

LAPORAN PRAKTIKUM
Analisis Runtun Waktu
Modul 8: *Extreme Learning Machine*



Nama Praktikan	Nomor Mahasiswa	Tanggal Kumpul	Tanda Tangan Praktikan
Dian Widya Lestari	19611129		

Nama Penilai	Tanggal Koreksi	Nilai	Tanda tangan	
			Asisten	Dosen
Duhania Oktasya Mahara				
Puspita Putri Nabilah				
Mujiati Dwi Kartikasari				

JURUSAN STATISTIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA
2021

Daftar Isi

Halaman sampul.....	i
Daftar Isi.....	ii
Daftar Tabel	iii
Daftar Gambar.....	iv
1 Pendahuluan	1
1.1 <i>Extreme Learning Machine</i>	1
1.1.1 Struktur <i>Extreme Learning Machine</i> (ELM)	1
1.2 Normalisasi Data	2
1.3 Proses <i>Training</i>	2
1.4 Proses <i>Testing</i>	3
2 Deskripsi Kerja.....	4
2.1 Studi Kasus.....	4
2.2 Langkah Kerja	4
3 Pembahasan.....	10
3.1 Data Studi Kasus	10
3.2 Analisis <i>Extreme Learning Machine</i>	12
3.2.1 Model Pertama <i>Hidden Layer</i> 2	12
3.2.2 Model Kedua <i>Hidden Layer</i> 3.....	13
3.2.3 Model Ketiga <i>Hidden Layer</i> 4	15
3.2.4 Model Keempat <i>Hidden Layer</i> 5.....	16
3.2.5 Model Kelima <i>Hidden Layer</i> 6	18
3.2.6 Model Keenam <i>Hidden Layer</i> 7	19
3.2.7 Model Ketujuh <i>Hidden Layer</i> 8	20
3.2.8 Model Kedelapan <i>Hidden Layer</i> 9	21
3.2.9 Model Kesembilan <i>Hidden Layer</i> 10	23
3.3 Evaluasi Kinerja Model	24
4 Penutup.....	27
4.1 Kesimpulan.....	27
5 Daftar Pustaka	28

Daftar Tabel

Tabel 3.1. Pemilihan Model Terbaik	24
---	----

Daftar Gambar

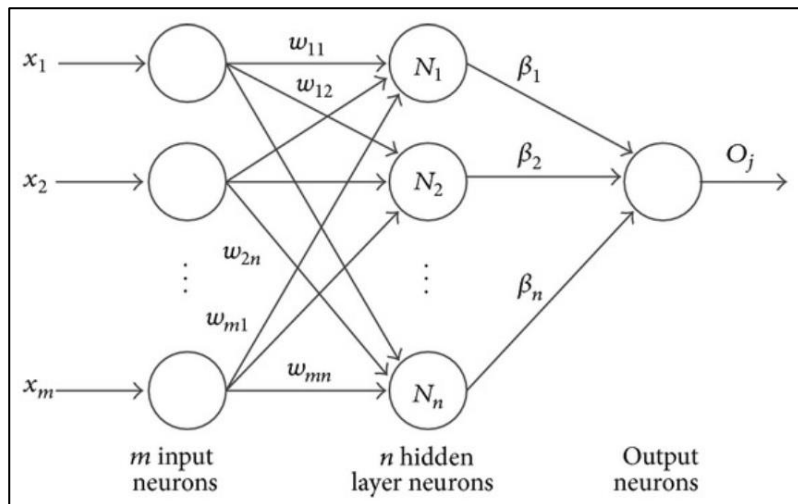
Gambar 2.1. Proses <i>Input Data</i>	4
Gambar 2.2. Pembagian Rasio Data <i>Training</i> dan <i>Testing</i>	4
Gambar 2.3. Sintaks Model 1	5
Gambar 2.4. Sintaks Model 2	5
Gambar 2.5. Sintaks Model 5	6
Gambar 2.6. Sintaks Model 4	6
Gambar 2.7. Sintaks Model 5	7
Gambar 2.8. Sintaks Model 7	7
Gambar 2.9. Sintaks Model 7	8
Gambar 2.10. Sintaks Model 8	8
Gambar 2.11. Sintaks Model 9	9
Gambar 2.12. Sintaks Plot Data Aktual, <i>Fitted</i> , dan <i>Forecasting</i>	9
Gambar 3.1. Data Aktual	10
Gambar 3.2. Data <i>Time Series</i>	11
Gambar 3.3. Plot Data KA	11
Gambar 3.4. Plot ELM Model 1	12
Gambar 3.5. <i>Fitted Value</i> dan Peramalan Model 1	13
Gambar 3.6. Plot ELM Model 2	13
Gambar 3.7. <i>Fitted Value</i> dan Peramalan Model 2	14
Gambar 3.8. Nilai <i>Error Data Testing</i> 2	14
Gambar 3.9. Plot ELM Model 3	15
Gambar 3.10. Nilai <i>Error Data Testing</i> 3	15
Gambar 3.11. <i>Fitted Value</i> dan Peramalan Model 3	16
Gambar 3.12. Plot ELM Model 4	16
Gambar 3.13. Nilai <i>Error Data Testing</i> 4	17
Gambar 3.14. <i>Fitted Value</i> dan Peramalan Model 4	17
Gambar 3.15. Plot ELM Model 5	18
Gambar 3.16. Nilai <i>Error Data Testing</i> 5	18
Gambar 3.17. <i>Fitted Value</i> dan Peramalan Model 5	18
Gambar 3.18. Plot ELM Model 6	19
Gambar 3.19. Nilai <i>Error Data Testing</i> 6	19
Gambar 3.20. <i>Fitted Value</i> dan Peramalan Model 6	20
Gambar 3.21. Plot ELM Model 7	20
Gambar 3.22. Nilai <i>Error Data Testing</i> 7	20
Gambar 3.23. <i>Fitted Value</i> dan Peramalan Model 7	21
Gambar 3.24. Plot ELM Model 8	21
Gambar 3.25. Nilai <i>Error Data Testing</i> 8	22
Gambar 3.26. <i>Fitted Value</i> dan Peramalan Model 8	22
Gambar 3.27. Plot ELM Model 9	23
Gambar 3.28. Nilai <i>Error Data Testing</i> 9	23
Gambar 3.29. <i>Fitted Value</i> dan Peramalan Model 9	23
Gambar 3.30. Grafik Model (7) Terbaik	25
Gambar 3.31. Grafik Model (9) Terbaik	26

1 Pendahuluan

1.1 Extreme Learning Machine

Extreme Learning Machine (ELM) merupakan jenis jaringan syaraf tiruan dengan satu *hidden layer* (lapisan tersembunyi) yang disebut dengan *single hidden layer feedforward neural network* (SLNs). Metode ini memiliki kecepatan pembelajaran yang lebih cepat dibandingkan metode jaringan syaraf tiruan konvensional seperti *backpropagation*. Metode ELM dibuat untuk mengatasi kelemahan dari jaringan syaraf tiruan (*feedforward*) terutama dalam hal *learning speed*. Berikut adalah gambaran arsitektur jaringan syaraf tiruan *Extreme Learning Machine* (Fikriya et al., 2017).

1.1.1 Struktur Extreme Learning Machine (ELM)



Jaringan *feedforward* menggunakan parameter-parameter yang ditentukan secara manual seperti *input weight* dan bias (Zulinda, 2020). *Input weight* dan bias ini dibangkitkan secara acak dalam satu rentang tertentu, maka dari itu praktikan dalam hasil analisis akan mendapatkan hasil analisis yang berbeda setiap praktikan running sintaks dengan bantuan *R Studio*. Dengan nilai yang diacak tersebut dapat menghindari hasil prediksi yang tidak stabil.

Keterangan:

y = Nilai *output layer*

β = *Output weight*

X = *Input data* yang digunakan

W = Bobot *input*

N = Jumlah *input neuron*

1.2 Normalisasi Data

Normalisasi data adalah proses perubahan bentuk data menjadi nilai yang lebih spesifik dalam rentang 0-1 (Alfiyatin et al., 2018). Salah satu metode yang dipakai yaitu dengan menggunakan metode *Min-Max Normalization*. Metode tersebut adalah metode normalisasi dengan melakukan transformasi linear terhadap data asli sehingga dihasilkan keseimbangan nilai perbandingan antar data saat sebelum dan sesudah proses.

$$X'_i = X_{minb} + \frac{(X_t - X_{min})(X_{maxb} - X_{minb})}{(X_{max} - X_{min})}$$

Keterangan:

X'_i = Nilai normalisasi

X_t = Nilai data aktual atau observasi

X_{min} = Nilai minimum pada data aktual

X_{max} = Nilai maksimum pada data aktual

X_{minb} = Nilai minimum baru

X_{maxb} = Nilai maksimum baru

1.3 Proses *Training*

Proses *training* harus dilalui sebelum melakukan proses prediksi. Tujuannya adalah untuk mendapatkan nilai *output weight*. Langkah-langkah proses *training* yaitu:

- Langkah pertama adalah menginialisasi *weight* dan bias. Nilai ini diinisialisasi secara acak.
- Tiap unit masukan $X_i (i = 1, 2, 3, \dots, n)$ menerima sinyal dan meneruskan sinyal tersebut ke semua unit pada *hidden layer*.

- c. Keluaran di *hidden layer* dihitung menggunakan fungsi aktivasi.
- d. Menghitung *output weight*. Dalam mendapatkan *output weight* proses pertama yang dilakukan yaitu mentranspose matriks hasil keluaran *hidden layer* dengan fungsi aktivasi (Zulinda, 2020).

