positif	564				

No	Penelitian	Data	Metode	Hasil
	<i>Classification using speech signal: The combination of SMOTE and Random Forests(Polat, 2019)</i>	Sumber : <i>UCI Machine Learning Database</i>		Recall: 87% F-measure: 86%

### 2.1.1. Optimasi Data Tidak Seimbang pada Interaksi *Drug Target* dengan *Sampling* dan *Ensemble Support Vector Machine*(2020)

Penelitian pertama dilakukan oleh Sekar& Kusuma (2020). Penelitian mengenai eksplorasi pengaruh sampling yang digabungkan dalam metode SVM pada data interaksi senyawa-protein. Data yang dipakai dalam penelitian pada *paper* ini merupakan *dataset* ion channel (IC), G-Protein Coupled Receptor (GPCR) dan nuclear receptor (NR) yang dipakai dalam penelitian Yamanishi, et. al pada tahun 2008.

Penelitian ini menghasilkan nilai akurasi yang sangat tinggi pada seluruh *dataset* yang mencapai rataan 96.1%. Nilai tersebut adalah bias karena kenyataannya akurasi yang tinggi tersebut adalah dari prediksi benar dari data mayoritas yang negatif sedangkan seluruh data positif pada NR salah diklasifikasikan. Penelitian yang telah dilakukan dapat membuktikan bahwa model SVM bias karena tidak dapat melakukan prediksi data positif dengan baik pada kasus data tidak seimbang. Selain itu model dengan hanya sampling pada satu sisi (*oversampling* data positif/ *undersampling* data negatif) juga masih bias karena pada model *oversampling* tidak bisa mendapat prediksi data positif sebaik model *undersampling*, sebaliknya model *undersampling* tidak bisa melakukan prediksi data negatif sebaik model *oversampling*.

### **2.1.2. *mLysPTMpred: Multiple Lysine PTM Site Prediction Using Combination of SVM with Resolving Data Imbalance Issue(2018)***

Penelitian pada tahun 2018 ini dikerjakan oleh Hasan& Ahmad (2018) dengan tujuan mengatasi masalah utama penelitian mengenai imbalance data *methylation* pada asam amino lisin. Dataset yang digunakan berasal dari *uniprot database* yang berjumlah 127 data positif dan 6.267 data negatif. Dalam hal ini, sebaran data menunjukkan bahwa data tersebut *imbalanced*. Telah banyak dilakukan komputasi mengenai hal tersebut. Namun, sebagian besar metode komputasi yang ada telah ditetapkan perlu perbaikan lebih lanjut untuk memprediksi berbagai PTMsites.

Pada penelitian ini, menggunakan metode klasifikasi *support vector machine* dan ekstraksi fitur PseAAC. Hasil akurasinya mencapai 83,73%. Hasil dari penelitian ini menunjukkan bahwa prediktor *mLysPTMpred* yang sederhana dan efisien untuk memprediksi beberapa situs PTM lisin. Hasil percobaan menunjukkan bahwa metode yang digunakan sangat menjanjikan dan dapat menjadi alat yang berguna untuk prediksi beberapa situs PTM lisin.

### **2.1.3. *A Hybrid Approach to Parkinson Disease Classification using speech signal: The combination of SMOTE and Random Forests(2019)***

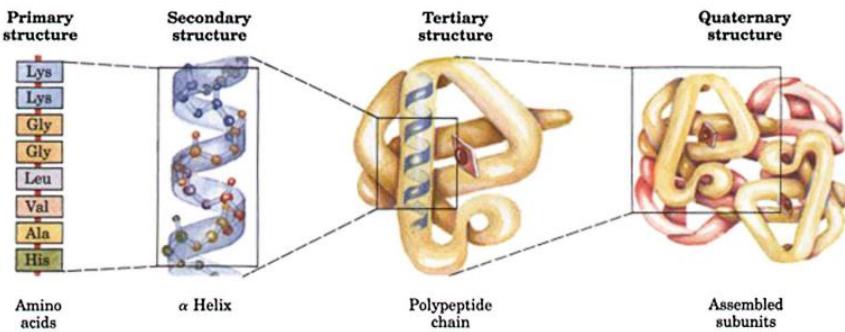
Penelitian selanjutnya mengenai SMOTE dan *random forest* dilakukan oleh Polat (2019). Penelitian ini mengusulkan metode baru untuk deteksi penyakit parkinson dengan fitur yang diperoleh dari *speech signals*. Deteksi dan diagnosis dini penyakit parkinson penting dalam hal perkembangan penyakit dan proses pengobatan. *Dataset* penyakit parkinson yang digunakan dalam

penelitian ini diperoleh dari UCI *Machine Learning Database*. Data tersebut menjadi 192 data negatif dan 564 positif.

Berdasarkan sebaran datanya, kumpulan data pada penelitian ini memiliki distribusi kelas yang tidak seimbang. Untuk mengubah *dataset* tidak seimbang ke *dataset* seimbang, digunakan algoritme SMOTE. Kemudian, setelah dikonversi ke distribusi kelas yang seimbang, metode klasifikasi *random forest* digunakan untuk klasifikasi *dataset* penyakit parkinson. Parkinson *dataset* dataset terdiri dari 753 atribut. Klasifikasi *random forest*nya menghasilkan sebesar 87,037% dalam klasifikasi parkinson *dataset*, sedangkan metode *hybrid* yang diusulkan (kombinasi dari SMOTE dan *random forest*) mencapai 94,89% keberhasilan klasifikasi. Hasil yang diperoleh menunjukkan bahwa hasil yang menjanjikan telah dicapai dalam diskriminasi parkinson *dataset* dengan metode *hybrid* ini.

## 2.2. Protein

Protein berasal dari bahasa Yunani “*proteious*” yang berarti pertama atau utama. Protein merupakan makromolekul yang menyusun lebih dari separuh bagian dari sel dan terbentuk dari rantai-rantai asam amino yang tersusun dari atom nitrogen, karbon, dan oksigen, beberapa jenis asam amino yang mengandung sulfur (metionin, sistin dan sistein) yang dihubungkan oleh ikatan *peptide* (Oliveira et al., 2018). Protein tersusun oleh lebih dari 50 asam amino. Menurut Oliveira et al. (2018) protein dapat dikelompokkan menjadi empat tingkat struktur, yaitu struktur primer, sekunder, tersier dan kuarterner. Gambar 1 mendeskripsikan tingkatan struktur protein.



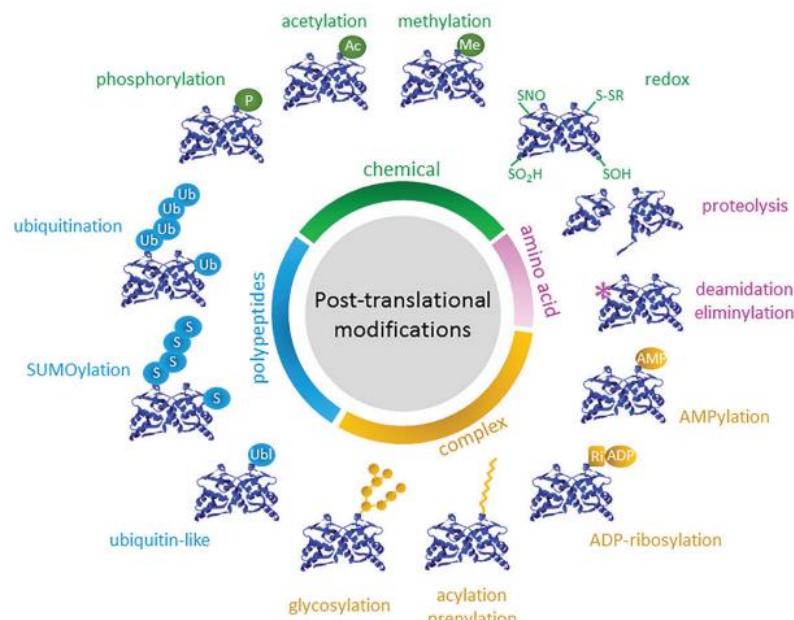
Gambar 1. Tingkatan Struktur Protein (Oliviera et.al., 2018).

