



TUGAS AKHIR - SM091332

**PENERAPAN METODE FILTER KALMAN DALAM  
PERBAIKAN HASIL PREDIKSI CUACA DENGAN  
METODE ARIMA**

TOMY KURNIAWAN  
NRP 1210 100 030

Dosen Pembimbing:  
Drs. Lukman Hanafi, M.Sc  
Dr. Erna Apriliani, M.Si

JURUSAN MATEMATIKA  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Institut Teknologi Sepuluh Nopember  
Surabaya 2014



FINAL PROJECT - SM091332

**THE APPLICATION OF KALMAN FILTER'S  
METHOD IN IMPROVING THE RESULT OF  
WEATHER PREDICTION BY ARIMA'S METHOD**

TOMY KURNIAWAN  
NRP 1210 100 030

Supervisors:  
Drs. Lukman Hanafi, M.Sc  
Dr. Erna Apriliani, M.Si

DEPARTMENT OF MATHEMATICS  
Faculty of Mathematics and Natural Sciences  
Institut Teknologi Sepuluh Nopember  
Surabaya 2014

# PENERAPAN METODE FILTER KALMAN DALAM PERBAIKAN HASIL PREDIKSI CUACA DENGAN METODE ARIMA

Nama Mahasiswa : Tomy Kurniawan  
NRP : 1210 100 030  
Jurusan : Matematika FMIPA-ITS  
Pembimbing : 1. Drs. Lukman Hanafi, M.Sc  
2. Dr. Erna Apriliani, M.Si

## Abstrak

*Time Series merupakan salah satu metode analisis data. Salah satu metodenya adalah ARIMA Box-jenkins yang nantinya digunakan untuk meramalkan data selanjutnya. Penelitian ini didasarkan pada pengamatan dan sesuai hasil model peramalan Analisis time series dari dua parameter meteorologi yaitu suhu udara dan kecepatan angin. Estimasi koefisien pada persamaan model akan dilakukan dengan menggunakan algoritma Filter Kalman. Setelah adanya penggunaan algoritma Filter Kalman akan dilakukan suatu pendekatan yang didasarkan pada koreksi linear dari bias prakiraan dalam penggunaan Filter Kalman. Selanjutnya akan lebih difokuskan pada studi parameter meteorologi satu waktu dimana diberikan  $m_i$  direct output dari model pada waktu  $t_i$  yang mengacu pada salah satu parameter (suhu udara atau kecepatan angin) serta  $y_i^0$  sebagai bias dari prakiraan. Estimasi ini memungkinkan dalam bentuk linier pada polinomial dengan  $n = 2$ . Simulasi ini menggunakan software minitab16, matlab R2010a, dan microsoft excel. Hasil simulasi berupa grafik hasil estimasi state dalam real, ARIMA dengan minitab, dan Filter Kalman.*

*Kata-kunci: ARIMA Box-Jenkins, Filter Kalman*

# THE APPLICATION OF KALMAN FILTER'S METHOD IN IMPROVING THE RESULT OF WEATHER PREDICTION BY ARIMA'S METHOD

Name : Tomy Kurniawan  
NRP : 1210 100 030  
Department : Mathematics FMIPA-ITS  
Supervisors : 1. Drs. Lukman Hanafi, M.Sc  
2. Dr. Erna Apriliani, M.Si

## Abstract

*Time Series is a method for data's analisis. One of them is ARIMA Box-Jenkins which use for predict the next data. This research is based on observation and according to forecast model's time series of two meteorologi's parameters which are temperature air and wind speed. Estimation of coefficient in model's equation by use it Kalman Filter's algorithm. After use Kalman Filter's algorithm will do an aproximation based on linear's correction from bias prediction in using Kalman Filter. After that, will be foccus on study only one meteorologi's parameter who let  $m_i$  as direct output from model in time  $t_i$  which according to one of parameter (temperature air or wind speed) with  $y_i^0$  as bias of prediction. This estimation only can do in linear's form by a polinomial with  $n = 2$ . In this simulation use software minitab16, matlab R2010a, and Microsoft Excel. The result of simulation is graphic of estimation state's result in real, ARIMA by minitab, and Kalman Filter.*

**Key-words:** ARIMA Box-Jenkins, Kalman Filter

# LEMBAR PENGESAHAN

## PENERAPAN METODE FILTER KALMAN DALAM PERBAIKAN HASIL PREDIKSI CUACA DENGAN METODE ARIMA

### *THE APPLICATION KALMAN FILTER'S METHOD IN IMPROVING THE RESULT OF WEATHER PREDICTION BY ARIMA'S METHOD*

#### TUGAS AKHIR

Diajukan untuk memenuhi salah satu syarat  
Untuk memperoleh gelar Sarjana Sains  
Pada bidang studi Permodelan  
Program Studi S-1 Jurusan Matematika  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Institut Teknologi Sepuluh Nopember Surabaya

Oleh :  
TOMY KURNIAWAN  
NRP. 1210 100 030

Menyetujui,

Dosen Pembimbing II,



Dr. Erna Apriliani, M.Si

NIP. 19660414 199102 2 001

Dosen Pembimbing I,



Drs. Lukman Hanafi, M.Sc

NIP. 19640624 198803 1 001

Mengetahui,

Jurusan Matematika  
IPA ITS



Dr. Erna Apriliani, M.Si

NIP. 19660414 199102 2 001

Surabaya, Februari 2014

## KATA PENGANTAR

*Assalamu'alaikum Warokhmatullahi Wabarokatuh*

*Rasa syukur Alhamdulillah* kepada Allah Subhanahu wa Ta'ala atas limpahan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan tugas akhir yang berjudul:

### **”PENERAPAN METODE FILTER KALMAN DALAM PERBAIKAN HASIL PREDIKSI CUACA DENGAN METODE ARIMA”**

sebagai salah satu syarat kelulusan Program Studi S1 Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

Penulis menyadari bahwa dalam penyusunan tugas akhir ini tidak lepas dari bantuan dan dukungan banyak pihak. Oleh karena itu, penulis mengucapkan terima kasih sebesar-besarnya kepada:

1. Drs. Lukman Hanafi, M.Sc selaku dosen pembimbing I yang telah mengarahkan dan membimbing penulis serta mendoakan yang terbaik.
2. Dr. Erna Apriliani, M.Si selaku Ketua Jurusan Matematika ITS sekaligus dosen pembimbing yang tidak bosan membantu ditengah kesibukan waktu beliau.
3. Prof. Dr. Basuki Widodo, M.Sc dan Drs. Herisman, M.Sc selaku dosen penguji tugas akhir yang telah memberikan kritik, saran dan sebagainya yang membangun Tugas Akhir ini.

4. Ibu Soleha, S.Si, M.Si selaku koordinator tugas akhir.
5. Dr. Darmaji, S.Si, MT selaku dosen wali.
6. Bapak, Ibu dosen, dan Kayawan Jurusan Matematika ITS serta mas, mbak, teman-teman yang baik secara langsung maupun tidak langsung turut membantu dalam menyelesaikan Tugas Akhir ini.

Penulis menyadari bahwa dalam tugas akhir ini masih banyak terdapat kekurangan. Oleh karena itu, kritik dan saran sangat penulis harapkan untuk kesempurnaan tugas akhir ini. Akhirnya, penulis berharap semoga tugas akhir ini dapat bermanfaat bagi semua pihak di kemudian hari.

Wassalamu'alaikum Warokhmatullahi Wabarokatuh

Surabaya, Februari 2014

Penulis

*Spesial thank's to:*

1. Bapak dan ibu yang tidak pernah berhenti memberikan dukungan baik moral, material, maupun spiritual sampai saat ini, dan entah sampai kapan. Semoga kelak saya bisa membalasnya. Aamiin.
2. Kedua kakakku yang selalu saja bisa menghiburku dengan tingkah polanya.
3. Mas Wawan, *matur nuwun sanget* telah mendukung hingga asistensi program Matlab sampai tengah malam.
4. Mas-mas dan mbak-mbak, mas fahim, mas ipin, mbak evita, ibu Hani yang telah membantu, terima kasih.
5. Mas Ali selaku asisten Ibu Soleha yang tak kenal lelah menghadapi saya dan teman-teman.
6. Teman-teman seperjuangan Ema, Agis, Mbak Ngatini, Mimi, Heni, Cahyo, Akil, Halwa, Irma, Yusti, Nanda, Selvi, Mevita.
7. Serta Kawan-kawan TeAm-B dan Aritsmatics yang turut *men-support*.

