

Implementasi Support Vector Machine untuk Klasifikasi Kasus Monkeypox: Pendekatan Oversampling dan Undersampling untuk Mengatasi Ketidakseimbangan Kelas

Cindy ^{1*}, Tiffany Sabatini ², Vincent Itan ³

^{1,2,3} Sistem Informasi, Universitas Universal

*Corresponding author E-mail: cindyyl.263@email.com

Article Info

Article history:

Received 24-06-2024

Revised 24-07-2024

Accepted 25-06-2024

Keyword:

Imbalanced Data, Monkeypox, Random Under Sampling, Random Over Sampling, Support Vector Machine

ABSTRACT

Monkeypox is an infectious disease caused by the monkeypox virus. This study applies the Support Vector Machine (SVM) method to classify monkeypox cases. Utilizing SVM aids in accurate diagnosis and prevention measures. Preprocessing involves Random Oversampling (ROS) and Random Undersampling (RUS) to address class imbalance in symptom datasets. SVM classification is based on systemic symptoms and clinical signs. Evaluation via Confusion Matrix assesses accuracy, sensitivity, specificity, and AUC, with average accuracy reaching 67.1% for imbalanced data and 36.5% for balanced data. The method outperforms conventional techniques, demonstrating its potential in monkeypox symptom pattern recognition. Results indicate higher accuracy in diagnosing monkeypox using SVM, despite class imbalances. This study contributes to understanding, predicting, and managing monkeypox outbreaks effectively.



Copyright © 2024. This is an open access article under the [CC BY](https://creativecommons.org/licenses/by/4.0/) license.

I. PENDAHULUAN

Cacar monyet merupakan penyakit menular yang disebabkan oleh virus *monkeypox* (juga dikenal sebagai virus MPX) tipe zoonosis. Virus MPX pertama kali ditemukan pada tahun 1958, dan diketahui menginfeksi sekelompok monyet jenis *Macaca Cynomolgus* [1].

Pada tahun 2022, *World Health Organization* (WHO) melaporkan penemuan kasus baru di luar negara endemis, yaitu Inggris. Seorang warga negara Inggris yang memiliki riwayat bepergian ke Nigeria membawa virus MPX pertama kali. Sejak Mei 2022 hingga Juni 2022, terdapat 3340 kasus terkonfirmasi dari Amerika dan pada 24 Juni 2022 telah terkonfirmasi seorang warga negara Taiwan yang terkonfirmasi virus MPX dengan riwayat bepergian dari Jerman. Hal ini membuktikan bahwa virus MPX telah mengalami transmisi di berbagai negara [1].

Pada penelitian ini digunakan metode *Support Vector Machine* (SVM) untuk melakukan klasifikasi dalam mendiagnosa *monkeypox*. Vapnik et al. mengembangkan *Support Vector Machine* (SVM) dan pertama kali mempresentasikannya di *Annual Workshop on Computational Learning Theory*. Berbeda dengan strategi

neural network yang berusaha mencari *hyperplane* pemisah antar class, SVM berusaha menemukan *hyperplane* yang terbaik pada *input space*. Prinsip dasar SVM adalah *linear classifier*, dan selanjutnya dikembangkan agar dapat bekerja pada problem non-linear. dengan memasukkan konsep *kernel trick* pada ruang kerja berdimensi tinggi. Perkembangan ini memberikan rangsangan minat penelitian di bidang *pattern recognition* untuk investigasi potensi kemampuan SVM secara teoritis maupun dari segi aplikasi. Saat ini, SVM telah berhasil diaplikasikan dalam problema dunia nyata (*real-world problems*), dan secara umum memberikan solusi yang lebih baik dibandingkan metode konvensional seperti misalnya *artificial neural network* [2].

