

# Perbandingan *Quadratic Discriminant Analysis* dan *Support Vector Machine* untuk Klasifikasi Tutupan Lahan di DKI Jakarta

Kamaluddin Junianto Dimas\*, Rahma Anisa\*, Itasia Dina Sulvianti\*

\*Departemen Statistika Institut Pertanian Bogor

**Abstrak**—DKI Jakarta is a center of government as well as economy and business of Indonesia, thus development projects in Jakarta continue every year. Therefore, monitoring for land use has to be improved in accordance to DKI Jakarta Spatial Planning. The attempt needs to be supported by continuous data availability regarding land cover condition in Jakarta. The aforementioned data collecting process become easier due to remote sensing technology development. Remote sensing technology can be utilized for analyzing the size of land use area by using classification analysis. It has been found that the level of accuracy depends on the type of classification method and number of training data. This research evaluated the level of overall accuracy, sensitivity, and specificity of Quadratic Discriminant Analysis (QDA) and Support Vector Machine (SVM) along with number of data training used in classifying Jakarta land cover in 2017. The results showed that in both methods, the variance of all the aforementioned criteria were getting smaller along with the increasing number of training data. QDA and SVM had similar performance based on overall accuracy and specificity. However, SVM was better than QDA on sensitivity.

**Kata kunci**—klasifikasi; diskriminan; jakarta; support vector machine; tutupan lahan

## I. PENDAHULUAN

### A. Latar Belakang

Banyaknya penduduk DKI Jakarta tahun 2016 berdasarkan data Badan Pusat Statistik Provinsi DKI Jakarta (2017) mencapai 10 juta jiwa dengan laju pertumbuhan penduduk per tahun sebesar 1.07%. Pertambahan banyaknya penduduk akan diikuti peningkatan kegiatan pembangunan di Jakarta. Dalam rangka menata kegiatan pembangunan di kota untuk menjaga interaksi yang harmonis dan seimbang antar manusia dengan lingkungan, salah satu upaya pemerintah adalah

mengeluarkan peraturan daerah tentang Rencana Tata Ruang Wilayah (RTRW) yang berlaku untuk jangka waktu 20 tahun dan ditinjau kembali setiap 5 tahun. Peraturan tersebut mengatur penyediaan lahan untuk kegiatan pembangunan serta penyediaan Ruang Terbuka Hijau (RTH) di wilayah kota.

Berdasarkan data Katadata (2017), pada Januari 2017, proporsi luas Ruang Terbuka Hijau (RTH) di Jakarta mencapai sekitar 9.98% dari total luas DKI Jakarta. RTH tersebut terdiri atas taman kota, taman lingkungan, taman interaktif, dan jalur hijau. Undang-Undang Nomor 26 Tahun 2007 menetapkan penyediaan RTH minimal 30% dari luas wilayah kota. Oleh karena itu, perlunya peningkatan pengawasan terhadap penggunaan lahan agar penggunaannya tetap sesuai dengan RTRW. Usaha tersebut perlu ditunjang dengan tersedianya data secara kontinu mengenai kondisi tutupan lahan di kota.

Perkembangan ilmu pengetahuan dan teknologi memudahkan proses pengumpulan data mengenai kondisi tutupan lahan dengan teknologi penginderaan jauh. Teknologi penginderaan jauh dapat dimanfaatkan untuk pemetaan tutupan lahan dengan analisis klasifikasi. Berbagai studi pada klasifikasi tutupan lahan telah dilakukan untuk mengetahui tingkat akurasi yang dihasilkan setiap metode serta faktor-faktor yang mempengaruhi. Muhammad (2014) melakukan sebuah studi untuk mengkaji performa beberapa metode klasifikasi terhadap tingkat keragaman tutupan lahan yang berbeda. Hasil studi Muhammad menunjukkan bahwa metode klasifikasi *Maximum Likelihood* atau yang dikenal sebagai *Quadratic Discriminant Analysis* (QDA) menghasilkan tingkat akurasi paling tinggi untuk setiap tingkat keragaman tutupan

lahan. Noi and Kappas (2018) melakukan studi untuk mengkaji tingkat akurasi beberapa metode klasifikasi nonparametrik dengan faktor berupa banyaknya data *training* yang digunakan dan data *training* yang seimbang dan tidak seimbang. Hasil studi tersebut menunjukkan bahwa metode klasifikasi *Support Vector Machine* (SVM) menghasilkan tingkat akurasi paling tinggi untuk setiap banyaknya dan kondisi data *training*. Berdasarkan penelitian-penelitian tersebut, setiap metode klasifikasi dapat menghasilkan tingkat akurasi yang berbeda. Akurasi ini pun juga bergantung pada data *training* dan tingkat keragaman pada citra yang dikaji.