Semoga Allah SWT membalas setiap kebaikan yang telah *sampeyan* berikan kepada saya, serta mengampuni segala dosa kita semua. Aamiin.



## DAFTAR ISI

HALAMAN JUDUL	i
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xv
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
BAB I    PENDAHULUAN	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Masalah dan Asumsi .....	3
1.4 Tujuan .....	4
1.5 Manfaat .....	4
1.6 Sistematika Penulisan .....	4
BAB II    TINJAUAN PUSTAKA	7
2.1 <b>Time Series</b> .....	7
2.1.1 <b>Kestasioneran Data</b> .....	7
2.1.2 <b>Model pada <i>Time Series</i></b> .....	9
2.1.3 <b>Prosedur ARIMA Box-Jenkins</b> ...	11
2.2 Model Matematika .....	16
2.3 Metode Filter Kalman .....	17
BAB III    METODE PENELITIAN	23
3.1 Studi Literatur .....	23

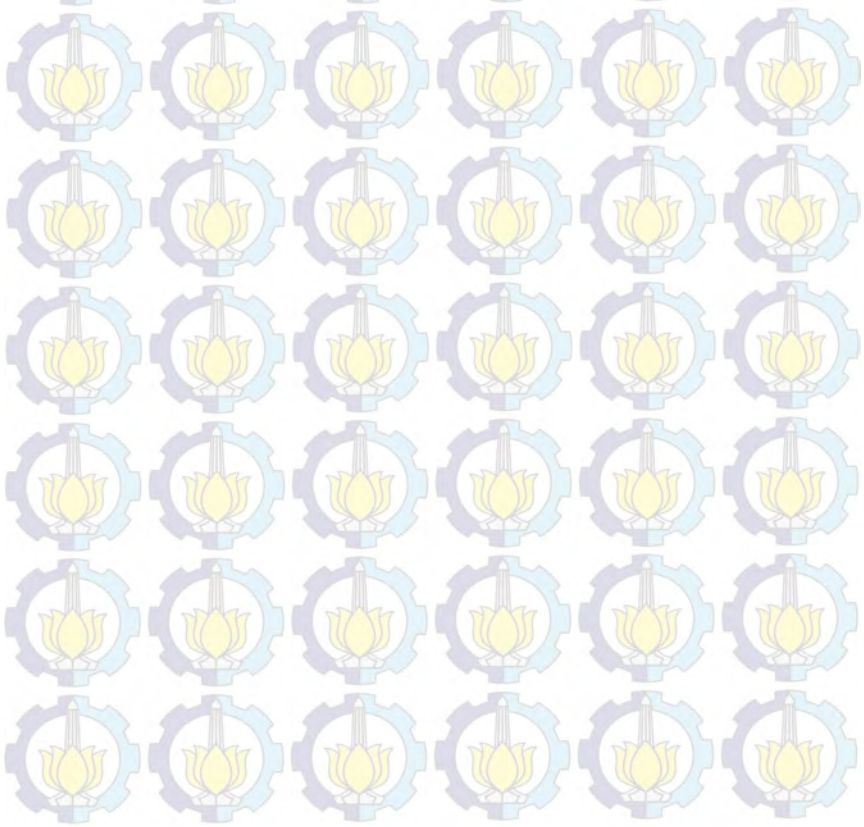
3.2	Analisa Model Data dan Peramalan dengan Time Series ARIMA Box-Jenkins .....	23
3.3	Penerapan Metode Filter Kalman dan Simulasi dengan Software Matlab .....	24
3.4	Penarikan Simpulan .....	24
BAB IV	ANALISIS DAN PEMBAHASAN	25
4.1	Analisis Model Data dan Peramalan .....	25
4.1.1	Analisis Model Data dan Peramalan pada Data Suhu Udara .....	25
4.1.2	Analisis Model Data dan Peramalan pada Data Kecepatan Angin .....	38
4.2	Penerapan Metode Filter Kalman dan Simulasi dengan Software Matlab .....	51
4.3	Simulasi dan Analisis Simulasi .....	53
4.3.1	Simulasi I pada Data Suhu Udara dan Kecepatan Angin .....	53
4.3.2	<b>Simulasi II pada Data Suhu Udara dan Kecepatan Angin .....</b>	<b>58</b>
4.3.3	<b>Simulasi III pada Data Suhu Udara dan Kecepatan Angin .....</b>	<b>63</b>
BAB V	PENUTUP	71
5.1	Kesimpulan .....	71
5.2	Saran .....	71
	DAFTAR PUSTAKA	73
	LAMPIRAN	75
I	Listing Program KF pada Suhu Udara	77
II	Listing Program KF pada Kecepatan Angin	81
III	Listing Program Plot Absolute Error pada Data Suhu Udara	85

IV	Listing Program Plot Absolute Error pada Data Kecepatan Angin	87
----	------------------------------------------------------------------	----

## DAFTAR GAMBAR

Gambar 4.1	<i>Hasil Plot Box-Cox Data Asli</i> .....	27
Gambar 4.2	<i>Hasil Plot Box-Cox Data Transformasi</i>	28
Gambar 4.3	<i>Plot Data Hasil Transformasi</i> .....	29
Gambar 4.4	(a) <i>Hasil Differences</i> dan (b) <i>Plot Hasil Differences</i> .....	30
Gambar 4.5	<i>Hasil Plot ACF (Autocorrelation Function)</i> .....	31
Gambar 4.6	<i>Hasil Plot PACF (Partial Autocorrelation Function)</i> .....	32
Gambar 4.7	<i>Hasil Model ARIMA(1,1,1)</i> .....	34
Gambar 4.8	<i>Plot Kenormalan ARIMA(1,1,1)</i> .....	35
Gambar 4.9	<i>Plot Peramalan ARIMA(1,1,1)</i> .....	37
Gambar 4.10	<i>Hasil Plot Box-Cox Data Asli</i> .....	40
Gambar 4.11	<i>Hasil Plot Box-Cox Data Transformasi</i>	41
Gambar 4.12	<i>Plot Data Hasil Transformasi</i> .....	42
Gambar 4.13	(a) <i>Hasil Differences</i> dan (b) <i>Plot Hasil Differences</i> .....	43
Gambar 4.14	<i>Hasil Plot ACF (Autocorrelation Function)</i> .....	44
Gambar 4.15	<i>Hasil Plot PACF (Partial Autocorrelation Function)</i> .....	45
Gambar 4.16	<i>Hasil Model ARIMA(0,1,1)</i> .....	47
Gambar 4.17	<i>Plot Kenormalan ARIMA(0,1,1)</i> .....	48
Gambar 4.18	<i>Plot Peramalan ARIMA(0,1,1)</i> .....	50
Gambar 4.19	<i>Hasil Simulasi I pada Data Suhu Udara</i>	54
Gambar 4.20	<i>Hasil Simulasi I pada Data Kecepatan Angin</i> .....	56
Gambar 4.21	<i>Hasil Simulasi II pada Data Suhu Udara</i>	59

Gambar 4.22 Hasil Simulasi II pada Data Kecepatan Angin .....	61
Gambar 4.23 Hasil Simulasi III pada Data Suhu Udara .....	64
Gambar 4.24 Hasil Simulasi III pada Data Kecepatan Angin .....	66
Gambar 4.25 Nilai Mutlak Kesalahan Peramalan pada Data Suhu Udara .....	69
Gambar 4.26 Nilai Mutlak Kesalahan Peramalan pada Data Kecepatan Angin .....	70



