Prediksi Penyakit Menggunakan Algoritma K-Means dan GA untuk Reduksi Dimensi dengan Mengintegrasikan SVM pada Data Berdimensi Tinggi.

Disease Prediction using K-Means and GA for Dimension Reduction by Integrating SVM on High Dimensional Data

Jodi noordiansyah[1], Fhira Nhita S.T., M.T.[2], Danang Triantoro M, S.Si.,M.T.[3]
[1,2,3]Ilmu Komputasi, Fakultas Informatika, Universitas Telkom
[1]jodinoordiansyah@gmail.com,[2]farid.alchair@gmail.com, [3]dto.lecture@gmail.com

Abstrak

Dimensionality adalah salah satu tantangan dalam *data mining*, tantangan ini meliputi jumlah atribut yang begitu besar sehingga sering disebut dengan *curse of dimensionality*. Semakin besar jumlah atribut maka semakin memakan waktu dan memerlukan upaya komputasi yang berlebihan sehingga data sulit untuk ditangani. Hal yang diperlukan untuk mengatasi tantangan ini adalah dengan cara mereduksi dimensi dari data tersebut.

Teknik reduksi yang dibahas pada tugas akhir ini adalah dengan menggunakan algoritma *K-Means* dengan cara pengelompokan data pada setiap *cluster*. Algoritma ini digunakan untuk mereduksi *record* yang kemudian dilanjutkan oleh GA sebagai *feature selection* untuk memilih atribut-atribut yang paling optimal berdasarkan nilai *fitness* tertinggi. Pencarian nilai *fitness* dilakukan dengan menggunakan metode klasifikasi yaitu SVM.

Hasil dari pengujian sistem menghasilkan data yang direduksi oleh *K-Means* memiliki akurasi yang lebih rendah untuk *dataset* tertentu dibandingkan tanpa menggunakan *K-Means.* Atribut optimal yang dihasilkan GA bervariasi berdasarkan parameter yang digunakan. Data yang digunakan adalah data penyakit berdimensi tinggi berupa ekspresi gen yaitu *colon tumor* dan *leukemia.* Akurasi rata-rata terbaik yang didapat pada data *colon tumor* adalah 92.86% dengan jumlah atribut terpilih yaitu 983 atribut, sedangkan untuk data *leukemia* selalu menghasilkan atribut yang berkualitas dengan rata-rata akurasi 100%.

Kata kunci : *dimensionality, data mining, K-Means,* GA, SVM

Abstract

Dimensionaliy is the one of data mining challenge, this challenge include large number of attribute this also called the curse of dimensionality. the greater number of attribute is more time consuming and need more excessive computational effort to handle. the things that needed to handle this challenge is to reduce the dimension of data.

The reduction technique that mentioned in this final task is to use K-Means algorithm to grouping the data into each cluster. The use of this algorithm is to reduce the record and then GA as feature selection to select the optimal attribute with higher fitness value. The search for fitness value can be done by classification method SVM.

The result of the system examination is data reduced by K-Means have lowest accuracy in certain dataset compared to without using K-Means. The result of optimal atribut which GA produce is varies based on the use of different parameter. The data that used is the high dimensional disease data in the form of gene expression, i.e. colon tumor and leukemia. The best average accuracy for colon tumor is 92.86% with selected attribute 983 attributes while leukemia always produce best attribute with average accuracy 100%.

Keywords : *dimensionality, data mining, K-Means, GA , SVM*

1. Pendahuluan

1.1. Latar Belakang

Dengan begitu banyaknya tantangan yang dimiliki oleh *data mining,* pengolahan data dalam berbagai macam penelitian menjadi sangat sulit untuk ditangani, tantangan tersebut meliputi [3]: *dimensionality, complex and heterogeneous data, data quality, data ownership and distribution, privacy preservation,* dan *streaming data.* Tantangan tersebut kemudian menimbulkan salah satu masalah dalam industri kesehatan [11] yaitu data penyakit berdimensi tinggi, kategori tantangan ini adalah *dimensionality.*

Dalam kasus ini, data tersebut memiliki jumlah atribut yang begitu banyak, semakin banyak atribut maka semakin banyak memakan waktu dan memanfaatkan upaya komputasi yang berlebihan sehingga data sulit untuk ditangani [5]. Maka dari itu, untuk menjawab tantangan dari masalah tersebut reduksi dimensi sangat diperlukan agar mendapatkan nilai akurasi yang lebih baik untuk setiap penelitian.

