

## Penerapan Particle Swarm Optimization untuk Penentuan Parameter Regularisasi pada Kernel Regularized Discriminant Analysis

Monica Widiarsi<sup>1,2</sup>, Army Justitia<sup>1</sup>, Agus Zainal Arifin<sup>1</sup>

<sup>1</sup> Laboratorium Vision and Image Processing, Jurusan Teknik Informatika  
Institut Teknologi Sepuluh Nopember (ITS), Surabaya

<sup>2</sup> Universitas Surabaya (Ubaya)

monicawidiarsi@yahoo.com, army\_justitia@yahoo.com, agusza@cs.its.ac.id

### Abstrak

*Metode Kernel Regularized Discriminant Analysis (KRDA) mampu mengatasi permasalahan singularitas dan overfitting yang disebabkan data dengan sampel sedikit berdimensi tinggi. Kinerja KRDA dipengaruhi oleh nilai parameter regularisasi yang digunakan. Paper ini bertujuan mengoptimalkan nilai parameter regularisasi dengan menerapkan Particle Swarm Optimization (PSO) pada KRDA. Metode PSO digunakan untuk memilih parameter regularisasi optimal berdasarkan nilai akurasi klasifikasi pada KRDA. Uji coba menggunakan 4 dataset yaitu iris, wine, dan ionosphere dari UCI, serta dataset svmguide2 dari aplikasi bioinformatika. Hasil uji coba menunjukkan metode PSO-KRDA dapat menentukan nilai parameter regularisasi optimal secara otomatis. Akurasi klasifikasi yang didapatkan dari uji coba menunjukkan PSO-KRDA lebih baik daripada KRDA dan SVM.*

**Kata kunci :** metode kernel, parameter regularisasi, KRDA, PSO

### 1. Pendahuluan

Analisis diskriminan merupakan salah satu metode yang sering digunakan pada klasifikasi. Namun, metode klasik analisa diskriminan, *Linear Discriminant Analysis* (LDA), tidak dapat mengatasi permasalahan singularitas ketika data memiliki dimensi tinggi dengan jumlah sampel yang sedikit [1]. PCA+LDA diusulkan dengan mengurangi dimensi data dikurangi terlebih dahulu menggunakan PCA, kemudian baru diterapkan LDA [2]. Metode RLDA diusulkan dapat mengatasi permasalahan singularitas. RLDA menambahkan pengali dari matriks identitas untuk matriks *total scatter*, yang dapat menstabilkan estimasi matriks kovarian. Informasi tambahan tersebut disebut parameter regularisasi ( $\sigma_R$ ) dengan  $\sigma_R > 0$  [3].

Fitur yang didapatkan dari LDA merupakan kombinasi linier dari data input asli, sehingga tidak dapat menangkap struktur tidak linier pada kumpulan data yang kompleks [4]. LDA diperluas untuk kasus tidak linier dengan menggunakan metode kernel. Metode *Kernel Uncorrelated Discriminant Analysis*

(KUDA) dapat digunakan untuk mengekstrak fitur yang tidak saling berkorelasi di dalam ruang dimensi tereduksi. Namun ketika matriks kernel pada KUDA *non singular*, maka KUDA memproyeksikan semua sampel dalam kelas yang sama ke satu vektor umum di ruang dimensi tereduksi. Jika jumlah sampel tiap kelas sangat besar, maka akan menimbulkan masalah *overfitting*, yang menghasilkan analisa diskriminan menjadi kurang optimal [5].

Untuk mengatasi permasalahan *overfitting* tersebut, Shuiwang Ji dan Jieping Ye (2008) mengusulkan metode *Kernel Regularized Discriminant Analysis* (KRDA) [5]. Namun, kinerja KRDA sangat bergantung pada nilai parameter regularisasi [5]. Akurasi dapat berubah-ubah sesuai dengan nilai parameter regularisasi yang digunakan. Oleh karena itu pemilihan nilai parameter regularisasi yang tepat sangat penting agar didapatkan nilai akurasi klasifikasi yang baik.