## DAFTAR TABEL

Tabel 4.1	<i>Data Suhu Udara Bulan Januari-April 2013</i> .....	25
Tabel 4.2	<i>Model Sementara Berdasarkan Plot ACF dan PACF</i> .....	33
Tabel 4.3	<i>Data Kecepatan Angin Bulan Januari-April 2013</i> .....	38
Tabel 4.4	<i>Model Sementara Berdasarkan Plot ACF dan PACF</i> .....	46
Tabel 4.5	<i>Nilai Mutlak Kesalahan Prediksi Suhu Udara pada Simulasi I</i> .....	54
Tabel 4.6	<i>Nilai Mutlak Kesalahan Prediksi Kecepatan Angin pada Simulasi I</i> .....	57
Tabel 4.7	<i>Nilai Mutlak Kesalahan Prediksi Suhu Udara pada Simulasi II</i> .....	59
Tabel 4.8	<i>Nilai Mutlak Kesalahan Prediksi Kecepatan Angin pada Simulasi II</i> .....	62
Tabel 4.9	<i>Nilai Mutlak Kesalahan Prediksi Suhu Udara pada Simulasi III</i> .....	64
Tabel 4.10	<i>Nilai Mutlak Kesalahan Prediksi Kecepatan Angin pada Simulasi III</i> .....	66
Tabel 4.11	<i>Nilai MAPE pada Hasil Peramalan Data suhu Udara</i> .....	68
Tabel 4.12	<i>Nilai MAPE pada Hasil Peramalan Data Kecepatan Angin</i> .....	70

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Indonesia sebagai Negara kepulauan memiliki jumlah penduduk 242.968.342 jiwa di tahun 2012 menduduki peringkat ke-empat Negara dengan jumlah penduduk terbanyak di dunia[1]. Dan Surabaya, sebagai salah satu kota di Indonesia yang merupakan ibukota Jawa Timur memiliki jumlah penduduk hampir tiga juta jiwa di tahun 2012 atau menempati posisi tertinggi ke-dua jumlah penduduknya setelah kota Jakarta[2]. Seiring dengan jumlah penduduk yang banyak, maka aktivitas oleh warga Surabaya juga akan semakin padat dan bervariasi. Berbagai macam aktivitas di kehidupan sehari-hari ini baik secara langsung maupun tidak langsung nantinya akan sangat terpengaruhi oleh kondisi cuaca yang terjadi pada hari tersebut. Misalnya pada penerbangan di Bandara Internasional Juanda, Kabupaten Sidoarjo, Jawa Timur, sempat dihentikan sementara waktu akibat cuaca buruk yang terjadi di kawasan Surabaya Selatan[3]. Ataupun dapat mengambil contoh sederhana di lingkungan sekitar seperti pada sektor pertanian, industri kerupuk, atau berbagai sektor lainnya yang mana jalannya berbagai aktivitas tersebut akan dipengaruhi oleh kondisi cuaca yang ada.

Pada umumnya cuaca terjadi karena suhu dan kelembaban yang berbeda antara satu tempat dengan tempat lainnya. Perbedaan ini bisa terjadi karena sudut pemanasan matahari yang berbeda dari satu tempat ke tempat lainnya

karena perbedaan lintang bumi[4]. Dengan kondisi cuaca yang cenderung berbeda-beda antar tempat dan waktu pada suatu daerah, sehingga perlu adanya informasi tentang prakiraan cuaca agar mendukung segala aktivitas yang akan dikerjakan. Telah muncul berbagai pendekatan-pendekatan dalam membuat prakiraan cuaca yang dikembangkan oleh Negara maju yakni suatu metode dengan menginterpretasikan data hasil pengamatan dan data model prakiraan cuaca atau *Numerical Weather Prediction* (NWP). Dalam penggunaan metode ini masih menunjukkan kesalahan yang sistematis karena setiap prakirawan akan menghasilkan interpretasi yang berbeda-beda. Dengan kata lain, metode ini masih bersifat subjektif[5]. Untuk mengurangi pengaruh tersebut, maka digunakan suatu pendekatan yang menggunakan Filter Kalman. Namun dalam hal ini, perbaikan bias prediksi diimplementasikan pada hasil peramalan dari *ARIMA* (*Autoregressive Integrated Moving Average*) *Box-Jenkins* dikarenakan telah adanya data pengukuran dari BMKG stasiun Juanda Surabaya yang juga merupakan fungsi waktu.

Pada tugas akhir ini, data hasil pengamatan oleh Badan Meteorologi Klimatologi dan Geofisika (BMKG) Stasiun Meteorologi Juanda Surabaya pada suhu udara dan kecepatan angin nantinya akan digambarkan pola yang ada dengan menggunakan Analisis Deret Waktu (*Time Series*). *Time series* adalah serangkaian pengamatan terhadap suatu variabel yang diambil dari waktu ke waktu dan dicatat secara berurutan menurut urutan waktu kejadiannya dengan interval waktu yang tetap[6]. Setiap pengamatan dinyatakan sebagai variabel random  $Z_t$  yang diperoleh berdasarkan indeks waktu tertentu ( $t_i$ ) dengan  $i = 1, 2, \dots, n$ , sehingga penulisan data *time Series* adalah  $Z_{t_1}, Z_{t_2}, Z_{t_3}, \dots, Z_{t_n}$ . Model ARIMA yang didapat, akan digunakan untuk melakukan prediksi



dan penggunaan Filter Kalman yang terdiri dari suatu himpunan atas persamaan matematika akan memberikan sebuah solusi komputasi yang efisien dari metode kuadrat terkecil. Sehingga nantinya metode Filter Kalman ini dapat diterapkan dalam perbaikan prediksi cuaca.

## 1.2 Rumusan Masalah

Dari latar belakang diatas diambil rumusan masalah yang akan diangkat pada tugas akhir antara lain:

- a. Bagaimana penerapan *time series* ARIMA pada data pengamatan oleh BMKG.
- b. Bagaimana penerapan metode Filter Kalman dalam perbaikan prediksi cuaca.

## 1.3 Batasan Masalah dan Asumsi

Dalam penelitian tugas akhir ini, permasalahan yang akan dibahas dibatasi ruang lingkup pembahasannya antara lain:

- a. Analisis deret waktu (*time series*) menggunakan *software* minitab16 dengan metode *ARIMA (Autoregressive Integrated Moving Average) Box-Jenkins* pada data hasil pengamatan oleh BMKG pada bulan Januari-April 2013.
- b. Parameter yang diestimasi adalah suhu udara dan kecepatan angin.
- c. Estimasi pada polinomial dengan  $n = 2$ .
- d. Simulasi dengan menggunakan *software* minitab16, Matlab R2010a, dan *Microsoft Excel*.

## 1.4 Tujuan

Tujuan dari tugas akhir ini antara lain:

1. Menganalisa dan mengevaluasi pola dari data yang ada dengan menggunakan *time series* metode *ARIMA Box-Jenkins*.
2. Menganalisa dan mengevaluasi hasil perbaikan prediksi cuaca dengan menggunakan metode Filter Kalman.

## 1.5 Manfaat

Manfaat yang diharapkan dari tugas Akhir ini adalah:

1. Mampu memberikan kontribusi di dunia penelitian dan ilmu pengetahuan mengenai penerapan *time series* metode *ARIMA Box-Jenkins* pada suatu data agar mengetahui pola yang sesuai.
2. Mampu memberikan informasi mengenai penerapan metode Filter Kalman dalam mengestimasi parameter.
3. Sebagai rujukan untuk penelitian selanjutnya yang berkenaan dengan cuaca maupun faktornya.

## 1.6 Sistematika Penulisan

Penulisan laporan tugas akhir ini disusun dalam 5 Bab, yaitu:

### 1. BAB I PENDAHULUAN

Pada bab ini diuraikan beberapa permasalahan mengenai prediksi cuaca. Sehingga dari permasalahan yang ada menjadi latar belakang dilaksanakannya penelitian pada tugas akhir ini yaitu perbaikan prediksi cuaca dengan metode Filter Kalman. Bab ini juga menguraikan rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan.