Penelitian ini mengkaji tingkat akurasi dari metode QDA dan SVM untuk klasifikasi kondisi tutupan lahan vegetasi dan nonvegetasi di DKI Jakarta tahun 2017. Selain itu, tingkat akurasi dikaji berdasarkan banyaknya data *training* yang digunakan.

## B. Tujuan

Penelitian ini bertujuan :

- 1) Membandingkan metode *Quadratic Discriminant Analysis* dan *Support Vector Machine* untuk klasifikasi tutupan lahan vegetasi dan nonvegetasi di provinsi DKI Jakarta tahun 2017.
- 2) Melihat pengaruh ukuran data *training* terhadap kinerja dari masing-masing metode.

## II. TINJAUAN PUSTAKA

### A. Quadratic Discriminant Analysis

Metode *Quadratic Discriminant Analysis* (QDA) dikenal juga sebagai metode *Maximum Likelihood Classification* (MLC) (Richards (2012)). Metode ini umum digunakan dalam pengolahan data citra penginderaan jauh. Metode ini dikembangkan berdasarkan teorema Bayes, dimana peluang piksel akan diklasifikasi sebagai kelas ke- $i$  berdasarkan nilai  $\mathbf{x}$  dinyatakan sebagai berikut:

$$p(y_i|\mathbf{x}) = \frac{p(y_i)f_i(\mathbf{x})}{\sum_{i=1}^2 p(y_i)f_i(\mathbf{x})}, i = 1, 2 \quad (1)$$

dengan:

- $y_i$  : kelas ke- $i$
  - $\mathbf{x}$  : vektor nilai piksel
  - $p(y_i)$  : peluang terpilihnya piksel kelas ke- $i$  secara acak dari populasi
  - $f_i(\mathbf{x})$  : fungsi kepekatan  $\mathbf{x}$  untuk piksel yang berasal dari kelas ke- $i$
- $\mathbf{x}$  dianggap menyebar mengikuti sebaran normal ganda sehingga:

$$f_i(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{p}{2}}|\Sigma_i|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i)\right) \quad (2)$$

dengan:

- $p$  : dimensi  $\mathbf{x}$
- $\Sigma_i$  : matriks ragam peragam kelas ke- $i$
- $\boldsymbol{\mu}_i$  : vektor nilai tengah kelas ke- $i$

Jika fungsi log diterapkan terhadap persamaan (1) dan nilai penyebut dianggap sama untuk setiap  $p(y_i|\mathbf{x})$ , maka:

$$\begin{aligned} g_i(\mathbf{x}) &= \ln p(y_i) - \frac{1}{2} \ln |\Sigma_i| - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i) \\ g_i(\mathbf{x}) &= \ln p(y_i) - \frac{1}{2} \ln |\Sigma_i| - \frac{1}{2} \mathbf{x}^T \Sigma_i^{-1} \mathbf{x} + \mathbf{x}^T \Sigma_i^{-1} \boldsymbol{\mu}_i - \\ &\quad \frac{1}{2} \boldsymbol{\mu}_i^T \Sigma_i^{-1} \boldsymbol{\mu}_i \end{aligned} \quad (3)$$

Fungsi  $g_i(\mathbf{x})$  adalah fungsi diskriminan untuk kelas ke- $i$ . Suatu piksel akan diklasifikasi ke dalam kelas ke- $i$  berdasarkan nilai  $g_i(\mathbf{x})$  terbesar (James et al. (2013)).

### B. Support Vector Machine

*Support Vector Machine* (SVM) adalah salah satu metode klasifikasi terbimbing nonparametrik. Metode SVM dipopulerkan oleh Vapnik dalam bukunya, *Statistical Learning Theory*. SVM bekerja dengan membuat sebuah bidang pemisah atau *hyperplane* yang dapat memisahkan data berdasarkan kelasnya. Selanjutnya akan dijelaskan algoritme untuk membuat sebuah *hyperplane* yang linier (Burges (1998)).

Pada kondisi data yang dapat dipisahkan secara sempurna, fungsi dari *hyperplane* dapat dinotasikan sebagai berikut:

$$y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1 \geq 0 \quad (4)$$

dengan:

- $y_i$  : kelas tutupan lahan ke- $i$ ,  $y_i \in \{-1, 1\}$   
 $\mathbf{w}$  : vektor normal terhadap *hyperplane*  
 $\mathbf{x}_i$  : vektor data *training* piksel ke- $i$ ,  $i = 1, \dots, n$   
 $b$  : jarak antara titik pusat dengan *hyperplane*