Paper ini bertujuan mengoptimalkan nilai parameter regularisasi dengan menerapkan PSO pada KRDA. PSO digunakan untuk menentukan nilai parameter regularisasi optimal berdasarkan akurasi klasifikasi pada KRDA. Diharapkan dengan penerapan PSO pada KRDA, dapat membantu penentuan nilai parameter regularisasi yang menghasilkan akurasi klasifikasi terbaik secara otomatis. Pada uji coba dibandingkan akurasi klasifikasi PSO-KRDA dengan KRDA dan SVM, dan didapatkan akurasi klasifikasi PSO-KRDA lebih baik daripada KRDA dan SVM.

Paper ini disusun dengan struktur sebagai berikut : bagian 2 membahas mengenai LDA. Bahasan mendetail mengenai KRDA dijelaskan pada bagian 3. Bagian 4 menjelaskan mengenai algoritma PSO untuk menentukan parameter regularisasi KRDA. Uji coba dan analisa dibahas pada bagian 5. Kesimpulan dituliskan pada bagian terakhir paper ini.

### 2. Linear Discriminant Analysis (LDA)

LDA merupakan metode untuk mereduksi dimensi data. LDA memproyeksikan data dimensi tinggi ke subruang dimensi yang lebih rendah dengan memaksimalkan pemisahan data dari kelas-kelas yang berbeda dan meminimalkan penyebaran data dari kelas yang sama secara bersamaan.

Jika terdapat matriks data  $X = [x_1, x_2, \dots, x_n] \in \mathfrak{R}^{m \times n}$ , di mana *data point* ke- $i$  adalah  $x_i \in \mathfrak{R}^m$  untuk  $i=1,2,\dots,n$ , LDA mendapatkan matriks transformasi  $G \in \mathfrak{R}^{m \times l}$  yang mentransformasikan  $x_i$  dari ruang dimensi  $m$  ke vektor di ruang dimensi  $l$ , sebagai berikut:

$$x_i \in \mathfrak{R}^m \rightarrow G^T x_i \in \mathfrak{R}^l \quad (1)$$

Diasumsikan bahwa matriks data  $X$  dikelompokkan sebagai  $X = [X_1, \dots, X_k]$ , dimana  $X_i \in \mathfrak{R}^{m \times n_i}$ , terdiri dari  $n_i$  *data point* dari kelas ke- $i$  dan  $\sum_{i=1}^k n_i = n$ . Dalam perhitungan LDA digunakan tiga matriks *scatter* yang disebut matriks *within-class* ( $S_w$ ), *between-class* ( $S_b$ ), dan *total scatter* ( $S_t$ ). Ketiga matriks tersebut didefinisikan sebagai berikut :

$$S_w = \frac{1}{n} \sum_{i=1}^k \sum_{x \in X_i} (x - c_i)(x - c_i)^T \quad (2)$$

$$S_b = \frac{1}{n} \sum_{i=1}^k n_i (c_i - c)(c_i - c)^T \quad (3)$$

$$S_t = \frac{1}{n} \sum_{i=1}^n (x_i - c)(x_i - c)^T \quad (4)$$

dimana  $c_i$  merupakan pusat/vektor rata-rata dari kelas ke- $i$  dan  $c$  merupakan vektor rata-rata global dari  $n$  data sampel. Matriks *total scatter* ( $S_t$ ) dapat didefinisikan dengan  $S_t = S_b + S_w$  [5].

