Optimasi Algoritma Klasifikasi Biner dengan *Tuning* Parameter pada Penyakit *Diabetes Mellitus*

Wijanarto[1], Rhatna Puspitasari[2]
Teknik Informatika
Universitas Dian Nuswantoro
Semarang, Indonesia
e-mail: [1]wijanarto@dsn.dinus.ac.id, [2]rhatnapuspitsari97@gmail.com

Diajukan: 18 Juni 2019; Direvisi: 14 Agustus 2019; Diterima: 15 Agustus 2019

Abstrak

Diabetes Melitus atau yang biasanya disebut dengan penyakit kencing manis merupakan penyakit yang terjadi akibat peningkatan kadar glukosa di dalam darah terlalu tinggi. Data World Health Organization (WHO), menunjukkan Indonesia menjadi negara keempat di dunia dengan angka penderita diabetes terbanyak dan mengalami peningkatan hingga 14 juta orang. Peningkatan kasus penyakit Diabetes melitus ini memerlukan suatu upaya penanggulangan dan pencegahan dini terhadap penyakit Diabetes melitus. Dalam penelitian ini akan dilakukan optimasi algoritma klasifikasi biner pada penyakit diabetes melitus mulai dari observasi, visualisasi, statistic deskriptif dataset, pre-processing dataset, penentuan baseline model, tuning parameter model dan finalisasi model. Penentuan baseline model diperoleh dengan mencari nilai akurasi tertinggi dari 3 algoritma linear (Logistic Regression, Linear Discriminant Analysis, K-nearest neighbor) atau 3 algoritma non- linear (Decision Tree, Naïve Bayes, Support Vector Machine) berdasarkan tuning parameternya dan yang menghasilkan akurasi optimal adalah Algoritma Support Vector Machine, sehingga dijadikan sebagai final model dengan parameter C sebesar 47 dengan kernel rbf dihasilkan rerata akurasi sebesar 77.3% pada data training dan 74.5% pada data testing, sementara berdasarkan confusion matrix dihasilkan precision 78%, recall 83%, f1-Score 81%, error rate 25%.

Kata kunci: *Optimasi, Klasifikasi Biner, Tuning Parameter, Support Vector Machine.*

Abstract

Diabetes mellitus or commonly referred to as diabetes is a disease that occurs due to an increase in blood glucose levels too high. Data from the World Health Organization (WHO) shows that Indonesia is the fourth country in the world with the highest number of diabetics and has increased to 14 million people. This increase in cases of diabetes mellitus requires an early prevention and prevention of diabetes mellitus. In this study, optimization of binary classification algorithms in Diabetes Mellitus will be carried out starting from observation, visualization, dataset descriptive statistics, pre-processing datasets, determining baseline models, tuning model parameters and finalizing models. Determination of the baseline model is obtained by finding the highest accuracy value of 3 linear algorithms (Logistic Regression, Linear Discriminant Analysis, K-nearest neighbor) or 3 non-linear algorithms (Decision Tree, Naïve Bayes, Support Vector Machine) based on tuning parameters and resulting accuracy optimal is the Support Vector Machine Algorithm, so it is used as a final model with parameter C of 47 with the rbf kernel resulting in an average accuracy of 77.3% in training data and 74.5% in testing data, while based on confusion matrix precision is 78%, recall 83%, f1 -Score 81%, 25% error rate.

Keywords*: Optimization, Binary Classification, Tuning Parameters, Support Vector Machine.*

1. Pendahuluan

Diabetes Melitus atau yang biasanya disebut dengan penyakit kencing manis merupakan penyakit yang terjadi akibat peningkatan kadar glukosa di dalam darah terlalu tinggi. Kadar gula di dalam darah dapat meningkat salah satunya disebabkan karena tubuh tidak dapat melepaskan atau menggunakan insulin secara normal. Setiap individu memiliki kadar glukosa yang bervariasi, kadar glukosa ini akan meningkat setelah makan kemudian akan kembali normal dalam waktu dua jam [1]. Menurut *World Health Organization* (WHO), Indonesia menjadi negara keempat di dunia yang memiliki angka penderita diabetes

 DOI: 10.30864/eksplora.v9i1.257

terbanyak dan mengalami peningkatan hingga 14 juta orang. Di mana jumlah penderita diabetes di Indonesia pada tahun 2000 adalah 8,4 juta orang setelah India (31,7 juta), Cina (20,8 juta) dan Amerika Serikat (17,7 juta). Untuk penderita diabetes di seluruh dunia, WHO melaporkan terdapat lebih dari 143 juta orang penderita, dan jumlah ini diproyeksikan prevalensinya akan meningkat menjadi dua kali lipat pada tahun 2030 dan sebanyak 77% di antaranya terjadi di negara berkembang. Selain di tingkat dunia, prevalensi diabetes melitus di Indonesia juga akan meningkat. Salah satu daerah di Indonesia yang memiliki banyaknya penderita diabetes melitus adalah Kabupaten Grobogan di mana data sampel diambil untuk penelitian ini. Diabetes melitus menjadi satu dari lima penyakit tidak menular yang paling banyak diderita di Kabupaten Grobogan. Kelima penyakit tersebut antara lain penyakit hipertensi dengan jumlah kasus yaitu (15.587 kasus), asma (6.344 kasus), diabetes melitus (4.297 kasus), gagal jantung (944 kasus) dan penyakit paru obstruksi kronik (459 kasus) [2].

