IMPLEMENTASI ALGORITMA *nu-SUPPORT VECTOR MACHINE* PADA MODEL KLASIFIKASI SEKUENS DNA MANUSIA
Studi Kasus: Penderita Diabetes Mellitus

SKRIPSI

OLEH:
MUHAMMAD FATHUN NUHA
NIM. 200601110106

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024

IMPLEMENTASI ALGORITMA *nu-SUPPORT VECTOR MACHINE* PADA MODEL KLASIFIKASI SEKUENS DNA MANUSIA
Studi Kasus: Penderita Diabetes Mellitus

SKRIPSI

Diajukan Kepada
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
untuk Memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Matematika (S.Mat)

Oleh
MUHAMMAD FATHUN NUHA
NIM. 200601110106

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2024

IMPLEMENTASI ALGORITMA *nu-SUPPORT VECTOR MACHINE* PADA MODEL KLASIFIKASI SEKUENS DNA MANUSIA
Studi Kasus: Penderita Diabetes Mellitus

SKRIPSI

Oleh
Muhammad Fathun Nuha
NIM. 200601110106

Telah Disetujui Untuk Diuji

Malang, 26 Agustus 2024

Dosen Pembimbing I	Dosen Pembimbing II
Ari Kusumastuti, M.Pd., M.Si.	Mohammad Nafie Jauhari, M.Si.
NIP. 19770521 200501 2 004	NIPPPK. 19870218 202321 1 018

Mengetahui,
Ketua Program Studi Matematika

Dr. Elly Susanti, M.Sc.
NIP. 19741129 200012 2 005

IMPLEMENTASI ALGORITMA *nu-SUPPORT VECTOR MACHINE* PADA MODEL KLASIFIKASI SEKUENS DNA MANUSIA
Studi Kasus: Penderita Diabetes Mellitus

SKRIPSI

Oleh
Muhammad Fathun Nuha
NIM. 200601110106

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Matematika (S.Mat.)
Tanggal 30 Agustus 2024

Ketua Penguji	: Dr. Usman Pagalay, M.Si.	.....
Anggota Penguji 1	: Juhari, M.Si.	.....
Anggota Penguji 2	: Ari Kusumastuti, M.Pd., M.Si.	.....
Anggota Penguji 3	: Mohammad Nafie Jauhari, M.Si.	.....

Mengetahui,
Ketua Program Studi Matematika

Dr. Elly Susanti, M.Sc.
NIP. 19741129 200012 2 005

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama	: Muhammad Fathun Nuha
NIM	: 200601110106
Program Studi	: Matematika
Fakultas	: Sains dan Teknologi
Judul Skripsi	: Implementasi Algoritma *nu-Support Vector Machine* Pada Model Klasifikasi Sekuens DNA Manusia Studi Kasus: Penderita Diabetes Mellitus

Menyatakan dengan sebenar-benarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya sendiri, bukan merupakan pengambilan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan dan pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada Daftar Pustaka. Apabila di kemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 30 Agustus 2024

Pernyataan,

Muhammad Fathun Nuha

NIM. 200601110106

MOTO

"*Allah Ta'ala berfirman, 'Aku sesuai persangkaan hamba-Ku.*" (HR. Bukhari no. 7405 dan Muslim no. 2675)

"*Apapun yang menjadi takdirmu, akan mencari jalannya menemukanmu*." – Abi bin Abi Thalib

"*Menyesali nasib tidak akan mengubah keadaan. Terus berkarya dan bekerjalah yang membuat kita berharga.*" – Abduraahman Wahid

PERSEMBAHAN

Skripsi ini penulis dipersembahkan:

Pertama, kepada Ibu tercinta Ninik Isqiqomah yang telah melahirkan, membimbing dengan kesabaran yang luar biasa, serta menjadi penyemangat penulis hingga terselesaikannya pendidikan sarjana ini.

Kedua, kepada Bapak paling hebat Tjahjoso Saptanaadi yang telah menjadi inspirasi dan teladan bagi penulis, serta memberikan dukungan penuh baik dari segi finansial maupun pemikiran.

Ketiga, kepada ketiga srikandi dari keluarga penulis, Hana, Lika dan Ifa yang turut memberi warna dan kehangatan saat penulis mengalami banyak permasalahan.

Keempat dan terakhir, kepada Bude Asih, keluarga serta teman dan sahabat penulis lainnya yang memberikan dukungan, motivasi dan semangat selama perjalanan menempuh penddidikan.

KATA PENGANTAR

Segala puji dan syukur kehadirat Allah SWT yang telah senantiasa memberikan karunia dan rahmat-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul "Implementasi Algoritma *nu-Support Vector Machine* Pada Model Klasifikasi Sekuens DNA Manusia Studi Kasus: Penderita Diabetes Mellitus" ini yang disusun sebagai salah satu syarat kelulusan dari program Sarjana Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Selama penulisan skripsi ini, penulis mendapatkan banyak pengetahuan, pengalaman, bimbingan, dukungan, juga arahan dan saran dari berbagai pihak yang telah membantu penulis dalam perjalanan menyelesaikan skripsi ini. Untuk itu penulis ingin menyampaikan ucapan terima kasih kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim.
2. Dr. Sri Harini, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim.
3. Dr. Elly Susanti, S.Pd., M.Sc., selaku ketua Program Studi Matematika, Universitas Islam Negeri Maulana Malik Ibrahim.
4. Ari Kusumastuti, M.Pd., M.Si., selaku dosen pembimbing I.
5. Mohammad Nafie Jauhari, M.Si., selaku dosen pembimbing II.
6. Seluruh dosen Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim.
7. Kedua orang tua penulis yang selalu memberikan dukungan penuh, nasihat, keteladanan, dan kasih sayang dalam setiap perjalanan penulis. Terima kasih atas cinta, pemahaman, dan doa yang selalu diberikan.
8. Teman-teman dan rekan keorganisasian baik di lingkungan kampus maupun luar kampus yang telah memberikan banyak pembelajaran dan inspirasi yang luar biasa sehingga peneliti mendapatkan motivasi untuk terus berkembang dan belajar banyak hal baru serta seru.
9. Seluruh sahabat dan teman-teman seperjuangan dari Matematika.

10. Sahabat sekaligus teman paling istimewa yang telah hadir di hidup penulis, Alfi, terima kasih banyak atas segala bantuan, perasaan dan dukungan yang diberikan pada penulis dalam menyelesaikan skripsi ini.
11. Seluruh sahabat dan teman-teman seperjuangan dari kamar 10B Pondok Pesantren Gasek Sabilurrosyad.
12. Seluruh sahabat dan teman-teman seperjuangan lainnya dari penulis yang telah memberikan dukungan moral, semangat, motivasi, dan Kerjasama yang tak tergantikan. Kebersamaan dalam perjalanan dan petualangan penulis selama masa kuliah telah memberikan keceriaan dan kekuatan dalam menghadapi setiap hambatan.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna, namun dengan ikhlas, penulis mempersembahkan sebagai bentuk apresiasi dan penghormatan kepada semua pihak yang telah memberikan kontribusi, dukungan, dan inspirasi kepada penulis. Semoga skripsi ini bermanfaat bagi pembaca dan dapat menjadi pijakan untuk perjalanan ilmiah yang lebih luas.

Malang, 30 Agustus 2024

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR LAMPIRAN

ABSTRAK

Nuha, Muhammad Fathun, 2024. **Implementasi Algoritma *nu-Support Vector Machine* Pada Model Klasifikasi Sekuens DNA Manusia Studi Kasus: Penderita Diabetes Mellitus**. Skripsi. Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing (I) Ari Kusumastuti, M.Pd., M.Si. (II) Mohammad Nafie Jauhari, M.Si.

Kata Kunci: Bioinformatika, Diabetes Mellitus, Klasifikasi DNA, *Machine Learning*, *nu-Support Vector Machine*

Perkembangan teknologi machine learning telah memungkinkan proses analisis data yang lebih kompleks dan akurat dalam bidang kesehatan dan diagnosa penyakit seperti diabetes mellitus. Penelitian ini bertujuan untuk mengimplementasikan algoritma *nu-Support Vector Machine* (*nu*-SVM) pada pembuatan model klasifikasi sekuens DNA manusia sehingga dapat mengidentifikasi penderita diabetes mellitus. Data sekuens DNA yang digunakan dalam penelitian ini merupakan data DNA dari manusia yang menderita diabetes melitus dan non-diabetes sebanyak 2509 data. Data tersebut diolah melalui tahapan *preprocessing* yang mencakup tahap *cleaning data* DNA, *k-mers encoding*, transformasi *CountVectorizer*, dan *oversampling* menggunakan SMOTE. Pembuatan model klasifikasi dilakukan menggunakan algoritma *nu*-SVM, dan dievaluasi menggunakan *Stratified K-Fold Cross-Validation* untuk mengetahui performa model. Hasilnya, Implementasi *nu*-SVM pada model klasifikasi sekuens DNA menghasilkan akurasi rata-rata sebesar 90%, presisi rata-rata sebesar 90,18%, serta *recall* dan *F1-score* rata-rata sebesar 90%. Hasil ini menunjukkan jika algoritma *nu*-SVM terbukti mampu mengidentifikasi sekuens DNA manusia dengan cepat dan akurat sehingga mampu memberikan kontribusi terhadap pengembangan sistem diagnosis dini penyakit yang lebih efektif dan efisien.

ABSTRACT

Nuha, Muhammad Fathun, 2024. **Implementation of *nu*-Support Vector Machine Algorithm on Human DNA Sequence Classification Model Case Study: Diabetes Mellitus Patients**. Thesis. Department of Mathematics, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Supervisor (I) Ari Kusumastuti, M.Pd., M.Si. (II) Mohammad Nafie Jauhari, M.Si.

Keywords: Bioinformatics, Diabetes Mellitus, DNA Classification, Machine Learning, *nu*-Support Vector Machine

The development of machine learning technology has enabled more complex and accurate data analysis processes in the field of health and diagnosis of diseases such as diabetes mellitus. This research aims to implement the *nu*-Support Vector Machine (*nu*-SVM) algorithm in human DNA sequence classification modeling so as to identify people with diabetes mellitus. The DNA sequence data used in this research are 2509 human DNA data consist of those human DNA which are indicated with diabetes mellitus and those which are not. The data is processed through preprocessing stages which include cleaning DNA data, k-mers encoding, CountVectorizer transformation, and oversampling using SMOTE. Classification modeling was performed using the *nu*-SVM algorithm, and evaluated using Stratified K-Fold Cross Validation to determine model performance. As a result, the implementation of *nu*-SVM in the DNA sequence classification model resulted in an average accuracy of 90%, an average precision of 90.18%, and an average recall and F1-score of 90%. These results show that the *nu*-SVM algorithm is proven to be able to identify human DNA sequences accurately so that it can contribute to the development of a more effective and efficient disease early diagnosis system.

المستخلص البحث

النهى، محمد فتح، ٢٠٢٤، تنفيذ خوارزمية آلة ناقل الدعم نو في نموذج تصنيف تسلسل الحمض النووي البشري، دراسة حالة: مرضى السكري، البحث العلمي قسم الرياضيات، كلية العلوم والتكنولوجيا جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج، المشرفة، (١) آري كوسوماستوتي، الماجستير، (٢) محمد نافع جوهري، الماجستير

الكلمات المفتاحية : خوارزمية آلة ناقل الدعم نو، تصنيف الحمض النوو، داء السكرى، التعلم الآلي بيونيفارماتيك

لقد اتاح تطور تكنولو جية التعلم الالي لاجراء عمليات تحليل البيانات اكثر تعقيدا ودقة في مجال الصحة وتشخيص الامراض مثل داء السكري الميليتوس، يهدف هذا البحث الى تطبيق خوارزمية الة ناقل الدعم النانوي (*nu*-SVM) ان بيانات تسلسل، الحمض النووى، المستخدمة فى هذه الدراسة هي بيانات بشرية تعانى من داء السكري بقدر ٢٥٠٩ بيانات. تمت معالجة البيانات من خلال مرحلة المعالجة المسبقة التى تضمنت تنظيف بيانات الحمض النووي وترميز (k-mers) وتحويل ناقل العد، واخذ عينات زائدة باستخدام (SMOTE) يتم اجراء نمذجة التصنيف باستخدام خوارزمية (*nu*-SVM)، ويتم تقييمها باستخدام التحقق المقاطع الطبقلى (*K-Fold Cross Validation*) لتحديد اداء النموذج. نتح عن تطبيق خوارزمية (*nu*-SVM) على النموذج تضيف تسلسل الحمض النووي متوسط دقة بنسبة 90%، ومتوسط دقة بنسبة 90,18%، ومتوسط استرجاع ودرجة (F1) بنسبة 90%. تظهر هذه النتائج ان خوارزمية (*nu*-SVM) اثبتت قدرتها على تحديد تسلسل الحمض النووي البشرى بسرعة ودقة حتى نتمكن من المساهمة فى تطوير نظام اكثر فعالية وكفاءة للتشحيص المبكر للامراض.

BAB I
PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi *machine learning* yang cukup pesat dalam beberapa tahun terakhir telah memungkinkan para peneliti menganalisis data yang lebih kompleks dan akurat. Hal ini membuka peluang baru dalam penelitian bidang kesehatan dan bioinformatika. Melalui berbagai teknik dalam *machine learning*, proses diagnosa terhadap berbagai macam penyakit dan gejala dapat dilakukan dengan lebih cepat, efisien serta dengan biaya klinis yang lebih murah (Kobat et al., 2021). Hal ini juga membuat para peneliti dapat melakukan analisis lebih mendalam dengan meneliti molekul biologis seperti sekuens DNA yang biasanya digunakan dalam identifikasi awal penyakit diabetes mellitus.

Diabetes melitus merupakan penyakit yang ditandai dengan kadar glukosa yang tinggi dalam darah atau hiperglikemia (ElSayed et al., 2023). Diabetes telah menjadi salah satu penyakit yang paling berbahaya yang menyebabkan kematian dari 1,5 juta orang di dunia pada 2019 (WHO, 2023). Seringkali penderita diabetes mellitus tidak sadar bahwa dirinya menderita diabetes hingga kondisi yang cukup parah. Hal ini terjadi karena minimnya gejala luar pada penderitanya sehingga menyebabkan diperlukannya diagnostik lebih mendalam pada kondisi biologis penderita.

Pada molekul biologis yang lebih kecil, diabetes melitus terjadi karena adanya mutasi dalam urutan gen insulin yang menyebabkan ketidakseimbangan dalam produksi hormon insulin dan mengganggu proses penyerapan glukosa dalam tubuh (Hamza et al., 2023). Insulin sendiri termasuk salah satu protein yang

diproduksi oleh DNA manusia. Hal ini memungkinkan proses diagnosa penyakit diabetes melitus dilakukan melalui identifikasi sekuens DNA manusia baik yang mengidap maupun yang berpotensi diabetes melitus.

Sekuens DNA atau Asam deoksiribonukleat merupakan merupakan suatu molekul biologis kompleks yang membawa informasi terkait dengan pewarisan sifat dan karakteristik suatu organisme. Sekuens DNA terdiri dari dua rantai nukleotida yang kompleks dan panjang mencapai 32.000 nukleotida sehingga sulit untuk diteliti tanpa bantuan teknologi tinggi (Gunasekaran et al., 2021). Pada penelitian ini, akan digunakan data sekuens DNA manusia penderita dan bukan penderita diabetes melitus yang didapat dari website NCBI. website NCBI merupakan portal komunitas yang menyediakan data sekuens gen, DNA dan protein berkualitas tinggi dari berbagai penelitian dan kolaborasi internasional. Sekuens DNA tersebut berbentuk string yang terdiri dari kombinasi nukleotida yang ada dalam DNA manusia yakni adenin (A), sitosin (C), guanin (G), dan timin (T) (Akkaya & Kalkan, 2021).

Data sekuens DNA manusia yang masih berbentuk string dari kombinasi nukleotida harus diolah terlebih dahulu agar dapat dimasukkan ke dalam model *machine learning*. Oleh karena itu sekuens DNA tersebut akan terlebih dahulu melalui proses *preprocessing data* untuk mengubahnya menjadi data numerik. Pertama data sekuens DNA akan melalui tahap *cleaning* yang mana data-data DNA tersebut akan dibersihkan dari sekuens DNA yang terindikasi duplikat, *missing value* dan juga sekuens DNA yang berkualitas buruk.

Tahapan berikutnya yang akan dilakukan ialah tahap *encoding* dan transformasi data untuk menangkap pola penting pada sekuens DNA. Langkah

pertama dari tahap ini ialah melakukan proses *encoding* menggunakan prinsip k-*mers* pada sekuens DNA untuk menangkap pola dan fitur dari pasangan k-*nukleotida* yang terdapat dalam masing-masing sekuens. Kemudian, data sekuens DNA hasil dari k-*mers encoding* sebelumnya akan ditransformasikan menjadi data numerik menggunakan metode transformasi *CountVectorizer*. *CountVectorizer* merupakan metode transformasi yang digunakan untuk menentukan nilai numerik berdasarkan frekuensi frase atau kata yang muncul dalam satu sekuens diantara semua sekuens DNA yang ada (Assery et al., 2019). Data sekuens yang telah mengalami tahapan ini nantinya telah berbentuk data numerik yang siap pakai dan bisa dimasukkan ke dalam model *machine learning*.

Pada penelitian ini, proses pelatihan model klasifikasi dari *machine learning* akan menggunakan algoritma *nu-Support Vector Machine* atau *nu*-SVM. Klasifikasi merupakan merupakan salah satu teknik *machine learning* yang bekerja dengan mengidentifikasi kelompok suatu data berdasarkan kemiripan karakteristiknya. Algoritma *nu*-SVM sendiri dipilih karena kelebihannya dalam mempelajari data dengan kompleksitas dan dimensi fitur yang tinggi, seperti data molekul biologis dan DNA makhluk hidup. Algoritma ini bekerja dengan memilih beberapa titik data sebagai batas (*support vector*) untuk menghasilkan fungsi bidang pemisah (*hyperplane*) yang memisahkan data menjadi beberapa kelas dengan batas-batas paling optimal. Dalam algoritma ini, parameter *nu*, dengan rentang nilai antara 0 sampai 1, akan digunakan sebagai parameter yang membatasi batas minimal dari titik batas pendukung (*support vector*) sehingga akan didapatkan *hyperplane* yang mampu memisahkan kelas data secara optimal (Hatmal et al., 2021).

Sebelum mengambil kesimpulan terhadap model, penting untuk melakukan validasi dan evaluasi untuk mengetahui keakuratan dari model klasifikasi yang telah dibuat. Pada tahap ini akan melibatkan pembagian dataset menjadi subset *training* dan *testing*, serta penggunaan metrik evaluasi seperti akurasi. Kemudian metode validasi yang akan digunakan adalah *stratified* k-*fold cross-validation* untuk melakukan evaluasi model klasifikasi yang telah dibuat. *Stratified* k-*fold cross-validation* akan membagi dataset menjadi k-bagian dengan tiap bagian memiliki jumlah distribusi kelas yang sama (Hamed et al., 2023). Dengan jumlah sampel kelas yang sama, metode ini dapat menghasilkan evaluasi yang konsisten dan baik.

Pada penelitian sebelumnya, Hamed et al., (2023) telah melakukan penelitian dengan mengkombinasikan beberapa algoritma *machine learning* dan pencocokan pola untuk melakukan klasifikasi pada sekuens DNA nasi melalui data aroma beras. Hasilnya mereka mampu mendapatkan model dengan akurasi paling tinggi mencapai 96,3% menggunakan algoritma SVM Linear dengan performa yang lebih efisien dengan waktu eksekusi 10 detik, serta disusul oleh algoritma SVM RBF dan SVM Sigmoid dengan akurasi yang sama yakni sebesar 92,5% dengan performa 4 detik lebih lama yakni sekitar 14,2 detik (Sharabiani et al., 2023). Pada penelitian lainnya oleh Zhang et al. (2022) telah melakukan prediksi terhadap modifikasi DNA N6-*Methyladenine* pada nasi dan tikus. Hasilnya, mereka mampu membuat model *machine learning* dengan akurasi yang tinggi sekitar 92,04% dan 95,51% menggunakan algoritma DB-SVM (H. Zhang et al., 2022).

Klasifikasi suatu penyakit berdasarkan pola sekuens DNA-nya merupakan hal penting dalam dunia kesehatan. Allah SWT berfirman dalam melalui Al-Qur'an yakni dalam Surat Al-An'am ayat 141 (Kemenag, 2024) yang berbunyi:

وَهُوَ الَّذِيْٓ اَنْشَاَ جَنّٰتٍ مَّعْرُوْشٰتٍ وَّغَيْرَ مَعْرُوْشٰتٍ وَّالنَّخْلَ وَالزَّرْعَ مُخْتَلِفًا اُكُلُهٗ وَالزَّيْتُوْنَ وَالرُّمَّانَ مُتَشَابِهًا وَّغَيْرَ مُتَشَابِهٍۗ كُلُوْا مِنْ ثَمَرِهٖٓ اِذَآ اَثْمَرَ وَاٰتُوْا حَقَّهٗ يَوْمَ حَصَادِهٖۖ وَلَا تُسْرِفُوْاۗ اِنَّهٗ لَا يُحِبُّ الْمُسْرِفِيْنَۙ ﴿١٤١﴾

Artinya : "*Dialah yang menumbuhkan tanaman-tanaman yang merambat dan yang tidak merambat, pohon kurma, tanaman yang beraneka ragam rasanya, serta zaitun dan delima yang serupa (bentuk dan warnanya) dan tidak serupa (rasanya). Makanlah buahnya apabila ia berbuah dan berikanlah haknya (zakatnya) pada waktu memetik hasilnya. Akan tetapi, janganlah berlebih-lebihan. Sesungguhnya Allah tidak menyukai orang-orang yang berlebih-lebihan.*" (Q.S. Al-An'am ayat 141)

Menurut Ibnu Katsir, ayat di atas ialah sebagai bukti bahwa Allah SWT menciptakan segala sesuatu yang ada di dunia ini, termasuk tanaman dan buah-buahan. Allah juga telah menciptakan jenis tanaman dan buah-buahan berdasarkan karakteristik yang mereka punya sendiri. Ada jenis tanaman yang tumbuhnya secara merambat dan ada pula yang tidak. Pada buah-buahan pula, meskipun buah tersebut memiliki bentuk yang sama dan serupa seperti zaitun dan delima, namun rasa yang dihasilkan kedua buah tersebut sangatlah berbeda (bin Ishaq, 2013).

