

KOMPARASI METODE PERAMALAN *AUTOMATIC CLUSTERING TECHNIQUE AND FUZZY LOGICAL RELATIONSHIPS DENGAN SINGLE EXPONENTIAL SMOOTHING*

Betik Endaryati¹, Robert Kurniawan²

^{1,2} Jurusan Komputasi Statistik Sekolah Tinggi Ilmu Statistik (STIS) Jakarta
Email: ¹betyendaryati@yahoo.co.id, ²robertk@stis.ac.id

Abstract

Automatic clustering technique and fuzzy logical relationships (ACFLR) is one of the forecasting method that used to predict time series data that can be applied in any data. Several previous studies said that this method has a good accuracy. Therefore, this study aims to compare the ACFLR methods with single exponential smoothing method and apply it to simulation data with uniform distribution. The performance of the method is measured based on MSE and MAPE. The results of the comparison of the methods showed that ACFLR has a higher forecasting accuracy than single exponential smoothing. This is evidenced by the value of MSE and MAPE of ACFLR is lower than single exponential smoothing.

Keywords: Fuzzy, Forecasting, Automatic Clustering-Fuzzy Logic Relationships, Single Exponential Smoothing

1. Pendahuluan

Selama ini telah banyak berkembang metode peramalan dalam statistika, salah satunya adalah metode *exponential smoothing*. Menurut Makridakis, *exponential smoothing* merupakan prosedur perbaikan yang dilakukan terus-menerus pada peramalan terhadap objek pengamatan terbaru^[4]. Menurut Raharja dkk, metode tersebut merupakan metode peramalan yang sering digunakan, karena memiliki kinerja yang baik, dimana metode ini memiliki nilai parameter dan pengaruh yang besar terhadap peramalan^[5]. Akan tetapi, metode tersebut belum mampu melakukan peramalan apabila data masa lalu yang tersedia berbentuk linguistik. Untuk mengatasi hal tersebut, banyak peneliti yang mengusulkan suatu metode dengan menggunakan konsep *soft computing*.

Menurut Zadeh, *soft computing* merupakan segolongan metode yang mampu mengolah data dengan baik walaupun di dalamnya terdapat ketidakpastian, ketidakakuratan, maupun kebenaran parsial^[7]. Pada dekade terakhir, konsep *soft computing* banyak digunakan untuk memecahkan masalah dalam berbagai bidang termasuk masalah peramalan. Hal ini seiring dengan banyaknya bidang yang membutuhkan peramalan yang akurat. Untuk memperoleh peramalan yang akurat, perlu dilakukan terlebih dahulu pemodelan data untuk melihat karakteristik dari suatu data. Biasanya pemodelan dan peramalan data dilakukan pada data runtun waktu (*time series*).

Salah satu metode pemodelan *time series* yang menggunakan konsep *soft computing* adalah *fuzzy time series*. Metode yang pertama kali diperkenalkan oleh Song, dkk^[6] tersebut mampu menangani data yang tidak lengkap dan berbentuk linguistik. Akan tetapi, Lee, dkk (2007) menyatakan bahwa hasil peramalan menggunakan metode tersebut masih memiliki nilai MSE yang relatif besar sehingga metode peramalan yang sudah ada terus dikembangkan untuk mendapatkan tingkat akurasi yang lebih tinggi. Oleh karena itu, pada tahun 2009 Chen, dkk memperkenalkan metode modifikasi dari *fuzzy time series* yang

dikenal dengan *Automatic Clustering Technique and Fuzzy Logical Relationship* (ACFLR).

ACFLR merupakan salah satu metode dengan konsep *fuzzy logic* yang digunakan untuk pemodelan data *time series*. Pada beberapa penelitian sebelumnya menunjukkan bahwa metode ini menghasilkan *Mean Square Error (MSE)* yang paling minimum diantara *Song and Chissom's method*, *Sullivan and Woodall's method*, *Chen's method*, dan *Huang's method*. Selain itu, Kurniawan (2011) dalam penelitiannya juga menyimpulkan bahwa metode ACFLR memiliki ketepatan peramalan yang lebih baik dibandingkan dengan ARIMA. Oleh karena itu, dalam penelitian ini peneliti ingin membandingkan metode ACFLR dengan metode *single exponential smoothing* berdasarkan keakuratan prediksinya. Dalam implementasinya, penelitian ini akan menggunakan data simulasi yang dibangkitkan berdasarkan distribusi uniform.