Struktur primer merupakan kumpulan ikatan peptida dari asam amino pembentuk protein tersebut. Struktur sekunder berkaitan dengan bentuk dari suatu rantai polipeptida. Struktur sekunder terbentuk dari ikatan hidrogen yang terjadi antara gugus-gugus amino dengan atom hidrogen pada rantai samping asam amino, sehingga membentuk lipatan-lipatan, misalnya membentuk  $\alpha$ -heliks. Struktur tersier merupakan bentuk tiga dimensi dari suatu protein. Interaksi struktur sekunder yang satu dengan struktur sekunder yang lain melalui ikatan hidrogen, ikatan ion, atau ikatan disulfida, misalnya terbentuk rantai double-heliks. Struktur kuarterner merupakan susunan subunit-subunit dalam protein oligomer. Struktur kuarterner melibatkan beberapa peptida sehingga membentuk suatu protein. Pada peristiwa ini, kadang-kadang terselip molekul atau ion lain yang bukan merupakan asam amino.

### 2.3. Post-translational Modification

*Post-translational Modification* (PTM) merupakan proses berubahnya rantai asam amino setelah mengalami biosintesis(Qiu et al., 2016). Proses ini meningkatkan keragaman fungsional protein dengan memperkenalkan gugus fungsi baru ke rantai asam amino protein(Hasan & Ahmad, 2018). *Post-translational Modification* (PTM) diklasifikasikan dalam kategori berdasarkan jenis modifikasi, yaitu bersifat *reversible* dan *irreversible*(Spoel, 2018). Modifikasi kimia

bersifat *reversible* meliputi fosforilasi, asetilasi, metilasi, dan modifikasi berbasis redoks. Modifikasi polipeptida juga *reversible* yang meliputi ubiquitinasi, *sumoylation* dan modifikasi lain oleh polipeptida *ubiquitin-like*. Selanjutnya, modifikasi oleh molekul kompleks bersifat *reversible* meliputi glikosilasi, perlekatan lipid (misalnya asilasi dan prenilasi), *ADP-ribosylation* (Ri-ADP) dan *AMPylation* (AMP). Sedangkan modifikasi asam amino (tanda bintang) atau *polypeptide backbone* tidak *irreversible*, meliputi deamidasi, eliminasi dan pembelahan oleh proteolisis. Modifikasi ini dapat dilihat pada Gambar 2.



Gambar 2. *Post-translational Modification* (Spoel et.al., 2018).

Jenis PTM yang ada pada Gambar 2 akan diuraikan penjelasannya yang dapat dilihat pada Tabel 2.

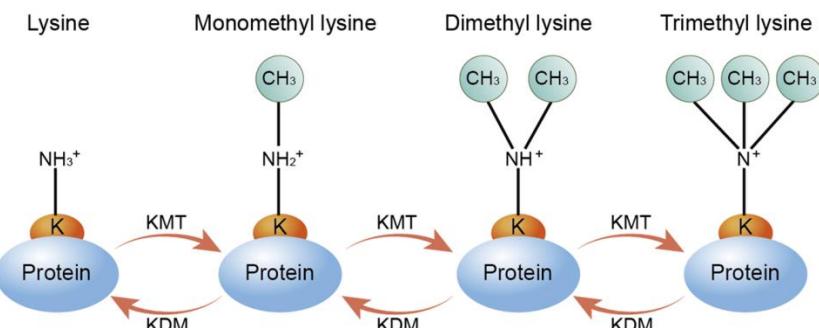
Tabel 2. Jenis PTM dan Penjelasannya (Spoel et.al., 2018)

No	Jenis PTM	Penjelasan
1	Phosphorylation	Penambahan gugus fosfat pada suatu protein

		atau molekul organik lain.
2	Acetylation	Pemberian gugus asetil pada suatu protein atau molekul suatu persenyawaan organik.
3	Methylation	Penambahan suatu gugus metil pada suatu substrat, atau penggantian suatu atom oleh gugus metil.
4	Redox	Perubahan keadaan oksidasi pada suatu atom.
5	Proteolysis	Pemecahan protein menjadi polipeptida atau asam amino yang lebih kecil.
6	Deamidation	Perubahan gugus fungsi amida dalam rantai samping asam amino asparagin atau glutamin menjadi gugus fungsi lain.
7	Eliminylation	Penghilangan fosfat dari fosfotreonin protein menggunakan reaksi eliminasi yang mengubahnya menjadi dehidrobutirin.
8	AMPylation	Penambahan molekul kovalen adenosin monofosfat pada rantai samping asam amino protein.
9	ADPribosylation	Penambahan satu atau lebih gugus ADP-ribosa ke protein.
10	Asylation	Penambahan gugus asil ke suatu senyawa protein.
11	Prenylation	Penambahan molekul hidrofobik ke protein atau biomolekul.
12	Glycylation	Melekatnya karbohidrat donor glikosil pada hidroksil atau gugus fungsi lain dari molekul lain untuk membentuk glikokonjugat.
14	SUMOylation	Perlekatan kovalen dari anggota keluarga protein SUMO (small ubiquitin-like modifier) ke residu lisin dalam protein target.
15	Ubiquitination	Penambahan protein ubikitin ke substrat protein.

## 2.4. Metilasi

Metilasi merupakan salah satu jenis PTM yang merupakan reaksi penggantian suatu atom atau molekul dengan gugus metil (Yulinda et al., 2013). Metilasi protein lisin menyebabkan perubahan minimal dalam ukuran dan status elektrostatik residu lisin. Metilasi lisin memiliki peranan penting dalam menentukan fungsi protein target dalam epigenetic (Luo, 2018). Metilasi protein lisin adalah PTM kritis dan dinamis yang dapat mengatur stabilitas dan fungsi protein. Modifikasi pasca-translasi ini diatur oleh *lysine methyltransferases and lysine demethylases* (Han et al., 2019). Proses metilasi protein lisin terdiri dari enzim menambahkan atau menghapus kelompok metil pada substrat tertentu. Gugus lisin protein dapat menerima hingga tiga gugus metil, menghasilkan *mono-*, *di-*, atau *tri*-metil lisin. Proses metilasi lisin dapat dilihat pada Gambar 3.



Gambar 3. *Lysine Methylation* (Han et.al., 2019).

## 2.5. Imbalanced Data

Masalah ketidakseimbangan kelas adalah masalah pembelajaran mesin yang melibatkan kelas positif dan kelas negatif yang hidup berdampingan namun tidak seimbang di dalam data. Kelas positif hanya mencakup sampel minor yang sering diabaikan dan salah diklasifikasikan ke dalam set sampel mayor. Masalah ketidakseimbangan kelas semakin menarik perhatian dengan penelitian

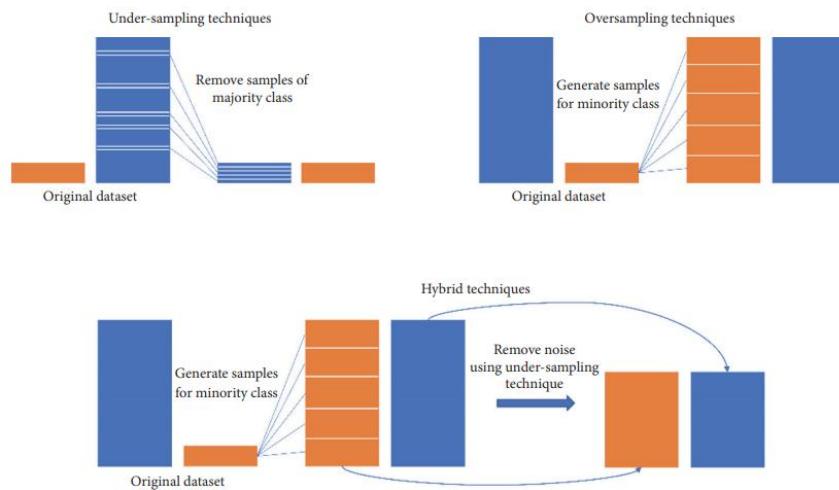
yang mendalam tentang big data karena di dalamnya terdapat lebih banyak data yang tidak seimbang (Fan et al., 2016).

