p-ISSN: 1979-5661
e-ISSN: 2622-321X

Implementasi IQR-SMOTE Untuk Mengatasi Ketidakseimbangan Kelas Pada Klasifikasi Diabetes menggunakan K-Nearest Neighbors

Muhammad Syaoki Faradisa[a1], Muliadi[a2], Dodon Turianto Nugrahadi[a3]
Irwan Budiman[a4], Dwi Kartini[a5]

[a]Ilmu Komputer Fakultas MIPA Universitas Lambung Mangkurat
Jl. A. Yani Km. 36 Banjarbaru, Kalimantan Selatan, telp. (0511) 473112
[1]syaokifaradisa09@gmail.com
[2]muliadi@ulm.ac.id
[3]dodonturianto@ulm.ac.id
[4]irwan.budiman@ulm.ac.id
[5]dwikartini@ulm.ac.id

Abstrak

Salah satu penyakit paling berbahaya adalah diabetes yang berada urutan ketiga paling mematikan di Indonesia setelah stroke dan jantung. Banyak cara untuk mendeteksi penyakit ini lebih dini, salah satunya adalah dengan melakukan klasifikasi menggunakan machine learning. Pada penelitian ini akan menggunakan teknik Interquartile Range untuk melakukan deteksi data outlier pada suatu dataset kemudian teknik SMOTE untuk melakukan oversampling data. Data diabetes memiliki jumlah 268 kelas diabetes dan sebanyak 500 kelas negatif. Penelitian dilakukan dengan membandingkan model K-Nearest Neighbors dengan dan tanpa oversampling pada data outlier berserta penerapan oversampling pada keseluruhan data untuk melihat model yang lebih baik dalam mengklasifikasikan diabetes. Dari perbandingan tersebut, diperoleh hasil bahwa model menggunakan oversampling pada data outlier dan keseluruhan data training (KNN + IQR-SMOTE) merupakan model yang terbaik dari semua model berdasarkan dengan performa f1-score sebesar 68,04%.

Keywords: *Diabetes, K-Nearest Neighbors, SMOTE, Interquartile Range, IQR-SMOTE, Data Outlier*

1. Pendahuluan

Diabetes Mellitus adalah penyakit yang disebabkan karena kurangnya hormon insulin atau tubuh yang tidak mampu memanfaatkan insulin yang merupakan hormon yang dihasilkan dari pankreas untuk memberikan sinyal kepada sel tubuh dalam penyerapan glukosa sehingga kadar glukosa (gula darah) tidak dapat terkendali [1].

Menurut IDF (International Diabetes Federation), bahwa Indonesia menempati urutan ke tujuh dari sepuluh negara dengan pasien diabetes tertinggi. Menurut WHO (World Health Organization) pada tahun 2030 mendatang, penderita diabetes di Indonesia terus meningkat signifikan sampai 21,3 juta jiwa jika tidak dilakukan upaya pencegahan. Menurut BPJS (Badan Penyelenggara Jaminan Sosial) kalau tidak melakukan upaya dalam mencegah diabetes, maka lambat laun pasti merugikan perekonomian nasional dan permasalahannya akan bertambah rumit mengingat banyak kondisi masyarakat Indonesia hidup di bawah garis kemiskinan [2].

Algoritma KNN (K-Nearest Neighbor) merupakan metode yang melakukan prediksi/klasifikasi terhadap objek baru berdasarkan jarak paling dekat dari objek yang ada berdasarkan mayoritas dari kelas data training. Kelas yang paling banyak muncul akan menjadi kelas hasil prediksi/klasifikasi [3]. Nilai k tetangga yang tidak dapat ditentukan secara matematik. Jadi, proses pelatihan pada dasarnya dengan mengobservasi sejumlah k sampai menghasilkan nilai k paling optimum [4] dan secara umum nilai K yang digunakan merupakan bilangan ganjil untuk menghindari adanya jarak/voting yang sama pada proses klasifikasi [5]. Pada metode K-Nearest Neighbors memiliki kelemahan beberapanya yaitu sensitif terhadap data berderau maupun pencilan [4] dan kinerja yang dipengaruhi oleh ketidakseimbangan kelas [6].

p-ISSN: 1979-5661
e-ISSN: 2622-321X

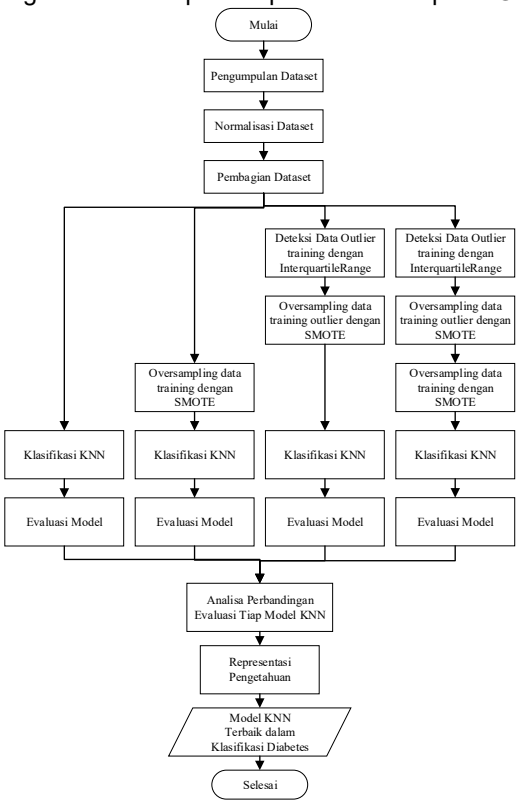
SMOTE (Synthetic Minority Oversampling Technique) merupakan teknik oversampling untuk membuat objek baru pada data. Pada penelitian Siringoringo (2018) dikatakan bahwa teknik SMOTE mampu meningkatkan hasil F1 Score secara signifikan pada metode K-Nearest Neighbors diantaranya seperti pada k = 1 dari 39,1% menjadi 82,2%, k = 3 dari 40,4% menjadi 82,2%, k = 5 dari 41,1% menjadi 82,1%, k = 7 dari 41,5% menjadi 81,6% dan pada k=9 dari 41,5% menjadi 81% [7]. Pada penelitian yang dilakukan oleh Nnamoko (2020), mengatakan bahwa data outlier merupakan data langka yang bisa diperbanyak menggunakan oversampling untuk membuat pola yang jelas serta menggunakan oversampling juga pada keseluruhan data untuk mengatasi ketidakseimbangan kelas (teknik IQR-SMOTE) dan mampu meningkatkan performa F1 Score pada metode Ripper dari 76,6% menjadi 83,6% dengan memakai IQR-SMOTE yang lebih baik daripada hanya memakai SMOTE sebesar 78,1%. Lalu juga meningkatkan F1 Score pada metode C4.5 dari 74,4% menjadi 89,5% dengan memakai IQR-SMOTE yang lebih baik daripada hanya memakai SMOTE sebesar 81,4% [8].