2. Tinjauan Pustaka

2.1 Single Exponential Smoothing

Metode *single exponential smoothing* digunakan untuk peramalan jangka pendek. Menurut Makridarkis (1999), model *single exponential smoothing* mengasumsikan bahwa data belfluktuasi di sekitar nilai *mean* yang tetap, tanpa trend atau pola pertumbuhan konsisten. Konstanta *smoothing* berkisar antara 0 dan 1. Nilai yang mendekati 1 memberikan penekanan terbesar pada nilai saat ini, sedangkan nilai yang dekat dengan 0 memberikan penekanan pada titik data sebelumnya. Bentuk umum yang digunakan untuk menghitung ramalan adalah:

$$F_{t+1} = \alpha X_t + (1 - \alpha)F_t \quad (1)$$

dimana F_{t+1} adalah nilai ramalan untuk periode berikutnya ($t+1$), α adalah konstanta pemulusan, X_t adalah nilai X yang sebenarnya pada periode t , dan F_t adalah peramalan untuk periode t .

2.2 Fuzzy Logic

Fuzzy secara bahasa dapat diartikan samar, dengan kata lain logika *fuzzy* adalah logika yang samar. Pada logika *fuzzy* suatu nilai dapat bernilai "true" dan "false" secara bersamaan. Tingkat "true" atau "false" nilai dalam logika *fuzzy* tergantung pada bobot keanggotaan yang dimilikinya. Logika *fuzzy* memiliki derajat keanggotaan dalam rentang 0 (nol) hingga 1 (satu), berbeda dengan logika digital yang hanya memiliki dua keanggotaan 0 atau 1 saja pada satu waktu.

Logika *fuzzy* sering digunakan untuk mengekspresikan suatu nilai yang diterjemahkan dalam bahasa (*linguistic*). Misalnya, besarnya laju kendaraan yang diekspresikan dengan sangat cepat, cepat, agak cepat, pelan, agak pelan, dan sangat pelan.

Menurut Kusumadewi, logika *fuzzy* dapat dianggap sebagai kotak hitam yang berhubungan antara ruang input menuju ruang output^[3]. Kotak hitam tersebut berisi cara atau metode yang dapat digunakan untuk mengolah data *input* menjadi *output* dalam bentuk informasi yang baik. Contoh konsep logika *fuzzy* adalah penumpang taksi berkata pada sopir taksi seberapa cepat laju kendaraan yang diinginkan, sopir taksi akan mengatur pijakan gas taksinya.

2.3 Automatic Clustering Technique

Algoritma ini digunakan untuk menentukan interval. Menurut Chen, *et al.*^[1], langkah-langkah dalam algoritma *automatic clustering* adalah sebagai berikut:

Langkah 1:

Mengurutkan data *numeric* secara *ascending*. Diasumsikan tidak ada data yang bernilai sama (*duplicate*) dan akan ditampilkan seperti d_1, d_2, \dots, d_n . Berdasarkan urutan data tersebut, kemudian dihitung beda rata-ratanya (*average_dif*). Beda rata-rata dihitung dari beda rata-rata antara dua data yang bersebelahan. Beda rata-rata (*average_dif*) dihitung dengan menggunakan rumus berikut:

$$average_dif = \frac{\sum_{i=0}^{n-1} (d_{i+1} - d_i)}{(n-1)} \quad (2)$$

Langkah 2:

Mengambil data urutan pertama dari data yang telah diurutkan ke dalam pengelompokkan yang sudah ada. Berdasarkan nilai *average_dif*, ditentukan apakah data pertama yang telah diurutkan tersebut dapat diletakkan dalam kluster yang ada atau diletakkan dalam kluster baru dengan prinsip sebagai berikut:

Prinsip 1:

Mengasumsikan bahwa kluster yang ada adalah kluster pertama dan di dalamnya hanya ada satu data d_1 dan d_2 adalah data yang berdekatan dengan d_1 , ditampilkan sebagai $\{d_1\}, d_2, \dots, d_n$.