Setiap *hyperplane* memiliki nilai *margin* sebesar  $\frac{2}{\|\mathbf{w}\|}$ . *Optimal hyperplane* didapat dengan memaksimalkan jarak antara *hyperplane* dengan piksel terdekat dari *hyperplane*. *Margin* maksimum didapat dengan meminimumkan  $\|\mathbf{w}\|^2$  sehingga fungsi objektifnya dapat ditulis sebagai:

$$\min\left\{\frac{1}{2}\|\mathbf{w}\|^2\right\} \quad (5)$$

dengan kendala

$$y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1 \geq 0$$

Masalah optimasi pada persamaan (5) dapat diselesaikan dengan metode Lagrangian. Persamaan Lagrangian yang dapat dibentuk berdasarkan persamaan (5) yaitu:

$$LP = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^n \alpha_i (y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1), \alpha_i \geq 0 \quad (6)$$

$LP$  disebut sebagai *Lagrangian Primal Problem*,  $\alpha_i$  adalah koefisien pengali Lagrangian.  $LP$  dapat diturunkan terhadap  $\mathbf{w}$  dan  $b$ . Selanjutnya, melalui beberapa operasi aritmatika didapatkan persamaan:

$$LD = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \quad (7)$$

$LD$  disebut sebagai *Lagrangian Dual Problem* dan  $\mathbf{x}_j$  adalah vektor *training* piksel ke- $j$ . Melalui  $LD$ , nilai  $\alpha_i$  dari tiap piksel dapat diketahui. Jika suatu piksel memiliki nilai  $\alpha_i > 0$ , piksel tersebut disebut sebagai *support vector*. *Support vector* adalah piksel yang posisinya terletak di *hyperplane* dan piksel tersebut menentukan bentuk dari *hyperplane*. Fungsi disriminan SVM yaitu:

$$f(x) = \text{sgn}\left(\sum_{i=1}^{ns} \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b\right) \quad (8)$$

dan  $b$  adalah Chang and Lin (2011):

$$b = -\frac{1}{ns} \sum_{i=1}^{ns} y_i \left(\sum_{j=1}^{ns} \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j - 1\right) \quad (9)$$

dengan:

- $\mathbf{x}$  : vektor piksel yang akan ditentukan kelasnya  
 $\mathbf{x}_i, \mathbf{x}_j$  : vektor data *training* piksel ke- $i, j$  yang menjadi *support vector*  
 $ns$  : banyaknya data *training* yang merupakan *support vector*

Algoritme di atas dapat digunakan untuk data yang terpisah secara sempurna. Ketika kondisi tersebut tidak dipenuhi, peubah  $\xi_i$ , yang disebut sebagai *slack variable*, ditambahkan ke dalam formulasi SVM. Fungsi objektif pada persamaan (5) menjadi:

$$\min\left\{\frac{1}{2}\|\mathbf{w}\|^2 + C \sum_i \xi_i\right\} \quad (10)$$

dengan kendala

$$y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0$$

Parameter  $C$  adalah parameter besarnya penalti akibat kesalahan klasifikasi yang nilainya ditentukan oleh peneliti. Persamaan *Lagrangian Primal* berdasarkan fungsi objektif di atas yaitu :

$$LP = \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_i \xi_i - \sum_{i=1}^n \alpha_i (y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i) - \mu_i \sum_i \xi_i \quad (11)$$

dengan kendala

$$\alpha_i \geq 0, \xi_i \geq 0$$

Koefisien  $\mu_i$  merupakan koefisien pengali Lagrangian untuk  $\xi_i$ .  $LP$  dapat diturunkan terhadap  $\mathbf{w}$ ,  $b$ , dan  $\xi_i$ . Selanjutnya, melalui beberapa operasi aritmatika didapatkan persamaan:

$$LD = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j, 0 \leq \alpha_i \leq C \quad (12)$$

Jika tidak memungkinkan untuk memisahkan piksel dengan *hyperplane* yang linier, data terlebih dahulu ditransformasi ke dalam ruang dimensi yang lebih tinggi menggunakan fungsi kernel  $K(\mathbf{x}_i, \mathbf{x}_j)$ . Berdasarkan hasil studi yang dilakukan Kavzoglu and Colkesen (2009) serta Cho et al. (2017), penelitian ini akan memakai fungsi kernel RBF pada metode SVM. Fungsi kernel RBF

memiliki rumus:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp -\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2 \quad (13)$$

dengan  $\gamma > 0$  adalah parameter dari fungsi kernel RBF. Fungsi diskriminan SVM yang menggunakan fungsi kernel yaitu :

$$f(x) = \text{sgn}\left(\sum_{i=1}^{ns} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b\right) \quad (14)$$

### III. METODE

#### A. Data

Penelitian ini menggunakan data Landsat 8 yang direkam pada tanggal 19 Juli dan 4 Agustus 2017 dengan *path/row* 122/064 yang diperoleh dari situs USGS LandsatLook (<https://landsatlook.usgs.gov>). Peubah penjelas pada penelitian ini adalah *band* 4, 5, dan 6 dengan resolusi 30 m. Peubah respons yang diamati adalah bentuk tutupan lahan yang terdiri atas vegetasi dan nonvegetasi. Unit amatannya adalah piksel. Pemilihan *band* yang digunakan pada penelitian ini merujuk pada penelitian yang dilakukan oleh Putri (2017). Data penunjang yang digunakan adalah Peta Administrasi Provinsi DKI Jakarta.