Dilakukannya penelitian ini, berupaya untuk melakukan pencegahan dan penanganan kasus *monkeypox* agar tidak mengalami peningkatan. Pada penelitian ini menggunakan SVM sebagai metode pengujian untuk mendapatkan akurasi yang lebih baik. Penelitian ini juga melakukan *preprocessing data* menggunakan teknik kombinasi *Random Oversampling* (ROS) dan *Random Undersampling* (RUS) untuk mengatasi ketidakseimbangan kelas pada dataset gejala-gejala *monkeypox*. ROS digunakan untuk meningkatkan jumlah

sampel dari kelas minoritas, sementara RUS digunakan untuk mengurangi jumlah sampel dari kelas mayoritas. Dengan menggunakan teknik ini, dataset menjadi lebih seimbang, sehingga memungkinkan SVM untuk mengklasifikasikan gejala-gejala *monkeypox* dengan lebih baik. Uji ini dilakukan berdasarkan gejala-gejala sistemik dan tanda-tanda klinis dalam tubuh pasien. Sehingga berdasarkan hasil klasifikasi tersebut dapat diketahui apa saja penyebabnya dan bagaimana penanganannya. Kesimpulan dari tujuan program kami dibuat yakni, untuk meneliti, memprediksi dan mengetahui pola gejala *monkeypox* dengan menggunakan metode SVM.

II. METODE

Penelitian ini menggunakan metode eksperimen. Dengan tujuan untuk meningkatkan akurasi dalam memprediksi *monkeypox* menggunakan SVM untuk menganalisis data pasien yang didiagnosa memiliki penyakit *monkeypox*.

2.1 Pengumpulan Data

Data yang digunakan dalam penelitian ini merupakan data sekunder yang didapatkan dari <https://www.kaggle.com/datasets/muhammad4hmed/monkeypox-patients-dataset>. Data tersebut *dipublish* oleh Muhammad Ahmed. Datasetnya tersusun atas 11 nama variabel dan 25000 data pasien [3].

2.2 Data Preprocessing

Pada tahap ini, data yang masih mentah akan dibersihkan dengan mengisi nilai yang kosong, menghilangkan data *noisy* dan menyelesaikan inkonsistensi data yang ditemukan. Kemudian ada 1 variabel yang dihapus yaitu, *Patient_ID* karena variabel tersebut tidak mempengaruhi terhadap kelas penyakit *monkeypox*. Variabel independen pada data ini terdapat 10 variabel dengan keterangan ditunjukkan seperti Tabel 1 di bawah:

Tabel 1. Keterangan Variabel Independen Dataset

NAMA VARIABEL	KETERANGAN
<i>Patient_ID</i>	ID Pasien
<i>Systemic Illness</i>	Penyakit Sistemik (Gejala penyakit pada salah satu organ yang berkaitan dengan sistem metabolisme tubuh manusia)
<i>Rectal Pain</i>	Nyeri Rektal (Sakit di bagian anus atau rektum)
<i>Sore Throat</i>	Sakit Tenggorokan
<i>Penile Oedema</i>	Pembengkakan pada Penis
<i>Oral Lesions</i>	Lesi pada Mulut (Luka, benjolan, atau bercak-bercak di area mana pun di mulut)
<i>Solitary Lesion</i>	Lesi Tunggal (Luka atau

	kerusakan jaringan atau organ tertentu dalam tubuh)
<i>Swollen Tonsils</i>	Pembengkakan pada Amandel
<i>HIV Infection</i>	Terinfeksi HIV
<i>Sexually Transmitted Infection</i>	Infeksi Menular Seksual

Pada tahap *preprocessing* ini, teknik kombinasi *Random Oversampling* (ROS) dan *Random Undersampling* (RUS) digunakan untuk mengatasi ketidakseimbangan kelas pada dataset. *Random Oversampling* (ROS) merupakan penambahan data dari kelas minoritas ke dalam data training secara acak. Proses penambahan ini diulang sampai jumlah data kelas minoritas sama dengan jumlah kelas mayoritas. *Random Oversampling* (ROS) bertujuan untuk meningkatkan ukuran kelas minoritas dengan mensintesis sampel baru atau dataset training dengan menduplikasi secara acak sampel kelas minoritas [4-5].

Random Undersampling (RUS) adalah pendekatan undersampling yang sangat sederhana yang secara acak menghapus *instance* dari kelas mayoritas untuk menyeimbangkan dataset sebelum menerapkan teknik klasifikasi [5-6].