1.4 Proses Testing

Pada proses ini bertujuan untuk mengevaluasi metode ELM dari hasil proses *training* sebelumnya. Lalu pada proses ini juga untuk mengetahui akurasi dari program (Alfiyatin et al., 2018).

- a. Mengetahui nilai W_{min} , b dan β dari proses *training*.
- b. Menghitung nilai matriks keluaran pada *hidden layer*.
- c. Menghitung hasil keluaran yang diperoleh dengan menggunakan $Y = H * \beta$.
- d. Denormalisasi hasil prediksi menggunakan rumus denormalisasi data.
- e. Terakhir menghitung evaluasi dari pembentukan model dengan RMSE, MSE, ataupun MAPE.

2 Deskripsi Kerja

2.1 Studi Kasus

1. Gunakan data penumpang kereta api di Indonesia.
2. Lakukan analisis ELM dengan ketentuan berikut:
 - a. Pembagian *training* dan *testing* adalah 80%:20%.
 - b. Uji jumlah *neuron* pada *hidden layer* dengan menggunakan *neuron* 2-10.
 - c. Lanjutkan analisis ELM dengan jumlah *neuron* yang memberikan *error* terkecil dari kriteria b.

2.2 Langkah Kerja

1. Pertama, praktikan lakukan *import data* ke dalam *R* dengan sintaks `read.csv` sebagai berikut.

```
library(nnfor)
library(forecast)

# Data tanpa musiman
ka = read.csv("C:/Users/DIANWL/Documents/# Semester 5/6 PRAK ARW/dataKA.csv")
View(ka)
```

Gambar 2.1. Proses *Input Data*

2. Kemudian menentukan jumlah *data training* dan *test* dengan perbandingan rasio sebesar 80%:20%.

```
# Data train dan test dg rasio 80%:20%
n = round(0.80*NROW(ka.ts), 0)
train = window(ka.ts, start = c(2006,1), end = c(2013, 12))
test = window(ka.ts, start = c(2014,1))
```

Gambar 2.2. Pembagian Rasio Data *Training* dan *Testing*

3. Selanjutnya praktikan membuat model dengan *hidden layer* 2-10. Model pertama dengan *hidden layer* sebesar 2 *neurons*.

```
# Model ELM
# model 1 hd=2 #####
model1 = elm(train,
  m = frequency(ka.ts),
  hd = 2,
  type = c("lasso", "ridge", "step", "lm"),
  reps = 20, comb = c("median", "mean", "mode"),
  lags = NULL, keep = NULL, difforder = NULL,
  outplot = TRUE, sel.lag = FALSE,
  direct = FALSE,
  allow.det.season = FALSE,
  det.type = c("auto", "bin", "trg"),
  xreg = NULL, xreg.lags = NULL, xreg.keep = NULL,
  barebone = TRUE,
  model = NULL,
  retrain = TRUE)

plot(model1)
model1$MSE
model1$lags
model1$fitted
pred1 = forecast(model1, 24)
# Ukuran kesalahan untuk data testing
error_1 = test-pred1$mean
rmse_1 = sqrt(model1$MSE)
mape_1 = mean(abs(error_1)/test)*100
```

Gambar 2.3. Sintaks Model 1

4. Kemudian membuat model yang kedua dengan *hidden layer* sebesar 3 *neuron*.

```
# model 2 hd=3 #####
model2 = elm(train,
  m = frequency(ka.ts),
  hd = 3,
  type = c("lasso", "ridge", "step", "lm"),
  reps = 20, comb = c("median", "mean", "mode"),
  lags = NULL, keep = NULL, difforder = NULL,
  outplot = TRUE, sel.lag = FALSE,
  direct = FALSE,
  allow.det.season = FALSE,
  det.type = c("auto", "bin", "trg"),
  xreg = NULL, xreg.lags = NULL, xreg.keep = NULL,
  barebone = TRUE,
  model = NULL,
  retrain = TRUE)

plot(model2)
model2$MSE
model2$lags
model2$fitted
pred2 = forecast(model2, 24)
# Ukuran kesalahan untuk data testing
error_2 = test-pred2$mean
rmse_2 = sqrt(model2$MSE)
mape_2 = mean(abs(error_2)/test)*100
```

Gambar 2.4. Sintaks Model 2

5. Berikutnya model ketiga dengan *hidden layer* dengan banyak *neuron* 4.

```
# model 3 hd=4 #####
model3 = elm(train,
  m = frequency(ka.ts),
  hd = 4,
  type = c("lasso", "ridge", "step", "lm"),
  reps = 20, comb = c("median", "mean", "mode"),
  lags = NULL, keep = NULL, difforder = NULL,
  outplot = TRUE, sel.lag = FALSE,
  direct = FALSE,
  allow.det.season = FALSE,
  det.type = c("auto", "bin", "trg"),
  xreg = NULL, xreg.lags = NULL, xreg.keep = NULL,
  barebone = TRUE,
  model = NULL,
  retrain = TRUE)

plot(model3)
model3$MSE
model3$lags
model3$fitted
pred3 = forecast(model3, 24)
# Ukuran kesalahan untuk data testing
error_3 = test-pred3$mean
rmse_3 = sqrt(model3$MSE)
mape_3 = mean(abs(error_3)/test)*100
```

Gambar 2.5. Sintaks Model 5

6. Lalu dibentuk model keempat dengan banyaknya *neuron* 5.

```
# model 4 hd=5 #####
model4 = elm(train,
  m = frequency(ka.ts),
  hd = 5,
  type = c("lasso", "ridge", "step", "lm"),
  reps = 20, comb = c("median", "mean", "mode"),
  lags = NULL, keep = NULL, difforder = NULL,
  outplot = TRUE, sel.lag = FALSE,
  direct = FALSE,
  allow.det.season = FALSE,
  det.type = c("auto", "bin", "trg"),
  xreg = NULL, xreg.lags = NULL, xreg.keep = NULL,
  barebone = TRUE,
  model = NULL,
  retrain = TRUE)

plot(model4)
model4$MSE
model4$lags
model4$fitted
pred4 = forecast(model4, 24)
# Ukuran kesalahan untuk data testing
error_4 = test-pred4$mean
rmse_4 = sqrt(model4$MSE)
mape_4 = mean(abs(error_4)/test)*100
```

Gambar 2.6. Sintaks Model 4

7. Selanjutnya model ELM dibentuk dengan jumlah *neuron* 6.

```
# model 5 hd=6 #####
model5 = elm(train,
  m = frequency(ka.ts),
  hd = 6,
  type = c("lasso", "ridge", "step", "lm"),
  reps = 20, comb = c("median", "mean", "mode"),
  lags = NULL, keep = NULL, difforder = NULL,
  outplot = TRUE, sel.lag = FALSE,
  direct = FALSE,
  allow.det.season = FALSE,
  det.type = c("auto", "bin", "trg"),
  xreg = NULL, xreg.lags = NULL, xreg.keep = NULL,
  barebone = TRUE,
  model = NULL,
  retrain = TRUE)

plot(model5)
model5$MSE
model5$lags
model5$fitted
pred5 = forecast(model5, 24)
# Ukuran kesalahan untuk data testing
error_5 = test-pred5$mean
rmse_5 = sqrt(model5$MSE)
mape_5 = mean(abs(error_5)/test)*100
```

Gambar 2.7. Sintaks Model 5

8. Selanjutnya membuat model keenam *hidden layer* dengan jumlah *neuron* sebanyak 7.

```
# model 6 hd=7 #####
model6 = elm(train, |
  m = frequency(ka.ts),
  hd = 7,
  type = c("lasso", "ridge", "step", "lm"),
  reps = 20, comb = c("median", "mean", "mode"),
  lags = NULL, keep = NULL, difforder = NULL,
  outplot = TRUE, sel.lag = FALSE,
  direct = FALSE,
  allow.det.season = FALSE, #data bukan musiman maka
  det.type = c("auto", "bin", "trg"),
  xreg = NULL, xreg.lags = NULL, xreg.keep = NULL,
  barebone = TRUE,
  model = NULL,
  retrain = TRUE)

plot(model6)
model6$MSE
model6$lags
model6$fitted
pred6 = forecast(model6, 24)
# Ukuran kesalahan untuk data testing
error_6 = test-pred6$mean
rmse_6 = sqrt(model6$MSE)
mape_6 = mean(abs(error_6)/test)*100
```

Gambar 2.8. Sintaks Model 7

9. Dibentuk model selanjutnya model ketujuh dengan *hidden layer* sebesar 8 *neurons*.

```
# model 7 hd=8 #####
model7 = elm(train,
  m = frequency(ka.ts),
  hd = 8,
  type = c("lasso", "ridge", "step", "lm"),
  reps = 20, comb = c("median", "mean", "mode"),
  lags = NULL, keep = NULL, difforder = NULL,
  outplot = TRUE, sel.lag = FALSE,
  direct = FALSE,
  allow.det.season = FALSE,
  det.type = c("auto", "bin", "trg"),
  xreg = NULL, xreg.lags = NULL, xreg.keep = NULL,
  barebone = TRUE,
  model = NULL,
  retrain = TRUE)

plot(model7)
model7$MSE
model7$lags
model7$fitted
pred7 = forecast(model7, 24)
# Ukuran kesalahan untuk data testing
error_7 = test-pred7$mean
rmse_7 = sqrt(model7$MSE)
mape_7 = mean(abs(error_7)/test)*100
```

