PREDIKSI METILASI PADA *SEQUENCE* PROTEIN *ARGININE* MENGGUNAKAN RANDOM FOREST

(**Skripsi)**

Disusun Oleh :

WIWIT MUDYANINGSIH

JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
2019

ABSTRACT

METHYLATION PREDICTION IN ARGININE PROTEIN SEQUENCE USING RANDOM FOREST

BY

WIWIT MUDYANINGSIH

One of the popular Post-Translational Modification (PTM) is methylation. Methylation can occur in the amino acid arginine. Arginine methylation carries out and regulates several important biological functions, including gene regulation and signal transduction. Experimental identification of arginine methylation sites is a daunting task because it is expensive and requires more time and energy. Therefore reliable predictions play an important task in predicting quickly and identifying possible methylation sites in proteomes. In the study using the Random Forest method which is one of the data mining techniques for classification, and previously performed feature extraction, feature extraction used is CTD, PseAAc, AA index and QSO by using the protr and bioseqclass packages in the r programming language. and obtained 138 variables. This study has 3 types of experiments, training set in 1:1 ratio, Testing data set and Independent data set. The prediction perform reasonably well with the highest accuracy obtained in the Independent Data experiment that is equal to 98.08%, while for Training Data the ratio of 1: 1 gets 93.76% and Testing gets the lowest accuracy that is equal to 80.32%.

Keywords: *Methylation, Random Forest, Prediction, Arginine, Feature Extraction, Machine Learning*

ABSTRAK

PREDIKSI METILASI PADA *SEQUENCE* PROTEIN *ARGININE* MENGGUNAKAN RANDOM FOREST

OLEH

WIWIT MUDYANINGSIH

Salah satu *Post-Translational Modification* (PTM) yang populer adalah metilasi**.** Metilasi dapat terjadi dalam asam amino arginine**.** Metilasi arginin melakukan dan mengatur beberapa fungsi biologis penting, termasuk regulasi gen dan transduksi sinyal. Identifikasi eksperimental situs *arginine* metilasi adalah tugas yang berat dikarenakan mahal serta memerlukan waktu dan tenaga yang lebih. Oleh karena itu prediksi yang handal memainkan tugas penting dalam memprediksi dengan cepat dan mengidentifikasi kemungkinan situs metilasi di proteomes. Pada penelitian menggunakan metode Random forest yang merupakan salah satu teknik data mining untuk melakukan klasifikasi, serta sebelumnya dilakukan *feature extraction*, *feature extraction* yang digunakan yaitu CTD, PseAAc, AA index dan QSO dengan menggunakan *package* protr dan *bioseqclass* pada bahasa pemrograman R Programming., dan didapatkan 138 variabel. Penelitian ini memiliki 3 jenis eksperimen yaitu data Rasio 1:1, *Testing* dan *Independent* dataset. Prediksi ini berkinerja yang cukup baik dengan didapatkan akurasi tertinggi pada percobaan Data *Independent* yaitu sebesar 98,08 %, sedangkan untuk data Training Rasio 1:1 mendapatkan 93,76 % dan *Testing* mendapat akurasi terendah yaitu sebesar 80,32 %.

Kata kunci: Metilasi**,** Random Forest, Prediksi, *Arginine*, Ekstraksi Fitur, *Machine Learning*

PREDIKSI METILASI PADA *SEQUENCE* PROTEIN *ARGININE* MENGGUNAKAN RANDOM FOREST

Disusun Oleh :

WIWIT MUDYANINGSIH

(**Skripsi)**

Sebagai Salah Satu Syarat Untuk Mencapai Gelar

SARJANA KOMPUTER

Pada

Jurusan Ilmu Komputer

Fakultas Matematika Dan Ilmu Pengetahuan Alam

JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
2019

Judul Skripsi	:	**PREDIKSI METILASI PADA *SEQUENCE* PROTEIN *ARGININE* MENGGUNAKAN RANDOM FOREST**
Nama Mahasiswa	:	Wiwit Mudyaningsih
No. Pokok Mahasiswa	:	1517051225
Jurusan	:	Ilmu Komputer
Fakultas	:	Matematika dan Ilmu Pengetahuan Alam

MENYETUJUI

1. Komisi Pembimbing

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

Bambang Hermanto, S.Kom., M.Cs.
NIP. 19790912 200812 1 002

2. Mengetahui
Ketua Jurusan Ilmu Komputer
FMIPA Universitas Lampung

Dr. Ir. Kurnia Muludi, M.S.Sc.
NIP. 19640616 198902 1 001

MENGESAHKAN

1. Tim Penguji

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

   Sekretaris : **Bambang Hermanto, S.Kom., M.Cs.** .....

   Penguji
   Bukan Pembimbing : **Dr. Eng. Admi Syarif** .....

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

Drs. Suratman, M.Sc.
NIP. 19640604 199003 1 002

Tanggal Lulus Ujian Skripsi : **30 September 2019**

PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul "Prediksi Metilasi Pada *Sequence* Protein *Arginine* Dengan Menggunakan Metode Random Forest" merupakan karya saya sendiri dan bukan karya orang lain. Semua tulisan yang tertuang di skripsi ini telah mengikuti kaidah penulisan karya tulis ilmiah Universitas Lampung. Apabila dikemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar yang saya terima.

Bandar Lampung, September 2019

WIWIT MUDYANINGSIH

NPM. 1517051225

RIWAYAT HIDUP

Penulis dilahirkan pada tanggal 27 Juli 1997 di Lampung Selatan sebagai anak tunggal dengan Ayah yang bernama Sari Megantoro dan Ibu bernama Neneng Wida Ningsih. Penulis sekarang bertempat tinggal pada Dusun Taman Jaya RT 012 RW 004 Desa Taman Sari Kecamatan Ketapang Lampung Selatan Lampung. Pendidikan yang telah diselesaikan penulis sekolah dasar di SDN 4 Banjar Agung Kota Serang Banten dan selesai pada tahun 2009. Kemudian penulis melanjutkan ke pendidikan menengah pertama di SMPN 17 Kota Serang Banten selesai pada tahun 2012, kemudian melanjutkan ke pendidikan menengah atas di SMK Miftahul Hidayah Pasir Sakti Lampung Timur yang diselesaikan pada tahun 2015.

Pada tahun 2015 penulis terdaftar sebagai mahasiswa di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung. Selama menjadi mahasiswa beberapa kegiatan yang dilakukan penulis antara lain pada bulan Januari 2018 penulis melaksanakan Kerja Praktik (KP) di Tribun Lampung, dan pada bulan Juli 2018 melaksanakan Kuliah Kerja Nyata (KKn) di Desa Pempen Kecamatan Gunung Pelindung, Lampung Timur.

MOTTO

Perangi Waktu di Dunia mu dengan Hal-Hal yang positif

Awali tujuan hidup dengan mimpi Karena mimpi yang akan melahirkan impian.
Sebab Impian adalah jembatan menuju kesuksesan

PERSEMBAHAN

Puji Syukur atas berkah dan rahmat dari Allah Subhanahu Wata'ala,

Kupersembahkan skripsi ini untuk orang-orang yang yang mencintai dan menyayangiku...

Teruntuk mamah, bapak, ayah yang tidak pernah lelah dan bosan berdoa untuk anaknya, terimakasih yang telah menopang jatuhku dan menuntunku, terimakasih untuk jadi saksi disetiap langkahku...

Teruntuk nenek, mbah putri dan mbah kakong, terimakasih atas dukungan dan semangat yang selalu diberikan untuk cucumu...

Teruntuk teman-temanku yang selalu menunggu dan menemani...

Almamater Tercinta,
UNIVERSITAS LAMPUNG

SANWACANA

Alhamdulillah, segala puji bagi Allah SWT atas segala limpahan ridho, hidayah, dan inayah-Nya sehingga Tugas Akhir dengan judul "Prediksi Metilasi Pada Sequence Protein Arginine Menggunakan Metode Random Forest" ini dapat penulis selesaikan dengan baik dan lancar. Shalawat serta Salam tetap tercurah untuk sang revolusioner sejati, Muhammad SAW yang telah menunjukkan kepada kita dari zaman kegelapan ke zaman yang terang-benderang yaitu Dinul Islam. Skripsi ini disusun untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer Universitas Lampung. Dengan segala keterbatasan yang penulis miliki, masih banyak kekurangan-kekurangan yang harus diperbaiki. Semoga hasil penelitian ini dapat berguna, khususnya bagi dunia Bioinformatika. Dalam penulisan Skripsi ini, penulis banyak mendapat bantuan dari berbagai pihak. Oleh karena itu, ucapan terima kasih penulis sampaikan kepada:

1. Allah SWT atas segala rahmat dan hidayah-Nya hingga Tugas Akhir ini dapat terselesaikan dengan baik.
2. Ayahanda dan Ibunda tercinta yang dengan penuh kesabaran dan pengorbanannya selalu memberikan dorongan, bantuan material maupun non material agar penulis dapat menyelesaikan studi.

1. Bapak Favorisen R. Lumbanraja, Ph.D selaku Dosen Pembimbing I yang telah meluangkan waktu serta dengan penuh kesabaran telah memberikan bimbingan dalam penyusunan Skripsi.
2. Bapak Bambang Hermanto, S.Kom., M.Cs selaku pembimbing II yang telah ikut memberikan bimbingan, saran dan masukan guna penyempurnaan dalam penulisan skripsi ini.
3. Bapak Dr. Eng. Admi Syarif selaku dosen pembahas skripsi, yang telah memberikan bimbingan, saran dan masukkan guna penyempurnaan dalam penulisan skripsi ini.
4. Bapak Drs. Suratman, M.Sc. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
5. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc. selaku Ketua Jurusan Ilmu Komputer Universitas Lampung
6. Ibu Anie Rose Irawati, S.T., M.Cs., selaku pembimbing akademik yang selalu memberi arahan dalam menjalankan perkuliahan.
7. Bapak dan Ibu Dosen Jurusan Ilmu Komputer yang telah memberikan ilmu dan pengalaman hidup selama penulis menjadi mahasiswa.
8. Ibu Ade Nora Maela, Mas Naufal, Mas Irsan, Mas Zai dan Mbak Lia yang telah membantu memudahkan dalam segala urusan administrasi dan lain-lain kepada penulis di Jurusan Ilmu Komputer.
9. Seluruh kelurga dan saudaraku yang telah memberikan dukungan selama proses perkuliahan yang tidak bisa disebutkan satu persatu.
10. Nurul Istiqomah, Devi, Jannati, Nina, dan lain-lain yang selalu menjadi teman dalam mengerjakan skripsi.

11. Rekan-rekan Ilmu Komputer 2015 yang tidak bisa disebutkan satu persatu, terimakasih untuk segala dukungan, bantuan serta kebersamaannya selama ini.
12. Semua pihak yang tidak bisa penulis sebutkan satu persatu, terimakasih atas bantuan dan dukungannya.

