PENERAPAN MACHINE LEARNING PADA PREDIKSI DIABETES dan PERKIRAAN WAKTU LATIHAN FISIK

PROYEK UTS PEMBELAJARAN MESIN
KELAS C

OLEH

AMALIA DAMAYANTI HUSAINI

202131002

FAKULTAS TELEMATIKA ENERGI

INSTITUT TEKNOLOGI PERUSAHAAN LISTRIK NEGARA

JAKARTA

2023

Abstrak

Penelitian ini menggunakan dua metode analisis statistic, yaitu Regresi dan Naïve Bayes. Kedua model metode ini digunakan untuk menganalisis dataset terkait exercises dan diabetes. Model pertama menggunakan regresi linear untuk mengidentifikasi pemantauan kondisi pasien untuk menghindari kelelahan ekstrem. Penelitian ini mengusulkan model komputasi untuk memperkirakan kelelahan selasa Latihan sit-to-stand (STS). Model ini memanfaatkan 32 fitur kinematic STS dan detak jantung dari sensor Kinect dan Zephyr. Model hutan acak dengan 60 sub-klasifikasi mencapai akurasi 82,5% dalam mengklasifikasi tiga tingkat kelelahan. Hasil menunjukkan bahwa gerakan tubuh bagian atas adalah fitur kunci, dengan kontribusi dari gerakan tubuh bagian bawah dan detak jantung.

Model kedua menggunakan naïve bayes untuk memaparkan pendekatan pembelajaran yang diawasi untuk menciptakan alat prediksi risiko yang efisien, dengan analisis fitur untuk mengevaluasi dan mengeksplorasi hubungan fitur dengan diabetes. Gejala umum diabetes digunakan dalam melatih dan menguji beberapa model Machine Learning. Berbagai model Machine Learning dievaluasi dengan metrik Presisi, Recall, F-Measure, Akurasi dan AUC, dibandingkan melalui validasi silang dan pemisahan data 10kali lipat.

Pendekatan ini memberikan kontribusi pada pemahaman lebih lanjut tentang faktor-faktor yang berkaitan dengan Latihan fisik dan diabetes, memberikan dasar perkiraan waktu dan Tindakan preventif diabetes yang lebih efektif. Integrasi metode regresi dan klasifikasi Naïve Bayes menghasilkan analisis komprehensif yang dapat diterapkan dalam pemahaman dan prediksi fenomena kompleks dalam masyarakat.

Kata Kunci — Pembelajaran Mesin, Regresi, Klasifikasi, Naïve Bayes, Latihan Fisik, Prediksi Diabetes

Abstract

This research uses two statistical analysis methods, namely Regression and Naïve Bayes. Both models were used to analyze datasets related to exercises and diabetes. The first model uses linear regression to identify the monitoring of the patient's condition to avoid extreme fatigue. This study proposes a computational model to estimate fatigue during sit-to-stand (STS) exercise. The model utilizes 32 kinematic features of STS and heart rate from Kinect and Zephyr sensors. The random forest model with 60 sub-classifications achieved 82.5% accuracy in classifying three levels of fatigue. Results showed that upper body movement was the key feature, with contributions from lower body movement and heart rate.

The second model uses naïve bayes to expose a supervised learning approach to create an efficient risk prediction tool, with feature analysis to evaluate and explore the relationship of features with diabetes. Common diabetes symptoms were used in training and testing several Machine Learning models. Various Machine Learning models were evaluated with Precision, Recall, F-Measure, Accuracy and AUC metrics, compared through cross-validation and 10-fold data splitting.

This approach contributes to further understanding of the factors related to physical exercise and diabetes, providing a basis for time estimation and more effective diabetes preventive measures. The integration of regression and Naïve Bayes classification methods results in a comprehensive analysis that can be applied in the understanding and prediction of complex phenomena in society.

Keywords — Machine Learning, Regression, Klasification, Naïve Bayes, Physical Exercises, Diabetes Prediction

DAFTAR ISI

DAFTAR ISI

Lembar Judul ..... **i**
Abstrak ..... **ii**
DAFTAR ISI ..... **iv**
BAB I **2**
PENDAHULUAN **2**
1.1 Latar Belakang 2
1.2 Rumusan Masalah 1
1.3 Tujuan 1
1.4 Manfaat 1
BAB II **2**
KAJIAN PUSTAKA **2**
2.1 Penelitian yang Relevan 2
2.2 Pembelajaran Mesin 10
2.3 Regresi 11
2.4 Klasifikasi 11
2.5 Algoritma Naïve Bayes (disesuaikan dengan Algoritma yang akan Anda bahas) 11
2.6 Kajian Pustaka lainnya 12
BAB III **13**
HASIL DAN PEMBAHASAN **13**
3.1 Regresi 13
3.2 Algoritma Naïve Bayes 17
BAB IV **22**
PENUTUP **22**
4.1 Kesimpulan 22
4.2 Saran 22
DAFTAR PUSTAKA **23**

BAB I
PENDAHULUAN

1.1 Latar Belakang

Penelitian ini disusun sebagai bagian dari tugas Ujian Semester bagi mahasiswa Informatika di Institut Teknologi-PLN dengan fokus pada mata kuliah Pembelajaran Mesin. Mata kuliah ini menawarkan wawasan mendalam ke dalam konsep, teori, dan aplikasi dari teknik-teknik pembelajaran mesin yang menjadi tulang punggung kecerdasan buatan modern. Dalam kerangka ini, penelitian ini bertujuan untuk memberikan mahasiswa pemahaman yang lebih baik tentang implementasi praktis algoritma pembelajaran mesin dalam konteks aplikasi dunia nyata.

Pada diabetes, tubuh secara tidak efisiensi menghasilkan sedikit atau tidak insulin. Peningkatan gula darah (hiperglikemia) dan gangguan metabolisme glukosa terjadi baik sebagai akibat dari penurunan sekresi insulin atau karena penurunan sensitivitas sel-sel tubuh terhadap aksi hormon ini (insulin). Diabetes sering tidak memiliki gejala. Jika mereka terjadi, gejalanya mungkin termasuk haus, sering buang air kecil, makan berlebihan dan lapar, kelelahan, penglihatan kabur, mual, muntah dan penurunan berat badan (meskipun makan berlebihan). Beberapa orang lebih mungkin untuk mengembangkan dia-betes. Berbagai faktor dapat dipertimbangkan untuk mengevaluasi risiko terkait untuk terjadinya. Secara khusus, model ML telah banyak digunakan untuk mengukur risiko terjadinya penyakit dengan asumsi berbagai fitur atau faktor risiko. Dalam konteks bagian ini, tujuan kami adalah untuk menyajikan karya-karya yang relevan tentang diabetes.