Matriks terpusat didefinisikan sebagai berikut :

$$H_w = \frac{1}{\sqrt{n}} \left[ X_1 - c_1 \left( e^{(1)} \right)^T, \dots, X_k - c_k \left( e^{(k)} \right)^T \right] \quad (5)$$

$$H_b = \frac{1}{\sqrt{n}} \left[ \sqrt{n_1} (c_1 - c), \dots, \sqrt{n_k} (c_k - c) \right] \quad (6)$$

$$H_t = \frac{1}{\sqrt{n}} (X - ce^T) \quad (7)$$

dimana  $e^{(i)} \in \mathfrak{R}^n$  dan  $e \in \mathfrak{R}^n$  adalah vektor satu. Matriks terpusat berhubungan dengan matriks *scatter*  $S_w$ ,  $S_b$ , dan  $S_t$ . Hubungan tersebut dapat dilihat dari persamaan berikut ini:

$$S_w = H_w H_w^T, \quad S_b = H_b H_b^T, \quad \text{dan} \quad S_t = H_t H_t^T. \quad (8)$$

Sifat dari *trace* matriks terpenuhi sebagai berikut :

$$\text{trace}(S_w) = \frac{1}{n} \sum_{i=1}^k \sum_{x \in X_i} \|x - c_i\|_2^2 \quad (9)$$

$$\text{trace}(S_b) = \frac{1}{n} \sum_{i=1}^k n_i \|c_i - c\|_2^2 \quad (10)$$

*Trace* dari  $S_w$  mengukur jarak antara *data point* menuju pusat kelas, dan *trace* dari  $S_b$  mengukur jarak antara masing-masing pusat kelas ke pusat data keseluruhan.

Ketiga matriks *scatter* di ruang dimensi yang telah direduksi ditransformasikan oleh matriks transformasi  $G$ , ditunjukkan sebagai berikut,

$$S_w^L = G^T S_w G, \quad S_b^L = G^T S_b G, \quad \text{dan} \quad S_t^L = G^T S_t G. \quad (11)$$

Transformasi dalam LDA bertujuan untuk memaksimalkan struktur perbedaan data di ruang dimensi yang telah direduksi. Hal ini dapat dicapai dengan meminimalkan  $\text{trace}(S_w^L)$  dan memaksimalkan  $\text{trace}(S_b^L)$  secara bersamaan, sehingga menghasilkan permasalahan optimasi LDA sebagai berikut:

$$G^* = \arg \max_G \left\{ \text{trace} \left( (S_w^L)^{-1} S_b^L \right) \right\} \quad (12)$$

Solusi untuk menyelesaikan permasalahan optimasi di atas sama dengan menyelesaikan permasalahan *generalized eigenvalue* sebagai berikut :

$$S_b y_i = \lambda S_w y_i \quad (13)$$

*Eigenvector* yang bersesuaian dengan  $k-1$  *eigenvalue* terbesar akan membentuk kolom  $G$ . Ketika  $S_w$  *non singular*, permasalahan disederhanakan menjadi permasalahan *eigenvalue* sebagai berikut:

$$S_w^{-1} S_b y_i = \lambda y_i \quad (14)$$

Ketika dimensi data lebih besar daripada jumlah sampel, semua matriks *scatter* menjadi *singular*, sehingga LDA tidak dapat diaplikasikan. Permasalahan ini disebut dengan singularitas pada LDA [1].

### 3. Kernel Regularized Discriminant Analysis (KRDA)

#### 3.1 Metode Kernel

Metode kernel merupakan suatu teknik memetakan dataset dari ruang input ke ruang fitur yang memiliki dimensi tinggi dengan pemetaan tidak linier,  $\phi$ . Sehingga, komponen-komponen tidak linier dapat diekstrak menggunakan algoritma linier di ruang fitur. Pemetaan dari ruang input ke ruang fitur berdimensi tinggi dinotasikan dengan :