Jika $d_2 - d_1 \leq average_dif$ maka d_2 diletakkan pada kluster yang ada dimana terdapat d_1 . Selanjutnya membuat kluster baru untuk d_2 dan kluster yang memuat d_2 tersebut menjadi kluster yang ada.

Prinsip 2:

Mengasumsikan kluster yang ada bukan kluster pertama dan hanya ada satu d_j di kluster yang ada dan d_k adalah data yang berdekatan dengan d_j . Menganggap bahwa d_j adalah data terbesar di kluster yang berada di atas kluster yang ada, ditampilkan sebagai $\{d_1, \dots\}, \dots, \{\dots, d_i\}, \{d_j\}, d_k, \dots, d_n$.

Jika $d_k - d_j \leq average_dif$ dan $d_k - d_j \leq d_j - d_1$ maka d_k diletakkan pada kluster dimana terdapat d_j . Selanjutnya membuat kluster baru untuk d_k dan biarkan kluster tersebut terdapat pada kluster yang ada.

Prinsip 3:

Mengasumsikan kluster yang ada bukan kluster pertama dan ada lebih dari satu data pada kluster yang ada. Menganggap bahwa d_i adalah data terbesar pada kluster yang ada dan d_j adalah kluster terdekat dengan d_i , ditampilkan sebagai $\{d_i, \dots\}, \dots, \{\dots\}, \{\dots, d_i\}, d_j, \dots, d_n$.

Jika $d_j - d_i \leq average_dif$ dan $d_j - d_i \leq cluster_dif$ maka d_j diletakkan pada kluster dimana terdapat d_i . Selanjutnya membuat kluster baru dimana d_j termasuk dalam kluster yang ada. Perbedaan rata-rata antara setiap pasangan data yang berdekatan dalam kluster dinyatakan dengan *cluster_dif*.

$$cluster_dif = \frac{\sum_{i=0}^{n-1} (c_{i+1} - c_i)}{(n-1)} \quad (3)$$

dengan c_1, c_2, \dots, c_n merupakan data dalam kluster yang ada.

Langkah 3:

Berdasarkan pengelompokkan yang dihasilkan dari langkah 2, selanjutnya menyesuaikan isi dari kluster tersebut berdasarkan prinsip-prinsip berikut ini:

1. Jika kluster memiliki lebih dari dua data, maka amankan data terkecil dan terbesar, kemudian hapus data yang lainnya.
2. Jika kluster tepat memiliki dua data, maka jangan diubah.

3. Jika kluster hanya memiliki satu data d_q , maka letakkan nilai ' $d_q - average_dif$ ' dan ' $d_q + average_dif$ ' ke dalam kluster dan hapus d_q dari kluster ini. Jika situasi berikut terjadi, maka:
 - a. Jika situasi terjadi pada kluster pertama, maka hapus nilai ' $d_q - average_dif$ '.
 - b. Jika situasi terjadi pada kluster terakhir, hapus nilai ' $d_q + average_dif$ '.
 - c. Jika nilai ' $d_q - average_dif$ ' lebih kecil dari nilai terkecil dalam kluster di atasnya, maka tindakan berdasarkan prinsip 3 dibatalkan.

Langkah 4 :

Mengasumsikan bahwa hasil kluster yang didapatkan dari langkah 3 ditampilkan sebagai berikut:

$$\{d_1, d_2\}, \{d_3, d_4\}, \{d_5, d_6\}, \dots, \{d_{n-1}, d_n\}$$

Selanjutnya, mentransformasikan kluster-kluster tersebut ke dalam interval yang bersebelahan dengan sub langkah berikut:

1. Mentransformasikan kluster pertama $\{d_1, d_2\}$ dalam interval $[d_1, d_{2aaa}]$.
2. Jika interval yang ada $[d_1, d_{2aaa}]$ dan kluster yang ada adalah $\{d_1, d_2\}$, maka:
 - a. Jika $d_j \geq d_k$, maka $\{d_1, d_2\}$ dalam kluster yang ada ditransformasi ke dalam interval $[d_1, d_{2aaa}]$ dan kluster selanjutnya $\{d_m, d_n\}$ menjadi kluster yang ada.
 - b. Jika $d_j < d_k$, maka transformasikan $\{d_k, d_l\}$ dalam interval $[d_k, d_{laaa}]$. Selanjutnya, $[d_k, d_{laaa}]$ menjadi interval dan kluster berikutnya $\{d_m, d_n\}$ menjadi klusternya. Jika nilai dalam interval adalah $[d_i, d_{jaaa}]$ dan nilai klusternya adalah $\{d_k\}$, maka transformasikan interval $[d_i, d_{jaaa}]$ ke dalam $[d_i, d_{kaaa}]$. Kemudian $[d_i, d_{kaaa}]$ menjadi interval dan kluster berikutnya menjadi kluster yang ada.
3. Melakukan pemeriksaan berulang terhadap interval yang ada hingga semua kluster ditransformasikan menjadi interval.

Langkah 5:

Untuk setiap interval yang diperoleh dari langkah 4, kemudian bagi setiap interval tersebut dalam p sub interval dimana $p \geq 1$.

2.4 Automatic Clustering Technique And Fuzzy Logical Relationships

Setelah mendapatkan interval dengan menggunakan *automatic clustering technique*, selanjutnya nilai peramalan dapat dihitung menggunakan algoritma sebagai berikut:

Langkah 1:

Menghitung titik tengah setiap interval dengan menggunakan algoritma *clustering* di atas.

Langkah 2:

Mengasumsikan terdapat interval u_1, u_2, \dots, u_n , kemudian mendefinisikan setiap *fuzzy set* A_i , dimana $1 \leq i \leq n$, sebagai berikut:

$$\begin{aligned}
 A_1 &= \frac{1}{u_1} + \frac{0,5}{u_2} + \frac{0}{u_3} + \dots + \frac{0}{u_6} + \frac{0}{u_7} \\
 A_2 &= \frac{0,5}{u_1} + \frac{1}{u_2} + \frac{0,5}{u_3} + \dots + \frac{0}{u_6} + \frac{0}{u_7} \\
 A_3 &= \frac{0}{u_1} + \frac{0,5}{u_2} + \frac{1}{u_3} + \dots + \frac{0}{u_6} + \frac{0}{u_7} \\
 &\vdots \\
 A_7 &= \frac{0}{u_1} + \frac{0}{u_2} + \frac{0}{u_3} + \dots + \frac{0,5}{u_6} + \frac{1}{u_7}
 \end{aligned}$$

Langkah 3:

Fuzzify setiap data dalam *fuzzy set*. Jika data termasuk dalam u_i dimana $1 \leq i \leq n$, maka data tersebut di-*fuzzify* dalam A_i .

Langkah 4:

Membuat hubungan logika *fuzzy* berdasarkan data yang didapatkan pada langkah 3. Jika data ter-*fuzzy* dari tahun t dan $t+1$ adalah A_j dan A_k , maka selanjutnya membuat hubungan logika *fuzzy*, bagi hubungan logika *fuzzy* dalam kelompok hubungan logika *fuzzy*, dimana ia mempunyai keadaan sekarang yang sama diletakkan dalam grup yang sama.

Langkah 5:

Menghitung angka ramalan dari data dengan prinsip sebagai berikut:

1. Jika data ter-*fuzzy* dari periode t adalah A_j dan hanya terdapat satu hubungan logika *fuzzy* dalam grupnya dimana keadaan sekarang adalah $A_j \rightarrow A_k$, maka data yang diramalkan dari periode $(t+1)$ adalah m_k , dimana m_k adalah titik tengah dari interval u_k , dan nilai keanggotaan maksimum dari *fuzzy set* A_k terjadi pada interval u_k .
2. Jika data ter-*fuzzy* dari periode t adalah A_j dan terdapat hubungan logika *fuzzy* pada grup relasi logika fungsi dimana keadaan sekarangnya adalah A_j , maka data aktual dari periode $t+1$ dihitung dengan rumus berikut:

$$\frac{x_1 \times m_{k1} + x_2 \times m_{k2} + \dots + x_p \times m_{kp}}{x_1 + x_2 + \dots + x_p} \quad (4)$$

dimana x_1 menyatakan jumlah hubungan logika *fuzzy* dalam grup hubungan logika *fuzzy*. m_{k1}, m_{k2}, \dots , dan m_{kp} adalah titik tengah dari interval u_{k1}, u_{k2}, \dots , dan u_{kp} , dan nilai keanggotaan maksimum dari *fuzzy set* $A_{k1}, A_{k2}, \dots, A_{kp}$ terjadi pada interval u_{k1}, u_{k2}, \dots , dan u_{kp} .

3. Jika data aktual ter-*fuzzy* dari periode adalah A_j dan terdapat sebuah hubungan logika *fuzzy* dalam grup hubungan logika *fuzzy* dimana keadaan sekarangnya adalah $A_j \rightarrow \#$, dimana simbol $\#$ menyatakan nilai yang tidak diketahui, maka data teramalkan dari periode $t+1$ adalah m_j , dimana m_j adalah titik tengah dari interval u_j dan nilai keanggotaan maksimum dari *fuzzy set* A_j terjadi pada u_j .

2.5 Mean Square Error (MSE) dan Mean Absolute Percentage Error (MAPE)

Mean Squared Error (MSE) adalah nilai yang diharapkan dari kuadrat *error*. *Error* menunjukkan seberapa besar perbedaan hasil estimasi dengan nilai aslinya. Berikut merupakan rumus penghitungan dari MSE:

$$MSE = \frac{1}{n} \sum_{t=1}^n (\hat{y}_t - y_t)^2 \quad (5)$$

dimana n adalah jumlah sampel, \hat{y}_t adalah nilai peramalan pada periode ke- t , dan y_t adalah nilai aktual pada periode ke- t .

Mean Absolute Percentage Error (MAPE) menghitung deviasi antara data aktual dengan nilai peramalan kemudian dihitung persen rata-ratanya. Menurut Zainun dan Majid, suatu model mempunyai kinerja sangat bagus jika nilai MAPE di bawah 10% dan mempunyai kinerja bagus jika nilai MAPE berada di antara 10% dan 20%^[8]. MAPE dapat dihitung dengan rumus sebagai berikut:

$$MAPE = \left(\frac{1}{n} \sum_{t=1}^n \left| \frac{(\hat{y}_t - y_t)}{y_t} \right| \times 100\% \right) \tag{6}$$

dimana n adalah jumlah sampel, \hat{y}_t adalah nilai peramalan pada periode ke-t, dan y_t adalah nilai aktual pada periode ke-t

3. Metodologi Penelitian

3.1 Sumber Data

Data yang digunakan dalam penelitian ini adalah data simulasi yang dibangkitkan berdasarkan distribusi uniform. Data simulasi menggunakan data berdistribusi uniform kontinyu dengan nilai minimal 1 dan maksimal 1000.

3.2 Metode Penelitian

Metode yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Membangkitkan data berdistribusi uniform kontinyu dengan nilai minimal 1 dan maksimal 1000 sejumlah 100, 200, 500, dan 1000.
2. Mengolah data dengan ACFLR
3. Mengolah data dengan *single exponential smoothing*
4. Membandingkan kedua metode berdasarkan MSE dan MAPE

4. Pembahasan

4.1 Penerapan ACFLR

Berdasarkan Tabel 1 dan Tabel 2 terlihat bahwa untuk data simulasi dengan jumlah observasi 100, nilai MSE dan MAPE terkecil berada pada sub interval $p = 5$ yaitu sebesar 1091,277 dan 0,02056223. Perulangan berhenti pada sub interval $p = 5$ sehingga nilai peramalan yang terbaik untuk data simulasi yang telah dibangkitkan sebanyak 100 ada pada sub interval tersebut. Nilai peramalan tersebut dikatakan sangat baik karena nilai MAPE berada diantara 1-10 %.