Kerangka kerja untuk prediksi diabetes yang terdiri dari pengklasifikasi pembelajaran mesin yang berbeda, seperti K-Nearest Neighbor, Decision Trees, Random Forest, AdaBoost, Naive Bayes dan XGBoost dan jaringan saraf Multilayer Perceptron. Pengklasifikasi ansambel yang mereka usulkan adalah pengklasifikasi berkinerja terbaik dengan sensitivitas, spesifisitas, tingkat kelalaian palsu, rasio peluang diagnostik dan AUC masing-masing 0,789, 0,934, 0,092, 66,234 dan 0,950.

PE adalah alat mendasar untuk mencegah dan mengobati banyak penyakit tidak menular seperti penyakit kardiovaskular, kanker, stroke, dan diabetes. Oleh karena itu, untuk membantu pasien dan staf klinis untuk mencapai tujuan rehabilitasi tertentu, PE telah dimasukkan ke dalam program rehabilitasi yang berbeda. Di satu sisi, PE digunakan untuk meningkatkan kemampuan kardiovaskular dan pernapasan pasien dalam sesi rehabilitasi jantung dan paru. Dengan demikian, mengingat bahwa duduk dan berdiri adalah beberapa kegiatan yang paling umum.

Tes sit-to-stand (STS) banyak dilaksanakan dalam rehabilitasi fisik. Tes ini terdiri dari duduk dan berdiri dari kursi secepat mungkin selama periode yang ditentukan (antara 30 hingga 120 detik), dan ini dianggap sebagai salah satu latihan tersulit. Oleh karena itu, penelitian telah menunjukkan bahwa sangat diperlukan untuk meningkatkan VO2MAX dan menilai keadaan fisik pasien. Namun, karena intensitasnya yang tinggi, diperlukan pemantauan khusus dibandingkan dengan HIE lainnya.

Oleh karena itu, mengingat pentingnya tes STS dalam program rehabilitasi dan risiko membawa pasien ke kondisi kelelahan tinggi selama sesi, ada kebutuhan untuk mengembangkan metode yang memungkinkan mengelola intensitas latihan.

1.2 Rumusan Masalah

1. Bagaimana penerapan algoritma regresi dapat meningkatkan akurasi prediksi produksi Latihan fisik ?
2. Sejauh mana algoritma Naïve Bayes dapat mengidentifikasi pola dalam kinerja prediksi penyakit diabetes ?

1.3 Tujuan

Menyelidiki bagaimana hasil penelitian ini dapat memberikan pandangan lebih dalam tentang bagaiman algoritma regresi dan naïve bayes dapat diintegrasikan dalam berbagai domain, untuk memberikan solusi yang terintegrasi dan inovatif.

1.4 Manfaat

- Bagi Akademik

  Penerapan machine learning membuka peluang dan memperkaya metode untuk penelitian yang mendalam dan publikasi ilmiah di bidang Kesehatan dan olahraga. Ini menciptakan pemahaman serta mengembangkan model yang lebih dalam tentang hubungan antara dua variable yang kompleks dan berkontribusi pada pengembangan metode analisi data yang lebih canggih.

- Bagi Praktis

  Penerapan model machine learning pada diabetes dan kegiatan fisik dapat membantu individu untuk mengambil Langkah-langkah pencegahan dini, seperti merubah pola hidup, berolahraga secara rutin, serta memperbanyak konsumsi air mineral.

BAB II
KAJIAN PUSTAKA

2.1 Penelitian yang Relevan

Untuk memperkuat hasil penelitian, pada Bab ini berisikan tentang beberapa penelitian terdahulu yang akan dibahas sebagai pembanding serta pedoman dalam memahami dan merancang sebuah metode yang digunakan. Sebagai pembanding penelitian maka akan dirangkum penelitian terdahulu pada Tabel 2.1 sebagai berikut :

Tabel 2.1 Perbandingan Penelitian Dengan Penelitian yang Relevan

No.	1.
Judul	Investigating Health-Related Features and Their Impact on the Prediction of Diabetes Using Machine Learning
Penulis	Hafiz Farooq Ahmad, Hamid Mukhtar, Hesham Alaqail, Mohamed Seliaman and Abdulaziz Alhumam
Tahun	2021
Hasil	Untuk setiap kumpulan data, dua jenis eksperimen dilakukan dengan semua pengklasifikasi. Pada percobaan pertama, kesembilan fitur masukan digunakan. Pada percobaan kedua, kami melakukan seleksi dan eliminasi fitur sebelum melatih dan mengevaluasi pengklasifikasi, yang mengakibatkan hilangnya satu fitur (Gender = M) dari kumpulan data. Dengan delapan fitur terakhir, kami melakukan tugas prediksi sekali lagi. Untuk mengukur kinerja setiap pengklasifikasi, kami menggunakan kinerja yang diterima secara luas. statistik kinerja: Akurasi, presisi, perolehan, dan skor F1 [58]. Untuk evaluasi model, kami menggunakan validasi silang 10 kali lipat di semua percobaan. Pengklasifikasi RF menggunakan n = 100 estimator dengan kedalaman maksimal disetel ke 40. Parameter lain dibiarkan sebagai default oleh perpustakaan scikit-learn. Kedua kumpulan data dievaluasi dengan konfigurasi

	model yang sama. Untuk memungkinkan reproduksi pemisahan yang sama pada percobaan yang berbeda, kami menggunakan benih yang sama untuk menghasilkan keadaan acak untuk kedua kumpulan data
Keterkaitan Penelitian	Dalam penelitian ini, kami menjalankan dua jenis eksperimen untuk setiap kumpulan data, masing-masing melibatkan penggunaan semua pengklasifikasi. Eksperimen pertama dilakukan dengan mempertahankan kesembilan fitur masukan, sementara pada eksperimen kedua, kami melakukan seleksi dan eliminasi fitur sebelum melatih dan mengevaluasi pengklasifikasi. Sebagai contoh, satu fitur (Gender = M) dihapus dari kumpulan data, dan kami melanjutkan tugas prediksi dengan delapan fitur tersisa. Kinerja pengklasifikasi dievaluasi menggunakan metrik yang diterima secara luas, termasuk akurasi, presisi, perolehan, dan skor F1. Proses evaluasi model dilakukan dengan menggunakan validasi silang 10 kali lipat dalam semua eksperimen. Pengklasifikasi Random Forest (RF) dijalankan dengan n = 100 estimator, dan kedalaman maksimal disetel ke 40, dengan parameter lainnya menggunakan nilai default dari perpustakaan scikit-learn. Kedua kumpulan data dievaluasi dengan konfigurasi model yang seragam, dan untuk memastikan reproduksi yang konsisten dalam eksperimen yang berbeda, kami menggunakan benih yang sama untuk menghasilkan keadaan acak pada kedua kumpulan data.Top of Form
No.	2.