Penulis menyadari Skripsi ini masih jauh dari sempurna, karena hal tersebut tidak lepas dari kelemahan dan keterbatasan penulis. Penulis juga mengharapkan kritik dan saran yang bersifat membangun dari semua pihak. Akhirnya penulis berharap agar Skripsi ini berguna sebagai tambahan ilmu pengetahuan serta dapat memberikan manfaat baik kepada penulis maupun kepada pembaca pada umumnya.

Bandar Lampung, September 2019
Penulis,

Wiwit Mudyaningsih

DAFTAR ISI

Halaman

DAFTAR GAMBAR

Gambar **Halaman**

DAFTAR TABEL

Tabel **Halaman**

DAFTAR KODE

Kode **Halaman**

I. PENDAHULUAN

B. Latar Belakang

Post Translational Modifications (PTM) merupakan modifikasi kimia yang memainkan peran kunci dalam proteomik fungsional karena banyak digunakan oleh sel untuk meningkatkan struktur dan biokimia (Vuzman, Hoffman, dan Levy 2012). PTM meningkatkan fungsi Proteome untuk mengelola semua aspek biologi sel normal. Mereka didasarkan pada lampiran kovalen dari kelompok fungsional (Didonna dan Benetti 2016). Modifikasi ini termasuk Fosforilasi, Glikosilasi, *Ubiquitination*, Nitrosilasi, Metilasi, Asetilasi, *Lipidation* Dan Proteolisis dan mempengaruhi hampir semua aspek biologi sel normal dan *patogenesis.* Modifikasi ini mengacu pada apa saja perubahan dalam urutan asam amino dari protein atau modifikasi dari sisi asam amino rantai, asam amino terminal atau karboksil (Walsh, Garneau-Tsodikova, dan Gatto 2005). Umumnya, pasca translasi ini modifikasi mempengaruhi struktur, stabilitas, aktivitas, lokalisasi seluler atau substrat spesifisitas protein. PTM memberikan kompleksitas pada *proteome* untuk beragam fungsi dengan jumlah gen terbatas.

Salah satu dari modifikasi pasca translasi ialah metilasi, metilasi merupakan proses biokimia yang penting, metilasi sendiri adalah proses melalui

sekelompok enzim spesifik, *methyltransferase* memodifikasi protein dengan menambahkan kelompok metil. Metilasi digunakan untuk momoderisasi dan mengendalikan ekspresi gen tertentu berdasarkan kondisi tertentu . Untuk mengidentifikasi secara akurat apakah residu nukleotida termetilasi di bawah konteks urutan protein tertentu. Metilasi terjadi pada atom nitrogen di N-terminal biasanya tidak dapat dibalik dan menciptakan residu asam amino baru. Metalisis pada protein biasanya terdapat pada *Arginine, lisin, kelompok histidin, proline, dan karboksil,* paling umum pada lisin dan Arginine residu, setidaknya di sel eukariotik (Lee et al. 2005). Modifikasi pada eukariotik yang memainkan peran sentral dalam pemeliharaan stabilitas genom, pembungkaman gen, epigenetik yang berbeda, analisis gene silencing, genomic imprinting, dan penyakit (Buck-Koehntop et al. 2012). Metilasi kemudian menentukan apakah transkripsi gen diaktifkan atau ditekan, sehingga mengarah pada hasil biologis yang berbeda (Santos dan Lindner 2017).

Pada penelitian metilasi terdapat beberapa contoh yaitu, oleh Pawan Kumar, et al., (2017) tentang PRmePRed: *A protein Arginine methylation prediction tool*, penelitian ini Identifikasi eksperimental situs Arginin metilasi dengan tujuan memainkan tugas penting dalam screening yang cepat dan identifikasi kemungkinan situs metilasi di proteomes. Metode yang digunakan menggunakan SVM, Dataset digunakan untuk membangun prediktor tidak bias dan telah diverifikasi input eksperimen. Selain itu, PRmePRed menunjukkan hasil kinerja MCC 0.737, Accuracy 0.8683, Sensitivity 0.8709 dan Specificity 0.866. Penelitian yang lain yaitu milik Weiwei Zhang, et al.,

(2013) membahas tentang pola metilasi DNA mendorong untuk mengembangkan penggolong untuk memprediksi metilasi spesifik. Selanjutnya, metode diidentifikasi fitur genomik yang berinteraksi dengan metilasi DNA, menjelaskan mekanisme yang terlibat dalam metilasi DNA modifikasi dan regulasi, dan menghubungkan proses epigenetik yang berbeda.

Mengklasifikasi suatu data yang disusun secara sistematis ke dalam suatu kelompok dapat mengetahui suatu individu berada pada kelompok tertentu, pada penelitian ini metode *classifier* yang digunakan adalah *random forest*. *Random forest* adalah metode klasifikasi yang banyak digunakan yang menggabungkan ide *bagging* pohon klasifikasi dan pengacakan fitur subset, didasarkan pada keputusan pohon dan menggunakan ide-ide agregasi (Breiman 2001). skema ini untuk membangun prediktor *ensemble* dengan seperangkat pohon keputusan yang tumbuh di *subspaces* data yang dipilih secara acak. Meskipun minat dan penggunaan praktis, telah ada sedikit eksplorasi sifat statistik *random forest*, dan sedikit yang diketahui tentang kekuatan matematika yang mendorong algoritma. Pengklasifikasi *random forest* telah terbukti menghasilkan pohon dengan bias rendah dan memiliki korelasi rendah antara masing-masing pohon, menciptakan pengklasifikasi efisien, terutama pada data berdimensi tinggi dan struktur pohon secara eksplisit menangkap interaksi di antara fitur.

Maka dengan ini penulis membuat Prediksi Metilasi Pada *Sequence* Protein *Arganine* Menggunakan *Random Forest* penelitian ini lebih berfokus pada urutan atau *sequence* protein Arginine dengan tujuan memprediksi dan

mengklasifikasikan *sequence* protein Arginine mana yang telah termetilasi dan tidak termetilasi dengan menggunakan metode *random forest*.

C. Rumusan Masalah

Berdasarkan latar belakang yang telah disebutkan, maka dapat dirumuskan masalah yaitu bagaimana membuat prediksi metilasi pada *sequence* protein Arginine yang dapat melihat yang mana yang telah termetilasi ataupun yang tidak termetilasi dengan menggunakan Random Forest?

D. Tujuan Penelitian

Berdasarkan latar belakang diatas maka tujuan dari penelitian skripsi ini adalah :

1. Melakukan *feature extraction* pada *sequence* protein Arginine.
2. Mengklasifikasikan *sequence* protein Arginine yang terjadi metilasi menggunakan metode Random Forest.
3. Membandingkan hasil kinerja dari klasifikasi prediksi metilasi pada *sequence* protein *arginine* dengan penelitian Kumar, dkk.

E. Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut :

1. Menambah pengetahuan tentang pengklasifikasian menggunakan metode Random Forest.
2. Menambah pengetahuan tentang *feature extraction* yang baik untuk *sequence* protein.

3. Mengetahui perbandingan klasifikasi dengan data Rasio 1:1, Testing yang berbeda serta data independent.

F. Batasan Masalah

Untuk membatasi permasalahan agar tidak meluas karena keterbatasan penulis maka batasan masalah dari skripsi ini dibatasi dengan beberapa hal, yaitu :

1. Penggunaan metode hanya pada metode Random Forest dan *feature extraction* menggunakan Package protr dan BioSeqClass.
2. Data yang digunakan dalam data protein yang di dapat dari *study literature*
3. Metilasi pada *sequence* protein pada penelitian ini berfokus pada protein Arginine.

II. LANDASAN TEORI

B. Asam Amino

Protein penting bagi tubuh manusia, komponen utama dalam penyusun protein adalah asam amino yang berfungsi sebagai metabolisme dalam tubuh. Asam amino adalah unit dasar protein yang mengandung gugus amino dan gugus karboksilat, asam amino memainkan peran utama dalam mengatur berbagai proses terkait dengan ekspresi gen. Asam amino dibagi dua kelompok yaitu asam amino esensial dan non-esensial . Asam amino esensial adalah asam amino yang tidak dapat dibuat oleh tubuh dan harus diperoleh dari makanan sumber protein, sedangkan asam amino non esensial merupakan asam amino yang dapat dibuat oleh tubuh manusia. Asam amino terdiri atas unsur-unsur karbon, hidrogen, oksigen, dan nitrogen. Unsur nitrogen adalah unsur utama protein sebanyak 16% dari berat protein. Asam amino digunakan dalam pembentukan protein. Jika asam amino kurang, maka sintesis protein tidak terjadi. Terdapat 20 jenis asam amino yang dapat dilihat pada Tabel 1

Tabel 1. Macam-macam Asam Amino (Akram et al. 2011).

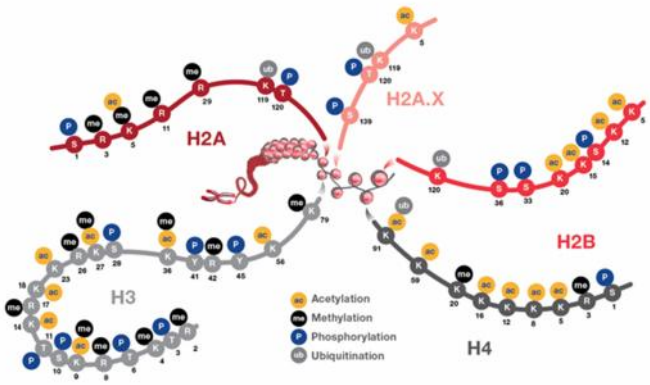
No	Essensial Amino Acids	Non-Essensial Amino Acids
1	Lysine	Cysteine
2	Methionine	Tyrosine
3	Valine	Serine
4	Tryptophan	Alanine
5	Isoleucine	Asparagines
6	Histidine	Aspartic Acid
7	Phenylalanine	Glutamic Acid
8	Threonine	Glycine
9	Leucine	Hydroxylysine
10	Arginine	Proline

C. *Post Translational Modifications* (PTM)

Post Translational Modifications (PTM) dapat dikatakan sebagai modifikasi kovalen yang meningkatkan keragaman struktur dan biofisik protein dan dengan demikian memperkaya informasi yang tersimpan di dalamnya genom (Didonna dan Benetti 2016). Peningkatan kemampuan dari pengkodean genetika protein menciptakan kapasitas fungsional baru, diversifikasi *posttranslasional* dari *proteome* menyinari molekul yang mendasarinya logika untuk akuisisi epigenetik fungsi protein baru. Macam-macam dari PTM yaitu *Glycosylation, Phosphorylation, Ubiquitination,* S-nitrosylation , *Lipidation, Acetylation, Proteolysis dan Methylation*

Alkyl substituents melekat ke daerah spesifik protein oleh enzim PTM. Pengenalan gugus *alkil* yang demikian menghasilkan perubahan *hidrofobik* dari protein yang dimodifikasi. Jenis alkilasi protein yang paling umum adalah protein metilasi. Metilasi adalah PTM termediasi yang terkenal oleh *methyltransferases*. Satu kelompok *metil* karbon adalah ditambahkan ke nitrogen atau oksigen (N- dan O-metilasi, resp.) pada rantai samping asam amino, meningkatkan *hidrofobisitas* protein atau menetralkan muatan negatif ketika terikat ke *karboksilat* asam. Sementara N-metilasi adalah *ireversibel*, O-metilasi berpotensi *reversibel*. Metilasi terjadi begitu sering donor metil utamanya, *S-adenosyl methionine* (SAM), disarankan sebagai substrat enzimatik yang paling banyak digunakan setelahnya ATP (Santos dan Lindner 2017) .