Hal ini menunjukkan bahwa klasifikasi, atau pengkategorian berdasarkan kesamaan ciri dan sifat, merupakan bagian yang tidak dapat dipisahkan dari proses penciptaan alam semesta. Klasifikasi memiliki peran penting dalam berbagai aspek kehidupan, termasuk dalam bidang ilmu pengetahuan dan kesehatan. Dalam ilmu pengetahuan, klasifikasi membantu para ilmuwan untuk mempelajari dan memahami berbagai fenomena dengan lebih baik, seperti klasifikasi pada gen dan

molekul biologis yang dapat membantu para ahli biologi untuk memahami fenomena mikrobiologis pada tubuh makhluk hidup (Lencz & Malhotra, 2009).

Klasifikasi juga sangat penting dalam bidang kesehatan. Dengan mengklasifikasikan penyakit berdasarkan karakteristiknya, seperti gejala, penyebab, dan tingkat keparahan, para dokter dapat mendiagnosis penyakit dengan lebih akurat dan menentukan tindakan pencegahan dan pengobatan yang tepat. Dengan demikian, ayat ini tentu saja sejalan dengan tujuan dari penelitian ini, yakni untuk mengidentifikasi dan mengklasifikasikan DNA dari manusia yang mengidap penyakit diabetes mellitus. Analisis tersebut sangat penting untuk dilakukan mengingat dapat memberikan wawasan berharga tentang pewarisan sifat, kerentanan penyakit, serta target terapi dan pengobatan potensial dari karakteristik DNA-nya (Hamza et al., 2023).

Berdasarkan uraian sebelumnya, telah dilakukan berbagai macam analisis dan klasifikasi pada barisan DNA menggunakan *machine learning*. Namun, pada penelitian ini, penulis ingin melakukan proses klasifikasi dengan metode yang berbeda yakni dengan algoritma *nu*-SVM. *nu*-SVM sebagai bentuk modifikasi dari SVM memiliki keunggulan daripada metode yang telah dilakukan sebelumnya khususnya dalam memprediksi data dari sekuens DNA yang cukup kompleks dan memiliki dimensi tinggi. Diharapkan, melalui penelitian ini, penulis berharap dapat berkontribusi dalam dunia penelitian dan kesehatan dengan mengoptimalkan metode pendeteksi diabetes mellitus yang lebih awal berdasarkan sekuens DNA.

1.2 Rumusan Masalah

Berdasarkan uraian pada latar belakang yang telah dipaparkan sebelumnya, maka rumusan masalah yang diambil dalam penelitian ini adalah sebagai berikut:

1. Bagaimana tahapan preprocessing dan persiapan data sekuens DNA manusia yang digunakan dalam model klasifikasi sekuens DNA pada penderita diabetes mellitus ?
2. Bagaimana tahapan pembuatan model klasifikasi sekuens DNA pada penderita diabetes mellitus menggunakan algoritma *nu-Support Vector Machine* ?
3. Bagaimana hasil evaluasi dari model klasifikasi sekuens DNA pada penderita diabetes mellitus menggunakan algoritma *nu-Support Vector Machine* ?

1.3 Tujuan Penelitian

Tujuan penelitian merupakan suatu hal penting dari sebuah penelitian. Oleh karena itu, berdasarkan uraian rumusan masalah pada sub bab sebelumnya, maka tujuan penelitian ini sebagai berikut:

1. Untuk mengetahui bagaimana tahapan preprocessing dan persiapan data sekuens DNA manusia yang digunakan dalam model klasifikasi sekuens DNA pada penderita diabetes mellitus.
2. Untuk mengetahui bagaimana tahapan pembuatan model klasifikasi sekuens DNA pada penderita diabetes mellitus menggunakan algoritma *nu-Support Vector Machine.*

3. Untuk mengetahui bagaimana hasil evaluasi dari model klasifikasi sekuens DNA pada penderita diabetes mellitus menggunakan algoritma *nu-Support Vector Machine*.

1.4 Manfaat Penelitian

Melakukan sebuah penelitian harus sejalan dengan manfaat yang akan dihasilkan dikemudian hari. Hal ini berarti, penelitian yang dilakukan bukan hanya sebagai kajian teori belaka namun juga dapat membawa dampak positif dalam kehidupan nyata. Maka berdasarkan uraian tujuan diatas penelitian ini diharapkan mampu memberikan kontribusi dan manfaaat kepada pembaca sebagai berikut:

1. Dengan mengetahui dan menerapkan tahapan preprocessing dan persiapan data sekuens DNA manusia yang benar maka akan didapat data yang berintegritas dan berkualitas baik sehingga dapat lebih akurat pada saat digunakan dalam pembuatan model klasifikasi sekuens DNA pada penderita diabetes mellitus.
2. Dengan mengetahui dan menerapkan tahapan pembuatan model klasifikasi sekuens DNA pada penderita diabetes mellitus menggunakan algoritma *nu-Support Vector Machine* yang baik maka akan dapat dihasilkan model yang akurat untuk mengklasifikasikan DNA manusia penderita diabetes mellitus.
3. Dengan mengetahui seberapa baik akurasi dan performa model melalui hasil evaluasi dari model klasifikasi sekuens DNA pada penderita diabetes mellitus yang telah dibuat maka dapat diketahui seberapa akurat model yang dibuat dalam mengelompokkan DNA manusia penderita diabetes mellitus melalui penerapan algoritma *nu-Support Vector Machine*.

1.5 Batasan Masalah

Agar tidak terjadi perluasan masalah, maka penelitian ini menggunakan Batasan-batasan masalah sebagai berikut:

1. Nilai k-*mers* yang digunakan dalam proses membagi sekuens DNA menjadi substring adalah 3-*mers* sesuai dengan prinsip kodon dan juga agar biaya komputasi tidak terlalu besar.
2. Jumlah kombinasi kata yang digunakan pada *n-bag-of-words* ialah $n = 4$ kata.
3. Perancangan model klasifikasi menggunakan algoritma *machine learning nu-Support Vector Machine* dengan nilai parameter *nu* (ν) = 0,1.
4. Proses validasi menggunakan *stratified k-fold cross validation* akan menggunakan nilai k = 5 yang membagi dataset menjadi 5 bagian atau subset, dengan perulangan sebanyak 3 kali proses pembagian data agar pembagian data acak dapat dilakukan secara merata sehingga model yang dihasikan dapat dievaluasi dengan lebih konsisten.

1.6 Definisi Istilah

Terdapat beberapa istilah yagng digunakan pada penelitian ini diantaranya:

Machine Learning : Ilmu yang mempelajari pembuatan sistem Computer yang dapat belajar dari data

Support Vector : Titik data yang terletak diantara bidang pemisah atau *hyperplane*

Hyperplane : Sebuah bidang multidimensi yang memisahkan antar kelas atau kategori pada data

Margin : Jarak paling dekat *hyperplane* dengan titik pada

tiap kelas

Kernel : Fungsi yang digunakan untuk mentransformasi data sehingga dapat memetakan data ke ruang dimensi yang lebih tinggi.

Data *Training* : Kumpulan data yang digunakan untuk melatih model klasifikasi

Data *Testing* : Kumpulan data yang digunakan untuk mengevaluasi kinerja model yang telah dilatih.

K-Mers : Kombinasi pasangan nukleotida yang Menyusun DNA manusia

BAB II

KAJIAN TEORI

2.1 Teori Pendukung

2.1.1 Lagrange Multiplier

Metode *Lagrange Multiplier* merupakan salah satu teknik optimasi yang digunakan untuk mencari titik lokal maksimum dan minimum dari sebuah fungsi yang memiliki persamaan batas. Kelebihan dari metode ini adalah tidak diperlukannya parameterisasi eksplisit dari kendala sehingga membuatnya dapat memecahkan masalah optimasi yang kompleks dengan lebih efisien (Tran et al., 2023). Metode ini juga dapat mengatasi permasalahan pada fungsi yang memiliki ketaksamaan batas melalui teknik *Lagrange Multiplier* tingkat lanjut yang dikenal sebagai kondisi Karush-Kuhn-Tucker. Fungsi Lagrange diformulasikan oleh persamaan:

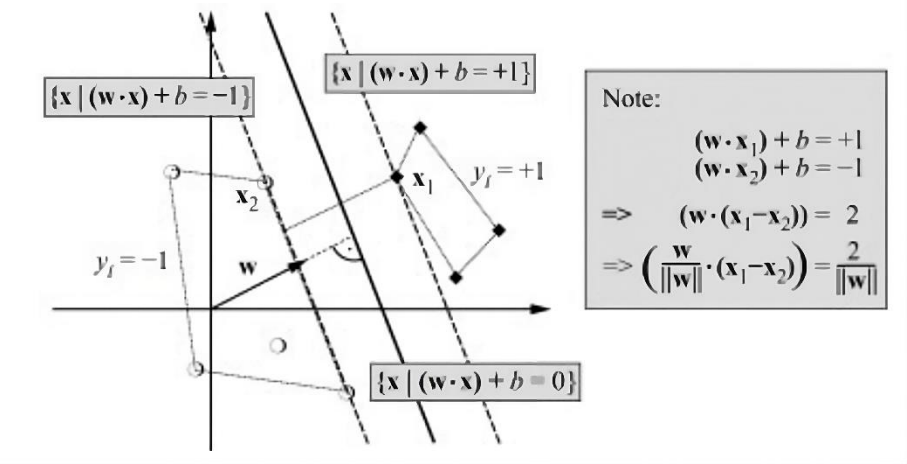
$$L(x, \lambda) = f(x) - \lambda\big(g(x)\big) = 0 \qquad (2.1)$$

dimana $f(x)$ merupakan fungsi tujuan yang memiliki batas $g(x)$ dan λ merupakan variabel *Lagrange*.

2.1.2 *nu-Support Vector Machine*

nu-Support Vector Machine atau *nu*-SVM merupakan salah satu variasi dari algoritma SVM. Pada dasarnya prinsip kerja dari algoritma *nu*-SVM dan SVM adalah sama yakni dengan memisahkan data ke dalam dua Kategori menggunakan bantuan bidang pemisah (*hyperplane*). *nu*-SVM memiliki parameter tambahan yaitu *nu (v)* yang berfungsi sebagai pengontrol *margin error* dan jumlah dari

support vector, sedangkan pada SVM biasa jumlah *support vector* tidak ditentukan (Setiawan et al., 2023). *nu*-SVM dan SVM merupakan salah satu algoritma *machine learning* yang banyak digunakan dalam pemodelan *machine learning* karena kemampuannya dalam mempelajari data dengan dimensi tinggi. Algoritma ini bekerja dengan mengkategorikan data dengan memisahkan data tersebut ke dalam dua kelompok data atau kelas yang disimbolkan sebagai kelas positif {+1} dan negatif {-1} menggunakan bantuan bidang pemisah berdimensi tinggi (*hyperplane*).



Gambar 2.1 Hyperplane Pada Algoritma SVM

Hyperplane paling optimal didapatkan saat *margin* antara data dan bidang pemisah bernilai maksimal. *Margin* merupakan jarak dari titik terdekat data (*support vector*) dengan bidang *hyperplane*-nya (Huang et al., 2018). Pada klasifikasi multi-kelas, baik SVM ataupun *nu*-SVM membuat $\frac{n(n-1)}{2}$ model ,dengan n adalah jumlah kelas, yang pada tiap modelnya dilakukan pembentukan *hyperplane* dengan membagi kelompok data menjadi 2 kelas, dengan satu kelas menjadi kelas positif {+1} sedangkan beberapa kelas lainnya dianggap sebagai

kelas negatif {-1}. Proses pembuatan model ini akan dilakukan berulang-kali dengan pemilihan kedua kelas yang berbeda hingga terbentuk beberapa bidang *hyperplane* yang dapat memisahkan data menjadi beberapa kelas.

Fungsi pemisah (*hyperplane*) pada SVM umumnya didefiniskan sebagai persamaan:

$$\mathrm{w} \cdot x_i + b = 0 \tag{2.2}$$

$$\mathrm{w} \in \mathbb{R}^N; x_i \in \mathbb{R}^N; b \in R;$$

$i = 1,2,$ ...,jumlah observasi; dan $N =$ dimensi data

dimana permasalahan utama (*primal*) tersebut $\mathrm{w} \cdot x_i$ merupakan perkalian *dot product* dan w adalah nilai vektor bobot, x_i adalah vektor data ke-i, serta b adalah nilai bias. Misalkan terdapat $\mathrm{w} = \begin{bmatrix} 0{,}5 \\ 1 \\ 0{,}5 \end{bmatrix}$, $x_1 = \begin{bmatrix} x_{1.1} \\ x_{1.2} \\ x_{1.3} \end{bmatrix}$ dan $b = 0{,}5$ maka

$$\rightarrow \begin{bmatrix} 0{,}5 \\ 1 \\ 0{,}5 \end{bmatrix} \cdot \begin{bmatrix} x_{1.1} \\ x_{1.2} \\ x_{1.3} \end{bmatrix} + 0{,}5 = 0$$

$$\rightarrow (0{,}5.\, x_{1.1} + x_{1.2} + 0{,}5.\, x_{1.3}) + 0{,}5 = 0$$

dimana bentuk diatas merupakan fungsi *hyperplane* yang memisahkan kedua data.

Data x tersebut akan dapat dikelompokan ke dalam kelas positif {+1}, yaitu kelas pada saat data berada pada bagian kanan *hyperplane* dan direpresentasikan dengan $w \cdot x_i + b \geq 1$ atau kelas negatif {-1}, yaitu kelas pada saat data berada pada bagian kiri *hyperplane* dan direpresentasikan dengan $\mathrm{w} \cdot x_i + b \leq (-1)$ dimana x_i merupakan observasi ke-i dan $i = 1,2, \ldots, n$. Kemudian untuk menentukan *hyperplane* optimal akan ditentukan nilai margin (d) atau jarak maksimal antara 2 kelas data tersebut, sehingga didapatkan:

$$\max d(\mathrm{w}, x_{i,j}, b) = \frac{\left|(\mathrm{w} \cdot x_i + b + 1) - (\mathrm{w} \cdot x_j + b - 1)\right|}{\|\mathrm{w}\|}$$

$$\max d(\mathrm{w}, x_{i,j}, b) = \frac{\left|(\mathrm{w} \cdot x_i + b + 1) - (\mathrm{w} \cdot x_j + b - 1)\right|}{\|\mathrm{w}\|}$$

$$\max d(\mathrm{w}, x_{i,j}, b) = \frac{2}{\|\mathrm{w}\|} \qquad (2.3)$$

Untuk melakukan pembuktian dari persamaan jarak maksimal diatas kita misalkan nilai $\mathrm{w} = [0{,}5; 1; 0{,}5]^T$, $x_1 = [4; -1; -1]$, $x_2 = [-4; 1; 1]$ dan $b = 0{,}5$ sehingga didapatkan:

$$\max d(w, x_{i,j}, b) = \frac{\left|\left(\begin{bmatrix}0{,}5\\1\\0{,}5\end{bmatrix} \cdot \begin{bmatrix}-1\\2\\-1\end{bmatrix} + 0{,}5 + 1\right) - \left(\begin{bmatrix}0{,}5\\1\\0{,}5\end{bmatrix} \cdot \begin{bmatrix}1\\-3\\1\end{bmatrix} + 0{,}5 - 1\right)\right|}{\sqrt{0{,}5^2 + 1^2 + 0{,}5^2}}$$

$$\max d(w, x_{i,j}, b) = \frac{|(2{,}5) - (-3{,}5)|}{1{,}22} = \frac{6}{1{,}22} = 4{,}92$$

Jadi, jarak maksimal dari dua nilai x diatas adalah 4,91.

Pencarian margin maksimal dalam menentukan hyperplane optimal di SVM dapat ditemukan dengan mencari vektor w berdimensi terkecil dalam sebuah masalah optimasi konveks untuk menemukan hyperplane optimal. Oleh karena itu, margin maksimal dalam algoritma *nu*-SVM dapat diformulasikan sebagai masalah pemrograman kuadrat (*Quadratic Programming*) sebagai berikut:

$$\min d(\mathrm{w}, x_{i,j}, b) = \frac{1}{2}\|\mathrm{w}\| \qquad (2.4)$$

dengan $y_i(\mathrm{w} \cdot x_i + b) \geq 1, i = 1{,}2, \ldots, M$, dimana x_i adalah data ke-i dan y_i adalah label atau kelas dari data x_i.

Pada penerapan SVM dalam dunia nyata, jumlah data yang sangat besar seringkali membuat hyperplane tidak dapat memisahkan kategori data secara sempurna (Chen et al., 2005). Oleh karenanya dikenalkan variabel *slack* (ξ) untuk

memberikan fleksibilitas lebih pada batasan (constraints) sehingga didapatkan Persamaan (2.5) yang diformulasikan sebagai:

$$y_i(\mathrm{w} \cdot x_i + b) \geq 1 - \xi_i, \qquad \xi_i \geq 0; i = 1,2, \dots, n \qquad (2.5)$$

Untuk memberikan gambaran terhadap peran variabel *slack* ξ_i akan diberikan ilustrasi dengan nilai sebelumnya dimana x_1 adalah kelas positif, x_2 adalah kelas negatif, sehingga didapatkan:

$$y_1(\mathrm{w} \cdot x_1 + b) \geq 1 - \xi_1$$

$$\rightarrow 1\left(\begin{bmatrix}0{,}5\\1\\0{,}5\end{bmatrix} \cdot \begin{bmatrix}-1\\2\\-1\end{bmatrix} + 0{,}5\right) = 1 - \xi_1$$

$$\rightarrow 1{,}5 - 1 = \xi_1$$

$$\rightarrow \xi_1 = 0{,}5$$

kemudian untuk kelas negatif

$$y_2(\mathrm{w} \cdot x_2 + b) \geq 1 - \xi_2$$

$$\rightarrow -1\left(\begin{bmatrix}0{,}5\\1\\0{,}5\end{bmatrix} \cdot \begin{bmatrix}1\\-3\\1\end{bmatrix} + 0{,}5\right) = 1 - \xi_2$$

$$\rightarrow -(-1{,}5) - 1 = \xi_2$$

$$\rightarrow \xi_2 = 0{,}5$$

Jadi, didapatkan fungsi *hyperplane* $y_{i^+}(\mathrm{w} \cdot x_{i^+} + b) \geq 0{,}5$ untuk kelas positif dan $y_{i^-}(\mathrm{w} \cdot x_{i^-} + b) \geq 0{,}5$ untuk kelas negatif.

Kemudian dari persamaan (2.5) diatas didapatkan *soft margin* yang dapat memisahkan data dengan baik yang akan menghasilkan masalah optimasi konveks dan diformulasikan ke dalam persamaan SVM atau biasa disebut C-SVM sebagai berikut:

$$\min \tau(\mathrm{w}, \xi_i) = \frac{1}{2}\|\mathrm{w}\|^2 + C\sum_{i=1}^{n}\xi_i \qquad (2.6)$$

$$C > 0; i = 1,2, \dots, n$$

Untuk memberikan gambaran terhadap fungsi C-SVM diatas, maka akan diberikan ilustrasi menggunakan nilai sebelumnya dengan *C* = 1 sehingga didapatkan:

$$\min \tau(\mathrm{w}, \xi_i) = \frac{1}{2}(\mathrm{w}^T \cdot \mathrm{w}) + C(\xi_1 + \xi_2)$$

$$\min \tau(\mathrm{w}, \xi_i) = \frac{1}{2}(0{,}5^2 + 1^2 + 0{,}5^2) + 1(0{,}5 + 0{,}5)$$

$$\min \tau(\mathrm{w}, \xi_i) = 1{,}75$$

Jadi, didapatkan nilai $\min \tau(\mathrm{w}, \xi_i) = 1{,}75$. Nilai tersebut nantinya akan dapat digunakan untuk memecahkan permasalahan dualitas pada algoritma C-SVM sehingga hyperplane optimal dapat ditemukan.

Berbeda dengan SVM biasa (C-SVM), pada *nu*-SVM parameter *C* digantikan oleh parameter *nu* (v) dengan $v \in [0,1]$ yang mana merepresentasikan batas minimal dari *support vector* dan batas maksimal dari *margin error*. Persamaan pada *nu*-SVM diformulasikan sebagai:

$$\min \tau(\mathrm{w}, \xi, \rho) = \frac{1}{2}\| \mathrm{w} \|^2 - v\rho + \frac{1}{2}\sum_{i=1}^{M}\xi_i \qquad (2.7)$$

dengan constraint

$$y_i\big((\mathrm{w} \cdot x_i) + b\big) \geq \rho - \xi_i \qquad (2.8)$$

$$\rho \geq 0; \xi_i \geq 0; i = 1,2, \dots, M$$

Variabel ρ merupakan variabel tambahan yang perlu optimasi. Secara umum, pada bidang pemisah antarkelas {+1} dan {-1} variabel $\xi_i = 0$, sehingga didapatkan:

$$y_i\big((\mathrm{w} \cdot x_i) + b\big) \geq \rho = 1$$

$$\rightarrow \rho = 1$$

$$\rightarrow \rho\left(\frac{2}{\|\mathrm{w}\|}\right) = \frac{2}{\|\mathrm{w}\|}$$

Yang berarti 2 kelas yang ada dipisahkan oleh margin maksimal $\frac{2\rho}{\|\mathrm{w}\|}$.

Melihat bahwa persamaan (2.7) merupakan fungsi tujuan dari *nu*-SVM yang berbentuk kuadrat dan constrainnya pada persamaan (2.8) yang bersifat linier dalam parameter *w* dan *b* ini disebut sebagai masalah optimasi konveks. Hal ini dapat diselesaikan dengan menggunakan metode Lagrange Multiplier (α) sehingga didapatkan bentuk permasalahan ganda (*dual*) dari algoritma *nu*-SVM yaitu:

$$\begin{aligned} L(\mathrm{w}, v, \xi_i, b, \rho, \alpha_i, \beta_i, \delta) & \\ &= \left[\frac{1}{2} \| \mathrm{w} \|^2 - v\rho + \frac{1}{m}\sum_{i=1}^{m} \xi_i\right] \\ &- \sum_{i=1}^{m} \alpha_i\big[\big(y_i\big((\mathrm{w} \cdot x) + b\big) - \rho + \xi_i\big) \\ &- (-\delta\rho + \beta_i\xi_i)\big] \end{aligned} \tag{2.9}$$

dengan $\alpha_i, \beta_i, \delta \geq 0$, dimana $\alpha_i\beta_i$, dan δ merupakan variabel dualitas *Lagrange*, serta $m = 1,2, \ldots$, jumlah data observasi.