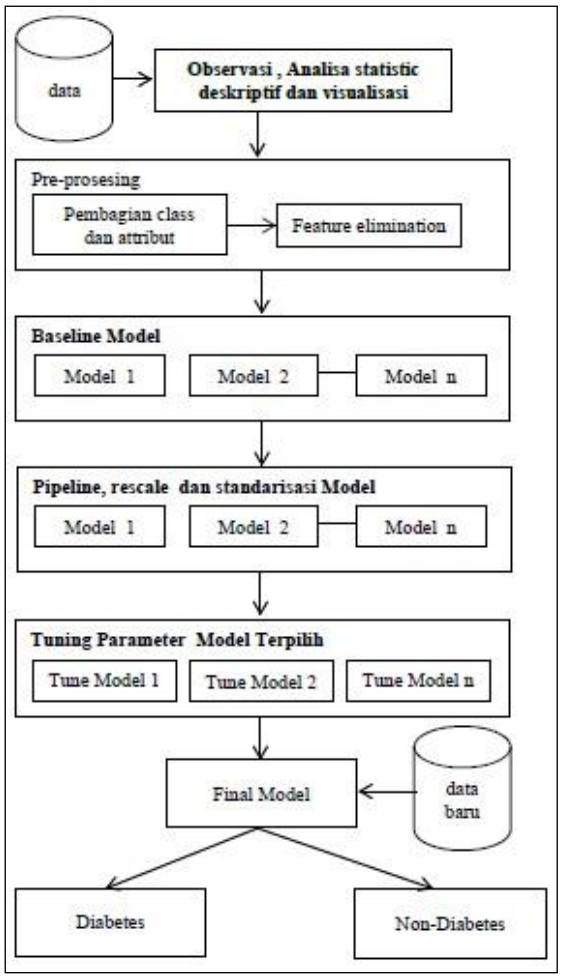
Melihat kondisi tersebut maka pencegahan dan penanganan secara dini terhadap penyakit diabetes melitus menjadi sangat penting untuk dilakukan dalam rangka membantu mengklasifikasikan jumlah penderita penyakit diabetes melitus. Penelitian terkait untuk klasifikasi terhadap diabetes melitus telah dilakukan dengan menggunakan model Naïve Bayes [3] menghasilkan akurasi sebesar 80%. Demikian juga dalam [1], yang mencoba membandingkan teknik C 4.5 dengan Naïve Bayes yang menghasilkan akurasi sebesar 73.30% , nilai AUC sebesar 0.733 pada algoritma C4.5, sedangkan Naïve Bayes menghasilkan akurasi sebesar 75.13% dan nilai AUC adalah 0.810. Walaupun kelebihan metode Naïve Bayes sangat sederhana, efisien dan hanya memerlukan komputasi matematika yang tidak terlalu kompleks [4], namun kelemahan teknik ini yaitu memerlukan pengetahuan awal untuk mengambil suatu keputusan, tingkat keberhasilan metode ini sangat tergantung pada pengetahuan awal yang diberikan [5]. Sedangkan C4.5 dapat mengolah data secara berkesinambungan dengan proses yang lebih cepat dan menghasilkan pohon keputusan yang menggambarkan aturan agar lebih mudah untuk dipahami dan diimplementasikan, sedangkan kelemahan yang dimiliki adalah apabila kelas dan kriteria yang digunakan terlalu banyak maka akan terjadi tumpang tindih yang mengakibatkan bertambahnya waktu dalam pengambilan keputusan dan jumlah memori yang diperlukan [6]. Dalam penelitian lainnya [7], dengan menggunakan metode *K-Nearest Neighbor* didapatkan hasil yang bagus dan tangguh terhadap *training* data yang *noisy* dan efektif apabila data latihnya besar, tetapi terdapat kelemahan dalam menentukan parameter K (jumlah dari tetangga terdekat) yang terbaik atribut mana yang harus digunakan untuk mendapatkan hasil yang terbaik. Sementara dalam [8], melakukan uji terhadap beberapa algoritma prediksi, *Logistic Regression* (LR), *Linear Discriminant Analysis* (LDA), *Quadratic Discriminant Analysis* (QDA), dan *K-Nearest Neighbor* (KNN), semua algoritma menghasilkan standar deviasi (AUC) di bawah 0.01 dengan 10-*fold cross validation*, tetapi yang menghasilkan performa terbaiknya adalah LR dengan AUC 78%. Penelitian yang dilakukan oleh [9] meng-*ensemble* teknik *multi layer perceptron* dan *Bayesian Net Classification* untuk data *diabetes mellitus* menghasilkan akurasi 81.89%, dengan 6 fitur dan 786 *dataset*. Teknik *ensemble* lainnya (*AdaBoost*, *Gradient Boosted Trees*, dan *Random Forest*) dalam [10], menghasilkan akurasi 73.88% yang diaplikasikan pada *dataset Pima Indian*. Dalam [11], Empat model klasifikasi yang terkenal yaitu, *Decision Tree*, Jaringan Syaraf Tiruan, Regresi Logistik, dan Naïve Bayes kemudian dengan teknik *Bagging* dan *Boosting* dilakukan untuk mendapatkan hasil model yang paling kuat, lalu dilakukan perbandingan dengan *Random Forest* yang merupakan kombinasi dari *Bagging* dan *Decision Tree* dan hasilnya optimalnya adalah *Random Forest* dengan akurasi sebesar 85.558%. Penelitian lainnya [12], menyajikan studi dan analisis dari empat algoritma klasifikasi yaitu J48, *Random Tree*, *Decision Tree* dan Naïve Bayes untuk *Diabetic dataset* dan kinerjanya dibandingkan dengan menggunakan ukuran seperti waktu komputasi, contoh yang diklasifikasikan dengan benar dan salah, statistik kappa, Presisi, *Recall* dan F-Score. Hasil percobaan menunjukkan bahwa J48 memberikan akurasi yang lebih baik daripada *decision tree* dan Naïve Bayes. *Boosting* model juga dilakukan dalam [13], pengujian statistik non-parametrik dilakukan pada ratusan hasil indeks pengukuran medis antara populasi diabetes dan non-diabetes. Dua algoritma peningkatan yang umum, *Adaboost.M1* dan *LogitBoost*, dipilih untuk membuat model mesin untuk diagnosis diabetes berdasarkan data uji klinis ini, yang melibatkan total 35.669 individu. Model klasifikasi mesin yang dibangun oleh kedua algoritma ini memiliki klasifikasi yang sangat baik kemampuannya. Di sini, model klasifikasi *LogitBoost* sedikit lebih baik daripada model klasifikasi *Adaboost.M1*. Keakuratan keseluruhan dari model klasifikasi *LogitBoost* mencapai 95,30% ketika menggunakan validasi silang 10 kali lipat. Nilai *true positive*, *true negative*, *false positive*, dan *false negative* dari model klasifikasi biner masing-masing adalah 0,921, 0,969, 0,031, dan 0,079, dan area di bawah kurva karakteristik operasi penerima mencapai 0,99.