#### B. Prosedur Analisis Data

Proses analisis data akan menggunakan perangkat lunak R 3.4.3, dan QGIS 3.03. Langkah-langkah yang dilakukan pada penelitian ini adalah sebagai berikut:

##### 1) Praproses Citra.

- a) Melakukan komposit citra untuk mendapatkan tampilan visual citra yang berwarna. Kombinasi *band* yang digunakan untuk komposit citra yaitu kombinasi 6-5-4.
- b) Koreksi Citra  
Koreksi citra dilakukan dengan membuat *mask* untuk tutupan awan menggunakan metode *blueband* dengan *threshold* 14000 dan *mask* untuk bayangan awan. Piksel yang tertutup awan akan diganti dengan piksel pada lokasi yang sama dari data citra Landsat 8

tanggal 4 Agustus 2017. Metode tersebut dilakukan berdasarkan petunjuk penggunaan QGIS yang ditulis oleh L (2016).

- c) Melakukan pemotongan citra untuk mendapatkan data citra yang sesuai dengan Peta Administrasi Provinsi DKI Jakarta. Wilayah Kepulauan Seribu tidak dimasukkan ke dalam penelitian karena kondisi tutupan lahan yang menjadi fokus pengamatan adalah tutupan lahan di wilayah perkotaan.

##### 2) Melabelkan data

Tabel I  
DEFINISI AREA VEGETASI DAN NONVEGETASI

Area	Definisi
Vegetasi	Taman kota, taman wisata alam, taman rekreasi, taman lingkungan perumahan dan permukiman, taman lingkungan perkantoran dan gedung komersial, taman hutan raya, hutan lindung, kebun binatang, pemakaman umum, lahan pertanian, kawasan jalur hijau, semak dan belukar, lapangan olahraga, area penyangga lapangan udara
Nonvegetasi	Lahan terbuka, permukiman, jaringan jalan, pelabuhan, danau atau waduk, rawa, kali

Pelabelan data adalah upaya untuk mempersiapkan data *training*. Proses pelabelan dilakukan dengan menggunakan analisis gerombol *K-means*. Gerombol-gerombol yang terbentuk akan digabungkan menjadi gerombol vegetasi dan gerombol nonvegetasi. Setelah itu, sebanyak 1% piksel dari tiap gerombol akan dipilih secara acak untuk divalidasi menggunakan data historis wilayah Jakarta yang tersedia pada *Google Earth* sehingga akan didapatkan data piksel yang telah diberi label dengan benar. Definisi vegetasi dan nonvegetasi yang digunakan dalam penelitian ini dapat dilihat pada Tabel I. Definisi tersebut mengacu pada Peraturan Menteri Dalam Negeri Nomor 1 Tahun 2007 dan Standar Klasifikasi Penutupan Lahan yang dikeluarkan oleh Badan Informasi Geospasial.

3) Menyusun data *training* dan data *testing*

Tabel II  
ALOKASI BANYAKNYA DATA TRAINING UNTUK KLASIFIKASI

Kelas	Banyaknya piksel per skenario							
Vegetasi	20	30	50	100	250	500	750	1000
Nonvegetasi	20	30	50	100	250	500	750	1000

Piksel yang telah melalui proses pembelajaran akan digunakan untuk menyusun data *training* dan data *testing*. Sebanyak 10% dari piksel tersebut akan dipilih secara acak untuk menyusun data *testing* dan 90% piksel lainnya akan digunakan untuk menyusun data *training*. Untuk mengetahui pengaruh banyaknya data *training* yang digunakan terhadap hasil akurasi klasifikasi, banyaknya piksel data *training* ditentukan dengan skenario yang dijabarkan dalam Tabel II.

4) Melakukan klasifikasi dengan metode QDA dengan menggunakan skenario data *training* pada Tabel II. Setiap skenario dilakukan sebanyak 100 ulangan.