Pada penelitian sebelumnya, hanya digunakan klasifikasi dengan metode K-Nearest Neighbors dengan menggunakan teknik SMOTE untuk mengatasi ketidakseimbangan kelas tanpa melakukan oversampling pada data outlier untuk meningkatkan jumlah kasus langka. Diketahui bahwa penelitian menggunakan IQR-SMOTE mampu menghasilkan kinerja yang cukup baik untuk mengatasi data outlier dan ketidakseimbangan kelas dimana kedua hal tersebut merupakan kelemahan dari metode K-Nearest Neighbors itu sendiri.

Penelitian ini menggunakan klasifikasi menggunakan K-Nearest Neighbors untuk memperbaiki kelemahannya yaitu sensitif terhadap data outlier dan kinerja KNN yang dipengaruhi oleh ketidakseimbangan kelas dengan melakukan oversampling SMOTE pada data outlier yang bertujuan untuk meningkatkan jumlah kasus yang langka dan menggunakan oversamplin SMOTE pada data keseluruhan untuk mengatasi ketidakseimbangan kelas untuk melihat kinerja dari model dalam mengklasifikasikan penyakit diabetes.

2. Metodologi Penelitian

Adapun alur penelitian yang dilakukan dapat direpresentasikan pada Gambar 1.



Gambar 1. Flowchart Alur Penelitian

p-ISSN: 1979-5661
e-ISSN: 2622-321X

2.1. Pengumpulan Dataset

Data diabetes yang digunakan untuk penelitian ini adalah Pima Indian Diabetes yang didapatkan dari situs kaggle.com yang memiliki sebanyak 8 fitur dan 1 kelas berupa diabetes dan bukan diabetes dan memiliki jumlah 268 kelas positif (diabetes) dan 500 kelas negatif (bukan diabetes) dan memiliki Imbalance Ratio sebesar 1,86.

2.2. Normalisasi Dataset

Melakukan normalisasi tiap fitur memiliki rentang yang sama sebesar 0 sampai 1 menggunakan MinMaxScaler agar tidak terjadi ketimpangan rentang fitur. Contohnya seperti fitur DiabetesPedigreeFunction dengan rentang 0,08 sampai dengan 2,42 yang sangat jauh rentangnya dengan fitur Glucose yang memiliki rentang 0 sampai 199. Penyesuaian rentang tiap fitur ini akan berguna bagi metode K-Nearest Neighbors untuk melakukan perhitungan jarak tiap data.

2.3. Pembagian Dataset

Pembagian dataset akan dilakukan menggunakan K-Fold Crossvalidation sebanyak 5 partisi yang akan membagi data dengan rasio data training 80% dan data testing 20% pada setiap iterasi.

2.4. Deteksi Outlier

Mengindentifikasi data outlier pada data training di setiap iterasi K-Fold Crossvalidation untuk menemukan kasus langka pada setiap iterasi yang akan dilakukan oversampling nantinya pada data outlier.

2.5. Oversampling SMOTE

Oversampling akan dilakukan untuk membuat data sintetis/buatan pada data training tiap iterasi K-Fold Crossvalidation. Pada penelitian ini terdapat 2 pendekatan dalam penggunakan oversampling SMOTE antara lain oversampling akan digunakan pada data outlier untuk meningkatkan kasus langka pada data training lalu kedua penggunaan oversampling pada keseluruhan data untuk mengatasi ketidakseimbangan kelas.

2.6. Klasifikasi K-Nearest Neighbors

Klasifikasi penyakit diabetes akan dilakukan menggunakan algoritma KNearest Neighbors. Terdapat 4 percobaan klasifikasi menggunakan K-Nearest Neighbors antara lain, Pertama klasifikasi hanya dengan menggunakan K-Nearest Neighbors tanpa melakukan oversampling apapun, kedua melakukan klasifikasi menggunakan K-Nearest Neighbors dengan penggunaan oversampling SMOTE pada keseluruhan data training tiap fold untuk mengatasi ketidakseimbangan kelas, ketiga melakukan klasifikasi dengan melakukan oversampling SMOTE hanya pada data outlier untuk menambah jumlah kasus langka pada data training dan terakhir melakukan klasifikasi dengan K-Nearest Neighbors dengan oversampling data outlier dan keseluruhan data.

2.7. Evaluasi

Melakukan evaluasi berupa akurasi, recall, presisi dan f1-score dengan menggunakan confusion matrix pada tiap-tiap percobaan klasifikasi menggunakan K-Nearest Neighbors dengan dan tanpa oversampling.

2.8. Analisa Perbedaan Performa Model

Melakukan perbandingan dari performa yang didapatkan dari setiap percobaan klasifikasi K-Nearest Neighbors seperti akurasi, recall, presisi dan f1-score. Pada penelitian ini akan digunakan evaluasi berupa f1-score untuk memilih mana model terbaik untuk mengklasifikasikan kelas positif (diabetes) pada dataset *Pima Indian Diabetes.*

p-ISSN: 1979-5661
e-ISSN: 2622-321X

3. Hasil dan Pembahasan

3.1. Hasil

a. Pengumpulan Dataset

Dataset akan diambil dari situs kaggle.com yang memiliki jumlah data sebanyak 768 data dengan 8 fitur dan 2 kelas berupa diabetes (positif) dan bukan diabetes (bukann diabetes) yang memiliki jumlah kelas diabetes sebanyak 268 dan kelas bukan diabetes sebanyak 500 data.

Tabel 1. Contoh Dataset Pima Indian Diabetes

No	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes Predigree Function	Age	Outcome
1	6	148	72	35	0	33.6	0.627	50	1
2	1	85	66	29	0	26.6	0.351	31	0
3	8	183	64	0	0	23.3	0.672	32	1
4	1	89	66	23	94	28.1	0.167	21	0
....	....	....	....	....	....	....	....	....	....
765	2	122	70	27	0	36.8	0.34	27	0
766	5	121	72	23	112	26.2	0.245	30	0
767	1	126	60	0	0	30.1	0.349	47	1
768	1	93	70	31	0	30.4	0.315	23	0

b. Normalisasi Data

Normalisasi dilakukan agar tiap fitur pada dataset memiliki rentang yang sama yaitu 0 sampai 1 menggunakan *MinMaxScaler* agar tidak memiliki ketimpangan rentang fitur. Pada Tabel 2 dapat dilihat data yang sudah ternormalisasi.