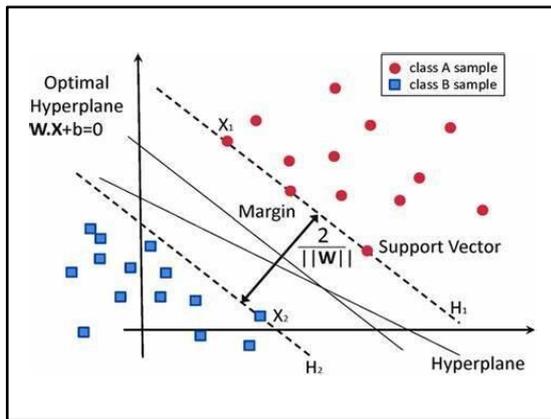
2.3 Classification

Tahapan klasifikasi adalah tahapan dilakukannya prediksi terhadap dataset untuk menemukan suatu pola yang sebelumnya telah melewati tahap *preprocessing*. Pada tahap ini metode klasifikasi yang digunakan adalah algoritma *Support Vector Machine* (SVM) [7].

2.4 Metode SVM

Metode analisis yang digunakan yaitu *Support Vector Machine* (SVM). Metode ini dapat digunakan untuk menangani beberapa kasus, terutama untuk pengklasifikasian data.

Support Vector Machine (SVM) adalah metode learning machine yang bekerja atas prinsip *Structural Risk Minimization* (SRM) dengan tujuan menemukan *hyperplane* terbaik yang memisahkan dua buah *class* pada *input space*. Model algoritma SVM merupakan salah satu algoritma dari metode klasifikasi, yang bekerja dengan cara mencari suatu garis (*hyperplane*) untuk memisahkan dua kelompok data. *Hyperplane* Dengan pemisah terbaik dapat ditemukan dengan mengukur margin dari *hyperplane* dan mencari titik maksimum. Kernel harus digunakan untuk mencapai keberhasilan banyak algoritma klasifikasi untuk permukaan linier. Dengan demikian dapat diketahui bahwa jenis kernel dapat mempengaruhi hasil klasifikasi yang dilakukan. *Hyperplane* adalah garis pemisah terbaik antara dua kelas. Untuk mencari *hyperplane* dapat dilakukan dengan mencari margin *hyperplane* dan mencari titik maksimum. Margin adalah jarak antara data terdekat di antara dua kelas yang berbeda, yang disebut dengan *support vector* [8].



Gambar 1. Klasifikasi Data menggunakan *Support Vector Machine* (SVM)

Pada gambar 1 menunjukkan bahwa garis H_1 , H_2 , dan $hyperplane$ merupakan pemisah kedua kelas. Dimana X adalah *dot product* dari variabel dan konstanta pada setiap notasi dan W adalah nilai yang tegak lurus dengan X [8].

$$w \cdot X_i + b \leq -1 \quad (1)$$

Persamaan 1 merupakan *hyperplane* yang bersinggungan terhadap data yang ada pada kelas A (H_1).

$$w \cdot X_i + b \geq +1 \quad (2)$$

Persamaan 2 merupakan *hyperplane* yang bersinggungan terhadap data yang ada pada kelas B (H_2).

$$w \cdot X + b = 0 \quad (3)$$

Persamaan 3 merupakan *hyperplane* yang berada di antara *hyperplane* kelas A dan kelas B (Garis *Hyperplane*). Sedangkan untuk data yang bersinggungan dengan H_1 di kelas A dan H_2 di kelas B disebut dengan *Support Vector*.