Judul	Prediction of Diabetes Disease using Machine Learning Model
Penulis	Amandeep Sharma, Kalpna Guleria, Nitin Goyal
Tahun	2021
Hasil	Makalah penelitian ini menyajikan prediksi diabetes menggunakan pembelajaran mesin model. algoritma pembelajaran yang diawasi seperti Regresi Logistik, Naïve Bayes, Jaringan Syaraf Tiruan, Pohon Keputusan telah digunakan untuk membuat analisis model untuk mengetahui apakah pasien menderita diabetes atau tidak. Akurasi mewakili kesempurnaan suatu algoritma. Model prediksi menunjukkan regresi logistik menampilkan akurasi 80,43% yang merupakan yang tertinggi di antara semuanya. Algoritma Naïve Bayes dan pohon keputusan menampilkan hasil yang sangat kompetitif. Keakuratan Naïve Bayes algoritma sebesar 76.95% dan algoritma Decision tree mempunyai akurasi sebesar 76.52% sehingga final hasil kedua pengklasifikasi sangat dekat satu sama lain. ANN (Saraf Buatan Pengklasifikasi Jaringan) memiliki akurasi 75,21%, yang merupakan yang terendah di antara yang lainnya. Selain akurasi, F-score juga merupakan ukuran lain yang efektif untuk dievaluasi model prediksi. Nilai F-ukuran dapat direpresentasikan

	antara 0 sampai 1. Jika F- nilai ukuran pengklasifikasi apa pun mendekati 1 berarti model pengklasifikasi mewakili kinerja yang lebih baik. Pengklasifikasi regresi logistik mewakili 0,863 F- mengukur, yang tertinggi di antara pengklasifikasi lainnya dan F-measure untuk keputusan tersebut pengklasifikasi pohon adalah 0,817 terendah di antara model lainnya. F- Ukur untuk Naïve Bayes dan Pengklasifikasi ANN masing-masing adalah 0,834 dan 0,819. Oleh karena itu, disimpulkan bahwa untuk kumpulan data diabetes ini, regresi logistik mewakili akurasi dan skor F tertinggi membuat model analitik untuk deteksi diabetes di antara pembelajaran mesin lainnya algoritma
Keterkaitan Penelitian	Dalam penelitian ini, kami mengusulkan prediksi diabetes menggunakan model pembelajaran mesin. Algoritma pembelajaran yang diawasi, seperti Regresi Logistik, Naïve Bayes, Jaringan Syaraf Tiruan, dan Pohon Keputusan, digunakan untuk melakukan analisis guna menentukan apakah pasien menderita diabetes atau tidak. Akurasi dianggap sebagai ukuran keunggulan suatu algoritma, dengan Regresi Logistik menunjukkan akurasi tertinggi sebesar 80,43%. Algoritma Naïve Bayes dan Pohon Keputusan juga menampilkan hasil yang sangat kompetitif, dengan akurasi masing-masing sebesar 76.95% dan 76.52%. Meskipun Akurasi ANN (Pengklasifikasi Jaringan Saraf Buatan) sedikit lebih rendah, yaitu 75,21%, parameter evaluasi F-score menunjukkan bahwa Regresi Logistik memiliki nilai tertinggi, yaitu 0,863, sementara Pohon Keputusan memiliki nilai terendah, yaitu 0,817. F-score untuk Naïve Bayes dan ANN masing-masing adalah 0,834 dan 0,819. Oleh karena itu, dapat disimpulkan bahwa untuk kumpulan data diabetes ini, Regresi Logistik menjadi pilihan utama dengan akurasi dan skor F tertinggi, menjadikannya model analitik unggul dalam deteksi diabetes dibandingkan dengan algoritma pembelajaran mesin lainnya.

No.	3.
Judul	Machine Learning Approach for Fatigue Estimation in Sit-to-Stand Exercise
Penulis	Andrés Aguirre 1,†, Maria J. Pinto 1,†, Carlos A. Cifuentes 1,* , Oscar Perdomo 2 , Camilo A. R. Díaz 3 and Marcela Múnera
Tahun	2021
Hasil	Pertama-tama, penelitian dilakukan untuk mendapatkan kumpulan data sebanyak 660 register sit-to-stand. Itu terdiri dari 32 fitur latihan kinematik/temporal dan detak jantung, masing-masing karakteristik diberi label kondisi kelelahan (rendah, sedang, dan tinggi) berdasarkan Borg nilai skala yang diberikan oleh peserta. Proses analisis dilakukan untuk menentukan fitur terkait yang paling relevan terhadap kondisi kelelahan. Untuk tujuan ini, perilaku dan pola masing-masing diekstraksi karakteristik dianalisis. Hasil penelitian menunjukkan bahwa fitur yang paling penting adalah kedalaman perpindahan bagian tubuh bagian atas, diikuti dengan waktu berdiri dan jantung kecepatan. Oleh karena itu, dapat diasumsikan bahwa kondisi fisiologis pengguna lebih tinggi ciri tubuh, dan ciri tubuh bagian bawah berisi informasi yang relevan mengenai kelelahan

	estimasi selama latihan STS. Akhirnya, pendekatan model estimasi kelelahan diusulkan dengan tujuan untuk menunjukkan hal tersebut fitur-fitur ini dapat diimplementasikan untuk memperkirakan kelelahan dengan akurasi 82,5% dengan sensor yang dapat diakses dan praktis, yang menurut penelitian serupa, dapat diterima jangkauan. Selain itu, model ini memungkinkan klasifikasi tiga kondisi kelelahan: rendah, sedang, dan tinggi. Hal ini memungkinkan peningkatan pemantauan kondisi kelelahan individu mengoptimalkan kinerja mereka dan, akibatnya, pelaksanaan latihan. Oleh karena itu, ini pekerjaan menyajikan pengembangan alat potensial untuk skenario rehabilitasi fisik dan aplikasi telemedis yang telah menjadi area penting selama keadaan darurat global ini disebabkan oleh COVID19.
Keterkaitan penelitian	penelitian ini mengusulkan pendekatan model estimasi kelelahan yang menggunakan fitur-fitur tersebut untuk memperkirakan kelelahan dengan akurasi 82,5%. Model ini dapat diimplementasikan dengan sensor yang praktis dan dapat diakses. Selain itu, model ini memungkinkan klasifikasi tiga tingkat kelelahan: rendah, sedang, dan tinggi. Pendekatan ini dapat meningkatkan pemantauan kondisi kelelahan individu, mengoptimalkan kinerja mereka, dan pada akhirnya, meningkatkan pelaksanaan latihan. Dengan demikian, penelitian ini menghadirkan alat potensial yang dapat digunakan dalam rehabilitasi fisik dan aplikasi telemedis, yang semakin penting dalam konteks darurat global seperti pandemi COVID-19.
No.	4.