Tabel 2. Dataset Ternormalisasi

No	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes Predigree Function	Age	Outcome
1	0.353	0.744	0.590	0.354	0	0.501	0.234	0.483	1
2	0.059	0.427	0.541	0.293	0	0.396	0.117	0.167	0
3	0.471	0.920	0.525	0	0	0.347	0.254	0.183	1
4	0.059	0.447	0.541	0.232	0.111	0.419	0.038	0	0
....	....	....	....	....	....	....	....	....	....
765	0.118	0.613	0.574	0.273	0	0.548	0.112	0.1	0
766	0.294	0.608	0.590	0.232	0.132	0.390	0.071	0.15	0
767	0.059	0.633	0.492	0	0	0.449	0.116	0.433	1
768	0.059	0.467	0.574	0.313	0	0.453	0.101	0.033	0

c. Pembagian Data

Pembagian data akan dilakukan menggunakan K-Fold Crossvalidation sebanyak 5 partisi data yang akan dilakukan sebnayak 5 kali/iterasi. Pada setiap iterasi, satu partisi akan digunakan sebagai data testing (20%) dan sisanya akan digunakan sebagai data training (80%).

Iterasi	Jumlah Data				
	Partisi 1	Partisi 2	Partisi 3	Partisi 4	Partisi 5
1	153				
2		154			
3			153		
4				154	
5					154
	Testing	Training			

Iterasi	Data Prediksi
1	153
2	154
3	153
4	154
5	154
Jumlah	768

Gambar 2. Pembagian K-Fold Crossvalidation

p-ISSN: 1979-5661
e-ISSN: 2622-321X

d. Deteksi Outlier
Deteksi outlier akan dilakukan dengan menggunakan *InterquartileRange* yang akan dilakukan setiap iterasi/fold data training. Data outlier akan dideteksi per fitur dataset. Lalu menggabungkan keseluruhan data/baris yang memiliki data outlier pada fiturnya dan melakukan penghapusan data yang sama. Pada Tabel 3 dapat dilihat persebaran jumlah data outlier di setiap iterasi per fitur berserta total data sebelum dan sesudah dilakukan penghapusan data yang sama/redudant.

Tabel 3. Persebaran Data Outlier Setiap Iterasi

Fitur	Iterasi					Total
	1	2	3	4	5	
Fitur Pregnancies	4	4	2	2	4	16
Fitur Glucose	3	4	4	4	5	20
Fitur BloodPressure	42	38	38	35	38	191
Fitur SkinThickness	0	1	1	1	1	4
Fitur Insulin	25	26	25	31	22	129
Fitur BMI	17	17	12	12	18	76
Fitur DiabetesPedigreeFunction	23	26	22	24	21	116
Fitur Age	8	8	11	7	11	45
Total	**122**	**124**	**115**	**116**	**120**	**597**
Total Data Unik	**106**	**110**	**103**	**104**	**103**	**526**

Pada Tabel 3, dapat terlihat bahwa pada fitur *Pregnancies* dan *Glucose* terdapat sekitar 4 data outlier pada setiap iterasinya, fitur *BloodPressure* terdapat sekitar 38 data outlier pada setiap iterasinya, fitur *SkinThickness* terdapat sekitar 1 data outlier pada setiap iterasinya, fitur *Insulin* terdapat sekitar 25 data outlier pada setiap iterasinya, fitur *BMI* terdapat sekitar 17 data outlier pada setiap iterasinya, fitur *DiabetesPedigreeFunction* terdapat sekitar 24 data outlier pada setiap iterasinya dan fitur *Age* terdapat sekitar 8 data outlier pada setiap iterasinya.

Tabel 4. Persebaran Data Outlier Per Kelas

Jenis Data	Kelas	Iterasi				
		1	2	3	4	5
Outlier	Diabetes	58	56	53	57	50
	Non Diabetes	48	54	50	47	53
Jumlah Data		**106**	**110**	**103**	**104**	**103**
Non Outlier	Diabetes	157	158	162	157	164
	Non Diabetes	352	346	350	353	347
Jumlah Data		**509**	**504**	**512**	**510**	**511**
Total data		**615**	**614**	**615**	**614**	**614**

Dapat terlihat dari Tabel 4, bahwa jumlah data outlier pada setiap iterasi akan berbeda-beda dan pada kelas positif (diabetes) akan mendapatkan sekitar 58 data outlier dan 157 data non outlier. Lalu pada kelas negatif (non-diabetes) akan mendapatkan sekitar 48 data outlier dan sekitar 352 data non outlier.

e. Oversampling SMOTE
Oversampling menggunakan SMOTE akan dilakukan pada data training pada tiap-tiap iterasi K-Fold Crossvalidation dengan dua pendekatan yaitu oversampling pada data outlier dan oversampling pada keseluruhan data. Pertama melakukan oversampling pada keseluruhan data untuk mengatasi ketidakseimbangan kelas. Pada Tabel 5, dapat dilihat persebaran kelas sebelum melakukan oversampling pada keseluruhan data dan pada Tabel 6, dapat dilihat persebaran kelas setelah oversampling data keseluruhan. Terlihat bahwa setelah oversampling keseluruhan terdapat sekitar 400 data pada kelas positif dan negatif.

p-ISSN: 1979-5661
e-ISSN: 2622-321X

Tabel 5. Persebaran Kelas Sebelum Oversampling Keseluruhan Data

Kelas	Iterasi				
	1	2	3	4	5
Positif	215	214	214	214	214
Negatif	400	400	400	400	400
Jumlah	**615**	**614**	**615**	**614**	**614**

Tabel 6. Persebaran Kelas Setelah Oversampling Keseluruhan Data

Kelas	Iterasi				
	1	2	3	4	5
Positif	403	394	395	389	397
Negatif	400	400	400	400	400
Jumlah	**803**	**794**	**795**	**789**	**797**

Lalu pendekatan kedua yaitu dengan melakukan oversampling pada data outlier saja yang bertujuan untuk meningkatkan kasus langka yang dimiliki pada data. Oversampling pada data outlier ini akan ditingkatkan sebanyak 3x lipat. Pada Tabel 7, dapat dilihat persebaran data outlier per kelas sebelum melakukan oversampling dan pada Tabel 8, dapat dilihat persebaran data outlier per kelas setelah melakukan oversampling.