Gambar 2.9. Sintaks Model 7

10. Berikutnya dibentuk model kedelapan dengan *hidden layer* jumlah *neuron* 9.

```
# model 8 hd=9 #####
model8 = elm(train,
  m = frequency(ka.ts),
  hd = 9,
  type = c("lasso", "ridge", "step", "lm"),
  reps = 20, comb = c("median", "mean", "mode"),
  lags = NULL, keep = NULL, difforder = NULL,
  outplot = TRUE, sel.lag = FALSE,
  direct = FALSE,
  allow.det.season = FALSE,
  det.type = c("auto", "bin", "trg"),
  xreg = NULL, xreg.lags = NULL, xreg.keep = NULL,
  barebone = TRUE,
  model = NULL,
  retrain = TRUE)

plot(model8)
model8$MSE
model8$lags
model8$fitted
pred8 = forecast(model8, 24)
# Ukuran kesalahan untuk data testing
error_8 = test-pred8$mean
rmse_8 = sqrt(model8$MSE)
mape_8 = mean(abs(error_8)/test)*100
```

Gambar 2.10. Sintaks Model 8

11. Dan bentuk model terakhir dengan *hidden layer* sebesar 10 *neuron* sebagai berikut.

```
# model 9 hd=10 #####
model9 = elm(train, |
  m = frequency(ka.ts),
  hd = 10,
  type = c("lasso", "ridge", "step", "lm"),
  reps = 20, comb = c("median", "mean", "mode"),
  lags = NULL, keep = NULL, difforder = NULL,
  outplot = TRUE, sel.lag = FALSE,
  direct = FALSE,
  allow.det.season = FALSE, #data bukan musiman maka
  det.type = c("auto", "bin", "trg"),
  xreg = NULL, xreg.lags = NULL, xreg.keep = NULL,
  barebone = TRUE,
  model = NULL,
  retrain = TRUE)

plot(model9)
model9$MSE
model9$lags
model9$fitted
pred9 = forecast(model9, 24)
# Ukuran kesalahan untuk data testing
error_9 = test-pred9$mean
rmse_9 = sqrt(model9$MSE)
mape_9 = mean(abs(error_9)/test)*100
```

Gambar 2.11. Sintaks Model 9

12. Setelah membuat model dengan *hidden layer* yang telah ditentukan, praktikan buat plot data aktual, *fitted value*, dan prediksi (peramalan) dari model terbaik.

```
# Plot data aktual, fitted, prediksi
autoplot(ka.ts, color = 'blue') +
  autolayer(model9$fitted, color = 'red')
  autolayer(pred9, color = 'green')
```

*terbentuk 2 model >>
model 7 dan model 9

Gambar 2.12. Sintaks Plot Data Aktual, *Fitted*, dan *Forecasting*

3 Pembahasan

3.1 Data Studi Kasus

Sebelumnya praktikan telah memaparkan langkah kerja dalam analisis ELM. Pada bagian ini akan dijelaskan secara rinci mengenai pengerjaan studi kasus. Pertama, praktika memasukkan data KA ke dalam *R*. Data tersebut merupakan data *time series* dimana waktu yang diambil berurut mulai dari 2006-2015, berikut datanya.

	Tahun	Bulan	Total
1	2006	1	11828
2	2006	2	11931
3	2006	3	13314
4	2006	4	12909
5	2006	5	13575
⋮	⋮	⋮	⋮
115	2015	7	27612
116	2015	8	27796
117	2015	9	27549
118	2015	10	28718
119	2015	11	27669
120	2015	12	29831

Gambar 3.1. Data Aktual

Berdasarkan **Gambar 3.1** data Kereta Api memiliki tiga variabel yaitu tahun, bulan, dan total (penumpang), dengan banyaknya data sebesar 120. Dikarenakan data sudah terurut dari waktu terlama hingga terbaru, praktikan tidak perlu melakukan proses pengurutan data.

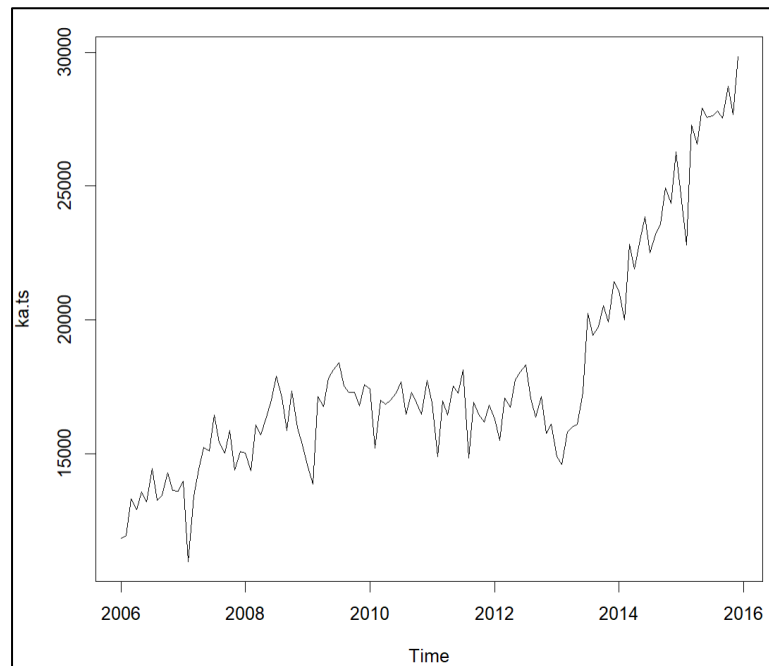
Selanjutnya praktikan melakukan proses tahapan pertama analisis dengan data KA yang masih berbentuk dataframe, disini praktikan ubah data KA ke bentuk *data time series*. Menggunakan perintah dalam *R* yaitu `ts ()` untuk mengubah data menjadi objek *time series*. Data *time series* KA disimpan dalam objek `ka.ts` berikut datanya.

```
> ka.ts
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2006	11828	11931	13314	12909	13575	13203	14433	13255	13436	14290	13631	13614
2007	13960	10969	13409	14415	15232	15104	16454	15419	15033	15866	14391	15084
2008	15027	14378	16071	15711	16363	17010	17887	17108	15879	17337	15973	15332
2009	14494	13869	17132	16775	17824	18143	18385	17527	17281	17281	16778	17581
2010	17424	15207	16992	16832	16988	17259	17680	16477	17301	16908	16469	17733
2011	16891	14890	16978	16441	17522	17265	18132	14846	16921	16461	16179	16811
2012	16283	15490	17090	16746	17771	18062	18309	17056	16368	17127	15773	16104
2013	14900	14594	15826	16000	16113	17301	20245	19423	19738	20534	19919	21417
2014	21092	19998	22836	21908	22988	23840	22500	23199	23593	24923	24356	26275
2015	24676	22790	27267	26565	27910	27562	27612	27796	27549	28718	27669	29831

Gambar 3.2. Data Time Series

Pada sintaks data *time series* terdapat atribut `start = c (2006,1)` menandakan bahwa data KA dimulai dari tahun 2006 bulan Januari (1). Kemudian atribut `freq = 12` menunjukkan bahwa periode yang digunakan adalah bulanan (12). Setelah data diubah menjadi bentuk data runtun waktu maka dapat dilihat pola data sebagai berikut.



Gambar 3.3. Plot Data KA

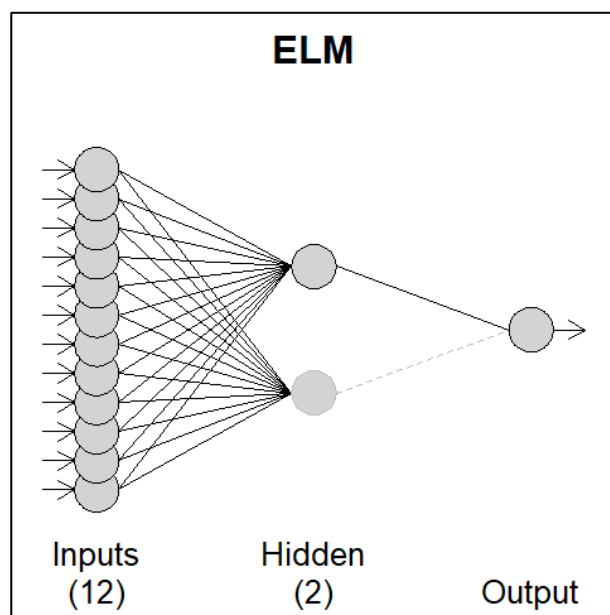
Berdasarkan hasil grafik di atas **Gambar 3.3** data KA tidak memiliki pola seasonal atau musiman. Setelah itu dilakukan analisis *extreme learning machine* (ELM).

3.2 Analisis Extreme Learning Machine

3.2.1 Model Pertama Hidden Layer 2

Selanjutnya pada tahap ini data akan dibagi menjadi dua kelompok yaitu *data training* dan *data testing*. Data tersebut akan dibagi dengan komposisi rasio sebesar 80% untuk *data training* dan sisanya 20% sebagai *data testing*. Pada **Gambar 2.2** praktikan lakukan perhitungan dalam objek *n* dengan fungsi `round`. Fungsi `round` berguna untuk membulatkan nilai dalam argumen pertamanya ke jumlah tempat desimal yang ditentukan dimana *default* 0. Didapatkan nilai *n* sebesar 96 data, artinya data *training* yang digunakan sebanyak 96 data dan sisanya dipakai untuk *data testing*. Pada objek *train* data diambil mulai dari Januari 2016 sampai (*end*) bulan Desember 2013. Lalu objek *test* dimulai dari sisanya maka bulan Januari 2014.

Fungsi dari *training* adalah bagian proses pengenalan pola-pola yang telah dinormalisasikan agar sistem dapat menentukan *weight* atau bobot yang memetakan antara *input* dan target *output* yang diinginkan. Studi kasus diatas diperintahkan bahwa jumlah *neuron* sebesar 2 sampai 10 *neurons*. *Running* sintaks pada **Gambar 2.3** lalu didapatkan hasil berikut ini.



Gambar 3.4. Plot ELM Model 1

Selanjutnya untuk mendapatkan pengukuran kesalahan *error* pada data testing model pertama digunakan sintaks `modell$MSE`, `rmse_1 = sqrt(modell$MSE)`, dan `mape_1 = mean(abs(error_1)/test)*100`, didapatkan hasil berturut-turut sebagai berikut 1384160, 1176.503, dan 10.42104. Dengan *fitted value* dan hasil *forecasting* pada model 1 untuk 24 periode kedepan, sebagai berikut * hasil *fitted value* (kecocokan) adalah untuk melihat jarak atau perbedaan dari model yang dibuat jika value tidak jauh hslny dr data aktual maka model tersebut baik/layak di use