Judul	Data-Driven Machine-Learning Methods for Diabetes Risk Prediction
Penulis	Elias Dritsas and Maria Trigka
Tahun	2022
Hasil	Kebiasaan dan gaya hidup dunia modern merupakan dampak dari meningkatnya kejadian tersebut diabetes. Para profesional medis kini memiliki kesempatan, dengan kontribusi dari teknik pembelajaran mesin, untuk menilai risiko relatif dan memberikan pedoman yang sesuai dan intervensi untuk pengelolaan dan pengobatan atau pencegahan diabetes. Dalam artikel penelitian ini, kami menerapkan beberapa model pembelajaran mesin untuk mengidentifikasi individu yang berisiko terkena diabetes berdasarkan faktor risiko tertentu. Eksplorasi data melalui analisis faktor risiko dapat membantu mengidentifikasi hubungan antara fitur-fitur tersebut dan diabetes. Analisis kinerja menunjukkan bahwa pra-pemrosesan data merupakan langkah utama dalam perancangan model yang efisien dan akurat untuk kejadian diabetes. Khususnya, setelah menerapkan SMOTE dengan validasi silang 10 kali lipat, Random Forest dan KNN mengungguli model lainnya dengan akurasi 98,59%. Demikian pula melamar SMOTE dengan pembagian persentase (80:20), Random Forest dan KNN mengungguli model lainnya dengan akurasi 99,22%. Dalam kedua

	kasus tersebut, menerapkan SMOTE, usulan kami model lebih unggul daripada karya penelitian terkait yang dipublikasikan berdasarkan kumpulan data [36]. dengan fitur yang sama yang kami andalkan dalam hal akurasi. Di masa depan, kami bertujuan untuk memperluas kerangka pembelajaran mesin melalui penggunaan metode pembelajaran mendalam dengan menerapkan algoritma Long-Short-Term-Memory (LSTM) dan Convolutional Neural Networks (CNN) dalam dataset yang sama dan membandingkan hasilnya dalam hal akurasi dengan karya terbitan yang relevan.
Keterkaitan Penelitian	Hasil kinerja menunjukkan bahwa penerapan teknik SMOTE, terutama dengan validasi silang 10 kali lipat, menghasilkan Random Forest dan KNN dengan akurasi yang mencapai 98,59%. Demikian pula, dengan pembagian persentase 80:20, Random Forest dan KNN mampu mencapai akurasi sebesar 99,22%. Hasil ini menunjukkan keunggulan model yang diusulkan dalam perbandingan dengan karya penelitian terkait yang menggunakan kumpulan data yang sama. Melihat ke depan, penelitian ini merencanakan perluasan kerangka pembelajaran mesin dengan menerapkan algoritma Long-Short-Term-Memory (LSTM) dan Convolutional Neural Networks (CNN) pada dataset yang sama, dengan tujuan membandingkan hasilnya dalam hal akurasi dengan penelitian yang relevan sebelumnya.
No.	5.
Judul	Use of Machine-Learning and Load–Velocity Profiling to Estimate 1-Repetition Maximums for Two Variations of the Bench-Press Exercise

Tahun	2021
Hasil	Investigasi saat ini menunjukkan bahwa bench-press 1RM bisa akurat diperkirakan dari data uji kecepatan beban yang berasal dari beban submaksimal (40–80% 1RM) dan tanpa perlu menggunakan MVT. Selain itu, hasil OLS menunjukkan relatif sederhana model dapat digunakan untuk memperkirakan bench-press eksentrik-konsentris dan konsentris saja 1RM, dan model pembelajaran mesin tidak diperlukan untuk tujuan ini. Secara kolektif, ini Hasilnya bermanfaat bagi pelatih kekuatan dan pengondisian karena mendukung latihan memperkirakan 1RM tanpa data MVT.
Keterkaitan Penelitian	Penelitian terbaru menunjukkan bahwa nilai satu repetisi maksimal (1RM) pada latihan bench-press dapat diestimasi secara akurat menggunakan data kecepatan beban dari beban submaksimal (40–80% 1RM), tanpa memerlukan metode uji maksimal. Hasil analisis Ordinary Least Squares (OLS) menunjukkan bahwa model yang relatif sederhana dapat digunakan untuk memprediksi baik bench-press eksentrik-konsentris maupun konsentris saja 1RM, tanpa memerlukan model pembelajaran mesin. Temuan ini memberikan manfaat yang signifikan bagi pelatih kekuatan dan pengondisian, karena memungkinkan perkiraan 1RM pada latihan bench-press tanpa bergantung pada data uji maksimal.

2.2 Pembelajaran Mesin

Pembelajaran mesin adalah *"* Bidang studi yang memberikan komputer kemampuan untuk belajar tanpa diprogram secara eksplisit(Arthur Samuel, 1959). Pembelajaran mesin adalah studi tentang algoritme komputer yang memungkinkan program komputer ditingkatkan secara otomatis melalui pengalaman(Tom M. Mitchell (1997)). Machine learning dapat didefinisikan sebagai metode komputasi berdasarkan pengalaman untuk meningkatkan

performa atau membuat prediksi yang akurat(Mohri et.al, 2012). Pembelajaran mesin adalah disiplin ilmu yang memfokuskan pada pengembangan teknik-teknik yang memungkinkan komputer untuk mengatasi tugas-tugas yang kompleks dengan bantuan data.(Christopher M. Bishop.,2006)

2.3 Regresi

Analisis regresi merupakan suatu proses statistik untuk mengestimasi hubungan antara variabel-variabel, yakni berupa teknik-teknik memodelkan dan melakukan analisis beberapa variabel atas dasar bentuk hubungan antara satu variabel tak bebas dan satu atau lebih variabel bebas (prediktor) (Amstrong, 2012:689).Algoritma regresi adalah pendekatan statistik yang digunakan untuk memodelkan hubungan antara satu atau lebih variabel independen (juga disebut prediktor atau fitur) dan variabel dependen (juga disebut target). Tujuan utama dari regresi adalah memahami dan memodelkan hubungan tersebut sehingga kita dapat melakukan prediksi atau estimasi terhadap variabel dependen berdasarkan nilai-nilai variabel independen yang diketahui.

2.4 Klasifikasi

Secara umum, metode pada klasifikasi dibagi menjadi empat tipe berdasarkan cara pembelajarannya, yaitu supervised learning, unsupervised learning, semi-supervised learning, dan reinforcement learning (Yu & He, 2019). Klasifikasi adalah suatu proses memilih dan mengelompokan buku-buku perpustakaan atau bahan pustaka lainnya atas dasar tertentu serta diletakannya secara bersama-sama disuatu tempat. Algoritma klasifikasi bekerja dengan melatih model menggunakan data yang sudah memiliki label, dan model tersebut kemudian dapat digunakan untuk memprediksi kelas dari data baru yang belum diberi label. Hasil klasifikasi dapat memberikan wawasan dan informasi yang berharga untuk pengambilan keputusan dalam berbagai konteks, mulai dari pengelompokan email sebagai spam atau bukan spam hingga identifikasi jenis penyakit berdasarkan gejala.