Tabel 7. Persebaran Kelas Outlier Sebelum Oversampling

Kelas	Iterasi				
	1	2	3	4	5
Positif	58	56	51	55	50
Negatif	48	54	50	48	53

Tabel 8. Persebaran Kelas Outlier Setelah Oversampling

Kelas	Iterasi				
	1	2	3	4	5
Positif	174	168	153	165	150
Negatif	144	162	150	144	159

Tabel 9. Persebaran Kelas Pada Data Training Setelah Oversampling Outlier

Jenis Data	Kelas	Iterasi				
		1	2	3	4	5
Outlier	Diabetes	174	168	153	165	150
	Non Diabetes	144	162	150	144	159
Jumlah Data		**318**	**330**	**303**	**309**	**309**
Non Outlier	Diabetes	157	158	162	157	164
	Non Diabetes	352	346	350	353	347
Jumlah Data		**509**	**504**	**512**	**510**	**511**
Total data		**827**	**834**	**817**	**820**	**820**

Dapat terlihat dari Tabel 9, bahwa setiap iterasi mendapatkan data outlier sekitar 309 dan non outlier sekitar 509 data. Pada Tabel 10 dapat dilihat dalam persebaran kelas positif dan negatif pada data outlier setiap iterasi K-Fold Crossvalidation.

p-ISSN: 1979-5661
e-ISSN: 2622-321X

Tabel 10. Persebaran Kelas Data training Setiap Iterasi Setelah Oversampling Outlier

Kelas	Iterasi				
	1	2	3	4	5
Positif	331	326	317	324	314
Negatif	496	508	500	496	506
Jumlah	**827**	**834**	**817**	**820**	**820**

Terlihat dari Tabel 10, bahwa setiap iterasi akan mendapatkan kelas positif sekitar 320 data dan kelas negatif akan mendapatkan sekitar 500 data dengan jumlah 820 data pada keseluruhan data. Lalu yang terakhir adalah dengan melakukan dua pendekatan yaitu melakukan oversampling pada data outlier dan juga keseluruhan data. Pada Tabel 10, terlihat persebaran kelas sebelum melakukan oversampling pada keseluruhan data tetapi sudah melakukan oversampling pada data outlier. Dapat dilihat pada Tabel 11 persebaran data setelah melakukan oversampling pada keseluruhan data.

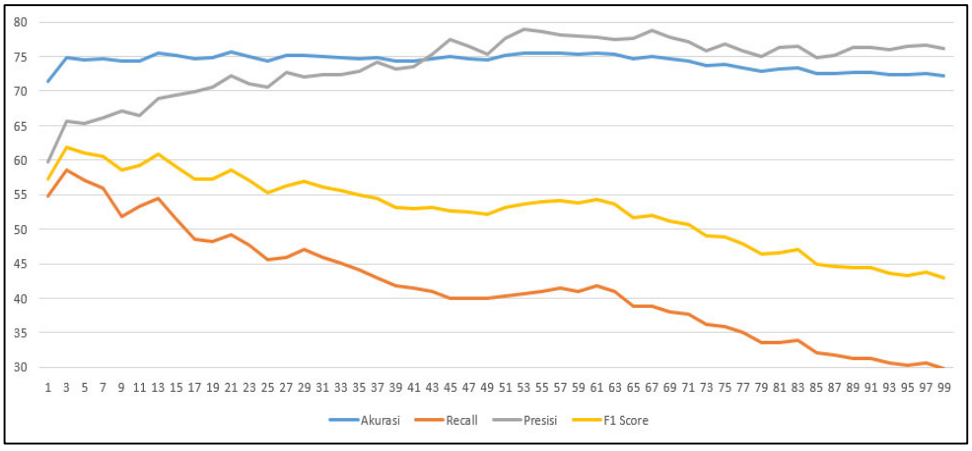
Tabel 11. Persebaran Kelas Setelah Oversampling Outlier dan Keseluruhan Data

Kelas	Iterasi				
	1	2	3	4	5
Positif	504	508	493	509	514
Negatif	496	508	500	496	506
Jumlah	**1000**	**1016**	**993**	**1005**	**1020**

Dapat terlihat dari Tabel 11, bahwa setiap iterasi akan mendapatkan sekitar 500 data pada kedua kelas dengan jumlah data sekitar 1000 pada data training.

f. Klasifikasi dan Evaluasi

Metode klasifikasi yang akan digunakan adalah metode KNN (K-Nearest Neighbors) untuk melakukan klasifikasi penyakit diabetes dengan dataset *Pima Indian Diabetes*. Pada penelitian ini akan dilakukan beberapa kombinasi teknik oversampling menggunakan SMOTE dan melakukan observasi K=1 sampai K=100 dengan bilangan ganjil pada metode K-Nearest Neighbors.



Gambar 3. Grafik Observasi K percobaan Pertama

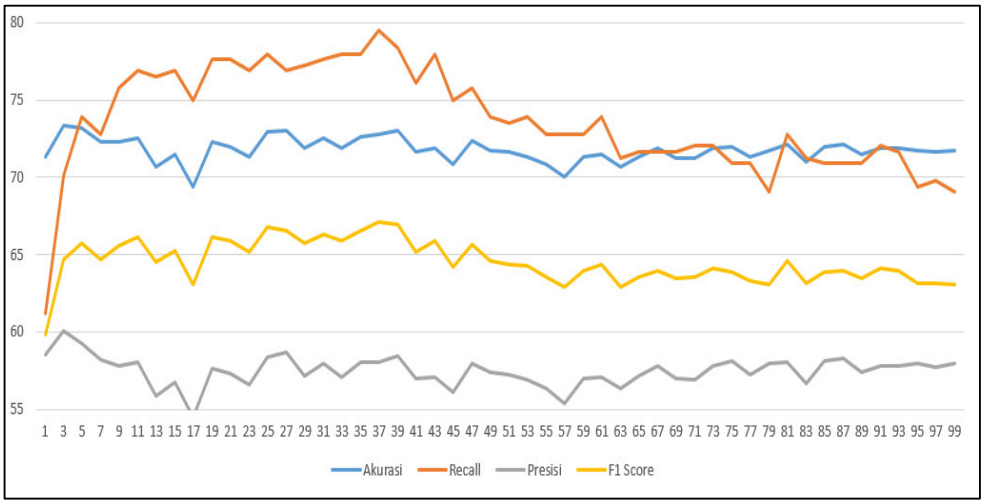
Dapat terlihat dari Gambar 3, bahwa performa f1-score akan terus mengalami penurunan seiring bertambahnya nilai K yang berarti semakin tinggi K maka model akan semakin tidak baik dalam mengklasifikasikan kelas positif dan pada penelitian pertama didapatkan 3 model terbaik yang terlihat pada Tabel 12.

