

## PENERAPAN ALGORITMA *GLOWWORM SWARM OPTIMIZATION* PADA MODEL *GEOGRAPHICALLY WEIGHTED REGRESSION* DENGAN KERNEL ADAPTIF

I Gede Hardi Karmana<sup>1§</sup>, Luh Putu Ida Harini<sup>2</sup>, Ketut Jayanegara<sup>3</sup>,  
I Putu Eka Nila Kencana<sup>4</sup>

<sup>1</sup>Program Studi Matematika – Universitas Udayana [Email: dkar.math@gmail.com]

<sup>2</sup>Program Studi Matematika – Universitas Udayana [Email: ballidah@unud.ac.id]

<sup>3</sup>Program Studi Matematika – Universitas Udayana [Email: ktjayanegara@unud.ac.id]

<sup>4</sup>Program Studi Matematika – Universitas Udayana [Email: i.putu.enk@unud.ac.id]

<sup>§</sup>Corresponding Author

### ABSTRACT

*This study aimed to apply glowworm swarm optimization (GSO) algorithm as an alternate way to obtain optimal bandwidth in geographically weighted regression (GWR) model with adaptive kernel function. The result showed that GSO was able to obtain optimal bandwidth with lower cross validation (CV) value than the traditional way that was using k-nearest neighbor (KNN) algorithm. Unfortunately, the running time of GSO was far slower than KNN was.*

**Keywords:** *geographically weighted regression (GWR), bandwidth, algorithm, k-nearest neighbor (KNN), glowworm swarm optimization (GSO)*

### 1. PENDAHULUAN

Transportasi adalah suatu cara untuk berpindah dari satu lokasi ke lokasi lainnya. Seiring perubahan zaman, teknologi transportasi juga semakin berkembang. Perkembangan ini memengaruhi hubungan manusia pada tiap daerah. Hal ini menyebabkan perubahan pada suatu daerah akan semakin mudah memengaruhi daerah lainnya. Kondisi tersebut kemudian dicoba untuk dijelaskan melalui model yang disebut *geographically weighted regression (GWR)*.

GWR merupakan suatu model regresi dengan pembobot berdasarkan jarak geografis masing-masing wilayah. Pembobot diperoleh dengan mentransformasi jarak antar wilayah ke dalam suatu fungsi yang disebut fungsi kernel. Fungsi kernel pada GWR dibagi menjadi dua yaitu fungsi kernel tetap dan kernel adaptif (Caraka & Yasin, 2017).

Secara umum, perbedaan kedua fungsi kernel tersebut adalah fungsi kernel tetap menggunakan *bandwidth* yang sama untuk semua wilayah, sedangkan fungsi kernel adaptif bisa menggunakan *bandwidth* yang berbeda untuk tiap wilayah. Penentuan *bandwidth* yang optimal pada fungsi kernel adaptif cenderung lebih memakan waktu

daripada fungsi kernel tetap sehingga banyak aplikasi statistika memilih untuk menyederhanakan solusi dengan menggunakan algoritma *k-nearest neighbor (KNN)*. Namun, penggunaan algoritma KNN justru menghilangkan sifat adaptif karena *bandwidth* disesuaikan berdasarkan jumlah tetangga yang sama untuk seluruh daerah. Ini berarti diperlukan suatu algoritma yang mampu mempertahankan sifat adaptif tersebut. Salah satu algoritma tersebut adalah *glowworm swarm optimization (GSO)*.

*Glowworm swarm optimization (GSO)* merupakan suatu algoritma optimasi yang berbasis *swarm intelligence*. Ini berarti GSO mencari solusi optimal dengan membangkitkan suatu gerombolan dan memanfaatkan informasi yang dimiliki masing-masing individu pada gerombolan tersebut (Krishnanand & Goshe, 2006).