Ketidakseimbangan ukuran sampel di antara label kelas membuat sulit untuk mendapatkan akurasi klasifikasi yang tinggi di banyak bidang ilmiah, termasuk diagnosis medis, bioinformatika, biologi, dan manajemen perikanan. Kesulitan ini disebut sebagai "imbalanced data"(Komori & Eguchi, 2019).Sebuah *dataset* dapat dikatakan tidak seimbang apabila setiap kelas yang ada tidak merepresentasikan data dengan simbang(Chawla, 2009).

Untuk mengatasi masalah ini dapat dilakukan perlakuan dengan beberapa metode, seperti metode *sampling* dan metode *cost-sensitive* (Sekar& Kusuma, 2020). Pada metode *sampling*, dilakukan modifikasi data sehingga menghasilkan distribusi yang seimbang. Ada beberapa teknik yang dapat digunakan, antara lain *random oversampling*, *undersampling* atau *sampling* dengan melakukan generasi dan reduksi data. Metode *cost-sensitive* melakukan pengolahan data dengan mempertimbangkan cost pada kelas positif dan negatif. Sampel yang termasuk kelas positif mendapat *cost* yang lebih besar.

## 2.6. Synthetic Minority Oversampling Technique

*Synthetic Minority Oversampling Technique* (SMOTE) merupakan salah satu metode yang digunakan untuk menangani *imbalanced data problem*. *Synthetic Minority Oversampling Technique* (SMOTE) menggunakan data minoritas dan membuat data sintetis dari data tersebut(Chawla et al., 2002). SMOTE telah digunakan pada banyak penelitian lain dan memiliki performa yang baik dalam kasus data tidak seimbang(Maciejewski & Stefanowski, 2011). Beberapa teknik dalam algoritme SMOTE adalah teknik *undersampling*, *oversampling*, dan *hybrid*(Le et al., 2019). Ilustrasi untuk ketiga teknik tersebut dapat dilihat pada Gambar 4.



Gambar 4. Ilustrasi *oversampling*, *undersampling*, dan *hybrid* teknik (Le et.al., 2019).

Data sintesis tersebut dibuat berdasarkan *k-nearest neighbor*. Prinsip kerja *k-nearest neighbor* (KNN) adalah mencari jarak terdekat antara data yang akan dievaluasi dengan *k* tetangga terdekatnya dalam data latih (training)(Barro et al., 2013). Dengan *k* merupakan banyaknya tetangga terdekat. Pembuatan data sintesis yang berskala numerik berbeda dengan kategorik. Data numerik diukur jarak kedekatannya dengan jarak *Euclidean* sedangkan data kategorik lebih sederhana yaitu dengan nilai modus. Perhitungan jarak antar contoh kelas minor yang peubahnya berskala kategorik dilakukan dengan rumus *Value Difference Metric* (VDM) yang dapat dilihat pada Persamaan 1.

Dengan keterangan sebagai berikut.

$\Delta(X, Y)$  : jarak antara amatan X dengan Y

$w_x, w_y$  : bobot amatan (dapat diabaikan)

$N$  : banyaknya peubah penjelas

$\delta(x_i y_i)^r$  : jarak antar kategori, dengan rumus yang dapat dilihat pada Persamaan 2.

Dengan keterangan sebagai berikut.

$\delta(V_1, V_2)$  : jarak antara nilai  $V_1$  dan  $V_2$

$C_{li}$  : banyaknya  $V_1$  yang termasuk kelas i

$C_1$  : banyaknya nilai 1 terjadi

*i* : banyaknya kelas;  $i = 1, 2, \dots, m$

*n* : banyaknya kategori

$k$  : konstanta (biasanya 1)

Langkah – langkah pembuatan data sintetis untuk :

## 1) Data Numerik

- a. Hitung perbedaan antar vektor utama dengan ketinggian terdekatnya.
  - b. Kalikan perbedaan dengan angka yang diacakdiantara 0 dan 1.
  - c.Tambahkan perbedaan tersebut ke dalam nilaiutama pada vektor utama asal sehingga diperolehvektor utama baru.

## 2) Data Kategorik

- Pilih mayoritas antara vektor utama yang dipertimbangkan dengan k-tetangga terdekatnya untuk nilai nominal. Jika terjadi nilai samamaka pilih secara acak.
  - Jadikan nilai tersebut data contoh kelas buatanbaru.

Berikut ini contoh penggunaan metode SMOTE pada data tidak seimbang yang sederhana. Tabel 3 menunjukkan kelas data yang tidak seimbang.

Tabel 3. Contoh Data Tidak Seimbang

<b>Class</b>	<b>Atribut 1</b>	<b>Atribut 2</b>	<b>Atribut 3</b>
0	4.44	4.29	4.44

Class	Atribut 1	Atribut 2	Atribut 3
0	4.17	3.97	4.5
1	4.35	4.38	4.35
1	4.7	4.38	4.38
1	4.5	4.5	4.35
1	4.36	3.95	3.95
1	4.5	4.5	4.75
1	4.35	4.35	4.17

Konsep awal algoritme SMOTE adalah mengidentifikasi kelas minoritas dan kelas mayoritas berdasarkan jumlah kelas pada data. Setelah kelas minoritas teridentifikasi (*class 0*), selanjutnya SMOTE akan menghitung data sintetik yang akan terbentuk berdasarkan pada rumus  $Ty = (N/100) * Tx$ . Variabel  $Tx$  merupakan jumlah data kelas minoritas,  $Ty$  adalah kelas minoritas akhir (setelah ditambah data sintetis), dan  $N$  adalah persentasi oversampling dengan nilai kelipatan 100. Sebagai contoh, persentasi oversampling yang diberikan adalah 300, sehingga jumlah kelas minoritas akan dinaikkan setara dengan kelas mayoritas yakni 6 dengan perhitungan  $Ty = (300/100) * 2$ . Selanjutnya, komputer akan menghitung jumlah  $k$  terdekat (nilai  $k$  berdasarkan masukkan) dari setiap data kelas minoritas sampai jumlah  $Ty$  kali dan menyimpan data dalam *array*. Kemudian, komputer akan memilih secara acak data minoritas, dan menentukan data minoritas terdekat lainnya (sejumlah  $k$ ) yang dihitung dengan persamaan jarak *euclidean*. Dari jumlah tetangga terdekat yang terdeteksi, maka algoritme SMOTE memilih acak data minoritas tertentu dan akan membentuk data baru (data sintetis) antara dua data tersebut.