2.5 Algoritma Naive Bayes (disesuaikan dengan Algoritma yang akan Anda bahas)

Naive Bayes adalah algoritma yang mudah diimplementasikan dan memiliki akurasi yang cukup tinggi dalam banyak kasus. Naive Bayes dapat digunakan untuk klasifikasi biner, klasifikasi multikelas, dan regresi. Naive Bayes adalah algoritma klasifikasi probabilistik yang digunakan dalam pembelajaran mesin. Algoritma ini didasarkan pada teorema Bayes dan asumsi independensi antara fitur.

2.6 Kajian Pustaka lainnya

Dalam pendekatan naïve Bayes semua asumsi bersifat bebas kondisi. Algoritma semacam ini dapat digunakan dalam bidang membangun model di mana dataset memiliki jumlah kejadian yang sangat besar jumlah kejadian. Teorema Bayes didefinisikan sebagai berikut:

P(d/y)=(P(y/d)*P(d))/P(y)

Dimana :

P(d) menandakan sebuah kelas yang merupakan mantan dari P(x/c).

P(y/d) digunakan untuk menunjukkan probabilitas kemungkinan.

P(d/y) mengidentifikasi probabilitas posterior.

P(y) menunjukkan probabilitas prediktor sebelumnya.

Pendekatan ini juga disebut pendekatan pembelajar instan, yang mencapai prediksi kesimpulan dengan sangat cepat untuk sebuah kelas. Ini menunjukkan hasil terbaik untuk masalah klasifikasi multikelas masalah klasifikasi Dibandingkan dengan regresi logistik, ini mengungguli karena membutuhkan lebih sedikit data untuk pelatihan. Memiliki banyak aplikasi, seperti identifikasi dalam teks, penyaringan spam, model rekomendasi, dan analisis sentimental.

BAB III
HASIL DAN PEMBAHASAN

3.1 Regresi

3.1.1 Pengumpulan Data

```
In [2]: import pandas as pd

        data = pd.read_csv('R03_excercises.csv')
        data
```

Out[2]:

	Daily Exercise Time (X)	Weight Loss (Y)
0	0.5	0.2
1	1.0	0.4
2	1.5	0.7
3	2.0	1.0
4	1.0	0.5
...	...	...
69	3.6	1.8
70	1.8	0.7
71	1.9	0.8
72	3.4	1.7
73	1.0	0.4

74 rows × 2 columns

Pengumpulan data model regresi linear terdapat sebuah data yaitu ‘R03_exercises.csv’. agar data dapat terbaca, terlebih dahulu import library menggunakan ‘import pandas as pd’ yang fungsinya untuk memanipulasi data serta menganalisis data. Setelah itu buat inisialisasi untuk datanya yaitu ‘data’ setelah itu masukkan source code untuk membaca data dan menginput nama data yang ingin dibaca yaitu ‘pd.read_csv(R03_exercises.csv)’. kemudian, pemanggilan data cukup dengan menuliskan inisialisasi data yang tadi sudah dibuat.

3.1.2 Preprocessing Data

```
In [4]: from sklearn.model_selection import train_test_split

        X = data['Daily Exercise Time (X)'].values.reshape(-1,1)
        Y = data['Weight Loss (Y)'].values

        x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=.2)
```

- **from sklearn.model_selection import train_test_split** : Kode ini mengimpor fungsi train_test_split dari modul model_selection yang terdapat dalam pustaka scikit-learn. Fungsi ini digunakan untuk membagi dataset menjadi dua bagian: satu untuk pelatihan model dan yang lainnya untuk pengujian.
- **X = data['Daily Exercise Time (X)'].values.reshape(-1,1)**
  Y = data['Weight Loss (Y)'].values .
  Kode ini mengambil dua kolom dari dataframe atau array data, yaitu 'Daily

Exercises Time (X)' dan 'Weight Loss (Y)', dan menyimpannya dalam variable X dan Y. Variabel X berisi fitur (Daily Exercises Time), sedangkan Y berisi target atau label (Weight Loss). Fungsi values digunakan untuk mengonversi data dalam kolom tersebut menjadi bentuk array.

- **x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.2).** X dan Y adalah data yang ingin dibagi. test_size=0.2 menentukan bahwa 20% dari data akan digunakan untuk pengujian, sedangkan 80% akan digunakan untuk pelatihan. Anda dapat mengubah nilai test_size sesuai kebutuhan.

3.1.3 Pembentukan Model

```
In [6]: from sklearn.linear_model import LinearRegression

        model = LinearRegression()
        model.fit(x_train, y_train)

Out[6]: LinearRegression()
```

- from sklearn.linear_model import LinearRegression: Kode ini mengimpor kelas LinearRegression dari modul linear_model yang terdapat dalam Pustaka scikit-learn. Regresi linear adalah metode statistik yang digunakan untuk memodelkan hubungan linear antara variabel dependen (dalam hal ini, y_train) dan satu atau lebih variabel independen (dalam hal ini, x_train).
- model = LinearRegression(): Kode ini membuat objek model menggunakan kelas LinearRegression(). Objek model ini akan merepresentasikan model regresi linear yang akan dilatih menggunakan data pelatihan.
- model.fit(x_train, y_train) : Kode ini menggunakan metode fit dari objek model untuk melatih model dengan menggunakan data pelatihan. Proses pelatihan pada regresi linear melibatkan menemukan parameter (koefisien dan intersep) yang meminimalkan selisih kuadrat antara nilai sebenarnya (y_train) dan nilai yang diprediksi oleh model (x_train). Dengan kata lain, model belajar menyesuaikan garis regresi yang paling baik menggambarkan hubungan antara fitur (x_train) dan target (y_train).

  Pada variabel "Daily Exercises Time" (X), LinearRegression() digunakan untuk memodelkan hubungan linier antara variabel ini (sebagai fitur atau variable independen) dan variabel "Weight Loss" (Y) sebagai variabel target atau variable dependen. Dengan kata lain, kita menggunakan regresi linear untuk mencoba menemukan suatu garis (linear) yang paling baik menggambarkan hubungan antara produksi (X) dan luas panen (Y) dalam dataset.

3.1.4 Analisis akurasi Model

```
In [8]: print(f'Akurasi Regresi: {model.score(x_train, y_train)}')

        Akurasi Regresi: 0.9811627531061387
```

- print(f'Akurasi Regresi: {model.score(x_train, y_train)}') : digunakan untuk mencetak nilai akurasi dari model regresi linear pada data pelatihan.
- Dalam konteks regresi linear, metrik yang sering digunakan untuk mengukur kinerja model pada data pelatihan adalah koefisien determinasi (R-squared). Koefisien determinasi berkisar antara 0 hingga 1, dan semakin mendekati 1, semakin baik model memahami variasi dalam data. Nilai 1 menunjukkan bahwa model dapat menjelaskan seluruh variasi data. Jadi, baris kode tersebut mencetak nilai koefisien determinasi dari model regresi linear pada data pelatihan. Nilai tersebut memberikan indikasi seberapa baik model linear cocok dengan data pelatihan yang diberikan.