Berdasarkan beberapa penelitian terdahulu yang terkait, penelitian ini mencoba mencari metode optimal pada data kecil untuk mengklasifikasikan penyakit *diabetes mellitus* berdasarkan 3 algoritma linear dan 3 algoritma non-linear yang dijadikan sebagai *baseline* model untuk dilakukan *tuning* parameter

terhadapnya sehingga menghasilkan model yang optimal yang dapat menentukan klasifikasi penyakit diabetes atau non-diabetes.

2. Metode Penelitian

Dalam menentukan algoritma atau model *machine learning* yang terbaik sangatlah sulit terutama pada *supervised learning*, karena sangat terbuka kemungkinan model yang terbaik tergantung pada parameter yang diberikannya. Dalam penelitian ini akan menggunakan kerangka atau *framework* penelitian seperti pada Gambar 1 di bawah ini.



Gambar 1. Kerangka kerja penelitian.

2.1. *Dataset*

Dataset yang digunakan adalah *dataset* penyakit diabetes melitus dari RSUD. Dr. R Soedjati Purwodadi, yang terdiri dari 271 data *record*, yang terdiri dari 160 data diklasifikasikan sebagai penderita diabetes dan 111 data sebagai non-diabetes, juga terdapat 8 atribut, 7 atribut dengan 1 *class*, seperti Tabe1 1 di bawah ini.

Tabel 1. *Feature dataset.*

ID Feature	Nama Feature
X1	Usia
X2	Gender
X3	Gula Darah Sesaat
X4	Tekanan Darah
X5	Riwayat DM
X6	*Alkoholic*
X7	Perokok
TARGET	Diabetes (1) atau Non Diabetes(0)

2.2. Observasi, *Statistic* Deskriptif, dan Visualisasi

Kerangka kerja penelitian ini pertama-tama akan dilakukan observasi dengan *statistic* deskriptif, untuk mendapatkan gambaran mengenai data yang akan dilakukan pemrosesan awal (jumlah, tipe, unit, skala) dan *statistic* deskriptif (jumlah atribut, nilai *maximum*, minimum, rerata, deviasi, Q1,Q2,Q3) dengan pustaka ***pandas***. Visualisasi data akan dilakukan dengan ***matplotlib*** dan ***seaborn*** yang terdiri dari analisis *univariate* untuk melihat *histogram, density, distribusi cumulative*, analisis *bivariate* dengan *box* dan *whisker* untuk melihat sebaran Q1,Q2,Q3,Q4 dan *outlier dataset*, analisis *multivariate* dengan *scatter* plot *matrix* untuk melihat sebaran data dan terakhir analisis korelasi *matrix* pada atribut untuk melihat korelasi antar atribut sebelum dilakukan *feature* eliminasi dan meyakinkan apakah data perlu dilakukan pemrosesan awal yang lebih baik.

2.3. *Pre-prosesing*: Pembagian Data dan Eliminasi Fitur

Setelah didapatkan observasi awal maka langkah selanjutnya adalah melakukan pemrosesan data awal yaitu melakukan pembagian data (*class* dan atribut), *k-fold validation* (*cross validation* dipilih 10-*fold validation*) dan feature *elimination* dengan pustaka ***sklearn*** dengan mengaplikasikan metode *chi-square, feature classification, mutual information classification*. Ketiga metode tadi diaplikasikan pada fungsi *SelectKBest* untuk mendapatkan *k* feature terpilih dari pustaka *sklearn*, dalam klasifikasi biner akurasi akan lebih meningkat jika atribut lebih sederhana dan nilai data lebih homogen.

2.4. Baseline Model: Best Feature Selection

Langkah selanjutnya adalah menentukan *baseline* model atau algoritma dari 3 algoritma linier (*Logistic Regression*, *Linear Discriminant Analysis*, *K-Nearest Neighbors*) dan 3 algoritma non-linear (*Decision Tree*, *Gaussian Naïve Bayes*, *Support Vector Machine*) berdasarkan metode *feature eliminiation* yang terbaik (*chi-square, feature classification*, atau *mutual information classification*). Berdasarkan *baseline* ini maka dapat ditentukan model terbaik untuk *feature* terpilih dengan akurasi tertinggi.