```
> modell$fitted
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2007	14049.84	11058.84	13498.84	14504.84	15321.84	15193.84	16543.84	15508.84	15122.84	15955.84	14480.84	
2008	15173.84	15116.84	14467.84	16160.84	15800.84	16452.84	17099.84	17976.84	17197.84	15968.84	17426.84	16062.84
2009	15421.84	14583.84	13958.84	17221.84	16864.84	17913.84	18232.84	18474.84	17616.84	17370.84	17370.84	16867.84
2010	17670.84	17513.84	15296.84	17081.84	16921.84	17077.84	17348.84	17769.84	16566.84	17390.84	16997.84	16558.84
2011	17822.84	16980.84	14979.84	17067.84	16530.84	17611.84	17354.84	18221.84	14935.84	17010.84	16550.84	16268.84
2012	16900.84	16372.84	15579.84	17179.84	16835.84	17860.84	18151.84	18398.84	17145.84	16457.84	17216.84	15862.84
2013	16193.84	14989.84	14683.84	15915.84	16089.84	16202.84	17390.84	20334.84	19512.84	19827.84	20623.84	20008.84

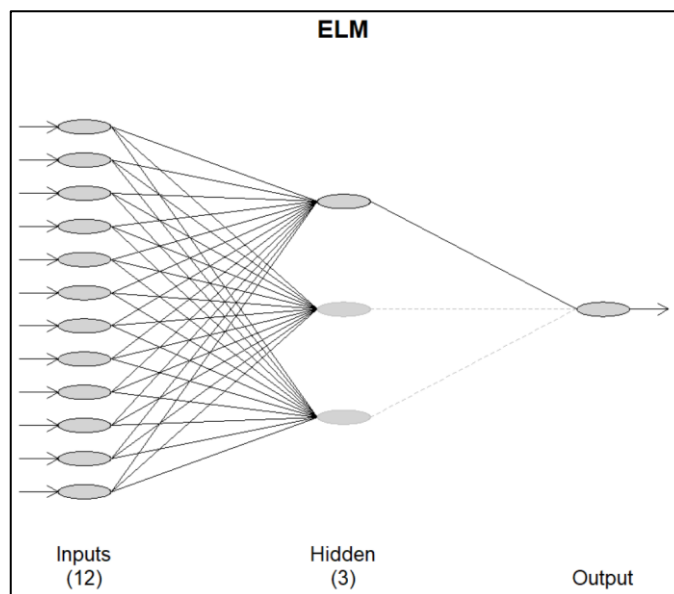
```
> pred1 = forecast(modell, 24)
> pred1
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2014	21506.84	21596.69	21686.53	21776.37	21866.22	21956.06	22045.90	22135.75	22225.59	22315.43	22405.28	22495.12
2015	22584.96	22674.81	22764.65	22854.49	22944.34	23034.18	23124.02	23213.87	23303.71	23393.55	23483.40	23573.24

Gambar 3.5. *Fitted Value* dan Peramalan Model 1

3.2.2 Model Kedua *Hidden Layer 3*

Berikutnya dibentuk model kedua dengan *hidden layer 3 neurons*. Pada model ini praktikan dapat *input* sintaks **Gambar 2.4** dan didapatkan hasil sebagai berikut.



Gambar 3.6. Plot ELM Model 2

Diperoleh bahwa dengan *inputs* sebesar 12 didapatkan *hidden layers* sebanyak 3 *neurons* dan ada satu *beta* ($\beta = 1$) yang berpengaruh terhadap nilai Y

ditunjukkan pada garis tegas. Selanjutnya diperoleh nilai *fitted value* dan peramalan dari pembentukan model ke dua.

```
> model2$fitted
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2007	14049.84	11107.76	13498.84	14504.84	15321.84	15244.92	16509.79	15520.09	15122.84	15955.84	14480.84	
2008	15200.42	14980.12	14467.84	16160.84	15800.84	16452.84	17126.69	17918.71	17197.84	15971.94	17310.78	16062.84
2009	15427.00	14580.35	13958.84	17221.84	16864.84	17913.84	18241.43	18474.84	17616.84	17370.84	17330.62	16867.84
2010	17670.84	17483.60	15467.34	17081.84	16921.84	17077.84	17348.84	17759.96	16585.36	17390.84	16928.05	16584.15
2011	17822.84	16900.68	15130.26	17067.84	16530.84	17611.84	17354.84	18118.02	15055.17	17010.84	16550.84	16280.18
2012	16900.84	16372.84	15712.54	17179.84	16835.84	17860.84	18169.06	18233.83	17305.62	16457.84	17151.47	15878.44
2013	16193.84	14988.80	14710.52	15915.84	16089.84	16202.84	17390.84	20273.24	19512.84	19827.84	20570.70	20008.84

```
> pred2 = forecast(model2, 24)
> pred2
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2014	21506.84	21596.69	21686.53	21776.37	21866.22	21956.06	22045.90	22135.75	22225.59	22315.43	22405.28	22495.12
2015	22584.96	22674.81	22764.65	22854.49	22944.34	23034.18	23124.02	23213.87	23303.71	23393.55	23483.40	23573.24

Gambar 3.7. *Fitted Value* dan Peramalan Model 2

Selanjutnya praktikan mengukur kesalahan pada *data testing* model ke dua dengan *input* sintaks **Gambar 2.5** diperoleh hasil berikut ini.

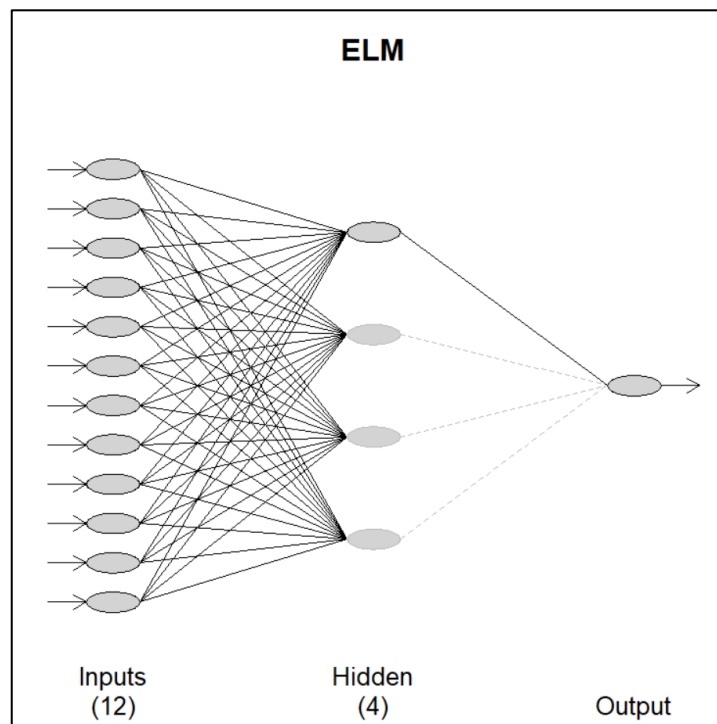
```
> model2$MSE
[1] 1384160
> rmse_2
[1] 1176.503
> mape_2
[1] 10.42104
```

Gambar 3.8. Nilai *Error Data Testing* 2

Didapatkan nilai pengukuran kesalahan model untuk *data testing*. Berikutnya dibentuk model ketiga dengan jumlah *neuron* yang telah ditentukan 4 pada *hidden layer*.

3.2.3 Model Ketiga *Hidden Layer 4*

Kemudian model ketiga dengan *hidden layer* sebanyak 4 *neurons*. Praktikan telah mengetahui untuk sintaks dalam pembentukan model ELM ini pada **Gambar 2.6** berikut hasilnya.



Gambar 3.9. Plot ELM Model 3

Berdasarkan **Gambar 3.9** bahwa terdapat 1 *beta* ($\beta = 1$) yang berpengaruh terhadap *Y*. setelah mengetahui bahwa terdapat variabel *beta* pada hasil grafil ELM, selanjutnya praktikan melakukan pengukuran kesalahan pada model ketiga pada sintaks `model3$MSE`, `rmse_3 = sqrt(model3$MSE)`, dan `mape_3 = mean(abs(error_3)/test)*100`, berikut hasilnya.