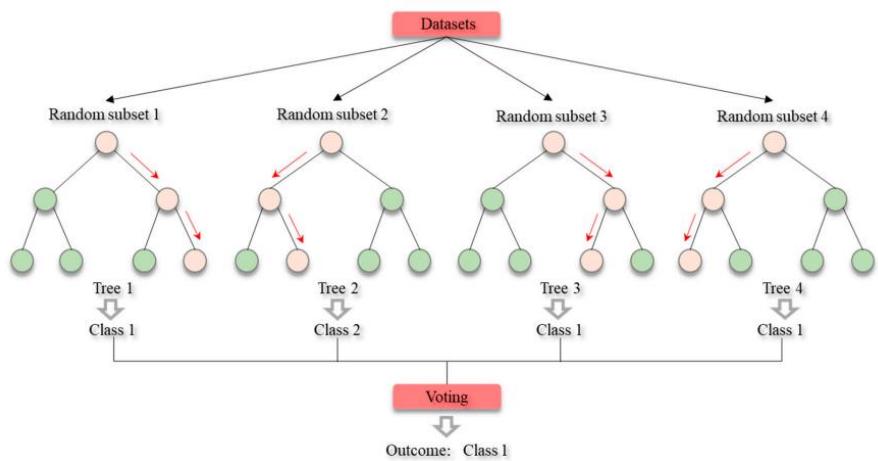
## 2.7. Random Forest

*Random forest* adalah salahsatu metode klasifikasi data. Metode ini pertama kali diusulkan oleh Breiman pada tahun 2001. Dimana terdapat proses agregasi pohon keputusan, yaitu penggabungan pohon dalam *random forest* untuk mencari nilai terbesar berdasarkan hasil

*voting*(Xiao et al., 2012). Metode ini banyak digunakan karena menghasilkan kesalahan yang lebih rendah, memberikan akurasi yang bagus dalam klasifikasi, dapat menangani data pelatihan yang jumlahnya sangat besar, serta efektif untuk mengatasi data yang tidak lengkap(Primajaya & Sari, 2018).

Langkah-langkah rinci dari *Random forest*(Yang et al., 2019) adalah sebagai berikut.

1. Pilih beberapa data di training set sebanyak  $k$ .
2. Membuat *decision tree* dari  $k$  data yang sudah dipilih sebelumnya.
3. Pilih jumlah  $n$ -*tree* (kumpulan pohon-pohon) yang ingin dibuat. Selanjutnya mengulangi langkah 1 dan 2. Intinya, terus membuat *decision tree* sebanyak-banyaknya (umumnya sebanyak 200 kali, 300, 500, dan seterusnya).
4. Setelah sejumlah besar pohon dihasilkan, data baru diprediksi dengan menggabungkan hasil semua pohon, dengan strategi voting mayoritas.



Gambar 5. Ilustrasi alur kerja metode *random forest* (Yang et.al., 2019).

Gambar 5 mengilustrasikan alur kerja metode *random forest*. Saat membentuk sebuah pohon klasifikasi, pendekatan yang dilakukan adalah dengan memisahkan sebuah masalah menjadi sub masalah.

Keputusan pada simpul teratas akan dipisahkan menjadi dua simpul dimana kedua simpul tersebut terdiri dari pernyataan benar dan salah.

Ada beberapa algoritma yang digunakan untuk memilih keputusan misalnya adalah indeks Gini yang didefinisikan pada Persamaan 3.

$$G = \sum_{i=1}^I p_{mi}(1 - p_{mi}) .....(3)$$

Dimana  $p_{mi}$  adalah proporsi pengamatan pada kelas  $i$  dan simpul ke  $m$ . Persamaan 3 merupakan ukuran keheterogenan simpul. Himpunan bagian yang dihasilkan dari proses pemilihan harus lebih homogen dibandingkan simpul induknya. Kemudian langkah selanjutnya adalah menentukan kriteria *goodness of split*. Cara yang dilakukan adalah dengan mengoptimalkan fungsi indeks Gini.

Selain menggunakan indeks Gini, dalam menentukan pohon keputusan juga dapat dilakukan dengan cara menghitung nilai entropy sebagai penentu tingkat ketidakmurnian atribut dan nilai information gain. Menghitung nilai entropy dapat menggunakan rumus seperti persamaan 4 untuk satu atribut, persamaan 5 untuk dua atribut menggunakan tabel frekuensi, dan menentukan nilai *information gain* menggunakan Persamaan 6.

$$\text{Entropy}(S) = \sum_{i=1}^C - p_i \log_2 p_i .....(4)$$

Dimana :

- $S$  = himpunan dataset
- $C$  = jumlah kelas
- $p_i$  = probabilitas frekuensi kelas ke- $i$  dalam dataset

$$\text{Entropy}(T, X) = \sum_{c \in X} p_c E_c .....(5)$$

Dimana :

$(T,X)$  = atribut T dan atribut X

$P_c$  = probabilitas kelas atribut

$E_c$  = Nilai Entropy kelas atribut

Dimana :

$S$  = himpunan dataset

*A* = atribut

$|S_i|$  = jumlah sampel untuk nilai i

$|S|$  = jumlah seluruh sampel data

*Entropy* ( $S_i$ ) = entropy untuk sampel yang memiliki nilai i

## 2.8. Feature Extraction

*Feature extraction* merupakan proses mengubah data mentah menjadi fitur numerik yang dapat diproses sambil mempertahankan informasi dalam kumpulan data asli(Guyon & Elisseeff, 2006).Teknik ini merupakan proses transformasi yang mengekstrak nilai numerik dari urutan tekstual(Ismail et al., 2016). Metode ini mengekstraksi dua puluh asam amino dari serangkaian fitur dan penggabungan sejumlah informasi posisi asam amino dalam sekuens. Fitur diekstraksi dengan mempertimbangkan probabilitas kejadian asam amino di berbagai urutan. Dalam penelitian ini menggunakan beberapa *feature extraction*yaitu CTD, AAIndex, *Hydrophobicity*, dan PseAAC.

Protein *descriptor* adalah tools untuk melakukan ekstraksi fitur pada urutan protein. Protein deskriptor yang digunakan untuk melakukan *feature extraction* ini menghasilkan ekstraksi fitur *static length* dan *dynamic length*. *Static length* adalah protein *descriptor* yang nilai fiturnya tidak berubah sepanjang apapun *sequences*-nya. *Dynamic length* adalah protein *descriptor* yang nilai fiturnya berubah-ubah sesuai panjang, urutan dan parameter yang digunakan (Xu, Y. et al., 2020).

Berikut ini penjelasan dari *feature extraction* yang digunakan pada penelitian ini.

### **2.8.1. Composition, Transition, and Distribution (CTD)**

*Composition*, *Transition*, dan *Distribution* merupakan fitur ekstraksi yang merepresentasikan sifat *physicochemical* asam amino. CTD menghasilkan 21 fitur untuk setiap PCP (*Physicochemical Properties*) (Hou et al., 2020). *Composition* (C) merupakan pembagian jumlah asam amino sifat tertentu ( $N_e$ ) dengan jumlah total asam amino keseluruhan ( $N$ ). *Transition* (T) mengukur perbandingan perubahan sifat asam amino dengan asam amino dari kelas yang berbeda. *Transition* (T) menghitung jumlah peptida yang dikode ( $N_{nm} + N_{mn}$ ) berbanding panjang *sequence* ( $N$ ) dikurangi satu. *Distribution* (D) menggambarkan distribusi setiap atribut terhadap panjang rantai *sequence*. Dalam melakukan ekstraksi fitur CTD, dapat menggunakan fungsi *featureCTD* pada *package BioSeqClass*. Kode program untuk CTD dapat dilihat pada Pseudocode 1.

```
featureCTD(seq_pos, class = aaClass("aaV"))
```

Pseudocode 1. Kode program *featureCTD*.

Pada Pseudocode 1, parameter *seq* merupakan vektor *string* untuk urutan protein, *class* merupakan daftar untuk kelas sifat biologi, sedangkan *aaClass* merupakan daftar kelompok asam amino yang tergantung pada sifat fisik kimianya. Misal, diberikan contoh *sequence* “MPAESGKRFKPSKYV” dan “MDFEDDYTHSACRNT” sebagai *input* data. Setelah data diolah menggunakan kode program yang ada pada Pseudocode 1, maka akan menghasilkan *output* yang dapat dilihat pada Tabel 4.

Tabel 4. Output penggunaan *featureCTD*

Sequence	C_small	T_small_large	D_medium1st	...
MPAESGKRFKPSKYV	0.4	0.28	0.13	...
MDFEDDYTHSACRNT	0.53	0.5	0.13	...