2.5. *Pipelining*: *Rescale* dan *Standardization*

Setelah didapatkan *baseline* model yang dicurigai menghasilkan nilai terbaik, maka langkah selanjutnya adalah melakukan penyekalaan ulang dan standardisasi data dan membandingkannya dengan baseline model dengan feature terpilih, sehingga dari hasil perbandingan tersebut didapatkan model terbaik dari *dataset* yang sudah distandardisasi dengan fitur terpilih, yang mungkin lebih dari satu model yang dicurigai menghasilkan akurasi yang bagus. Langkah ini mengaplikasikan fungsi ***Pipeline*** dari pustaka *sklearn* dengan perhitungan ***StandardScaler*** seperti persamaan 1 berikut di bawah:

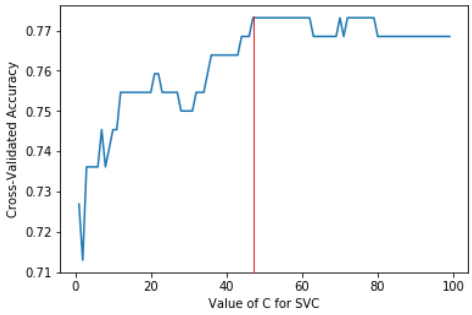
$$\overline{X} = \frac{X_i - mean(x)}{SD(x)} \tag{1}$$

X_i = nilai variabel *input* data mentah X untuk kasus pelatihan ke-*i*
Mean(X) = rata-rata vektor fitur tersebut
SD(X) = nilai standar yang berhubungan dengan X_i

Dua cara paling berguna untuk menstandardisasi *input* adalah *mean* 0 dan standar deviasi 1, serta *midrange* 0 dan *range* 2 (mis., minimum -1 dan maksimum 1). Perhatikan bahwa statistik seperti *mean* dan standar deviasi dihitung dari data pelatihan, bukan dari validasi atau data uji. Validasi dan data uji harus distandardisasi menggunakan statistik yang dihitung dari data pelatihan.

2.6. *Tuning Parameter*

Tuning parameter dilakukan pada beberapa model yang dihasilkan setelah dilakukan *pipelining*. Tuning parameter sesuai dengan model yang terpilih dengan memberi nilai yang optimal (ditunjukkan dengan ***grafik siku/elbow graphic***, titik perubahan (garis merah) merupakan nilai optimal dari *list* parameter yang diberikan pada Gambar 2) sesuai parameter algoritma, nilai selanjutnya cenderung sama atau bahkan menurun.



Gambar 2. *Elbow Graph*.

2.7. Finalisasi Model

Hasil akhir setelah *tuning* parameter adalah model terpilih yang terakhir (Final Model), model ini adalah yang dihasilkan yang terbaik berdasarkan parameter yang optimal (berdasarkan grafik siku). Dari final model ini akan diuji performanya dengan ***confusion matrix*** untuk mendapatkan *Acuracy*, *Presision*, *Recall* dan *F1-Score* serta diuji dengan data baru untuk menghasilkan klasifikasi biner yang diinginkan apakah *diabetes* atau *non-diabetes*, seperti Tabel 2 di bawah ini.

Tabel 2. *Confussion Matrix*.

Aktual vs Prediksi	**Positive**	**Negative**	**Precission**
Positive	TP	FP	TP/(TP+FP)
Negative	FN	TN	TP/(TP+FP)
Recall	TP/(TP+FN)	TP/(TP+FN)	

Kinerja model dapat dievaluasi berbagai kinerja langkah-langkah: akurasi klasifikasi, sensitivitas, dan spesifisitas. Langkah-langkah ini dievaluasi menggunakan *true positive* (TP), *true* negatif (TN), *false positive* (FP), dan *false negative* (FN). Nilai Akurasi didapatkan dengan (TP+TN)/(TP+TN+FP+FN), sementara F1-Score adalah 2.TP/2.TP+FP+FN. Aplikasi dari kerangka kerja penelitian diatas menggunakan pustaka *machine learning* (***numpy, pandas, matplotlib, seaborn, pickle, sklearn***) dari bahasa Python 3.7.

3. Hasil dan Pembahasan

Berdasarkan kerangka kerja penelitian yang diusulkan pada bagian sebelumnya, maka didapatkan hasil sebagai berikut di bawah ini.