Pencarian titik minimal disebut juga dengan *Quadratic Programming* (QP). Penentuan margin diperlukan untuk menentukan titik minimal yaitu dengan $\frac{1}{\|w\|}$. Berikut ini adalah persamaan untuk mencari titik minimal:

$$\min_w \tau(w) = \frac{1}{2} \|w\|^2 \quad (4)$$

Dengan memperhatikan nilai *constrain*:

$$y_i (X_i \cdot w + b) - 1 \geq 0, \forall_i \quad (5)$$

Sampel data yang ada tidak semua memiliki data yang terpisah secara *linear* sehingga tidak bisa menggunakan SVM *linear*. Apabila dipaksakan nantinya akan memberikan hasil klasifikasi yang buruk dan tidak optimal. Sehingga harus merubah SVM *linear* menjadi SVM *non-linear* agar dapat berjalan dengan optimal, caranya yaitu dengan

memanfaatkan metode *kernel*. Pendekatan ini berbeda dengan metode klasifikasi secara umum, yang sebenarnya mengurangi dimensi awal untuk menyederhanakan proses komputasi dan meningkatkan performansi [9].

2.5 Evaluasi Confusion Matrix

Metode evaluasi yang digunakan pada penelitian adalah *Confusion Matrix*. *Confusion Matrix* atau biasa juga disebut *Error Matrix* adalah metode digunakan dalam melakukan perhitungan akurasi untuk proses klasifikasi atau *Supervised Learning*. Pada perhitungan akurasi terdapat empat 4 kombinasi nilai prediksi dan nilai aktual [8].

		Actual Values	
		1 (Positive)	0 (Negative)
Predicted Values	1 (Positive)	<p>TP (True Positive)</p>	<p>FP (False Positive) Type I Error</p>
	0 (Negative)	<p>FN (False Negative) Type II Error</p>	<p>TN (True Negative)</p>

Gambar 2. *Confusion Matrix*

- Nilai *True Positive* (TP) merupakan data positif yang diprediksi benar.
- Nilai *True Negative* (TN) merupakan data jumlah dari data negatif yang terdeteksi dengan benar.
- Nilai *False Positive* (FP) atau disebut juga *Type-1 Error* merupakan data negatif namun terdeteksi sebagai data yang positif.
- Nilai *False Negative* (FN) atau disebut juga *Type-2 Error* adalah kebalikan dari *True Positive*, di mana data positif tetapi terdeteksi sebagai data negatif [9].

Evaluasi dihitung menggunakan rumus akurasi, sensitifitas, spesifisitas, dan AUC sebagai berikut:

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (6)$$

$$\text{Sensitifitas} = \frac{TP}{TP + FN} \times 100\% \quad (7)$$

$$\text{Spesifisitas} = \frac{TN}{TN + FP} \times 100\% \quad (8)$$

$$\text{AUC} = \frac{\text{Sensitifitas} + \text{Spesifisitas}}{TP + FN} \quad (9)$$

III. HASIL DAN PEMBAHASAN

3.1 Install Package

Pada bagian *install packages* ini, dilakukan instalasi dan penggunaan tiga paket yang berbeda, yaitu:

3.1.1 "ROSE" Package

Paket "ROSE" diinstal dan dimuat menggunakan fungsi `install.packages('ROSE')` dan `library(ROSE)`. Paket ini digunakan untuk menerapkan teknik *Random Over Sampling* (ROS) yang merupakan salah satu teknik untuk menangani ketidakseimbangan kelas pada dataset.

3.1.2 "e1071" Package

Paket "e1071" diinstal dan dimuat dengan menggunakan perintah `install.packages('e1071')` dan `library('e1071')`. Paket ini menyediakan fungsi untuk pembangunan model *Support Vector Machine* (SVM), yang akan digunakan dalam analisis klasifikasi.

3.1.3 "readxl" Package

Paket "readxl" diinstal dan dimuat dengan perintah `install.packages('readxl')` dan `library('readxl')`. Paket ini memungkinkan penggunaan fungsi untuk membaca data dari file Excel ke dalam lingkungan pemrograman R.