Sama halnya dengan data simulasi yang dibangkitkan sejumlah 200, nilai peramalan terbaik ada pada sub interval 5. Begitu juga dengan data yang dibangkitkan sejumlah 500, dimana MSE terkecil berada pada sub interval $p = 5$. Dengan demikian dapat dikatakan bahwa nilai peramalan terbaik pada data simulasi sejumlah 500 berada pada sub interval $p = 5$. Pada data simulasi yang dibangkitkan sejumlah 1000, nilai MSE dan MAPE terkecil berada pada sub interval $p = 9$. Hasil peramalan pada sub interval ini dapat dikatakan sangat baik karena memiliki nilai MAPE sangat kecil yaitu sebesar 0,026576714 atau 2%.

Tabel 1. MSE ACFLR pada Data Simulasi Menurut Jumlah Observasi dan Sub Interval

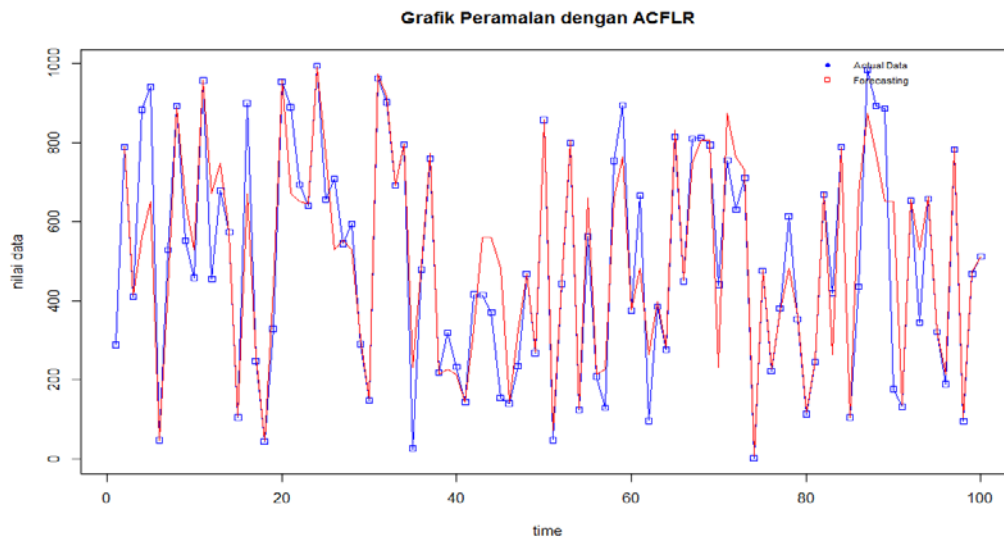
| MSE | Data Simulasi | | | |
|-------|---------------|------------|------------|------------|
| | 100 | 200 | 500 | 1000 |
| (1) | (2) | (3) | (4) | (5) |
| p = 1 | 13035,0790 | 17474,7658 | 13151,0643 | 18416,8704 |
| p = 2 | 8939,4210 | 12113,0898 | 8547,4180 | 12421,1192 |
| p = 3 | 7342,0190 | 8827,0119 | 5898,1555 | 6756,9242 |
| p = 4 | 3428,5670 | 4934,6702 | 5443,3189 | 4709,6978 |
| p = 5 | 1091,2770 | 3172,5853 | 3068,6547 | 3227,9705 |
| p = 6 | | | | 1929,5382 |
| p = 7 | | | | 1853,0785 |
| p = 8 | | | | 1545,3680 |
| p = 9 | | | | 1304,7569 |