$$\phi : X \rightarrow \phi(X) \quad (15)$$

Dalam proses *training* untuk mendapatkan vektor diskriminan bergantung pada *inner product* dari data yang sudah dipetakan pada ruang baru yang berdimensi lebih tinggi. Karena ruang fitur yang sangat tinggi tersebut, maka pemetaan  $\phi$  sangat sulit diketahui dan sangat sulit dipahami secara mudah, sehingga perhitungan *inner product* tersebut dapat digantikan dengan fungsi kernel  $K(\overline{x_i}, \overline{x_j})$  yang mendefinisikan secara implisit pemetaan  $\phi$ . Hal ini disebut "*kernel trick*" yang dirumuskan dengan

$$K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \quad (16)$$

Dengan kernel trick, hanya perlu diketahui fungsi kernel yang akan digunakan tanpa mengetahui fungsi pemetaan nonlinier  $\phi$ . Pada penelitian ini menggunakan fungsi kernel Gaussian. Persamaan dari kernel Gaussian [4] sebagai berikut :

$$K(\overrightarrow{x_i}, \overrightarrow{x_j}) = -\exp\left(-\frac{\|\overrightarrow{x_i} - \overrightarrow{x_j}\|_2^2}{2\sigma^2}\right) \quad (17)$$

dimana  $\sigma$  adalah parameter kernel. Untuk mengoptimalkan klasifikasi, diperlukan penentuan nilai parameter kernel yang sesuai dengan sebaran data yang akan digunakan.

### 3.2 Metode Regularisasi

Metode regularisasi melibatkan penambahan informasi untuk mengatasi permasalahan singularitas. Idenya adalah dengan menambahkan nilai konstan  $\sigma_R$ , yang disebut sebagai parameter regularisasi, ke elemen diagonal dari matriks *total scatter*, dimana  $\sigma_R > 0$ . Regularisasi akan menstabilkan estimasi matriks kovarian sehingga dapat memperbaiki performa klasifikasi.

### 3.3 KRDA

KRDA merupakan metode yang diusulkan untuk mengatasi permasalahan singularitas dan *overfitting*. Kinerja metode KRDA sangat bergantung pada nilai parameter regularisasi dan parameter kernel yang digunakan. Dengan nilai parameter regularisasi dan parameter kernel yang tepat, permasalahan singularitas dan *overfitting* pada *uncorrelated discriminant analysis* dapat teratasi. Jika nilai parameter regularisasi terlalu besar akan mengganggu informasi matriks *scatter*, sedangkan jika terlalu kecil tidak cukup efektif untuk menyelesaikan permasalahan singularitas dan *overfitting* [5].

Algoritma KRDA bertujuan mencari vektor-vektor diskriminan. Vektor diskriminan tersebut merupakan *eigenvector* yang bersesuaian dengan  $(k-1)$  *eigenvalue* maksimum dari  $(S_t^K + \sigma_R K)^+ S_b^K$  yang dihasilkan dari persamaan berikut ini [5] :

$$B^* = \arg \max_B \left\{ \text{trace} \left( \left( B^T (S_t^K + \sigma_R K) B \right)^{-1} B^T S_b^K B \right) \right\} \quad (18)$$

dimana

$$S_t^K = K^2$$

$$S_b^K = K H H^T K = H_b^K \left( H_b^K \right)^T$$

dan nilai parameter regularisasi  $\sigma_R > 0$ .  $K$  merupakan matriks kernel.

Persamaan (18) memperlihatkan kolom dari  $B$  yang diberikan oleh  $(k-1)$  *eigenvector* teratas ekuivalen dengan menyelesaikan permasalahan *generalized eigenvalue*, sebagai berikut :

$$(S_t^K + \sigma_R K)^+ S_b^K \beta_i = \lambda_i \beta_i \quad (19)$$

Properti KRDA dibangun dengan meregularisasi  $S_t^K$  dalam KRDA, yaitu dengan meregularisasi nilai *eigenvalue nonzero*.