## 2. BAB II TINJAUAN PUSTAKA

Bab ini menjabarkan tentang teori-teori yang berhubungan dengan model pendekatan permasalahan, metode *time series*, model *ARIMA Box-Jenkins*, dan metode Filter Kalman. Teori-teori tersebut digunakan sebagai acuan dalam mengerjakan tugas akhir.

## 3. BAB III METODE PENELITIAN

Pada bab ini dijelaskan tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir.

## 4. BAB IV ANALISIS DAN PEMBAHASAN

Bab ini membahas proses analisis model dari data dengan menggunakan metode *time series* dan model *ARIMA Box-Jenkins*, yaitu dengan mengevaluasi kemungkinan model terhadap uji *white noise* dan uji *significant*. Selanjutnya menerapkan metode Filter Kalman untuk memperoleh nilai estimasi dari persamaan *state*. Hasil simulasi berupa grafik hasil estimasi *state* dalam *real*, ARIMA, dan Filter Kalman.

## 5. BAB V PENUTUP

Pada bab ini dirumuskan kesimpulan dan saran dari hasil analisis dan pembahasan yang telah dilakukan.

## BAB II TINJAUAN PUSTAKA

### 2.1 Time Series

*Time series* adalah serangkaian pengamatan terhadap suatu variabel yang diambil dari waktu ke waktu dan dicatat secara berurutan menurut urutan waktu kejadiannya dengan interval waktu yang tetap[6]. Setiap pengamatan dinyatakan sebagai variabel random  $Z_t$  yang diperoleh berdasarkan indeks waktu tertentu ( $t_i$ ) dengan  $i = 1, 2, \dots, n$ , sehingga penulisan data *time Series* adalah  $Z_{t_1}, Z_{t_2}, Z_{t_3}, \dots, Z_{t_n}$ . Beberapa hal yang perlu diperhatikan di dalam *time series* metode *ARIMA Box-Jenkins*, yaitu kestasioneran data, fungsi autokorelasi dan fungsi autokorelasi parsial.

#### 2.1.1 Kestasioneran Data

Data *Time series* merupakan data yang berurutan menurut waktu. Data yang nantinya dapat digunakan untuk peramalan dengan *Time Series ARIMA Box-Jenkins* adalah data yang stasioner baik dalam *mean* maupun *varians*. Untuk data yang stasioner maka nilai-nilai autokorelasinya akan turun secara cepat menuju nol. Sedangkan jika nilai-nilai autokorelasinya turun secara lambat menuju nol selama beberapa waktu maka data tersebut tidak stasioner[7].

Data *Time series* dikatakan stasioner dalam *varians* jika *varians* dari data tidak dipengaruhi oleh waktu. Kestasioneran data secara *varians* dapat dilihat dari hasil transformasi Box-Cox di mana dikatakan suatu stasioner jika nilai *rounded value*-nya bernilai 1[6]. Data yang tidak

stasioner dalam *varians* perlu dilakukan proses transformasi agar *varians*-nya menjadi konstan. Transformasi berawal dari ide Box-Cox dimana Box dan Cox memperkenalkan suatu bentuk transformasi dengan *power transformation* sebagai berikut[6]:

$$T(Z_t) = \frac{(Z_t^\lambda - 1)}{\lambda}; \lambda \neq 0 \quad (2.1)$$

dengan nilai  $\lambda$  adalah parameter transformasi. Khusus nilai  $\lambda = 0$  dilakukan pendekatan sebagai berikut:

$$\lim_{\lambda \rightarrow 0} T(Z_t) = \lim_{\lambda \rightarrow 0} Z_t^{(\lambda)} = \lim_{\lambda \rightarrow 0} \frac{Z_t^\lambda - 1}{\lambda} = \ln(Z_t) \quad (2.2)$$

Beberapa nilai  $\lambda$  yang umumnya digunakan dan transformasi yang sesuai ditampilkan pada Tabel 2.1.

Tabel 2.1. Transformasi Box-Cox[6]

Nilai estimasi $\lambda$	Transformasi
-1,0	$\frac{1}{Z_t}$
-0,5	$\frac{1}{\sqrt{Z_t}}$
0,0	$\ln Z_t$
0,5	$\sqrt{Z_t}$
1	$Z_t$ (tidak ada transformasi)

Ketidastasioneran dalam *mean* dapat dideteksi dengan plot *time series* dari data. Jika plot *time series* berfluktuasi di sekitar rata-rata yang konstan, maka dapat dikatakan data tersebut telah stasioner dalam *mean*. Ketidastasioneran dalam *mean* dapat diatasi dengan menggunakan *differencing*. Proses *differencing* orde ke-d sebagai berikut[7]:

$$Z_t = (1 - B)^d Z_t \quad (2.3)$$

### 2.1.2 Model pada Time Series

Proses pada *time series* secara umum memiliki beberapa model, diantaranya Model AR (*Autoregressive*), MA (*Moving Average*), Model campuran ARMA (*Autoregressive Moving Average*), ARIMA (*Autoregressive Integrated Moving Average*), dan model SARIMA (*Seasonal Autoregressive Integrated Moving Average*)[7]. Model AR (*Autoregressive*) pada orde p menyatakan bahwa suatu model dimana pengamatan pada waktu ke-t berhubungan linear dengan pengamatan waktu sebelumnya  $t_{-1}, t_{-2}, t_{-p}$ . Bentuk fungsi persamaan untuk model AR pada orde p adalah[6]:

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + a_t \quad (2.4)$$

Model AR pada orde 1 yaitu:  $\phi_1 Z_{t-1} + a_t$

Model AR pada orde 2 yaitu:  $\phi_1 Z_{t-1} + \phi_2 Z_{t-2} + a_t$

Model MA (*Moving Average*) pada orde q menyatakan bahwa suatu model yang merupakan suatu penyimpangan pengamatan masa lalu dengan pengamatan waktu ke-t. Bentuk fungsi persamaan untuk model MA pada orde (q) adalah[6]:

$$Z_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q} \quad (2.5)$$

Model MA pada orde 1 yaitu:  $Z_t = a_t - \theta_1 a_{t-1}$

Model MA pada orde 2 yaitu:  $Z_t = (1 - \theta_1 B - \theta_2 B^2) a_t$

Model ARMA merupakan model gabungan antara model AR (*Autoregressive*) dengan MA (*Moving Average*) yang kadang ditulis dengan notasi ARMA (p,q). Bentuk

fungsi model ARMA pada orde p dan q adalah[6]:

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q} \quad (2.6)$$

Model ARMA (1,1) pada orde p=1 dan q=1 yaitu:

$$Z_t = \phi_1 Z_{t-1} + a_t - \theta_1 a_{t-1} \quad (2.7)$$

Model ARIMA (p, d, q) yang dikenalkan oleh Box dan Jenkins dengan orde p sebagai operator dari AR, orde d merupakan *differencing*, dan orde q sebagai operator dari MA. Model ini digunakan untuk data *time series* yang telah di *differencing* atau sudah stasioner dalam *mean*, dimana d adalah banyaknya hasil *differencing*, bentuk persamaan untuk model ARIMA adalah[6]:

$$\phi_p(B)(1 - B)^d Z_t = \phi_0 + \phi_q(B)a_t \quad (2.8)$$

Fungsi orde (p) untuk operator dari AR yang telah stasioner:

$$\phi_p(B) = (1 - \phi_1 B - \phi_1 B^2 - \dots - \phi_p B^p) \quad (2.9)$$

Fungsi dari orde (q) untuk operator MA yang telah stasioner:

$$\theta_q(B) = (1 - \theta_1 B - \theta_1 B^2 - \dots - \theta_q B^q) \quad (2.10)$$

dimana:

- $\phi_p$  = orde p pada koefisien komponen AR
- $\theta_q$  = orde q pada koefisien komponen MA
- B = *backshift operator*
- d = *differencing*

### 2.1.3 *Prosedur ARIMA Box-Jenkins*

Prosedur Box-Jenkins digunakan untuk memilih model ARIMA yang sesuai pada data *time series*. Prosedur ini meliputi empat tahapan yaitu identifikasi, penaksiran dan pengujian parameter, pemeriksaan diagnosis pada residual dan tahap terakhir adalah peramalan[9].