  Algoritma regresi linear bekerja dengan mencari hubungan linier antara variabel independen (fitur) dan variabel dependen (target). Regresi linear sederhana (untuk satu fitur) dapat diilustrasikan dengan persamaan garis:

$$Y = b_0 + b_1 \cdot X$$

Y adalah variabel dependen (target),
X adalah variabel independen (fitur),
b_0 adalah intersep (nilai Y ketika $X = 0$),
b_1 adalah koefisien regresi (menunjukkan seberapa banyak Y berubah ketika X berubah).

  Tujuan algoritma regresi linear adalah menemukan nilai b0 dan b1 yang menghasilkan garis regresi terbaik yang paling mendekati data observasi yang sebenarnya

3.1.5 Pengujian Model

```
In [10]: predik = model.predict(x_test)
         predik

Out[10]: array([0.39144874, 1.10241372, 1.75868908, 1.59462024, 0.33675913,
                0.99303449, 1.37586179, 0.44613836, 1.26648256, 1.15710333,
                0.93834488, 0.6102072 , 1.21179295, 0.50082797, 1.48524101])
```

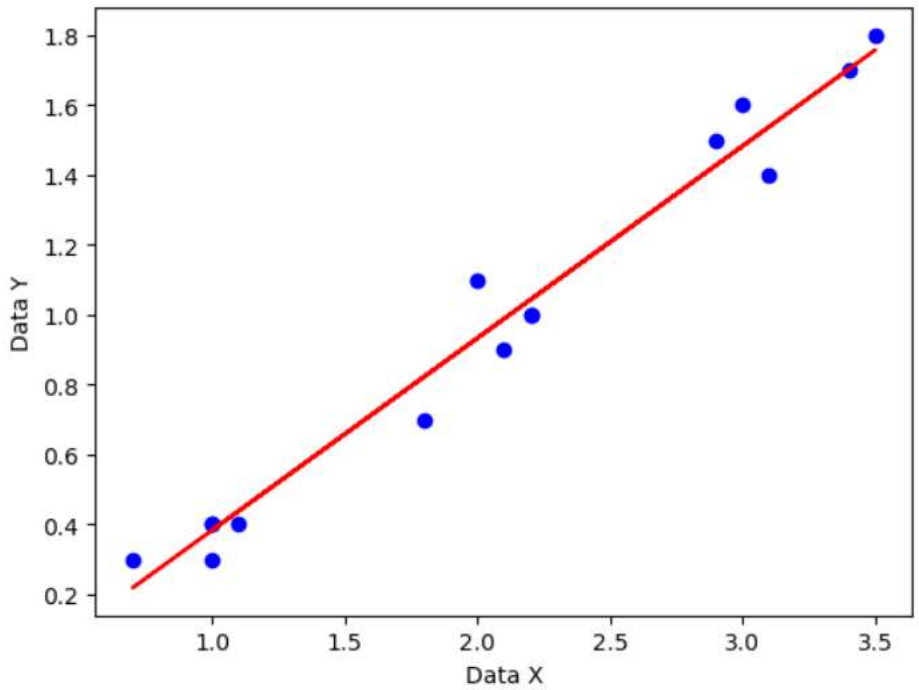
- model.predict(x_test): Metode ini digunakan untuk membuat prediksi dengan menggunakan model yang telah dilatih pada data pengujian (x_test). Hasil prediksi tersebut akan disimpan dalam variabel predik.
- predik: Variabel ini berisi hasil prediksi yang dihasilkan oleh model pada data pengujian. Struktur variabel ini tergantung pada jenis tugas yang

sedang dijalankan (klasifikasi, regresi, dll.). Sebagai contoh, jika Anda melakukan klasifikasi, predik mungkin berisi label kelas prediksi untuk setiap sampel dalam data pengujian

3.1.6 Visualisasi Model

```
In [6]: import matplotlib.pyplot as plt

        plt.scatter(x_test, y_test, c='blue')
        plt.plot(x_test, predik, c='red')
        plt.xlabel('Data X')
        plt.ylabel('Data Y')
        plt.show()
```



• plt.scatter(x_test, y_test, c='blue'): Membuat scatter plot dengan menggunakan data pengujian. Setiap titik pada plot ini mewakili pasangan nilai dari x_test (fitur) dan y_test (nilai sebenarnya).
• plt.plot(x_test, predik, c='red'): Menambahkan garis regresi linear yang diprediksi oleh model pada plot. Garis ini mencoba memodelkan hubungan antara variabel independen (x_test) dan variabel dependen (predik).
• plt.xlabel('Data X') dan plt.ylabel('Data Y'): Menetapkan label pada sumbu-x dan sumbu-y, memberikan konteks terhadap data yang ditampilkan.
• plt.show(): Menampilkan plot.

Dari visualisasi regresi yang dibuat, dapat dilakukan beberapa analisis

tergantung pada pola dan karakteristik dari plot tersebut:
1. Ketepatan Pemetaan:
Jika garis regresi linear (garis merah) sejajar dengan sebagian besar titik data biru, itu menunjukkan bahwa model regresi linear dapat secara cukup baik memodelkan hubungan antara variabel independen dan variabel dependen.
2. Distribusi Residual (Selisih Antara Prediksi dan Nilai Sebenarnya):
Perhatikan distribusi selisih antara nilai sebenarnya (titik biru) dan prediksi model (garis merah). Jika distribusi ini merata dan terdistribusi secara acak di sepanjang sumbu y, itu menunjukkan bahwa model Anda mungkin sesuai dengan data dengan baik.
Analisis visual seperti ini membantu memberikan pemahaman intuitif tentang kinerja model regresi linear pada data tertentu. Namun, penting untuk diingat bahwa analisis tersebut tidak selalu cukup untuk membuat keputusan final, dan evaluasi model yang komprehensif melibatkan penggunaan lebih dari satu metode evaluasi dan statistik.