```
> model3$MSE
[1] 1352163
> rmse_3
[1] 1162.825
> mape_3
[1] 10.4165
```

Gambar 3.10. Nilai *Error Data Testing 3*

Selanjutnya dilakukan peramalan dengan 24 periode kedepan dan tentukan *fitted value* dan peramalan 24 periode kedepan.

```
> model3$fitted
```

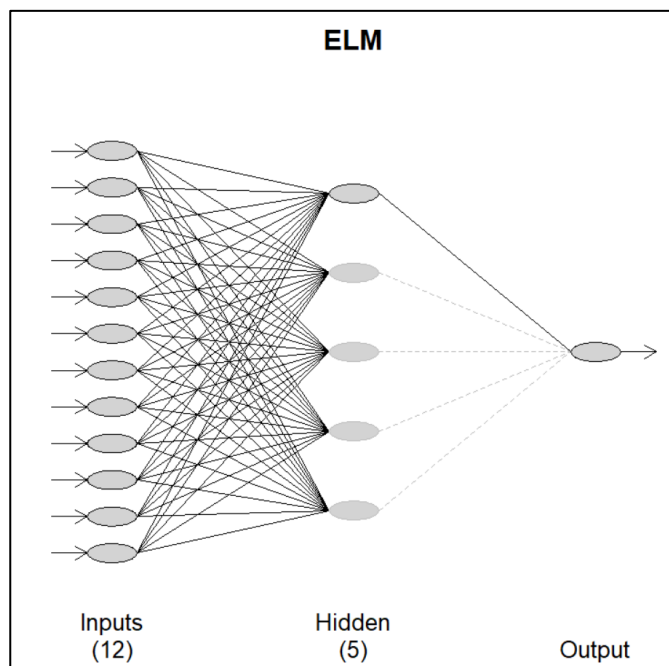
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2007		14036.43	11114.03	13498.52	14504.84	15321.84	15213.44	16498.03	15508.84	15161.71	15896.41	14480.84
2008	15173.84	15067.69	14467.84	16160.84	15800.84	16452.84	17103.92	17964.53	17197.84	15968.84	17363.63	16068.70
2009	15421.84	14573.79	13958.84	17221.84	16864.84	17913.84	18232.84	18474.84	17616.84	17370.84	17364.85	16867.84
2010	17670.84	17473.63	15347.38	17065.03	16921.84	17077.84	17348.84	17728.85	16566.84	17390.84	16981.75	16559.90
2011	17822.84	16944.26	15024.57	17066.99	16530.84	17611.84	17354.84	18173.53	14955.23	17010.84	16550.84	16268.84
2012	16900.84	16325.56	15579.84	17179.84	16835.84	17860.84	18162.12	18306.51	17219.26	16457.84	17206.80	15889.59
2013	16193.84	14989.84	14689.32	15915.84	16089.84	16202.84	17390.84	20320.55	19512.84	19827.84	20590.95	20043.08
> pred3 = forecast(model3, 24)												
> pred3												
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2014	21504.34	21572.72	21686.53	21776.37	21866.22	21950.59	22045.90	22135.75	22225.59	22315.43	22405.28	22495.12
2015	22584.96	22674.81	22764.65	22854.49	22944.34	23034.18	23124.02	23213.87	23303.71	23393.55	23483.40	23573.24

Gambar 3.11. *Fitted Value* dan Peramalan Model 3

Pada hasil **Gambar 3.11** prediksi tahun 2014 diperoleh nilai penumpang KA pada bulan Januari sampai desember mengalami kenaikan secara signifikan dengan jumlah penumpang KA pada akhir tahun 2014 sebesar 22495.12, selain itu pada tahun berikutnya 2015 pada bulan Januari sampai Desember juga mengalami kenaikan jumlah penumpang dengan awal tahun sebesar 22584.96 dan diakhir tahun 2015 sebanyak 23573.24 penumpang KA.

3.2.4 Model Keempat *Hidden Layer 5*

Selanjutnya model selanjutnya dengan *hidden layer 5 neurons*. Sama dengan pembuatan model sebelumnya praktikan *input* sintaks pada **Gambar 2.6** berikut ini hasil pembentukan model ELM.



Gambar 3.12. Plot ELM Model 4

Dari hasil **Gambar 3.12** diketahui plot atau grafik model keempat dengan *hidden layer* sebanyak 5 *neurons* hanya terdapat 1 *neuron* atau 1 beta yang memiliki pengaruh terhadap Y. Selanjutnya untuk mengetahui kesalahan peramalan pada model keempat digunakan sintaks `model4$MSE`, `rmse_4 = sqrt(model4$MSE)`, dan `mape_4 = mean(abs(error_4)/test)*100`, maka diperoleh sebagai berikut.

```
> rmse_4
[1] 1176.503
> mape_4
[1] 10.42104
> model4$MSE
[1] 1384160
```

Gambar 3.13. Nilai *Error Data Testing* 4

Setelah didapatkan nilai *error* pada *data testing*. Selanjutnya untuk melihat *fitted value* dan peramalan dengan model keempat didapatkan hasil sebagai berikut.

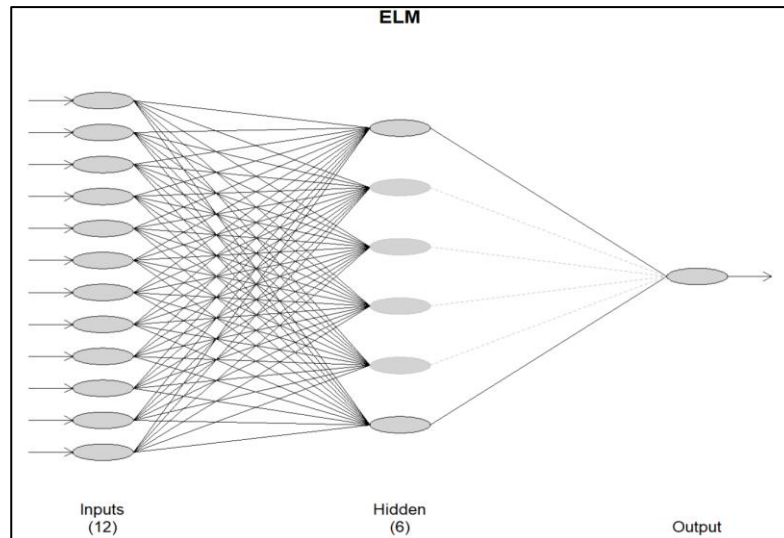
```
> model4$fitted
      Jan      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep      Oct      Nov      Dec
2007 14049.84 11058.84 13498.84 14504.84 15321.84 15193.84 16543.84 15508.84 15122.84 15955.84 14480.84
2008 15173.84 15116.84 14467.84 16160.84 15800.84 16452.84 17099.84 17976.84 17197.84 15968.84 17426.84 16062.84
2009 15421.84 14583.84 13958.84 17221.84 16864.84 17913.84 18232.84 18474.84 17616.84 17370.84 17370.84 16867.84
2010 17670.84 17513.84 15296.84 17081.84 16921.84 17077.84 17348.84 17769.84 16566.84 17390.84 16997.84 16558.84
2011 17822.84 16980.84 14979.84 17067.84 16530.84 17611.84 17354.84 18221.84 14935.84 17010.84 16550.84 16268.84
2012 16900.84 16372.84 15579.84 17179.84 16835.84 17860.84 18151.84 18398.84 17145.84 16457.84 17216.84 15862.84
2013 16193.84 14989.84 14683.84 15915.84 16089.84 16202.84 17390.84 20334.84 19512.84 19827.84 20623.84 20008.84
> pred4 = forecast(model4, 24)
> pred4
      Jan      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep      Oct      Nov      Dec
2014 21506.84 21596.69 21686.53 21776.37 21866.22 21956.06 22045.90 22135.75 22225.59 22315.43 22405.28 22495.12
2015 22584.96 22674.81 22764.65 22854.49 22944.34 23034.18 23124.02 23213.87 23303.71 23393.55 23483.40 23573.24
```

Gambar 3.14. *Fitted Value* dan Peramalan Model 4

Didapatkan hasil peramalan dan *fitted value* pada **Gambar 3.14** diketahui hasil prediksi untuk 24 periode kedepan atau selama dua tahun kedepan. Tahun 2014 diketahui nilai peramalan mengalami fluktuasi dari bulan Januari sampai bulan Desember. Kemudian tahun selanjutnya mengalami peningkatan fluktuasi dari awal tahun sampai Desember 2015.

3.2.5 Model Kelima *Hidden Layer* 6

Lalu pada model selanjutnya kelima *hidden layer* dengan 6 *neurons*. Dengan menggunakan sintaks **Gambar 2.7** didapatkan hasilnya sebagai berikut.



Gambar 3.15. Plot ELM Model 5

Dari hasil plot model ELM kelima didapatkan hasil bahwa terdapat 2 beta yang memiliki pengaruh terhadap Y. Setelah didapatkan bahwa terdapat satu *neuron* dengan menggunakan enam *neurons* yang memiliki hubungan terhadap Y. Berikutnya praktikan mencari besaran kesalahan dengan pengukuran MAPE, RMSE, dan MSE.