Gambar 1 memberikan ilustrasi PMT yang terkait dengan partikel inti *nukleosom.*



Gambar 1. Representasi Modifikasi Pasca Translasi Yang Terkait Dengan Partikel Histon (Thermo Fisher Scientific 2016).

Representasi menunjukkan modifikasi pasca-translasi yang terkait dengan partikel *histon*. *Nukleosom* diwakili oleh bola merah yang dibungkus oleh DNA (ditampilkan dalam warna abu-abu). Juga digambarkan adalah posisi PTM yang terletak pada protein histon H2A (dan H2A.X), H2B, H3, dan H4. PTMs ini berdampak ekspresi gen dengan mengubah struktur kromatin dan merekrut pengubah histon. Peristiwa PTM memediasi beragam fungsi biologis seperti aktivasi transkripsi dan inaktivasi, kemasan kromosom, dan kerusakan DNA dan proses perbaikan.

Penulis melakukan metilasi pada *sequence* protein yang di dalamnya tersusun dari asam amino dan melakukan modifikasi dengan menambahkan gugus metil atau alkohol pada *sequence* protein, pada penelitian ini berfokus pada metilasi protein *arginine* atau protein *arginine methylation*.

Arginine merupakan salah satu dari asam amino yang ada pada protein, Metilasi arginin adalah modifikasi pasca-translasi lazim yang ditemukan pada kedua protein nuklir dan sitoplasma. Metilasi residu arginin dikatalisis oleh protein arginin *N-methyltransferase* (PRMT) keluarga enzim. Protein yang di metabolisme arginin terlibat dalam sejumlah proses seluler yang berbeda, termasuk regulasi transkripsi, metabolisme RNA dan perbaikan kerusakan DNA (Bedford dan Richard 2005).

D. Redundansi Data

Data protein yang baik akan mempengaruhi hasil dari prediksi suatu klasifikasi, membangun sebuah dataset yang baik harus menghapus sequence protein yang memiliki kesamaan atau kemiripan dengan sequence protein

yang lainnya. Redundansi dalam dataset terjadi ketika beberapa data yang sama yang muncul pada waktu yang sama pula. Dalam bioinformatika, redundansi dalam koleksi urutan protein terjadi ketika satu atau lebih sekuens yang sama / homolog muncul dalam set data yang sama. Jika sequence yang sama di klasifikasi dalam analisis tertentu akan menghasilkan bias yang tidak diinginkan. Software yang dapat digunakan untuk melakukan redundasi data adalah CD-HIT, Pisces, BlastClust, dan SkipRedundant. CD-HIT dan Pisces adalah standalone applications, sedangkan lainnya adalah bagian dari paket mandiri seperti BLAST (BlastClust) dan EMBOSS (SkipRedundant). Dalam redundansi harus menentukan persentase urutan dari suatu identitas atau Definition of percentage of sequence identity (PID) yaitu sebagai jumlah posisi sejajar di mana karakter yang cocok (asam amino) adalah identik dibagi dengan jumlah posisi sejajar (termasuk kesenjangan, jika ada) (Sikic dan Carugo 2010).

Salah satu nya yaitu *SkipRedundant* (EMBOSS) Dengan metode ini, semua urut keberpihakan berpasangan dihitung dengan menggunakan implementasi EMBOSS dari *Needleman-Wunsch global alignment algorithm*. Program ini dapat menggunakan dua prosedur untuk menghilangkan urutan redundansi, yaitu Jika sepasang protein mencapai persentase urutan identitas lebih besar dari ambang (ditentukan oleh pengguna) urutan terpendek dibuang dan Jika sepasang protein memiliki persentase urutan identitas yang terletak di luar rentang (ditentukan oleh pengguna) urutan terpendek dibuang. Setelah urutan telah dihapus daftar tersebut hanya berisi entri non-redundant.

E. *Feature Extraction*

Pemodelan atau prediksi tentu didalamnya melakukan tahapan dimana menghapus variabel yang tidak relevan, untuk memilih semua yang penting atau untuk menentukan subset yang cukup untuk memprediksi suatu hal (Genuer, Poggi, dan Tuleau-Malot 2015).

Fitur ini didefinisikan sebagai fungsi dari satu atau lebih pengukuran, masing-masing yang menentukan beberapa properti yang dapat dikuantifikasikan dari suatu objek, dan dihitung sedemikian rupa sehingga mengkuantifikasi beberapa karakteristik penting dari objek (Choras 2007). *Feature Extraction* adalah langkah penting untuk pemrosesan multimedia. Bagaimana caranya ekstrak fitur ideal yang dapat mencerminkan konten intrinsik dari gambar (Venkatraman dan Kulkarni 2012), proses perubahan suatu data input menjadi kelompok fitur untuk mengambil representasi minimal dari data *input* . Dalam penelitian ini *feature extraction* yang digunakan protein deskriptor menggunakan *package* protr dan *package* BioSeqClass .

Protein Deskriptor adalah tools untuk melakukan proses *feature extraction* pada protein *sequence,* protein deskriptor bisa juga sebagai substruktur lokal dari molekul protein, yang memungkinkan kita untuk membagi masalah asli menjadi satu set subproblem dan, akibatnya, untuk mengusulkan solusi algoritmik yang lebih efisien. Dalam literatur, seseorang dapat menemukan banyak aplikasi konsep deskriptor yang membuktikan kegunaannya untuk wawasan ke struktur protein 3D, tetapi pendekatan yang diusulkan disajikan lebih dari perspektif biologis daripada dari sudut pandang komputasi atau algoritmik. Algoritma yang efisien untuk identifikasi dan perbandingan

struktural deskriptor dapat menjadi komponen penting dari metode untuk penilaian kualitas struktural serta prediksi struktur tersier.

1. *Package* Protr

Package protr bertujuan untuk ekstraksi fitur *sequence* protein, yang dapat dengan mudah diaplikasikan dalam *Chemoinformatics, Bioinforamtics* dan *Chemogenomics research*. Paket ini dikembangkan oleh *Computational Biology dan Drug Design Group, Central South University* yaitu Nan Xiao, dkk pada tahun 2015. Paket protr menawarkan toolkit yang unik dan komprehensif untuk menghasilkan berbagai skema representasi numerik dari sekuens protein. Deskriptor termasuk digunakan secara luas dalam penelitian bioinformatika dan chemogenomics. Deskriptor yang umum digunakan yang tercantum dalam protr adalah *Amino Acid Composition (Amino Acid Composition/ Dipeptide Composition/ Tripeptide Composition),* CTD*(Composition/ Transition/ Distribution), quasi-sequence order, profile-based descriptors derived by Position-Specific Scoring Matrix* (PSSM) dan lain-lain. *Feature extraction* yang digunakan pada package protr adalah QSO (*Quasi Sequence Order*),

Feature extraction QSO ini menggunakan dimensi (dim: 20 + 20 + (2 *nlag)), dapat dilihat pada Kode Program 1

```
extractQSO(x, nlag = 17, w=0.1)))
```

Kode Program.1 Contoh *Code Feature Extraction* QSO pada *Package* protr

X : sebuah *character* vector , input dari data *sequence* protein

Nlag : Jeda maksimum, default adalah 30.

W : faktor bobot, standarnya adalah 0,1.

Selain QSO masih banyak lagi macam-macam *feature extraction* yang ada pada *package* protr, dapat dilihat pada Tabel 2

Tabel 2. Penjelasan Macam-Macam Protr (Xiao et al. 2015)

Descriptor Group	**Descriptor Name**	**Descriptor Dimension**	**Function Name**
Amino Acid Composition	Amino Acid Composition	20	extractAAC()
	Dipeptide Composition	400	extractDC()
	Tripeptide Composition	8000	extractTC()
Autocorrelation	Normalized Moreau-Broto Autocorrelation	240[1]	extractMoreauBroto()
	Moran Autocorrelation	240[1]	extractMoran()
	Geary Autocorrelation	240[1]	extractGeary()
CTD	Composition	21	extractCTDC(), extractCTDCClass()
	Transition	21	extractCTDT(), extractCTDTClass()
	Distribution	105	extractCTDD(), extractCTDDClass()
Conjoint Triad	Conjoint Triad	343	extractCTriad(), extractCTriadClass()

Tabel 2. Penjelasan Macam-Macam Protr (Xiao et al. 2015) (Lanjutan)

Descriptor Group	**Descriptor Name**	**Descriptor Dimension**	**Function Name**
Quasi-Sequence-Order	Sequence-Order-Coupling Number	60^2	extractSOCN()
	Quasi-Sequence-Order Descriptors	100^2	extractQSO()
Pseudo-Amino Acid Composition	Pseudo-Amino Acid Composition	50^3	extractPAAC()
	Amphiphilic Pseudo-Amino Acid Composition	804	extractAPAAC()

2. *Package* BioSeqClass

BioSeqClass merupakan salah satu *package* mengekstraksi fitur dari *Biological Sequences* yang ada pada bahasa pemrograman R. BioSeqClass untuk melakukan alur kerja umum untuk *feature extraction* ataupun klasifikasi berdasarkan urutan biologis yang berisi skema pengkodean yang beragam untuk RNA, DNA dan protein, mendukung pemilihan fitur, dan mengintegrasikan beberapa metode klasifikasi. Jadi selain guna *feature extraction* pada *package* BioSeqClass juga bisa untuk melakukan beberapa model klasifikasi. Namun pada penelitian kali ini penulis menggunakan package BioSeqClass guna *feature extraction* diantaranya :

a. CTD (*Composition, Transition and Distribution)*

fitur CTD telah berhasil digunakan dalam banyak fungsional dan struktural terkait Studi tentang protein. Dalam CTD, C (*composition*) adalah singkatan dari komposisi asam amino, pada composition jumlah asam amino residu dengan properti tertentu dibagi dengan total jumlah asam amino dalam urutan protein, Composition sebagai persentase global untuk setiap kelas yang dikodekan dalam urutan protein.

$Cr = \frac{nr}{n}$ …… *(1)*

di mana nr adalah jumlah asam amino tipe r dalam urutan yang disandikan dan N adalah panjang urutan.

T (*Transition*) mewakili *persentase* dengan frekuensi asam amino dengan sifat khusus diikuti oleh asam amino dengan properti lain, deskriptor transisi dapat dihitung:

$Trs = \frac{nrs+nsr}{N-1}$ $rs = 12,13,23$ …… *(2)*

di mana nrs, nsr adalah jumlah dipeptide yang dikodekan sebagai rs dan sr dalam urutan; N adalah panjang urutan.