Penelitian sebelumnya mengenai GWR dilakukan oleh Utari dkk (2019) menunjukkan bahwa GWR dengan fungsi kernel adaptif lebih bagus daripada GWR dengan fungsi kernel tetap maupun regresi linier berganda pada data dengan heterogenitas spasial. Penelitian sebelumnya mengenai GSO dilakukan oleh Krishnanand dan Ghose (2009)

menunjukkan bahwa GSO dapat dipakai untuk mencari titik-titik ekstrim lokal maupun global pada berbagai fungsi.

Berdasarkan penjabaran yang telah diberikan, penulis melakukan penelitian untuk mengaplikasikan GSO pada GWR dengan fungsi kernel adaptif.

## 2. METODE PENELITIAN

### 2.1 Fungsi Kernel dan *Bandwidth*

Fungsi kernel merupakan suatu fungsi yang mentransformasikan nilai ke dalam rentang 0 hingga 1. Fungsi kernel pada GWR mentransformasikan nilai jarak antar daerah (amatan) menjadi nilai bobot. Jarak antar daerah dihitung dengan (Caraka & Yasin, 2017):

$$d_{ij} = \sqrt{(u_i - u_j)^2 + (v_i - v_j)^2} \quad (1)$$

dengan:

$d_{ij}$  : jarak Euclidean antara daerah ke- $i$  dengan daerah ke- $j$ ,

$(u_i, v_i)$  : letak geografis daerah ke- $i$ ,

$(u_j, v_j)$  : letak geografis daerah ke- $j$ .

Fungsi kernel yang dibagi menjadi fungsi kernel tetap dan fungsi kernel adaptif tergantung pada nilai *bandwidth* yang digunakan. Kernel tetap menggunakan *bandwidth* yang sama untuk seluruh daerah, sedangkan kernel adaptif dapat menggunakan nilai *bandwidth* yang bervariasi untuk tiap daerah. Selain itu, kernel dibagi pula menjadi beberapa bentuk fungsi. Salah satunya yang sering digunakan adalah fungsi kernel Gaussian. Nilai dari fungsi kernel adaptif Gaussian ( $w_j(u_i, v_i)$ ) dihitung dengan persamaan:

$$w_j(u_i, v_i) = e^{-\left(\frac{d_{ij}}{h_i}\right)^2} \quad (2)$$

dengan  $h_i$  adalah *bandwidth* kernel Gaussian untuk daerah ke- $i$ .

### 2.2 Geographically Weighted Regression

*Geographically Weighted Regression* (GWR) adalah suatu metode statistika yang memodelkan suatu pengaruh variabel bebas pada variabel dengan melihat heterogenitas spasial antar daerah. Heterogenitas spasial adalah suatu kondisi saat pengaruh variabel bebas terhadap variabel terikat pada suatu daerah ternyata bervariasi dengan daerah lainnya. Model GWR ditulis sebagai berikut (Caraka & Yasin, 2017).

$$y_i = \beta_0(u_i, v_i) + \sum_{k=1}^p \beta_k(u_i, v_i)x_{i,k} + \varepsilon_i \quad (3)$$

dengan:

$y_i$  : nilai variabel terikat amatan ke- $i$ ,

$x_{i,k}$  : nilai variabel bebas ke- $k$  amatan ke- $i$ ,

$(u_i, v_i)$  : letak geografis daerah (amatan) ke- $i$ ,

$\beta_0(u_i, v_i)$  : nilai *intercept* amatan ke- $i$ ,

$\beta_k(u_i, v_i)$  : nilai *slope* amatan ke- $i$ ,

$\varepsilon_i$  : nilai galat pada amatan ke- $i$  yang diasumsikan berdistribusi  $N(0, \sigma^2)$ .