Reduksi dimensi ini dilakukan untuk mengurangi bagian-bagian atribut yang tidak diperlukan pada data yang sedang diteliti, contohnya: terdapat nilai dari salah satu atribut tidak ada dalam data (*missing value)*

hal ini mungkin sekali terjadi dikarenakan informasi yang diperlukan tidak dapat diperoleh. Selain itu, terdapat juga atribut yang karakteristiknya berbeda diantara atribut-atribut lain (*outlier*) dan adanya *error* pada data (*noise*). Itulah gangguan-gangguan yang terdapat pada data dikarenakan kesalahan dalam pengumpulan informasi.

Data mining menggunakan *Evolutionary Algorithms* (EAs) atau bisa disebut sebagai *Evolutionary data mining* dapat digunakan untuk permasalahan tersebut. EAs dapat membantu *data mining* untuk dapat mereduksi dimensi dengan cara membuat aturan-aturan secara acak yang kemudian akan diseleksi untuk mendapatkan atribut-atribut yang paling optimal. Banyak studi yang membahas tentang *evolutionary data mining* diantaranya: penggunaan *evolutionary data mining* pada penyakit diabetes [11], kumpulan data medis [5], prediksi penyakit jantung [9] dan masih banyak lagi.

Algoritma *K-Means* sebagai *instance reduction* dapat digunakan untuk menghilangkan gangguan terhadap data dan *Genetic Algorithm* (GA) digunakan untuk memilih atribut yang paling optimal dalam data tersebut. *Support Vector Machine* (SVM) digunakan sebagai *tools* untuk melakukan klasifikasi berdasarkan keluaran yang telah diperoleh oleh GA untuk mendapatkan akurasi yang lebih baik[5,11]. Sebelumnya penelitian terhadap *K-Means*, GA, dan SVM ini pernah dilakukan dalam memprediksi penyakit diabetes dan memperoleh akurasi sebesar 98% [11]. Pada tugas akhir ini dilakukan prediksi penyakit menggunakan algoritma *K-Means*, GA dan SVM pada kasus data berdimensi tinggi.

1.2. Perumusan Masalah

Permasalahan yang dapat diselesaikan dalam tugas akhir ini terdiri dari:

1. Bagaimana implementasi *K-Means* dan GA dengan mengintegrasikan SVM pada data penyakit berdimensi tinggi ?
2. Bagaimana cara kerja *K-Means* untuk dapat menghilangkan gangguan data dan GA untuk memilih atribut paling yang optimal pada data penyakit berdimensi tinggi ?
3. Bagaimana prediksi yang didapatkan SVM pada data penyakit berdimensi tinggi ?
4. Bagaimana performansi yang didapatkan SVM pada data penyakit berdimensi tinggi ?

1.3. Batasan Masalah

Adapun batasan masalah dari tugas akhir ini, yaitu :

1. *Data set* yang digunakan untuk tugas akhir ini adalah data penyakit berupa data ekspresi gen yang berasal dari *Kent Ridge Bio-medical Data Set Repository* [6].
2. Algoritma *K-Means Clustering* digunakan sebagai algoritma untuk mereduksi *record*.
3. Algoritma EAs yang digunakan adalah GA yang berfungsi sebagai *feature selection.*
4. SVM digunakan sebagai klasifikasi.

1.4. Tujuan

Tujuan untuk menyelesaikan masalah tersebut adalah :

1. Mengimplementasikan Algoritma *K-Means* dan GA untuk mereduksi dimensi dengan mengitegrasikan SVM pada data penyakit berdimensi tinggi.
2. Mengetahui cara kerja *K-Means* dan GA pada data penyakit berdimensi tinggi.
3. Mengetahui hasil prediksi SVM pada data penyakit berdimensi tinggi.
4. Menganalisis hasil performansi yang didapatkan SVM pada data penyakit berdimensi tinggi.

2. Tinjauan Pustaka

2.1. *Data Mining*

Data mining adalah bagian dari proses penemuan pengetahuan terbesar yang meliputi tugas *preprocessing* seperti *data extraction*, *data cleaning*, *data fusion*, *data reduction*, dan *feature construction*, dan juga meliputi tugas *post processing* seperti *pattern* and *model interpretation*, *hypothesis confirmation*, dan sebagainya [3]. Proses data mining ini cenderung berulang dan interaktif.