### 2.8.2. AA Index

AAindex merupakan basis data numerik sekuen protein yang berisi berbagai sifat *physicochemical* dan *biochemical* asam amino(Kawashima et al., 2008). Menurut Kawashima, basis data AAindex terbagi menjadi tiga, yaitu AAindex1, AAindex2, dan AAindex3. AAindex1 terdapat 544 indeks asam amino. AAindex2 terdapat 94 matriks asam amino yang terbagi menjadi 67 matriks simetris dan 27 matriks non-simetris. AAindex3 terdapat 47 matriks asam amino yang terdiri dari 44 matriks simetris dan 3 matriks non-simetris (Kawashima et al., 2008).

Dalam mengukur sifat *physicochemical* dan *biochemical* pada AAindex, dapat menggunakan fitur *featureAAindex* pada *package* BioSeqClass. *FeatureAAindex* menghitung atribut pada asam amino berdasarkan basis data AAindex dalam bentuk matriks. Kode program untuk *featureAAindex* dapat dilihat pada Pseudocode 2.

```
featureAAindex(seq,aaindex.name="all")
```

Pseudocode 2. Kode program *featureAAindex*.

Pada Pseudocode 2, parameter *seq* merupakan vector string untuk urutan protein, *aaindex.name* merupakan string nama sifat fisikokimia dan biokimia yang ada di AAIndex. Misal, diberikan contoh *sequence* “MDSLAAPQDRLVEQL” dan “MGSGPIDPKELLKGL” sebagai data *input*. Setelah data diolah

menggunakan kode program yang ada pada Pseudocode 2 dengan menggunakan `aaindex.name="ANDN920101"`, maka akan menghasilkan *output* yang dapat dilihat pada Tabel 5.

Tabel 5. *Output* penggunaan *featureAAindex*

Sequence	ANDN920101_1	ANDN920101_2	ANDN920101_n
MDSLAAPQDRLVEQL	4.52	4.76	...
MGSGPIDPKELLKGL	4.52	3.97	...

### 2.8.3. *Pseudo Amino Acid Composition (PseAAC)*

*Pseudo Amino Acid Composition (PseAAC)* merupakan metode prediksi lokalisasi subseluler dan prediksi tipe membran protein (Liu et al., 2005). PseAAC berisi serangkaian informasi lebih dari 20 komponen, dimana 20 komponen pertama merepresentasikan urutan asam amino. Komponen tambahan berikutnya merupakan gabungan informasi urutan komponen asam amino semu(Liu et al., 2005). Untuk mengekstraksi asam amino menggunakan metode PseAAC dapat menggunakan fungsi *featurePseudoAACComp* pada *library BioSeqClass*. Fungsi *featurePseudoAACComp* mengukur komposisi asam amino yang dikodekan dalam 20+d dimensi. Dimensi 20+d mewakili 20 jenis asam amino dan komponen tambahan asam amino semu. Kode program untuk *featurePseudoAACComp* dapat dilihat pada Pseudocode 3.

```
featurePseudoAACComp(seq,d)
```

Pseudocode 3. Kode program *featurePseudoAACComp*.

Pada Pseudocode 3, parameter *seq* merupakan vektor *string* untuk urutan protein dan *d* merupakan parameter bertipe integer dengan nilai (*d*>=1). Nilai maksimalnya tidak boleh sama/melebihi panjang *sequence* yang diberikan. Misal, diberikan contoh *sequence*

“MSKSASPKEPEQLRK” dan “MASATDSRYGQKESS” sebagai data *input*. Setelah data diproses menggunakan kode program pada Pseudocode 3, maka akan menghasilkan *output* yang dapat dilihat pada Tabel 6.

Tabel 6. *Output* penggunaan *featurePseAAComp*

Sequence	PAC:R	PAC:K	PAC:y	PAC:1	PAC:n
MSKSASPKEPEQLRK	0.05	0.15	...	0.4	...
MASATDSRYGQKESS	0.05	0.05	...	0.8	...

#### 2.8.4. *Hydrophobicity*

*Hydrophobic* merupakan sifat protein yang menolak atau anti air yang mana permukaan material dengan air akan terbentuk sudut lebih besar. Untuk mengukur persentase tersebut, dikenal istilah skala *hydrophobicity*. Skala *hydrophobicity* merupakan nilai yang menentukan *hydrophobic* (menolak air) atau *hydrophilic* (menarik air) residu asam amino(Kyte & Doolittle, 1982). Semakin positif nilai, maka semakin *hydrophobic* suatu residu. Jika menunjukkan negatif, maka residu tersebut *hydrophilic*. Analisis *hydrophobicity* digunakan untuk memahami identifikasi struktur dasar sekunder protein.

Untuk mengekstraksi fitur *hydrophobicity*, dapat menggunakan *featureHydro* pada *library BioSeqClass*. *FeatureHydro* sendiri menghasilkan sebanyak 15 *feature*. Fungsi *featureHydro* akan mengembalikan nilai pengukuran efek *hydrophobic* menggunakan parameter *hydro.method*. Kode program untuk *featureHydro* dapat dilihat pada Pseudocode 4.

```
featureHydro(seq,hydro.method)
```

Pseudocode 4. Kode program *featureHydro*.

Pada Pseudocode 4, parameter *seq* merupakan vektor *string* untuk urutan protein dan *hydro.method* adalah parameter bertipe *string* yang merupakan pengkodean efek hidrofobik protein berupa salah satu string "kpm" atau "SARAH1". Misal, diberikan contoh *sequence* "MGSGPIDPKELLKGL" dan "MTHSPATSEDEERHS" sebagai data *input*. Setelah data diproses menggunakan kode program pada Pseudocode 4, maka akan menghasilkan *output* yang dapat dilihat pada Tabel 7.

Tabel 7. *Output* penggunaan *featureHydro*

Sequence	H:1	H:2	H:...	H:15
MGSGPIDPKELLKGL	2.35	4.04	...	4.04
MTHSPATSEDEERHS	2.35	4.92	...	4.92

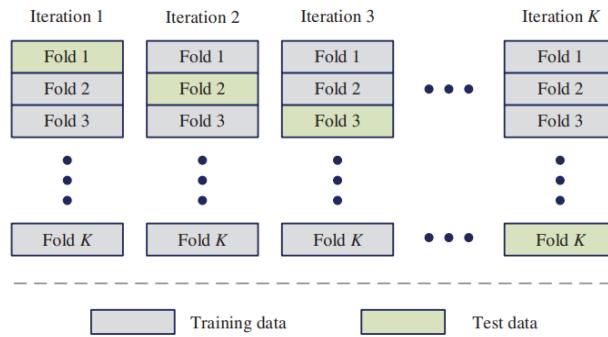
## 2.9. Cross-validation

*Cross-validation* adalah metode statistik yang dapat digunakan untuk melatih dan mengevaluasi kinerja model atau algoritma dimana data dipisahkan menjadi dua segmen yaitu data latih dan data uji (Refaeilzadeh et al., 2016). Model atau algoritma dilatih oleh segmenlatih (*training*) dan divalidasi oleh segmen uji (*testing*). Selanjutnya pemilihan jenis *cross-validation* dapat didasarkan pada ukuran *dataset*. Biasanya *k-fold cross-validation* digunakan karena dapat mengurangi waktu komputasi dengan tetap menjaga keakuratan estimasi.

### 2.9.1. K-foldCross Validation

Teknik *k-foldCross Validation* (KCV) adalah salah satu pendekatan yang paling banyak digunakan oleh peneliti untuk pemilihan model dan estimasi kesalahan pengklasifikasi (Rodriguez et al., 2009). Dalam *k-fold cross-validation*, pertama-tama data dipartisi menjadi *k* segmen atau lipatan berukuran sama (atau hampir sama).