Dua langkah dilakukan untuk menghitung *eigenvector* dari  $(S_t^K + \sigma_R K)^+ S_b^K$ , yang merupakan vektor diskriminan pada KRDA [5]. Langkah pertama adalah menghitung *Singular Value Decomposition* (SVD) matriks kernel  $K$  untuk diambil nilai matriks *left singular* ( $U_1$ ), menurut persamaan :

$$K = U_1 \Sigma_r U_1^T \quad (20)$$

dimana  $\Sigma_r \in \mathcal{R}^{r \times r}$  adalah diagonal, dan  $r = \text{rank}(K)$ .

Langkah kedua adalah menghitung SVD dari  $\tilde{\Sigma}^{-1/2} U_1^T H_b^K$ , menurut persamaan berikut :

$$\tilde{\Sigma}^{-1/2} U_1^T H_b^K = U_b \Sigma_b V_b^T \quad (21)$$

dan *eigenvector* diberikan oleh kolom dari matriks  $U_1 \tilde{\Sigma}^{-1/2} U_b$ , dimana  $\tilde{\Sigma} = \Sigma_r^2 + \sigma \Sigma_r$ .

Proses KRDA secara keseluruhan dapat dilihat pada langkah-langkah berikut ini :

1. Pembangkitan matriks kernel *training* ( $K_{\text{Training}}$ ) dan matriks kernel *testing* ( $K_{\text{Testing}}$ ) dengan menggunakan fungsi kernel.
  2. Menghitung matriks kernel terpusat ( $H_b^K$ ) dengan menggunakan  $K_{\text{training}}$ .
  3. Menghitung SVD matriks  $K_{\text{Training}}$  untuk diambil nilai *left singular* matriks ( $U_1$ ),  $K_{\text{Training}} = U_1 \Sigma_r U_1^T$ .
  4. Menghitung matriks kernel terpusat tereduksi,  $H_{b,L} \leftarrow U_1^T H_b^K$  dan  $r = \text{rank}(K_{\text{Training}})$
  5. Menghitung matriks kernel *training* dan matriks kernel *testing* tereduksi,  $K_L^i \leftarrow U_1^T K_{\text{Training}}$  dan  $K_L^i \leftarrow U_1^T K_{\text{Testing}}$ .
  6. Menghitung matriks regularisasi,  $\tilde{\Sigma} \leftarrow (\Sigma_r^2 + \sigma \Sigma_r)^{-1/2}$
  7. Menghitung SVD dari matriks kernel regularisasi,  $\tilde{\Sigma} K_{b,L} = U_b \Sigma_b V_b^T$ .
  8. Menghitung  $G$  yang merupakan matriks diskriminan KRDA,  $G \leftarrow \tilde{\Sigma} U_b$ .
  9. Menghitung matriks transformasi KRDA,  $K_L^i \leftarrow G^T K_L^i$  dan  $K_L^i \leftarrow G^T K_L^i$ .
  10. Lakukan klasifikasi 1NN untuk matriks  $K_L^i$  dan  $K_L^i$  untuk menghitung akurasi klasifikasi.
- Pada uji coba ini digunakan metode 1-*Nearest Neighbour* (1NN) sebagai mesin klasifikasi.

## 4. Penentuan Nilai Regularisasi dengan Particle Swarm Optimization (PSO)

### 4.1 Algoritma PSO

Algoritma PSO diperkenalkan oleh Kennedy dan Eberhart pada tahun 1995, yang diinspirasi oleh perilaku sosial dari binatang, seperti sekumpulan burung dalam suatu gerombolan (*swarm*) [6]. PSO terdiri dari sekumpulan partikel yang mencari posisi terbaik, yang merupakan solusi terbaik untuk masalah optimasi dalam ruang fitur pencarian. Pada setiap generasi, setiap partikel bergerak ke arah posisi lokal terbaik dan posisi global terbaik. Proses pergerakan dan update kecepatan serta posisi dari setiap partikel dinyatakan dengan persamaan [7] :

$$V_{id}^{t+1} = \omega V_{id}^t + C_1 \cdot \text{rand.} (p_{id}^t - x_{id}^t) + C_2 \cdot \text{rand.} (p_{gbest}^t - x_{id}^t) \quad (22)$$