```
> model15$MSE
[1] 1201371
> rmse_5
[1] 1096.071
> mape_5
[1] 9.963129
```

Gambar 3.16. Nilai *Error Data Testing* 5

```
> model15$fitted
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2007		13985.27	11245.57	13401.06	14496.54	15240.41	15359.06	16381.07	15624.54	15243.97	15747.70	14499.72
2008	15286.23	14854.28	14656.34	16161.21	15846.94	16373.67	17220.18	17885.72	17203.93	16109.30	17191.94	16201.56
2009	15451.35	14459.68	14107.20	17065.68	16871.06	17879.78	18331.88	18479.40	17486.30	17435.48	17172.65	16922.04
2010	17620.50	17386.47	15574.25	16970.72	16965.40	17077.23	17385.04	17686.92	16680.26	17362.37	16888.95	16645.17
2011	17812.44	16803.76	15222.08	16937.32	16525.09	17595.61	17394.71	18047.04	15178.22	16904.42	16428.16	16422.70
2012	16891.53	16158.51	15812.62	17067.44	16817.73	17818.98	18296.66	18101.22	17468.17	16427.27	17072.79	15949.04
2013	16167.39	14928.65	14798.87	15814.92	16102.71	16208.70	17409.09	20156.97	19537.24	19933.13	20415.80	20095.14

```
> pred5 = forecast(model15, 24)
> pred5
```

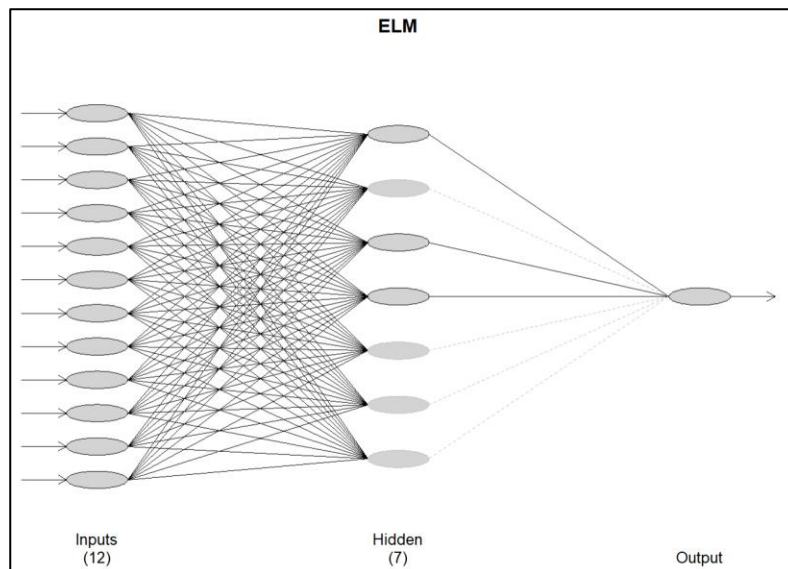
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2014	21393.25	21304.12	21589.18	21821.88	21837.82	21937.17	22267.96	22305.76	22340.90	22586.63	22504.54	22671.91
2015	22746.23	22778.52	22849.85	22970.99	23057.55	23127.36	23247.13	23331.14	23455.54	23522.04	23535.50	23685.01

Gambar 3.17. *Fitted Value* dan Peramalan Model 5

Untuk selanjutnya peramalan dan *fitted value* didapatkan hasil pada prediksi untuk 24 bulan kedepan atau 2 tahun (2014 dan 2015). Pada peramalan tahun 2014 pada awal bulan sampai Maret mengalami fluktuasi, namun bulan selanjutnya mengalami peningkatan signifikan sampai bulan Desember. Kemudian tahun 2015 terjadi kenaikan sampai akhir tahun.

3.2.6 Model Keenam *Hidden Layer 7*

Model selanjutnya dengan *hidden layer* sebesar 7 *neurons*. Dari hasil di bawah ini **Gambar 3.18** menggunakan tujuh *neurons* didapatkan 3 *neurons* atau ada 3 *beta* yang memiliki pengaruh terhadap Y. Kemudian model keenam diperoleh nilai *error* dari RMSE, MSE, dan MAPE pada **Gambar 3.19**.



Gambar 3.18. Plot ELM Model 6

```
> model6$MSE
[1] 1120049
> rmse_6
[1] 1058.324
> mape_6
[1] 10.73255
```

Gambar 3.19. Nilai *Error Data Testing* 6

Selanjutnya telah diperoleh peramalan dan *fitted value* dengan menggunakan perintah sintaks `model6$fitted` dan sintaks peramalan yang tersimpan dalam objek `pred6 = forecast (model6, 24)`, gambar di bawah ini hasilnya.

```
> model6$fitted
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2007	13888.19	11362.84	13379.73	14430.10	15314.60	15335.07	16341.06	15628.97	15269.50	15691.48	14529.36	
2008	15419.54	14561.72	14673.60	16167.55	15829.74	16399.26	17276.57	17757.41	17241.21	16170.34	17112.79	16195.34
2009	15505.22	14337.19	14110.16	17052.84	16855.78	17925.60	18320.57	18546.94	17616.66	17472.77	17246.27	16988.00
2010	17573.77	17206.71	15684.83	16876.82	16942.45	17182.60	17353.82	17646.36	16674.12	17372.99	16712.64	16758.53
2011	17837.69	16600.32	15341.54	16885.83	16507.00	17678.52	17394.50	17973.89	15209.25	16901.83	16334.66	16445.29
2012	16856.63	15990.44	15931.05	16986.01	16844.75	17834.70	18368.10	17872.15	17535.52	16448.48	17056.38	16044.34
2013	16185.27	14797.47	14868.09	15777.27	16096.37	16272.01	17446.19	20048.92	19539.48	19925.39	20353.06	20189.92

```
> pred6 = forecast(model6, 24)
> pred6
```

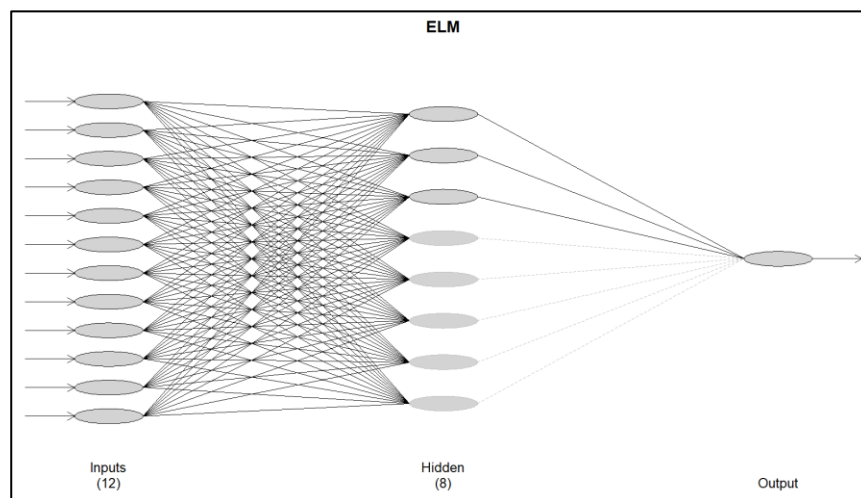
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2014	21368.88	21266.04	21488.59	21569.52	21615.51	21524.30	22058.73	21878.63	22040.59	22328.59	22147.38	22391.57
2015	22615.26	22655.19	22672.13	22847.02	22928.80	22948.09	23096.67	23139.84	23232.21	23353.75	23428.45	23521.16

Gambar 3.20. Fitted Value dan Peramalan Model 6

Diperoleh bahwa pada tahun 2014 bulan Januari sampai Desember mengalami fluktuasi. Tahun berikutnya (2015) nilai peramalan mengalami peningkatan sampai akhir tahun 2015.

3.2.7 Model Ketujuh *Hidden Layer* 8

Kemudian model ketujuh dengan jumlah 8 *neurons* diperoleh bahwa terdapat empat *neurons*, berarti ada 3 *beta* yang memiliki hubungan pada Y. Telah diketahui grafik dari model ketujuh, maka dilakukan perhitungan untuk melihat nilai *error* pada model yang dibentuk (**Gambar 3.22**).



Gambar 3.21. Plot ELM Model 7

```
> rmse_7 = sqrt(model7$MSE)
> mape_7 = mean(abs(error_7)/test)*100
> model7$MSE
[1] 1115716
> rmse_7
[1] 1056.275
> mape_7
[1] 9.404988
```

Gambar 3.22. Nilai Error Data Testing 7

Setelah mendapat nilai *error* pada pembentukan model, praktikan dapat mengetahui nilai peramalan dan *fitted value* yang diperoleh sebagai berikut. Lalu peramalan pada bulan 2014 diperoleh nilai peramalan mengalami penurunan walaupun tidak signifikan. Tahun 2015 peramalan mengalami peningkatan secara signifikan sampai akhir tahun.