2.2. Clustering

Clustering adalah teknik yang umum digunakan untuk *statistical data analysis,* yang digunakan di berbagai bidang termasuk *machine learning*, *data mining*, *pattern recognition*, *image analysis*, dan *bioinformatics* [7], *Clustering* bertugas untuk mempartisi suatu objek kedalam sebuah kelompok (*cluster*) yang memiliki kemiripan antara satu sama lain dibandingkan dengan *cluster* lain.

2.2.1. K-Means Clustering

K-Means adalah salah satu algoritma *clustering* yang bersifat *unsupervised learning* (tidak memerlukan label kelas) [16]. Algoritma ini pengelompokkan objek yang diberikan kedalam beberapa *cluster*, *k* [7]. *K-Means* memilih *k* secara acak sebagai titik pusat *(centroid).* pengelompokkan data dilakukan dengan cara

menghitung seberapa dekat objek tersebut dengan *centroid*. Setelah itu, lakukan perhitungan ulang terhadap *centroid* tersebut untuk menghasilkan *centroid* baru, proses ini dilakukan sampai *centroid* baru tidak berubah terhadap *centroid* lama. Algoritma ini bertujuan untuk meminimalkan *objective function* yang dikenal sebagai *Sum Square Error (SSE).*
Fungsi *SSE* diberikan sebagai berikut[18].

$$( ) \quad \sum \sum (\| \quad \|) \qquad \textbf{(2.2)}$$

Keterangan :
μ : *centroid*
x : sub bagian dari data,
k : *cluster*
: jarak terdekat pada cluster
‖ ‖ : *eucledian distance* antara dan

Berikut adalah tahapan algoritma *K-Means clustering.*
1. Pilih *k* secara acak untuk dijadikan sebagai inisialisasi jumlah *centroid*.
2. Tentukan *centroid* awal.
3. Hitung jarak antar titik data terhadap *centroid*.
4. Hitung *centroid* baru.
5. Ulangi langkah ke-3 dan ke-4sampai *centroid* tidak berubah.

2.2.1.1. *K-Means Clustering* untuk Penanganan *Outlier*

K-Means dapat digunakan untuk mereduksi jumlah *record* dengan cara penghapusan *outlier*. *Outlier* adalah sebuah titik data dengan jarak terjauh dibandingkan dengan data lainnya. Pereduksian ini dilakukan dengan memilih indeks dari *record* yang dikelompokan. Apabila terdapat indeks *record* yang tidak masuk kedalam *cluster* manapun atau sebuah *cluster* memiliki jumlah *record* yang sedikit atau data tersebut terpaksa masuk pada sebuah *cluster* maka *cluster* tersebut dihapuskan [10].

2.3. *Evolutionary Algorithms* (EAs)

Evolutionary Algorithm (EAs) adalah algoritma-algoritma optimasi yang berbasis evolusi biologi yang ada di dunia nyata [14]. Dalam teori evolusi, suatu individu dalam sebuah populasi akan saling berkompetisi untuk dapat bertahan hidup di suatu daerah yang memiliki sumber daya terbatas. Tingkat adaptasi pada setiap individu dapat menentukan individu mana yang akan tetap bertahan hidup dan individu mana yang akan musnah.

2.3.1. *Genetic Algorithm* (GA)

Genetic Algorithm (GA) adalah salah satu algoritma EAs. GA pertama kali dipublikasikan oleh John Holland (1975) di Amerika Serikat[14]. Pada saat itu, GA memiliki bentuk yang sangat sederhana sehingga disebut *Simple* GA. Ciri utama dari Algoritma ini adalah menitikberatkan pada rekombinasi *(crossover)* [15]. GA dapat digunakan sebagai *tools* pencarian yang optimal untuk memilih *subset* dari beberapa atribut [12], GA memiliki banyak keturunan yang dapat menjelajahi ruang solusi pada waktu bersamaan [17].
.

2.4. *Support Vector Machine (SVM)*

Support Vector Machine (SVM) merupakan metode *supervised learning* (memiliki label kelas) untuk analisis data, pengenalan pola, klasifikasi dan regresi [5] yang dapat digunakan untuk permasalahan data bersifat *linear* dan *nonlinear*. Cara kerja SVM yaitu menggunakan *nonlinear mapping* untuk mengubah data aktual ke dalam ruang data berdimensi tinggi [2], hal ini dilakukan untuk mengetahui garis pemisah atau *hyperplane* yang bertujuan sebagai sebuah batas keputusan atau bisa disebut dengan *decision boundary* yang memisahkan *record* satu dengan *record* lainnya.