p-ISSN: 1979-5661
e-ISSN: 2622-321X

Tabel 12. Performa 3 Terbaik Percobaan Pertama

No	K	Akurasi	Recall	Presisi	F1 Score
1	3	74,87 %	58,58 %	65,69 %	61,93 %
2	5	74,48 %	57,09 %	65,09 %	60,96 %
3	13	75.52 %	54,48 %	54,48 %	60,83 %

Dapat terlihat dari Tabel 12, bahwa performa tertinggi berupa f1-score untuk mendeteksi kelas positif yaitu pada K = 3. Selanjutnya percobaan kedua adalah dengan melakukan oversampling SMOTE pada keseluruhan data untuk mengatasi ketidakseimbangan kelas dan juga akan melakukan observasi K dari K =1 sampai K=100 dengan bilangan ganjil.



Gambar 4. Grafik Observasi K Percobaan Kedua

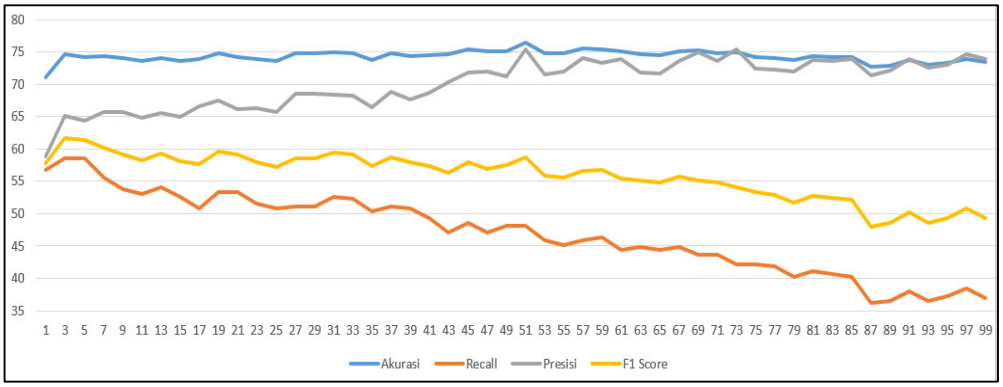
Dapat terlihat dari Gambar 4, bahwa performa f1-score akan terus mengalami peningkatan seiring bertambahnya nilai K sampai K = 37 dan akan mengalami penurunan setelahnya dan pada penelitian kedua didapatkan 3 model terbaik yang terlihat pada Tabel 13.

Tabel 13. Performa 3 Terbaik Percobaan Kedua

No	K	Akurasi	Recall	Presisi	F1 Score
1	37	72,79 %	79,48 %	58,04 %	67,09 %
2	39	73,05 %	78,36 %	58,50 %	66,99 %
3	25	72,92 %	77,99 %	58,38 %	66,77 %

Dapat terlihat dari Tabel 13, bahwa performa tertinggi berupa f1-score untuk mendeteksi kelas positif yaitu pada K = 37. Selanjutnya percobaan ketiga adalah dengan melakukan oversampling SMOTE pada data outlier untuk meningkatkan kasus langka pada data outlier dan juga akan melakukan observasi K dari K =1 sampai K=100 dengan bilangan ganjil.

p-ISSN: 1979-5661
e-ISSN: 2622-321X



Gambar 5. Grafik Observasi K Percobaan Ketiga

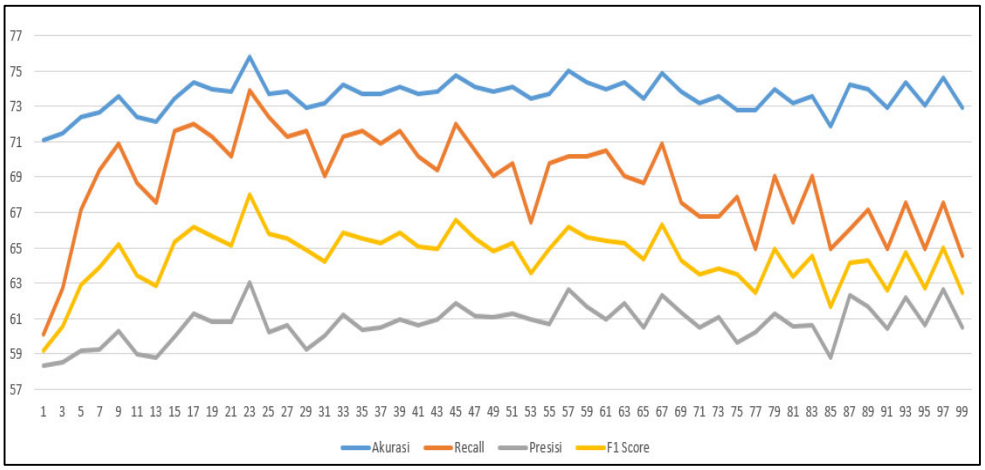
Dapat terlihat dari Gambar 5, bahwa performa f1-score akan terus mengalami penurunan seiring bertambahnya nilai K yang berarti semakin tinggi K maka model akan semakin baik dalam mengklasifikasikan kelas positif dan pada penelitian ketiga didapatkan 3 model terbaik yang terlihat pada Tabel 14.

Tabel 14. Performa 3 Terbaik Percobaan Ketiga

No	K	Akurasi	Recall	Presisi	F1 Score
1	3	74,61 %	58,58 %	65,15 %	61,69 %
2	5	74,22 %	58,58 %	64,34 %	61,33 %
3	7	74,35 %	55,60 %	65,64 %	60,20 %

Dapat terlihat dari Tabel 14, bahwa performa tertinggi berupa f1-score untuk mendeteksi kelas positif yaitu pada K = 3.

Selanjutnya percobaan keempat adalah dengan melakukan oversampling SMOTE pada data outlier dan keseluruhan data untuk meningkatkan kasus langka pada data outlier dan juga akan mengatasi ketidakseimangan kelas dengan melakukan observasi K dari K=1 sampai K=100 dengan bilangan ganjil.