```
> model7$fitted
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2007	13953.29	11428.31	13423.32	14414.85	15325.96	15397.76	16261.82	15604.83	15211.80	15650.64	14536.52	
2008	15457.60	14639.51	14717.14	16226.64	15800.84	16423.40	17200.75	17770.01	17247.53	16158.73	17038.99	16229.35
2009	15580.06	14415.76	14188.79	17182.84	16869.15	17907.50	18267.09	18453.04	17560.58	17412.68	17189.14	16945.26
2010	17672.61	17342.44	15881.83	16952.47	16921.47	17186.04	17356.90	17582.34	16731.49	17386.01	16839.42	16674.80
2011	17822.84	16679.58	15341.03	16939.88	16530.84	17681.63	17395.03	17948.50	15308.79	16859.59	16390.59	16515.38
2012	16905.89	16024.08	16013.38	17106.53	16838.99	17860.84	18304.28	17902.73	17503.04	16411.13	16997.89	16068.53
2013	16194.66	14844.74	14983.81	15845.15	16096.47	16278.58	17358.13	20081.30	19557.02	19866.54	20258.66	20124.92

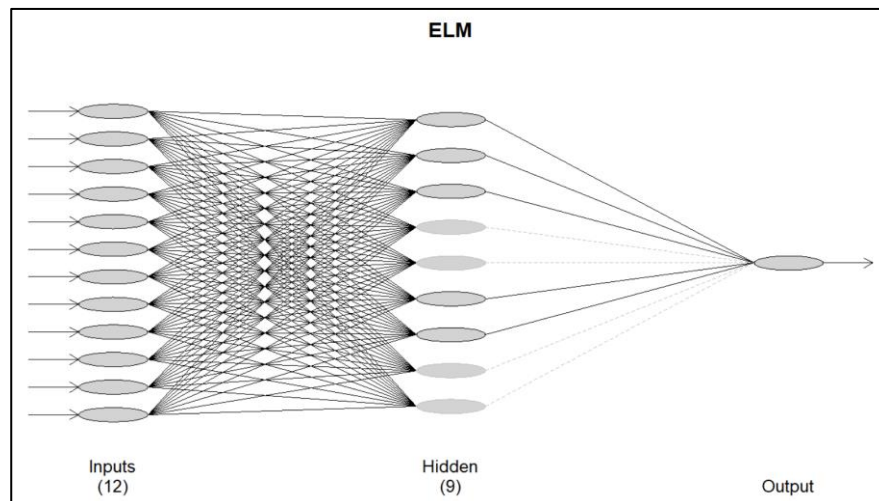
```
> pred7 = forecast(model7, 24)
> pred7
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2014	21495.70	21388.82	21585.76	21780.41	21866.22	21973.08	22259.32	22295.42	22406.63	22602.61	22533.53	22780.04
2015	22916.73	23038.37	23160.11	23282.20	23388.64	23505.42	23624.46	23723.15	23830.60	23937.55	24036.22	24141.66

Gambar 3.23. *Fitted Value* dan Peramalan Model 7

3.2.8 Model Kedelapan *Hidden Layer* 9

Model selanjutnya dengan *hidden layer* 9 *neurons*, kemudian didapatkan grafik model kedelapan didapatkan dua *neurons*, artinya 2 *beta* memiliki pengaruh terhadap Y.



Gambar 3.24. Plot ELM Model 8

Berikutnya diperoleh nilai pengukuran kesalahan pada model kedelapan dengan MSE, RMSE, dan MAPE.

```
> rmse_8 = sqrt(model8$MSE)
> mape_8 = mean(abs(error_8)/test)*100
> model8$MSE
[1] 1037654
> rmse_8
[1] 1018.653
> mape_8
[1] 11.24389
```

Gambar 3.25. Nilai *Error Data Testing* 8

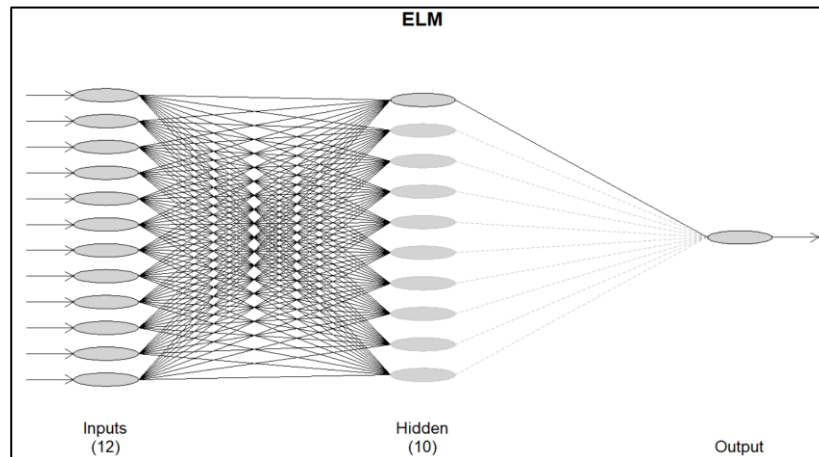
```
> model8$fitted
      Jan      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep      Oct      Nov      Dec
2007    13776.33 11528.02 13352.94 14392.20 15263.47 15463.53 16195.37 15755.89 15301.11 15638.14 14456.77
2008 15398.27 14489.24 14819.28 16210.15 15876.45 16426.50 17276.88 17767.52 17266.14 16152.18 17034.16 16254.45
2009 15550.26 14208.04 14201.66 17126.42 16864.95 17934.68 18292.92 18454.78 17546.61 17463.69 17099.82 16867.26
2010 17496.20 17153.12 15894.67 16913.12 17012.95 17231.67 17375.71 17560.67 16826.87 17363.26 16793.64 16745.97
2011 17832.07 16503.19 15438.73 16865.43 16538.55 17703.65 17389.08 17915.66 15390.86 16796.86 16342.44 16564.57
2012 16800.53 15843.32 16106.50 16909.68 17000.65 17843.12 18367.65 17759.12 17626.49 16410.52 16998.73 16033.96
2013 16118.75 14734.96 14996.65 15795.63 16169.85 16289.22 17400.87 20012.04 19670.69 19929.15 20233.51 20178.80
> pred8 = forecast(model8, 24)
> pred8
      Jan      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep      Oct      Nov      Dec
2014 21341.67 20973.51 21288.24 21398.55 21375.17 21501.39 21855.81 21742.68 21864.41 22121.45 21950.70 22361.34
2015 22387.97 22395.23 22553.52 22574.01 22696.61 22741.36 22896.42 22992.96 23125.41 23243.19 23341.49 23631.78
```

Gambar 3.26. *Fitted Value* dan Peramalan Model 8

Nilai peramalan dengan model kedelapan menggunakan 9 *neurons*, diperoleh pada tahun 2014 peramalan mengalami naik turun walaupun pada bulan Juni sampai Desember mengalami kenaikan secara signifikan. Kemudian 2015, mengalami peningkatan sampai akhir tahun.

3.2.9 Model Kesembilan *Hidden Layer* 10

Terakhir pembentukan model kesembilan dengan *hidden layer* sebanyak 10 *neurons*. Setelah praktikan *running* sintaks pada **Gambar 2.11** didapatkan hasil **Gambar 3.27** artinya dengan menggunakan *hd (hidden layer)* 10 hanya terdapat *beta* sebanyak 6 yang memiliki pengaruh terhadap *Y*.



Gambar 3.27. Plot ELM Model 9

Kemudian diperoleh pula nilai pengukuran kesalahan peramalan dengan model kesembilan.

```
> rmse_9 = sqrt(model9$MSE)
> mape_9 = mean(abs(error_9)/test)*100
> model9$MSE
[1] 1037129
> rmse_9
[1] 1018.395
> mape_9
[1] 10.94333
```

Gambar 3.28. Nilai *Error Data Testing* 9

Lalu untuk nilai *fitted value* dan peramalan dengan menggunakan *hidden layer* 10 *neurons*, maka diperoleh gambar di bawah ini.