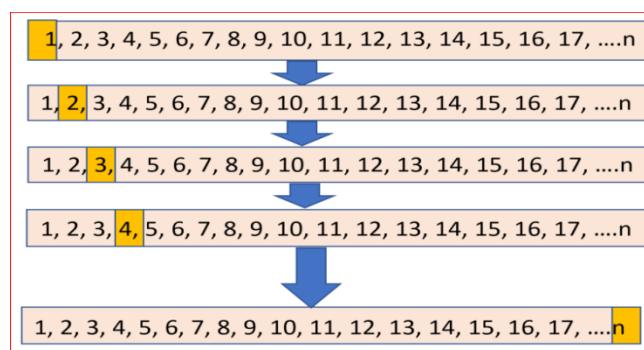
Selanjutnya iterasi  $k$  dari *training* dan *testing* dilakukan sedemikian rupa sehingga dalam setiap iterasi lipatan, data yang berbeda diadakan untuk validasi, sedangkan lipatan  $k-1$  yang tersisa digunakan untuk *training*(Ren et al., 2019). Ilustrasi dari *k-fold Cross Validation* dapat dilihat pada Gambar 6.



Gambar 6. *K-Fold Cross Validation* (Ren et.al., 2019).

### 2.9.2. *Leave-One-Out Cross-Validation*

*Leave-One-Out Cross-Validation* (LOOCV) merupakan turunan dari metode *k-Fold Cross Validation*, di mana yang dipilih adalah sebesar jumlah data. Misalkan *dataset* yang memiliki  $n$  *sample*, maka percobaan dilakukan sebanyak  $n$  kali. Setiap percobaan menggunakan sebanyak  $n-1$  *sample* untuk data *training* dan sisanya untuk *testing*. Ilustrasi dari *Leave-One-Out Cross-Validation* dapat dilihat pada Gambar 7.



Gambar 7. *Leave-One-Out Cross Validation*.

## 2.10. Performance Classification

Untuk mengukur *performance classification*, dibutuhkan *confusion matrix* yang merupakan tabel pengukuran performa untuk masalah klasifikasi *machine learning* dimana keluaran dapat berupa dua kelas atau lebih(Lewis & Brown, 2001). *Confusion matrix* adalah tabel dengan 4 kombinasi berbeda dari nilai prediksi dan nilai aktual. Contoh matriks evaluasi untuk dua kelas data dapat dilihat pada Tabel 8.

Tabel 8. Evaluasi Matriks Dua Kelas Data

		<b>Actual Value</b>	
		TN (True Negative)	FN (False Negative)
<b>Predicted Value</b>	FP (False Positive)	TP (True Positive)	

Pada penelitian ini, terdapat lima parameter akurasi yang digunakan untuk mengukur kinerja model klasifikasi yaitu : *accuracy*, *precision*, *recall*, *specificity* dan *f-measure*. Berikut ini penjelasan dan rumus perhitungan dari masing-masing parameter tersebut.

### 2.10.1. Accuracy (*Error Rate*)

*Accuracy* merupakan salah satu cara untuk mengukur seberapa sering algoritma mengklasifikasikan titik data dengan benar. Akurasi adalah jumlah titik data yang diprediksi dengan benar dari semua titik data(Powers, 2020). Dengan kata lain, ini didefinisikan sebagai jumlah *truepositive* dan *truenegative* dibagi dengan jumlah *true positive*, *truenegative*, *falsepositive*, dan *falsenegative*. *Truepositive* atau *truenegative* adalah titik data yang algoritma klasifikasikan dengan benar sebagai benar atau salah. Sebaliknya, *falsepositive* atau *falsenegative* adalah titik data yang salah

diklasifikasikan oleh algoritma. Untuk lebih jelasnya, dapat dilihat pada Persamaan 7.

$$Accuracy = \frac{(TP + TN)}{(TP + FP + FN + TN)} \dots \dots \dots (7)$$

### 2.10.2. *Precision*

*Precision* merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif. Nilai ini menggambarkan akurasi antara data yang diminta dengan hasil prediksi yang diberikan oleh model(Powers, 2020). Hal ini dapat dideskripsikan dalam bentuk persamaan yang dapat dilihat pada Persamaan 8.

### 2.10.3. Recall(*Sensitivity*)

*Recall* atau *sensitivity* merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. Rasio ini menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi(Powers, 2020). Hal ini dapat dilihat pada Persamaan 9.

#### 2.10.4. *Specificity*

*Specificity* merupakan rasio kebenaran memprediksi negatif dibandingkan dengan keseluruhan data negatif (Powers, 2020). Agar lebih jelas, dapat dilihat pada Persamaan 10.

### **2.10.5. *F-measure***

*F-measure* merupakan perbandingan rata-rata *precision* dan *recall* yang dibobotkan(Powers, 2020). Perhitungannya dapat dilihat pada Persamaan 11.

$$F - measure = \frac{2 * (Recall * Precision)}{(Recall + Precision)}.....(11)$$

### **III. DATA DAN METODOLOGI**

#### **3.1. Waktu dan Tempat**

##### **3.1.1. Tempat Penelitian**

Penelitian ini dikerjakan di Laboratorium Rekayasa Perangkat Lunak (RPL), Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung.

##### **3.1.2. Waktu dan Jadwal (Rencana) Penelitian**

Penelitian ini dilakukan pada bulan November 2021 hingga bulan April 2022 melalui tiga tahapan, yaitu perancangan penelitian, pelaksanaan penelitian, dan evaluasi penelitian. Rencana alur dan waktu secara lebih rinci dapat dilihat pada Tabel 9.

Tabel 9. Rencana Alur dan Waktu Penelitian

Tabel 2 menunjukkan rencana kegiatan dalam penelitian yang terdiri dari tiga tahapan sebagai berikut.

1. Perancangan Penelitian

Tahap pertama pada perancangan ini dimulai dengan kegiatan mengumpulkan data, menentukan tema dan judul penelitian. Kemudian, dilanjutkan dengan merancang metode dan alur penelitian yang digambarkan dalam bentuk *flowchart* atau diagram alir rancangan penelitian.

2. Pelaksanaan Penelitian

Setelah data diperoleh, selanjutnya melakukan *cleaning* data guna membersihkan fitur data yang tidak perlu atau sama. Kegiatan dilanjutkan dengan melakukan ekstraksi fitur, pemodelan SMOTE, pemodelan dan prediksi *random forest*, serta yang terakhir pada tahapan ini yaitu melakukan validasi menggunakan *K-Fold Cross Validation*. Tahap pelaksanaan penelitian ini membutuhkan waktu kurang lebih 14 minggu.

3. Evaluasi Penelitian

Tahap evaluasi merupakan suatu kegiatan menganalisis hasil pelaksanaan penelitian menggunakan akurasi. Pada tahap ini digunakan *confusion matrix* sebagai alat ukur performa masalah klasifikasi *machine learning*.

### 3.2. Data dan Alat

#### 3.2.1. Data

Data yang digunakan pada penelitian ini bersumber dari penelitian Hasan & Ahmad (2018). *Dataset* ini merupakan data metilasi *sequence* protein lisin pada manusia yang berjumlah 1172 dengan 1000 data positif dan 172 data negatif. Setiap *sequence* memiliki panjang yang berbeda-beda, panjang dari masing-masing *sequences* dataset tersebut antara 38 sampai 1105 panjang *sequences*. Data protein terbagi menjadi dua yaitu data uji dan data latih. Data ini diperoleh dari situs web <http://www.uniprot.org/> dengan memberikan berbagai batasan seperti *experimental assertion* untuk kolom *evidence*, hanya mempertimbangkan protein *sequences* manusia, dan menggunakan kata kunci *methylation* pada opsi pencarian. Bentuk *dataset* yang digunakan pada penelitian ini dapat dilihat pada Tabel 10.