Terakhir D (*Distribution*) didefinisikan sebagai panjang rantai yang pertama, 25%, 50%, 75% dan 100% asam amino dari karakteristik tertentu berada. Contoh program dapat dilihat pada Kode Program 2 .

```
featureCTD(seq,class=elements("aminoacid"))
```

Kode Program. 2 Contoh *Code Feature Extraction* CTD pada *Package* BioSeqClass

Seq : Vektor string untuk urutan protein, DNA, atau RNA.

Class : Daftar untuk kelas sifat biologis, dapat diproduksi oleh elemen dan aaClass

b. AA Index

featureAAindex mengembalikan matriks yang mengukur sifat fisikokimia dan biokimia asam amino oleh AAindex. Jika parameter aaindex.name = "all", semua properti di AAindex akan dipertimbangkan, dan setiap baris mewakili fitur dari satu urutan pengkodean dengan vektor numerik 531 * N dimensi. Jika parameter aaindex.name adalah nama properti di AAindex, setiap baris merepresentasikan fitur dari satu urutan pengkodean dengan vektor numerik dimensi N. contoh program dapat dilihat pada Kode Program 3.

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

Kode Program.3 Contoh *Code Feature Extraction* AA Index pada *Package* BioSeqClass

seq : vektor string untuk urutan protein, DNA, atau RNA.

aaindex.name : string untuk nama sifat fisikokimia dan biokimia di AAindex.

c. Pseudo AA Composite

featurePseudoAAComp mengembalikan matriks yang mewakili komposisi asam amino semu. Setiap baris mewakili fitur dari satu urutan pengkodean dengan vektor numerik 20 + d dimensi. 20 fitur

pertama menunjukkan komposisi 20 asam amino. Fitur d terakhir menunjukkan sambungan antara asam amino X (i) dan X (i + d). Nilai kopling dipengaruhi oleh hidrofobisitas, hidrofilisitas dan massa asam amino. Contoh program dapat dilihat pada Kode Program 4.

```
featurePseudoAAComp(seq,d,w=0.05)
```

Kode Program.4 Contoh *Code Feature Extraction* PseAAC pada *Package* BioSeqClass

Seq : vektor string untuk urutan protein, DNA, atau RNA.

d : integer yang digunakan sebagai parameter featurePseudoAAComp (d> = 1) . Kopling antara asam amino X (i) dan X (i + d) dianggap sebagai fitur.

w : nilai numerik untuk faktor bobot efek urutan urutan di featurePseudoAAComp.

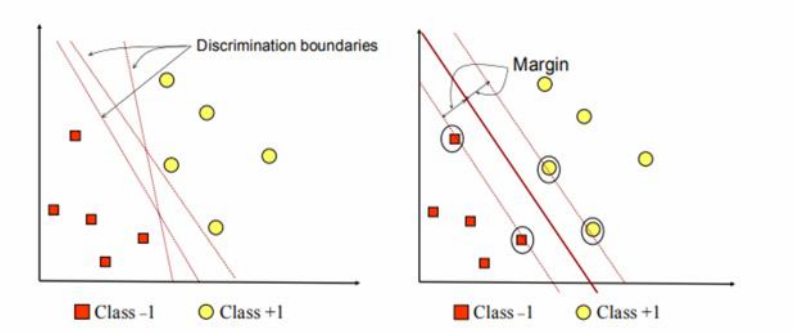
F. *Classification*

Klasifikasi merupakan cara menemukan suatu model untuk digunakan dalam label yang belum diketahui kelasnya guna memprediksi suatu kelas, klasifikasi dapat mendeskripsikan dan membedakan kelas atau konsep suatu data (Han, Kamber, dan Pei 2006).

1. *Support Vector Machine* (SVM)

Support Vector Machine (SVM) diperkenalkan pada tahun 1992 oleh Vapnik sebagai susuanan keserasian konsep-konsep baik dalam bidang

pattern recognition. SVM dalam bidang *pattern recognition* masih dalam katagori baru, walaupun demikian, evaluasi kemampuannya dalam berbagai aplikasinya menempatkannya sebagai *state of the art* dalam *pattern recognition* serta termasuk dalam metode yang berkembang dengan pesat. SVM adalah metode *learning machine* yang bekerja atas prinsip *Structural Risk Minimization* (SRM) dengan tujuan menemukan *hyperplane* terbaik yang memisahkan dua buah class pada *input space.* Tulisan ini membahas teori dasar SVM dan aplikasinya dalam bioinformatika, khususnya pada analisa ekspresi gen yang diperoleh dari analisa *microarray* (Nugroho, Witarto, dan Handoko 2003).



Gambar 2. Contoh SVM (Nugroho, Witarto, dan Handoko 2003)

SVM secara didasarkan pada teori belajar statistik. Untuk masalah dipisahkan secara linear SVM memnggunakan *maximum margin hyper-plane* untuk memisahkan contoh yang milik dua kelas yang berbeda dan untuk memisahkan masalah non-linear , SVM pertama mengubah data ke dalam ruang fitur dimensi yang lebih tinggi dan kemudian mempekerjakan maximum margin linear hyper plane. Ada empat kernel dasar yang dapat digunakan dalam SVM, yaitu:

Linear : $K(X_i, X_j) = X_i^T X_j$ .....*(3)*

Polynomial : $K(X_i, X_j) = (\gamma X_i^T X_j + r)^d, \gamma > 0$ .....*(4)*

(RBF) : $K(X_i, X_j) = exp(-\gamma \|X_i - X_j\|^2), \gamma > 0$.....*(5)*

Sigmoid : $K(X_i, X_j) = tanh(\gamma X_i^T X_j + r)$. .....*(6)*

γ, r, dan d adalah parameter kernel. RBF jauh pilihan yang paling populer dari jenis kernel yang digunakan di Support Vector Machines. Hal ini terutama karena RBF kernel non-linear memetakan sampel ke dalam ruang dimensi yang lebih tinggi sehingga, tidak seperti kernel linear, dapat menangani kasus ketika hubungan antara label kelas dan atribut adalah nonlinear (Kumar et al. 2017).

2. K-Nearest Neighbor

K-Nearest Neighbor (K-NN) termasuk kelompok *instance-based learning*. Algoritma ini juga merupakan salah satu teknik *lazy learning* (Wu et al. 2008). Metode K-NN dapat menentukan suatu benda yang nantinya dikelompokan dan dipisahkan sesuai dengan kelasnya, berjalan dengan cara mengukur kedekatan antara benda baru dengan benda lama atau objek pada data baru atau data *testing* (Leidiyana 2013). K-NN relatif tidak sensitif terhadap *error* dalam dataset, dan K-NN dapat digunakan untuk mengelola dataset yang memiliki ukuran besar.

KNN merupakan algoritma supervised learning dimana hasil dari query instance yang baru diklasifikan berdasarkan mayoritas dari kategori pada algoritma KNN. Dimana kelas yang paling banyak muncul yang nantinya

akan menjadi kelas hasil dari klasifikasi. Kedekatan didefinisikan dalam jarak metrik, seperti jarak Euclidean (Johar., Yanosma., dan Anggriani. 1999).

3. *Artificial Neural Network*

Artificial Neural Network (ANN) atau jaringan syaraf tiruan merupakan cara sebuah informasi secara tersebar dan pararel yang memiliki kecenderungan dengan system kerja jaringan otak dalam mengelola informasi. ANN memiliki elemen pemrosesan mirip *neuron* yang sangat sederhana (disebut *node* atau *neuron* buatan) terhubung satu sama lain dengan pembobotan. Bobot pada setiap koneksi dapat disesuaikan secara dinamis hingga output yang diinginkan dihasilkan untuk input yang diberikan. Model *neuron* buatan terdiri dari *linear* kombinasi yang diikuti oleh fungsi aktivasi. Berbagai jenis fungsi aktivasi dapat dimanfaatkan untuk jaringan, namun yang umum, yang cukup untuk sebagian besar aplikasi, adalah fungsi *tangen sigmoidal* dan *hiperbolik*. Dalam sebagian besar aplikasi, fungsi *transfer tangen hiperbolik* merupakan representasi yang lebih baik dibandingkan dengan fungsi *transfer sigmoid* (Saracoglu 2008).

4. *Random Forest*

Random Forest merupakan salah satu dari pengembangan metode *Classification and Regression Tree* (CART) dengan menerapkan metode *bootstrap aggregating (bagging)* dan *random feature selection* (Klusowski 2018). CART (*Classification and Regression Tree*)

merupakan metode eksplorasi data yang didasarkan pada teknik pohon keputusan. Pohon klasifikasi dihasilkan saat peubah *respon* berupa data kategorik, sedangkan pohon regresi dihasilkan saat peubah *respons* berupa data numerik. Pohon terbentuk dari proses pemilahan *rekursif biner* pada suatu gugus data sehingga nilai peubah *respons* pada setiap gugus data hasil pemilahan akan lebih (Dewi et al. 2011)

Algoritme *random forest* merupakan algoritma yang sesuai untuk digunakan pada klasifikasi data yang besar dan pada algoritma *random forest* tidak memiliki pemangkasan variabel seperti pada algoritma decision tree. Metode ini menghubungkan banyak pohon untuk membuat klasifikasi dan prediction class. Pada *random forest* pembuatan pohon dilakukan dengan cara melakukan pelatihan sampel data. *Sampling with replacement* merupakan metode yang digunakan untuk mengambil data sample. Klasifikasi berjalan jika semua pohon terbentuk. Penentuan klasifikasi pada *random forest* ini diambil berdasarkan pilihan dari pohon masing-masing dan pilihan terbanyak yang menjadi pemenang.

Algoritma untuk membangun *random forest* memiliki langkah yang dinamakan dengan *bootstrap* (*bag*) yaitu, pada gugus data yang terdiri dari n amatan dan p peubah penjelas yaitu, lakukan pengambilan contoh acak yang berukuran n dengan pemulihan pada gugus data. selanjutnya dengan menggunakan contoh *bootstrap*, pohon dibangun sampai mencapai ukuran terbesar yaitu tanpa pemangkasan. Pembangunan pohon dilakukan dengan menerapkan *random feature selection* yaitu m peubah penjelas dipilih secara acak dimana $m << p$, selanjutnya pemilah

terbaik dipilih berdasarkan m peubah penjelas. Ulangi langkah tersebut sebanyak k kali untuk membuat sebuah *forest* yang terdiri dari k pohon. Tahapan pembuatan model klasifikasi menggunakan algoritma *random forest* dilakukan setelah membuat pemodelan data latih menggunakan package *random forest* di R. Metode random forest harus menentukan m jumlah variabel prediktor yang diambil secara acak dan k pohon yang akan dibentuk agar mendapatkan hasil yang optimal. Ukuran contoh peubah penjelas (m) saat menggunakan metode random forest sangat mempengaruhi korelasi dan kekuatan masing-masing pohon. Untuk menentukan m yaitu jumlah variabel prediktor yang diambil secara acak dengan nilai p adalah banyak variabel independent (bebas), terdapat tiga cara untuk mendapatkan nilai m untuk mengamati error OOB yaitu:

$m = ½|\sqrt{p}|$ ……..*(7)*

$m = |\sqrt{p}|$ ……..... *(8)*

$m = 2x|\sqrt{p}|$ ……*(9)*

dimana *p* adalah total variabel .