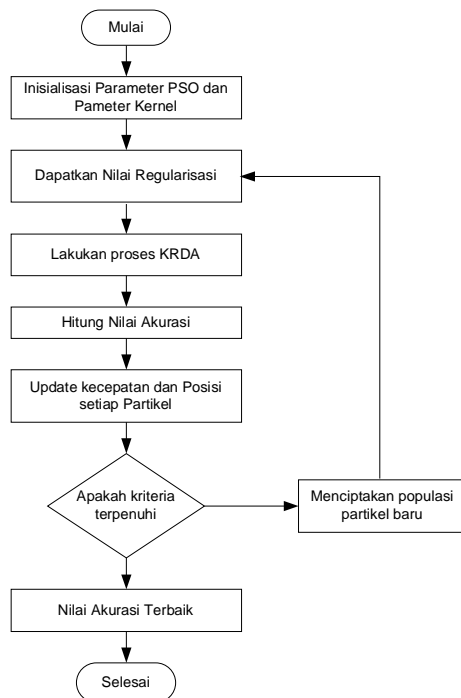
$$X_{id}^{t+1} = X_{id}^t + \alpha v_{id}^t \quad (23)$$

dimana  $t$  = iterasi ke- $t$ ,  $C_1$  and  $C_2$  = faktor pembelajaran,  $rand$  = random angka positif [0... 1] berdistribusi normal,  $\alpha$  = faktor konstrain untuk mengontrol bobot kecepatan,  $\omega$  = koefisien bobot inersia,  $x_{id}$  = posisi partikel  $i$ ,  $v_{id}$  = kecepatan dari partikel  $i$ ,  $p_{id}$  = posisi terbaik dari partikel  $i$ ,  $p_{gbest}$  = posisi terbaik dari semua partikel.

Bobot inersia digunakan untuk mengontrol konvergensi tingkah laku partikel PSO. Untuk mengurangi bobot pada setiap pertambahan iterasi generasi, digunakan persamaan berikut [7] :

$$\omega = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{iter_{max}} * iter \quad (24)$$

dimana  $\omega_{max}$  dan  $\omega_{min}$  adalah bobot inersia maksimum dan minimum yang diperbolehkan,  $iter$  adalah indeks generasi sekarang dan  $iter_{max}$  adalah jumlah maksimum generasi.



**Gambar 1.** Diagram Alir Proses Penentuan Parameter Regularisasi dengan Algoritma PSO

#### 4.2 Penentuan parameter regularisasi KRDA dengan PSO

Penentuan nilai parameter regularisasi  $\sigma_R$  pada KRDA dilakukan oleh PSO. Proses iterasi/pembangkitan generasi PSO dilakukan terhadap perubahan nilai partikel yang menyatakan nilai parameter regularisasi  $\sigma_R$ . Nilai *fitness* tiap partikel diukur dari nilai akurasi klasifikasi pada KRDA menggunakan nilai parameter regularisasi  $\sigma_R$  yang dihasilkan oleh setiap partikel PSO. Untuk setiap

iterasi, nilai partikel diupdate kecepatan dan posisinya, dan dipilih partikel terbaik yang menghasilkan akurasi klasifikasi terbaik. Iterasi generasi berakhir sampai kriteria selesai terpenuhi, dan didapatkan parameter regularisasi optimal yang mempunyai akurasi klasifikasi terbaik. Proses optimasi parameter regularisasi oleh PSO ditunjukkan pada Gambar 1.