Estimasi parameter dari model GWR dapat dihitung sebagai berikut:

$$\hat{\beta}(u_i, v_i) = (X^T W(u_i, v_i) X)^{-1} X^T W(u_i, v_i) Y \quad (4)$$

dengan:

$$\hat{\beta}(u_i, v_i) = \begin{bmatrix} \beta_0(u_i, v_i) \\ \beta_1(u_i, v_i) \\ \vdots \\ \beta_p(u_i, v_i) \end{bmatrix}, X = \begin{bmatrix} 1 & x_{1,1} & \dots & x_{1,p} \\ 1 & x_{2,1} & \dots & x_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & \dots & x_{n,p} \end{bmatrix}$$

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, W(u_i, v_i) = \begin{bmatrix} w_1(u_i, v_i) & 0 & \dots & 0 \\ 0 & w_2(u_i, v_i) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & w_n(u_i, v_i) \end{bmatrix}$$

Matriks  $W(u_i, v_i)$  merupakan matriks pembobot amatan ke- $i$  dan  $w_j(u_i, v_i)$  merupakan nilai bobot amatan ke- $i$  terhadap amatan ke- $j$ . Matriks pembobot diperoleh melalui transformasi fungsi kernel.

Pada penentuan matriks pembobot GWR, nilai *bandwidth* yang digunakan merupakan nilai *bandwidth* yang menghasilkan pembobot yang paling optimal. Salah satu kriteria yang digunakan untuk menentukan apakah suatu *bandwidth* menghasilkan pembobot yang optimal adalah dengan menggunakan nilai *Cross Validation* (CV). Nilai CV dirumuskan sebagai:

$$CV = \sum_{i=1}^n (y_i - \hat{y}_{\neq i}(h_i))^2 \quad (5)$$

dengan  $\hat{y}_{\neq i}(h_i)$  merupakan nilai estimasi variabel terikat untuk amatan ke- $i$  tanpa menyertakan variabel bebas ke- $i$  pada estimasi parameter modelnya. *Bandwidth* yang optimal adalah *bandwidth* yang menghasilkan nilai CV minimal (Caraka & Yasin, 2017).

### 2.3 Koefisien Determinasi ( $R^2$ )

Koefisien determinasi merupakan suatu ukuran yang menyatakan seberapa besar variabel bebas dapat menjelaskan variabel terikat. Nilai koefisien determinasi yang semakin besar menandakan variabel-variabel bebas pada model semakin mampu menjelaskan variabel terikat. Koefisien

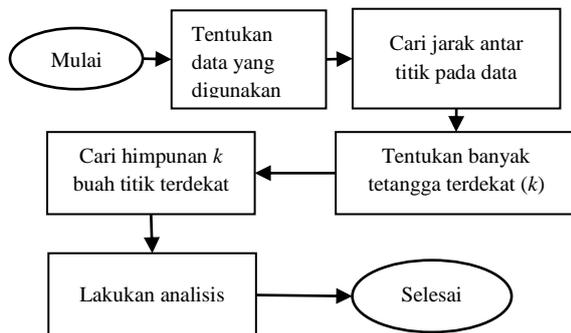
determinasi dirumuskan sebagai (Caraka & Yasin, 2017):

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (6)$$

dengan  $\hat{y}_i$  menyatakan nilai dugaan variabel terikat untuk amatan ke- $i$  dan  $\bar{y}$  merupakan nilai rata-rata variabel terikat.

## 2.4 Algoritma k-Nearest Neighbor

Algoritma *k-nearest neighbor* (KNN) adalah suatu algoritma yang termasuk metode *machine learning*. Ini berarti algoritma KNN dapat digunakan untuk mencari solusi optimal dari suatu himpunan data. Algoritma KNN digunakan untuk mengklasifikasikan data berdasarkan jarak terdekat ke- $k$  dari suatu titik pada data (Sutton, 2012). Algoritma KNN dijabarkan pada Gambar 1.