Dengan penggunaan *nonlinear mapping* tersebut, data yang memiliki dua kelas dapat selalu dipisahkan menggunakan *hyperplane*. *Hyperplane* dapat ditemukan oleh *support vector* atau *record* yang menjadi data latih dan *margin* yang merupakan jarak *hyperplane* ketitik terdekat antara dua himpunan (kelas) yang didefinisikan oleh *support vector*. Kecepatan waktu latih SVM dapat sangat lambat namun memiliki segi keakuratan yang sangat tinggi yang bergantung dengan kemampuannya memodelkan *nonlinear decision boundary* yang begitu kompleks [2].

Dalam permasalahan *nonlinear* tidak ada garis yang dapat memisahkan kelas [2]. Maka dari itu, permasalahan *linear* dapat diperluas agar dapat menyelesaikan permasalahan *nonlinear* dengan cara mengubah

data kedalam dimensi yang lebih tinggi. Data diubah menjadi *nonlinear mapping* yang didefinisikan sebagai ( ) sehingga menghasilkan fungsi kernel sebagai berikut [2].

( ) ( ) ( ) (2.3)

Untuk setiap data latih yang didefinisikan sebagai ( ) ( ) dapat digantikan oleh ( )
Berikut adalah *dual problem* untuk permasalahan *nonlinear* [11].

$$\sum \quad -\sum\sum \qquad (\quad) \tag{2.4}$$

didefinisikan menjadi ( ). Fungsi *kernel* yang digunakan adalah *Radial Basis Function* (RBF) kelebihan RBF *kernel* dapat ditemukan di referensi [17], dalam pemograman SVM dibutuhkan *feature scalling* [4] dan menggunakan *k-fold crossvalidation* [8].

2.5. Evaluasi Model

Untuk memvisualisasikan performansi dari SVM, *confusion matrix* dibentuk. *Evaluation metrics* yang digunakan adalah *sensivity, specificity, positively predicted* dan *negatively predicted. Sensivity* adalah kemampuan untuk mengidentifikasi secara benar bahwa pasien terkena penyakit dan *specificity* adalah kemampuan untuk mengidentifikasi secara benar bahwa pasien tidak terkena penyakit. Sedangkan *positively predicted* dan *negatively predicted* adalah proporsi nilai positif dan negatif yang diprediksikan [11].

( ) ——— **(2.5)**

( ) ——— **(2.6)**

( ) ——— **(2.7)**

( ) **(2.8)**
———

Berikut adalah ilustrasi dari *confusion matrix* [11].

Tabel 2-1 Confusion Matrix

Actual	*Predicted*	
	Positive	*Negative*
Positive	*True Positive (TP)*	*False Negative (FN)*
Negative	*False Postive (FP)*	*True Negative (TN)*

3. Perancangan Sistem

3.1. Deskripsi Sistem

Sistem yang dibangun bertujuan untuk menghasilkan prediksi dan akurasi yang baik dengan cara mereduksi data menggunakan *data mining* dengan bantuan EAs dan SVM untuk mengklasifikasikan data. Data masukan yang digunakan merupakan data berdimensi tinggi, data tersebut kemudian melalui tahap *preprocessing.* dalam tahap ini jumlah *record* pada data direduksi dengan menggunakan algoritma *K-Means* dan menghasilkan data hasil reduksi *record.* Data tersebut melalui proses normalisasi dengan menggunakan *feature scalling* dan menghasilkan data yang telah ternormalisasi. Setelah data masukan ternormalisasi, data tesebut kemudian melalui proses GA dan dilatih lalu di uji oleh SVM untuk memilih atribut terbaik berdasarkan nilai *fitness.* Hasil keluaran dari GA+SVM kemudian melalui proses prediksi SVM untuk mengetahui hasil akurasi dan prediksi dari atribut yang terpilih oleh GA. Berikut adalah skema umum dari sistem yang dibangun

3.2. Dataset

Data yang digunakan merupakan data penyakit berdimensi tinggi berupa ekspresi gen yang diperoleh dari *Kent Ridge Bio-medical Data Set Repository* [11]. Data yang berada dalam *repository* tersebut meliputi