Berikutnya, untuk menemukan solusi dari bentuk Lagrange tersebut akan dilakukan optimasi pada persamaan (2.8) dengan melakukan turunan parsial

terhadap parameter $w, b, \xi_i, \text{dan } \rho$, untuk memenuhi kondisi Karush-Kuhn-Tucker (KKT) (Chen et al., 2005) yaitu:

$$\frac{\partial}{\partial w_i} L(\mathrm{w}, v, \xi_i, b, \rho, \alpha_i, \beta_i, \delta) = 0, \;\; i = 1,2, \dots, n \tag{3.0}$$

$$\frac{\partial}{\partial b} L(\mathrm{w}, v, \xi_i, b, \rho, \alpha_i, \beta_i, \delta) = 0, \;\; i = 1,2, \dots, n \tag{3.1}$$

$$\frac{\partial}{\partial \xi_i} L(\mathrm{w}, v, \xi_i, b, \rho, \alpha_i, \beta_i, \delta) = 0, \;\; i = 1,2, \dots, n \tag{3.2}$$

$$\frac{\partial}{\partial \rho} L(\mathrm{w}, v, \xi_i, b, \rho, \alpha_i, \beta_i, \delta) = 0, \;\; i = 1,2, \dots, n \tag{3.3}$$

$$\sum_{i=1}^{n} \alpha_i f(x_i) = 0, \;\; i = 1,2, \dots, n \tag{3.4}$$

$$\alpha_i \geq 0, \;\; i = 1,2, \dots, n \tag{3.5}$$

Kemudian, dari kondisi KKT diatas maka didapatkan kondisi untuk algoritma *nu*-SVM sebagai berikut:

$$\mathrm{w} = \sum_{i=1}^{m} \alpha_i y_i x_i \tag{3.6}$$

$$\alpha_i + \beta_i = \frac{1}{m} \tag{3.7}$$

$$\sum_{i=1}^{m} \alpha_i y_i = 0 \tag{3.8}$$

$$\sum_{i=1}^{m} \alpha_i - \delta = v \tag{3.9}$$

Kemudian, substitusi kondisi yang didapat pada persamaan (3.6) dan (3.9) pada persamaan Lagrange (L) sehingga didapatkan solusi optimal dengan bentuk permasalahan optimasi dualitas untuk *nu*-SVM yang diformulasikan dengan:

$$\max L(\alpha) = -\frac{1}{2}\sum_{i=1}^{m} \alpha_i \alpha_j y_i y_j (x_i x_j) \quad (4.0)$$

dengan syarat

$$0 \leq \alpha_i \leq \frac{1}{m} \quad (4.1)$$

$$\sum_{i=1}^{m} \alpha_i y_i = 0 \quad (4.2)$$

$$\sum_{i=1}^{m} \alpha_i \geq v \quad (4.3)$$

Fungsi *Lagrange Multiplier* diatas dapat juga ditulis dalam bentuk persamaan untuk memudahkan pemecahan masalah sehingga bentuk akhirnya akan menjadi:

$$\min L(\alpha) = \frac{1}{2}\sum_{i=1}^{m} \alpha_i \alpha_j y_i y_j (x_i x_j) \quad (4.4)$$

dengan syarat

$$0 \leq \alpha_i \leq \frac{1}{m} \quad (4.5)$$

$$\sum_{i=1}^{m} \alpha_i y_i = 0 \quad (4.6)$$

$$\sum_{i=1}^{m} \alpha_i = v \quad (4.7)$$

Selanjutnya, akan dicari nilai *hyperplane* optimal dengan mencari nilai bobot vektor w dan bias *b* berdasarkan persamaan (3.6) yang telah ditemukan sebelumnya, sehinga didapatkan fungsi keputusan untuk menentukan kelas dari data yang diformulasikan dengan:

$$f(x_i) = sign(\mathrm{w} \cdot x + b)$$

$$f(x_i) = sign\left(\sum_{i=1}^{m} \alpha_i y_i (x_i x_j) + b\right) \tag{4.8}$$

$$f(x_i) = \begin{cases} +1, & \sum_{i=1}^{m} \alpha_i y_i (x_i x_j) + b \geq 0 \\ -1, & \sum_{i=1}^{m} \alpha_i y_i (x_i x_j) + b < 0 \end{cases}$$

dimana bias b adalah

$$b = \frac{1}{m} \sum_{i=1}^{m} \left(\frac{1}{y_i} - \sum_{j=1}^{m} \alpha_i y_i (x_i x_j)\right) \tag{4.9}$$

2.1.3 Kernel

Pada proses pembuatan model klasifikasi menggunakan algoritma *nu*-SVM, biasanya terdapat data yang tidak bisa dipisahkan secara linier (non-linier). Hal ini terjadi karena kompleksitas data yang berasal dari dunia nyata sehingga membuat data tersebut perlu ditransformasikan terlebih dahulu ke dalam ruang dengan dimensi lebih tinggi sehingga data tersebut dapat disesuaikan ke dalam *hyperplane* linier. Disinilah akan digunakan fungsi untuk mentransformasikan data tersebut yang dinamakan dengan fungsi *kernel*.

Fungsi kernel merupakan fungsi yang digunakan untuk mentransformasikan input data x_i ke dalam dimensi yang lebih tinggi (*feature space*) dari ruang Hilbert $\mathbb{H}$ yang nantinya membuat data dapat dipisahkan secara linier (Chen et al., 2005). Ruang Hilbert pada $\mathbb{R}$ adalah sebuah ruang vektor yang terbentuk dari hasil kali skalar (*dot-product*), yang memenuhi bentuk norm yaitu

$$\|\mathrm{u}\| = \sqrt{u_1^2 + u_2^2 + \cdots + u_n^2} = \sqrt{\mathrm{u} \cdot \mathrm{u}} \qquad (5.0)$$

Sedemikian sehingga setiap barisan Cauchy mempunyai limit di H (Chipot, 2009). Misalkan terdapat suatu vektor $\mathrm{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \\ 1 \\ 3 \end{bmatrix}$, maka norm dari vektor tersebut adalah

$$\|\mathrm{u}\| = \sqrt{1^2 + (-2)^2 + 1^2 + 3^2} = \sqrt{15}$$

Bentuk umum ruang $\mathbb{R}^n$ (Ruang Hilbert) sendiri yang dilengkapi dengan produk skalar diformulasikan dengan:

$$(x, y) = x \cdot y = \sum_{i=1}^{i=n} x_i y_i \quad ; \forall x_i, y_i \in \mathbb{R}^N \qquad (5.1)$$

$i = 1,2, \ldots, n$ (dimensi data)

Fungsi kernel sendiri diformulasikan dengan persamaan:

$$K: \mathbb{R}^N \rightarrow K(x_i, x_j)$$

$$K(x_i, x_j) = (\phi(x_i).\phi(x_j)) \qquad (5.2)$$

dimana $\varphi: R^N \rightarrow \mathbb{H}: x_i = \varphi(x_i)$ adalah fungsi pemetaan data ke dalam ruang hilbert.

Fungsi $K(x_i, x_j)$ tersebut memungkinkan kita untuk melakukan perkalian *dot product* pada *feature space* $\mathbb{H}$ berdimensi *N* dari data x_i dan x_j. Adanya kernel tersebut membuat data kelas yang ada dapat dipisahkan secara linier menggunakan algoritma *nu*-SVM yang direpresentasikan dengan persamaan:

$$\min L(\alpha) = \frac{1}{2} \sum_{i,j=1}^{j=n} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \qquad (5.3)$$

Sehingga diperoleh pula fungsi keputusan dengan formulasi matematika yaitu:

$$f(x_i) = sign\left(\sum_{i,j=1}^{j=n} \alpha_i y_i K(x_i, x_j) + b\right) \quad (5.4)$$

dengan nilai bias b

$$b = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{y_i} - \sum_{j=1}^{m} \alpha_i y_i K(x_i x_j)\right) \quad (5.5)$$

Pada input data yang didapatkan terdapat kemungkinan bahwa data tersebut terdapat dalam ruang fitur dengan jumlah fitur yang tak terhingga, sehingga diperlukan fungsi *kernel* yang tepat untuk memisahkan agar kelas data dapat tepat dipisahkan secara linear. Terdapat beberapa jenis kernel yang sering digunakan diantaranya:

1. Kernel Linear

$$K(x_i, x_j) = x_i^T \cdot x_j \quad (5.6)$$

2. Kernel Polinomial

$$K(x_i, x_j) = (\gamma \cdot x_i^T \cdot x_j + r)^p, \qquad \gamma > 0 \quad (5.7)$$

3. Kernel *Radial Basis Fuction (RBF)*

$$K(x_i, x_j) = \exp\left(-\gamma |x_i - x_j|^2\right), \qquad \gamma > 0 \quad (5.8)$$

4. Kernel *Sigmoid*

$$K(x_i, x_j) = \tanh(\gamma \cdot x_i^T \cdot x_j + r) \quad (5.9)$$

2.1.4 *Confusion Matrix*

Confusion Matrix merupakan metrik yang digunakan untuk menganalisis performa atau kinerja dari seberapa baik suatu model klasifikasi dalam mengenali kelas dari suatu data dengan benar (Ramli et al., 2022). Performa model klasifikasi akan diukur menggunakan 4 komponen utama dalam *confusion matrix* yaitu:

Tabel 2.1 Tabel *Confusion Matrix*

		Data Prediksi	
		Benar	Salah
Data Aktual	Benar	*True Positive* (TP)	*False Positive* (FP)
	Salah	*False Negative* (FN)	*True Negative* (TN)

Dimana tiap komponen dari *confusion matrix* diatas adalah:

1. *True Positive* (TP), yaitu data dari kelas positif yang diklasifikasikan dengan benar sebagai kelas positif.
2. *False Positive* (FP), yaitu data dari kelas negatif yang diklasifikasikan dengan salah sebagai kelas positif.
3. *True Negative* (TN), yaitu data dari kelas negatif yang diklasifikasikan dengan benar sebagai kelas negatif.
4. *False Negative* (FN), yaitu data dari kelas positif yang diklasifikasikan dengan salah sebagai kelas negatif.

Melalui komponen utama dari *confusion matrix* diatas, maka performa dan kinerja dari model klasifikasi yang dibuat dapat diketahui dengan menghitung metriks performa berupa Presisi untuk mengukur seberapa baik kemampuan model untuk memprediksi data kelas positif, *Recall* (sensitivitas) yang

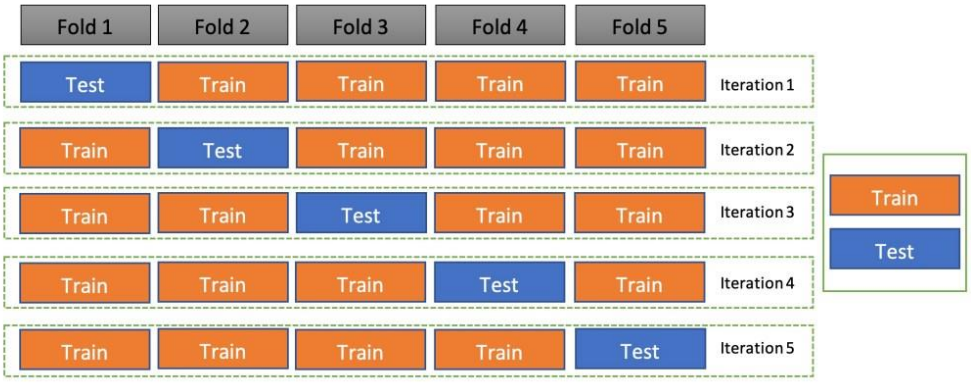
menggambarkan seberapa baik model dalam mengidentifikasi kelas positif dengan benar, Akurasi dan *F1-Score* sebagai metriks yang menggambarkan keseimbangan antara presisi dan *recall* (sensitivitas) model klasifikasi. rumus dari keempat metriks tersebut dapat dilihat pada tabel dibawah:

Tabel 2.2 Rumus Metriks Performa Pada *Confusion Matrix*

Presisi	Akurasi
$\frac{TP}{TP+FP}\text{x}100\%$	$\frac{TP+TN}{TP+TN+FP+FN}\text{x}100\%$
Recall	*F1-Score*
$\frac{TP}{TP+FN}\text{x}100\%$	$2\text{x}\frac{Presisi\ \text{x}\ Recall}{Presisi+\ Recall}\text{x}100\%$

2.1.5 *Stratified K-Fold Cross Validation*

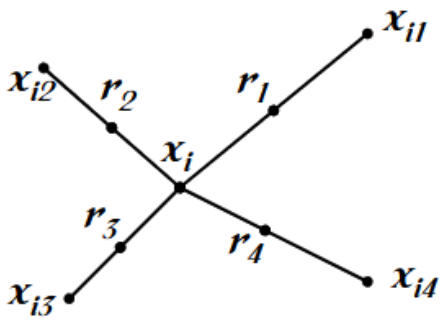
Stratified K-Fold Cross Validation merupakan salah satu metode modifikasi dari metode *K-Fold Cross Validation* biasa. Metode ini bekerja dengan membagi dataset menjadi *k* bagian yang dengan jumlah distribusi yang sama ditiap bagian subsetnya. Hal ini memungkinkan proses validasi dan evaluasi lebih terkontrol dengan jumlah sampel tiap kelas sama sehingga akurasi yang dihasilkanpun akan konsisten (T R et al., 2023). Bagian dari *k-1* data nantinya akan digunakan sebagai data latih (*training data*) sedangkan sisanya sebagai data uji (*testing data*). Proses pembagian tersebut akan dilakukan secara acak dan bergantian hingga semua kombinasi yang mungkin dari *k-1* data digunakan sebagai model pelatihan. Kemudian, performa dari model akan dievaluasi dengan rata-rata hasil akurasi dari tiap *k* pelatihan sehingga didapatkan nilai akurasi final dari model klasifikasi yang dibuat (T R et al., 2023).



Gambar 2.2 Ilustrasi *Stratified K-Fold Cross Validation*

2.1.6 SMOTE

Synthetic Minority Oversampling Technique atau lebih dikenal dengan SMOTE merupakan salah satu teknik *oversampling* data yang cukup efektif dalam menyeimbangkan dataset. *Oversampling* dilakukan pada saat terdapat ketidakseimbangan pada dataset dimana terdapat selisih jumlah data yang cukup besar dari kelas mayoritas dan minoritas. Pada dasarnya, SMOTE bekerja dengan membuat interpolasi pada beberapa data minoritas yang berada pada *neighborhood* yang telah ditentukan. Untuk alasan ini, prosedur ini dikatakan difokuskan pada "ruang fitur" daripada "ruang data", dengan kata lain, algoritme ini didasarkan pada nilai fitur dan hubungannya, alih-alih mempertimbangkan titik data secara keseluruhan (Fernández et al., 2018).



Gambar 2.3 Ilustrasi Pembuatan Data Sintesis dengan Metode SMOTE

Sebuah kelas minoritas x_i dipilih sebagai basis untuk membuat titik data sintesis baru. Kemudian, berdasarkan metrik jarak tertentu, beberapa lingkungan (*neighbors*) terdekat dari kelas yang sama (x_{i1} hingga x_{i4}) akan dipilih dari data latih. Setelah itu, interpolasi acak akan diterapkan sehingga didapatkan data sintetik baru yakni r_1 hingga r_4.

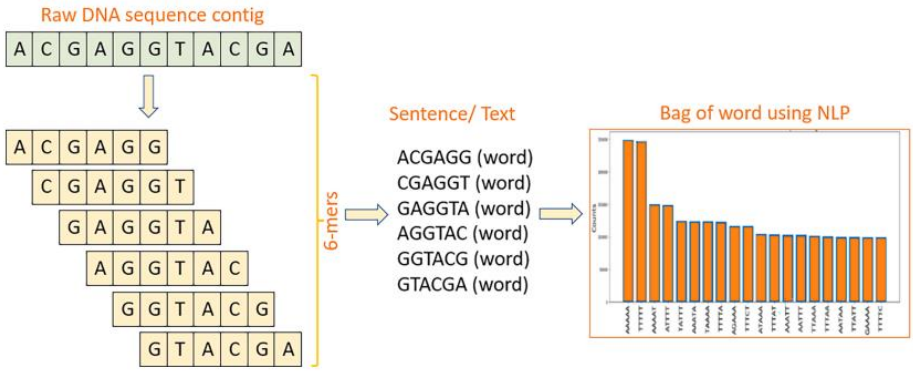
2.1.7 *CountVectorizer*

CountVectorizer merupakan salah satu metode untuk mentransformasikan data ke dalam bentuk vektor numerik. Metode ini biasanya sering digunakan pada analisis data dalam bentuk sentimen atau kalimat. Metode CountVectorizer bekerja dengan menghitung frekuensi kemunculan dari suatu kata (*term*) pada suatu sentiment atau biasa dikenal dengan dokumen (Assery et al., 2019). Nantinya, nilai dari tiap frekuensi dari *term* yang muncul akan digunakan sebagai fitur atau karakteristik yang dimiliki oleh sentiment.

2.1.8 *K-Mers Encoding*

K-mers encoding merupakan teknik untuk mengkodekan struktur dari suatu sekuens biologis menjadi *substring* dengan panjang k yang mewakili suatu

pasangan atau struktur dalam sekuens biologis (Solis-Reyes et al., 2018). *K-mers*, yang terbentuk dari pasangan *nukleotida*, banyak digunakan dalam konteks komputasi untuk mengambil kemungkinan pola yang terdapat ditingkat genomik dan analisis sekuens gen (Alshayeji et al., 2023). Pada DNA, sekuens ACGT memiliki empat monomer (A, C, G, dan T), tiga 2-*mer* (AC, CG, GT), dua 3-*mer* (ACG dan CGT), dan satu 4-*mer* (ACGT). Biasanya, *k-mer* mengacu pada semua barisan yang memiliki panjang k. Meskipun secara teori tampak mudah, menghitung k-*mer* dalam kumpulan data sekuens yang cukup besar dapat membuat memori komputer kelebihan kapasitas memori karena besarnya substring yang dihasilkan.



Gambar 2.4 Ilustrasi K-mers Encoding pada Sekuens DNA

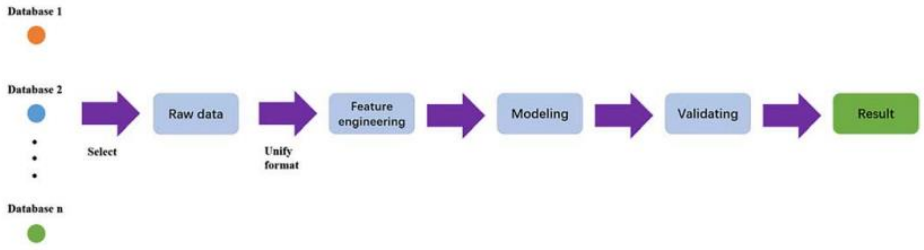
2.1.9 *Machine Learning*

Machine Learning merupakan bidang studi yang memanfaatkan komputer untuk mempelajari suatu hubungan dan makna dari kumpulan data dan observasi (Janiesch et al., 2021). *Machine learning* menggunakan berbagai macam algoritma yang berbeda untuk memecahkan masalah pada data (Mahesh, 2020). Berdasarkan jenisnya, *machine learning* dapat dibagi menjadi tiga kategori utama,

yakni *supervised learning*, *unsupervised learning* dan *reinforcement learning* (Janiesch et al., 2021).

Supervised learning adalah jenis *machine learning* yang mempelajari fungsi yang memetakan input fitur dari suatu data ke dalam suatu output berdasarkan contoh yang diberikan. *Unsupervised learning* merupakan kebalikan dari *supervised learning* dikarenakan jenis ini tidak memiliki contoh dari *input* fitur dan *output* sehingga mesin harus belajar secara mandiri. Sedangkan jenis *reinforcement learning* adalah jenis *machine learning* yang merupakan optimalisasi gabungan dari kedua metode sebelumnya, dengan menerapan prinsip pemberian *reward* pada hasil pembelajarannya (Mahesh, 2020).

Machine learning memiliki beberapa langkah penting sebelum membentuk suatu model permasalahan.



Gambar 2.5 Alur Kerja Machine Learning

Langkah-langkah tersebut dilakukan agar input data dapat diproses dengan mesin komputer sehingga didapat model dengan tepat untuk menyelesaikan permasalahan yang ada. Beberapa Langkah tersebut, diantaranya:

1. *Data Selection*

   Data selection process atau proses pemilihan data dalam *machine learning* diambil dengan mempertimbangkan tipe, kualitas, dan format dari data. Pemilihan data dengan integritas tinggi mampu mencegah hilangnya

informasi dalam proses pembuatan model baik itu karena adanya data yang hilang maupun rusak (Wei et al., 2019).

2. *Feature Engineering*

   Feature engineering dapat dilakukan oleh pembuat model untuk mendapatkan informasi dari sebuah data yang sesuai dengan target penelitian untuk mendapatkan keakuratan model yang maksimal. Pada dasarnya, *feature engineering* memilih beberapa fitur penting dari data untuk memperoleh karakteristik yang sesuai dan berpengaruh besar terhadap prediksi target (Wei et al., 2019).

3. *Model Construction*

   Data yang telah dibersihkan akan dibagi menjadi dua bagian, yakni *training* dan *testing data*. Selanjutnya, proses pembuatan model akan dimulai dengan melatih data untuk proses pelatihan (*training data*) dengan menggunakan algoritma yang telah dipilih untuk pelatihan. Model yang telah terbentuk tersebut kemudian akan digunakan untuk membuat prediksi terhadap data uji (*testing data*) untuk mengukur keakuratan model (Wei et al., 2019).

4. *Validating Model*

   Validating model dalam *machine learning* merupakan proses evaluasi dari model yang telah dibuat sebelumnya dengan menggunakan data yang tidak terlihat dan berbeda dari *training data*. Metode yang digunakan pada penelitian ini adalah *K-Fold Cross Validation* (Wei et al., 2019).