```
> model9$fitted
      Jan      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep      Oct      Nov      Dec
2007 13891.23 11524.82 13253.24 14499.44 15204.33 15533.48 16160.92 15549.10 15324.99 15575.10 14564.68
2008 15375.95 14577.34 14773.06 16151.54 15876.86 16276.26 17256.98 17692.50 17187.05 16225.11 16981.13 16333.39
2009 15509.24 14476.07 14366.76 17077.43 16913.14 17847.94 18371.05 18410.67 17466.77 17502.67 17129.18 17010.06
2010 17592.56 17252.84 15984.89 16806.92 17047.44 17131.55 17394.04 17575.10 16734.67 17343.90 16756.80 16808.98
2011 17863.88 16617.86 15434.70 16853.85 16492.25 17613.86 17416.68 17883.61 15457.31 16740.70 16309.37 16608.81
2012 16983.98 15986.75 16018.00 16937.59 16816.67 17795.59 18394.21 17806.99 17592.73 16420.97 16992.85 16151.48
2013 16121.34 14910.26 15011.18 15818.66 16126.78 16231.60 17406.02 19955.46 19459.93 19833.42 20218.38 20187.00
> pred9 = forecast(model9, 24)
> pred9
      Jan      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep      Oct      Nov      Dec
2014 21235.98 21018.37 21289.80 21631.43 21564.65 21565.38 22050.26 21947.10 22110.92 22332.65 22210.00 22419.83
2015 22533.45 22559.56 22543.83 22738.31 22786.30 22773.85 22964.62 22966.58 23028.57 23155.78 23165.23 23231.18
```

Gambar 3.29. *Fitted Value* dan Peramalan Model 9

Dari hasil peramalan **Gambar 3.29** diketahui pada tahun 2014 mengalami naik turun (fluktuasi) sampai akhir tahun. Untuk tahun 2015, mengalami kenaikan nilai peramalan walaupun pada bulan Februari dan Maret sempat turun.

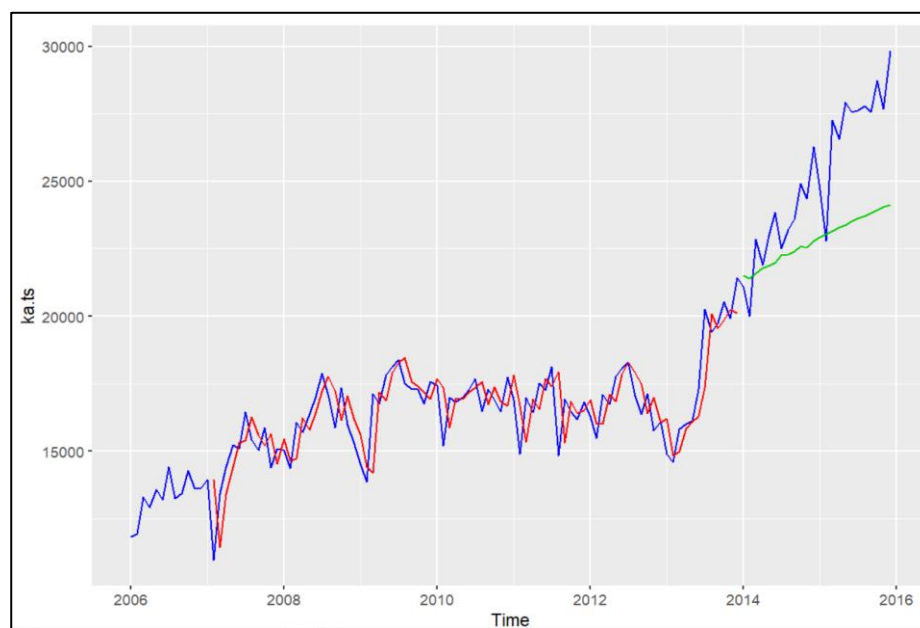
3.3 Evaluasi Kinerja Model

Setelah praktikan membentuk model-model dengan *hidden layer* yang ditentukan dimana jumlah *neuron* 2 sampai 10 *neurons*. Berikut di bawah ini hasil rangkuman namun untuk pembandingan hanya digunakan awal bulan dan tahun yang dipakai 2014 dan 2015 sesuai banyaknya peramalan yaitu 24 bulan atau dua tahun mendatang.

Tabel 3.1. Pemilihan Model Terbaik

Model	Tahun	Aktual	Forecasting	MSE	RMSE	MAPE
1	2014	21092	21506.84	1384160	1176.503	10.42104
	2015	24676	22584.96			
2	2014	21092	21506.84	1384160	1176.503	10.42104
	2015	24676	22584.96			
3	2014	21092	21504.34	1352163	1162.825	10.4165
	2015	24676	22584.96			
4	2014	21092	21506.84	1384160	1176.503	10.42104
	2015	24676	22584.96			
5	2014	21092	21393.25	1201371	1096.071	9.963129
	2015	24676	22746.23			
6	2014	21092	21368.88	1120049	1058.324	10.73255
	2015	24676	22615.26			
7	2014	21092	21495.70	1115716	1056.275	9.404988
	2015	24676	22916.73			
8	2014	21092	21341.67	1037654	1018.653	11.24389
	2015	24676	22387.97			
9	2014	21092	21235.98	1037129	1018.395	10.94333
	2015	24676	22533.45			

Dari hasil tabel di atas praktikan dapat menyimpulkan dalam memperoleh model terbaik dilihat dengan nilai *error(s)* terkecil. Sehingga diketahui pada **Tabel 3.1** terdapat model yang memiliki masing-masing keunggulan, maka terbentuk dua model terbaik. Model terbaik pertama, nilai MSE terkecil diperoleh pada Model 9, sebesar 1037129, artinya selisih data peramalan dengan data sebenarnya adalah 1037129 dan pada model 9 diperoleh nilai RMSE terkecil sebesar 1018.395, artinya sebesar 1018.395 (dalam %) nilai yang diprediksi dekat dengan nilai yang diamati atau diobservasi. Kemudian model terbaik kedua, Model 7 dengan MAPE = 9.404988 dimana berarti model tersebut memiliki kinerja sangat akurat karena berada dalam rentang $< 10\%$ dengan selisih rata-rata nilai peramalan dengan nilai sebenarnya adalah 9.40%. Berikut plot dari Model 7.

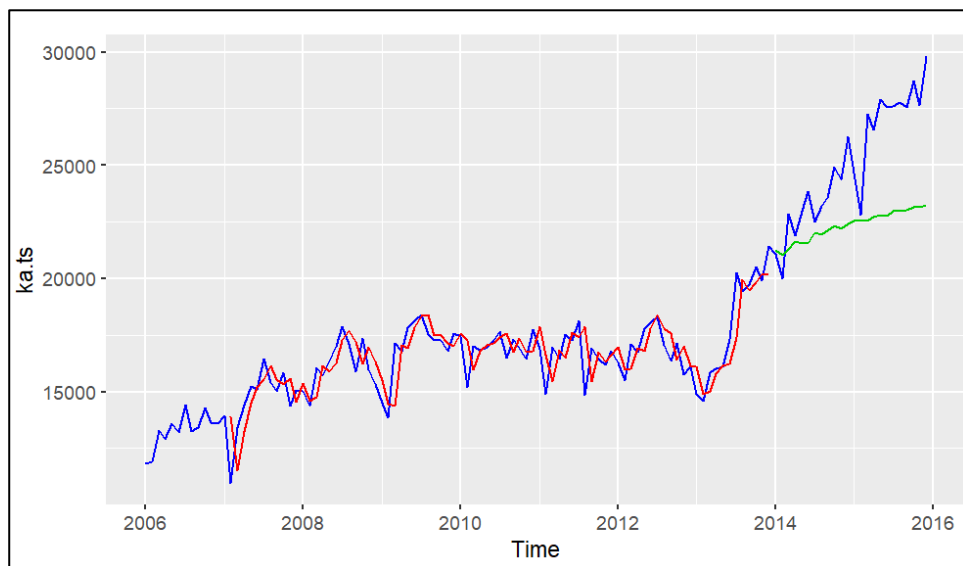


Gambar 3.30. Grafik Model (7) Terbaik

Diperoleh informasi dari grafik model 7 nilai peramalan untuk 24 periode mendatang memiliki pola naik walaupun ada sedikit fluktuasi pada garis peramalan, namun pergerakan garis cenderung mengalami kenaikan. Untuk garis berwarna merah merupakan data *fitted* (kecocokan) pada *line red* terlihat mengikuti pola data aktual sehingga model 7 cocok menjadi model terbaik.

Kemudia Model 9 diperoleh peramalan untuk 24 periode mendatang. Diketahui berdasarkan **Gambar 3.29** peramalan pada tahun 2014 mengalami

kenaikan secara tidak signifikan karena nilai berfluktuasi sampai akhir tahun 2014, dan tahun 2015 juga meningkat walaupun tidak signifikan dan terjadi sampai akhir tahun (Desember 2015), untuk jelasnya berikut grafik model 9 dimana garis biru data aktual, merah untuk *fitted value*, dan hijau nilai peramalan.



Gambar 3.31. Grafik Model (9) Terbaik

4 Penutup

4.1 Kesimpulan

Berdasarkan hasil analisis yang telah dilakukan pada bab sebelumnya dapat disimpulkan bahwa:

1. Sebelum dilakukan pengujian, data yang diperoleh diubah kedalam bentuk objek *time series* dengan perintah dalam R `ts()`. Selain itu, data runtun waktu harus berurutan dengan waktu terlama hingga terbaru.
2. Melakukan *preprocessing* data. Data tersebut akan dibagi dengan komposisi rasio sebesar 80% untuk *data training* dan sisanya 20% sebagai *data testing*. Kemudian membuat model dengan *hidden layer* jumlah *neuron(s)* 2 sampai 10.
3. Telah dibentuk model-model dan dilakukan pengukuran kesalahan data *testing* dari model-model yang dibuat diperoleh model terbaik, Model 7 dan Model 9.
4. Model keduanya memiliki keunggulan masing-masing. Model 7 memiliki nilai MAPE terkecil yaitu 9.404988, dan Model 9 diperoleh nilai MSE dan RMSE terkecil masing-masing 1037129 dan 1018.395.
5. Model 7 hasil peramalan menunjukkan memiliki pola naik walaupun ada sedikit fluktuasi pada garis peramalan, namun pergerakan garis cenderung mengalami kenaikan.
6. Lalu nilai peramalan yang didapat dari model 9 yaitu, data peramalan untuk 24 periode mendatang mengalami fluktuasi dari bulan Januari sampai Desember. Tahun 2015 mengalami kenaikan walaupun pada awal bulan sampai bulan Maret mengalami penurunan bulan Januari sebesar 22533.45 sampai Maret 22543.83.

5 Daftar Pustaka

- Alfiyatin, A. N., Mahmudy, W. F., Ananda, C. F., & Anggodo, Y. P. (2018). Penerapan Extreme Learning Machine (Elm) Untuk Peramalan Laju Inflasi Di Indonesia Implementation Extreme Learning Machine for Inflation Forecasting in Indonesia. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIK)*, 6(2), 179–186. <https://doi.org/10.25126/jtiik.20186900>
- Fikriya, Z. A., Irawan, M. I., & Soetrisno., S. (2017). Implementasi Extreme Learning Machine untuk Pengenalan Objek Citra Digital. *Jurnal Sains Dan Seni ITS*, 6(1). <https://doi.org/10.12962/j23373520.v6i1.21754>
- Zulinda. (2020). *Penerapan Jaringan Syaraf Tiruan Extreme Learning Machinde dan Backpropagation untuk Memprediksi Harga Saham PT BANK MANDIRI (PERSERO) TBK.*