Gambar 1. Diagram Alir Algoritma KNN  
 (Sumber: Sutton, 2012)

## 2.5 Algoritma Glowworm Swarm Optimization (GSO)

*Glowworm Swarm Optimization* (GSO) diperkenalkan oleh K.N. Krishnanand dan D. Ghose pada tahun 2005. Algoritma ini terbagi menjadi tiga bagian yaitu tahap pembaharuan luciferin, pergerakan, dan pembaharuan jangkauan ketetanggaan (Krishnanand & Goshe, 2006).

Tahap pembaharuan luciferin dirumuskan sebagai:

$$l_i(t) = (1 - \rho)l_i(t-1) + \gamma.F(x_i(t-1)) \quad (7)$$

dengan

- $\rho$  : konstanta peluruhan luciferin,
- $\gamma$  : konstanta peningkatan luciferin,
- $l_i(t)$  : jumlah luciferin iterasi ke- $t$ ,
- $l_i(t-1)$  : jumlah luciferin iterasi ke- $(t-1)$ ,

$x_i(t-1)$  : posisi *glowworm* ke- $i$  saat iterasi ke- $(t-1)$ ,

$F(x_i(t-1))$  : nilai *fitness glowworm* ke- $i$  saat iterasi ke- $(t-1)$ .

Tahap pergerakan dirumuskan sebagai:

$$x_i(t) = x_i(t-1) + s \left( \frac{x_j(t-1) - x_i(t-1)}{\|x_j(t-1) - x_i(t-1)\|} \right) \quad (8)$$

Dengan  $x_i(t)$  adalah posisi *glowworm* ke- $i$  saat iterasi ke- $t$ ,  $x_j(t-1)$  adalah *glowworm* yang menjadi tujuan,  $s$  adalah ukuran langkah, dan  $\| \quad \|$  adalah jarak Euclidean. Pergerakan dilakukan setelah *glowworm* yang menjadi tujuan dipilih berdasarkan peluang:

$$prob_{ij}(t) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} (l_k(t) - l_i(t))} \quad (9)$$

dengan

$prob_{ij}(t)$ : peluang *glowworm* ke- $i$  memilih *glowworm* ke- $j$  sebagai tujuan,

$N_i(t)$  : Himpunan tetangga dari *glowworm* ke- $i$ .

$N_i(t)$  didefinisikan sebagai:

$$N_i(t) = \{j | d_{ij}(t) < r_g^i(t), l_i(t) < l_j(t)\}$$

dengan

$d_{ij}(t)$  : jarak Euclidean antara *glowworm* ke- $i$  dan ke- $j$ ,

$r_g^i(t)$  : jangkauan ketetanggaan *glowworm* ke- $i$ .

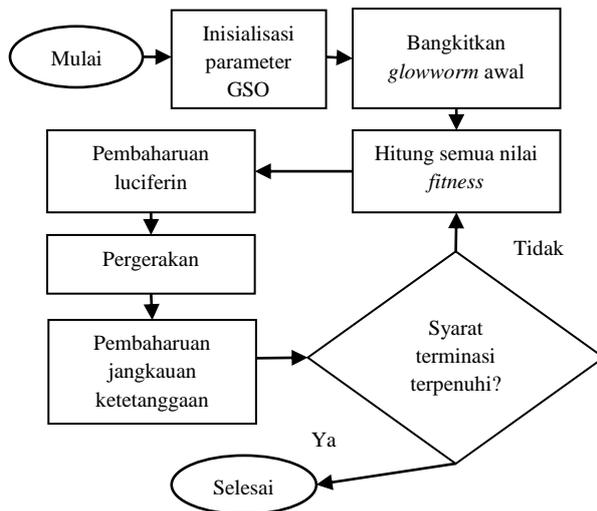
Pembaharuan jangkauan ketetanggaan dirumuskan sebagai:

$$r_g^i(t) = \min\{r_s, \max(0, r_g^i(t-1) + \theta(n_t - |N_i(t)|))\} \quad (10)$$

dengan

- $r_s$  : jangkauan maksimal tetangga,
- $\theta$  : konstanta perubahan jangkauan,
- $n_t$  : banyak maksimal tetangga.