Gambar 6. Grafik Observasi K Percobaan Keempat

Dapat terlihat dari Gambar 6, bahwa performa f1-score akan terus mengalami peningkatan seiring bertambahnya nilai K sampai K = 100 dengan bilangan ganjil dan akan mengalami penurunan setelahnya dan pada penelitian ketiga didapatkan 3 model terbaik yang dapat dilihat pada Tabel 15.

p-ISSN: 1979-5661
e-ISSN: 2622-321X

Tabel 15. Performa 3 Terbaik Percobaan Keempat

No	K	Akurasi	Recall	Presisi	F1 Score
1	23	75,78 %	73,88 %	63,06 %	68,04 %
2	45	74,74 %	72,01 %	61,86 %	66,55 %
3	67	74,87%	70,90 %	62,30 %	66,32 %

Dapat terlihat dari Tabel 15, bahwa performa tertinggi berupa f1-score untuk mendeteksi kelas positif yaitu pada K = 23.

g. Perbandingan Performa

Membandingkan performa dari setiap percobaan K-Nearest Neighbors tanpa dan dengan menggunakan oversampling SMOTE berupa akurasi, recall, presisi dan f1-score untuk mengetahui model terbaik untuk mengklasifikasikan penyakit diabetes. Pada Tabel ***16***, dapat terlihat perbandingan performa dari model terbaik setiap percobaan.

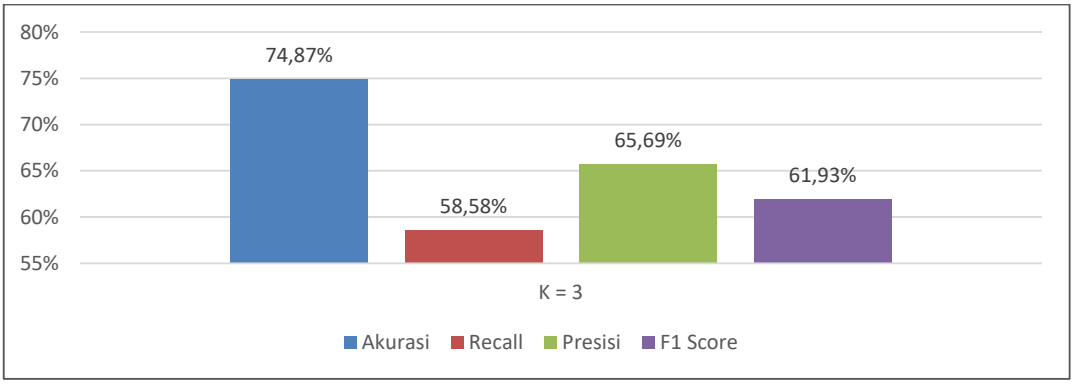
Tabel 16. Perbandingan Model Terbaik Setiap Percobaan

Model	K	Akurasi	Recall	Presisi	F1 Score
KNN	3	74,87%	58,58%	**65,69%**	61,93%
KNN + SMOTE	37	72,79%	**79,48%**	58,04%	67,09%
KNN Oversampling Outlier	3	74,61%	58,58%	65,15%	61,69%
KNN + IQR-SMOTE	23	**75,78%**	73,88%	63,06%	**68,04%**

Pada Tabel ***16***, dapat terlihat bahwa model terbaik untuk mengklasifikasikan penyakit diabetes berupa f1-score adalah pada model keempat yaitu KNN + IQR-SMOTE sebesar 68,04 %.

3.2. Pembahasan

Hasil kinerja K-Nearest Neigbors tanpa menggunakan oversampling dihasilkan akurasi sebesar 74,87%, recall 58,58%, presisi sbesar 65,69% dan f1-score 61,93% dengan pembagian dataset menjadi training dan testing menggunakan K-fold Crossvalidation k=5 yang akan membagi data menjadi 5 subset data. Satu bagian dari 5 subset data ini akan dijadikan data testing bergantian setiap iterasi yang mendapatkan data testing sekitar sebanyak 76 data. Pada klasifikasi percobaan pertama ini didapatkan performa terbaik untuk mengklasifikasikan kelas positf berupa f1-score yaitu pada K = 3. Grafik performa percobaan pertama dapat dilihat pada Gambar 7.

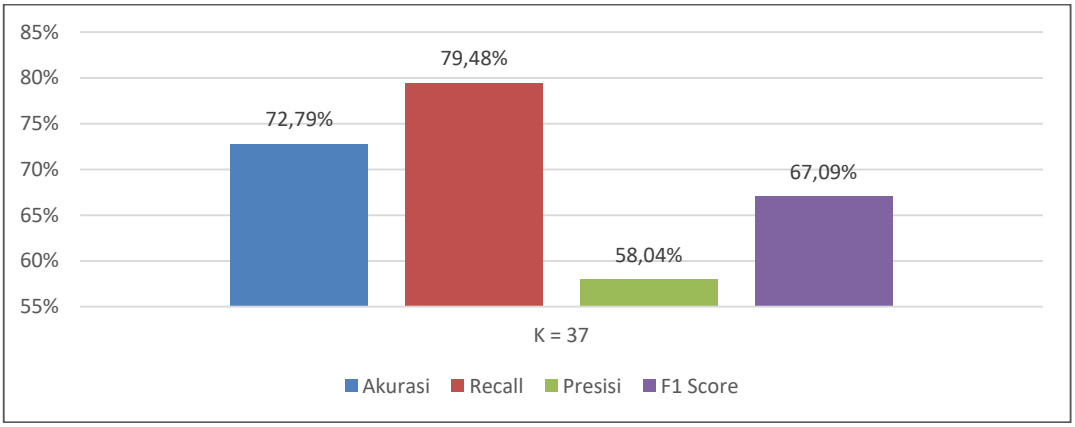


Gambar 7. Grafik Perbandingan Performa Percobaan Pertama

Pada percobaan kedua, kinerja K-Nearest Neighbors menggunakan oversampling SMOTE keseluruhan data training untuk mengatasi ketidakseimbangan kelas mendapatkan performa berupa akurasi 72,79%, recall 79,48%, presisi 58,04%, dan f1-score 67,09% dengan pembagian