3.2 Algoritma Naive Bayes (disesuaikan dengan Algoritma yang Anda gunakan)

3.2.1 Pengumpulan Data

```
In [2]: import pandas as pd

        data = pd.read_csv('K02_diabetes.csv')
        data
```

Out[2]:

	gender	age	hypertension	heart_disease	smoking_history	bmi	HbA1c_level	blood_glucose_level	diabetes
0	Female	80.0	0	1	never	25.19	6.6	140	0
1	Female	54.0	0	0	No Info	27.32	6.6	80	0
2	Male	28.0	0	0	never	27.32	5.7	158	0
3	Female	36.0	0	0	current	23.45	5.0	155	0
4	Male	76.0	1	1	current	20.14	4.8	155	0
...	...	...	...	...	...	...	...	...	...
99995	Female	80.0	0	0	No Info	27.32	6.2	90	0
99996	Female	2.0	0	0	No Info	17.37	6.5	100	0
99997	Male	66.0	0	0	former	27.83	5.7	155	0
99998	Female	24.0	0	0	never	35.42	4.0	100	0
99999	Female	57.0	0	0	current	22.43	6.6	90	0

100000 rows × 9 columns

Dataset yang digunakan yaitu diabetes. Terdapat beberapa analisis untuk mengetahui seseorang yang terkena penyakit diabetes atau tidak. Berikut beberapa data analisis nya seperti gender, age, hypertension, heart disease, smoking history, bmi, HbA1c level, dan blood glucose level.

3.2.2 Preprocessing Data

```
In [4]: from sklearn.model_selection import train_test_split
        from sklearn.preprocessing import LabelEncoder

        encode = LabelEncoder()
        for col in data:
            if data[col].dtype == 'object':
                data[col] = encode.fit_transform(data[col])
        data
```

```
In [5]: X = data.drop('diabetes', axis=1)
        Y = data['diabetes']

        x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=.2)
```

- **from sklearn.model_selection import train_test_split**
  from sklearn.preprocessing import LabelEncoder
  Impor perpustakaan yang diperlukan dari scikit-learn. train_test_split digunakan untuk membagi data menjadi set pelatihan dan pengujian, dan LabelEncoder digunakan untuk menyandikan variabel kategori.
- **encode = LabelEncoder()**
  Buat instance kelas LabelEncoder, yang akan digunakan untuk menyandikan variabel kategori.
- **for col in data:**
  Ulangi setiap kolom dalam data DataFrame.
- **if data[col].dtype == 'object':**
  Periksa apakah tipe data kolom saat ini adalah 'objek'. Ini biasanya menunjukkan bahwa kolom tersebut berisi data kategorikal.
- **data[col] = encode.fit_transform(data[col])**
  data
  Gunakan metode fit_transform pada LabelEncoder untuk mengubah nilai kategorikal di kolom saat ini (data[col]) menjadi label numerik. Nilai yang diubah menggantikan nilai kategorikal asli di DataFrame.
  Metode fit_transform menyesuaikan encoder dengan nilai unik di kolom dan kemudian mengubah nilai tersebut menjadi label numerik.
- Proses setelahnya adalah menentukan bahwa 20% dari data akan dialokasikan sebagai data uji, sedangkan 80% akan menjadi data latih.

3.2.3 Pembentukan Model

```
In [7]: from sklearn.naive_bayes import GaussianNB

        model = GaussianNB()
        model.fit(x_train, y_train)

Out[7]: GaussianNB()
```

Membuat objek model GaussianNB. Pada tahap ini, model diinisialisasi dengan parameter default. lalu dibuatkan model.fit sebagai pola datar. x_train adalah data fitur latih, dan y_train adalah data target latih. Model NB akan "mempelajari" pola dalam data latih, sehingga dapat digunakan untuk melakukan prediksi pada data baru. NB adalah salah satu algoritma machine learning yang sederhana dan sering digunakan, terutama untuk masalah klasifikasi.Dalam konteks klasifikasi, GaussianNB digunakan untuk mengklasifikasikan suatu data berdasarkan mayoritas kelas dari k tetangga terdekatnya. Misalnya, jika suatu data memiliki k tetangga terdekat yang kebanyakan termasuk dalam kelas A, maka data tersebut akan diklasifikasikan sebagai kelas A.

3.2.4 Analisis akurasi Model

```
In [9]: print(f'Akurasi Model: {model.score(x_train, y_train)}')

        Akurasi Model: 0.9033625
```

- Akurasi Model Data : {}: Ini adalah string yang akan dicetak. {} adalah tempat penampung (placeholder) untuk nilai yang akan dimasukkan ke dalam string tersebut.
- format(): Ini adalah metode string di Python yang digunakan untuk memasukkan nilai ke dalam string. Dalam hal ini, nilai yang dimasukkan ke dalam tempat penampung {} adalah hasil dari ekspresi di dalam format().
- model.score(x_test, y_test): Ini adalah panggilan metode score dari model yang telah dilatih (model). Metode ini mengukur kinerja model dengan memberikan nilai akurasi, yang merupakan rasio prediksi yang benar terhadap jumlah total sampel. Dalam hal ini, model dievaluasi menggunakan data pengujian (x_test) dan label yang seharusnya benar (y_test).

Menghitung akurasi model pada algoritma K-Nearest Neighbors (KNN) melibatkan perbandingan antara prediksi yang dihasilkan oleh model dengan nilai sebenarnya pada data pengujian. Akurasi dapat dihitung menggunakan formula berikut:

$$\text{Akurasi} = \frac{\text{Jumlah Prediksi Benar}}{\text{Jumlah Total Data Pengujian}}$$

Dari pengujian data diatas di dapatkan Angka 0.7375 adalah nilai akurasi

dari model pada data pengujian, yang berarti sekitar 73.75% dari prediksi model sesuai dengan label yang seharusnya pada data pengujian tersebut. Tingkat akurasi sebesar 73.75% dapat dianggap efektif atau tidak, tergantung pada konteks aplikasi dan karakteristik data yang Anda gunakan

3.2.5 Pengujian Model

```
In [11]: predik = model.predict(x_test)
         predik

Out[11]: array([0, 0, 0, ..., 0, 1, 0], dtype=int64)
```