Menurut Breiman (2001), penggunaan *m* yang tepat akan menghasilkan random forest dengan korelasi antar pohon cukup kecil namun kekuatan setiap pohon cukup besar yang ditunjukkan dengan perolehan error OOB bernilai kecil. Terdapat respons suatu amatan diprediksi dengan menggabungkan (*aggregating*) hasil prediksi *k* pohon. Pada masalah klasifikasi dilakukan berdasarkan majority vote atau kategori atau kelas

yang paling sering muncul sebagai hasil prediksi dari *k* pohon klasifikasi. Terdapat data *out-of-bag* (OOB), yaitu sepertiga amatan gugus data asli yang tidak termuat dalam contoh bootstrap pada setiap iterasinya (Breiman 2001). Error OOB bergantung pada korelasi antar pohon dan kekuatan (strength) masing-masing pohon dalam random forest dimana peningkatan korelasi dapat meningkatkan error OOB sedangkan peningkatan pohon dapat menurunkan error OOB (Breiman 2001). Error OOB dihitung dari proporsi misklasifikasi hasil prediksi random forest dari seluruh amatan gugus data asli.

Random forest menggunakan gini index untuk menentukan kelas akhir disetiap pohon. Pada final class dari setiap pohon dikumpulkan dan dipilih oleh nilai-nilai weight untuk membangun classifier akhir. Random forest menggunakan gini index diambil dari system CART untuk membangun decision trees. Gini index pada node impurity adalah ukuran yang paling umum dipilih untuk masalah klasifikasi:

$$Gini(T) = 1 - \sum_{j=1}^{n} (pj)^2 \quad \text{.....(10)}$$

Jika T dataset dibagi menjadi 2 himpunan bagian yaitu T1 dan T2 dengan ukuran N1 dan N2 masing-masing, index pada split data berisi contoh-contoh dari kelas n, index gini (T) didefinisikan sebagai berikut :

$$Gini(T) = \frac{N_1}{N} gini(T_1) + \frac{N_2}{N} gini(T_2) \quad \text{.....(11)}$$

Contoh perhitungan random forest sebagai berikut :

Contoh berikut ini menunjukan pembangunan single tree menggunakan dataset yang disingkat. Hanya dua atribut asli yang diambil untuk konstruksi pohon ini. Data dengan 2 atribut yaitu Home_type dan Salary. Dapat dilihat pada Tabel 3.

Tabel 3. Contoh Soal Random Forest

Record	Atribut		Class
	Home_type	Salary	
1	31	3	1
2	30	1	0
3	6	3	0
4	15	4	1
5	10	4	0
6	20	1	1
7	24	3	0
8	33	4	1
9	27	1	1

Asumsikan atribut pertama yang akan di split adalah atribut Home_type Split yang mungkin untuk Home_type atribut dalam rentang node kiri dari $6<x<33$, di mana x adalah nilai split. Semua nilai nilai lain pada setiap split dari node anak kanan. Split yang memungkinkan untuk attribute Home_type dalam dataset adalah:

- Home_type < 6
- Home_type < 10
- Home_type < 15
- Home_type < 20
- Home_type < 24
- Home_type < 27
- Home_type < 30
- Home_type < 31
- Home_type < 33

Mengambil split pertama, index gini dihitung sebagai berikut :

Partisi setelah split biner pada Home_type < 6 dengan Random Forest dapat dilihat pada Tabel 4

Tabel 4. Perhitungan Soal Random Forest

Atribut	Number Of Record		
	Zero (0)	One (1)	N=9
Home_type <= 6	1	0	N1= 1
Home_type >6	3	5	N2=8

Kemudian Gini D1 Gini D2 dan gini split dapat dihitung sebagai berikut:

Gini(Home_type <= 6)= $1-\left(\left(\frac{1}{1}\right)^2+\left(\frac{0}{1}\right)^2\right)=0$

Gini(Home_type >6)= $1-\left(\left(\frac{3}{8}\right)^2+\left(\frac{5}{8}\right)^2\right)=0{,}46875$

Gini Split= $\left(\frac{1}{9}\right)$ x 0 + $\left(\frac{8}{9}\right)$ x 0,46875=0,416667

Lanjutkan lakukan dengan cara yang sama untuk mencari gini split untuk <10, <15,<20, <24, <27, <30, <31, <33. Sehingga akan mendapatkan seluruh gini split :

Tabulates nilai gini index untuk atribut Home_type setiap kemungkinan split dapat dilihat pada Tabel 5.

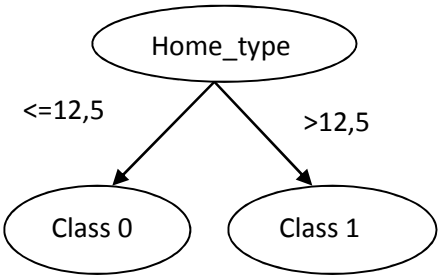
Tabel 5. Nilai Gini Split pada Setiap Kemungkinan

Gini Split	Value
Gini Split (Home_type <=6)	0,416667
Gini Split (Home_type <=10)	0,31746
Gini Split (Home_type <=15)	0,444444
Gini Split (Home_type <=20)	0,488889
Gini Split (Home_type <=24)	0,433333
Gini Split (Home_type <=27)	0,481481
Gini Split (Home_type <=30)	0,380952
Gini Split (Home_type <=31)	0,444444
Gini Split (Home_type <=33)	0,493827

Dari tabel diatas didapatkan bahwa Gini index terendah pada Home_type <= 10 yaitu 0,31746.. Dalam random forest, split dimana gini index terendah dipilih pada nilai split, namun karena nilai-nilai atribut Home_type yang continue, titik tengah setiap pasangan nilai berturut dipilih sebagai titik best split

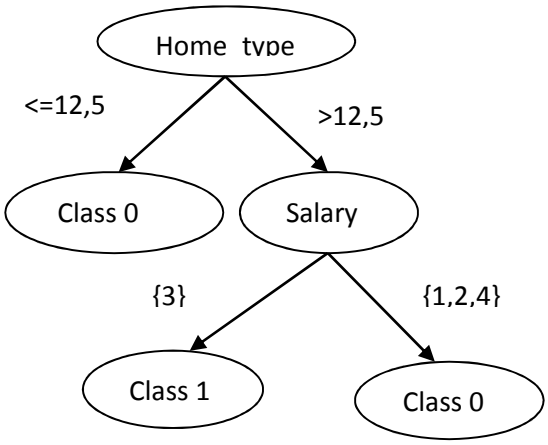
Best split pada contoh ini adalah Home_type = (10+15)/2 = 12,5

Jadi best split bukan pada Home_type <=10. Decision Tree yang pertama ditampilkan pada Gambar 3



Gambar 3. Tree Soal Random Forest Variabel ke- satu

Prosedur diulang untuk atribut sisa dalam dataset. Pada contoh ini nilai gini indeks dari atribut kedua yaitu “salary” atribut telah dihitung dengan prosedur seperti Home_type yaitu mencari Gini Index dan Gini Split. Nilai terendah indeks gini dipilih sebagai best split untuk atribut, sehingga decision tree akhir akan ditunjukan seperti tree pada Gambar 4



Gambar 4. Tree Soal Random Forest Seluruh Variabel

Hasil *rule* dari tree tersebut untuk aturan keputusan untuk *decision tree* yang digambarkan diatas adalah :

IF HOME_TYPE <=12,5 MAKA NILAI CLASS 0.

IF HOME_TYPE >12,5 AND SALARY IS 3 MAKA NILAI CLASS 1.

IF HOME_TYPE >12,5 AND SALARY IS 1/2/4 MAKA NILAI CLASS 0.

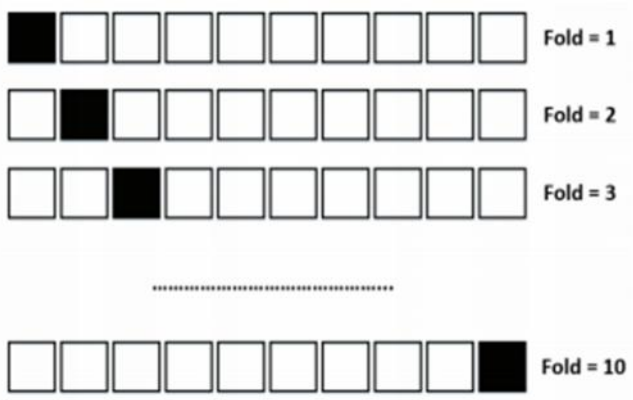
G. *Cross Validation*

Cross Validation ini digunakan untuk membandingkan suatu algoritma pembelajaran yang memiliki 2 segmen dalam pembagian data dimana segmen yang pertama digunakan untuk belajar atau melatih sebuah model dan segmen yang ke dua digunakan untuk memvalidasi model, *cross validation* ini digunakan untuk metode statistic untuk mengevaluasi. *Cross over* digunakan untuk set pelatihan dan validasi sehingga setiap data titik memiliki kesempatan untuk divalidasi (Refaeilzadeh, Tang, dan Liu 2017).