3.1. Observasi dan Visualisasi Data Statistik Deskriptif

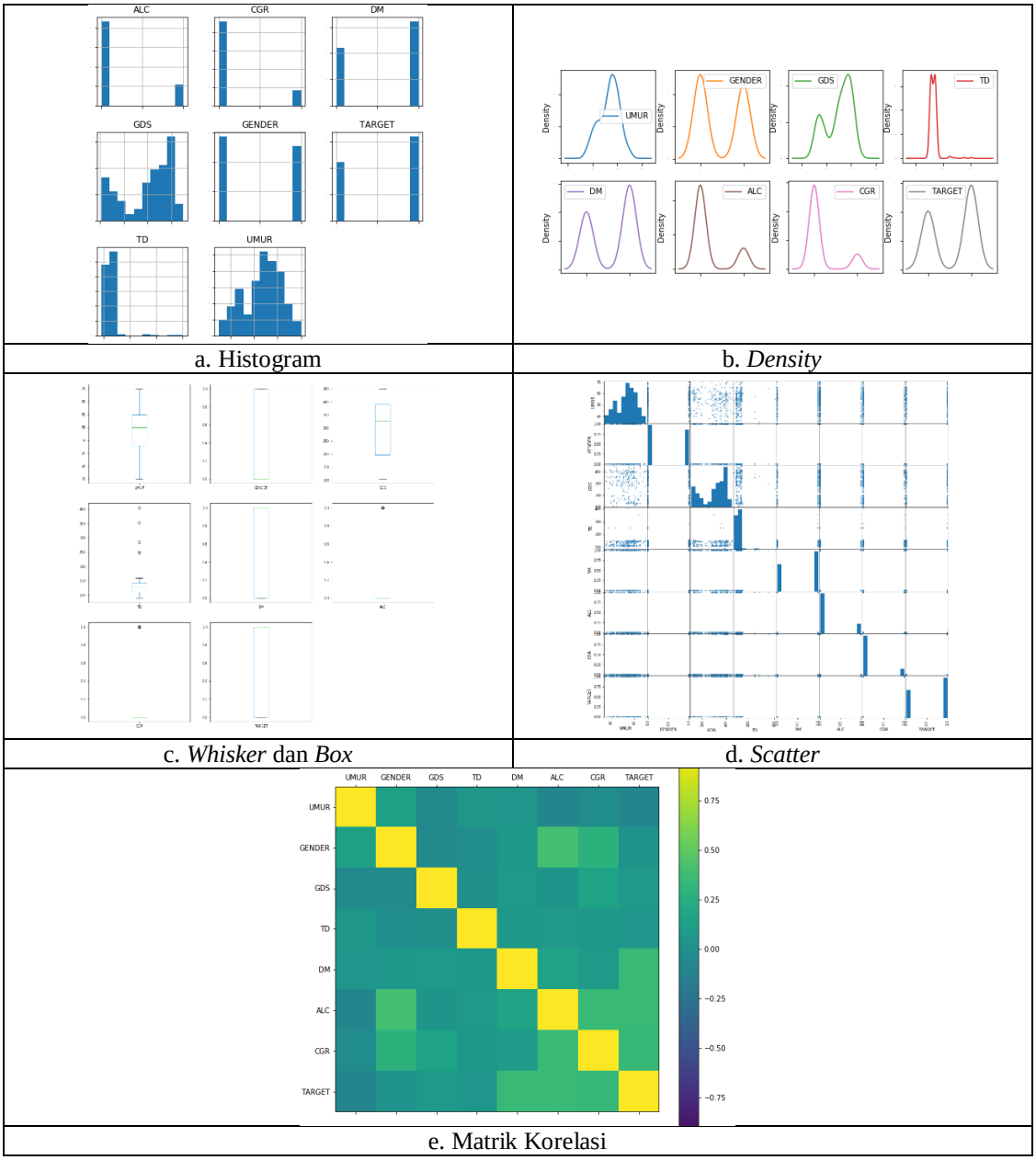
Data diperoleh dari penelitian sebelumnya [15], dengan ukuran sebesar (271 baris, 8 *feature*) selengkapnya terlihat pada Tabel 3 berikut ini:

Tabel 3. Statistik deskriptif.

	UMUR	**GENDER**	**GDS**	**TD**	**DM**	**ALC**	**CGR**	**TARGET**
count	271.00	271.00	271.00	271.00	271.00	271.00	271.00	271.00
mean	53.49	0.47	300.90	130.70	0.59	0.20	0.15	0.59
std	8.00	0.50	101.19	31.13	0.49	0.40	0.36	0.49
min	35.00	0.00	104.00	90.00	0.00	0.00	0.00	0.00
25%	48.00	0.00	197.00	110.00	0.00	0.00	0.00	0.00
50%	55.00	0.00	327.00	128.00	1.00	0.00	0.00	1.00
75%	60.00	1.00	390.00	142.00	1.00	0.00	0.00	1.00
max	70.00	1.00	450.00	405.00	1.00	1.00	1.00	1.00
TARGET								
0	111							
1	160							
dtype: int64								

Berdasarkan Tabel 3 di atas maka dapat disimpulkan sementara bahwa data memiliki 271 baris dengan 8 kolom/fitur, semuanya memiliki tipe integer, terdapat fitur dengan skala berbeda usia termuda adalah 35 tahun dan tertua adalah 70 tahun, rerata usianya 53 dengan standar deviasi sebesar 8 (7.99),

Nilai Gula darah sesaat (GDS) terendah sebesar 104 dan tertinggi sebesar 450, dengan rerata sebesar 300, dan standar deviasinya sebesar 101, sementara Tekanan Darah (TD) terendah sebesar 90, tertinggi sebesar 405, reratanya 130 dengan standar deviasi sebesar 3, dari 271 data, TARGET: 111=NON-DM, dan 160=DM, berikut visualisasi yang dihasilkan pada Gambar 3 berikut:



a. Histogram

b. *Density*

c. *Whisker* dan *Box*

d. *Scatter*

e. Matrik Korelasi

Gambar 3. Visualisasi *Dataset.*

Gambar 3.a memperkuat dugaan bahwa *dataset* tidak normal (hanya terdapat 2 fitur yang cenderung normal GDS dan UMUR), hal ini diperkuat dengan Gambar 3.b, c, dan d arah *skewness* terlihat jelas pada GDS dan UMUR. Sementara hasil korelasi matrik menunjukkan hampir tidak terdapat korelasi baik secara positif maupun *negative* antara 2 fitur yang berhubungan.