Langkah-langkah algoritma GSO ditunjukkan pada Gambar 2.



Gambar 2. Diagram Alir GSO  
(Sumber: Oramus, 2010)

### 3. HASIL DAN PEMBAHASAN

Penelitian ini menggunakan data sekunder kecelakaan lalu lintas tiap kecamatan di Bali tahun 2017 yang diambil dari penelitian Utari dkk. (2019). Data terdiri dari satu variabel terikat ( $Y$ ) dan tiga variabel bebas ( $X$ ) yaitu:

- $Y$  = jumlah kecelakaan lalu lintas,
- $X_1$  = kepadatan penduduk (jiwa/km<sup>2</sup>),
- $X_2$  = rata-rata curah hujan (mm),
- $X_3$  = penduduk usia 15-29 tahun (jiwa).

Penelitian tersebut menyatakan bahwa terdapat heterogenitas spasial pada data yang digunakan. Selain itu, diperoleh kesimpulan bahwa model GWR dengan kernel adaptif Gaussian adalah model terbaik untuk kasus pada penelitian tersebut. Dengan mengacu pada hasil penelitian Utari dkk, tahapan analisis pada penelitian ini dijabarkan sebagai berikut:

1. Menghitung statistik deskriptif data dan jarak antar daerah.
2. Menentukan *bandwidth* optimal berdasarkan nilai *CV* menggunakan algoritma KNN dan GSO.
3. Menentukan estimasi model menggunakan algoritma KNN dan GSO.
4. Membandingkan kinerja algoritma KNN dan GSO berdasarkan koefisien determinasi dan *running time*.

Khusus untuk algoritma GSO, parameter algoritma diperoleh dari penelitian Krishnanand dan Ghose (2005) dengan modifikasi pada beberapa parameter. Parameter GSO meliputi  $\rho = 0,4$ ;  $\gamma = 0,6$ ;  $\theta = 0,08$ ;  $n_t = 5$ ;  $s = 0,2$ ; dan  $r_s = 150$ .

Nilai  $s$  dan  $r_s$  dimodifikasi untuk menyesuaikan dengan rentang jarak antar daerah. Selain itu, individu yang dibangkitkan ada sebanyak 5 dengan iterasi sebanyak 500 dan percobaan dilakukan sebanyak 10 kali. Aplikasi yang digunakan adalah Matlab R2015a.

#### 3.1 Perhitungan Statistik Deskriptif Data

Statistik deskriptif data mengacu pada penelitian Utari dkk. (2019). Hasil statistik deskriptif ditampilkan pada Tabel 1.

Tabel 1. Statistik Deskriptif Data

Variabel	Rataan	Simpangan Baku	Minimum	Maksimum
$Y$	33,91	31,784	6	147
$X_1$	1.253,39	1.382,407	162	6009
$X_2$	223,32	70,314	93,250	393,125
$X_3$	17.223,33	17.000,176	2.960	92.540

Sumber: (Utari dkk., 2019)

Berdasarkan Tabel 1 dapat dilihat bahwa rata-rata kecelakaan ( $Y$ ), kepadatan penduduk ( $X_1$ ), dan jumlah penduduk usia 15-29 tahun ( $X_3$ ) lebih mendekati nilai minimum dengan simpangan yang hampir sama besar dengan rata-ratanya. Ini berarti terdapat beberapa daerah dengan nilai yang jauh melampaui daerah lainnya. Diperoleh bahwa daerah tersebut merupakan daerah sekitar Kota Denpasar.

Lain halnya dengan rata-rata curah hujan ( $X_2$ ) yang nilainya tidak terlampaui jauh antar daerah. Rata-rata curah hujan berada antara rentang 90 mm hingga 400 mm dengan curah hujan terendah terjadi di daerah Seririt dan tertinggi di daerah Bebandem.