2.1.10 *Natural Language Processing*

Natural Language Processing atau biasa dikenal dengan NLP adalah gabungan antara komputasi sains, kecerdasan buatan dan ilmu linguistik, yang ditujukan untuk membuat mesin mampu memahami bahasa manusia (Khurana et al., 2023). NLP merupakan salah satu permasalahan yang seringkali dipecahkan dengan pendekatan *machine learning*. Pada dasarnya, NLP bekerja dengan memecah data *string* atau kalimat menjadi komponen-komponen dasar yang lebih mudah untuk diolah sehingga dapat dilakukan proses identifikasi hubungan antar komponen pada bahasa tersebut (Alshayeji et al., 2023). NLP dapat diimplementasikan pada berbagai permasalahan, seperti ekstraksi informasi penting dari suatu teks (identitas, hubungan, dll.), terjemah antar bahasa, ringkasan suatu teks, *chatbot* berbasis AI, analisis sentimen pengguna sosial media, penelitian dibidang kesehatan yang melibatkan data *genomics*, dan sebagainya.

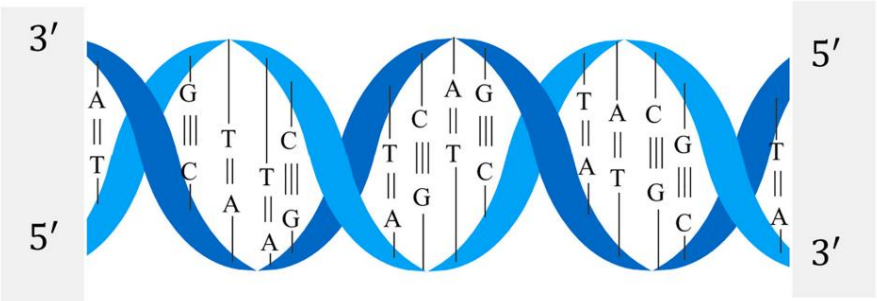
Menurut Li (2021), *natural language* dan sekuens biologis memiliki karakteristik yang sama sebagai suatu bahasa (Li et al., 2021). Sekuens biologis, seperti DNA dan RNA, menyimpan semua informasi yang menentukan struktur dan fungsinya, serta kalimat yang berisi semua informasi yang mendefinisikan sintaks dan semantiknya (X. Zhang et al., 2020). Hubungan antara sekuens, struktur, dan fungsi dalam sekuens tersebut memiliki kesamaan dengan hubungan antara kalimat, sintaks, dan semantik dalam linguistik dan bahasa manusia. Sehingga, implementasi teknik NLP pada sekuens biologis memiliki peran penting dalam mengidentifikasi dan memahami makna didalamnya.

2.1.11 Klasifikasi

Klasifikasi merupakan salah satu metode pendekatan pada *machine learning* dan *data mining* yang sering digunakan pada pengelompokan data. Klasifikasi juga merupakan salah satu bagian dari *supervised learning* dimana mesin dapat belajar dari data yang telah memiliki label. Klasifikasi bergantung pada jumlah klasifikasi atau kategori dari target yang digunakan pada data, sehingga berbagai algoritma berbeda dapat diterapkan pada metode ini (Yang, 2018). Pada klasifikasi biner sederhana biasanya digunakan algoritma yang memiliki performa cukup baik seperti regresi logistik, *decision tree* dan SVM. Namun, pada kasus kehidupan nyata, dimana jumlah target klasifikasi biasanya lebih dari dua memerlukan algoritma lanjutan yang lebih rumit dikarenakan banyaknya kategori yang membutuhkan komputasi yang lebih kompleks.

2.1.12 Sekuens DNA

DNA atau Asam Deoksiribonukleat adalah molekul yang membawa informasi genetik suatu organisme dan dapat diwariskan pada generasi berikutnya. Fragmen yang mengandung informasi genetik tersebut akan diperlukan dalam proses pembentukan komponen lainnya dalam sel seperti protein dan RNA. Sekuens DNA terhubung oleh empat jenis basa nukleotida, yaitu adenin (A), sitosin (C), guanin (G), dan timin (T) (Akkaya & Kalkan, 2021). Urutan basa nukleotida tersebut berkontribusi pada keragaman molekul dan karakteristik dari DNA. DNA memiliki struktur dalam heliks ganda seperti ditunjukkan pada ilustrasi dibawah.



Gambar 2.6 Rantai Nukleotida Pada Sekuens DNA

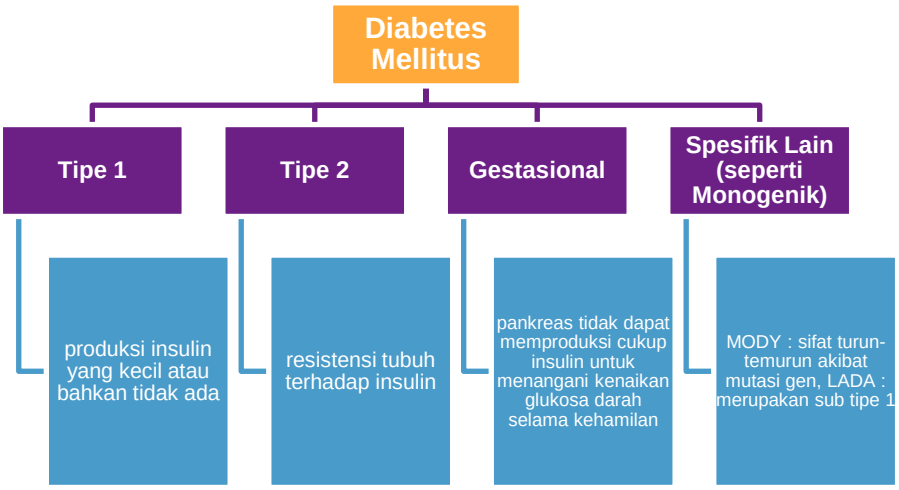
DNA berperan untuk mengkode dan membentuk molekul protein dari sel tubuh. Protein merupakan makromolekul biologis yang terdiri dari satu atau lebih rantai panjang asam amino. Setiap asam amino dalam sintesis protein diatur oleh rangkaian tiga nukleotida yang disebut sebagai “kode” atau “kodon” (dalam RNA). Salah satu protein dalam tubuh manusia yang diproduksi oleh DNA tersebut adalah insulin yang berperan dalam proses pengolahan dan penyerapan glukosa dalam tubuh. Mutasi dalam urutan gen insulin dapat menyebabkan ketidakseimbangan dalam produksi hormon insulin yang akhirnya dapat mengakibatkan diabetes melitus.

2.1.13 Diabetes Mellitus

Diabetes adalah suatu penyakit dari sistem endokrin yang didiagnosa dengan tingkat glukosa dalam darah yang relatif cukup tinggi (Cole & Florez, 2020). Menurut Cho (2018) diabetes tergolong ke dalam salah satu penyakit paling umum dengan tingkat pertumbuhan tinggi di seluruh dunia, dan ditaksir menjangkit hingga 693 juta jiwa pada 2045 (Cho et al., 2018). Tingkat glukosa yang tinggi tersebut dapat disebabkan oleh kurangnya tingkat insulin, resistensi terhadap insulin atau keduanya (Cole & Florez, 2020). Penderita diabetes memiliki resiko yang tinggi terhadap terpicunya penyakit lain seperti penyakit

jantung, penyakit arteri perifera dan serebrovaskular, obesitas, katarak, disfungsi ereksi, dan penyakit hati (Kazi & Blonde, 2001).

Menurut World Health Organization (WHO) diabetes melitus dibedakan menjadi empat jenis, yakni diabetes melitus tipe 1 (T1DM), diabetes melitus tipe 2 (T2DM), diabetes melitus gestasional, serta diabetes monogenik (E. El-Attar et al., 2022). Diabetes mellitus tipe 1 disebabkan oleh proses penghancuran autoimun dari sel B-pankreas tanpa produksi hormon insulin. Diabetes mellitus tipe 2, yang mana merupakan tipe paling dominan, disebabkan karena kurangnya produksi insulin atau berkurangnya sensitivitas dari reseptor insulin yang mengakibatkan terhalangnya penyerapan glukosa ke dalam sel. Diabetes mellitus gestasional terjadi hanya pada saat seorang wanita sedang mengandung dengan rata-rata kejadian pada 5-15% wanita diseluruh dunia. Sedangkan diabetes monogenik, yang seringkali salah diagnosa sebagai diabetes mellitus tipe 1 atau 2, disebabkan oleh mutasi pada gen tunggal atau klaster dari beberapa gen.



Gambar 2.7 Tipe-tipe Diabetes mellitus

Diabetes memiliki beberapa gejala seperti meningkatnya rasa haus, poliuria (keadaan sering buang air kecil), kaburnya penglihatan, dan penurunan berat badan yang cukup signifikan. Namun, pada diabetes mellitus tipe 2 (T2DM), gejala yang ditimbulkan cukup ringan hingga mungkin tidak ada, dikarenakan peningkatan kadar gula darah yang berlangsung lambat. Akibatnya, tanpa adanya pengujian seperti biokimia, hiperglikemia yang cukup untuk menyebabkan perubahan patologis dan fungsional sistem kerja tubuh dapat hadir untuk jangka waktu yang lama sebelum diagnosis sehingga dapat mengakibatkan komplikasi. Terdapat empat jenis tes diagnosa untuk diabetes yang direkomendasikan, yakni pengukuran glukosa darah pada orang yang berpuasa, tes toleransi glukosa oral, tes Glycosylated Hemoglobin, dan tes glukosa darah acak saat seseorang memiliki tanda dan gejala diabetes (Kazi & Blonde, 2001).

2.2 Kajian Integrasi Topik dengan Al-Quran/Hadits

DNA berperan penting sebagai molekul biologi yang menentukan sifat dan karakteristik suatu organisme hidup. Melalui DNA, seseorang dapat mewariskan sifat, karakteristik dan bahkan penyakit yang ada dalam tubuh seseorang tersebut pada keturunannya. Pewarisan sifat melalui DNA juga telah disebutkan dalam firman Allah SWT melalui Al-Qur'an yakni dalam Surat Al-Qiyamah ayat 37-39 (Kemenag, 2024) yang berbunyi:

اَلَمْ يَكُ نُطْفَةً مِّنْ مَّنِيٍّ يُّمْنٰىۙ ٣٧ثُمَّ كَانَ عَلَقَةً فَخَلَقَ فَسَوّٰىۙ ٣٨فَجَعَلَ مِنْهُ الزَّوْجَيْنِ الذَّكَرَ وَالْاُنْثٰىۗ ٣٩

Artinya: "*Bukankah dia dahulu setetes mani yang ditumpahkan (ke dalam rahim)? Kemudian, (mani itu) menjadi sesuatu yang melekat, lalu Dia menciptakan dan menyempurnakannya. Lalu, Dia menjadikan darinya sepasang laki-laki dan perempuan.*" (Q.S. Al-Qiyamah ayat 37-39)

Beberapa ulama telah menafsirkan ayat di atas sebagai salah satu penjelasan mengenai proses penciptaan manusia melalui reproduksi biologis manusia. M. Quraish Shihab (2002) dalam buku tafsirnya menjelaskan bahwa proses awal terciptanya manusia dimulai saat air mani memasuki rahim dan bertemu dengan indung telur. Kemudian terbentuklah *'alaqah* yaitu sesuatu yang terus membelah dirinya dan menempel pada dinding rahim. Sehingga, pada saat itulah Allah SWT dengan segala kuasanya menciptakan manusia dan menyempurnakan proses penciptaan tersebut hingga terciptalah bentuk manusia yang sempurna dan *mutfah* (tercipta dengan sepasang laki-laki dan perempuan) (Shihab, 2002).

Sebagaimana dijelaskan pada ayat tersebut, Allah telah mengingatkan manusia tentang asal muasal penciptaan manusia. Manusia diciptakan dari setetes air mani yang kemudian ditumpahkan (ke dalam rahim). Lalu, dari mani itulah akan terbentuk menjadi segumpal darah, kemudian Allah menciptakan, dan menyempurnakannya hingga menjadi janin manusia yang hidup. Allah telah menjelaskan seluruh proses tentang diciptakannya manusia melalui proses kehamilan dari hubungan sepasang suami dan istri. Secara tidak langsung, ayat ini menyiratkan tentang pewarisan sifat manusia. Saat bayi diciptakan di dalam Rahim seseorang, sifat-sifat yang dimiliki orang tuanya akan diwariskan kepada anak tersebut. Dalam genetika, proses pewarisan sifat-sifat ini berhubungan dengan adanya transfer materi genetik (DNA) dari orang tua kepada anak lewat proses reproduksi dan kehamilan. Ilmu genetika modern menyatakan jika setiap manusia akan mewarisi genetik dari kedua orang tuanya meskipun salah satunya resesif (kurang nampak). Genetik ini mengandung informasi yang mengatur berbagai sifat dan karakteristik fisik, biologis, dan kondisi terhadap suatu penyakit tertentu.

2.3 Kajian Topik Dengan Teori Pendukung

Penelitian ini memiliki tujuan guna mengidentifikasi dan mengklasifikasikan sekuens DNA dari manusia sehat dan yang menderita penyakit diabetes mellitus tipe 1 maupun tipe 2. Sekuens DNA sebagai penyusun dasar antar organisme pada dasarnya memiliki karakteristik dan struktur yang berbeda. Pada manusia, sekuens DNA terdiri dari empat macam nukleotida yang berpasangan dan dapat memiliki panjang hingga 32.000 nukleotida. Sekuens DNA tersebut membentuk beragam struktur yang kompleks dan variasi karakteristik pada tiap individu pemilik sekuens DNA. Oleh karena itu, dengan mengidentifikasi dan mengklasifikasikan sekuens DNA manusia untuk menemukan pola karakteristik dari penyakit diabetes mellitus dapat menjadi metode identifikasi penyakit dan pengobatan yang akurat namun memerlukan usaha yang cukup sulit.

Penggunaan *machine learning* dalam membantu proses klasifikasi sekuens DNA mampu mengidentifikasi pola dari data sekuens DNA yang sangat kompleks. Melalui klasifikasi, DNA akan dipisahkan dan dikategorikan berdasarkan kesamaan karakteristik tiap kelasnya. Pada penelitian ini, dikembangkan model *machine learning* menggunakan algoritma *nu*-SVM yang mampu mengklasifikasi data dengan jumlah feature dan dimensi yang tinggi. Sekuens DNA akan diidentifikasi dan diklasifikasikan berdasarkan probabilitas munculnya kodon pada sekuensnya. Data sekuens DNA yang digunakan pada penelitian ini bersumber dari situs web NCBI sebagai penyedia data sekuens biologi suatu organisme untuk suatu penelitian ilmiah. Data sekuens DNA yang diambil terdiri dari tiga kelas data, yakni sekuens DNA penderita diabetes mellitus tipe 1 dan tipe 2, serta sekuens DNA non-diabetes mellitus.

Tahapan pertama yang dilakukan dalam proses klasifikasi ialah penyiapan dan *preprocessing* data sekuens DNA. Tahap penyiapan dilakukan dengan memilih data yang sesuai dengan kebutuhan, sedangkan pada tahap *preprocessing* dilakukan pembersihan data sekuens DNA dari *whitespace*, serta penghapusan sekuens yang rusak dan tidak dapat diidentifikasi. Tahap kedua ialah tahap *feature engineering* atau ekstraksi fitur dari sekuens DNA dengan membagi sekuens yang panjang tersebut menjadi substring yang terdiri dari tiga nukleotida atau dapat disebut sebagai kodon. Kodon tersebut digabungkan kembali dengan metode *n-bag-of-words* yang menggabungkan beberapa kata menjadi satu fitur sehingga tiap fiturnya akan terdiri dari empat urutan kodon DNA. Kemudian dilakukan proses transformasi data menjadi data numerik agar data hasil ekstraksi fitur dapat dimasukkan ke dalam model *machine learning*.

Setelah tahap transformasi data, dilakukan *oversampling* terhadap data untuk menyeimbangkan data dari kelas kurang dominan dengan membuat data sintesis menggunakan metode SMOTE. Kemudian dilakukan klasifikasi sekuens DNA manusia yang menderita penyakit diabetes mellitus dan tidak menggunakan algoritma *nu*-SVM. *nu*-SVM merupakan modifikasi dari algoritma SVM biasa untuk mengoptimalkan pembentukan *hyperplane*. Keunggulan dari metode ini adalah dapat menghasilkan model klasifikasi cukup baik dan akurat pada data DNA dengan dimensi tinggi. Melalui algoritma ini, penulis mengharapkan penelitian ini dapat dihasilkan hasil klasifikasi dengan tingkat akurasi yang tinggi.

BAB III

METODE PENELITIAN

3.1 Jenis Penelitian

Jenis penelitian yang dilakukan oleh peneliti adalah penelitian kuantitatif dengan melibatkan perhitungan rumus-rumus pada data sekuens DNA. Hasil perhitungan nantinya akan disimpulkan berdasarkan nilai angka yang diperoleh dengan asumsi bahwa hasil angka tersebut dapat mempermudah pernyataan, perbandingan, dan penjelasan terkait dengan akurasi. Selanjutnya, peneliti melakukan penelitian berdasarkan tahapan-tahapan yang terstruktur dan sistematis. Data yang digunakan pada penelitian ini adalah data sekuens DNA yang akan diubah dari data string menjadi data angka. Selanjutnya data akan diproses hingga akhirnya diklasifikasikan menggunakan algoritma *nu*-SVM.

3.2 Data dan Sumber Data

Data yang digunakan pada penelitian ini dikumpulkan dari website NCBI (*https://www.ncbi.nlm.nih.gov/*) dengan memilih sekuens DNA manusia sehat yang tidak memiliki diabetes mellitus, DNA penderita diabetes mellitus tipe 1 dan tipe 2. Data sekuens DNA yang didapatkan memiliki format file berbentuk .fasta sehingga perlu dilakukan *preprocessing* terhadap data yang ada dengan membersihkan data dari duplikasi, missing value, dan DNA yang tidak diketahui, serta mengkonversi menjadi file .csv. setelah melalui proses *preprocessing*, diperoleh data final yang siap untuk diolah lebih lanjut dengan sekuens DNA manusia sehat berjumlah 1000 data, sekuens DNA manusia penderita diabetes

mellitus tipe 1 berjumlah 213 data, sedangkan pada penderita tipe 2 diabetes mellitus berjumlah 1296 data.

Adapun website NCBI yang digunakan sebagai sumber pencarian data merupakan penyedia data penelitian yang dirancang untuk mendukung penelitian terkait sekuens gen, DNA dan protein dengan ukuran data yang cukup besar. NCBI juga merupakan portal komunitas untuk data sekuens dari GenBank, RefSeq, dan repositori NCBI lainnya. *Database* dari NCBI menyediakan sekuens gen, DNA dan protein dengan kualitas tinggi yang diperoleh dari berbagai penelitian dan seringkali digunakan sebagai data kolaborasi dari *International Nucleotide Sequence Database Collaboration* (INSDC).

3.3 Perancangan Sistem

Sistem yang akan dirancang yakni sistem yang bisa digunakan untuk mengidentifikasi berbagai macam sekuens atau barisan DNA manusia yang termasuk penderita diabetes mellitus atau tidak. Sistem ini akan mengidentifikasi kelompok dari sekuens DNA mentah manusia yang akan dimasukkan ke dalam model berdasarkan perhitungan secara matematis menggunakan algoritma *nu*-SVM. Nantinya sistem akan memproses dan mengolah sekuens DNA tersebut hingga akhirnya dapat mengidentifikasi kelompok sekuens DNA yang telah dimasukkan.

3.4 Teknik Analisis Data

3.4.1 Proses Preprocessing Data

Pada proses *Preprocessing* terdapat beberapa tahapan sebagai berikut:

1. Data sekuens DNA mentah yang masih berbentuk file .fasta akan dikonversikan menjadi file berformat .csv dan kemudian akan dibersihkan dari duplikasi, *missing value*, dan data DNA yang tidak diketahui untuk menjaga kualitas data.
2. Data sekuens DNA siap pakai yang telah diproses tadi akan berbentuk string dan akan dibagi menjadi *substring* dengan prinsip 3-*mers encoding* sehingga data DNA akan menjadi teks dalam bentuk *list* dari kumpulan 3-*mers string*.
3. Hasil *encoding* kemudian akan dijadikan menjadi bentuk kalimat utuh yang tersusun dari kata-kata dasar dan data target akan dipisahkan dari data sekuens.
4. Setelah didapatkan data DNA yang berbentuk kalimat dari Kumpulan *substring* 3-*mers*, data sekuens tersebut akan ditransformasikan menggunakan teknik *n-bag-of-words* sehingga akan dihasilkan fitur berupa n-kombinasi dari *k-mers* yang kemudian diubah menjadi bentuk numerik menggunakan *CountVectorizer*, serta pada bagian data target akan ditransformasikan ke dalam bentuk numerik pula menggunakan fungsi mapping.
5. Kedua data yang telah diubah ke dalam bentuk numerik tersebut, yakni data fitur dan target, akan dibagi menjadi dua menjadi data *training* dan *testing*.

3.4.2 Proses *Training Data*

1. Proses *training data* (pelatihan) untuk membuat model klasifikasi akan menggunakan data *training* sebanyak 80% dari seluruh data.

2. Proses pelatihan akan menggunakan parameter *nu* $v = 0{,}05$.
3. Pertama, model klasifikasi akan dibuat dengan meminimumkan fungsi *Lagrange Multiplier* sehingga didapatkan nilai variabel *dual* α.
4. Berikutnya, dicari nilai dari kernel linear $K(x_i, x_j)$ dengan melakukan perkalian *dot product* pada data x_i dan x_j.
5. Kemudian, dari hasil nilai α akan dicari juga nilai dari vektor bobot w dan bias b dengan mensubstitusikan α tersebut ke persamaan bobot w dan bias b.
6. Nilai α, kernel $K(x_i, x_j)$, bobot w dan bias b tersebut akan dimasukkan ke dalam fungsi keputusan $f(x_i)$ sehingga didapatkan persamaan *hyperplane* optimal dari model klasifikasi algoritma *nu*-SVM.

3.4.3 Proses *Testing Data*

1. Sekuens DNA akan digunakan sebagai data *testing* sebanyak 20% dari seluruh data.
2. Pada model klasifikasi *nu*-SVM yang telah dilatih akan dilakukan pengujian dengan menggunakan data uji yang belum pernah dilakukan pelatihan untuk mengetahui apakah model tersebut dapat mengenali data dengan baik.