#### 1. Identifikasi

Identifikasi model ARIMA dapat dilakukan dengan melihat plot *time series*, plot ACF, dan plot PACF. Secara umum ada beberapa model yang dapat digunakan yaitu model *Autoregressive* (AR), *Moving Average* (MA), *mixed Autoregressive-Moving Average Processes* (ARMA) dan *Autoregressive Integrated Moving Average Processes* (ARIMA).

#### 2. Estimasi dan Pengujian Signifikansi Parameter

Ada beberapa metode yang dapat digunakan untuk mengestimasi parameter, antara lain: metode *Moment*, metode *Least Square*, metode *Maximum Likelihood Estimation*, metode *Unconditional Least Square*, dan metode *Nonlinear Estimation*[6].

Pada bagian ini, hanya akan dibahas mengenai metode *Maximum Likelihood Estimation* (MLE). Metode yang akan dipakai adalah *Conditional Maximum Likelihood Estimation* atau MLE bersyarat. Model stasioner ARMA (p,q) secara umum:

$$\dot{Z}_t = \phi_1 \dot{Z}_{t-1} + \dots + \phi_p \dot{Z}_{t-p} + a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q} \quad (2.11)$$

dengan dan  $\dot{Z}_t = Z_t - \mu$  adalah i.i.d.  $N(0, \sigma_a^2)$  *white noise*.



Fungsi densitas probabilitas dari  $a = (A_1, a_2, \dots, a_n)$  adalah:

$$P(a|\phi, \mu, \theta, \sigma_a^2) = (2\pi\sigma_a^2)^{-\frac{n}{2}} \exp\left[-\frac{1}{2\sigma_a^2} \sum_{i=1}^n a_i^2\right] \quad (2.12)$$

Sehingga Persamaan 2.11 dapat ditulis:

$$a_t = \theta_1 a_{t-1} + \dots + \theta_q a_{t-q} + \dot{Z}_t - \phi_1 \dot{Z}_{t-1} - \dots - \phi_p \dot{Z}_{t-p} \quad (2.13)$$

selanjutnya dapat diturunkan fungsi *Likelihood* dari parameter  $(\phi, \mu, \theta, \sigma_a^2)$ . Diberikan  $Z = (Z_1, Z_2, \dots, Z_n)'$  dan kemudian diasumsikan kondisi awal  $Z_* = (Z_{1-p}, \dots, Z_{-1}, Z_0)'$  dan  $a_* = (a_{1-q}, \dots, a_{-1}, a_0)'$ , fungsi *Conditional Log Likelihood* adalah:

$$\ln L_*(a|\phi, \mu, \theta, \sigma_a^2) = -\frac{n}{2} \ln 2\phi\sigma_a^2 - \frac{S_*(\phi, \mu, \theta)}{2\sigma_a^2} \quad (2.14)$$

dengan

$$S_*(\phi, \mu, \theta) = \sum_{t=1}^n a_t^2(\phi, \mu, \theta|Z_*, a_*, Z) \quad (2.15)$$

adalah fungsi jumlah kuadrat bersyarat.

Banyak alternatif untuk menetapkan kondisi awal  $Z_*$  dan  $a_*$ , berdasarkan asumsi bahwa  $Z_t$  stasioner dan  $a_t$  *white noise*, maka  $Z_t$  yang tidak diketahui dapat diganti dengan *mean sampel*  $\bar{Z}$  dan  $a_t$  yang tidak diketahui dapat diganti dengan nilai harapannya, yaitu nol. Model pada 2.11, dapat juga diasumsikan  $a_p = a_{p-1} = \dots = a_{p+1-q} = 0$  dan menghitung  $a_t$  untuk  $t \geq p+1$  menggunakan  $\dot{Z}_t$ . Fungsi jumlah kuadrat bersyarat pada  $S_*(\phi, \mu, \theta)$  akan menjadi:

$$S_*(\phi, \mu, \theta) = \sum_{t=p+1}^n a_t^2(\phi, \mu, \theta|Z) \quad (2.16)$$

yang lebih sering digunakan pada program komputer.

Setelah mendapatkan estimasi parameter  $\hat{\phi}$ ,  $\hat{\mu}$ , dan estimasi  $\hat{\sigma}_a^2$  dari  $\sigma_a^2$  dapat dihitung dengan:

$$\hat{\sigma}_a^2 = \frac{S_*(\hat{\phi}, \hat{\mu}, \hat{\theta})}{df} \quad (2.17)$$

dimana  $df$  adalah derajat bebas dengan jumlah derajat bebas sama dengan jumlah syarat yang digunakan pada penjumlahan dari  $S_*(\hat{\phi}, \hat{\mu}, \hat{\theta})$  dikurangi jumlah parameter yang diestimasi[6].

Parameter yang telah diestimasi kemudian diuji signifikansinya untuk melihat apakah parameter tersebut layak digunakan atau tidak. Parameter yang tidak layak akan dikeluarkan dari model. Uji signifikansi dilakukan dengan menggunakan hipotesis sebagai berikut:

Hipotesis :

$$H_0 : \phi_p = 0 \text{ atau } \theta_q = 0$$

$$H_1 : \phi_q \neq 0 \text{ atau } \theta_q \neq 0$$

Statistik Uji :

$$t = \frac{\hat{\phi}_p}{SE(\hat{\phi}_p)} \text{ atau } t = \frac{\hat{\phi}_q}{SE(\hat{\phi}_q)}$$

Tolak  $H_0$  jika  $|t_{hitung}| > t_{\alpha/2, n-p}$  atau jika  $p - value < \alpha$

### 3. Uji Asumsi Residual

Dalam menentukan model ARIMA yang terbaik, harus memenuhi dua asumsi *residual* yaitu berdistribusi normal dan *white noise*.

- *White Noise*

*White noise* merupakan proses dimana tidak terdapat korelasi dalam deret *residual* dari suatu distribusi dengan rata-rata konstan  $E(a_t) = \mu_a$ , biasanya diasumsikan sama dengan nol, varians konstan  $Var(a_t) = \sigma_a^2$  dan  $\gamma_k = Cov(a_t, a_{t+k}) = 0$ [6]. Berikut ini merupakan pengujian *white noise* :

Hipotesis :

$H_0 : \rho_1 = \rho_2 = \dots = \rho_k = 0$  (*residual* memenuhi syarat *white noise*)

$H_1 : \text{minimal ada satu } \rho_i \neq 0 \text{ untuk } i = 1, 2, \dots, k$   
(*residual* belum memenuhi syarat *white noise*)

Statistik Uji :

$$Q^* = n(n+2) \sum_{k=1}^K \frac{\hat{\rho}_k^2}{(n-k)}$$

Daerah Penolakan :

Tolak  $H_0$  jika  $Q^* > \chi^2_{\alpha, df=K-p-q}$  atau *p-value*  $< \alpha$

- Distribusi Normal Di bawah ini adalah cara pengujian kenormalan data dengan *Kolmogorov Smirnov Test*[12]:

Hipotesis :

$H_0$  : *Residual* berdistribusi normal

$H_1$  : *Residual* tidak berdistribusi normal

Statistik Uji :

$$D = \sup_x |S(x) - F_0(x)|$$

dengan :

$S(x)$  = fungsi peluang kumulatif yang dihitung dari data sampel

$F_0(x)$  = fungsi peluang kumulatif dari distribusi normal

$\sup$  = nilai supremum untuk semua  $x$  dari selisih mutlak  $S(x)$  dan  $F_0(x)$

Daerah Penolakan :

Tolak  $H_0$  jika  $D > D_{(1-\alpha, n)}$  atau  $p\text{-value} < \alpha$

#### 4. Peramalan

Tahapan terakhir setelah melalui tiga tahapan di atas, adalah peramalan. Dalam praktek, model yang ditemukan bukan model yang sebenarnya, melainkan hanya pendekatannya saja yang selalu mengandung kesalahan, baik dalam langkah identifikasi maupun estimasi. Hasil ramalan dikatakan baik, jika nilai ramalannya dekat data aktual serta memiliki tingkat kesalahan yang paling kecil.