3.2 Import Dataset

Dataset awal yang berisi informasi tentang penyakit *monkeypox* diperoleh dari platform *Kaggle* dan diunduh dalam format *Excel*. Selanjutnya data tersebut diimpor ke *Google Colab* untuk dilakukan analisis.

```

kelasImbalanced
  0   1
304 496
1: 192
    
```

Gambar 3. Hasil Kelas Data Imbalanced

Data yang diimpor menunjukkan adanya data imbalanced. Dataset terdiri dari total 25000 data, hanya 800 data yang diambil untuk dianalisis dalam lingkungan *Google Colab*. Dari 800 data tersebut, terdapat 304 data yang dikategorikan sebagai kelas 0 yang mungkin mewakili individu yang sehat dan 496 data dikategorikan sebagai kelas 1 yang mungkin mewakili individu yang terinfeksi penyakit *monkeypox*. Dengan demikian, terdapat selisih sebanyak 192 data antara jumlah kelas 0 dan kelas 1, menunjukkan data imbalanced dalam dataset ini.

3.3 Sampling Data

Pada tahap ini, dilakukan penerapan teknik sampling data yang menggunakan kombinasi antara *Random Oversampling* (ROS) dan *Random Undersampling* (RUS). ROS digunakan untuk menambah sampel pada kelas minoritas (kelas 1), sedangkan RUS digunakan untuk mengurangi sampel pada kelas mayoritas (kelas 0). Teknik ini bertujuan untuk menciptakan dataset yang seimbang antara kelas 0 dan kelas 1.

```

data.balanced <- ovun.sample(kelasImbalanced~, data=dataImbalanced,
N=nrow(dataImbalanced), p=0.5, seed=12, method="both")$data
names(data.balanced)[10]<-"kelasBalanced"
summary(data.balanced)
table(data.balanced$kelasBalanced)
selisihkelasBalanced <- table(data.balanced$kelasBalanced)[2]-table(data.balanced$kelasBalanced)[1]
selisihkelasBalanced
    
```

Gambar 4. Kode Sampling Data

```

Systemic.Illness Rectal.Pain Sore.Throat Penile.Oedema
Length:800 Mode :logical Mode :logical Mode :logical
Class :character FALSE:420 FALSE:402 FALSE:372
Mode :character TRUE :380 TRUE :398 TRUE :428
Oral.Lesions Solitary.Lesion Swollen.Tonsils HIV.Infection
Mode :logical Mode :logical Mode :logical Mode :logical
FALSE:406 FALSE:393 FALSE:379 FALSE:402
TRUE :394 TRUE :407 TRUE :421 TRUE :398
Sexually.Transmitted.Infection kelasBalanced
Mode :logical 1:410
FALSE:416 0:390
TRUE :384

 1 0
410 390
0: -20
    
```

Gambar 5. Hasil Sampling Data

Setelah dilakukan proses sampling menggunakan kombinasi teknik ROS & RUS, dataset yang dihasilkan dinamakan sebagai data balanced. Dengan demikian, setelah proses sampling, terdapat 390 sampel untuk kelas 0 dan 410 sampel untuk kelas 1. Selisih antara jumlah sampel kelas 1 dan kelas 0 adalah -20, menunjukkan adanya penyesuaian sehingga jumlah sampel untuk kedua kelas menjadi lebih seimbang setelah proses sampling.

3.4 Pembuatan Data Uji

Pada tahap ini, dilakukan pembagian dataset menjadi tiga bagian data uji yang berbeda menggunakan pendekatan *K-Fold Cross Validation*. *K-fold cross validation* merupakan salah satu dari teknik yang difungsikan untuk memilah data menjadi train data serta test data. Teknik ini banyak diterapkan peneliti karena didapati mengurangi bias yang didapatkan di dalam pengambilan sebuah sampel [10].

Pertama, data uji pertama dibuat dengan mengambil 33% pertama dari dataset yang diimpor. Data uji kedua kemudian dibuat dengan mengambil 33% bagian kedua dari dataset, yaitu dari posisi ke-33% hingga ke-66%. Terakhir, data uji ketiga dibuat dengan mengambil 33% bagian terakhir dari dataset, dimulai dari posisi ke-66% hingga akhir dataset. Setelah pembagian dataset menjadi tiga bagian data uji, masing-masing bagian diberi nama *dataUji1*, *dataUji2*, dan *dataUji3*.

```

dataUji1 <- dataImbalanced[1:(0.33*nrow(dataImbalanced)), ]
aktual1 <- dataUji1$kelasImbalanced
summary(dataUji1)

dataUji2 <- dataImbalanced[(0.33*nrow(dataImbalanced)+1):(0.66*nrow(dataImbalanced)), ]
aktual2 <- dataUji2$kelasImbalanced
summary(dataUji2)

dataUji3 <- dataImbalanced[(0.66*nrow(dataImbalanced)+1):max(nrow(dataImbalanced)), ]
aktual3 <- dataUji3$kelasImbalanced
summary(dataUji3)
    
```