Tabel 10. Bentuk Dataset Metilasi Lisin

Entry	Sequence
Q9C0B5	MPAESGKRFKPSKYVPVAAAIFLVGATTLFFAFT
Q5BKZ1	MDFEDDYTHSACRNTYQGFNGMDRDYGPGSYGG
P47914	MAKSKNHTTHNQSRKWHRNGIKKPRSQRYESLKG

#### 3.2.2. Alat

##### 3.2.2.1. Hardware

*Hardware* yang digunakan adalah sebagai berikut.

- a. *Processor* : Intel Core i5-8250U CPU (6M Cache, up to 3.40 GHz)
- b. RAM : 8.00 GB, DDR4, 2133MHz
- c. *Storage* : HDD Toshiba MQ04ABF100 1 TB 5400 RPM
- d. *Network interface* : Qualcomm Atheros QCA9377 Wireless Network Adapter, 433 Mbps, 2.4 GHz

- e. *Video Graphics Array (VGA)* : Nvidia GeForce 930MX VRAM 2GB

### **3.2.2.2. Software**

*Software* yang digunakan adalah sebagai berikut.

- a. *Operating system* : Microsoft Windows 10 Home Single Language 64-bit Version 10.0.19042 Build 19042
- b. *Tools* : R Programming 3.6.3& R Studio 1.3.1093
- c. *Library*

Beberapa *library* yang akan digunakan adalah sebagai berikut.

#### *1. LibraryrandomForest 4.6-14*

*LibraryrandomForest* merupakan *package* yang digunakan untuk melakukan pemodelan prediksipada klasifikasi dan regresi berdasarkan algoritme *random forest*.

#### *2. Librarysmotefamily 0.4.1*

*Library smotefamily* menyediakan fitur fungsi dan konsep SMOTE pada bahasa pemrograman R, termasuk fungsi algoritme SMOTE.

#### *3. LibraryCaret 6.0-84*

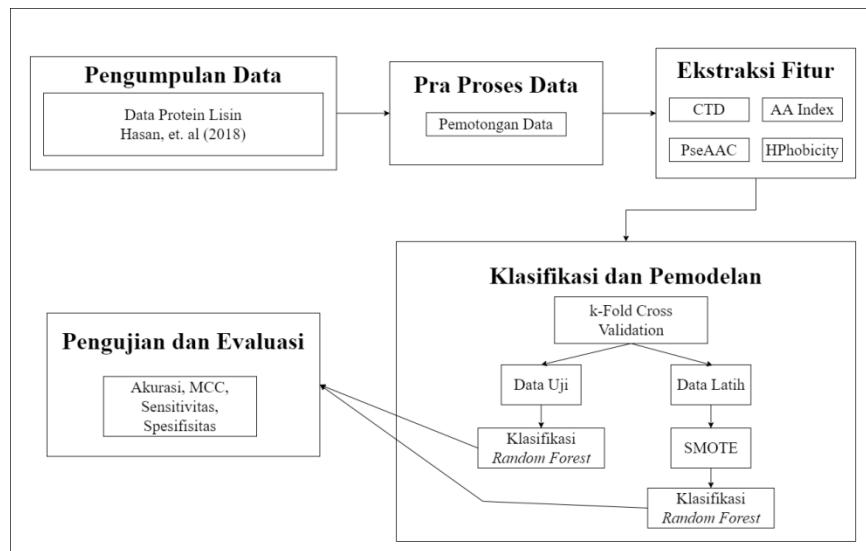
*Library Caret* merupakan *package* yang digunakan untuk melakukan pemodelan prediksi terhadap klasifikasi dan percobaan regresi suatu data. *Library* ini digunakan untuk mengukur hasil klasifikasi menggunakan *confusion matrix*.

#### *4. Library Protr 1.6-2*

*Library* ini menyediakan layanan ekstraksi fitur untuk mengubah *dataset* sebelumnya menjadi data yang data dikelola.

### 3.3. Metodologi

Alur kerja penelitian ini didasari oleh penelitian yang berjudul “*mLysPTMpred: Multiple Lysine PTM Site Prediction Using Combination of SVM with Resolving Data Imbalance Issue*”(Hasan & Ahmad, 2018). *Flowchart* atau diagram alir rencana penelitian dapat diilustrasikan pada Gambar 8.



Gambar 8. Diagram Alir Rencana Penelitian.

#### 3.3.1. Pengumpulan Data

*Dataset* metilasi *sequence* protein lisinini didapatkan dari penelitian Hasan & Ahmad (2008) yang merupakan *dataset benchmark*dalam prediksi metilasi protein lisin. *Dataset* ini terdiri dari 1172 jumlah data dengan 1000 data positif dan 172 data negatif.

#### 3.3.2. Pra-proses Data

Pra-proses data melalui tahap pembersihan data. Tahap pembersihan data merupakan proses menghilangkan urutan protein yang mengandung huruf X.

### **3.3.3. Ekstraksi Fitur**

Pada tahap ekstraksi fitur, dilakukan eksperimen agar data protein sebelumnya menjadi data yang dapat dikelola yaitu mengubah data *string* menjadi numerik. Ekstraksi fitur yang digunakan dalam penelitian ini adalah *Composition, Transition, and Distribution* (CTD), AAindex, dan *Hydrophobicity* dan PseAAC. Hasil dari keempat fitur ekstraksi ini digabung menjadi satu yang kemudian akan diproses ke tahap selanjutnya.

### **3.3.4. Klasifikasi dan Pemodelan**

Pemodelan dilakukan dengan menggunakan metode SMOTE dan metode klasifikasi *random forest*. Pemodelan SMOTE bertujuan untuk menyeimbangkan kelas minoritas dan mayoritas. Setelah seimbang, data dibagi menjadi dua bagian, yaitu data uji dan data latih. Presentase pembagian data uji dan data latih sebesar 30% untuk data uji dan untuk 70% data latih.

### **3.3.5. Pengujian dan Evaluasi**

Pemodelan yang telah dilakukan pada tahap sebelumnya hingga menghasilkan prediksi, selanjutnya dilakukan pengujian menggunakan *confusion matrix*. Komponen pengukuran yang digunakan adalah *accuracy*, *sensitivity*, *specificity*, dan MCC.

## V. PENUTUP

### 5.1. Simpulan

Adapun simpulan pada penelitian klasifikasi ketidakseimbangan data metilasi protein lisin menggunakan algoritme SMOTE dan metode *random forest* adalah sebagai berikut :

1. Setelah dilakukan pengujian, ketidakseimbangan data pada metilasi *sequence* protein lisin ini mempengaruhi hasil prediksi klasifikasi. Berikut hasil klasifikasi yang didapatkan.
  - a. Pada skenario pertama yaitu pembagian data 80% training dan 20% testing, hasil akurasi yang didapatkan setelah menggunakan SMOTE yaitu sebesar 95,65%. Sedangkan tanpa penerapan SMOTE, didapatkan akurasi sebesar 90%.
  - b. Pada skenario kedua yaitu pembagian data 90% training dan 10% testing, hasil akurasi yang didapatkan setelah menggunakan SMOTE yaitu sebesar 94,77%. Sedangkan tanpa penerapan SMOTE, didapatkan akurasi sebesar 90,69%.
2. Setelah dilakukan perbandingan dengan penelitian sebelumnya, hasil penelitian sebelumnya memiliki nilai akurasi yang lebih rendah. Penelitian yang dilakukan oleh (Hasan et al., 2018) memiliki akurasi 11,92% lebih rendah daripada penelitian menggunakan klasifikasi *random forest* dan metode SMOTE.