Selanjutnya untuk mencari jarak antar daerah digunakan persamaan (1) dengan mentransformasikan letak astronomis masing-masing daerah ke dalam satuan jarak (kilometer). Setelah jarak antar daerah diperoleh maka dilanjutkan dengan penentuan *bandwidth* optimal.

#### 3.2 Penentuan *Bandwidth* Optimal pada Algoritma KNN dan GSO

Perhitungan *bandwidth* optimal pada algoritma KNN dan GSO mengalami proses yang berbeda. Hal ini dikarenakan algoritma KNN yang merupakan algoritma deterministik yang selalu menghasilkan nilai tetap, sedangkan algoritma GSO merupakan algoritma probabilistik yang menghasilkan

nilai tidak tetap. Hasil untuk beberapa daerah dapat dilihat pada Tabel 2.

Tabel 2. *Bandwidth* dan Nilai CV

Daerah	<i>Bandwidth</i>	
	KNN	GSO
Tabanan (D1)	13,24	113,9 ± 0,02
Klungkung (D2)	14,05	20,75 ± 0
Jembrana (D3)	40,11	7,46 ± 0,4
Nilai CV	194,06	136,8 ± 0,91

Sumber: Data diolah (2019).

Tabel 2 memperlihatkan bahwa beberapa daerah memiliki *bandwidth* optimal yang berbeda. Selain itu, nilai CV pada GSO lebih kecil daripada nilai CV pada KNN. Ini berarti algoritma GSO mampu mencari *bandwidth* yang lebih optimal daripada algoritma KNN. Setelah mendapatkan *bandwidth* optimal kemudian dilanjutkan dengan mencari matriks pembobot dan estimasi parameter model.

### 3.3 Penentuan Estimasi Model pada Algoritma KNN dan GSO

Estimasi model diperoleh dengan mencari estimasi parameter berdasarkan persamaan (4). Hasil untuk beberapa amatan dapat dilihat pada Tabel 3.

Tabel 3. Estimasi Parameter Model

Daerah	KNN			
	$\beta_0$	$\beta_1$	$\beta_2$	$\beta_3$
D1	52,05	0,0028	-0,179	0,0014
D2	16,29	0,01	-0,059	0,0010
D3	25,74	0,0052	-0,088	0,0013
Daerah	GSO			
	$\beta_0$	$\beta_1$	$\beta_2$	$\beta_3$
D1	24,39	0,0073	-0,082	0,0010
D2	20,40	0,0085	-0,069	0,0010
D3	-3,20	-0,0658	0,017	0,0039

Sumber: Data diolah (2019).

Estimasi parameter model pada Tabel 3 menggunakan algoritma KNN dan GSO berbeda. Perbedaan ini disebabkan oleh pemilihan *bandwidth* optimal yang berlainan. Hal ini memperjelas bahwa pemilihan *bandwidth* memengaruhi estimasi model yang diperoleh. Contohnya adalah estimasi model daerah Jembrana (D3) yaitu untuk algoritma KNN:

$y_{D3} = 25,74 + 0,0052x_1 - 0,088x_2 + 0,0013x_3$   
 sedangkan untuk algoritma GSO:

$$y_{D3} = -3,20 - 0,0658x_1 + 0,017x_2 + 0,0039x_3$$

Terlihat bahwa koefisien beberapa variabel pada algoritma KNN menjadi terbalik (berlawanan tanda) pada algoritma GSO. Sebagai perbandingan, estimasi model daerah Jembrana menggunakan algoritma KNN menunjukkan bahwa setiap kenaikan satu satuan kepadatan penduduk ( $x_1$ ) akan mengakibatkan kenaikan angka kecelakaan ( $y$ ) sebesar 0,0052 satuan. Namun, kenaikan satu satuan kepadatan penduduk ( $x_1$ ) pada estimasi model menggunakan algoritma GSO justru akan menurunkan angka kecelakaan ( $y$ ) sebesar 0,0658 satuan. Hal yang serupa terjadi pada rata-rata curah hujan ( $x_2$ ). Selain itu, kenaikan satu satuan penduduk usia 15 – 29 tahun ( $x_3$ ) pada estimasi model dengan algoritma KNN akan meningkatkan angka kecelakaan ( $y$ ) sebesar 0,0013 satuan, sedangkan pada estimasi model dengan algoritma GSO justru menjadi tiga kali lipat yaitu meningkatkan angka kecelakaan ( $y$ ) sebesar 0,0039 satuan. Estimasi model lainnya juga diinterpretasikan dengan cara serupa.