Gambar 6. Pembuatan Data Uji

Tujuannya adalah untuk membedakan bagian-bagian dari dataset untuk digunakan sebagai data uji secara bergantian. Dengan demikian, setiap bagian dari dataset akan menjadi data uji setidaknya sekali, sehingga memastikan keberagaman data yang diuji.

3.5 Pembangunan Model SVM

```
model.svm.imbalanced <- svm(kelasImbalanced~., data=dataImbalanced)
model.svm.imbalanced
```

Gambar 7. Model SVM pada Data Imbalanced

```
Call:
svm(formula = kelasImbalanced ~ ., data = dataImbalanced)

Parameters:
  SVM-Type:  C-classification
 SVM-Kernel: radial
      cost:  1

Number of Support Vectors: 623
```

Gambar 8. Hasil Model SVM pada Data Imbalanced

Pada tahap ini, dilakukan pembangunan model *Support Vector Machine* (SVM) pada dataset yang mengalami ketidakseimbangan kelas (*imbalanced*). Model SVM dibangun untuk melakukan klasifikasi terhadap data dengan memperhatikan pola yang ada dalam atribut-atribut yang diberikan. Meskipun dataset awalnya tidak seimbang antara kelas 0 dan kelas 1, model SVM akan mencoba untuk menyesuaikan diri dengan kondisi ini dan membangun garis pemisah (*hyperplane*) yang optimal untuk membedakan kedua kelas.

```
model.svm.balanced <- svm(kelasBalanced~., data=data.balanced)
model.svm.balanced
```

Gambar 9. Model SVM pada Data Balanced

```
Call:
svm(formula = kelasBalanced ~ ., data = data.balanced)

Parameters:
  SVM-Type:  C-classification
 SVM-Kernel: radial
      cost:  1

Number of Support Vectors: 692
```

Gambar 10. Hasil Model SVM pada Data Balanced

Selanjutnya, dilakukan pembangunan model SVM pada dataset yang telah mengalami proses sampling sehingga menjadi seimbang antara kelas 0 dan kelas 1 (*balanced*). Model SVM dibangun dengan mempertimbangkan keseimbangan antara kedua kelas, sehingga diharapkan dapat menghasilkan hasil klasifikasi yang lebih baik daripada model yang dibangun pada data yang tidak seimbang. Dengan

demikian, proses pembangunan model SVM dilakukan dua kali: pertama pada data yang tidak seimbang (*imbalanced*) dan kedua pada data yang telah disampling sehingga seimbang (*balanced*). Hal ini bertujuan untuk membandingkan performa model SVM pada kedua jenis dataset tersebut.

3.6 Performa Klasifikasi pada Data Imbalanced dan Data Imbalanced

Tabel 2. Hasil Performa Klasifikasi Data Uji 1

Data	Akurasi	Spesifisitas	Sensitifitas	AUC
Imbalanced	63,6	0,2	0,9	0,6
Balanced	34,5	0,4	0,3	0,4

Dalam uji pertama, terlihat bahwa akurasi pada data imbalanced (63,6%) jauh lebih tinggi daripada pada data balanced (34,5%). Namun, nilai AUC pada data imbalanced (0,6) lebih tinggi daripada pada data balanced (0,4). Selisih akurasi antara data imbalanced dan balanced adalah 29,1%, sedangkan selisih AUC-nya adalah 0,2. Ini menunjukkan bahwa model pada data imbalanced mampu membedakan kedua kelas dengan lebih baik daripada pada data balanced. Ini mungkin disebabkan oleh fakta bahwa model pada data imbalanced memiliki lebih banyak informasi untuk mempelajari pola kelas minoritas, meskipun akurasi secara keseluruhan lebih rendah.