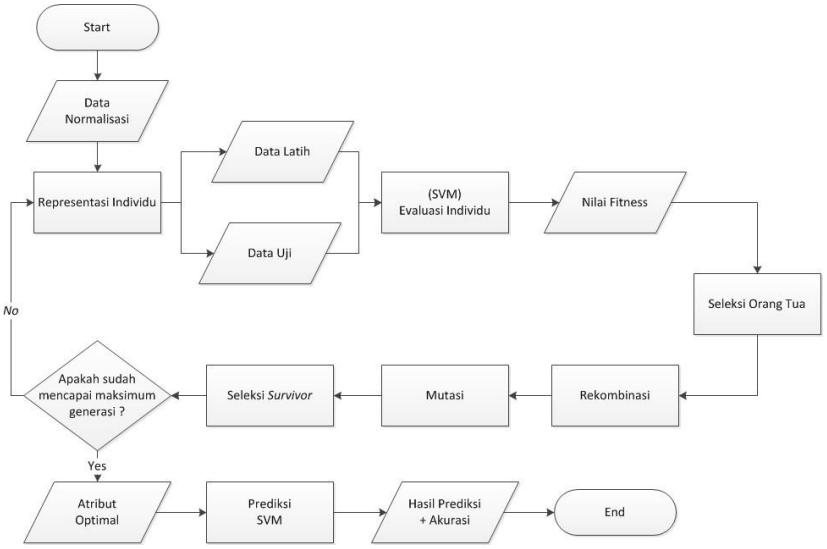
gene expression data, *protein profiling data* dan *genomic sequence data* yang sudah pernah dipublikasikan dalam berbagai jurnal yaitu *colon tumor dan Leukemia*

.

3.3. Alur Kerja GA+SVM

Data masukan merupakan data yang telah ternormalisasi kemudian data melalui proses GA yang berawal dari representasi individu yaitu membangkitkan sejumlah individu dengan nilai acak 0 dan 1 dengan panjang gen setara dengan jumlah atribut data yang dimasukan. jika nilai yang dibangkitkan sama dengan 0 maka atribut yang bersangkutan tidak digunakan, sebaliknya jika data yang dibangkitkan sama dengan 1 maka atribut yang bersangkutan digunakan untuk memulai pengolahan data.

Flowchart GA + SVM dapat dilihat digambar 3.3-1. Informasi yang didapat dari representasi individu diterapkan pada data masukan dan membaginya menjadi data latih dan data uji. pembagian data tersebut digunakan untuk melakukan proses evaluasi individu dengan menggunakan SVM, proses ini menghasilkan nilai *fitness* untuk setiap individu yang diberikan. Individu tersebut menjalani proses seleksi orang tua dengan menggunakan *roullete wheel.*



Gambar 3.3-1 Flowchart GA+ SVM

Proses tersebut menghasilkan individu orang tua yang kemudian akan direkombinasi untuk menghasilkan individu baru yang lebih baik atau lebih buruk. Individu baru yang dihasilkan kemudian menjalani proses mutasi, proses ini membangkitkan bilangan acak, jika bilangan acak yang dihasilkan kurang dari probabilitas mutasinya maka nilai gen dalam individu tersebut akan digantikan posisinya yaitu nilai 0 menjadi 1 dan 1 menjadi 0. Seleksi *survivor* dilakukan dengan menggantikan individu lama dengan individu baru.

Setelah melalui proses tersebut akan dilakukan pegecekan apakah sistem sudah mencapai maksimum generasi yang diberikan atau belum. Jika belum maka algoritma GA + SVM berulang. Jika sudah maka dihasilkanlah atribut yang paling optimal. Atribut tersebut menjalani proses prediksi oleh SVM yang akan menghasilkan kelas prediksi sesuai informasi yang diberikan dan mencatat nilai akurasi dari atribut optimal tersebut.

4. Pengujian dan analisis

Pada bab ini berisi pembahasan hasil pengujian berdasarkan skenario pengujian yang diajukan pada bab perancangan sistem. pada bab ini juga dijelaskan analisis terhadap hasil pengujian tersebut. Pengujian data diatas dilakukan dengan menggunakan beberapa kombinasi parameter yang ditunjukan pada tabel 4-1.