Tabel 2. MAPE ACFLR pada Data Simulasi Menurut Jumlah Observasi dan Sub Interval

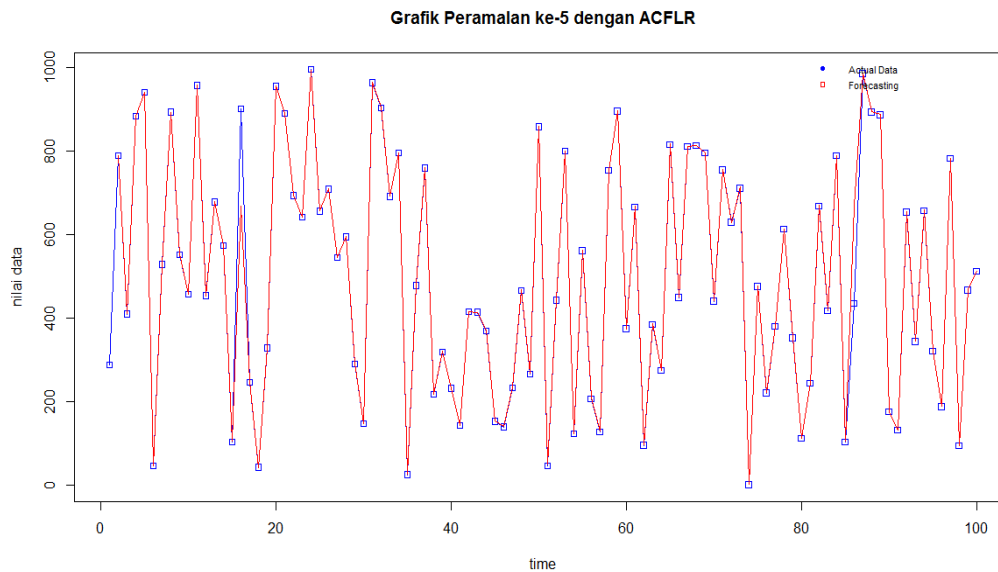
| MAPE | Data Simulasi | | | |
|---------|---------------|------------|------------|------------|
| | 100 | 200 | 500 | 1000 |
| (1) | (2) | (3) | (4) | (5) |
| $p = 1$ | 0,29459069 | 0,53999157 | 0,25628030 | 0,54296804 |
| $p = 2$ | 0,21495755 | 0,40041408 | 0,16208453 | 0,40666209 |
| $p = 3$ | 0,1754952 | 0,34942241 | 0,12927325 | 0,12945741 |
| $p = 4$ | 0,1410288 | 0,13456531 | 0,11437667 | 0,18091681 |
| $p = 5$ | 0,02056223 | 0,07269669 | 0,06817082 | 0,11156836 |
| $p = 6$ | | | | 0,03450182 |
| $p = 7$ | | | | 0,03625742 |
| $p = 8$ | | | | 0,02808764 |
| $p = 9$ | | | | 0,02657671 |

Berdasarkan Gambar 1 dan Gambar 2 dapat dilihat perbedaan nilai peramalan data simulasi sejumlah 100 antara sub interval pertama ($p = 1$) dengan sub interval kelima belas ($p = 5$). Pada gambar 1 terlihat bahwa masih banyak terjadi *error* sehingga nilai MSE dan MAPE pada $p = 1$ relatif besar. Lain halnya dengan Gambar 2, dimana pada gambar tersebut menunjukkan data aktual dan ramalan relatif sama sehingga nilai MSE dan MAPE nya kecil. Dapat dikatakan bahwa nilai peramalan pada $p = 5$ lebih baik dibandingkan dengan nilai peramalan pada $p = 1$.

Berdasarkan Gambar 1 dapat dilihat perbedaan nilai peramalan data simulasi sejumlah 100 antara sub interval pertama ($p = 1$) dengan sub interval kelima belas ($p = 5$). Pada Gambar 1 terlihat bahwa masih banyak terjadi *error* sehingga nilai MSE dan MAPE pada $p = 1$ relatif besar. Lain halnya dengan Gambar 2, dimana pada gambar tersebut menunjukkan data aktual dan ramalan relatif sama sehingga nilai MSE dan MAPE nya kecil. Dapat dikatakan bahwa nilai peramalan pada $p = 5$ lebih baik dibandingkan dengan nilai peramalan pada $p = 1$.



Gambar 1. Grafik Nilai Peramalan ACFLR untuk $p = 1$ pada $n = 100$



Gambar 2. Grafik Nilai Peramalan ACFLR untuk $p = 5$ pada $n = 100$

4.2 Perbandingan Ketepatan Peramalan antara AFCLR dengan *Single Exponential Smoothing*

Perbandingan ketepatan kedua metode yaitu ACFLR dan *singleexponential smoothing* dapat dilakukan dengan melihat nilai MSE dan MAPE dari masing-masing metode. Metode yang memiliki MSE dan MAPE yang lebih kecil memiliki ketepatan peramalan yang lebih baik.