## 2.2 Model Matematika

Pada penelitian ini lebih ingin mengetahui penggunaan fungsi linier dalam algoritma Filter Kalman untuk memperbaiki dalam prakiran cuaca. Penelitian ini didasarkan pada pengamatan dan sesuai hasil model peramalan Analisis deret waktu *time series* dari dua parameter meteorologi yaitu suhu udara dan kecepatan angin. Setelah adanya penggunaan algoritma Filter Kalman akan dilakukan suatu pendekatan yang didasarkan pada koreksi linear dari bias prakiraan dalam penggunaan Filter Kalman. Selanjutnya akan lebih difokuskan pada studi parameter meteorologi satu waktu dimana diberikan  $m_i$  *direct output* dari model pada waktu  $t_i$  yang mengacu pada salah satu parameter suhu udara atau kecepatan angin) serta  $y_i^0$  sebagai bias dari perkiraan. Diberikan polinomial:

$$y_i^0 = a_{0,i} + a_{1,i}.m_i + \dots + a_{n-1,i}.m_i^{n-1} + \epsilon_i \quad (2.18)$$

Dengan koefisien  $(a_{j,i})$ ,  $j = 0, 1, \dots, n-1$  adalah parameter yang harus diestimasi dengan Filter Kalman dan  $\epsilon_i$  merupakan konstanta. Diasumsikan sebagai *state vector* yang dibentuk oleh koefisien  $(a_{j,i})$  yaitu  $x(t_i) = [a_{0,i} \ a_{1,i} \ a_{2,i} \ \dots \ a_{n-1,i}]^T$ , sebagai pengamatan bias  $y_i^0$ , sebagai matriks pengamatan  $H_i = [1 \ m_i \ m_i^2 \ \dots \ m_i^{n-1}]$ , dan sebagai matriks sistem adalah matriks identitas  $I_n$ . Dalam hal ini, persamaan sistem dan pengamatan seperti sebagai berikut:

$$x^t(t_{i+1}) = x^t(t_i) + \eta(t_i) \quad (2.19)$$

$$y_i^0 = H_i[x^t(t_i)] + \epsilon_i \quad (2.20)$$

Estimasi ini memungkinkan dalam bentuk linier pada polinomial dengan  $n = 2$ .

### 2.3 Metode Filter Kalman

Metode Filter Kalman secara umum digunakan untuk mengestimasi variabel state  $x_k \in R^n$  dari sistem dinamik stokastik linear[13]

$$x_{k+1} = A_k x_k + B_k u_k + G_k w_k \quad (2.21)$$

dengan pengukuran  $z_k \in R^p$  yang memenuhi[13]

$$z_k = H_k x_k + v_k \quad (2.22)$$

untuk  $z_k \in R^p$  adalah variabel input yang diberikan secara deterministik. Variabel  $w_k \in R^l$  dan  $v_k \in R^p$  masing-masing menyatakan noise sistem dan noise pengukuran. Kedua variabel ini diasumsikan *white noise*, tidak berkorelasi satu sama lain maupun dengan nilai estimasi awal  $\bar{x}_0$ , dan mempunyai distribusi peluang normal. Jika variansi dari  $w_k$  adalah  $Q_k$  dan variansi dari  $v_k$  adalah  $R_k$ , maka dipenuhi  $\bar{w}_k = 0, \bar{v}_k = 0, \overline{w_k w_k^T} = Q_k$  dan  $\overline{v_k v_k^T} = R_k$ . Dalam hal ini simbol garis di atas (*overbar*) menunjuk pada pengertian nilai ekspektasi atau *mean* dari suatu variabel random. Dengan demikian, dapat dituliskan  $w_k \sim N(0, Q_k)$  dan  $v_k \sim N(0, R_k)$ .

Pada sebelumnya telah diuraikan efek dari sistem dinamik terhadap nilai *mean* dan kovariansi dari  $x_k$  dan  $z_k$ . Untuk *mean* dari state  $x_k$ , dapat dituliskan

$$\begin{aligned} \bar{x}_{k+1} &= \overline{A_k x_k + B_k u_k + G_k w_k} \\ &= A_k \bar{x}_k + B_k \bar{u}_k + G_k \bar{w}_k \end{aligned}$$

Karena noise proses  $w_k$  adalah *white noise*, maka dipenuhi  $\bar{w}_k = 0$ . Sedangkan,  $u_k$  adalah variabel input yang deterministik, sehingga berlaku  $\bar{u}_k = u_k$ . Oleh karena itu, didapat

$$\bar{x}_{k+1} = A_k \bar{x}_k + B_k u_k \quad (2.23)$$

Bentuk  $\bar{x}_{k+1}$  pada Persamaan (2.23) sekaligus juga menggambarkan nilai dari variabel estimasi  $\hat{x}_{k+1}$ .

Untuk menentukan kovariansi dari state  $x_{k+1}$ , dapat dituliskan[13]

$$\begin{aligned} P_{x_{k+1}} &= \overline{(x_{k+1} - \bar{x}_{k+1})(x_{k+1} - \bar{x}_{k+1})^T} \\ &= \overline{[A_k(x_k - \bar{x}_k) + G_k w_k][A_k(x_k - \bar{x}_k) + G_k w_k]^T}. \end{aligned}$$

Bentuk terakhir ini identik dengan

$$\begin{aligned} &A_k \overline{(x_k - \bar{x}_k)(x_k - \bar{x}_k)^T} A_k^T + G_k \overline{w_k(w_k - \bar{w}_k)^T} A_k^T + A_k \overline{(x_k - \bar{x}_k)w_k^T} G_k^T \\ &+ G_k \overline{w_k w_k^T} G_k^T \end{aligned}$$

sehingga diperoleh

$$P_{x_{k+1}} = A_k P_{x_k} A_k^T + G_k P_{x_k w_k} A_k^T + A_k P_{x_k w_k} G_k^T + G_k Q_k G_k^T.$$

Dengan mengingat bahwa  $x_k$  dan  $w_k$  tidak berkorelasi (yaitu berarti  $P_{w_k x_k} = P_{x_k w_k} = 0$ ), maka didapat

$$P_{x_{k+1}} = A_k P_{x_k} A_k^T + G_k Q_k G_k^T. \quad (2.24)$$

Dalam hal ini juga telah disebutkan bahwa nilai kovariansi dari state  $x_k$  sama dengan nilai kovariansi errornya. Dengan demikian, Persamaan (2.24) sekaligus juga menunjukkan nilai dari kovariansi *error state*  $x_k$ .

$$\bar{z}_k = \overline{H_k x_k + v_k} = H_k \bar{x}_k + \bar{v}_k.$$

Karena  $v_k$  adalah *white noise*, maka didapat

$$\bar{z}_k = H_k \bar{x}_k \quad (2.25)$$

Sedangkan kovariansi dari pengukuran

$$\begin{aligned} P_{z_k} &= \overline{(z_k - \bar{z}_k)(z_k - \bar{z}_k)^T} \\ &= \overline{[H_k(x_k - \bar{x}_k) + v_k][H_k(x_k - \bar{x}_k) + v_k]^T} \end{aligned}$$