Tabel 3. Hasil Performa Klasifikasi Data Uji 2

Data	Akurasi	Spesifisitas	Sensitifitas	AUC
Imbalanced2	71,2	0,2	0,9	0,6
Balanced	37,9	0,4	0,4	0,4

Pada uji kedua, terlihat bahwa akurasi pada data imbalanced (71,2%) masih jauh lebih tinggi daripada pada data balanced (37,9%). Namun, nilai AUC pada data imbalanced (0,6) lebih tinggi daripada pada data balanced (0,4). Selisih akurasi antara data imbalanced dan balanced adalah 33,3%, sedangkan selisih AUC-nya adalah 0,2. Ini menunjukkan bahwa model pada data imbalanced memiliki kemampuan yang lebih baik dalam membedakan kedua kelas. Hal ini mungkin disebabkan oleh pembelajaran yang lebih baik terhadap pola kelas minoritas pada data imbalanced.

Tabel 4. Hasil Performa Klasifikasi Data Uji 3

Data	Akurasi	Spesifisitas	Sensitifitas	AUC
Imbalanced	66,5	0,3	0,9	0,6
Balanced	37,1	0,3	0,4	0,4

Dalam uji ketiga, akurasi pada data imbalanced (66.5%) masih lebih tinggi daripada pada data balanced (37.1%). Namun, nilai AUC pada data imbalanced (0,6) lebih tinggi daripada pada data balanced (0,4). Selisih akurasi antara data imbalanced dan balanced adalah 29,4%, sedangkan selisih AUC-nya adalah 0,2. Ini menunjukkan bahwa model pada data imbalanced memiliki kemampuan yang lebih baik dalam membedakan kedua kelas. Sama seperti pada uji sebelumnya, hal ini mungkin disebabkan oleh pembelajaran yang lebih baik terhadap pola kelas minoritas pada data imbalanced.

Tabel 5. Hasil Performa Klasifikasi Data Rata-Rata Ketiga Data Uji

Data	Akurasi	Spesifisitas	Sensitifitas	AUC
Rata-Rata Imbalanced	67,1	0,2	0,9	0,6
Rata-Rata Balanced	36,5	0,4	0,4	0,4

Secara rata-rata dari ketiga data uji, akurasi pada data imbalanced (67,1%) masih jauh lebih tinggi daripada pada data balanced (36,5%). Namun, nilai AUC pada data imbalanced (0,6) sedikit lebih tinggi daripada pada data balanced (0,4). Selisih akurasi antara data imbalanced dan balanced secara rata-rata adalah 30,6%, sedangkan selisih AUC-nya adalah 0,2. Ini menunjukkan bahwa meskipun model pada data imbalanced memiliki performa yang lebih baik dalam memprediksi kelas mayoritas, kemampuannya dalam membedakan kedua kelas keseluruhan tidak jauh berbeda dengan model pada data balanced.

IV. KESIMPULAN

Penelitian ini menerapkan metode *Support Vector Machine* (SVM) untuk mengklasifikasikan penyakit *monkeypox*, dengan tujuan meningkatkan akurasi prediksi penyakit. Penggunaan teknik kombinasi *Random Oversampling* (ROS) dan *Random Undersampling* (RUS) terbukti efektif dalam mengatasi ketidakseimbangan kelas pada dataset gejala *monkeypox*. Evaluasi metrik, termasuk akurasi, spesifisitas, dan AUC, menunjukkan disparitas yang signifikan antara dataset yang imbalanced dan seimbang balanced. Meskipun data imbalanced menghasilkan akurasi yang lebih tinggi 67,1% dibandingkan dengan data balanced 36,5%, spesifisitas untuk data imbalanced lebih rendah 0,2 daripada data balanced 0,4. Namun, nilai AUC tetap sedikit lebih tinggi untuk data imbalanced 0,6 daripada data balanced 0,4, menunjukkan kemampuan SVM untuk membedakan antara kelas meskipun adanya kelas imbalanced. Temuan ini

menegaskan efektivitas SVM dalam mengidentifikasi gejala dan berkontribusi pada pemahaman dan manajemen yang lebih baik terhadap wabah *monkeypox*.