Tabel 4-1 Parameter yang Digunakan untuk Pengujian GA+SVM

Kombinasi parameter	Maksimum Generasi	Ukuran Populasi	Probabilitas Rekombinasi	Probabilitas Mutasi
1	20	50	0.6	0.05
				0.1
			0.8	0.05
				0.1

Kombinasi parameter	Maksimum Generasi	Ukuran Populasi	Probabilitas Rekombinasi	Probabilitas Mutasi
2	10	100	0.6	0.05
				0.1
			0.8	0.05
				0.1
3	5	200	0.6	0.05
				0.1
			0.8	0.05
				0.1

Parameter tersebut akan digunakan untuk menguji sistem yang dibangun yaitu pengujian terhadap GA + SVM , GA digunakan sebagai *feature selection* dan SVM digunakan sebagai perhitungan nilai *fitness*, prediksi serta akurasi dan pengujian terhadap *K-Means* + GA + SVM. Dengan *K-Means* digunakan sebagai algoritma untuk mereduksi jumlah *record* yang digunakan. Parameter yang digunakan bertujuan untuk menguji setiap 1000 individu yang dibangkitkan oleh GA.

Perhitungan nilai *fitness* yang digunakan adalah sebagai berikit.

(2.32)

Pencarian nilai *fitness* pertama dilakukan dengan menggunakan *3-Fold crossvalidation* (3FCV). Pemilihan angka 3 ini didasari oleh jumlah *record* yang berada dalam data yang digunakan agar data tersebut memiliki porsi data uji tidak terlalu sedikit dan perkiraan data latih dan data uji yang digunakan kurang lebih mendekati 70% dan 30%, kedua menggunakan persentase pembagian data latih dan uji sebesar 70% dan 30%. Berikut adalah hasil rata-rata dari semua pengujian yang telah dilakukan berdasarkan akurasi dan jumlah atribut.

Tabel 4-2 Hasil Pengujian Dataset Colon Tumor dan Leukemia

Colon Tumor (GA+SVM)				Colon Tumor (Kmeans+GA+SVM)			
Atribut Optimal	70%/30%	Atribut Optimal	3FCV	Atribut Optimal	70%/30%	Atribut Optimal	3FCV
946	84.21%	1003	90.91%	950	78.57%	983	92.86%
Leukemia (GA+SVM)				Leukemia (Kmeans+GA+SVM)			
Atribut Optimal	70%/30%	Atribut Optimal	3FCV	Atribut Optimal	70%/30%	Atribut Optimal	3FCV
3472	95.45%	3508	100%	3518	100%	3529	100%

Dari hasil pengujian diatas terlihat bahwa penggunaan *K-Means* pada data *colon tumor* akurasi yang diperoleh lebih rendah dibandingkan tanpa menggunakan *K-Means.* Hal ini terjadi dikarenakan jumlah *record* pada data yang digunakan lebih sedikit dibandingkan GA+SVM. Sedangkan untuk *Leukemia* akurasi yang diperoleh *k-fold crossvalidation* selalu menghasilkan akurasi 100% dan *K-Means* berhasil menghilangkan *outlier* pada data saat pengujian menggunakan persentase pembagian 70%/30%. Berikut adalah hasil reduksi *record* dengan menggunakan *K-Means.*

Tabel 4-3 Hasil Reduksi Record dengan Menggunakan K-Means

Nama Penyakit	Jumlah *Record*	Jumlah *Record* Setelah Reduksi	Jumlah *Outlier*
ColonTumor	62	45	17
Leukemia	72	48	24

Jumlah *cluster* yang digunakan untuk mereduksi *record* dari data yang digunakan adalah *k*=2, didasari dengan ada atau tidak adanya *outlier*. Apabila jumlah *record* yang menempati satu *cluster* lebih banyak dibandingkan dengan *cluster* lain maka *record* tersebut tidak mengandung *outlier*, sebaliknya apabila jumlah *record* yang menempati satu *cluster* lebih sedikit dibandingkan dengan *cluster* lain maka *record* tersebut mengandung *outlier*. Untuk data *colon tumor* , *K-Means* mendeteksi terdapat 17 *record* mengandung *outlier* dan 45 *record* tidak mengandung *outlier.* Untuk data *leukemia*, *K-Means mendeteksi* terdapat 24 *record* mengandung *outlier* dan 48 *record* tidak mengandung *outlier.* Berikut adalah perbandingan performansi antara persentase pembagian 70% / 30% dengan *k-fold crossvalidation.*

Tabel 4--4 Perbandingan Performansi pembagian 70/30 dan K-fold Crossvalidation(1)

GA + SVM 70/30 (*Colon Tumor)*	
Atribut Optimal	946
Sensitivity	71%
Specificity	92%
Positif predicted value	83%
Negatif predicted value	85%
Akurasi	84.21%