5) Melakukan klasifikasi dengan metode SVM Langkah pertama yang dilakukan adalah menentukan parameter  $C$  pada persamaan (10) dan  $\gamma$  pada persamaan (13) yang akan dipakai untuk setiap skenario data *training*. Nilai  $C$  dan  $\gamma$  yang dicobakan yaitu  $10^{-9}$ ,  $10^{-6}$ ,  $10^{-3}$ ,  $10^0$ ,  $10^3$  untuk parameter  $\gamma$  dan  $10^{-2}$ ,  $10^0$ ,  $10^2$ ,  $10^4$ ,  $10^6$  untuk parameter  $C$  sehingga terdapat 25 kombinasi parameter. Kinerja 25 kombinasi 25 parameter tersebut diukur dengan metode *cross validation*. Nilai *k-fold* yang digunakan yaitu 10 *fold*. Proses penentuan parameter diulang 100 kali pada tiap skenario mengingat terdapat 100 set data *training* untuk tiap skenario. Setiap kombinasi parameter akan menghasilkan 100 nilai *error* pada tiap skenario Kombinasi parameter yang memberikan rata-rata nilai *error* terkecil pada masing-masing skenario akan digunakan untuk membangun model SVM

6) Mengevaluasi hasil klasifikasi Hasil kinerja klasifikasi dapat dirangkum

dalam bentuk matriks yang disebut sebagai matriks konfusi (Tabel III).

Tabel III  
MARIKS KONFUSI

Kelas Hasil Prediksi	Kelas Aktual	
	Vegetasi	Nonvegetasi
Vegetasi	A	B
Nonvegetasi	C	D

dengan:

- A : banyaknya piksel vegetasi yang diprediksi sebagai vegetasi
- B : banyaknya piksel nonvegetasi yang diprediksi sebagai vegetasi
- C : banyaknya piksel vegetasi yang diprediksi sebagai nonvegetasi
- D : banyaknya piksel nonvegetasi yang diprediksi sebagai nonvegetasi

Matriks tersebut dapat digunakan untuk menghitung nilai *overall accuracy*, *sensitivity*, *specificity* dengan persamaan sebagai berikut:

$$OverallAccuracy = \frac{A + D}{A + B + C + D} \tag{15}$$

$$Sensitivity = \frac{A}{A + C} \tag{16}$$

$$Specificity = \frac{D}{B + D} \tag{17}$$

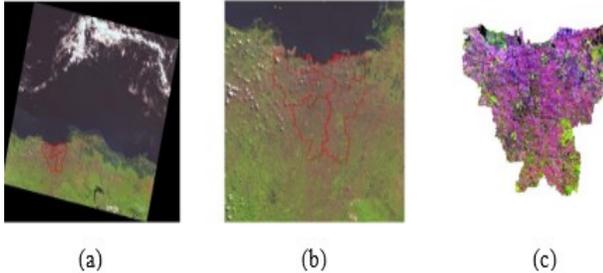
Berdasarkan persamaan di atas, nilai *overall accuracy*, *sensitivity*, *specificity* memiliki skala 0 sampai 1. Semakin besar nilainya, semakin baik kinerja klasifikasinya

7) Membandingkan nilai *Overall Accuracy*, *Sensitivity*, dan *Specificity* pada kedua metode

Proses membandingkan nilai-nilai tersebut menggunakan diagram kotak garis. Uji t digunakan apabila perbedaan nilai kedua metode tidak terlihat pada diagram kotak garis.

#### IV. HASIL DAN PEMBAHASAN

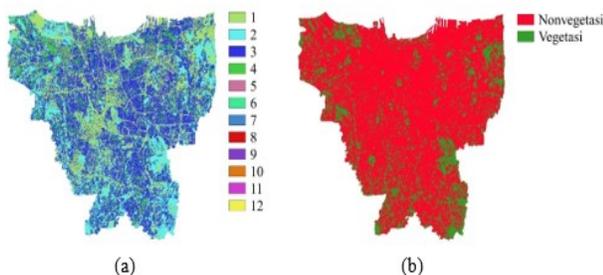
##### A. Praproses Citra



Gambar 1. Data citra sebelum dan sesudah praproses citra: (a) Data citra hasil komposit citra (b) Area citra yang akan dipotong (c) Data citra sesudah pemotongan citra

Berdasarkan Gambar 1(a), terdapat beberapa daerah tertutup oleh awan dan bayangan, termasuk wilayah Jakarta. Piksel yang tertutup awan dan bayangan ditangani dengan membuat *mask* untuk awan dengan menggunakan metode *blueband* serta membuat *mask* untuk bayangan. Selanjutnya, piksel yang tertutup awan dan bayangan digantikan dengan piksel pada lokasi yang sama dari data citra Landsat tanggal 4 Agustus 2017. Pada akhir praproses, data citra dipotong sesuai dengan Peta Administrasi Provinsi DKI Jakarta. Praproses citra menghasilkan data citra yang terdiri atas 712530 piksel

##### B. Pelabelan Data



Gambar 2. Proses penggerombolan dengan *k-means*: (a) 12 gerombol (b) 2 gerombol

Proses pelabelan data dilakukan dengan bantuan metode penggerombolan *K-means* yang dapat dilihat pada Gambar 2. Penggerombolan dilakukan

dengan nilai  $k$  yaitu 12, 24, 36, dan 48. Hasil penggerombolan dievaluasi dengan melihat rasio antara total keragaman antar gerombol (BSS) dengan total keragaman dalam gerombol (WSS) dan nilai Calinski-Harabaz (CH) yang ditampilkan dalam Tabel IV.