Kesimpulan ini juga menggambarkan kesesuaian dengan tujuan penelitian yang dijabarkan dalam pendahuluan. Meskipun data imbalanced memiliki akurasi yang lebih tinggi, namun terdapat pengorbanan pada spesifisitasnya. Namun demikian, SVM tetap terbukti efektif dalam membedakan antara kelas penyakit *monkeypox* meskipun adanya ketidakseimbangan kelas. Hal ini menegaskan pentingnya penggunaan metode SVM dalam mendiagnosis dan mencegah penyakit *monkeypox* secara efisien, sesuai dengan tujuan penelitian untuk meningkatkan pengenalan pola gejala penyakit tersebut.

DAFTAR PUSTAKA

- [1] A. Y. Pratama, R. J. S. Toisuta, and J. Y. Tamba, "Tinjauan atas Monkeypox", *Cermin Dunia Kedokteran*, vol. 50, no. 2, pp. 75–81, Feb. 2023.
- [2] A.S. Nugroho, A.B. Witarto, dan D. Handoko, "Application of Support Vector Machine in Bioinformatics," dalam *Proceeding of Indonesian Scientific Meeting in Central Japan, Gifu-Japan*, 20 Desember 2003.
- [3] Muhamad Ahmed. (2022). Monkey-Pox PATIENTS Dataset. [Data set]. Kaggle. <https://doi.org/10.34740/KAGGLE/DSV/4271503>.
- [4] O. Dhifa, Z. Jurusan, M. Fakultas, and M. Dan, "PENANGANAN IMBALANCE DATA DENGAN RANDOM OVERSAMPLING (ROS) PADA KLASIFIKASI PENDERITA DIABETES MENGGUNAKAN SUPPORT VECTOR MACHINE (SVM) (Skripsi)," 2023. Accessed: Apr. 29, 2024.
- [5] A. R. Purnajaya and F. D. Hanggara, "Perbandingan Performa Teknik Sampling Data untuk Klasifikasi Pasien Terinfeksi Covid-19 Menggunakan Rontgen Dada", *JAIK*, vol. 5, no. 1, pp. 37-42, Jun. 2021.
- [6] A. R. Purnajaya and F. D. Hanggara, "Using Data Sampling Technique for Improving Classification of Covid-19 and Lung Diseases", *UNISSET 2021*, EAI, pp. 167-175, Aug. 2022, <https://doi.org/10.4108/eai.2-12-2021.2320242>
- [7] N. P. D. T. Yanti and I. M. D. P. Asana, "Sistem Klasifikasi Pengajuan Kredit dengan Metode Support Vector Machine (SVM)," *Jurnal Sistem Cerdas*, vol. 6, no. 2, pp. 123–133, Aug. 2023.
- [8] Purbolaksono, M. D., Irvan Tantowi, M., Imam Hidayat, A., & Adiwijaya, A. (2021). Perbandingan Support Vector Machine dan Modified Balanced Random Forest dalam Deteksi Pasien Penyakit Diabetes. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(2), 393 - 399. <https://doi.org/10.29207/resti.v5i2.3008>
- [9] A. Zuhairah, "Penerapan Algoritma Random Forest, Support Vector Machines (SVM) Dan Gradient Boosted Tree (GBT) Untuk Deteksi Penipuan (Fraud Detection) Pada Transaksi Kartu Kredit," (Skripsi), Feb. 2022.
- [10] T. Ridwansyah, "Implementasi Text Mining Terhadap Analisis Sentimen Masyarakat Dunia Di Twitter Terhadap Kota Medan Menggunakan K-Fold Cross Validation Dan Naïve Bayes Classifier," *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 2, no. 5, pp. 178–185, Apr. 2022.