Tabel 3. Perbandingan Nilai MSE pada Data Simulasi dengan Metode *Single Exponential Smoothing* dan ACFLR

| N | <i>Single Exponential Smoothing</i> | ACFLR |
|------|-------------------------------------|-----------|
| (1) | (2) | (3) |
| 100 | 87448,84957 | 1091,2770 |
| 200 | 79807,57664 | 3172,5853 |
| 500 | 84475,698787 | 3068,6547 |
| 1000 | 85031,85231 | 1304,7569 |

Berdasarkan Tabel 3 dapat dilihat bahwa MSE yang dihasilkan dengan menggunakan metode ACFLR pada data simulasi relatif lebih kecil dibandingkan metode *single exponential smoothing*. Hal ini berarti bahwa metode ACFLR memiliki ketepatan peramalan yang lebih akurat dibandingkan dengan metode *single exponential smoothing*.

Berdasarkan Tabel 4 dapat dilihat bahwa MAPE yang dihasilkan dengan menggunakan metode ACFLR pada data simulasi relatif lebih kecil dibandingkan metode *single exponential smoothing*. Hal ini berarti bahwa metode ACFLR memiliki ketepatan peramalan yang lebih akurat dibandingkan dengan metode *single exponential smoothing*.

Tabel 4. Perbandingan Nilai MAPE pada Data Simulasi dengan Metode *Single Exponential Smoothing* dan ACFLR.

| N | <i>Single Exponential Smoothing</i> | ACFLR |
|----------|--|--------------|
| (1) | (2) | (3) |
| 100 | 108,72190 | 0,02056223 |
| 200 | 99,43176 | 0,07269669 |
| 500 | 145,78030 | 0,06817082 |
| 1000 | 192,28260 | 0,02657671 |

5. Kesimpulan

Pada penelitian ini dapat diambil kesimpulan bahwa metode ACFLR memiliki ketepatan peramalan yang lebih baik dibanding dengan *single exponential smoothing*. Hal ini dibuktikan oleh nilai MSE dan MAPE yang dihasilkan ACFLR lebih kecil dibandingkan MSE dan MAPE yang dihasilkan oleh *single exponential smoothing*.

DAFTAR PUSTAKA

1. Chen, Wang, dan Pan, Forecasting Enrollments Using Automatic Clustering Techniques and Fuzzy Logical Relationships, *Experts System with Applications*, 2009, 36: 11070-11076,.
2. Kurniawan, R., *Metode Automatic Clustering- Fuzzy Logic Relationships untuk Peramalan Data Univariate* [Tesis], Institut Teknologi Sepuluh Nopember, Surabaya, 2011
3. Kusumadewi dan Purnomo, *Aplikasi Logika Fuzzy dan Pendukung Keputusan*. Graha Ilmu, Yogyakarta, 2003.
4. Makridakis S., Wheelwright S.C, dan McGee V.E, Jilid 1 edisi kedua, terjemahan Untung S. Andriyanto dan Abdul Basith. *Metode dan Aplikasi Peramalan*. Erlangga, Jakarta, 1999.
5. Raharja, Alda, dkk, Penerapan Metode Exponential Smoothing untuk Peramalan Penggunaan Waktu Telepon Dipt.Telkomsel DIVRE3 Surabaya. *SISFO Jurnal Sistem Informasi*, 1-9. 28 Maret 2015. <https://www.academia.edu/5477647/ITS-Undergraduate-14344-paperpdf>, 2010.
6. Song dan Chissom, Fuzzy Time Series and its models. *Fuzzy Sets and Systems*, 1993, Vol. 54: 269-277.
7. Zadeh, L.A, Prof, *Fuzzy Sets*. University of California, California, 1965.
8. Zainun, N. Y. dan Majid, M. Z. A, *Low Cost House Demand Predictor*, Universitas Teknologi Malaysia, 2003.