#### 5. Uji Coba dan Analisa

Uji coba menggunakan 4 dataset, yaitu *iris*, *wine*, *ionosphere* yang didapat dari *UCI Machine Learning Repository* [8] serta dataset *svmguide2* dari aplikasi bioinformatika [9]. Deskripsi dari masing-masing dataset dapat dilihat dari Tabel 1. Parameter PSO yang digunakan pada uji coba ini sebagai berikut : jumlah generasi diset 30, jumlah partikel sebanyak 30, koefisien  $C_1$  dan  $C_2$  masing-masing ditentukan 0.2, sedangkan bobot inersia masing-masing  $\omega_{max} = 0.9$  dan  $\omega_{min} = 0.4$ .

Pada uji coba akan dianalisa kinerja penerapan PSO pada KRDA terhadap dua macam data training, yaitu data *training singular* dan *non singular*.

Data	Iris	Wine	Ionosphere	Svmguide2
Record	150	178	351	391
Atribut	4	13	34	20
Kelas	3	3	2	3

Pemilihan sampel data *training singular* dan *non singular* dilakukan secara random dan merata untuk semua kelas. Data *singular* adalah data yang mempunyai jumlah sampel kurang dari jumlah atribut dataset. Jumlah sampel data *training singular* yang digunakan untuk dataset Iris sejumlah 3 record, dataset Wine sejumlah 12 record, dataset Ionosphere sejumlah 32 record, dan dataset SVMGuide2 sejumlah 18 record. Dataset *non singular* yang digunakan pada uji coba adalah data yang dipartisi secara random ke dalam data *training* dan data *testing* dengan rasio sebesar 1:1.

Uji coba dilakukan sebanyak 5 kali percobaan untuk setiap dataset. Untuk setiap percobaan, ditentukan terlebih dahulu nilai parameter kernel ( $\sigma$ ) yang akan digunakan sebagai input pada metode PSO-KRDA. Penentuan nilai  $\sigma$  disesuaikan dengan data percobaan yang digunakan. Setelah itu, metode PSO-KRDA dijalankan untuk mendapatkan nilai parameter regularisasi ( $\sigma_R$ ) optimal yang mempunyai akurasi klasifikasi terbaik. Hasil dari penentuan nilai  $\sigma$ ,  $\sigma_R$ , dan akurasi terbaik menggunakan PSO-KRDA untuk setiap dataset dapat dilihat pada Tabel 2. Kemudian, untuk setiap dataset *non singular*, hasil akurasi klasifikasi rata-rata menggunakan PSO-KRDA dibandingkan dengan akurasi klasifikasi menggunakan metode KRDA dan algoritma SVM pada penelitian [5]. Data training yang digunakan pada penelitian [5] adalah data *non singular* dengan proporsi yang sama dengan percobaan PSO-KRDA (rasio data *training*:data *testing* = 1:1). Perbandingan

akurasi klasifikasi rata-rata yang dihasilkan oleh algoritma PSO-KRDA, KRDA dan SVM dapat dilihat

Pada Tabel 2 dapat dilihat kinerja penerapan PSO pada KRDA dalam penentuan  $\sigma_R$ . PSO dapat membantu menentukan  $\sigma_R$  optimal secara otomatis. Nilai  $\sigma_R$  optimal bergantung pada nilai  $\sigma$  sebagai input PSO dan data yang digunakan. Hasil akurasi klasifikasi menggunakan PSO-KRDA untuk semua dataset *non singular* sudah baik. Namun, akurasi klasifikasi untuk dataset *singular* pada dataset Ionosphere masih belum baik. Oleh karena itu,

perbaikan masih perlu dikembangkan untuk penelitian selanjutnya.

Pada Tabel 3, akurasi klasifikasi rata-rata PSO-KRDA lebih baik dibandingkan dengan klasifikasi KRDA dan klasifikasi SVM. Hal ini menunjukkan penerapan PSO pada KRDA membantu optimasi  $\sigma_R$  secara otomatis, serta dapat memperbaiki akurasi klasifikasi dibandingkan dengan menggunakan KRDA. Sedangkan SVM dalam klasifikasi hanya menggunakan nilai  $\sigma$ , dan didapatkan dari uji coba hasil akurasi tidak sebaik PSO-KRDA. Hal ini menunjukkan bahwa parameter regularisasi yang tepat dapat meningkatkan akurasi klasifikasi.