1. *K-Fold Cross-Validation*

Dalam *k-fold cross-validation*, data dipartisi pertama ke dalam segmen yang sama (atau hampir sama) atau *fold*. Selanjutnya k iterasi pelatihan dan validasi dilakukan sedemikian rupa sehingga dalam setiap iterasi a lipatan yang berbeda dari data diadakan untuk validasi sedangkan sisa k 1 lipatan digunakan untuk belajar. Data biasanya dikelompokkan sebelum dibagi menjadi k lipatan. Stratifikasi adalah proses menata ulang data untuk memastikan setiap lipatan merupakan perwakilan keseluruhan yang baik. Misalnya dalam masalah klasifikasi biner di mana setiap kelas terdiri dari 50% dari data, itu terbaik untuk mengatur data sedemikian

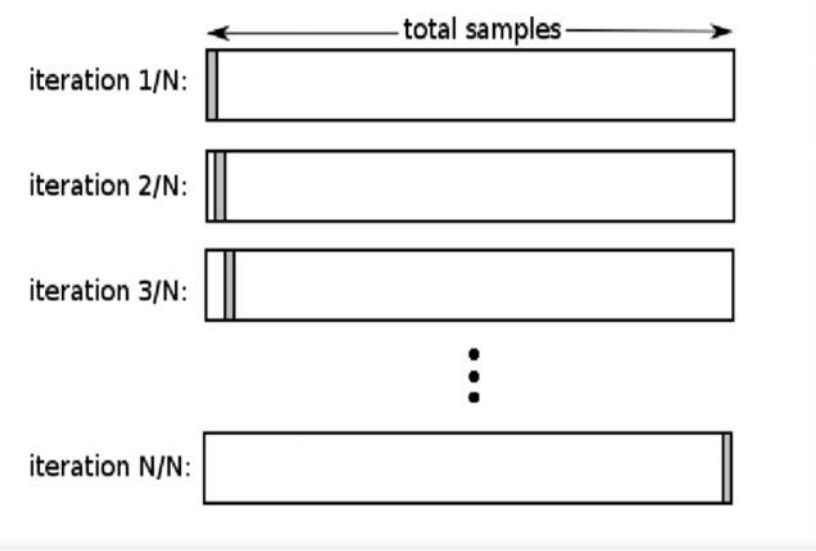
rupa sehingga di setiap lipatan, masing-masing kelas terdiri dari sekitar setengah instances. Biasanya 10-*fold cross-validation* (k = 10) adalah yang paling umum. *Cross Validation* digunakan untuk mengevaluasi atau membandingkan belajar algoritma sebagai berikut: di setiap iterasi, satu atau Algoritma pembelajaran lebih banyak menggunakan k 1 kali lipat data untuk belajar satu atau lebih model, dan kemudian yang dipelajari model diminta untuk membuat prediksi tentang data dalam lipatan validasi. Kinerja setiap pembelajaran algoritma pada setiap lipatan dapat dilacak menggunakan beberapa yang telah ditentukan metrik kinerja seperti akurasi. Atas selesai, k sampel dari metrik kinerja akan tersedia untuk setiap algoritma. Metodologi yang berbeda seperti rata-rata dapat digunakan untuk mendapatkan agregat mengukur dari sampel ini, atau sampel ini bisa digunakan dalam uji hipotesis statistik untuk menunjukkan itu satu algoritma lebih unggul dari yang lain (Refaeilzadeh, Tang, dan Liu 2017). Gambar 5 di bawah ini merupakan ilustrasi dari *k–fold cross validation*



Gambar 5. Ilustrasi *k-fold* untuk k=10 (Lukito dan Chrismanto 2015)

2. *Leave-One-Out Cross-Validation* (LOOCV)

Leave-one-out cross-validation (LOOCV) adalah spesial kasus *k-fold cross-validation* di mana k sama dengan jumlah instance dalam data. Dengan kata lain di setiap iterasi hampir semua data kecuali untuk observasi tunggal digunakan untuk pelatihan dan modelnya diuji pada pengamatan tunggal itu. Estimasi akurasi diperoleh menggunakan LOOCV diketahui hampir tidak ada bias tetapi memiliki varians yang tinggi, menyebabkan tidak dapat diandalkan perkiraannya. Model ini masih banyak digunakan ketika tersedia data sangat jarang, terutama di bioinformatika di mana hanya lusinan sampel data yang tersedia (Refaeilzadeh, Tang dan Liu, 2008). LOOCV adalah bentuk khusus dari cross-validation, yaitu jumlah fold sama dengan jumlah data training. LOOCV ini sering digunakan untuk memperkirakan kemampuan generalisasi penggolongan statistik (yaitu kinerja pada data yang sebelumnya tidak terlihat) (Cawley dan Talbot 2003). Pendekatan ini meninggalkan 1 titik data dari data pelatihan, yaitu jika ada n titik data dalam sampel asli maka, sampel n-1 digunakan untuk melatih model dan poin p digunakan sebagai set validasi. Ini diulang untuk semua kombinasi di mana sampel asli dapat dipisahkan dengan cara ini, dan kemudian kesalahan dirata-ratakan untuk semua percobaan, untuk memberikan keefektifan secara keseluruhan. Dapat di lihat pada Gambar 6.



Gambar 6. Ilustrasi *Leave-One-Out Cross-Validation* (Cawley dan Talbot 2003).

3. *Hold-Out Validation*

Pada *Hold-out cross-validation* data dibagi menjadi dua dataset yang berbeda yang diberi label sebagai pelatihan dan dataset pengujian Keuntungan dari metode ini adalah itu proporsi dari ketiga himpunan data ini tidak dibatasi secara ketat. Dalam hal ini, ada kemungkinan distribusi yang tidak merata dari berbagai kelas data ditemukan dalam pelatihan dan dataset uji. Untuk memperbaikinya, dataset pelatihan dan uji dibuat dengan distribusi kelas data yang berbeda. Proses ini disebut stratifikasi (Reitermanov 2010).

Untuk menghindari *over-fitting*, set tes *independen* lebih disukai. Pendekatan alami adalah membagi data yang tersedia menjadi dua bagian yang tidak tumpang tindih: satu untuk pelatihan dan yang lain untuk pengujian. Data tes diadakan dan tidak melihat selama pelatihan. Masa berlaku validasi penonaktifan tumpang tindih antara data pelatihan dan

data uji, menghasilkan perkiraan yang lebih akurat untuk generalisasi kinerja algoritma. Kelemahannya adalah itu prosedur ini tidak menggunakan semua data yang tersedia dan hasilnya sangat tergantung pada pilihan untuk pelatihan / tes terpisah. Instansi yang dipilih untuk dimasukkan dalam set tes mungkin terlalu mudah atau terlalu sulit untuk digolongkan dan ini dapat mengubah hasilnya. Selanjutnya, data dalam set tes mungkin berharga untuk pelatihan dan jika itu diadakan kinerja prediksi, kembali memimpin untuk hasil yang miring. Masalah-masalah ini bisa sebagian ditangani dengan mengulang beberapa validasi hold-out kali dan rata-rata hasil, tetapi kecuali pengulangan ini dilakukan secara sistematis, beberapa data dapat dimasukkan dalam set tes beberapa kali sementara yang lain tidak termasuk sama sekali, atau sebaliknya beberapa data mungkin selalu jatuh dalam set tes dan tidak pernah mendapat kesempatan berkontribusi pada fase pembelajaran. Untuk menghadapi ini menantang dan memanfaatkan data yang tersedia secara maksimal, *k-fold cross-validation* digunakan (Refaeilzadeh, Tang, dan Liu 2017).

H. Evaluasi Matrix

Evaluasi Matrix atau *Confusion Matrix* merupakan metode untuk penilaian dengan menggunakan tabel matrix (Leidiyana 2013). Dengan penjelasan *confusion matrix* dapat disajikan seperti pada Tabel 6.

Tabel 6. Representasi Hasil Proses Klasifikasi pada *Confusion Matrix*

Kelas	**Terklasifikasi Positif**	**Terklasifikasi Negatif**
Positif	TP (True Positive)	FN (False Negative)
Negatif	FP (False Positive)	TN (True Negative)

- TP, yaitu jumlah data positif yang terklasifikasi benar oleh sistem.
- TN, yaitu jumlah data negatif yang terklasifikasi benar oleh sistem.
- FN, yaitu jumlah data negatif namun terklasifikasi salah oleh sistem.
- FP, yaitu jumlah data positif namun terklasifikasi salah oleh sistem

Berikut ini merupakan macam-macam dari evaluasi matrik:

1. *Recall*

Recall adalah proporsi jumlah dokumen yang dapat berhasil di temukan kembali oleh sebuah pencarian dalam sistem temu kembali (Dwiyantoro 2017). Sedangkan nilai dari *recall* atau *sensitivity* merupakan proporsi jumlah kasus positif yang sebenarnya yang diprediksi positif secara benar. *Recall* digunakan pula dalam psikologi untuk menjelaskan proses mengingat yang dikerjakan otak manusia (Powers dan Ailab 2011). Kata lain untuk recall dalam bahasa inggris adalah remember, recollect, remind. Di bidang IR *(Information Retrieval*), *recall* berkaitan dengan kemampuan menemukan kembali butir informasi yang sudah tersimpan. Jadi, terjemahan bebasnya mungkin adalah ”penemuan kembali”. Untuk menghitung nilai *recall* digunakan rumus sebagai berikut :

$$Recall = \frac{TP}{TP + FN} \times 100\,\% \ldots\ldots (12)$$

2. ***Precision***

Precision merupakan jumlah kelompok dokumen yang relevan dari total jumlah dokumen yang ditemukan oleh system. Nilai *precision* atau dikenal juga dengan nama *confidence* merupakan proporsi jumlah kasus yang diprediksi positif yang juga positif benar pada data yang sebenarnya (Powers dan Ailab 2011) . *Precision* dapat diartikan sebagai kepersisan atau kecocokan (antara permintaan informasi dengan jawaban terhadap permintaan itu. seberapa persis atau cocok data tersebut untuk keperluan prediksi, bergantung pada seberapa relevan data tersebut. Untuk menghitung nilai *Precision* digunakan rumus sebagai berikut (Dwiyantoro 2017) :

$$Precision = \frac{TP}{TP + FP} \times 100\% \ldots\ldots (13)$$

3. ***Accuracy***

Accuracy merupakan persentase jumlah *record* data yang diklasifikasikan secara benar oleh sebuah algoritma dapat membuat klasifikasi setelah dilakukan pengujian pada hasil klasifikasi tersebut (Han, Kamber, dan Pei 2006).

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} x\ 100\% \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (14)$$

4. *Matthews Correlation Coefficient* (MCC)

Matthews Correlation Coefficient (MCC) pertama kali diperkenalkan oleh B.W. Matthews untuk menilai kinerja prediksi struktur sekunder protein. Kemudian, itu menjadi ukuran kinerja yang banyak digunakan dalam penelitian biomedis. MCC dan *Area Under ROC Curve* (AUC) telah dipilih sebagai metrik elektif dalam inisiatif yang dipimpin FDA AS MAQC-II yang bertujuan untuk mencapai sebuah konsensus tentang praktik terbaik untuk pengembangan dan validasi model prediktif untuk personalisasi obat (Boughorbel, Jarray, dan El-Anbari 2017).

Matthews Correlation Coefficient (MCC) didefinisikan dalam hal *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN). Ini juga dapat ditulis ulang dalam hal TP, γ dan π sebagai berikut:

$$MCC= \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} x\ 100\% \ldots\ldots (15)$$

5. *True Positive Rate (TPR) or Sensitivity dan True Negative Rate (TNR) or Specificity*

Sensitivity dan specificity di gunakan penilaian prediksi mencakup berbagai jenis informasi. Misalnya untuk kondisi penyakit tertentu,

penilaian terbaik yang mungkin dapat dipilih berdasarkan atribut ini. Sensitivitas, spesifisitas dan akurasi banyak digunakan statistik untuk menggambarkan prediksi. Secara khusus, mereka digunakan untuk mengukur seberapa bagus dan dapat diandalkan suatu prediksi. Sensitivitas mengevaluasi seberapa baik prediksi tersebut mendeteksi suatu penyakit positif. Spesifitas memperkirakan seberapa besar kemungkinan pasien tanpa penyakit dapat dikesampingkan dengan benar. Kurva ROC adalah presentasi grafis dari hubungan antara keduanya sensitivitas dan spesifisitas dan membantu menentukan model optimal melalui penentuan ambang terbaik untuk penilaian prediksi. Akurasi mengukur seberapa akurat prediksi mengidentifikasi dan mengecualikan kondisi tertentu. Ketepatan dari prediksi dapat ditentukan dari sensitivitas dan spesifisitas dengan adanya prevalensi (Zhu, Zeng, dan Wang 2010).