Tabel IV  
HASIL PENGGEROMBOLAN DENGAN K-MEANS

Banyaknya $k$	Rasio WSS dan BSS	Nilai CH
12	1.909	123690.160
24	3.107	96262.940
36	4.099	83449.190
48	4.556	69073.820

Tabel 4 menunjukkan bahwa semakin banyak nilai  $k$ , nilai rasio BSS dan WSS semakin kecil. Hal tersebut disebabkan semakin banyak gerombol yang dipakai, total keragaman dalam gerombolnya semakin kecil dan antar gerombolnya semakin besar. Namun, penggerombolan dengan nilai  $k = 12$  menghasilkan nilai CH terbesar. Nilai tersebut menunjukkan bahwa penggerombolan dengan nilai  $k = 12$  merupakan penggerombolan yang terbaik. Kedua belas gerombol tersebut digabung menjadi 2 gerombol, yaitu gerombol vegetasi dan gerombol nonvegetasi. Gerombol vegetasi terdiri atas 138649 piksel dan gerombol nonvegetasi terdiri atas 573881 piksel.

Selanjutnya, tiap gerombol diambil piksel secara acak sebanyak 1% dari banyaknya piksel tiap gerombol, yaitu 1584 piksel vegetasi dan 5937 piksel nonvegetasi. Piksel yang terpilih divalidasi menggunakan data historis wilayah Jakarta tahun 2017. Proses validasi menghasilkan 1632 piksel vegetasi dan 5888 piksel nonvegetasi yang akan digunakan untuk membuat data *training* dan data *testing*. Sebanyak 10% piksel tersebut dipilih secara acak untuk menyusun data *testig* dan sebanyak 90% sisanya akan digunakan untuk menyusun data *training*. Data *testing* tersusun dari 164 piksel vegetasi dan 589 piksel nonvegetasi

##### C. Klasifikasi dengan QDA dan SVM

Pada proses klasifikasi menggunakan QDA, tidak ada parameter yang nilainya harus ditentukan

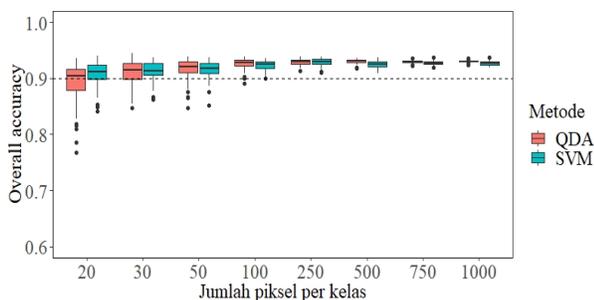
oleh peneliti. Nilai parameter tersebut ditentukan oleh statistik masing-masing data *training*. Hasil akhir dari proses ini yaitu 800 model QDA yang terdiri atas 100 model untuk tiap skenario.

Proses klasifikasi menggunakan SVM diawali dengan menentukan parameter  $C$  dan  $\gamma$  yang akan dipakai. Nilai  $\gamma$  dan  $C$  yang dicobakan yaitu  $10^{-9}$ ,  $10^{-6}$ ,  $10^{-3}$ ,  $10^0$ , dan  $10^3$  untuk parameter  $\gamma$  dan  $10^{-2}$ ,  $10^0$ ,  $10^2$ ,  $10^4$ , dan  $10^6$  untuk parameter  $C$  sehingga terdapat 25 kombinasi parameter. Hasil proses penentuan parameter ditampilkan dalam Tabel V. Kombinasi parameter yang diinginkan yaitu kombinasi yang menghasilkan rata-rata nilai *error* terkecil pada masing-masing skenario data *training*. Kombinasi tersebut akan digunakan untuk membuat model SVM untuk setiap set data *training*. Hasil akhir dari proses ini yaitu 800 model SVM yang terdiri atas 100 model untuk setiap skenario.