3.4.4 Proses Evaluasi

Proses evaluasi dilakukan untuk menganalisis dan mengetahui seberapa baik performa model klasifikasi *nu*-SVM dalam mengidentifikasi dan mengklasifikasi sekuens DNA pada penyakit diabetes mellitus. Proses evaluasi akan menggunakan metode *Stratified K-Fold Cross Validation* dengan membagi

data menjadi 5 bagian atau subset yang memiliki distribusi kelas yang sama. Kemudian akan dilakukan proses *training* dan *testing* pada 5 subset data tersebut dengan kombinasi subset yang berbeda sehingga didapatkan akurasi dari masing-masing pelatihan. Proses ini akan diulang hingga 3 kali sehingga proses pembagian data dan pelatihan dapat dilakukan dengan merata. Setelah didapatkan semua akurasi maka akan dihitung rata-rata dari nilai akurasi yang didapatkan oleh *Stratified K-Fold Cross Validation* sebagai hasil akurasi akhir dari model klasifikasi sekuens DNA manusia penderita diabetes mellitus.

BAB IV

PEMBAHASAN

4.1 Preprocessing Data

Pada tahapan *preprocessing* ini, data sekuens DNA manusia penderita dan bukan penderita diabetes mellitus akan dibersihkan dan ditransformasikan sesuai format yang diperlukan sehingga data tersebut dapat dilatih dengan menggunakan algoritma *nu-Support Vector Machine*. Sebelumnya, dataset sekuens DNA manusia dalam format *.fasta* ke dalam file *jupyter notebook* dan kemudian melakukan konversi dataset menjadi format *.csv* dengan bantuan *library Biopython*. Berikutnya, akan dilakukan tahapan *preprocessing data* sekuens DNA lebih lanjut yang akan diuraikan sebagai berikut:

4.1.1 Cleaning Data

Proses pembersihan dataset DNA manusia untuk analisis diabetes mellitus melibatkan beberapa tahapan penting. Pertama, data DNA dari penderita diabetes mellitus tipe 1, tipe 2, dan bukan penderita diabetes akan digabungkan menjadi satu dataset untuk memudahkan analisis. Kedua, akan dilakukan penghapusan *whitespace* pada data DNA yang memiliki *whitespace* berlebih dan juga penghapusan salah satu sekuens pada data sekuens yang terindentifikasi duplikat. Menurut Akkaya & Kalkan (2021) pada penelitian sebelumnya, data sekuens DNA manusia terdiri dari 4 nukleutida yaitu A, C, G, dan T. Oleh karena itu, untuk menjaga kualitas dari data DNA manusia yang akan digunakan dalam pembuatan model, maka data akan dibersihkan dari sekuens DNA berkualitas buruk, tidak lengkap atau tidak diketahui. Berikutnya, dilakukan filter pada panjang sekuens

sehingga hanya data dengan panjang sekuens yang lebih dari 1000 dan kurang dari 10.000 nukleotida yang akan digunakan. Berikut adalah ilustrasi dari proses *cleaning data* pada data sekuens DNA manusia dengan menggunakan data asli pada index ke 1,2, 502 dan 1301:

Tabel 4.1 Sampel Data Sekuens DNA manusia

Index	Sekuens	Panjang	Label
1	AGCCCTCCAGGACAGGCT GC AGAAGAG CAT	644	DMT1
2	AGCCCTCCAGGACAGGCT GC AGAAGAG CAT	525	DMT1
502	AGAAGATATCTCACATGA AACTGGCTGCCTGCCAC	6272	DMT2
1301	CCNNCNGGACGGCGCCTC CTTCACCTACAAGCGAT	26770	DMT2

Sampel data sekuens DNA manusia diatas merupakan data yang diambil dari dataset sekeuns DNA pada index 1, 2, 502, dan 1301. Data sekuens DNA pada index 1 dan 2 merupakan data sekuens DNA penderita diabetes mellitus tipe 1 dengan masing-masing sekuens memiliki panjang 644 dan 525 nukleotida. Sementara data sekuens DNA pada index 502 dan 1301 merupakan data sekuens DNA penderita diabetes mellitus tipe 2 dengan masing-masing sekuens memiliki panjang 6272 dan 26770 nukleotida.

Tabel 4.2 Proses *Cleaning Data* Sekuens DNA

Proses Penghapusan W*hitespace*	
Sebelum	Sesudah
AGCCCTCCAGGACAGGCTGC AGAAGAG CAT	AGCCCTCCAGGACAGGCTGCAG AAGAGCAT
AGCCCTCCAGGACAGGCTGC AGAAGAG CAT	AGCCCTCCAGGACAGGCTGCAG AAGAGCAT
AGAAGATATCTCACATGAAACT GGCTGCCTGCCAC	AGAAGATATCTCACATGAAACT GGCTGCCTGCCAC
CCNNCNGGACGGCGCCTCCTTC ACCTACAAGCGAT	CCNNCNGGACGGCGCCTCCTTC ACCTACAAGCGAT
Proses Penghapusan Data Duplikat	
Sebelum	Sesudah
AGCCCTCCAGGACAGGCTGCAG AAGAGCAT	AGCCCTCCAGGACAGGCTGCAG AAGAGCAT
AGCCCTCCAGGACAGGCTGCAG AAGAGCAT	(dihapus)
Proses Penghapusan Data Sekuens Buruk	
Sebelum	Sesudah
CC**NN**C**N**GGACGGCGCCTCCTTC ACCTACAAGCGAT	(dihapus)
Proses Filter Sekuens dengan Panjang 1000 – 10.000 *Nukleotida*	
Sebelum	Sesudah
AGCCCTCCAGGACAGGCTGCAG AAGAGCAT	dihapus

AGAAGATATCTCACATGAAACT GGCTGCCTGCCAC	AGAAGATATCTCACATGAAACT GGCTGCCTGCCAC

Tabel 4.3 Frekuensi Data Sekuens DNA Manusia

Sebelum *Cleaning Data*		
DMT1	DMT2	NON-DM
213	1296	1000
Setelah *Cleaning Data*		
DMT1	DMT2	NON-DM
145	443	989

4.1.2 K-mers encoding

K-mers encoding akan membagi struktur DNA menjadi pasangan *substring* dengan dengan panjang *k-nukleotida* untuk mengidentifikasi pola dan karakteristik dari sekuens DNA yang ada. Pola atau karaktersitik inilah nantinya yang akan digunakan sebagai fitur dari data sekuens DNA. Pada penelitian ini, akan digunakan nilai k = 3 sehingga data DNA akan diekstraksi menjadi pasangan 3-*nukleotida*. Proses k-mers encoding akan dilakukan dengan bantuan *function k-mers encoding* yang tela dibuat menggunakan python. Berikut merupakan tampilan data DNA yang telah diekstraksi polanya menggunakan *k-mers encoding*:

Tabel 4.4 Hasil *K-Mers Encoding*

Sekuens DNA	K-Mers
AGAAGATATC TCACATGAAA	AGA, GAA, AAG, AGA, GAT, ATA, TAT, ATC, TCT, CTC, TCA, CAC, ACA, CAT,ATG, TGA, GAA,

CTGGCTGCCT GCCAC	AAA, AAC, ACT, CTG, TGG, GGC, GCT, CTG, TGC, GCC, CCT, CTG, TGC, GCC, CCA, CAC

4.1.3 Transformasi *CountVectorizer*

Algoritma dari *machine learning* memerlukan masukan data dalam bentuk numerik agar pelatihan dan pembuatan model dapat dilakukan. Data sekuens DNA yang telah diekstraksi pola dan karakteristiknya menggunakan k-mers encoding akan ditransformasikan menjadi data numerik menggunakan metode *CountVectorizer*. Namun, sebelumnya akan dilakukan proses ekstraksi ulang pada pola k-mers yang telah ada dengan menggunakan metode *n-bag-of-words*. Metode *n-bag-of-words* membuat kombinasi baru dari 3-mers yang telah dibuat. Dengan menggunakan n = 4, maka metode *4-bag-of-words* akan mengkombinasikan 4 pasangan 3-mers menjadi pola dan fitur-fitur baru. Hal ini akan dilakukan berulang kali hingga semua kombinasi untuk 4 pasang 3-mers tercapai.

Tabel 4.5 Ekstraksi Fitur 4-*Bag-of-Words*

K-Mers
AGA, GAA, AAG, AGA, GAT, ATA, TAT, ATC, TCT, CTC, TCA, CAC, ACA, CAT, ATG, TGA, GAA, AAA, AAC, ACT, CTG, TGG, GGC, GCT, CTG, TGC, GCC, CCT, CTG, TGC, GCC, CCA, CAC
Hasil 4-Bag-of-Words
AGA AGA AGA AGA, AGA AGA AGA GAA, AGA AGA AGA AAG, AGA AGA AGA AGA, AGA AGA AGA GAT, AGA AGA AGA ATA, AGA AGA AGA TAT, AGA AGA AGA ATC, AGA AGA AGA TCT, AGA AGA AGA CTC, AGA AGA AGA TCA, …, CAC CAC CAC CAC

Selanjutnya, akan dilakukan proses transformasi dengan *CountVectorizer*. *CountVectorizer* akan menghasilkan nilai fitur dari data sekuens DNA yang ada berdasarkan frekuensi dari munculnya pasangan 3-mers dari *4-bag-of-words* pada data DNA. Penerapan *CountVectorizer* ini akan menggunakan bantuan *software* python dengan memanfaatkan *library* sklearn.

Berikut merupakan contoh hasil dari penerapan *CountVectorizer* pada dataset sekuens DNA manusia yang diambil sebagai sampel perhitungan. Data yang diambil merupakan data pada index 642, 803 dan 951 dengan data index 642 dan 803 merupakan DNA penderita diabetes mellitus tipe 1, serta data index 951 merupakan DNA bukan penderita diabetes mellitus.

Tabel 4.6 Hasil Nilai *CountVectorizer*

Fitur	Frekuensi		
	x_1	x_2	x_3
Fitur 1	11	4	8
Fitur 2	3	5	4
Fitur 3	4	0	4
Fitur 4	1	5	8
Fitur 5	2	3	4
Fitur 6	1	2	2
Fitur 7	0	0	0
Fitur 8	4	2	3
Fitur 9	2	1	8
Fitur 10	5	1	4
Fitur 11	1	2	1
y_i	1	1	-1

Tabel 4.6 diatas merupakan hasil dari proses transformasi data dengan *CountVectorizer* yang dilakukan pada 3 data sekuens DNA. Data pertama (x_1) yang memiliki kelas positif $(y_1 = 1)$ menghasilkan nilai frekuensi kemunculan fitur dengan fitur 1 sebanyak 11 , fitur 2 sebanyak 3, fitur 3 sebanyak 4, fitur 4 sebanyak 1, fitur 5 sebanyak 2, fitur 6 sebanyak 1, fitur 7 sebanyak 0, fitur 8 sebanyak 4, fitur 9 sebanyak 2, fitur 10 sebanyak 5 dan fitur 11 sebanyak 1.

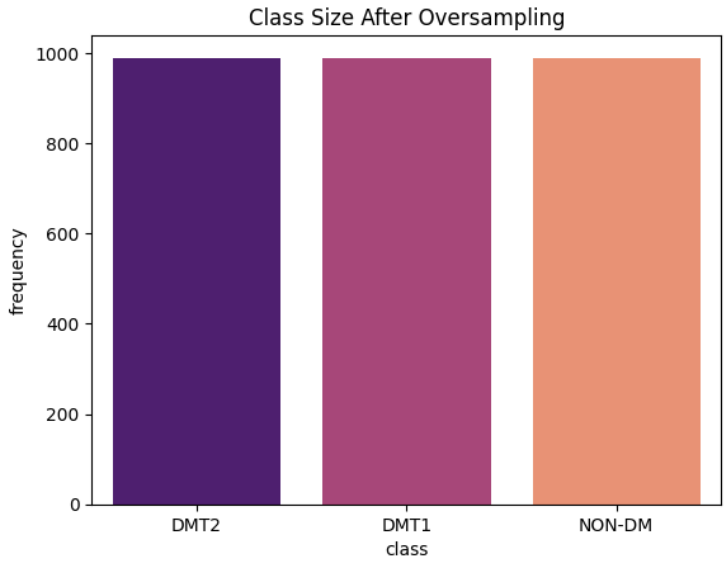
Kemudian untuk data ketiga(x_3) yang memiliki kelas negatif $(y_3 = -1)$ menghasilkan nilai frekuensi kemunculan fitur dengan fitur 1 sebanyak 8 , fitur 2 sebanyak 4, fitur 3 sebanyak 4, fitur 4 sebanyak 8, fitur 5 sebanyak 4, fitur 6 sebanyak 2, fitur 7 sebanyak 0, fitur 8 sebanyak 3, fitur 9 sebanyak 8, fitur 10 sebanyak 4 dan fitur 11 sebanyak 1.

4.1.4 Oversampling Data

Berdasarkan Tabel 4.3 diatas, terlihat bahwa data sekuens DNA pada penelitian kali ini memiliki jumlah yang tidak seimbang. Hal ini dapat mengakibatkan kurangnya performa dari model klasifikasi. Oleh karena itu, untuk membuat data tersebut seimbang akan digunakan teknik *oversampling data*, yaitu dengan menyamakan jumlah sampel data dari nilai kelas minoritas dengan memperbanyak data tersebut hingga jumlahnya setara dengan nilai mayoritas.

SMOTE bekerja dengan cara membuat data sintesis baru berdasarkan kemiripan karakteristik data dari kelas tersebut. Pertama, SMOTE memilih beberapa sampel data dari kelas minoritas dalam dataset. Kemudian, pada sampel minoritas yang telah dipilih, SMOTE akan mencari tetangga terdekatnya yang ada

dalam ruang fitur. Setelah menemukan tetangga terdekat untuk suatu sampel minoritas, SMOTE akan membuat sampel sintetis baru di antara sampel minoritas dan tetangga terdekatnya. Sampel sintetis ini berada pada garis yang menghubungkan dua sampel tersebut di dalam ruang fitur. Akhirnya, sampel sintetis baru ini akan ditambahkan ke dataset sebagai data baru dari kelas yang minoritas sehingga akan meningkatkan jumlah data dari kelas tersebut. Berikut grafik jumlah sebaran data pada tiap kelas setelah dilakukan *oversampling* dengan SMOTE:



Gambar 4.1 Hasil *Oversampling* Menggunakan SMOTE

Tabel 4.7 Frekuensi Data Sekuens DNA Setelah *Oversampling*

Sebelum *Oversampling*			
DMT1	DMT2	NON-DM	**TOTAL**
145	443	989	1577
Sebelum *Oversampling*			
DMT1	DMT2	NON-DM	**TOTAL**
989	989	989	2967

Setelah melakukan *oversampling* terlihat pada gambar 4.1 bahwa jumlah frekuensi semua kelas telah seimbang. Pada tabel 4.6 menunjukkan bahwa pada kelas minoritas telah dibuat data sintesis baru berdasarkan metode SMOTE sehingga jumlah ketiga kelas menjadi seimbang yakni berjumlah 989 data tiap kelasnya, dengan jumlah total data menjadi 2967 data.

4.2 Pembuatan Model Klasifikasi *nu*-SVM

Proses klasifikasi dimulai dengan membagi data menjadi dua, yaitu data *training* dan data *testing* dengan rasio 80:20. Rasio tersebut dipilih berdasarkan hasil uji coba yang telah dilakukan, di mana terbukti bahwa model yang dilatih dengan rasio tersebut mencapai tingkat akurasi yang tinggi. Pada data *training* yang terdiri dari beberapa variabel dan kelas target dijadikan sebagai data sampel untuk membentuk model klasifikasi dengan algoritma *nu*-SVM. Sedangkan data *testing* digunakan untuk mengevaluasi ketepatan model klasifikasi berdasarkan *confusion matrix* yang terbentuk.

4.2.1 Penyelesaian Model Klasifikasi Algoritma *nu*-SVM

Pada penyelesaian model klasifikasi dengan menggunakan algoritma *nu*-SVM akan dilakukan dengan melakukan simulasi pada 2 data observasi. Kemudian penyelesaian dapat dilakukan dengan mencari nilai minimum dari Persamaan (2.7) dengan *constraint* atau batasan yang terdapat pada Persamaan (2.8). Persamaan tersebut memiliki bentuk masalah optimasi konveks sehingga cukup sulit untuk dipecahkan. Solusi dari permasalahan tersebut adalah dengan

mengubah persamaan tersebut ke dalam bentuk fungsi *Lagrange Multiplier* sehingga dapat lebih mudah untuk diselesaikan.

$$L(x) = f(x) - \lambda(g(x) - c)$$

Maka dengan mensubstitusikan Persamaan (2.9) beserta fungsi batas (*constraints*) pada Persamaan (3.0) pada fungsi *Lagrange* diatas didapatkan:

$$L(\mathrm{w}, v, b, \xi_i, \rho, \alpha_i, \beta_i, \delta)$$

$$= \left[\frac{1}{2} \parallel \mathrm{w} \parallel^2 - v\rho + \frac{1}{3}\sum_{i=1}^{3} \xi_i\right]$$

$$- \sum_{i=1}^{3} \left(\alpha_i \left[y_i\left((\mathrm{w} \cdot x_i) + b\right) - \rho + \xi_i\right] - \delta\rho + \beta_i\xi_i\right)$$

dengan w, b, ξ_i, ρ merupakan variable primal; w merupakan vektor bobot, $v \in [0,1]$ merupakan parameter *nu*; $\rho, \xi_i \geq 0$; serta $\alpha_i, \beta_i, \delta \geq 0$ merupakan variabel dual dari *Lagrange*. Kemudian fungsi tersebut akan disederhanakan sehingga didapat:

$$L(\mathrm{w}, b, \xi, \rho, \alpha, \beta, \delta)$$

$$= \left[\frac{1}{2} \parallel \mathrm{w} \parallel^2 - v\rho + \frac{1}{3}\sum_{i=1}^{3} \xi_i\right]$$

$$- \sum_{i=1}^{3} (\alpha_i \left[y_i\left((\mathrm{w} \cdot x_i) + b\right)\right] - \alpha_i\rho + \alpha_i\xi_i - \delta\rho + \beta_i\xi_i)$$

$$= \left[\frac{1}{2} \parallel \mathrm{w} \parallel^2 - v\rho + \frac{1}{3}(\xi_1 + \xi_2 + \xi_3)\right]$$

$$- \left((\alpha_1 \left[y_1\left((\mathrm{w} \cdot x_1) + b\right)\right] - \alpha_1\rho + \alpha_1\xi_1 - \delta\rho + \beta_1\xi_1\right)$$

$$+ (\alpha_2 \left[y_2\left((\mathrm{w} \cdot x_2) + b\right)\right] - \alpha_2\rho + \alpha_2\xi_2 - \delta\rho + \beta_2\xi_2)$$

$$+ (\alpha_3 \left[y_3\left((\mathrm{w} \cdot x_3) + b\right)\right] - \alpha_3\rho + \alpha_3\xi_3 - \delta\rho + \beta_3\xi_3))$$

dari persamaan diatas akan dicari turunan pertama terhadap variable primal yang ada dalam fungsi *Lagrange* tersebut sehingga didapat *constraints* sebagai berikut:

$$\frac{\partial L}{\partial w_i} = 0$$

$$\mathrm{w} = (\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3) \quad (1)$$

$$\frac{\partial L}{\partial b} = 0$$

$$\alpha_1 y_1 + \alpha_2 y_2 + \alpha_3 y_3 = 0 \quad (2)$$

$$\frac{\partial L}{\partial \xi_i} = 0$$

$$\frac{\partial L}{\partial \xi_1} = \frac{1}{3} - (\alpha_1 + \beta_1) = 0$$

$$(\alpha_1 + \beta_1) = \frac{1}{3}$$

$$\frac{\partial L}{\partial \xi_2} = \frac{1}{3} - (\alpha_2 + \beta_2) = 0$$

$$(\alpha_2 + \beta_2) = \frac{1}{3}$$

$$\frac{\partial L}{\partial \xi_3} = \frac{1}{3} - (\alpha_3 + \beta_3) = 0$$

$$(\alpha_3 + \beta_3) = \frac{1}{3}$$

sehingga didapatkan

$$\alpha_i + \beta_i = \frac{1}{3} \quad (3)$$

$$\frac{\partial L}{\partial \rho} = 0$$

$$(\alpha_1 + \alpha_2 + \alpha_3 - 3\delta) = v \qquad (4)$$

Hasil dari constraints (1) hingga (4) tersebut akan dimasukkan ke dalam fungsi *Lagrange* sebelumnya sehingga didapatkan:

$$L(\alpha) = \left[\frac{1}{2} \parallel \mathrm{w} \parallel^2 - v\rho + \frac{1}{3}\sum_{i=1}^{3} \xi_i\right]$$

$$-\sum_{i=1}^{3}\left((\alpha_i\left[y_i\left((\mathrm{w}\cdot x_i)+b\right)\right] - \alpha_i\rho + \alpha_i\xi_i) - \delta\rho + \beta_i\xi_i\right)$$

$$= \left[\frac{1}{2} \parallel \mathrm{w} \parallel^2 - v\rho + \frac{1}{3}(\xi_1 + \xi_2 + \xi_3)\right]$$

$$-\left((\alpha_1\left[y_1\left((\mathrm{w}\cdot x_1)+b\right)\right] - \alpha_1\rho + \alpha_1\xi_1 - \delta\rho + \beta_1\xi_1\right)$$

$$+ (\alpha_2\left[y_2\left((\mathrm{w}\cdot x_2)+b\right)\right] - \alpha_2\rho + \alpha_2\xi_2 - \delta\rho + \beta_2\xi_2)$$

$$+ (\alpha_3\left[y_3\left((\mathrm{w}\cdot x_3)+b\right)\right] - \alpha_3\rho + \alpha_3\xi_3 - \delta\rho + \beta_3\xi_3)$$

$$= \left[\frac{1}{2}(\mathrm{w}^T.\mathrm{w}) - v\rho + \left(\frac{1}{3}\xi_1 + \frac{1}{3}\xi_2 + \frac{1}{3}\xi_3\right)\right]$$

$$-\left((\alpha_1\left[y_1\left((\mathrm{w}\cdot x_1)+b\right)\right] - \alpha_1\rho + \alpha_1\xi_1 - \delta\rho + \beta_1\xi_1\right)$$