Dengan mengingat asumsi bahwa  $x_k$  dan  $v_k$  tidak berko-relasi, maka didapat[13]

$$P_{z_k} = H_k P_{x_k} H_k^T + R_k \quad (2.26)$$

Sedangkan, kovariansi antara state  $x_k$  dan output  $z_k$  adalah

$$\begin{aligned} P_{x_k z_k} &= (x_k - \bar{x}_k)(z_k - \bar{z}_k)^T \\ &= \frac{(x_k - \bar{x}_k)[H_k(x_k - \bar{x}_k) + v_k]^T}{\phantom{=}} \end{aligned}$$

yaitu didapatkan

$$P_{x_k z_k} = P_{x_k} + H_k^T \quad (2.27)$$

Dari Persamaan (2.25) - (2.27) dapat disimpulkan bahwa distribusi variabel random gabungan  $x_k$  dan  $z_k$  adalah

$$\begin{bmatrix} x_k \\ z_k \end{bmatrix} \sim \left( \begin{bmatrix} \bar{x}_k \\ \bar{z}_k \end{bmatrix}, \begin{bmatrix} P_{x_k} & P_{x_k z_k} \\ P_{z_k x_k} & P_{z_k} \end{bmatrix} \right)$$

atau

$$\begin{bmatrix} x_k \\ z_k \end{bmatrix} \sim \left( \begin{bmatrix} \bar{x}_k \\ H_k \bar{x}_k \end{bmatrix}, \begin{bmatrix} p_{x_k} & P_{x_k} H_k^T \\ H_k P_{x_k} & H_k P_{x_k} H_k^T + R_k \end{bmatrix} \right) \quad (2.28)$$

Telah diketahui bahwa *estimator linear* terbaik untuk variabel random  $x_k$  jika diberikan  $\bar{x}_k, P_{x_k}$ , dan pengukuran  $z_k$  adalah

$$P_{\hat{x}_k} = (P_{x_k}^{-1} + H_k^T R_k^{-1} H_k)^{-1} \quad (2.29)$$

$$\hat{x}_k = \bar{x}_k + P_{\hat{x}_k} H_k^T R_k^{-1} (z_k - H_k \bar{x}_k) \quad (2.30)$$

Dengan  $\hat{x}_k$  menyatakan estimasi untuk  $x_k$  dan  $\tilde{x}_k = (x_k - \hat{x}_k)$  adalah error estimasinya.



Proses estimasi dengan menggunakan Filter Kalman terbagi dalam dua tahap, yaitu tahap *time update* dan tahap *measurement update*. Tahap *time update* (atau tahap prediksi) dipengaruhi oleh dinamika sistem; sedangkan tahap *measurement update* (atau tahap koreksi) dipengaruhi oleh adanya informasi tambahan berupa pengukuran. Pada tahap *time update* didefinisikan estimasi state  $\hat{x}_k^- \in \mathbb{R}^n$  yang sering disebut variabel *priori state estimate*; sedangkan estimasi state  $\hat{x}_k \in \mathbb{R}^n$  (yang disebut juga *posteriori state estimate* didefinisikan pada tahap *measurement update*. Dengan Persamaan (2.23), (2.24), (2.29), dan (2.30) dapat diturunkan suatu algoritma Filter Kalman untuk mengestimasi variabel state  $\hat{x}_k$ . Untuk *time step*  $k = 0$ , diperlukan nilai state awal  $\bar{x}_k$  dan kovariansi error  $P_{x_k}$  yang menggambarkan tingkat kepercayaan terhadap nilai *estimate state* awal.

Nilai estimasi tahap *measurement update* bergantung pada *residual* atau *measurement innovation*  $\tilde{z}_k = (z_k - H_k \bar{x}_k)$ . koefisien pembobotan dari *residual* tersebut seringkali disebut *Kalman Gain*, yaitu

$$K_k = P_k H_k^T R_k^{-1}$$

Jika terjadi kasus  $\det(P_{k+1}^-) = 0$ , maka nilai  $(P_{k+1}^-)^{-1}$  pada tahap *measurement update* tidak dapat ditentukan. Oleh karena itu bentuk alternatif dari tahap *measurement update* dengan melibatkan pengertian *Kalman Gain*, yaitu

$$K_{k+1} = P_{k+1}^- H_k^T (H_{k+1} P_{k+1}^- H_{k+1}^T + R_{k+1})^{-1} \quad (2.31)$$

$$P_{k+1} = (I - K_{k+1} H_{k+1}) P_{k+1}^- \quad (2.32)$$

$$\hat{x} = \hat{x}_{k+1}^- + K_{k+1} (z_{k+1} - H_{k+1} \hat{x}_{k+1}^-) \quad (2.33)$$

Bentuk alternatif Persamaan (2.31) - (2.33) ini lebih menguntungkan dari segi komputasi karena hanya memuat satu proses menginverskan matriks ukuran  $p \times p$ .

## Algoritma Filter Kalman

---

### 1. Model sistem dan model pengukuran

$$\begin{aligned}
 x_{k+1} &= A_k x_k + B_k u_k + G_k w_k \\
 z_k &= H_k x_k + v_k \\
 x_0 &\sim N(\bar{x}_0, Q_k) \\
 w_k &\sim N(0, Q_k) \\
 v_k &\sim N(0, R_k)
 \end{aligned}$$

### 2. Inisialisasi

$$\begin{aligned}
 P_0 &= P_{x_0} \\
 \hat{x}_0 &= \bar{x}_0
 \end{aligned}$$

### 3. Tahap *time update*:

– Kovariansi error :

$$P_{k+1}^- = A_k P_k A_k^T + G_k Q_k G_k^T$$

– Estimasi :

$$\hat{x}_{k+1}^- = A_k \hat{x}_k + B_k u_k$$

### 4. Tahap *measurement update*:

– Kovariansi error :

$$P_{k+1} = [(P_{k+1}^-)^{-1} + H_{k+1}^T R_{k+1}^{-1} H_{k+1}]^{-1}$$

– Estimasi :

$$\hat{x}_{k+1} = \hat{x}_{k+1}^- + P_{k+1} H_{k+1}^T R_{k+1}^{-1} (z_{k+1} - H_{k+1} \hat{x}_{k+1}^-)$$


---

*"Halaman ini sengaja dikosongkan"*

## BAB III METODE PENELITIAN

Bab ini menjelaskan metode-metode yang diterapkan untuk menyelesaikan permasalahan yang diangkat dalam Tugas Akhir. Dengan mengacu pada tinjauan pustaka yang terdapat pada bab sebelumnya, metode yang dijabarkan disini memperjelas apa saja yang dilakukan dalam menyelesaikan permasalahan tersebut.

### 3.1 Studi Literatur

Pada tahap ini dipaparkan berbagai teori yang menunjang tugas akhir ini. Teori penunjang itu meliputi pembahasan tentang olah data dalam mencari model yang sesuai dengan analisis *time series*. Selain itu juga dipelajari tentang metode estimasi Filter Kalman yang nantinya akan mengestimasi nilai koefisien dari polinomial.

### 3.2 Analisa Model Data dan Peramalan dengan Time Series ARIMA Box-Jenkins

Akan diambil data suhu udara dan kecepatan angin yang didapat dari BMKG stasiun Juanda-Surabaya pada bulan Januari, Februari, Maret, April, Mei, Juni, Juli, dan Agustus di tahun 2013. Dengan menggunakan Minitab16 akan di cek model dan peramalan dari sebagian data yaitu data pada bulan Januari, Februari, Maret, dan April di tahun 2013. Pertama melakukan pengecekan stasioner atau tidak dari data yang ada baik secara *varians* maupun *mean* serta uji normalitasnya. Selanjutnya plot ACF dan PACF pada data untuk menemukan prakiraan model. Lalu dari dugaan model akan diuji *white noise* dan *significant* dari model-model

tersebut. Setelah menemukan bentuk model maka akan dicari persamaan model yang sesuai serta hasil peramalan baik untuk suhu udara maupun kecepatan angin.