Tabel V  
KOMBINASI PARAMETER SVM TERPILIH UNTUK SETIAP SKENARIO

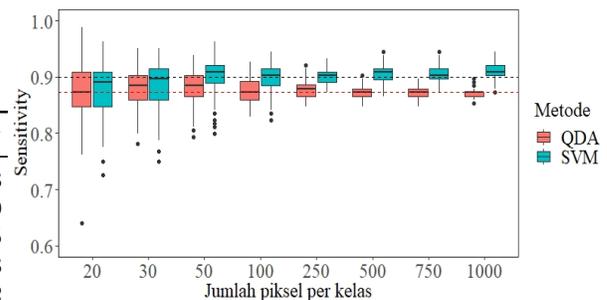
Nomor skenario	Banyaknya piksel per kelas	$C$	$\gamma$	<i>Error</i>
1	20	$10^6$	$10^{-6}$	0.098
2	30	$10^4$	$10^{-3}$	0.079
3	40	$10^6$	$10^{-6}$	0.085
4	100	$10^6$	$10^{-6}$	0.078
5	250	$10^4$	$10^{-3}$	0.072
6	500	$10^0$	$10^0$	0.069
7	750	$10^0$	$10^0$	0.068
8	1000	$10^0$	$10^0$	0.066

Hasil kinerja klasifikasi dari metode QDA dan SVM pada 800 set data ditunjukkan pada Gambar 3 sampai dengan Gambar 6.



Gambar 3. Diagram kotak garis untuk *overall accuracy*

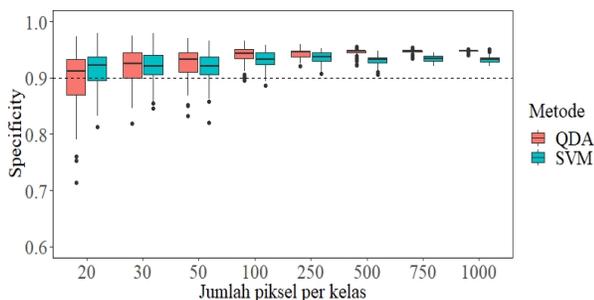
*Overall accuracy* (OA) adalah parameter ketepatan hasil klasifikasi untuk seluruh piksel. Gambar 3 menunjukkan semakin banyak data *training* yang digunakan, nilai median OA semakin meningkat dan keragaman OA semakin kecil. Secara umum, kedua metode menghasilkan nilai median OA di atas 0.9 untuk setiap skenario. Ketika menggunakan 20 piksel per kelas, nilai median OA metode QDA lebih kecil dari metode SVM. Namun, nilai median OA metode QDA cenderung lebih besar dari metode SVM ketika menggunakan data *training* sebanyak 30 piksel per kelas atau lebih walau selisihnya kecil. Ketika menggunakan 250 piksel per kelas, nilai median OA metode QDA dan SVM adalah sama. Secara lebih rinci, kedua metode menghasilkan nilai OA di atas 0.9 untuk setiap ulangan ketika menggunakan data *training* sebanyak 250 piksel per kelas atau lebih.



Gambar 4. Diagram kotak garis untuk *sensitivity*

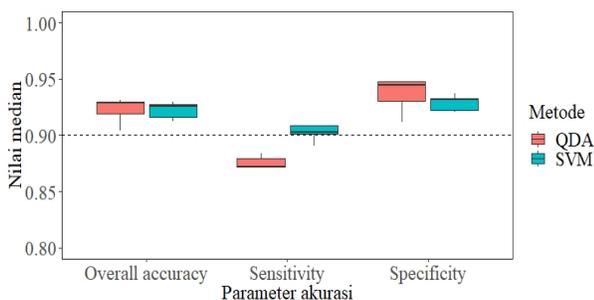
*Sensitivity* adalah parameter ketepatan hasil klasifikasi untuk piksel kelas vegetasi. Gambar 4 menunjukkan semakin banyak data *training* yang digunakan, keragaman nilai *sensitivity* dari kedua metode semakin kecil. Kedua metode menunjukkan kinerja yang berbeda pada nilai median *sensitivity*. Semakin banyak data *training* yang digunakan, nilai median *sensitivity* QDA semakin konvergen menuju 0.872 yang ditandai dengan garis berwarna merah pada Gambar 5. Nilai median *sensitivity* metode SVM cenderung meningkat seiring semakin banyak data *training* yang digunakan. Secara lebih rinci, nilai median *sensitivity* metode SVM mencapai 0.9 ketika banyaknya data *training* yang digunakan adalah 50 piksel per kelas atau

lebih.



Gambar 5. Diagram kotak garis untuk *specificity*

*Specificity* adalah parameter ketepatan hasil klasifikasi untuk piksel kelas nonvegetasi. Gambar 5 menunjukkan semakin banyak data *training* yang digunakan, nilai median *specificity* semakin meningkat dan keragaman nilai *specificity* semakin kecil. Secara umum, kedua metode menghasilkan nilai median *specificity* di atas 0.9 untuk setiap skenario. Ketika menggunakan 20 piksel per kelas, nilai median *specificity* metode QDA lebih kecil dari metode SVM. Namun, nilai median *specificity* metode QDA lebih besar dari metode SVM ketika menggunakan data *training* sebanyak 30 piksel per kelas atau lebih. Secara lebih rinci, kedua metode menghasilkan nilai *specificity* di atas 0.9 untuk setiap ulangan ketika menggunakan data *training* sebanyak 250 piksel per kelas atau lebih.