GA + SVM 3FCV (*Colon Tumor)*	
Atribut Optimal	1003
Sensitivity	75%
Specificity	100%
Positif predicted value	100%
Negatif predicted value	88%
Akurasi	90.91%

Dari hasil yang diperoleh diatas terlihat bahwa 3FCV meningkatkan performansi SVM dengan meningkatnya *sensitivity* dari 71% menjadi 75%, *specificity* dari 92% menjadi 100%, *Positif predicted value* dari 83% menjadi 100%, *Negatif predicted value* dari 84.21% menjadi 88%. Akurasi rata-rata untuk data *leukemia* adalah 100% untuk 3FCV dan 95.45% untuk persentase pembagian 70% dan 30.

Tabel 4-5 Perbandingan Performansi pembagian 70/30 dan K-fold Crossvalidation(2)

GA + SVM 70/30 (*Leukemia)*	
Atribut Optimal	3508
Sensitivity	100%
Specificity	89%
Positive predicted value	93%
Negative predicted value	100%
Akurasi	95.45%

GA + SVM 3FCV (*Leukemia)*	
Atribut Optimal	3472
Sensitivity	100%
Specificity	100%
Positive predicted value	100%
Negative predicted value	100%
Akurasi	100%

Dari hasil yang diperoleh diatas terlihat bahwa 3FCV meningkatkan performansi SVM dengan meningkatnya *specificity* dari 89% menjadi 100% dan *positive predicted value* dari 93% menjadi 100%.

Tabel 4-6 Perbandingan Performansi pembagian 70/30 dan K-fold Crossvalidation(3)

K-Means + GA + SVM 70/30 *(Colon Tumor)*	
Atribut Optimal	950
Sensitivity	80%
Specificity	78%
Positif predicted value	67%
Negatif predicted value	88%
Akurasi	78.57%

K-Means + GA + SVM 3FCV *(Colon Tumor)*	
Atribut Optimal	983
Sensitivity	83%
Specificity	100%
Positif predicted value	100%
Negatif predicted value	89%
Akurasi	92.86%

Dari hasil yang diperoleh diatas terlihat bahwa 3FCV meningkatkan performansi SVM dengan meningkatnya *sensitivity* dari 80% menjadi 83%, *specificity* dari 72% menjadi 100%, *Positif predicted value* dari 67% menjadi 100%, *Negatif predicted value* dari 88% menjadi 89%.

Tabel 4-7 Perbandingan Performansi pembagian 70/30 dan K-fold Crossvalidation(4)

K-Means + GA + SVM 70/30 *(Leukemia)*	
Atribut Optimal	3518
Sensitivity	100%
Specificity	100%
Positif predicted value	100%
Negatif predicted value	100%
Akurasi	100%

K-Means + GA + SVM 3FCV *(Leukemia)*	
Atribut Optimal	3529
Sensitivity	100%
Specificity	100%
Positif predicted value	100%
Negatif predicted value	100%
Akurasi	100%

Untuk data *leukemia*, pada persentase pembagian 70/30 dan 3FCV. performansi yang diperoleh SVM adalah sama untuk kedua skenario.

5. Kesimpulan dan Saran

5.1. Kesimpulan

Dari hasil pengujian dan analisis pada bab sebelumnya dapat disimpulkan bahwa:

1. *Feature selection* yang dilakukan GA menghasilkan kurang lebih setengah dari jumlah atribut sebenarnya yang dilakukan berdasarkan nilai *fitness* tertinggi. Penggunaan parameter yang berbeda membuat pemilihan atribut bervariasi namun akurasi yang didapat rata-rata sama untuk setiap atribut yang terpilih. Pembangkitan individu tergantung dengan kemunculan individu pertama yang memiliki nilai *fitness* tertinggi yang akan terpilih.