$$+ (\alpha_2\left[y_2\left((\mathrm{w}\cdot x_2)+b\right)\right] - \alpha_2\rho + \alpha_2\xi_2 - \delta\rho + \beta_2\xi_2)$$

$$+ (\alpha_3\left[y_3\left((\mathrm{w}\cdot x_3)+b\right)\right] - \alpha_3\rho + \alpha_3\xi_3 - \delta\rho + \beta_3\xi_3))$$

$$= \left[\frac{1}{2}\Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big)\Big) - (\alpha_1 + \alpha_2 + \alpha_3 - 3\delta)\rho + \big((\alpha_1 + \beta_1)\xi_1 + (\alpha_2 + \beta_2)\xi_2 + (\alpha_3 + \beta_3)\xi_3\big)\right]$$

$$- \Big((\alpha_1 \Big[y_1 \Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).x_1\Big) + b\Big)\Big] - \alpha_1\rho + \alpha_1\xi_1 - \delta\rho + \beta_1\xi_1\Big)$$

$$+ (\alpha_2 \Big[y_2 \Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).x_2\Big) + b\Big)\Big] - \alpha_2\rho + \alpha_2\xi_2 - \delta\rho + \beta_2\xi_2)$$

$$+ (\alpha_3 \Big[y_3 \Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).x_3\Big) + b\Big)\Big] - \alpha_3\rho + \alpha_3\xi_3 - \delta\rho + \beta_3\xi_3))$$

$$= \left[\frac{1}{2}\Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big)\Big) - (\alpha_1 + \alpha_2 + \alpha_3 - 3\delta)\rho + \big((\alpha_1 + \beta_1)\xi_1 + (\alpha_2 + \beta_2)\xi_2 + (\alpha_3 + \beta_3)\xi_3\big)\right]$$

$$- \Big((\alpha_1 \Big[y_1 \Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).x_1\Big) + b\Big)\Big] - \alpha_1\rho + \alpha_1\xi_1 - \delta\rho + \beta_1\xi_1\Big)$$

$$+ (\alpha_2 \Big[y_2 \Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).x_2\Big) + b\Big)\Big] - \alpha_2\rho + \alpha_2\xi_2 - \delta\rho + \beta_2\xi_2)$$

$$+ (\alpha_3 \Big[y_3 \Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).x_3\Big) + b\Big)\Big] - \alpha_3\rho + \alpha_3\xi_3 - \delta\rho + \beta_3\xi_3))$$

$$= \frac{1}{2}\Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big)\Big) - (\alpha_1 + \alpha_2 + \alpha_3)\rho + (3\delta)\rho + (\alpha_1 \xi_1) + (\alpha_2 \xi_2) + (\alpha_3 \xi_3) + (\beta_1 \xi_1) + (\beta_2 \xi_2) + (\beta_3 \xi_3) - \Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big)\Big) - (\alpha_1 y_1 b) - (\alpha_2 y_2 b) - (\alpha_3 y_3 b) + (\alpha_1 + \alpha_2 + \alpha_3)\rho - (\alpha_1 \xi_1) - (\alpha_2 \xi_2) - (\alpha_3 \xi_3) + 3\delta\rho - (\beta_1 \xi_1) - (\beta_2 \xi_2) - (\beta_3 \xi_3)$$

$$= -\frac{1}{2}\Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big)\Big) - (\alpha_1 y_1 b) - (\alpha_2 y_2 b) - (\alpha_3 y_3 b) + 3\delta\rho$$

$$= -\frac{1}{2}\Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big)\Big) - (\alpha_1 y_1 b) - (\alpha_2 y_2 b) - (\alpha_3 y_3 b) + (3v - \alpha_1 - \alpha_2 - \alpha_3)\rho$$

Perhatikan bahwa agar untuk memaksimalkan nilai dari fungsi *Lagrange* kita perlu membuat $\alpha_1 y_1 + \alpha_2 y_2 + \alpha_3 y_3 = 0$ dan $\alpha_1 + \alpha_2 + \alpha_3 \geq v$, yang mana pada perhitungan manual ini akan dibuat $\alpha_1 + \alpha_2 + \alpha_3 = v$, sehingga akan didapatkan fungsi maksimal dari *Lagrange* dengan formula:

$$\max L(\alpha_i) = -\frac{1}{2}\Big(\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big).\big((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\big)\Big)$$

$$\max L(\alpha_i) = -\frac{1}{2}\Big(\big((\alpha_1 y_1 x_1)(\alpha_1 y_1 x_1) + (\alpha_1 y_1 x_1)(\alpha_2 y_2 x_2) + (\alpha_1 y_1 x_1)(\alpha_3 y_3 x_3)\big) + \big((\alpha_1 y_1 x_1)(\alpha_2 y_2 x_2) + (\alpha_2 y_2 x_2)(\alpha_2 y_2 x_2) + (\alpha_2 y_2 x_2)(\alpha_3 y_3 x_3)\big) + \big((\alpha_1 y_1 x_1)(\alpha_3 y_3 x_3) + (\alpha_2 y_2 x_2)(\alpha_3 y_3 x_3) + (\alpha_3 y_3 x_3)(\alpha_3 y_3 x_3)\big)\Big)$$

$$\max L(\alpha_i) = -\frac{1}{2}\big((\alpha_1^2 y_1^2 x_1^2) + (\alpha_2^2 y_2^2 x_2^2) + (\alpha_3^2 y_3^2 x_3^2) + 2(\alpha_1 \alpha_2 y_1 y_2 x_1 x_2) + 2(\alpha_1 \alpha_3 y_1 y_3 x_1 x_3) + (\alpha_2 \alpha_3 y_2 y_3 x_2 x_3)\big)$$

Persamaan tersebut juga dapat disederhanakan dengan meminimumkan fungsi *Lagrange* diatas menjadi bentuk:

$$\min L(\alpha_i) = \frac{1}{2}\big((\alpha_1^2 y_1^2 x_1^2) + (\alpha_2^2 y_2^2 x_2^2) + (\alpha_3^2 y_3^2 x_3^2) + 2(\alpha_1 \alpha_2 y_1 y_2 x_1 x_2) + 2(\alpha_1 \alpha_3 y_1 y_3 x_1 x_3) + (\alpha_2 \alpha_3 y_2 y_3 x_2 x_3)\big)$$

Dengan kondisi dari permasalahan diatas akan adalah $\alpha_1 y_1 + \alpha_2 y_2 + \alpha_3 y_3 = 0$ dan $\alpha_1 + \alpha_2 + \alpha_3 = v$, dengan $\alpha_i + \beta_i = \frac{1}{3}$, sehingga nilai $\alpha_i \leq \frac{1}{3}$. Berikutnya akan dilakukan perhitungan untuk mencari nilai kernel $K(x_i, x_j)$ dengan menggunakan kernel linear secara manual.

Perhitungan kernel linear dilakukan dengan melakukan perkalian *dot product* pada data x_i yang akan digunakan dalam klasifikasi. Persamaan yang digunakan adalah persamaan kernel linear pada Persamaan (4.6) sehingga didapatkan:

$$K(x_i, x_j) = x_i^T \cdot x_j$$

$K(x_1, x_1) = x_1^T \cdot x_1$

$$= \big((11\text{x}11) + (3\text{x}3) + (4\text{x}4) + (1\text{x}1) + (2\text{x}2) + (1\text{x}1) + (0\text{x}0) + (4\text{x}4) + (2\text{x}2) + (5\text{x}5) + (1\text{x}1)\big) = 198$$

$$K(x_1, x_2) = x_1^T \cdot x_2$$

$$= \big((11\text{x}4) + (3\text{x}5) + (4\text{x}0) + (1\text{x}5) + (2\text{x}3) + (1\text{x}2) + (0\text{x}0) + (4\text{x}2) + (2\text{x}1) + (5\text{x}1) + (1\text{x}2)\big) = 89$$

$$K(x_1, x_3) = x_1^T \cdot x_3$$

$$= \big((11\text{x}8) + (3\text{x}4) + (4\text{x}4) + (1\text{x}8) + (2\text{x}4) + (1\text{x}2) + (0\text{x}0) + (4\text{x}3) + (2\text{x}8) + (5\text{x}4) + (1\text{x}1)\big) = 183$$

Nilai kernel tersebut akan dihitung untuk seluruh data sampel yang ada sehingga didapatkan nilai kernel yang ditunjukkan sebagai berikut:

Tabel 4.8 Nilai Kernel

$K(x_1, x_1)$	$K(x_1, x_2)$	$K(x_1, x_3)$
198	89	183
$K(x_2, x_1)$	$K(x_2, x_2)$	$K(x_2, x_3)$
89	89	128
$K(x_3, x_1)$	$K(x_3, x_2)$	$K(x_3, x_3)$
183	128	270

Nilai dari kernel pada tabel tersebut akan disubstitusikan ke dalam persamaan *Lagrange* dan didapatkan:

$$L(\alpha_i) = \frac{1}{2}\big((\alpha_1^2 y_1^2 x_1^2) + (\alpha_2^2 y_2^2 x_2^2) + (\alpha_3^2 y_3^2 x_3^2) + 2(\alpha_1 \alpha_2 y_1 y_2 x_1 x_2) + 2(\alpha_1 \alpha_3 y_1 y_3 x_1 x_3) + (\alpha_2 \alpha_3 y_2 y_3 x_2 x_3)\big)$$

$$= \frac{1}{2}(198\alpha_1^2 + 89\alpha_2^2 + 270\alpha_3^2 + 178\alpha_1\alpha_2 - 366\alpha_1\alpha_3 - 128\alpha_2\alpha_3)$$

Dengan syarat

(1) $\alpha_1 + \alpha_2 - \alpha_3 = 0 \rightarrow \alpha_1 + \alpha_2 = \alpha_3$

(2) $\alpha_1, \alpha_2, \alpha_3 \leq \frac{1}{3}$

(3) $\alpha_1 + \alpha_2 + \alpha_3 = v$

$$\alpha_1 + \alpha_2 + (\alpha_1 + \alpha_2) = 0{,}05$$

$$2\alpha_1 + 2\alpha_2 = 0{,}05$$

$$\alpha_1 + \alpha_2 = 0{,}025$$

sehingga $\alpha_3 = 0{,}025$ atau $\alpha_1 = 0{,}025 - \alpha_2$

Kemudian persamaan *Lagrange* diatas dapat juga ditulis menjadi:

$$\min L(\alpha_1, \alpha_2) = \frac{1}{2}\big(198\alpha_1^2 + 89\alpha_2^2 + 270(0{,}025)^2 + 178\alpha_1\alpha_2 - 366\alpha_1(0{,}025) - 128\alpha_2(0{,}025)\big)$$

$$\min L(\alpha_1, \alpha_2) = \tfrac{1}{2}(198\alpha_1^2 + 89\alpha_2^2 + 0{,}16875 + 178\alpha_1\alpha_2 - 9{,}15\alpha_1 - 3{,}2\alpha_2)$$

$$\min L(\alpha_2) = \frac{1}{2}(198(0{,}025 - \alpha_2)^2 + 89\alpha_2^2 + 0{,}16875 + 178(0{,}025 - \alpha_2)\alpha_2 - 9{,}15(0{,}025 - \alpha_2) - 3{,}2\alpha_2)$$

$$\min L(\alpha_2) = \frac{1}{2}(198(\alpha_2^2 - 0{,}05\alpha_2 + 0{,}000625) + 89\alpha_2^2 + 0{,}16875 + 4{,}45\alpha_2 - 178\alpha_2^2 - 0{,}22875 + 9{,}15\alpha_2 - 3{,}2\alpha_2)$$

$$\min L(\alpha_2) = \frac{1}{2}(198\alpha_2^2 - 9{,}9\alpha_2 + 0{,}12375 + 89\alpha_2^2 + 0{,}16875 + 4{,}45\alpha_2 - 178\alpha_2^2 - 0{,}22875 + 9{,}15\alpha_2 - 3{,}2\alpha_2)$$

$$\min L(\alpha_2) = \frac{1}{2}(109\alpha_2^2 + 0{,}5\alpha_2 - 0{,}06375)$$

$$\min L(\alpha_2) = 54{,}5\alpha_2^2 + 0{,}025\alpha_2 - 0{,}031875$$

Pada simulasi ini, nilai parameter v yang digunakan pada simulasi dan juga pemodelan seluruh data adalah 0,05. Kemudian untuk mencari nilai minimum dari fungsi *Lagrange Multiplier* akan dilakukan dengan bantuan bahasa pemrograman python pada google collab dan dihasilkan $L(\alpha_2) = -0{,}02173709$. Kemudian akan dilanjutkan untuk perhitungan nilai α sebagai berikut:

$$L(\alpha_2) = 54{,}5\alpha_2^2 + 0{,}025\alpha_2 - 0{,}031875$$

$$-0{,}02173709 = 54{,}5\alpha_2^2 + 0{,}025\alpha_2 - 0{,}031875$$

$$\alpha_{2.1,2.2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$\alpha_{2.1,2.2} = \frac{-0{,}025 \pm \sqrt{0{,}025^2 - (4(45{,}4) - 0{,}031875)}}{2(54{,}5)}$$

$$\alpha_{2.1,2.2} = \frac{-0{,}025 \pm \sqrt{4{,}088125}}{109}$$

$$\alpha_{2.1,2.2} = \frac{-0{,}025 \pm 2{,}021911}{109}$$

Karena $\alpha_i > 0$ maka

$$\alpha_2 = \frac{-0{,}025 + 2{,}021911}{109}$$

$$\alpha_2 = 0{,}018320 \approx 0{,}018$$

Sehingga $\alpha_1 = \alpha_3 - \alpha_2 = 0{,}025 - 0{,}018 = 0{,}007$.

Setelah nilai dari parameter α_1 dan α_2 ditemukan, kemudian akan dicari juga nilai vektor bobot w dan juga bias b sehingga didapatkan hasil:

$$\mathrm{w} = \alpha_1 y_1 x_1 + \alpha_2 y_2 x_2 + \alpha_3 y_3 x_3$$

$$\mathrm{w} = (0{,}007)(1)([11, 3, 4, 1, 2, 1, 0, 4, 2, 5, 1]^T)$$

$$+(0{,}018)(1)([4, 5, 0, 5, 3, 2, 0, 2, 1, 1, 2]^T)$$

$$+(0{,}025)(-1)([8, 4, 4, 8, 4, 2, 0, 3, 8, 4, 1]^T)$$

$$w = \begin{bmatrix} 0{,}148 \\ 0{,}040 \\ 0{,}054 \\ 0{,}013 \\ 0{,}027 \\ 0{,}013 \\ 0 \\ 0{,}054 \\ 0{,}027 \\ 0{,}067 \\ 0{,}013 \end{bmatrix} + \begin{bmatrix} 0{,}053 \\ 0{,}066 \\ 0 \\ 0{,}066 \\ 0{,}040 \\ 0{,}026 \\ 0 \\ 0{,}026 \\ 0{,}013 \\ 0{,}013 \\ 0{,}026 \end{bmatrix} - \begin{bmatrix} 0{,}213 \\ 0{,}106 \\ 0{,}106 \\ 0{,}213 \\ 0{,}106 \\ 0{,}053 \\ 0 \\ 0{,}080 \\ 0{,}213 \\ 0{,}106 \\ 0{,}027 \end{bmatrix} = \begin{bmatrix} -0{,}012 \\ 0 \\ -0{,}052 \\ -0{,}133 \\ -0{,}049 \\ 0{,}013 \\ 0 \\ 0 \\ -0{,}172 \\ -0{,}026 \\ 0{,}013 \end{bmatrix}$$

serta

$$b = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{y_i} - \sum_{j=1}^{m}\alpha_i y_i K(x_i x_j)\right)$$

$$b = \frac{1}{3}\left[\left(\frac{1}{y_1} - \left(\alpha_1 y_1 K(x_1 x_1) + \alpha_1 y_1 K(x_1 x_2) + \alpha_1 y_1 K(x_1 x_3)\right)\right)\right.$$

$$+\left(\frac{1}{y_2} - \left(\alpha_2 y_2 K(x_2 x_1) + \alpha_2 y_2 K(x_2 x_2) + \alpha_2 y_2 K(x_2 x_3)\right)\right)$$

$$\left.+\left(\frac{1}{y_3} - \left(\alpha_3 y_3 K(x_3 x_1) + \alpha_3 y_3 K(x_3 x_2) + \alpha_3 y_3 K(x_3 x_3)\right)\right)\right]$$

$$b = \frac{1}{3}\left[\left(\frac{1}{1} - \left(0{,}007(198) + 0{,}007(89) + 0{,}007(183)\right)\right)\right.$$

$$+\left(\frac{1}{1} - \left(0{,}018(89) + 0{,}018(89) + 0{,}018(128)\right)\right)$$

$$\left.+\left(\frac{1}{-1} - \left(0{,}025(183) + 0{,}025(128) + 0{,}025(270)\right)\right)\right]$$

$$b = \frac{1}{3}[(-2{,}290) + (-4{,}508) + (13{,}525)]$$

$$b = 2{,}242$$

Selanjutnya, setelah nilai parameter α, w, dan b ditemukan maka didapatkan persamaan *hyperplane* sebagai berikut:

$$f(x_i) = \mathrm{w} \cdot x_i + b$$

$$f(x_i) = \begin{bmatrix} -0{,}012 \\ 0 \\ -0{,}052 \\ -0{,}133 \\ -0{,}049 \\ 0{,}013 \\ 0 \\ 0 \\ -0{,}172 \\ -0{,}026 \\ 0{,}013 \end{bmatrix} \cdot x_i + b$$

Kemudian, untuk melakukan proses pengujian klasifikasi dengan menggunakan algoritma *nu*-SVM dapat dilakukan dengan memasukkan nilai parameter pada persamaan:

$$f(x_i) = sign(\mathrm{w} \cdot x_i + b)$$

atau dapat pula dituliskan sebagai bentuk permasalahan dual sehingga

$$f\big(\phi(x_i)\big) = sign\left(\sum_{j=1}^{j=3} \alpha_i y_i (\phi(x_i).(\phi(x_j))) + b\right)$$

$$f\big(\phi(x_1)\big) = sign\Big(\big((0{,}007).1.(\phi(x_1).\big(\phi(x_1)\big)\big) + \big((0{,}018).1.(\phi(x_1).\big(\phi(x_2)\big)\big) + \big((0{,}025).(-1).(\phi(x_1).\big(\phi(x_3)\big)\big) + 2{,}242\Big)$$

$$f\big(\phi(x_1)\big) = sign(1{,}386 + 1{,}602 - 4{,}575 + 2{,}242)$$

$$f\big(\phi(x_1)\big) = sign(0{,}655)$$

$$f\big(\phi(x_1)\big) = +1$$

Berdasarkan perhitungan manual dengan algoritma *nu*-SVM di atas, maka data pertama (x_1) dapat diklasifikasikan sebagai kelas positif (+1). Proses perhitungan diatas akan diterapkan pada seluruh data training untuk mendapatkan model klasifikasi *nu*-SVM yang optimal dalam klasifikasi sekuens DNA manusia pada penderita dan bukan penderita diabetes mellitus.

4.2.2 Hasil Klasifikasi Menggunakan Algoritma *nu*-SVM

Klasifikasi menggunakan algoritma NuSVM pada keseluruhan data sekuens DNA manusia penderita dan bukan penderita diabetes mellitus akan dilakukan dengan menggunakan bahasa pemrograman python pada *notebook google collab*. Parameter yang digunakan pada algoritma *nu*-SVM adalah parameter Nu $(v) = 0{,}05$ dan kernel linier. Setelah model klasifikasi *nu*-SVM didapatkan, dilakukan proses uji performa model dengan menggunakan *Confusion Matrix* dan didapatkan hasil performa dengan metrik sebagai berikut:

Tabel 4.9 Hasil *Confusion Matrix*

Confusion Matrix		Prediksi		
		DMT2	DMT1	NONDM
Aktual	DMT2	173	25	0
	DMT1	17	177	4
	NONDM	1	9	188

Berdasarkan Tabel 4.9 hasil confusion matrix diatas terlihat bahwa model ini mampu memprediksi sekuens DNA penderita diabetes mellitus tipe 2 (DMT2) dengan benar sebanyak 173 kali dari 198 kasus yang sebenarnya. Namun, ada 25 kasus DMT2 yang salah diklasifikasikan sebagai sekuens DNA penderita diabetes

mellitus tipe 1 (DMT1), dan tidak ada kasus DMT2 yang salah diklasifikasikan sebagai sekuens DNA manusia sehat (NONDM). Selain itu, terdapat 18 kasus di mana model salah memprediksi DMT1 atau NONDM sebagai DMT2. Ini menunjukkan bahwa model cukup baik dalam mengidentifikasi sekuens DNA penderita diabetes mellitus tipe 2 (DMT2), meskipun terdapat beberapa kesalahan.

Pada sekuens DNA penderita diabetes mellitus tipe 1 (DMT1), model berhasil memprediksi dengan benar 177 dari 198 kasus yang sebenarnya DMT1. Ada 17 kasus DMT1 yang salah diklasifikasikan sebagai DMT2 dan 4 kasus yang salah diklasifikasikan sebagai NONDM. Selain itu, terdapat 34 kasus di mana model salah memprediksi DMT2 atau NONDM sebagai DMT1. Ini menunjukkan bahwa model juga cukup andal dalam mengidentifikasi DMT1, meskipun terdapat beberapa kesalahan.