Gambar 6. Diagram kotak garis untuk perbandingan metode QDA dan SVM berdasarkan nilai median tiap skenario

Gambar 6 menunjukkan perbandingan kinerja metode QDA dan SVM pada nilai median *overall accuracy*, *sensitivity* dan *specificity*. Berdasarkan *overall accuracy*, metode QDA dan svm menunjukkan kinerja yang relatif sama. Kedua metode

terlihat menunjukkan kinerja yang berbeda ketika ditinjau berdasarkan *sensitivity* dan *specificity*. Metode SVM menunjukkan kinerja *sensitivity* lebih baik dari metode QDA karena nilai median *sensitivity* dari metode SVM dapat mencapai 0.9, sedangkan dari metode QDA tidak mencapai 0.9. Metode QDA menunjukkan kinerja *specificity* lebih baik dari metode SVM karena nilai median *specificity* dari metode QDA cenderung lebih besar dari metode SVM. Namun, diagram kotak garis *specificity* SVM terlihat beririsan dengan diagram kotak garis *specificity* QDA. Oleh karena itu, uji t dilakukan untuk menguji perbedaan kinerja *specificity* QDA dan SVM. Hipotesis dari uji t adalah  $H_0$ : kinerja *specificity* QDA relatif sama dengan SVM dan  $H_1$ : kinerja *specificity* QDA lebih baik dari SVM. Berdasarkan hasil uji t, didapatkan nilai  $t = 1.612$  dan nilai  $p\text{-value} = 0.068$  dengan nilai derajat kebebasan  $df = 10.147$ . Nilai  $p\text{-value}$  lebih besar dari  $\alpha = 0.05$  sehingga hasil uji t menunjukkan bahwa kinerja *specificity* QDA relatif sama dengan SVM. Selain itu Gambar 6 memperlihatkan kedua metode memiliki nilai *sensitivity* lebih kecil dari nilai *specificity*. Hal tersebut disebabkan data *testing* memiliki proporsi yang tak seimbang pada dua kelas.

## V. SIMPULAN

Hasil penelitian ini menunjukkan semakin banyak data *training* yang digunakan, keragaman dari *overall accuracy*, *sensitivity* dan *specificity* pada kedua metode semakin kecil. Berdasarkan nilai *overall accuracy* dan *specificity*, metode QDA dan SVM memiliki kinerja yang relatif sama. Namun, metode SVM lebih baik daripada QDA berdasarkan nilai *sensitivity*.

## DAFTAR PUSTAKA

- Badan Pusat Statistik Provinsi DKI Jakarta (2017). *JAKARTA DALAM ANGKA 2017*. Jakarta(ID): Badan Pusat Statistik Provinsi DKI Jakarta.
- Burges, C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery* 2(2), 121–167.

- Chang, C. and C. Lin (2011). Libsvm : A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2(3), 27.
- Cho, G., N. Gantulga, and Y. Choi (2017). A comparative study on multi-class svm and kernel function for land cover classification in a kompsat-2 image. *KSCE Journal of Civil Engineering* 21(5), 1894–1904.
- James, G., D. Witten, T. Hastie, and R. Tibshirani (2013). *An Introduction to Statistical Learning with Applications in R*. New York(USA): Springer.
- Katadata (2017). Ruang terbuka hijau di jakarta kurang dari 10 persen. [Internet]. [diunduh 2018 April 1]. Tersedia pada: <https://databoks.katadata.co.id/datapublish/2017/01/24/jumlah-ruang-terbuka-hijau-di-jakarta-mencapai-3100>.
- Kavzoglu, T. and I. Colkesen (2009). Kernel function analysis for support vector machines for land cover classification. *International Journal of Applied Earth Observation and Geoinformation* 11(5), 352–359.
- L, C. (2016). Semi-automatic classification plugin documentation. [Internet]. [diunduh 2018 Oktober 18]. Tersedia pada: <https://doi.org/10.1016/j.jag.2009.06.002>.
- Muhammad, F. (2014). *Kajian Beberapa Metode Klasifikasi Citra Digital terhadap Data Penginderaan Jauh [skripsi]*. Bogor(ID): Institut Pertanian Bogor.
- Noi, P. and M. Kappas (2018). Comparison of random forest, k-nearest neighbor, and support vector machine classifiers for land cover classification using sentinel-2 imagery. *Sensors* 18(1), 18–19.
- Putri, D. (2017). *Pemodelan Klasifikasi Tutupan Lahan Gambut Kabupaten Rokan Hilir Provinsi Riau Menggunakan Convolutional Neural Network [skripsi]*. Bogor(ID): Institut Pertanian Bogor.
- Richards, J. (2012). *Remote Sensing and Digital Image Analysis An Introduction*. New York(USA): Springer.