### 3.4 Perbandingan Kinerja Algoritma KNN dan GSO

Setelah estimasi model diperoleh, nilai koefisien determinasi dapat dihitung menggunakan persamaan (6). Selain memperhatikan nilai koefisien determinasi, masing-masing algoritma juga menampilkan durasi (*running time*) yang diperlukan untuk membentuk estimasi model. Hasil perbandingan kinerja algoritma dapat dilihat pada Tabel 4.

Tabel 4. Perbandingan Kinerja Algoritma

	KNN	GSO
$R^2$ (%)	90,64	95,21 ± 1,2
Durasi (s)	1,79 ± 0,23	180,73 ± 1,43

Sumber: Data diolah (2019).

Berdasarkan Tabel 4 diperoleh bahwa nilai rata-rata koefisien determinasi yang diperoleh algoritma GSO lebih besar daripada algoritma KNN. Di lain sisi, durasi algoritma GSO justru jauh lebih lambat daripada algoritma KNN. Ini berarti pemilihan *bandwidth* yang lebih optimal pada algoritma GSO akan diikuti oleh durasi algoritma yang lebih lambat.

## 4. KESIMPULAN DAN SARAN

### 4.1 Kesimpulan

Berdasarkan pembahasan yang telah dijabarkan maka diperoleh bahwa algoritma GSO mampu menghasilkan *bandwidth* yang lebih optimal daripada algoritma KNN, tetapi durasi yang diperlukan untuk menghasilkan estimasi model akan jauh lebih lambat.

### 4.2 Saran

Berdasarkan kesimpulan yang diperoleh maka saran yang dapat diberikan yaitu:

1. Parameter, jumlah individu, maupun iterasi pada algoritma GSO dapat dimodifikasi agar durasi (*running time*) algoritma tersebut lebih cepat.
2. Penelitian berikutnya dapat menggunakan algoritma probabilistik lainnya atau deterministik.

## DAFTAR PUSTAKA

- Caraka, R.E. & Yasin, H. 2017. *Geographically Weighted Regression (GWR) Sebuah Pendekatan Regresi Geografis*. First ed. Yogyakarta: Mobius.
- Krishnanand, K.N. & Ghose, D. 2006. *Glowworm Swarm Based Optimization*

*Algorithm for Multimodal Function with Collective Robotic Applications. Multiagent and Grid System – An International Jurnal*. Volume 2. Hal 209 – 222.

- Krishnanand, K.N. & Ghose, D. 2009. *Glowworm Swarm Optimization for Simultaneous Capture of Multiple Local Optima of Multimodal Functions. Swarm Intell*. Volume 3. Hal 87 – 124.

- Oramus, Piotr. 2010. *Improvements to Glowworm Swarm Optimization Algorithm. Computer Science*. Volume 10. Hal 7 – 20.

- Sutton, Oliver. 2012. *Introduction to k Nearest Neighbour Classification and Condensed Nearest Neighbour Data Reduction*. Leicester: University of Leicester.

- Utari, N. K. E. Y. Srinadi, I G. A. M. & Susilawati M. 2019. *Model Geographically Weighted Regression (GWR) Faktor-Faktor yang Memengaruhi Kecelakaan Lalu Lintas di Provinsi Bali. E-Jurnal Matematika*. Volume 8(2). Hal. 140 – 147.