## 5.2. Saran

Adapun saran yang dapat diberikan pada penelitian ini adalah sebagai berikut:

1. Untuk mengurangi kesalahan pengklasifikasian karena perbandingan antara data positif dan negatif yang cukup besar, dapat menggunakan teknik seleksi fitur yang diimplementasikan dengan algoritme SMOTE atau sejenisnya seperti DBSMOTE, SLS, dan ADASYN dengan metode klasifikasi tertentu.
2. Dapat menggunakan metode klasifikasi lain seperti *XGboost*, *K-Nearest neighbors*, atau *Artificial Neural Network* (ANN) untuk mendapatkan pembanding sebagai acuan hasil yang lebih baik dalam menangani ketidakseimbangan data.

## DAFTAR PUSTAKA

- Barro, R. A., Sulvianti, I. D., & Afendi, F. M. 2013. *Penerapan Synthetic Minority Oversampling. 1(1)*.
- Chairunisa, R., Adiwijaya, & Astuti, W. 2020. Perbandingan CART dan Random Forest untuk Deteksi Kanker berbasis Klasifikasi Data Microarray. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 4(5), 805–812.
- Chawla, N. V. 2009. Data Mining for Imbalanced Datasets: An Overview. In *Data Mining and Knowledge Discovery Handbook*. Springer.
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321–357.
- Fan, J., Niu, Z., Liang, Y., & Zhao, Z. 2016. Probability model selection and parameter evolutionary estimation for clustering imbalanced data without sampling. *Neurocomputing*, 211, 172–181.
- Guyon, I., & Elisseeff, A. 2006. An introduction to feature extraction. *Studies in Fuzziness and Soft Computing*, 207, 1–25.
- Han, D., Huang, M., Wang, T., Li, Z., Chen, Y., Liu, C., Lei, Z., & Chu, X. 2019. Lysine methylation of transcription factors in cancer. *Cell Death and Disease*, 10(4).
- Hasan, M. A. M., & Ahmad, S. 2018. mLysPTMpred: Multiple Lysine PTM Site Prediction Using Combination of SVM with Resolving Data Imbalance Issue. *Natural Science*, 10(09), 370–384.
- Hou, R., Wu, J., Xu, L., Zou, Q., & Wu, Y. J. 2020. Computational prediction of protein arginine methylation based on composition–transition–distribution

- features. *ACS Omega*, 5(42), 27470–27479.
- Indrawati, A., Ilmu, L., Indonesia, P., & Diabetes, P. I. 2021. *Penerapan Teknik Kombinasi Oversampling Dan Undersampling Hybrid Oversampling and Undersampling Techniques To Handling Imbalanced Dataset*. 4(1), 38–43.
- Ismail, H. D., Smith, M., & Kc, D. B. 2016. *FEPS: Feature Extraction from Protein Sequences webserver FEPS: Feature Extraction from Protein Sequences webserver*. June.
- Kasanah, A. N., Muladi, M., & Pujiyanto, U. 2019. Penerapan Teknik SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Objektivitas Berita Online Menggunakan Algoritma KNN. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 3(2), 196–201.
- Kawashima, S., Pokarowski, P., Pokarowska, M., Kolinski, A., Katayama, T., & Kanehisa, M. 2008. AAindex: Amino acid index database, progress report 2008. *Nucleic Acids Research*, 36(SUPPL. 1), 202–205.
- Komori, O., & Eguchi, S. 2019. *Introduction to Imbalanced Data*. 0, 1–10.
- Kyte, J., & Doolittle, R. F. 1982. A simple method for displaying the hydropathic character of a protein. *Journal of Molecular Biology*, 157(1), 105–132.
- Le, T., Vo, M. T., Vo, B., Lee, M. Y., & Baik, S. W. 2019. A Hybrid Approach Using Oversampling Technique and Cost-Sensitive Learning for Bankruptcy Prediction. *Complexity*.
- Lee, D. Y., Teyssier, C., Strahl, B. D., & Stallcup, M. R. (2005). Role of protein methylation in regulation of transcription. *Endocrine Reviews*, 26(2), 147–170.
- Lewis, H. G., & Brown, M. 2001. A generalized confusion matrix for assessing area estimates from remotely sensed data. *International Journal of Remote Sensing*, 22(16), 3223–3235.
- Liu, H., Yang, J., Wang, M., Xue, L., & Chou, K. C. 2005. Using Fourier

- spectrum analysis and pseudo amino acid composition for prediction of membrane protein types. *Protein Journal*, 24(6), 385–389.
- Luo, M. 2018. Chemical and Biochemical Perspectives of Protein Lysine Methylation [Review-article]. *Chemical Reviews*, 118(14), 6656–6705.
- MacIejewski, T., & Stefanowski, J. 2011. Local neighbourhood extension of SMOTE for mining imbalanced data. *IEEE SSCI 2011: Symposium Series on Computational Intelligence - CIDM 2011: 2011 IEEE Symposium on Computational Intelligence and Data Mining*, 104–111.
- Martin, C., & Zhang, Y. 2005. The diverse functions of histone lysine methylation. *Nature Reviews Molecular Cell Biology*, 6(11), 838–849.
- Martins de Oliveira, E., Estrella, J. C., Delbem, A. C. B., Nunes, L. H., Shishido, H. Y., & Reiff-Marganiec, S. 2018. Selection of computational environments for PSP processing on scientific gateways. *Helion*, 4(7).
- Polat, K. 2019. A hybrid approach to Parkinson disease classification using speech signal: The combination of SMOTE and random forests. *2019 Scientific Meeting on Electrical-Electronics and Biomedical Engineering and Computer Science, EBBT 2019*, 1–3.
- Powers, D. M. W. 2020. *Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation*. 37–63.
- Primajaya, A., & Sari, B. N. 2018. Random Forest Algorithm for Prediction of Precipitation. *Indonesian Journal of Artificial Intelligence and Data Mining*, 1(1), 27.
- Qiu, W. R., Sun, B. Q., Xiao, X., Xu, Z. C., & Chou, K. C. 2016. iPBM-mLys: Identifying multiple lysine PTM sites and their different types. *Bioinformatics*, 32(20), 3116–3123.
- Refaeilzadeh, P., Tang, L., Liu, H., Angeles, L., & Scientist, C. D. 2016. Encyclopedia of Database Systems. *Encyclopedia of Database Systems*.

- Ren, Q., Li, M., & Han, S. 2019. Tectonic discrimination of olivine in basalt using data mining techniques based on major elements: a comparative study from multiple perspectives. *Big Earth Data*, 3(1), 8–25.
- Rodriguez, J. D., Perez, A., & Lozano, J. A. 2009. Sensitivity analysis of k-fold cross validation in prediction error estimation Title. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(3), 569–575.
- Sekar Ramadhanti, N., & Ananta Kusuma, W. 2020. *OPTIMASI DATA TIDAK SEIMBANG PADA INTERAKSI DRUG TARGET DENGAN SAMPLING DAN ENSEMBLE SUPPORT VECTOR MACHINE*. 7(6).
- Spoel, S. H. 2018. Orchestrating the proteome with post-translational modifications. *Journal of Experimental Botany*, 69(19), 4499–4503.
- Xiao, J., Xie, L., He, C., & Jiang, X. 2012. Dynamic classifier ensemble model for customer classification with imbalanced class distribution. *Expert Systems with Applications*, 39(3), 3668–3675.
- Xu, Y., Wang, X., Wang, Y., Tian, Y., Shao, X., Wu, L. Y., & Deng, N. 2014. Prediction of posttranslational modification sites from amino acid sequences with kernel methods. *Journal of Theoretical Biology*, 344, 78–87.
- Yang, J., Gong, J., Tang, W., Shen, Y., Liu, C., & Gao, J. 2019. Delineation of urban growth boundaries using a patch-based cellular automata model under multiple spatial and socio-economic scenarios. *Sustainability (Switzerland)*,
- Yulinda, L., Wahyuningsih, T., & Pranowo, H. 2013. Metilasi Asam Galat Menggunakan Agen Metilasi Dimetil Sulfat (DMS) atau Dimetil Karbonat (DMC). *Bimipa*, 23(2), 198–210.