Pada kategori sekuens DNA manusia sehat (NONDM), model menunjukkan performa yang sangat baik dengan 188 prediksi benar dari 198 kasus yang sebenarnya NONDM. Hanya ada 1 kasus NONDM yang salah diklasifikasikan sebagai DMT2 dan 9 kasus yang salah diklasifikasikan sebagai DMT1. Terdapat hanya 4 kasus di mana model salah memprediksi DMT2 atau DMT1 sebagai NONDM. Hasil ini menunjukkan bahwa model sangat efektif dalam mengidentifikasi sekuens DNA manusia baik penderita maupun bukan penderita diabetes mellitus. Kemudian, dari hasil prediksi diatas didapatkan nilai performa sebagai berikut:

1. Akurasi

$$\text{Akurasi} = \frac{173 + 177 + 188}{173 + 177 + 188 + 25 + 17 + 4 + 1 + 9} \text{x } 100\%$$

$$= 90{,}57\%$$

2. Presisi

$$\text{Presisi}_{\text{DMT2}} = \frac{TP_{DMT2}}{TP_{DMT2} + FP_{DMT2}} = \frac{173}{173 + 17 + 1} \text{x } 100\% = 90{,}58\%$$

$$\text{Presisi}_{\text{DMT1}} = \frac{TP_{DMT1}}{TP_{DMT1} + FP_{DMT1}} = \frac{177}{177 + 25 + 9} \text{x } 100\% = 83{,}89\%$$

$$\text{Presisi}_{\text{NONDM}} = \frac{TP_{\text{NONDM}}}{TP_{\text{NONDM}} + FP_{\text{NONDM}}} = \frac{188}{188 + 4} \text{x } 100\% = 97{,}92\%$$

$$\text{Presisi} = \frac{\text{Presisi}_{\text{DMT2}} + \text{Presisi}_{\text{DMT1}} + \text{Presisi}_{\text{NONDM}}}{3} = 90{,}8\%$$

3. Recall

$$Recall_{\text{DMT2}} = \frac{TP_{DMT2}}{TP_{DMT2} + FN_{DMT2}} = \frac{173}{173 + 25} \text{x } 100\% = 87{,}37\%$$

$$Recall_{\text{DMT1}} = \frac{TP_{DMT1}}{TP_{DMT1} + FN_{DMT1}} = \frac{177}{177 + 17 + 4} \text{x } 100\% = 89{,}39\%$$

$$Recall_{\text{NONDM}} = \frac{TP_{\text{NONDM}}}{TP_{\text{NONDM}} + FN_{\text{NONDM}}} = \frac{188}{188 + 9 + 1} \text{x } 100\%$$

$$= 94{,}95\%$$

$$Recall = \frac{Recall_{\text{DMT2}} + Recall_{\text{DMT1}} + Recall_{\text{NONDM}}}{3} = 90{,}57\%$$

4. F1-*Score*

$$\text{F1}_{\text{DMT2}} = \frac{2 \text{ x } \text{Presisi}_{\text{DMT2}} \text{ x } \text{Recall}_{DMT2}}{\text{Presisi}_{\text{DMT2}} + \text{Recall}_{DMT2}}$$

$$= \frac{2 \text{ x } 0{,}9058 \text{ x } 0{,}8737}{0{,}9058 + 0{,}8737} \text{x} 100\% = 88{,}95\%$$

$$\text{F1}_{\text{DMT1}} = \frac{2 \text{ x } \text{Presisi}_{\text{DMT1}} \text{x } \text{Recall}_{DMT1}}{\text{Presisi}_{\text{DMT1}} + \text{Recall}_{DMT1}}$$

$$= \frac{2 \text{ x } 0{,}8389 \text{ x } 0{,}8939}{0{,}8389 + 0{,}8939} \text{x } 100\% = 86{,}55\%$$

$$\text{F1}_{\text{NONDM}} = \frac{2 \text{ x } \text{Presisi}_{\text{NONDM}} \text{ x } \text{Recall}_{\text{NONDM}}}{\text{Presisi}_{\text{NONDM}} + \text{Recall}_{\text{NONDM}}}$$

$$= \frac{2 \text{ x } 0{,}9792 \text{ x } 0{,}9495}{0{,}9792 + 0{,}9495} \text{x } 100\% = 96{,}41\%$$

$$\text{F1} = \frac{\text{F1}_{\text{DMT2}} + \text{F1}_{\text{DMT1}} + \text{F1}_{\text{NONDM}}}{3} = 90{,}64\%$$

Berdasarkan hasil perhitungan tersebut, terlihat bahwa model klasifikasi sekuens DNA manusia penderita dan bukan penderita diabetes mellitus menggunakan *nu*-SVM mampu menghasilkan akurasi prediksi sebesar 90,57%. Hal ini mengindikasikan bahwa algoritma *nu*-SVM mampu mengklasifikasikan sekuens DNA manusia dengan sangat baik. Berikutnya pada nilai presisi didapatkan rata-rata dari ketiga kelas sebesar 90,8%, yang berarti dari 594 sekuens DNA sebanyak 539 sekuens DNA berhasil diprediksi secara akurat sesuai dengan kelasnya. Terlihat pula pada nilai *recall* rata-rata dari ketiga kelas sebesar 90,57% yang berarti dari 594 sekuens DNA sebanyak 538 sekuens mampu diidentifikasi dengan benar oleh model. Kemudian diperoleh nilai F1-*Score* rata-rata sebesar 90,64% yang artinya model klasifikasi menggunakan algoritma *nu*-SVM mampu menyeimbangkan antara presisi dan *recall*.

4.3 Evaluasi Performa Model

Model klasifikasi sekuens DNA manusia menggunakan algoritma *nu*-SVM yang telah dibuat memiliki hasil dan performa yang cukup baik. Namun, sebelum digunakan pada kehidupan nyata, model klasifikasi perlu dievaluasi lebih lanjut sehingga performa dapat diukur dengan lebih akurat. Metode evaluasi yang akan digunakan adalah 5-*Fold Cross Validation* yang akan membagi data sekuens DNA manusia menjadi 5 bagian dengan setiap bagian akan diuji terhadap model klasifikasi yang akan dibuat dari 4 bagian sisanya. Jadi, model klasifkasi akan

melalui tahapan pelatihan dan pengujian sebanyak 5 kali sehingga performa yang didapat dapat diukur dengan lebih akurat terhadap seluruh data. Berikut merupakan hasil evaluasi yang didapatkan dari 5-*Fold Cross Validation*:

Tabel 4.10 Metriks Evaluasi dengan *K-Fold Cross Validation*

K	Presisi (%)	*Recall* (%)	*F1-Score* (%)	Akurasi (%)
1	91,08	90,91	90,96	90,91
2	91,64	91,58	91,57	91,58
3	92,17	91,91	91,89	91,91
4	88,16	87,67	87,67	87,69
5	87,83	87,19	87,3	87,18
Rerata	**90,18**	**90**	**90**	**90**

Berdasarkan Tabel 4.9, diperoleh rata-rata presisi sebesar 90,18%, *recall* sebesar 90%, F1-*Score* sebesar 90% dan rata-rata akurasi sebesar 90%. Hasil ini menunjukkan bahwa model klasifkasi sekuens DNA manusia menggunakan algoritma *nu*-SVM memiliki performa yang sangat baik dan akurat dalam mengklasifikasikan data sekuens DNA manusia.

4.4 Kajian Keislaman dengan Hasil Penelitian

Diabetes Mellitus yang terjadi karena tingginya Tingkat glukosa dalam darah merupakan salah satu penyakit paling berbahaya dan mematikan pada manusia. Minimnya gejala pada fase awal diabetes membuat sulitnya diagnosa yang akhirnya berujung pada meningkatnya kasus diabetes secara signifikan. Penderita diabetes memiliki resiko yang tinggi terhadap terpicunya komplikasi lain seperti penyakit jantung, katarak, disfungsi ereksi, penyakit hati, dan pada kasus yang parah dapat

menyebabkan kematian. Salah satu cara untuk mencegah diabetes mellitus ialah menerapkan pola hidup sehat dan rutin melakukan diagnosa terhadap kondisi tubuh.

Allah berfirman dalam Q.S. Al-Baqarah ayat 195 (Kemenag, 2024) yang berbunyi:

> *"Berinfaklah di jalan Allah, janganlah jerumuskan dirimu ke dalam kebinasaan, dan berbuat baiklah. Sesungguhnya Allah menyukai orang-orang yang berbuat baik"* (Q.S. Al-Baqarah: 195)

Melalui ayat diatas, Islam telah mengajarkan kita untuk senantiasa menjaga diri dari berbagai hal yang dapat membawa kepada kebinasaan yang mana dapat diterapkan melalui upaya untuk menjaga pola hidup sehat dan senantiasa melakukan diagnosa lebih awal agar dapat mendeteksi penyakit lebih dini, khususnya terhadap penyakit yang berbahaya seperti diabetes mellitus. Diabetes merupakan penyakit yang dapat menyebabkan berbagai komplikasi serius jika tidak terdiagnosis dan ditangani dengan baik. Dengan adanya metode klasifikasi sekuens DNA manusia yang lebih akurat menggunakan algoritma seperti *nu*-SVM, diharapkan deteksi dini terhadap diabetes mellitus tipe 1 dan tipe 2 dapat dilakukan dengan lebih efektif. Hal ini sejalan dengan perintah dalam ayat tersebut untuk tidak membiarkan diri jatuh ke dalam kebinasaan, di mana deteksi dini dan penanganan yang tepat dapat mencegah terjadinya komplikasi yang lebih parah.

Selain itu, menjaga kesehatan juga sebagai bentuk syukur atas nikmat yang diberikan oleh Allah SWT. Kesehatan adalah salah satu nikmat terbesar yang sering kali dilupakan, sebagaimana dinyatakan dalam hadis Nabi Muhammad SAW (Departemen Agama RI, 2010):

"Dua kenikmatan yang sering kali dilupakan oleh manusia: kesehatan dan waktu luang" (HR. Bukhari No. 6412, dari Ibnu ‘Abbas)

Penelitian ini, yang berfokus pada pembuatan model klasifikasi sekuens DNA manusia untuk mendeteksi diabetes lebih dini, merupakan bentuk upaya untuk menjaga dan mensyukuri nikmat kesehatan. Dengan mengembangkan teknologi dan metode yang lebih baik dalam diagnosa, kita dapat memaksimalkan potensi pencegahan penyakit dan menjaga kesehatan masyarakat secara umum.

BAB V

KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan pada bab-bab sebelumnya maka penulis memperoleh kesimpulan sebagai berikut:

1. Tahapan *preprocessing* dan persiapan data sekuens DNA manusia akan melalui beberapa tahapan untuk mengubah data sekuens DNA yang masih berbentuk *string* menjadi data numerik agar data dapat diproses untuk pembuatan model klasifikasi. Tahapan-tahapan tersebut adalah:
   a. Tahap konversi format dataset dari file berformat .fasta menjadi .csv untuk memudahkan proses analisis dan manipulasi data.
   b. *Cleaning data*, yaitu membersihkan data sekuens DNA dari data yang hilang, terduplikasi, mengandung *whitespace* berlebih, serta dari DNA yang tidak diketahui sehingga didapatkan data sekuens DNA manusia yang telah bersih sejumlah 145 data sekuens DNA manusia penderita diabetes mellitus tipe 1, 443 data sekuens DNA manusia penderita diabetes mellitus tipe 2, dan 989 data sekuens DNA manusia sehat.
   c. *3-mers encoding*, yaitu tahap untuk mengekstraksi fitur atau karakteristik dari sekuens dengan membagi sekuens DNA menjadi beberapa pasangan DNA yang terdiri dari 3 *i* (AAA, AAG, AAC, AAT, dst.).
   d. *CountVectorizer*, yaitu tahap dimana nilai fitur dari tiap sekuens DNA diekstraksi berdasarkan frekuensi munculnya *3-mers* pada sekuens

tersebut. Pada tahap ini, data DNA telah diekstraksi fitur dan karakteristiknya dalam representasi numerik.

e. *Oversampling data*, yaitu tahap pembuatan data sintesis untuk menyeimbangkan jumlah dari tiap kelas pada data sekuens DNA dan dihasilkan data sekuens DNA manusia final sebanyak 989 data sekuens DNA manusia pada penderita diabetes mellitus tipe 1, tipe 2, dan data sekuens DNA manusia sehat..

2. Proses pembuatan model klasifikasi sekuens DNA manusia dengan menggunakan algoritma *nu*-SVM dilakukan dengan menyelesaikan fungsi *Lagrange* pada dualitas *nu*-SVM:

$$L(\alpha_i) = \frac{1}{2}\big((\alpha_1^2 y_1^2 x_1^2) + (\alpha_2^2 y_2^2 x_2^2) + (\alpha_3^2 y_3^2 x_3^2) + 2(\alpha_1 \alpha_2 y_1 y_2 x_1 x_2) + 2(\alpha_1 \alpha_3 y_1 y_3 x_1 x_3) + (\alpha_2 \alpha_3 y_2 y_3 x_2 x_3)\big)$$

Kemudian, dengan menyelesaikan fungsi *Lagrange* sehingga didapatkan nilai α_i, w, dan b sehingga didapatkan fungsi hyperplane yaitu:

$$f(x_i) = \begin{bmatrix} -0{,}012 \\ 0 \\ -0{,}052 \\ -0{,}133 \\ -0{,}049 \\ 0{,}013 \\ 0 \\ 0 \\ -0{,}172 \\ -0{,}026 \\ 0{,}013 \end{bmatrix} \cdot x_i + b$$

Setelah itu, hasil dari model klasifikasi algoritma *nu*-SVM dengan parameter $v = 0{,}05$ dan kernel linear tersebut akan diukur menggunakan *confusion matrix* dan dihasilkan model klasifikasi yang berhasil memprediksi 173 data DNA penderita diabetes mellitus tipe 2 dengan benar dan 25 data sisanya

diprediksi salah, 177 data DNA penderita diabetes mellitus tipe 1 diprediksi dengan benar dan 21 sisanya diprediksi salah, serta 188 data DNA bukan penderita diabetes diprediksi dengan benar dan 10 sisanya diprediksi salah.

3. Tingkat akurasi rata-rata yang diperoleh dari klasifikasi sekuens DNA manusia pada penderita dan bukan penderita diabetes mellitus menggunakan algoritma *nu*-SVM dengan kernel linear dan parameter $v = 0{,}05$ adalah sebesar 90%. Hal ini menunjukkan bahwa algoritma *nu*-SVM mampu mengklasifikasikan sekuens DNA manusia pada penderita dan bukan penderita diabetes mellitus dengan sangat baik. Berdasarkan hasil penelitian ini, algoritma *nu*-SVM dapat menjadi alternatif metode klasifikasi yang dapat digunakan untuk mengklasifikasikan diagnosis penyakit diabetes mellitus berdasarkan sekuens DNA yang dimiliki.

5.2 Saran

Berdasarkan hasil penelitian diatas, maka guna menyempurnakan penelitian selanjutnya terkait topik ini penulis memberikan saran sebagai berikut:

1. Pada tahap *preprocessing* dan persiapan data, penelitian selanjutnya dapat menggunakan metode transformasi data yang berbeda pada sekuens DNA seperti metode TF-IDF untuk melakukan ekstraksi pada fitur sekuens DNA berbasis probabilitas munculnya fitur atau karakteristik tersebut.
2. Pada tahap pembuatan model klasifikasi, penelitian selanjutnya dapat dilakukan dengan menggunakan algoritma yang berbeda sehingga dapat diketahui algoritma mana yang mampu bekerja lebih optimal pada pembuatan model klasifikasi sekuens DNA manusia.

DAFTAR PUSTAKA

Akkaya, U. M., & Kalkan, H. (2021). Classification of DNA Sequences with k-mers Based Vector Representations. *2021 Innovations in Intelligent Systems and Applications Conference (ASYU)*, 1–5. https://doi.org/10.1109/ASYU52992.2021.9599084

Alshayeji, M. H., Sindhu, S. C., & Abed, S. (2023). Viral genome prediction from raw human DNA sequence samples by combining natural language processing and machine learning techniques. *Expert Systems with Applications*, *218*, 119641. https://doi.org/10.1016/j.eswa.2023.119641

Asiful Huda, S. M., Mohiuddin Shoikot, M., Jahan Ila, I., & Anower Hossain, M. (n.d.). *An Effective Machine Learning Approach for Sentiment Analysis on Popular Restaurant Reviews in Bangladesh*.

Assery, N., Xiaohong, Y., Almalki, S., Kaushik, R., & Xiuli, Q. (2019). Comparing learning-based methods for identifying disaster-related tweets. *Proceedings - 18th IEEE International Conference on Machine Learning and Applications, ICMLA 2019*, 1829–1836. https://doi.org/10.1109/ICMLA.2019.00295

bin Ishaq, A. bin M. bin A. (2013). *Tafsir Ibnu Katsir Jilid 8* (pp. 1–113).

Chen, P. H., Lin, C. J., & Schölkopf, B. (2005). A tutorial on v-support vector machines. *Applied Stochastic Models in Business and Industry*, *21*(2), 111–136. https://doi.org/10.1002/asmb.537

Chipot, M. (2009). *Elliptic Equations: An Introductory Course - Hilbert Space Techniques*. https://doi.org/10.1007/978-3-7643-9982-5

Cho, N. H., Shaw, J. E., Karuranga, S., Huang, Y., da Rocha Fernandes, J. D., Ohlrogge, A. W., & Malanda, B. (2018). IDF Diabetes Atlas: Global estimates of diabetes prevalence for 2017 and projections for 2045. *Diabetes Research and Clinical Practice*, *138*, 271–281. https://doi.org/10.1016/j.diabres.2018.02.023

Cole, J. B., & Florez, J. C. (2020). Genetics of diabetes mellitus and diabetes complications. *Nature Reviews Nephrology*, *16*(7), 377–390. https://doi.org/10.1038/s41581-020-0278-5

Departemen Agama RI. (2010). *Al-Qur'an dan terjemahan HR Bukhari no.6412 dari Ibnu 'Abbas*. Bumi Restu.

E. El-Attar, N., M. Moustafa, B., & A. Awad, W. (2022). Deep Learning Model to Detect Diabetes Mellitus Based on DNA Sequence. *Intelligent Automation & Soft Computing*, *31*(1), 325–338. https://doi.org/10.32604/iasc.2022.019970

ElSayed, N. A., Aleppo, G., Aroda, V. R., Bannuru, R. R., Brown, F. M., Bruemmer, D., Collins, B. S., Gaglia, J. L., Hilliard, M. E., Isaacs, D., Johnson, E. L., Kahan, S., Khunti, K., Leon, J., Lyons, S. K., Perry, M. Lou, Prahalad, P., Pratley, R. E., Seley, J. J., … Gabbay, R. A. (2023). 2. Classification and

Diagnosis of Diabetes: Standards of Care in Diabetes—2023. *Diabetes Care*, *46*(Supplement_1), S19--S40. https://doi.org/10.2337/dc23-S002

Fernández, A., García, S., Herrera, F., & Chawla, N. V. (2018). SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary. *Journal of Artificial Intelligence Research*, *61*, 863–905. https://doi.org/10.1613/jair.1.11192

Gunasekaran, H., Ramalakshmi, K., Rex Macedo Arokiaraj, A., Deepa Kanmani, S., Venkatesan, C., & Suresh Gnana Dhas, C. (2021). Analysis of DNA Sequence Classification Using CNN and Hybrid Models. *Computational and Mathematical Methods in Medicine*, *2021*, 1–12. https://doi.org/10.1155/2021/1835056

Hamed, B. A., Ibrahim, O. A. S., & Abd El-Hafeez, T. (2023). Optimizing classification efficiency with machine learning techniques for pattern matching. *Journal of Big Data*, *10*(1). https://doi.org/10.1186/s40537-023-00804-6

Hamza, L. abed Al., Lafta, H. A., & Al-Rashid, S. Z. (2023). Predictive Diabetes Mellitus From DNA Sequences Using Deep Learning. *Al-Bahir Journal for Engineering and Pure Sciences*, *3*(2). https://doi.org/10.55810/2312-5721.1042

Hatmal, M. M., Alshaer, W., Mahmoud, I. S., Al-Hatamleh, M. A. I., Al-Ameer, H. J., Abuyaman, O., Zihlif, M., Mohamud, R., Darras, M., Shhab, M. Al, Abu-Raideh, R., Ismail, H., Al-Hamadi, A., & Abdelhay, A. (2021). Investigating the association of CD36 gene polymorphisms (rs1761667 and rs1527483) with T2DM and dyslipidemia: Statistical analysis, machine learning based prediction, and meta-analysis. *PLoS ONE*, *16*(10 October), 1–29. https://doi.org/10.1371/journal.pone.0257857

Huang, S., Nianguang, C. A. I., Penzuti Pacheco, P., Narandes, S., Wang, Y., & Wayne, X. U. (2018). Applications of support vector machine (SVM) learning in cancer genomics. *Cancer Genomics and Proteomics*, *15*(1), 41–51. https://doi.org/10.21873/cgp.20063

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, *31*(3), 685–695. https://doi.org/10.1007/s12525-021-00475-2

Kazi, A. A., & Blonde, L. (2001). Classification of diabetes mellitus. In *Clinics in Laboratory Medicine* (Vol. 21, Issue 1). https://doi.org/10.5005/jp/books/12855_84

Kemenag. (2024). *Qur'an Kemenag*. https://quran.kemenag.go.id/

Khurana, D., Koli, A., Khatter, K., & Singh, S. (2023). Natural language processing: state of the art, current trends and challenges. *Multimedia Tools and Applications*, *82*(3), 3713–3744. https://doi.org/10.1007/s11042-022-13428-4

Kobat, M. A., Kivrak, T., Barua, P. D., Tuncer, T., Dogan, S., Tan, R. S., Ciaccio, E. J., & Acharya, U. R. (2021). Automated covid-19 and heart failure detection using dna pattern technique with cough sounds. *Diagnostics*, *11*(11). https://doi.org/10.3390/diagnostics11111962

Lencz, T., & Malhotra, A. K. (2009). Pharmacogenetics of antipsychotic-induced side effects. *Dialogues in Clinical Neuroscience*, *11*(4), 405–415. https://doi.org/10.31887/DCNS.2009.11.4/tlencz

Li, H.-L., Pang, Y.-H., & Liu, B. (2021). BioSeq-BLM: a platform for analyzing DNA, RNA and protein sequences based on biological language models. *Nucleic Acids Research*, *49*(22), e129--e129. https://doi.org/10.1093/nar/gkab829

Mahesh, B. (2020). Machine Learning Algorithms - A Review | Enhanced Reader. *International Journal of Science and Research*, *9*(1), 381–386. https://doi.org/10.21275/ART20203995

Qorib, M., Oladunni, T., Denis, M., Ososanya, E., & Cotae, P. (2023). Covid-19 vaccine hesitancy: Text mining, sentiment analysis and machine learning on COVID-19 vaccination Twitter dataset. *Expert Systems with Applications*, *212*, 118715. https://doi.org/10.1016/j.eswa.2022.118715

Ramli, N. E., Yahya, Z. R., & Said, N. A. (2022). Confusion Matrix as Performance Measure for Corner Detectors. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, *29*(1), 256–265. https://doi.org/10.37934/araset.29.1.256265

Ranjan, G. K. (2021). *Introduction to K-Fold Cross-Validation in Python*.