2. Penggunaan GA + SVM pada data *colon tumor* menghasilkan atribut terbaik sebesar 997 atribut dengan akurasi 95.23% namun pada kasus data *leukemia* nilai *fitness* yang didapat oleh setiap individu yang terpilih oleh GA selalu menghasilkan akurasi tetap yaitu 95.45% dan 100%.
3. *K-Means* mendeteksi *outlier* berdasarkan jumlah *record* yang berada dalam satu *cluster* Pada data *colon tumor* 17 *record* dinyatakan sebagai *outlier* dan 45 *record* tidak mengandung *outlier*. Pada data *leukemia* 24 *record* dinyatakan sebagai *outlier* dan 48 *record* tidak mengandung *outlier*. Reduksi yang dilakukan *K-Means* berpengaruh terhadap akurasi yang diperoleh, semakin sedikit data semakin sedikit pula data yang digunakan sebagai data latih yang menyebabkan akurasi rendah.
4. *K-fold crossvalidation* lebih baik dibandingkan dengan persentase pembagian data latih dan data uji sebesar 70% dan 30% dalam segi rata-rata akurasi yang dihasilkan dan dapat meningkatkan hasil performansi SVM dan meningkatkan persentase prediksi.
5. Pengujian dengan menggunakan GA + SVM lebih baik dibandingkan menggunakan *K-Means* untuk *dataset colon tumor* sedangkan untuk pengujian dengan menggunakan *K-Means* lebih baik dibandingkan hanya menggunakan GA + SVM untuk *dataset leukemia* dengan akurasi terbaik sebesar 100%.

5.2. Saran

Adapun saran yang penulis utarakan dengan selesainya tugas akhir ini.

1. Diperlukan algoritma lain selain *K-Means clustering* yang cocok untuk mendeteksi *outlier* pada data berdimensi tinggi, karena tidak semua data dapat ditangani oleh *K-Means clustering* yang menyebabkan akurasi pengujian lebih rendah pada *dataset* tertentu.

6. Daftar Pustaka

[1]. Chang, C.-C. & Lin, C.-J., 2015. *https://www.csie.ntu.edu.tw/~cjlin/libsvm/*. [Online] [Accessed 16 Maret 2016].

[2]. Han, J., Kamber, M. & Pei, J., 2012. *Data Mining Concept and Techniquee*. 3rd ed. USA: Elsevier.

[3]. Hermawati, F.A., 2013. *Data Mining*. Yogyakarta: ANDI.

[4]. Hsu, C.-W., Chang, C.-C. & Lin, C.-J., 2016. A Practical Guide to Support Vector Classification. p.4. [5].

Kumar, G.R., Ramachandra, D.G.A. & Nagamani, K., 2014. An Efficient Feature Selection System to Integrating SVM with Genetic Algorithm for Large Medical Datasets. *International Journal of Advanced Research in Computer Science and Software Engineering*.

[6]. Li, J., 2009. *http://levis.tongji.edu.cn/gzli/data/mirror-kentridge.html*. [Online] [Accessed 28 October 2015].

[7]. Madhulatha, T.S., 2012. An Overview on Clustering Methods. *IOSR Journal of Engineering*, 2(4).

[8]. Prasetyo, E., 2014. *DATA MINING Mengolah Data Menjadi Informasi Menggunakan Matlab*. Yogyakarta: ANDI.

[9]. Ratnakar, S., Rajeswari, K. & Jacob, R., 2013. Prediction of Heart Disease Using Genetic Algorithm for Selection of Optimal Reduced Set of Attributes. *International Journal of Advanced Computational Engineering and Networking*, 1(2).

[10]. Shantanam, T. & Padmavathi, M.S., 2014. Comparison of K-Means Clustering and Statistical. *International Conference on Science, Engineering and Management Research*, p.1.

[11]. Shantanam, T. & Padmavathi, M.S., 2015. Application of K-Means and Genetic Algorithms for Dimensional Reduction by Integrating SVM for Diabetes Diagnosis. *ScienceDirect*.

[12]. Srivastava, D.K. & Bhambu, L., 2005-2009. Data Classification Using Support Vector Machine. *Journal of Theoterical and Applied Information Technology*.

[13]. Suyanto, S.M., 2008. *Evolutionary Computing : Komputasi Berbasis "Evolusi" dan "Genetika"*. Bandung: INFORMATIKA.

[14]. Tiwari, R. & Shingh, M.P., 2010. Correlation-based Attribute Selection Using Genetic Algorithm. *International Journal of Computer Applications*, 4(8).

[15]. Velmuragan, D.T., 2012. Efficiency of k-Means and K-Medoids Algorithms for Clustering Arbitrary Data Points. *Int.J.Computer Technology & Applications*, 3(5).

[16]. Verma, G. & Verma, V., 2012. Role and Application of Genetic Algorithm in Data Mining. *International Journal of Computer Applications*, 48(17).

[17]. Zaki, M.J. & Jr., W.M., 2014. *DATA MINING : Fundamental Concepts and Algorithms*. New York: Cambridge University Press.