True Positive Rate (TPR) or Sensitivity adalah proporsi individu dengan kondisi positif yang diketahui untuk mana hasil prediksi positif (Zhu, Zeng, dan Wang 2010).

$$Sensitivity = \frac{TP}{TP + FN} x\ 100\% \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.. (16)$$

Rumus diatas dapat melihat penjelasannya pada table 2.1. Sensitivitas saja tidak dapat digunakan untuk menentukan apakah suatu tes berguna dalam praktik. Namun, tes dengan sensitivitas tinggi dapat dianggap sebagai indikator yang dapat diandalkan ketika hasilnya negatif, karena jarang meleset benar positif di antara mereka yang sebenarnya positif.

Misalnya, sensitivitas 100% berarti bahwa tes tersebut mengakui semua yang sebenarnya positif (Zhu, Zeng, dan Wang 2010).

True Negative Rate (TNR) or Specificity adalah Tingkat negatif yang sebenarnya adalah proporsi individu dengan kondisi negatif yang diketahui untuk mana tes hasilnya negatif. Angka ini sering disebut kekhususan (Zhu, Zeng, dan Wang 2010).

$$Specificity = \frac{TN}{TN + FP} x\ 100\ \% \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(17)$$

I. Penelitain Terdahulu

Penelitian ini dibuat tidak terlepas dari hasil penelitian-penelitian terdahulu yang pernah dibuat sehingga dijadikan sebagai bahan perbandingan hasil prediksi pada penelitian ini. Adapun hasil-hasil penelitian yang dijadikan perbandingan tidak terlepas dari topik yang diambil pada peneliat ini yaitu prediksi metilasi pada sequence protein arginine.

Penelitian yang di gunakan sebagai bahan acuan yaitu penelitian Pawan Kumar, Joseph Joy, Ashutosh Pandey, dan Dinesh Gupta pada tahun 2017 yang berjudul PRmePRed: *A protein arginine methylation prediction tool*, pada penelitian kumar dkk menggunakan metode *support vector machine* untuk pembuatan model prediksi. Penelitian kumar menggunakan *sequence* protein *arginine* dengan 5 jenis panjang *sequence* berbeda yaitu 19, 23, 27, 31 dan 35, dan dengan 3 jenis eksperimen yang berbeda yaitu percobaan data *training, testing* dan *independent*. *Feature extraction* yang digunakan *features Atchley factors, ASA, disorder, hydrophobicity, van der waal's volume* dan *AA frequency* sehingga keseluruhan variabel berjumlah 194 variabel. Pada

penelitian kumar dkk menggunakn CD-HIT 40 % kemiripan sebagai redundansi data atau menghilangkan kemiripan pada sequence.

Acuan penilaian hasil kinerja prediksi menggunakan akurasi, sensitivitas, spesifisitas dan MCC, panjang *sequence* terbaik pada prediksi yaitu dengan panjang 19, sehingga didapatkan akurasi terbesar pada eksperimen data *independent* yaitu 93 %, sedangkan untuk data *training* atau rasio 1:1 dan data *testing* mendapatkan akurasi masing masing 84% dan 90 %.

III. METODOLOGI PENELITIAN

A. Tempat dan Waktu Penelitian

1. Tempat

Penelitian ini dilakukan di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung yang beralamatkan di Jalan Prof. Dr. Soemantri Bojonegoro No. 1 Gedung Meneng, Bandar Lampung.

2. Waktu

Penelitian ini dilaksanakan pada semester ganjil tahun ajaran 2018/2019, dimulai dari awal Februari 2019 hingga akhir bulan Agustus 2019.

Pada penelitian tugas akhir ini penulis mendalami materi dengan mempelajari setiap materi tahap demi tahap hingga penelitian selesai. Pengumpulan data dilakukan setelah pengajuan judul dan dilakukan bersama dengan penyusunan proposal penelitian, penyusunan laporan dari mulai proposal laporan, Pembuatan program memerlukan waktu yang relative lama 10 hingga 11 minggu.

B. Data dan Alat

1. Data

Data yang digunakan pada penelitian ini adalah data yang diperoleh dari studi literature penelitian Kumar dkk pada tahun 2017 yang mengambil data sequence protein dari database UniProt (release 2015_06) dengan mencari istilah seperti “arginine”, “methylation”, “methylation sites”. penelitian Kumar, dkk terbagi menjadi 3 jenis percobaan yaitu percobaan dengan data training / rasio 1:1, percobaan data testing dan percobaan data independent, jumlah data penelitian kumar, dkk dapat dilihat pada Tabel. 7

Tabel. 7 Data *Sequence* Protein Arginine Metilasi (Kumar, dkk , 2017)

Kelas Data	**Dataset**	**Jumlah Sequence Protein**
Training	*Positif*	1038
	Negative	5190
Testing	*Positif*	260
	Negative	260
Independent	*Positif*	3033
	Negative	1131

2. Alat

Alat dan bahan yang digunakan pada penelitian ini adalah sebagai berikut:

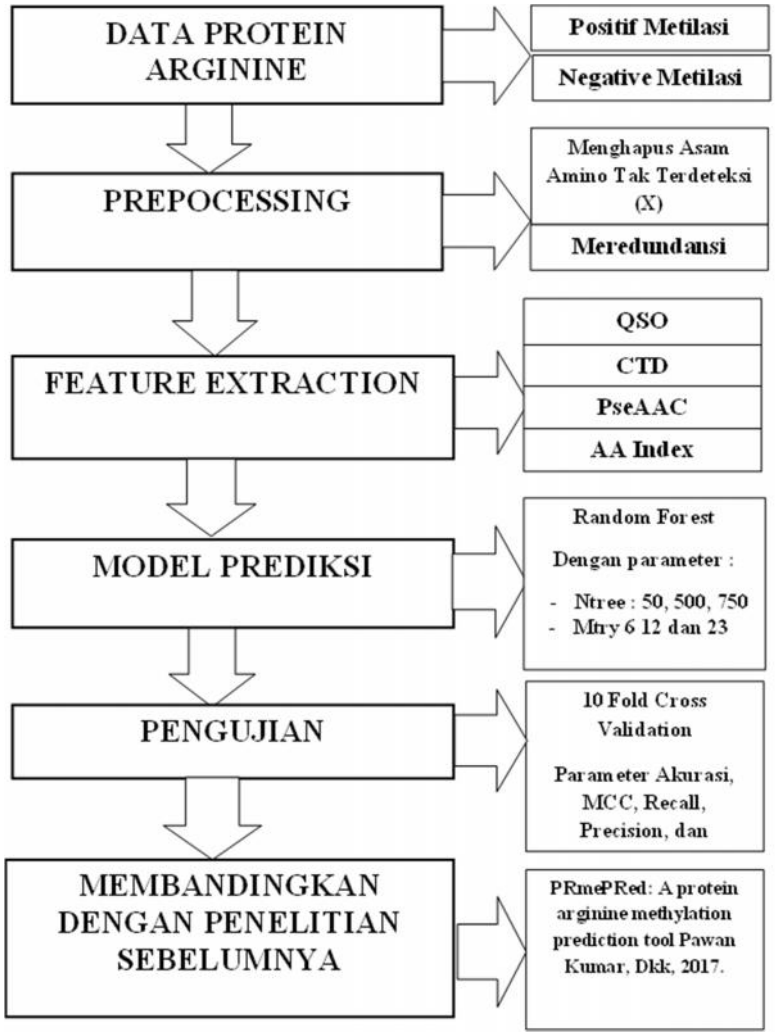
- Perangkat Keras

- Laptop dengan spesifikasi *Processor* Intel Core i3-3227U 1,9Ghz (Ivybridge) Celeron (R) CPU 1037U @ 1,80 GHz VGA Intel HD4000 Ram 6gb, HDD 750 GB, Led 14inch, DDR3 SDRAM, 4 threads, Cache 3 MB.
- Perangkat Lunak

  Perangkat lunak atau software yang digunakan dalam penelitian ini ialah Sistem Operasi Windows 7 Home Premium 64-Bit, Microsoft excel 2010, R Programming versi 3.5.1, R Studio versi 3.5.1, *package* caret versi 6.0-84, *package* BioSeqClass versi 1.40.0, *package* protr versi 1.6-1, dan *package* randomForest versi 4.6-14.

C. Metode Implementasi

Metode implementasi merupakan tahapan penelitian atau langkah-langkah yang dilakukan dalam penelitian. Metode implementasi yang dilakukan dalam Prediksi Metilasi Pada Sequence Protein dengan Metode Random Forest yang ditunjukan pada Gambar 7



Gambar 7. Tahapan Penelitian Prediksi Metilasi Sequence Protein Arginine menggunakan Random Forest.

Penjelasan metode implementasi yang dilakukan pada penelitian ini adalah sebagai berikut:

1. Feature Extraction / Ekstraksi Fitur

   Ekstraksi fitur merupakan proses pengambilan ciri yang terdapat didalam objek yang ada dalam gambar. Proses ekstraksi fitur bertujuan untuk mengambil atau mengekstraksi nilai-nilai unik dari suatu objek yang membedakan dengan objek yang lain. Pada penelitian ini menggunakan metode protein deskriptor adalah substruktur lokal dari molekul protein,

yang memungkinkan kita untuk membagi masalah asli menjadi satu set subproblem dan, akibatnya, untuk mengusulkan solusi algoritmik yang lebih efisien. Protein descriptor disini menggunakan package yang ada pada bahasa pemrograman R, yaitu package BioSeqClass dan Protr, dengan menggunakan 4 macam protein diskriptor yaitu CTD, QSO, Pseudo Amino Acid Composite dan AA index.

2. *Classification*

   Klasifikasi pada penelitian ini menggunakan *random forest* dimana klasifikasi dihasilkan saat peubah *respon* berupa data kategorik yaitu berupa data protein termetilasi atau pun tidak termetilasi. Penggunaan metode random forest untuk menghasilkan pohon gabungan telah memberikan dugaan yang lebih tinggi akurasinya dibandingkan dengan pohon tunggal.

D. Metode Pengujian

1. *Cross Validation*

K fold cross validation digunakan untuk melakukan pengujian, pada pengujian *cross validation* banyak menggunakan k 1 kali lipat data untuk belajar satu atau lebih model, pada penelitian ini menggunakan k 10, k 9 digunakan sebagai data training dan k 1 di gunakan sebagai data testing. Selanjutnya yang dipelajari model diminta untuk membuat prediksi tentang data dalam lipatan validasi. K-fold cross validation merupakan metode yang membagi himpunan contoh secara acak menjadi K himpunan bagian.