Schölkopf, B., Schölkopf, S., Smola, A. J., Williamson, R. C., & Rsise, P. L. B. (2000). *New Support Vector Algorithms*.

Setiawan, R., Zein, H., & Azdy, R. A. (2023). *Rice Leaf Disease Classification with Machine Learning : An Approach Using Nu-SVM*. *4*(3), 136–144.

Sharabiani, V. R., Khorramifar, A., Karami, H., Lozano, J., Tabor, S., Darvishi, Y., & Gancarz, M. (2023). Non-destructive test to detect adulteration of rice using gas sensors coupled with chemometrics methods. *International Agrophysics*, *37*(3), 235–244. https://doi.org/10.31545/intagr/166009

Shihab, M. Q. (2002). *Tafsir Al-Misbah Pesan, Kesan dan Keserasian AlQur'an* (Volume 9). Lentera Hati.

Solis-Reyes, S., Avino, M., Poon, A., & Kari, L. (2018). An open-source k-mer based machine learning tool for fast and accurate subtyping of HIV-1 genomes. *PLOS ONE*, *13*(11), e0206409. https://doi.org/10.1371/journal.pone.0206409

T R, M., V, V. K., V, D. K., Geman, O., Margala, M., & Guduri, M. (2023). The stratified K-folds cross-validation and class-balancing methods with high-performance ensemble classifiers for breast cancer classification. *Healthcare Analytics*, *4*(August), 100247. https://doi.org/10.1016/j.health.2023.100247

Tran, H. G., Ton-That, L., & Thao, N. G. M. (2023). Lagrange Multiplier-Based Optimization for Hybrid Energy Management System with Renewable Energy Sources and Electric Vehicles. *Electronics (Switzerland)*, *12*(21). https://doi.org/10.3390/electronics12214513

Wei, J., Chu, X., Sun, X. Y., Xu, K., Deng, H. X., Chen, J., Wei, Z., & Lei, M. (2019). Machine learning in materials science. *InfoMat*, *1*(3), 338–358. https://doi.org/10.1002/inf2.12028

WHO. (2023). World Health Organization Diabetes. In *World Health Organization* (Issue December, pp. 1–23).

Yang, F.-J. (2018). An Implementation of Naive Bayes Classifier. *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, 301–306. https://doi.org/10.1109/CSCI46756.2018.00065

Zhang, H., Zou, Q., Ju, Y., Song, C., & Chen, D. (2022). Distance-based Support Vector Machine to Predict DNA N6- methyladenine Modification. *Current Bioinformatics*, *17*(5), 473–482. https://doi.org/10.2174/1574893617666220404145517

Zhang, X., Beinke, B., Kindhi, B. Al, & Wiering, M. (2020). *Comparing Machine Learning Algorithms with or without Feature Extraction for DNA Classification*.

LAMPIRAN

Lampiran 1 Dataset Sekuens DNA Manusia Penderita Diabetes Mellitus
https://github.com/mobiltterbang/DNA_Classification_Project/blob/main/dataset/SKRIPSI_Data_DM_DNA_Sequence.csv

Lampiran 2 Perhitungan Manual Turunan Pertama Pada Variabel Primal

1. $\frac{\partial L}{\partial w_i} = 0$

$$\frac{\partial L}{\partial w_i} = \frac{\partial}{\partial w_i}\left(\frac{1}{2} \parallel \mathrm{w} \parallel^2 - v\rho + \frac{1}{3}(\xi_1 + \xi_2 + \xi_3)\right)$$

$$- \frac{\partial}{\partial w_i}\left((\alpha_1\left[y_1\left((\mathrm{w} \cdot x_1) + b\right)\right] - \alpha_1\rho + \alpha_1\xi_1 - \delta\rho + \beta_1\xi_1\right)$$

$$+ (\alpha_2\left[y_2\left((\mathrm{w} \cdot x_2) + b\right)\right] - \alpha_2\rho + \alpha_2\xi_2 - \delta\rho + \beta_2\xi_2)$$

$$+ (\alpha_3\left[y_3\left((\mathrm{w} \cdot x_3) + b\right)\right] - \alpha_3\rho + \alpha_3\xi_3 - \delta\rho + \beta_3\xi_3))$$

$$= 0$$

$$\frac{\partial L}{\partial w_i} = \frac{\partial}{\partial w_i}\left(\frac{1}{2}\sum_{i=1}^{11} w_i^2 - v\rho + \frac{1}{3}(\xi_1 + \xi_2 + \xi_3)\right)$$

$$- \frac{\partial}{\partial w_i}\left((\alpha_1\left[y_1\left(\left(\sum_{i=1}^{11} w_i x_{1i}\right) + b\right)\right] - \alpha_1\rho + \alpha_1\xi_1 - \delta\rho\right.$$

$$\left. + \beta_1\xi_1\right) + (\alpha_2\left[y_2\left(\left(\sum_{i=1}^{11} w_i x_{2i}\right) + b\right)\right] - \alpha_2\rho + \alpha_2\xi_2$$

$$- \delta\rho + \beta_2\xi_2) + (\alpha_3\left[y_3\left(\left(\sum_{i=1}^{11} w_i x_{3i}\right) + b\right)\right] - \alpha_3\rho$$

$$+ \alpha_3\xi_3 - \delta\rho + \beta_3\xi_3)) = 0$$

$$\sum_{i=1}^{11} w_i - \left((\alpha_1 y_1[x_{1.1}, x_{1.2}, \dots, x_{1.11}]^T) + (\alpha_2 y_2[x_{2.1}, x_{2.2}, \dots, x_{2.11}]^T)\right.$$

$$\left. + (\alpha_3 y_3[x_{3.1}, x_{3.2}, \dots, x_{3.11}]^T)\right) = 0$$

$$(w_1 + w_2 + \cdots + w_n) - ((\alpha_1 y_1[x_{1.1}, x_{1.2}, \dots, x_{1.11}]^T)s$$

$$+ (\alpha_2 y_2[x_{2.1}, x_{2.2}, \dots, x_{2.11}]^T)$$

$$+ (\alpha_3 y_3[x_{3.1}, x_{3.2}, \dots, x_{3.11}]^T)) = 0$$

$$\mathrm{w} - \left((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\right) = 0$$

$$\mathrm{w} = \left((\alpha_1 y_1 x_1) + (\alpha_2 y_2 x_2) + (\alpha_3 y_3 x_3)\right)$$

2. $\frac{\partial L}{\partial b} = 0$

$$\frac{\partial}{\partial b}\left(\frac{1}{2} \parallel \mathrm{w} \parallel^2 - v\rho + \frac{1}{3}(\xi_1 + \xi_2 + \xi_3)\right)$$

$$- \frac{\partial}{\partial b}\left((\alpha_1\left[y_1\left((\mathrm{w} \cdot x_1) + b\right)\right] - \alpha_1\rho + \alpha_1\xi_1 - \delta\rho + \beta_1\xi_1\right)$$

$$+ (\alpha_2\left[y_2\left((\mathrm{w} \cdot x_2) + b\right)\right] - \alpha_2\rho + \alpha_2\xi_2 - \delta\rho + \beta_2\xi_2)$$

$$+ (\alpha_3\left[y_3\left((\mathrm{w} \cdot x_3) + b\right)\right] - \alpha_3\rho + \alpha_3\xi_3 - \delta\rho + \beta_3\xi_3)) = 0$$

$$\frac{\partial}{\partial b}\left(\frac{1}{2} \parallel \mathrm{w} \parallel^2 - v\rho + \frac{1}{3}(\xi_1 + \xi_2 + \xi_3)\right)$$

$$- \frac{\partial}{\partial b}\left((\alpha_1 y_1(\mathrm{w} \cdot x_1) + \alpha_1 y_1 b - \alpha_1\rho + \alpha_1\xi_1 - \delta\rho + \beta_1\xi_1\right)$$

$$+ (\alpha_2 y_2(\mathrm{w} \cdot x_2) + \alpha_2 y_2 b - \alpha_2\rho + \alpha_2\xi_2 - \delta\rho + \beta_2\xi_2)$$

$$\left. + (\alpha_3 y_3(\mathrm{w} \cdot x_3) + \alpha_3 y_3 b - \alpha_3\rho + \alpha_3\xi_3 - \delta\rho + \beta_3\xi_3)\right) = 0$$

$$\alpha_1 y_1 + \alpha_2 y_2 + \alpha_3 y_3 = 0$$

3. $\frac{\partial L}{\partial \xi_i} = 0$

$$\frac{\partial L}{\partial \xi_1} = 0$$

$$\frac{\partial}{\partial \xi_1}\left(\frac{1}{2}\parallel \mathrm{w} \parallel^2 - v\rho + \frac{1}{3}(\xi_1+\xi_2+\xi_3)\right)$$

$$-\frac{\partial}{\partial \xi_1}\left((\alpha_1\left[y_1\left((\mathrm{w}\cdot x_1)+b\right)\right]-\alpha_1\rho+\alpha_1\xi_1-\delta\rho+\beta_1\xi_1\right)$$
$$+(\alpha_2\left[y_2\left((\mathrm{w}\cdot x_2)+b\right)\right]-\alpha_2\rho+\alpha_2\xi_2-\delta\rho+\beta_2\xi_2)$$
$$+(\alpha_3\left[y_3\left((\mathrm{w}\cdot x_3)+b\right)\right]-\alpha_3\rho+\alpha_3\xi_3-\delta\rho+\beta_3\xi_3))=0$$

$$\frac{1}{3}-(\alpha_1+\beta_1)=0$$

$$(\alpha_1+\beta_1)=\frac{1}{3}$$

$$\frac{\partial L}{\partial \xi_2}=0$$

$$\frac{\partial}{\partial \xi_2}\left(\frac{1}{2}\parallel \mathrm{w} \parallel^2 - v\rho + \frac{1}{3}(\xi_1+\xi_2+\xi_3)\right)$$

$$-\frac{\partial}{\partial \xi_1}\left((\alpha_1\left[y_1\left((\mathrm{w}\cdot x_1)+b\right)\right]-\alpha_1\rho+\alpha_1\xi_1-\delta\rho+\beta_1\xi_1\right)$$
$$+(\alpha_2\left[y_2\left((\mathrm{w}\cdot x_2)+b\right)\right]-\alpha_2\rho+\alpha_2\xi_2-\delta\rho+\beta_2\xi_2)$$
$$+(\alpha_3\left[y_3\left((\mathrm{w}\cdot x_3)+b\right)\right]-\alpha_3\rho+\alpha_3\xi_3-\delta\rho+\beta_3\xi_3))=0$$

$$\frac{1}{3}-(\alpha_2+\beta_2)=0$$

$$(\alpha_2+\beta_2)=\frac{1}{3}$$

$$\frac{\partial L}{\partial \xi_3}=0$$

$$\frac{\partial}{\partial \xi_3}\left(\frac{1}{2}\parallel \mathrm{w} \parallel^2 - v\rho + \frac{1}{3}(\xi_1+\xi_2+\xi_3)\right)$$

$$-\frac{\partial}{\partial \xi_1}\left((\alpha_1\left[y_1\left((\mathrm{w}\cdot x_1)+b\right)\right]-\alpha_1\rho+\alpha_1\xi_1-\delta\rho+\beta_1\xi_1\right)$$
$$+(\alpha_2\left[y_2\left((\mathrm{w}\cdot x_2)+b\right)\right]-\alpha_2\rho+\alpha_2\xi_2-\delta\rho+\beta_2\xi_2)$$
$$+(\alpha_3\left[y_3\left((\mathrm{w}\cdot x_3)+b\right)\right]-\alpha_3\rho+\alpha_3\xi_3-\delta\rho+\beta_3\xi_3))=0$$

$$\frac{1}{3}-(\alpha_3+\beta_3)=0$$

$$(\alpha_3 + \beta_3) = \frac{1}{3}$$

Sehingga, didapatkan $\alpha_i + \beta_i = \frac{1}{3}$

4. $\frac{\partial L}{\partial \rho} = 0$

$$\frac{\partial}{\partial \rho}\left(\frac{1}{2} \| \mathrm{w} \|^2 - v\rho + \frac{1}{3}(\xi_1 + \xi_2 + \xi_3)\right)$$

$$-\frac{\partial}{\partial \rho}\big((\alpha_1[y_1((\mathrm{w} \cdot x_1) + b)] - \alpha_1\rho + \alpha_1\xi_1 - \delta\rho + \beta_1\xi_1)$$

$$+ (\alpha_2[y_2((\mathrm{w} \cdot x_2) + b)] - \alpha_2\rho + \alpha_2\xi_2 - \delta\rho + \beta_2\xi_2)$$

$$+ (\alpha_3[y_3((\mathrm{w} \cdot x_3) + b)] - \alpha_3\rho + \alpha_3\xi_3 - \delta\rho + \beta_3\xi_3)) = 0$$

$$(\alpha_1 - \delta) + (\alpha_2 - \delta) + (\alpha_3 - \delta) - v = 0$$

$$(\alpha_1 - \delta) + (\alpha_2 - \delta) + (\alpha_3 - \delta) = v$$

Lampiran 3 Penjabaran Nilai Bias b

$$f(x_i) = y_i = \mathrm{w} \cdot x_i + b$$

$$y_i = \sum_{j=1}^{m} \alpha_i y_i(x_i x_j) + b$$

$$b = y_i - \sum_{j=1}^{m} \alpha_i y_i(x_i x_j)$$

Hitung bias untuk semua *support vector* i, $i = 1,2,\dots,m$ sehingga:

$$b_i = y_i - \sum_{j=1}^{m} \alpha_i y_i(x_i x_j)$$

dan didapatkan nilai bias b sebagai rata-rata dari nilai b yang dihitung dari setiap *support vector* yaitu:

$$b = \frac{1}{m}\sum_{j=1}^{m} b_i$$

$$b = \frac{1}{m}\sum_{j=1}^{m}\left[y_i - \sum_{j=1}^{m} \alpha_i y_i(x_i x_j)\right]$$

karena $y_i = \pm 1$ maka dapat dituliskan juga sebagai

$$b = \frac{1}{m}\sum_{i=1}^{m}\left[\frac{1}{y_i} - \sum_{j=1}^{m}\alpha_i y_i (x_i x_j)\right]$$

Lampiran 4 Script Code Python

```
# import librabries required
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import CountVectorizer,
TfidfVectorizer
from imblearn.over_sampling import SMOTE
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import RandomizedSearchCV
from sklearn.metrics import classification_report

import os
import warnings
warnings.filterwarnings('ignore')
```

```
# load dataset from github
def load_data(url):
  filepath = os.path.join(url, '?raw=true')
  data = pd.read_csv(filepath)
  return data

DNA = load_data(url_github)
DNA.head()
```

```
# define k-mers function to split the sequence into group of
k-mers words
def getKmers(sequence, size=3):
    return [sequence[x:x+size].lower() for x in
range(len(sequence) - size + 1)]

# apply the k-mers function
DNA['k-mers'] = DNA.apply(lambda x: getKmers(x['sequence']),
axis=1)
DNA[['sequence','k-mers']].head()
```

```
# DNA targets
DNA_targets = DNA['class'].copy()
print(f'DNA targets samples :')
print(f'{DNA_targets.sample(6)}')

# transform the k-mers sequences using CountVectorizer with
the bag-of-words = 4
# define the vectorizer
cv = CountVectorizer(ngram_range=(4, 4))

# apply the transformation
DNA_features_CV = cv.fit_transform(DNA_features)

# result samples
DNA_features_CV[0:100].toarray()

le = LabelEncoder()
DNA_targets = le.fit_transform(DNA_targets)
DNA_targets

# oversample the data using SMOTE
sm = SMOTE(random_state=42, sampling_strategy='not majority')
DNA_features_oversampled, DNA_targets_oversampled =
sm.fit_resample(DNA_features_CV, DNA_targets)

# split data again
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test =
train_test_split(DNA_features_oversampled,
DNA_targets_oversampled, stratify = DNA_targets_oversampled,
test_size = 0.20, random_state=42)
```

```
# Model the classification
from sklearn.svm import NuSVC

# define the model
cf = NuSVC(nu=0.05, kernel='linear')

# fit the model
cf.fit(X_train,y_train)

# predict data using model
predicted = cf.predict(X_test)

# model performance 01
print(classification_report(y_test, predicted))

# evaluation
from yellowbrick.classifier import ConfusionMatrix
from yellowbrick.classifier import ClassPredictionError
from yellowbrick.classifier import ROCAUC
from yellowbrick.style import set_palette

cfmatrix = ConfusionMatrix(cf)

cfmatrix.fit(X_train, y_train)
cfmatrix.score(X_test, y_test)
cfmatrix.show()

# Cross Validation
from sklearn.svm import NuSVC
from sklearn.model_selection import RepeatedStratifiedKFold
from sklearn.metrics import accuracy_score, f1_score,
precision_score, recall_score

# Assuming you have your dataset X and labels y

# Specify the number of splits and repeats
n_splits = 5  # Number of folds
n_repeats = 1  # Number of repeats

# Specify NuSVC parameters
nu_value = 0.05
```

```
# Create a RepeatedStratifiedKFold object
rskf = RepeatedStratifiedKFold(n_splits=n_splits,
n_repeats=n_repeats, random_state=42)

# Initialize NuSVC with linear kernel and specified
parameters
model = NuSVC(kernel='linear', nu=nu_value)

acc = []
prec = []
rec = []
f1score = []

# Perform repeated stratified cross-validation
for train_index, test_index in
rskf.split(DNA_features_oversampled,
DNA_targets_oversampled):
    X_train, X_test = DNA_features_oversampled[train_index],
DNA_features_oversampled[test_index]
    y_train, y_test = DNA_targets_oversampled[train_index],
DNA_targets_oversampled[test_index]

    # Train the NuSVC model
    model.fit(X_train, y_train)

    # Make predictions on the test set
    predictions = model.predict(X_test)

    # Evaluate accuracy
    accuracy = accuracy_score(y_test, predictions)
    precision = precision_score(y_test, predictions,
average="macro")
    recall = recall_score(y_test, predictions,
average="macro")
    f1 = f1_score(y_test, predictions, average="macro")

    acc.append(accuracy)
    prec.append(precision)
    rec.append(recall)
    f1score.append(f1)

    print(f"Accuracy:{accuracy}, Precision:{precision},
Recall:{recall}, F1:{f1}")

print(f"Average accuracy : {np.mean(acc)}")
print(f"Average precision : {np.mean(prec)}")
print(f"Average recall : {np.mean(rec)}")
print(f"Average f1 score : {np.mean(f1score)}")
```

RIWAYAT HIDUP

Penulis, Muhammad Fathun Nuha, lahir di Jombang pada tanggal 22 Juli 2001. Penulis lahir dari pasangan Bapak Tjahjoso Saptanaadi dan Ibu Ninik Istiqomah sebagai anak pertama dari 4 bersaudara. Pendidikan pertama penulis, ditempuh di SDN Jombatan 3, kemudian melanjutkan pendidikan menengah pertama di SMPN 2 Jombang dan lulus pada tahun 2017. Setelah itu, penulis menyelesaikan pendidikan menengah atas di SMAN 2 Jombang dan lulus pada tahun 2020. Pada tahun yang sama, penulis melanjutkan studi di Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang mengambil program studi Matematika di Fakultas Sains dan Teknologi. Selama masa kuliah, selain menyelesaikan tugasnya sebagai mahasiswa, penulis juga merupakan santri pada Pondok Pesantren Gasek Sabilurrosyad yang diasuh oleh Abah Kyai Marzuki Mustamar. Penulis juga aktif mengikuti berbagai kegiatan organisasi di dalam kampus seperti, termasuk kepanitiaan dalam KOMET 2021 dan 2022, SIGMA dan PBAK Fakultas. Selain itu, penulis juga merupakan pengurus HMPS "Integral" Matematika pada tahun 2022 sebagai kepala divisi *Internal Public Relation* dan sebagai anggota divisi kewirausahaan pada tahun 2021.

KEMENTERIAN AGAMA RI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
FAKULTAS SAINS DAN TEKNOLOGI
Jl. Gajayana No.50 Dinoyo Malang Telp. / Fax. (0341)558933

BUKTI KONSULTASI SKRIPSI

Nama	:	Muhammad Fathun Nuha
NIM	:	200601110106
Fakultas/Jurusan	:	Sains dan Teknologi/Matematika
Judul Skripsi	:	Implementasi Algoritma *nu-Support Vector Machine* Pada Model Klasifikasi Sekuens DNA Manusia Studi Kasus: Penderita Diabetes Mellitus
Pembimbing I	:	Ari Kusumastuti, M.Pd., M.Si.
Pembimbing II	:	Mohammad Nafie Jauhari, M.Si.

No	Tanggal	Hal	Tanda Tangan
1.	5 Desember 2023	Konsultasi Judul dan Bab I	1.
2.	9 Januari 2023	Konsultasi Bab I, II, dan III	2.
3.	25 Januari 2023	Konsultasi Kajian Agama	3.
4.	27 Januari 2023	ACC Kajian Agama Bab I dan II	4.
5.	8 Februari 2024	ACC Bab I, II, dan III	5.
6.	9 Februari 2024	ACC Seminar Proposal	6.
7.	15 Mei 2024	Konsultasi Revisi Seminar Proposal	7.
8.	20 Mei 2024	Konsultasi Bab IV dan V	8.
9.	22 Mei 2024	Konsultasi Kajian Agama Bab IV	9.
10.	28 Mei 2024	ACC Bab IV dan V	10.
11.	30 Mei 2024	ACC Kajian Agama Bab IV	11.
12.	16 Juni 2024	ACC Seminar Hasil	12.
13.	20 Juni 2024	Konsultasi Revisi Seminar Hasil	13.
14.	25 Juni 2024	ACC Matriks Revisi Seminar Hasil	14.

KEMENTERIAN AGAMA RI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
FAKULTAS SAINS DAN TEKNOLOGI
Jl. Gajayana No.50 Dinoyo Malang Telp. / Fax. (0341)558933

No	Tanggal	Hal	Tanda Tangan
15.	26 Agustus 2024	ACC Sidang Skripsi	15.
16.	30 Agustus 2024	ACC Keseluruhan	16.

Malang, 30 Agustus 2024
Mengetahui,
Ketua Program Studi Matematika

Dr. Elly Susanti, M.Sc.
NIP. 19741129 200012 2 005