**Tabel 1.** Parameter Kernel dan Parameter Regularisasi Optimal Masing-masing Dataset

Dataset	Data training	Parameter Kernel ( $\sigma$ )	Parameter Regularisasi ( $\sigma_R$ )	PSO-KRDA (%)
Iris	Singular	0.63	0.1	95.92
	Non Singular	0.41	0.1	100
Wine	Singular	0.7	0.4	98.80
	Non Singular	0.23	0.1	100
Ionosphere	Singular	0.94	0.01	67.40
	Non Singular	0.83	0.9	98.29
Svmguide2	Singular	0.19	0.19	74.93
	Non Singular	0.11	0.1	83.67

**Tabel 2.** Perbandingan Akurasi Klasifikasi Rata-rata antara PSO-KRDA, KRDA dan SVM

Dataset	PSO-KRDA (%)	KRDA (%)	SVM (%)
Iris	98.67	94.67	98.13
Wine	99.77	96.37	98.18
Ionosphere	96.46	92.69	95.89
Svmguide2	82.35	81.29	81.53

## 6. Kesimpulan

Penentuan nilai parameter kernel dan parameter regularisasi sangat berpengaruh dalam akurasi klasifikasi. Dengan nilai parameter kernel yang tepat sebagai input PSO, maka penerapan PSO pada KRDA dapat membantu menemukan parameter regularisasi yang optimal secara otomatis. Akurasi klasifikasi menggunakan PSO-KRDA lebih baik dibandingkan dengan KRDA dan SVM.

Ujicoba PSO-KRDA pada paper ini hanya berfokus pada dataset standar. Pengembangan penelitian ini dapat dilakukan untuk penerapan PSO-KRDA dalam penentuan parameter PSO maupun parameter kernel yang optimal. Juga dapat dikembangkan untuk aplikasi yang lebih kompleks dan tidak linier, seperti aplikasi pengenalan wajah dan klasifikasi teks.

## 7. Referensi

- [1] Chen, Li-Fen dkk., "A New LDA-based Face Recognition System which an Solve the Small Sample Size Problem", *Pattern Recognition* 3, pp. 1713-1726, 2000.
- [2] Belhumeur, P.N., dkk., "Eigenfaces versus Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, July 1997.
- [3] Friedman, J.H., "Regularized Discriminant Analysis", *Journal of the American Staistical Association*, vol. 84, no. 405, pp.165-175, 1989.
- [4] Herbrich, R., "Learning Kernel Classifiers Theory and Algorithms", *The MIT Press*, Massachusetts Institute of Technology, 2002.
- [5] Ji, Shuiwang dan Ye, Jieping, "Kernel Uncorrelated and Regularized Discriminant Analysis : A Theoretical and Computational Study", *IEEE Transaction on Knowledge and Data Engineering*, Vol. 20, No. 10, pp. 1311-1321, Oktober 2008.

- [6] Kennedy, J. dan Eberhart, R.C., "Particle Swarm Optimization", *Proceedings of the IEEE International Conference on Neural Networks IV*, pp. 1942–1948, Piscataway: IEEE, 1995.
- [7] Marinakis, Yanis., dkk., "Ant Colony and Particle Swarm Optimization for Financial Classification Problems", *Expert Systems with Applications*, no. 36, pp. 10604-10611, 2009.
- [8] D.J. Newman, S. Hettich, C.L. Blake, and C.J. Merz, "UCI Repository of Machine Learning Databases," <http://archive.ics.uci.edu/ml/>, 1998.
- [9] Gardy, J.L., dkk., "PSORT-B: Improving Protein Subcellular Localization Prediction for Gram-Negative Bacteria", *NucleicAcids Research*, vol.31 no.13, pp. 3613-3617, 2003.