3.2. *Pre-Processing*

Pada bagian ini dilakukan pembagian *class* dan atribut pada *dataset* serta *feature elimination*. Data dibagi menjadi 1 *class* (TARGET) dan 7 atribut (UMUR, GENDER, GDS, TD, DM, ALC, CGR) di mana GDS (Nilai Gula Dara Sesaat), DM (Kategori *Diabetes Mellitus* Keturunan atau tidak), TD (Nilai Tekanan Darah), ALC (Kategori *Alcoholic* atau tidak), CGR (Kategori Perokok atau tidak) dengan perbandingan

sebesar 20:80. *Feature elimination* dilakukan dengan 3 teknik dengan mencari nilai akurasi yang tertinggi yaitu *Chi-Square*, *ANOVA F-Value* dan *Mutual Information* Untuk Target Kontinyu terhadap 5 *feature* terbaik, dan didapatkan hasil pada Tabel 4 sebagai berikut:

Tabel 4. Skor Atribut *Feature Elimination*.

Metode	Skor							5 Atribut terpilih
	UMUR	GENDER	GDS	TD	DM	ALC	CGR	
Chi-Square	2.495	0.133	46.73	5.775	13.522	27.989	25.157	GDS, TD, DM, ALC, CGR
ANOVA F-Value	2.097	0.248	1.375	0.778	37.701	39.834	33.036	UMUR, GDS, DM, ALC, CGR
Mutual information	0.036	0.	0.062	0.083	0.092	0.113	0.077	GDS, TD, DM, ALC, CGR

3.3. *Baseline Model*

Setelah mendapatkan 5 urutan *feature* terbaik yang dipilih, maka kita akan menentukan *baseline* model berdasarkan fitur terbaik yang didapatkan dengan 3 teknik di atas menggunakan 3 algoritma klasifikasi linear dan 3 algoritma klasifikasi non-linear. Enam algoritma di atas dipilih berdasarkan intuisi dan seringnya algoritma klasifikasi di atas dipakai dalam *machine learning*, sehingga didapatkan hasil seperti Tabel 5 sebagai berikut:

Tabel 5. *Baseline model* terpilih.

Feature / Model	Chi-square		ANOVA F-Value		Mutual Information	
	Mean	STD	Mean	STD	Mean	STD
Logistic Regression	0.713636	0.089423	**0.718182**	0.091140	0.713636	0.089423
Linear Discriminant Analysis	0.713636	0.089423	**0.718182**	0.091140	0.713636	0.089423
K-Nearest Neighbors	0.611905	0.076023	0.481818	0.120457	0.491991	0.094849
Decision Tree	0.685281	0.057432	0.648485	0.072245	0.634199	0.092097
Gaussian Naïve Bayes	0.638961	0.125690	0.638961	0.125690	0.638961	0.125690
Support Vector Machine	0.639177	0.085008	0.596753	0.129274	0.509957	0.096378

Terlihat 2 model (*Logistic Regression* dan *Linear Discriminant Analysis*) dengan ANOVA F-Value menghasilkan akurasi tertinggi 71.8182 % maka Feature Elimination yang dipakai adalah adalah yang menggunakan teknik ANOVA F-Value (UMUR, GDS, DM, ALC, CGR) sehingga sementara *baseline* algoritma yang akan dilakukan studi adalah ***Logistic Regression*** dan ***Linear Discriminant Analysis***.

3.4. *Pipelining, Rescale,* dan Standardisasi Data

Walaupun *baseline* model telah tampak, namun masih berdasarkan *feature* terpilih, sementara *dataset* belum dilakukan *rescaling* dengan standardisasi. Pada Tabel 6 berikut akan dilakukan *rescale* dan menentukan model yang akan dilakukan *tuning* parameter dan berikut hasil yang didapatkan setelah *feature selection* dan standardisasi dengan fungsi ***StandardScaler:***

Tabel 6. Akurasi Model *Pipelining* dan Standardisasi.

Feature / Model	Chi-square		ANOVA F-Value		Mutual Information	
	Mean	STD	Mean	STD	Mean	STD
Logistic Regression	0.713636	0.089423	0.718182	0.091140	0.718182	0.091140
Linear Discriminant Analysis	0.713636	0.089423	0.718182	0.091140	0.713636	0.089423
K-Nearest Neighbors	0.680736	0.076381	0.713853	0.099623	0.710173	0.121677
Decision Tree	0.680736	0.072174	0.671861	0.066427	0.661472	0.099610
Naïve Bayes	0.638961	0.125690	0.638961	0.125690	0.638961	0.125690
Support Vector Machine	0.713636	0.089423	0.713203	0.097322	0.727489	0.088042

Berdasarkan hasil standardisasi dengan *pipelining* berdasarkan 5 *feature* terpilih, didapatkan hasil yang mengejutkan, KNN dan SVM mengambil alih performa, **K-Nearest Neighbors** (KNN) dengan metode *feature elimination* ANOVA F-Value sebesar **71.3853** dan *Support Vector Machine* (SVM) dengan metode *mutual information classification* sebesar **72.7489,** seperti pada Tabel 6 di atas. Dengan demikian *baseline* algoritma bergeser ke KNN dan SVM untuk dilakukan *tuning*.