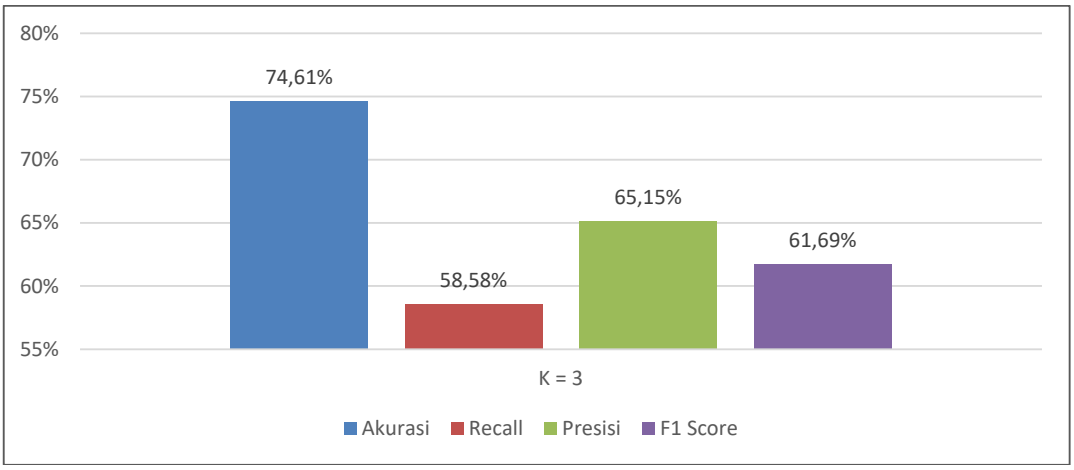
p-ISSN: 1979-5661
e-ISSN: 2622-321X

dataset menjadi training dan testing menggunakan K-fold Crossvalidation k=5 yang membagi data menjadi 5 subset data. Satu bagian dari 5 subset data akan dijadikan data testing bergantian setiap iterasi yang mendapatkan data testing sekitar sebanyak 76 data. Pada klasifikasi percobaan kedua ini didapatkan performa terbaik untuk mengklasifikasikan kelas positf berupa f1-score yaitu pada K = 37. Grafik perbandingan percobaan kedua dapat dilihat pada Gambar 8.



Gambar 8. Grafik Perbandingan Performa Percobaan Kedua

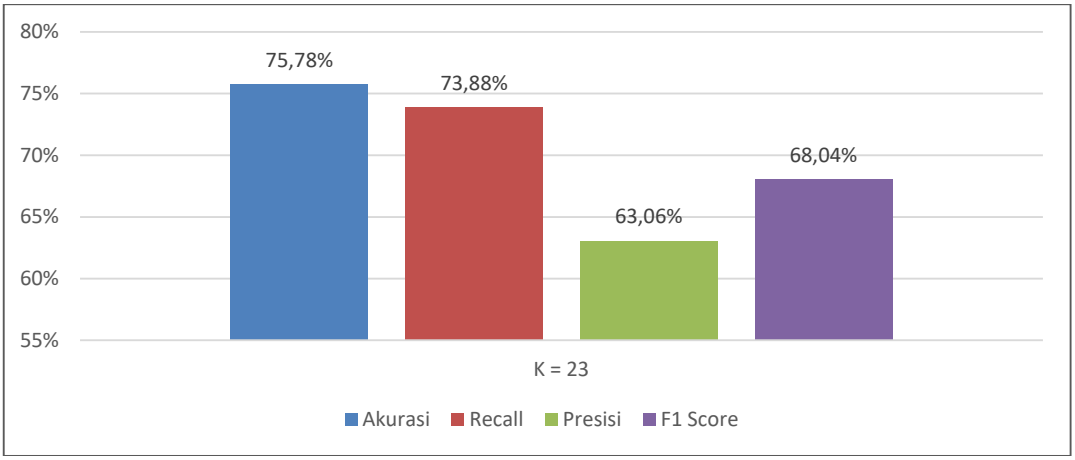
Pada percobaan ketiga, kinerja K-Nearest Neighbors dengan menggunakan oversampling SMOTE data outlier untuk meningkatkan kasus langka akan mendapatkan performa berupa akurasi 74,61%, recall, 58,58%, presisi 65,15%, dan f1-score 61,69% dengan pembagian dataset menjadi training dan testing menggunakan K-fold Crossvalidation k=5 yang akan membagi data menjadi 5 subset data. Satu bagian dari 5 subset data ini akan dijadikan data testing bergantian setiap iterasi yang mendapatkan data testing sekitar sebanyak 76 data. Pada klasifikasi percobaan kedua ini didapatkan performa terbaik untuk mengklasifikasikan kelas positf berupa f1-score yaitu pada K = 3. Grafik perbandingan percobaan ketiga dapat dilihat pada Gambar 9.



Gambar 9. Grafik Perbandingan Performa Percobaan Ketiga

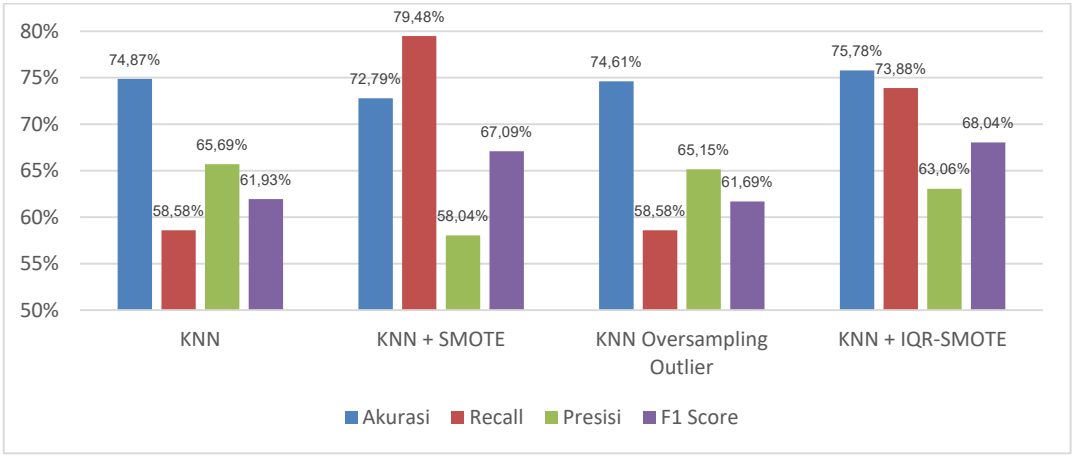
Pada percobaan terakhir, kinerja klasifikasi K-Nearest Neighbors dengan menggunakan oversampling SMOTE pada keseluruhan data training serta pada data outlier untuk mengatasi ketidakseimbangan kelas serta meningkatkan kasus langka akan mendapatkan performa berupa akurasi sebesar 75,78%, recall, sebesar 73,88%, presisi sebesar 63,06%, dan f1-score sebesar 68,04% dengan pembagian dataset menjadi training dan testing menggunakan K-fold Crossvalidation sebanyak k=5 yang akan membagi data menjadi 5 subset data. Satu bagian dari 5 subset data ini akan dijadikan data testing bergantian setiap iterasi yang mendapatkan data testing sekitar sebanyak 76 data. Pada klasifikasi percobaan kedua ini didapatkan performa terbaik untuk mengklasifikasikan kelas positf berupa f1-score yaitu pada K = 23. Grafik perbandingan percobaan keempat dapat dilihat pada Gambar 10.