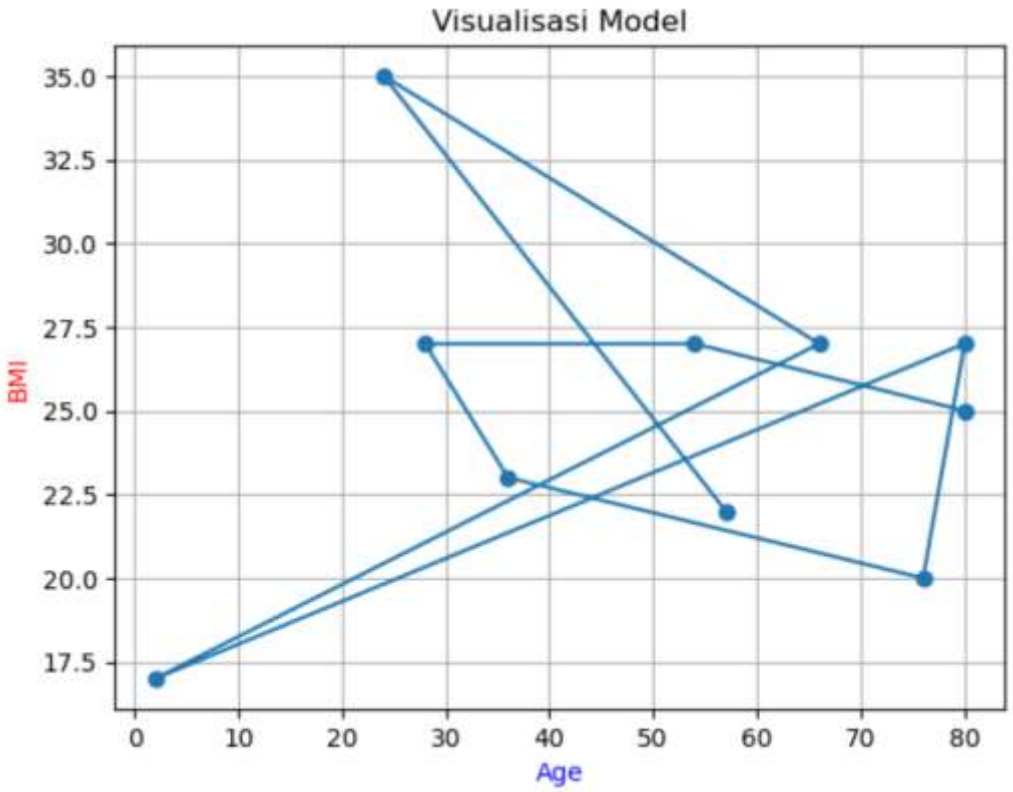
• model.predict(x_test): Metode ini digunakan untuk membuat prediksi dengan menggunakan model yang telah dilatih pada data pengujian (x_test). Hasil prediksi tersebut akan disimpan dalam variabel predik.
• predik: Variabel ini berisi hasil prediksi yang dihasilkan oleh model pada data pengujian. Struktur variabel ini tergantung pada jenis tugas yang sedang dijalankan (klasifikasi, regresi, dll.). Sebagai contoh, jika Anda melakukan klasifikasi, predik mungkin berisi label kelas prediksi untuk setiap sampel dalam data pengujian.

3.2.6 Visualisasi Model

```
In [18]: import matplotlib.pyplot as plt
         import numpy as np
         x = np.array([80, 54, 28, 36, 76, 80, 2, 66, 24, 57])
         y = np.array([25, 27, 27, 23, 20, 27, 17, 27, 35, 22])

         plt.plot(x, y, marker='o')
         plt.grid()

         plt.xlabel('Age', c='blue')
         plt.ylabel('BMI', c='red')
         plt.title('Visualisasi Model')
         plt.show()
```


Visualisasi Model
BMI
Age
35.0
32.5
30.0
27.5
25.0
22.5
20.0
17.5
0
10
20
30
40
50
60
70
80

BAB IV
PENUTUP

4.1 Kesimpulan

Penelitian ini dimulai dengan pengumpulan data dari 660 register latihan sit-to-stand, termasuk 32 fitur kinematik/temporal dan detak jantung, yang diberi label sesuai tingkat kelelahan. Analisis dilakukan untuk menentukan fitur yang paling relevan terkait kelelahan, mengidentifikasi bahwa perpindahan kedalaman bagian tubuh bagian atas, waktu stand-to-stand, dan detak jantung adalah fitur paling penting. Dari hasil ini, diusulkan model estimasi kelelahan yang mencapai akurasi 82,5% dengan menggunakan sensor yang praktis. Model ini dapat mengklasifikasikan tiga tingkat kelelahan dan dapat digunakan untuk memantau kondisi kelelahan individu, meningkatkan kinerja, dan memiliki potensi aplikasi dalam rehabilitasi fisik dan telemedicine, terutama selama keadaan darurat global seperti pandemi COVID-19.

Meningkatnya insiden diabetes dalam kebiasaan dan gaya hidup modern memberikan peluang bagi profesional medis untuk menilai risiko dan memberikan intervensi yang tepat. Dengan menerapkan teknik pembelajaran mesin, penelitian ini menggunakan berbagai model untuk mengidentifikasi individu yang berisiko diabetes berdasarkan faktor risiko. Analisis faktor risiko menyoroti pentingnya pra-pemrosesan data dalam desain model efisien. Hasil menunjukkan bahwa dengan menerapkan teknik SMOTE dan menggunakan validasi silang, model Random Forest dan KNN mencapai akurasi 98,59%, bahkan mencapai 99,22% dengan pembagian persentase 80:20. Model ini mengungguli penelitian terkait dan menunjukkan keunggulan dalam prediksi diabetes. Pekerjaan mendatang akan memperluas pendekatan dengan mengintegrasikan metode pembelajaran mendalam seperti LSTM dan CNN, serta membandingkannya dengan karya yang relevan untuk meningkatkan pemahaman dan akurasi prediksi.

4.2 Saran

Untuk penyempurnaan pembuatan laporan penelitian ini, kami mengharapkan adanya saran dari semua pihak baik dosen, seluruh mahasiswa, dokter spesialis serta atlit-atlit yang membaca laporan hasil penelitian ini.

DAFTAR PUSTAKA

Ahmad, H. F., Mukhtar, H., Alaqail, H., Seliaman, M., & Alhumam, A. (2021). Investigating health-related features and their impact on the prediction of diabetes using machine learning. *Applied Sciences*, *11*(3), 1173.

Sharma, A., Guleria, K., & Goyal, N. (2021). Prediction of diabetes disease using machine learning model. In *International Conference on Communication, Computing and Electronics Systems: Proceedings of ICCCES 2020* (pp. 683-692). Springer Singapore.

Aguirre, A., Pinto, M. J., Cifuentes, C. A., Perdomo, O., Díaz, C. A., & Múnera, M. (2021). Machine learning approach for fatigue estimation in sit-to-stand exercise. *Sensors*, *21*(15), 5006.

Dritsas, E., & Trigka, M. (2022). Data-driven machine-learning methods for diabetes risk prediction. *Sensors*, *22*(14), 5304.

Balsalobre-Fernández, C., & Kipp, K. (2021). Use of machine-learning and load–velocity profiling to estimate 1-repetition maximums for two variations of the bench-press exercise. *Sports*, *9*(3), 39.

Handayanna, F. (2012). Penerapan Particle Swarm Optimization Untuk Seleksi Atribut Pada Metode Support Vector Machine Untuk Prediksi Penyakit Diabetes. *Jakarta: Sekolah Tinggi Manajemen Informatika Dan Komputer Nusa Mandiri*.

Handayanna, F. (2015). PENERAPAN METODE SUPPORT VECTOR MACHINE MENGGUNAKAN OPTIMASI GENETIC ALGORITHM UNTUK PREDIKSI PENYAKIT DIABETES. *Jurnal Teknik Informatika*, *1*(2), 139-147.