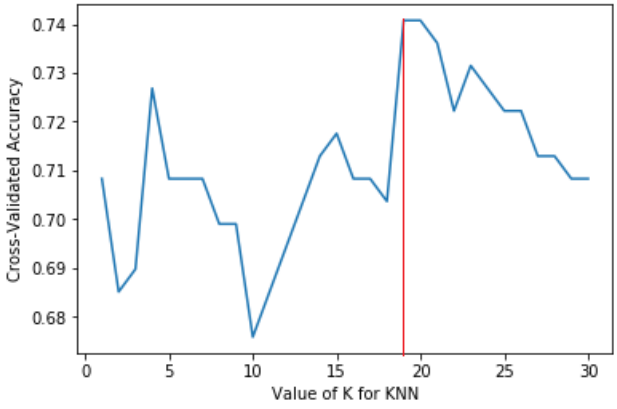
3.5. *Tuning* Parameter Model

Tuning parameter akan dilakukan terhadap 2 algoritma terpilih yang memiliki akurasi tertinggi setalah dilakukan *pipelining* yaitu KNN dan SVM. Parameter pada model KNN hanya didasarkan pada nilai k yang akan diberi parameter 1 hingga 31 dengan bobot ***uniform***, sementara *cross validation* ditetapkan dengan 10-fold *validation*, aplikasi *tuning* dilakukan dengan fungsi ***GridSearch*** dari *sklearn* sehingga kita dapat dengan mudah mendapatkan skor dan parameter terbaik, seperti ditampilkan pada Tabel 7 berikut ini.

Tabel 7. Tuning n *neighbors* pada model KNN.

Mean Score	SD Score	Neighbors	Bobot
0.708333	0.115261	1	
0.685185	0.127127	2	
0.689815	0.124167	3	
0.726852	0.141446	4	
0.708333	0.103100	5	
0.708333	0.085865	6	
0.708333	0.093411	7	
0.699074	0.090548	8	
0.699074	0.092030	9	
0.675926	0.098002	10	
0.685185	0.087256	11	
0.694444	0.104942	12	
0.703704	0.108746	13	
0.712963	0.107402	14	
0.717593	0.099310	15	
0.708333	0.088960	16	***uniform***
0.708333	0.094265	17	
0.703704	0.093484	18	
0.740741	0.102402	19	
0.740741	0.100675	20	
0.736111	0.095270	21	
0.722222	0.088126	22	
0.731481	0.088906	23	
0.726852	0.087814	24	
0.722222	0.094756	25	
0.722222	0.092020	26	
0.712963	0.089093	27	
0.712963	0.083580	28	
0.708333	0.085338	29	
0.708333	0.078743	30	

Dari Tabel 7 di atas maka kita dapat melihat pada posisi n=19, didapatkan akurasi sebesar **74%**, jika kita plot dalam bentuk *graphic elbow* pada Gambar 4 kita dapat melihat performa terbaiknya dengan visualisasi yang lebih jelas pada garis *vertical* merah sebagai berikut:



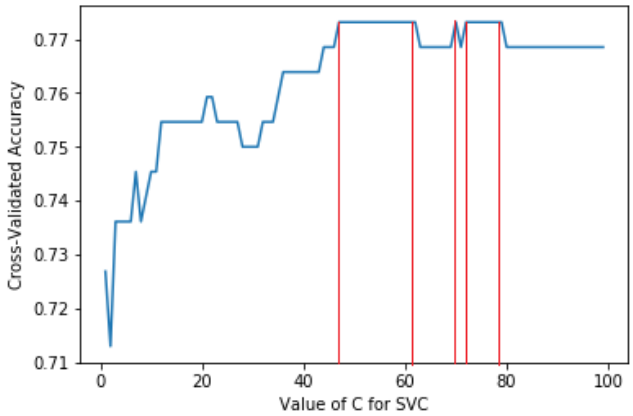
Gambar 4. *Grafic Elbow Tuning Paramater* Model KNN.

Sementara itu *tuning* pada model SVM, parameter yang diperlukan adalah nilai C, gamma, *kernel* yang akan dikombinasikan. Parameter C akan diberi nilai mulai dari 1 hingga 100, Parameter gamma antara 0.001 hingga 0.0001, sementara *kernel* yang dipilih adalah linear, *polynomial*, *sigmoid*. Ketiga parameter ini akan dikombinasikan dan untuk menghemat tempat yang ditampilkan adalah kombinasi yang menghasilkan nilai optimal, seperti pada Tabel 8 di bawah ini.

Tabel 8. Tuning Nilai C dan kernel pada model SVM.

Mean Score	SD Score	C	Kernel
0.726852	0.115261	1	
0.712963	0.127127	2	
0.736111	0.124167	3	
0.736111	0.141446	4	
0.736111	0.103100	5	
…	…	…	
0.754630	0.108180	20	
…	…	…	
0.763889	0.106730	40	
0.763889	0.106730	41	
0.763889	0.106730	42	
0.763889	0.106730	43	
0.768519	0.106024	44	
0.768519	0.106024	45	
0.768519	0.106024	46	***rbf***
0.773148	**0.107094**	**47**	
0.773148	**0.107094**	**48**	
0.773148	**0.107094**	**49**	
0.773148	**0.107094**	**50**	
…	**…**	**…**	
0.773148	**0.107094**	**60**	
…	**…**	**…**	
0.773148	**0.103089**	**70**	
…	…	…	
0.768519	0.100243	80	
…	…	…	
0.768519	0. 0.100243	90	
…	..	..	
0.768519	0.100243	99	

Dari Tabel 8 di atas maka kita dapat melihat pada posisi **C= 47 - 62 atau 70 atau 72 - 79** dengan kombinasi *kernel* **rbf**, didapatkan akurasi sebesar **77%**, jika kita plot dalam bentuk *graphic elbow* pada Gambar 5 kita dapat melihat performa terbaiknya dengan visualisasi yang lebih jelas pada garis *vertical* merah sebagai berikut:



Gambar 5. *Grafic Elbow Tuning Paramater* Model SVM.