p-ISSN: 1979-5661
e-ISSN: 2622-321X



Gambar 10. Grafik Perbandingan Performa Percobaan Keempat

Dari semua percobaan klasifikasi menggunakan K-Nearest Neighbors tanpa dan dengan menggunakan oversampling SMOTE didapatkan akurasi dan f1-score tertinggi yaitu pada model K-Nearest Neighbors dengan melakukan oversampling pada keseluruhan data dan data outlier (KNN + IQR-SMOTE) dengan akurasi sebesar 75,78% dan f1-score 68,04%. Lalu performa recall tertinggi didapatkan pada model K-Nearest Neighbors dengan melakukan oversampling pada keseluruhan data (KNN + SMOTE) sebesar 79,48%. Kemudian performa presisi tertinggi didapatkan pada model K-nearest Neighbors tanpa melakukan oversampling sebesar 65,69%. Jika melihat dari performa model untuk mengklasifikasikan kelas positif berupa f1-score didapatkan model terbaik yaitu pada K-Nearest Neighbors dengan melakukan oversampling pada keseluruhan data untuk mengatasi ketidakseimbangan kelas dan oversampling pada data outlier untuk meningkatkan kasus langka pada dataset yang mendapatkan f1-score sebesar 68,04%. Grafik perbandingan performa keseluruhan model dapat dilihat pada Gambar 11.



Gambar 11. Grafik Perbandingan Performa Keseluruhan Model

Pada penelitian sebelumnya meneliti mengenai penggunaan IQR-SMOTE pada C4.5 dan berhasil meningkatkan performa f1-score dari 74,4 menjadi 89,5% yang terbilang cukup tinggi [8]. Tetapi pada penelitian ini, IQR-SMOTE tidak meningkatkan performa dari f1-score secara signifikan dari 61,9% menjadi 68,04%. Hal ini kemungkinan terjadi karena pada oversampling tidak akan terlalu berpengaruh pada algoritma yang berbasis ketetanggaan yaitu K-Nearest Neighbors karena pada model ini hanya mengklasifikasikan berdasarkan tetangga terdekatnya. Tetapi oversampling sangat mempengaruhi model algoritma yang memiliki konsep berbasis pohon/tree seperti C4.5 karena pada konsep pohon ini akan mempertimbangkan keseluruhan

data. Oleh karena itu model yang berbasis tree ini sangat berpengaruh jika ditambahkan oversampling pada data.

4. Kesimpulan

Klasifikasi menggunakan metode K-Nearest Neighbors tanpa menggunakan oversampling akan mendapatkan performa akurasi sebesar 74,87%, recall sebesar 58,58%, presisi sebesar 65,69% dan f1-score sebesar 61,93%. Kemudian kalsifikasi K-Nearest Neighbors dengan menggunakan oversampling pada keseluruhan data untuk mengatasi ketidakseimbangan kelas dan oversampling pada data outlier untuk meningkatkan kasus langka mendapatkan akurasi sebesar 75,78%, recall sebesar 73,88%, presisi sebesar 63,06% dan f1-score sebesar 68,04%. Sehingga dalam penelitian ini didapatkan pengaruh oversampling pada keseluruhan data dan data outlier menggunakan SMOTE dapat meningkatkan performa f1-score dari metode K-Nearest Neighbors tanpa menggunakan oversampling SMOTE dengan peningkatan sebesar 6,11%.

DAFTAR PUSTAKA

[1] R. A. Nugroho, Tarno and A. Prahutama, “Klasifikasi Pasien Diabetes mellitus Menggunakan Metode *SMOOTH Support Vector Machine (SSVM)*” Jurnal Gaussian, vol. 6, no. 3, pp. 439-448, 2017.

[2] F. Nasution, Andilala and A. Z. Siregar, “Faktor Kejadian Diabetes Mellitus” Jurnal Ilmu Kesehatan, vol. 9, no. 2, pp. 94-102, 2021.

[3] M. A. Banjarsari, H. I. Budiman and A. Farmadi, “Penerapan K-Optimal Pada Algoritma KNN Untuk Prediksi Kelulusan Tepat waktu Mahasiswa Program Studi Ilmu Komputer FMIPA Unlam Berdasarkan IP Sampai Dengan Semester 4” Jurnal KLIK (Kumpulan Jurnal Ilmu Komputer), vol. 2, no. 2, pp. 50-64, 2015.

[4] Suyanto, Machine Learning Tingkat Dasar dan Lanjut, City: Bandung, 2018, pp. 133.

[5] A. G. Pertiwi, N. Bachtiar, R. Kusumaningrum, I. Waspada and A. Wibowo, “Comparison of Performance of K-Nearest Neighbor Algorithm Using Smote and K-Nearest Neighbor Algorithm Without Smote in Diagnosis of Diabetes Disease in Balance Data”, Journal of Physics: Conference Series, vol. 1524, no.012048, pp. 1-8, 2020, doi:10.1088/1742-6596/1524/1/012048.

[6] M. A. Mahfouz, A. Shoukry and M. A. Ismail, “EKNN: Ensemble Classifier Incorporating Connectivity and Density Into KNN with Application to Cancer Diagnosis” Journal Artificial Intelligence in Medicine, vol. 111, no. xxxx, 2021, doi: 10.1016/j.artmed.2020.101985.

[7] R. Siringoringo, “Klasifikasi Data Tidak Seimbang Menggunakan Algoritma SMOTE dan K-Nearest Neighbor” Jurnal ISD, vol.3 no.1, pp. 44-49, 2018.

[8] N. Nnamoko and I. Korkontzelos, “Efficient Treatment of Outliers and Class Imbalance for Diabetes Prediction” Artificial Intelligence In Medicine, vol. 104, no. 101815, pp. 1-12, 2020.