2. Evaluasi Matrik

Evaluasi matrik digunakan sebagai penilaian terhadap penelitian ini, hal-hal yang dinilai pada evaluasi matrik di penelitian ini ialah :

- Menilai *recall* dimana penilaian ini menilai dataset yang relevan dari data seluruh dataset yang diteliti.
- Penilaian *precision* dimana jumlah kasus yang diprediksi positif yang juga positif benar pada data yang sebenarnya.
- Penilaian terhadap akurasi prediksi bervariasi antara *sequence* protein yang terletak di posisi yang relatif berbeda.
- Melakukan penilaian atribut kelas yang menghasilkan nilai korelasi.

V. SIMPULAN DAN SARAN

A. Simpulan

Berdasarkan hasil penelitian yang dilakukan, dapat diambil simpulan sebagai berikut:

1. *Feature extraction* dengan menggunakan CTD, AA index, dan PseAAC pada package BioSeqClass dan QSO pada package Protr, masing-masing *feature extraction* memiliki variabel berjumlah CTD 21 variabel, AA index 19 variabel, PseAAC 24 variabel dan QSO 74 variabel, sehingga jumlah keseluruhan 138 variabel .
2. Data *Independent* memiliki kinerja lebih baik yaitu sebesar 98,08 % dibandingkan dengan data Rasio 1:1 dan *Testing* namun tidak efektif dikarenakan lebih banyak data dengan kelas negative sehingga kurang efektif dalam memprediksi metilasi pada protein *arginine*.
3. Percobaan metode *random forest* pada data Rasio 1:1 dan data *independent* memiliki kinerja lebih baik dari pada penelitian Kumar, dkk menggunakan metode SVM .

B. Saran

Berdasarkan penelitian yang dilakukan maka diperoleh beberapa saran untuk pengembangan penelitian ini lebih lanjut sebagai berikut:

1. Pada penelitian selanjutnya diharapkan agar menggunakan metode klasifikasi lainnya, misalnya KNN atau *Decision Tree* untuk melihat hasil klasifikasi yang berbeda, untuk melihat perbandingan hasil dan kualitas kinerja klasifikasi.
2. Menambahkan beberapa *feature extraction* yang ada pada *package* PROTR atau BioSeqClass, ataupun dengan menggunakan *package* yang lainnya.
3. Mencoba dengan *dataset* yang lain yang ada pada penelitian ini untuk klasifikasi data *Training* rasio 1:1.

DAFTAR PUSTAKA

Akram, M., H. M. Asif, M. Uzair, Naveed Akhtar, Asadullah Madni, S. M. Ali Shah, Zahoor Ul Hasan, dan Asmat Ullah. 2011. “Amino acids: A review article.” *Journal of Medicinal Plants Research* 5 (17): 3997–4000.

Bedford, Mark T., dan Stéphane Richard. 2005. “Arginine methylation: An emerging regulator of protein function.” *Molecular Cell* 18 (3): 263–72. https://doi.org/10.1016/j.molcel.2005.04.003.

Boughorbel, Sabri, Fethi Jarray, dan Mohammed El-Anbari. 2017.“Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric. ”*PLoS ONE* 12(6):1–17. https://doi.org/10.1371/journal.pone.0177678.

Breiman, L. 2001. “(impo)Random forests(book).” *Machine learning* 5–32.

Buck-Koehntop, Bethany A., Robyn L. Stanfield, Damian C. Ekiert, Maria A. Martinez-Yamout, H. Jane Dyson, Ian A. Wilson, dan Peter E. Wright. 2012. “Molecular basis for recognition of methylated and specific DNA sequences by the zinc finger protein Kaiso.” *Proceedings of the National Academy of Sciences of the United States of America* 109(38):15229–34.

Cawley, Gavin C., dan Nicola L.C. Talbot. 2003. “Efficient leave-one-out cross-validation of kernel fisher discriminant classifiers.” *Pattern Recognition* 36 (11): 2585–92. https://doi.org/10.1016/S0031-3203(03)00136-5.

Choras, Ryszard S. 2007. “Image Feature Extraction Techniques and Their Applications for CBIR and Biometrics Systems.” *International Journal of Biology and Biomedical Engineering* 1 (1): 6–15.

Dewi, Nariswari Karina, Utami Dyah Syafitri, Soni Yadi Mulyadi, Mahasiswa Departemen Statistika, dan Departemen Statistika. 2011. “Penerapan Metode Random Forest Dalam Driver Analysis.” *Forum Statistika Dan Komputasi* 16 (1): 35–43.

Didonna, Alessandro, dan Federico Benetti. 2016. “Post-translational modifications in neurodegeneration.” *AIMS Biophysics* 3 (1): 27–49. https://doi.org/10.3934/biophy.2016.1.27.

Dwiyantoro. 2017. “Sistem Temu Kembali Informasi.” *Informatikalogi.Com*, no. 2003. https://informatikalogi.com/sistem-temu-kembali-informasi/.

Genuer, Robin, Jean Michel Poggi, dan Christine Tuleau-Malot. 2015. “VSURF: An R package for variable selection using random forests.” *R Journal* 7 (2): 19–33.

Han, Jiawei, Micheline Kamber, dan Jian Pei. 2006. “Data mining: concepts and techniques. 2001.” *San Francisco: Morgan Kauffman*.

Johar., Asahar, Delfi Yanosma., dan Kurnia Anggriani. 1999. “Implementasi Metode K-Nearest Neighbor (Knn) Dan Simple Additive Weighting (Saw) Dalam Pengambilan Keputusan Seleksi Penerimaan Anggota Paskibraka.” *Clinics in Podiatric Medicine and Surgery* 16 (4): 725–42. http://www.embase.com/search/results?subaction=viewrecord&from=export&id=L29491586%0Ahttp://utahprimoprod.hosted.exlibrisgroup.com/openurl/UTAH/UTAHS?sid=EMBASE&issn=08918422&id=doi:&atitle=Osteochondral+lesions+of+the+talar+dome%3A+A+comprehensive+revi.

Klusowski, Jason M. 2018. “Complete Analysis of a Random Forest Model” 13: 1063–95. http://arxiv.org/abs/1805.02587.

Kumar, Pawan, Joseph Joy, Ashutosh Pandey, dan Dinesh Gupta. 2017. “PRmePRed: A protein arginine methylation prediction tool.” *PLoS ONE* 12 (8): 1–12. https://doi.org/10.1371/journal.pone.0183318.

Lee, David Y., Catherine Teyssier, Brian D. Strahl, dan Michael R. Stallcup. 2005. “Role of protein methylation in regulation of transcription.” *Endocrine Reviews* 26(2):147–70.

Leidiyana, Henny. 2013. “Penerapan Algoritma K-Nearest Neighbor Untuk Penentuan Resiko Kredit Kepemilikan Kendaraan Bemotor.” *Penelitian Ilmu Komputer Sistem Embedded dan Logic* 1 (1): 65–76.

Lukito, Yuan, dan Antonius R. Chrismanto. 2015. “Perbandingan Metode-Metode Klasifikasi untuk Indoor Positioning System.” *Jurnal Teknik Informatika dan Sistem Informasi* 1 (2): 123–31. https://doi.org/10.28932/jutisi.v1i2.373.

Nugroho, Anto Satriyo, Arief Budi Witarto, dan Dwi Handoko. 2003. “Application of Support Vector Machine in Bioinformatics.” *Proceeding of Indonesian Scientific Meeting in Central Japan*.

Powers, D M W, dan Ailab. 2011. “Evaluation: From Precision, Recall and F-Measure To Roc, Informedness, Markedness & Correlation.” *Journal of Machine Learning Technologies ISSN* 2 (1): 2229–3981. http://www.bioinfo.in/contents.php?id=51.

Refaeilzadeh, Payam, Lei Tang, dan Huan Liu. 2017. “Cross-Validation.”

Language in Zambia, 271–91. https://doi.org/10.4324/9781315106786-12.

Reitermanov, Z. 2010. “Data Splitting,” 31–36.

Santos, Ana L., dan Ariel B. Lindner. 2017. “Protein Posttranslational Modifications: Roles in Aging and Age-Related Disease.”*Oxidative Medicine and Cellular Longevity*2017.https://doi.org/10.1155/2017/5716409.

Saracoglu, Ö Galip. 2008. “An artificial neural network approach for the prediction of absorption measurements of an evanescent field fiber sensor.” *Sensors* 8 (3): 1585–94. https://doi.org/10.3390/s8031585.

Sikic, Kresimir, dan Oliviero Carugo. 2010. “Protein sequence redundancy reduction: comparison of various methods.” *Bioinformation* 5 (6): 234–39. https://doi.org/10.6026/97320630005234.

Thermo Fisher Scientific. 2016. “Overview of Post-Translational Modifications (PTMs).”thermofisher.com.2016.https://www.thermofisher.com/id/en/home/life-science/protein-biology/protein-biology-learning-center/protein-biology-resource-library/pierce-protein-methods/overview-post-translational-modification.html.

Venkatraman, Sitalakshmi, dan Siddhivinayak Kulkarni. 2012. “MapReduce neural network framework for efficient content based image retrieval from large datasets in the cloud.” *Proceedings of the 2012 12th International Conference on Hybrid Intelligent Systems, HIS 2012* 8 (4): 63–68. https://doi.org/10.1109/HIS.2012.6421310.

Vuzman, Dana, Yonit Hoffman, dan Yaakov Levy. 2012. “Modulating protein-DNA interactions by post-translational modifications at disordered regions.” *Pacific Symposium on Biocomputing* 188–99.

Walsh, Christopher T., Sylvie Garneau-Tsodikova, dan Gregory J. Gatto. 2005. “Protein posttranslational modifications: The chemistry of proteome diversifications.” *Angewandte Chemie - International Edition* 44(45):7342–72.

Wu, Xindong, Vipin Kumar, Quinlan J. Ross, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J. McLachlan, et al. 2008. *Top 10 algorithms in data mining*. *Knowledge and Information Systems*. Vol. 14. https://doi.org/10.1007/s10115-007-0114-2.

Xiao, Nan, Dong-sheng Cao, Zhu Min-Feng, dan Qing-song Xu. 2015. “protr : R package for generating various numerical representation schemes of protein sequence.” *Bioinformatics* 31 (11): 1857–59.

Zhang, Weiwei, Tim D. Spector, Panos Deloukas, Jordana T. Bell, dan Barbara E. Engelhardt. 2015. “Predicting genome-wide DNA methylation using methylation marks, genomic position, and DNA regulatory elements.”

Genome Biology 16 (1): 1–32. https://doi.org/10.1186/s13059-015-0581-9.

Zhu, Wen, Nancy Zeng, dan Ning Wang. 2010. “Sensitivity , Specificity , Accuracy , Associated Confidence Interval and ROC Analysis with Practical SAS ® Implementations K & L consulting services , Inc , Fort Washington , PA Octagon Research Solutions , Wayne.” *NESUG : Health Care and Life Sciences*, 1–9.