3.6. Finalisasi Model

Berdasarkan hasil *tuning* terhadap 2 algoritma di atas dan yang mempunyai nilai akurasi tertinggi adalah model SVM, dengan demikian maka model SVM menjadi model terakhir yang akan dipakai dalam

mengklasifikasi penyakit *diabetes mellitus* sesuai dengan atribut terpilih (Umur, Gender, Gula darah Sesaat, Tekanan Darah, Riwayat DM, *Alcoholic*, dan Perokok) yang sudah distandardisasi. Hasil performa dievaluasi dengan mengukur rerata akurasi pada data testing, *confussion matrix* dengan 10-fold *validation*, dihasilkan sebagai berikut akurasi sebesar 76%, sementara *confussion matrix* untuk data testing 54 *record* seperti Tabel 9 di bawah.

Tabel 9. *Confussion Matrix* data testing Model SVM.

Aktual vs Prediksi	Positive	Negative
Positive	12	8
Negative	6	29

	precision	recall	f1-score	Support
0	0.67	0.60	0.63	20
1	0.78	0.83	0.81	35

4. Kesimpulan

Berdasarkan hasil penelitian yang sudah dilakukan, maka sementara ini dapat disimpulkan model klasifikasi biner yang optimal dari 3 algoritma linear dan 3 algoritma non-linear, terhadap *dataset* kecil dengan atribut kecil adalah model *Support Vector Machine* (SVM). Model SVM ini optimal dengan parameter nilai C atau Center = 47-62 atau 70 atau 72-79, sementara *kernel* yang diberikan adalah **rbf (*radial basis function*,** nilai riil pada fungsi *gamma* yang nilainya tergantung pada jarak dengan titik *origin*). Model SVM menghasilkan akurasi optimal 76.3% dengan *kernel* rbf dan untuk nilai C antara 47-62 atau 70 atau 72-79, didapatkan akurasi tertinggi pada nilai C=70 sebesar 76.3%, sementara presisinya 78% dengan *recall* 83% dan skor F1 *test* sebesar 81%. Ke depan perlu dilakukan kombinasi tidak saja model linear, non-linear tetapi juga dengan model berbasis *neural network* dan *fuzzy neural network*. Selain itu perlu dipikirkan juga untuk melakukan komparasi kombinasi model *ensemble* (*boosting* dan *bagging*) dengan *tuning* parameter model linear dan non-linear.

Daftar Pustaka

[1] Fatmawati, "Perbandingan Algoritma Klasifikasi Data Mining Model C4.5 Dan Naïve Bayes Untuk Prediksi Penyakit Diabetes," *J. Techno Nusa Mandiri*, vol. XIII, no. 1, p. 50, 2016.

[2] J. J. Pangaribuan, "Mendiagnosa Penyakit Diabetes Mellitus Dengan Menggunakan Metode Extreme Learning Machine," 2016.

[3] I. L. Qurnia, E. Prasetyo, and R. F. Zainal, "Classification Of Diabetes Disease Using Nive Bayes Case Study : Siti Khadijah Hospital," 2016.

[4] A. W. W. Wayan Firdaus Mahmudy, "Klasifikasi Artikel Berita Secara Otomatis Menggunakan Metode Naïve Bayes Classifier Yang Dimodifikasi," *TEKNO*, vol. 21 Maret, 2014.

[5] S. Natalius, "Makalah II 2092 Probabilitas dan Statistik-Sem. I Tahun," 2010.

[6] F. Maspiyanti and J. Gatc, "Diagnosa Penyakit Jantung Pada Ponsel Menggunakan Pohon Keputusan," *J. Teknol. Terpadu*, vol. 1, no. 1, 2015.

[7] M. Lestari, "Penerapan Algoritma Klasifikasi Nearest Neighbor (K-NN) Untuk Mendeteksi Penyakit Jantung," 2014.

[8] B. G. Choi, S. W. Rha, S. W. Kim, J. H. Kang, J. Y. Park, and Y. K. Noh, "Machine learning for the prediction of new-onset diabetes mellitus during 5-year follow-up in non-diabetic patients with cardiovascular risks," *Yonsei Med. J.*, vol. 60, no. 2, pp. 191–199, Feb. 2019.

[9] P. A. Amit kumar Dewangan, "Classification of Diabetes Mellitus Using Machine Learning Techniques," *Int. J. Eng. Appl. Sci.*, vol. 2, no. 5, pp. 145–148, 2015.

[10] K. Akyol and B. Şen, "Diabetes Mellitus Data Classification by Cascading of Feature Selection Methods and Ensemble Learning Algorithms," *Int. J. Mod. Educ. Comput. Sci.*, vol. 10, no. 6, pp. 10–16, Jun. 2018.

[11] N. Nai-Arun and R. Moungmai, "Comparison of Classifiers for the Risk of Diabetes Prediction," in *Procedia Computer Science*, 2015, vol. 69, pp. 132–142.

[12] R. P. R. S. Suryakirani, "Comparative Study and Analysis of Classification Algorithms In Data Mining Using Diabetic Dataset," *IJSRST*, vol. 4, no. 2, pp. 299–304, 2018.

[13] P. Chen and C. Pan, "Diabetes classification model based on boosting algorithms," *BMC Bioinformatics*, vol. 19, no. 1, Mar. 2018.