### **3.3 Penerapan Metode Filter Kalman dan Simulasi dengan Software Matlab**

Pada tahap ini dilakukan simulasi penerapan algoritma Filter Kalman dengan *Software* Matlab untuk memperoleh hasil estimasi dari *state* dan parameter. Yang selanjutnya akan ditampilkan grafik dari koefisien polinomial dan grafik perbandingan antara data, ARIMA, dan Filter Kalman.

### **3.4 Penarikan Simpulan**

Berdasarkan hasil simulasi dapat diketahui seperti apa bentuk model dari data pengukuran BMKG serta metode Filter Kalman apakah dapat digunakan untuk mengestimasi koefisien dari polinomial serta dapat memperbaiki hasil peramalan dari model ARIMA.

## BAB IV ANALISIS DAN PEMBAHASAN

### 4.1 Analisis Model Data dan Peramalan

#### 4.1.1 Analisis Model Data dan Peramalan pada Data Suhu Udara

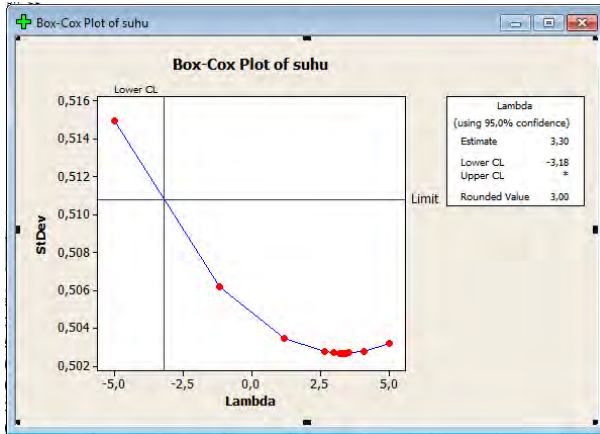
Pada tahapan ini akan dianalisis data yang didapat dari BMKG stasiun Juanda-Surabaya. Data yang didapat merupakan hasil pengamatan dari pihak terkait dengan menggunakan alat ukur suhu udara. Dan dalam hal ini data yang dianalisis adalah data suhu udara pada bulan Januari, Februari, Maret, dan April. Data yang ada akan diolah lebih lanjut dengan menggunakan *software* Minitab16. Dengan menggunakan metode analisis *time series* maka akan dicoba untuk mencari model data yang sesuai. Akan tetapi, kemungkinan model yang ada akan lebih dari satu tergantung dari hasil plot data yang ada. Berbagai kemungkinan model yang ada tersebut akan diuji kelayakannya sesuai dengan aturan-aturan yang ada pada metode analisis *time series* ini. Lalu akan diramalkan sebanyak 120 hari berikutnya.

Tabel 4.1: *Data Suhu Udara Bulan Januari-April 2013*

TANGGAL	JANUARI	FEBRUARI	MARET	APRIL
1	27,2	27,3	28,2	28,9
2	26,9	27	27,7	28,8
3	27,5	26,9	27,9	28,7
4	27,5	29	28,8	28,7
5	27,2	28,8	27	29,3
6	27,5	29,1	28,8	27,7

TANGGAL	JANUARI	FEBRUARI	MARET	APRIL
7	27,6	29,3	28,5	27,1
8	28,6	29,3	27,9	27,3
9	28,1	29,3	27	27,6
10	29,2	28,8	27,3	27,7
11	28,8	28	27	27,6
12	29	28,3	26,4	28,2
13	28,6	27,1	27,1	28,8
14	27	27,6	26,9	29,1
15	26,5	27,4	26,1	28,4
16	27	26,5	25,7	28
17	27,6	26,9	26,8	28,7
18	27,3	27,4	27,6	27,1
19	28,2	27,8	28	27,9
20	26,9	26,9	28,3	28,2
21	27,2	27,5	29,3	28,4
22	27,7	28,8	28,7	29,2
23	28,3	27,5	29	27,9
24	26,7	27,2	28,9	27,9
25	26,9	28	28,7	28,8
26	27,2	28	28,3	28,3
27	27,5	26,5	27,9	28,5
28	27,1	27,1	28,9	29,2
29	27,3		28,7	29,4
30	27,2		27,7	29,4
31	27,8		27,7	

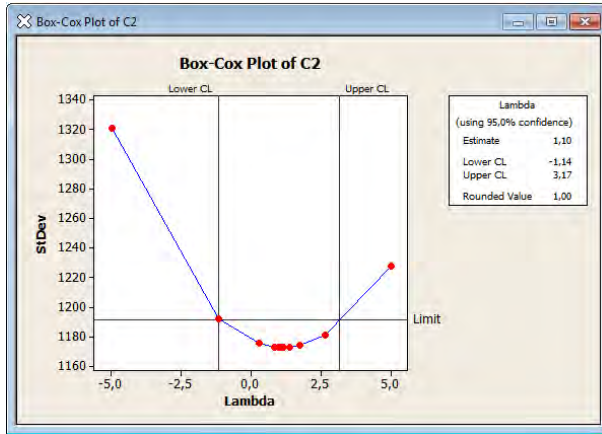
Dari data di atas akan diuji stasioneritasnya dalam *varians* dan *mean*. Syarat terpenting yang harus dipenuhi agar data dapat diolah dengan *time series* adalah stasioner, baik dalam *varians* maupun *mean*[6]. Sebuah deret disebut stasioner jika sifat statistiknya bebas dari waktu periode selama pengamatan[7].



Gambar 4.1: Hasil Plot Box-Cox Data Asli

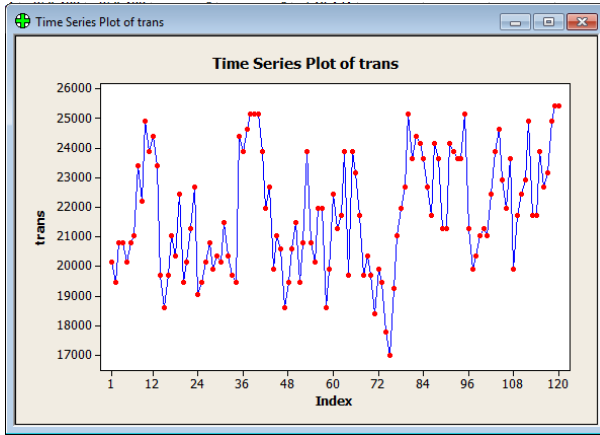
Pada tahapan ini akan dilakukan uji stasioneritas dalam *varians*. Suatu data dapat dikatakan stasioner dalam *varians* jika *varians* dari data tidak dipengaruhi oleh waktu. Kestasioneran data secara *varians* dapat dilihat dari hasil transformasi Box-Cox di mana dikatakan stasioner jika nilai rounded value-nya bernilai 1[6]. Dari data pada Tabel 4.1 akan dilakukan plot Box-Cox untuk mengetahui nilai *varians*-nya. Setelah dilakukan plot Box-Cox seperti pada Gambar 4.1 dapat diamati dalam kotak dialog bahwa nilai *lambda* berada diatas -3,16 dengan *estimate value* = 3,30 dan *rounded value* = 3,00. Hal ini dapat diartikan bahwa data yang didapat dari BMKG stasiun Juanda Surabaya tersebut belum stasioner terhadap *varians*, dikarekan nilai *rounded value*  $\neq 1$  sehingga untuk menstasionerkan suatu data yang tidak stasioner terhadap *varians* dapat dilakukan transformasi pada data yang tidak stasioner tersebut.





Gambar 4.2: Hasil Plot Box-Cox Data Transformasi

Setelah dilakukan transformasi, data hasil transformasi tersebut juga akan dilakukan plot Box-Cox untuk mengetahui kestasioneran dari data hasil transformasi tersebut. Karena pada plot Box-Cox awal didapat nilai *rounded value* = 3,00 maka dengan mengacu pada Tabel 2.1 transformasi yang dilakukan pangkat 3 dari data awal atau  $Z_t^{(\lambda)} = Z_t^{(3)}$ . Dan seperti pada Gambar 4.2 dapat diamati dalam kotak dialog bahwa nilai *lambda* berada diantara -1,14 dan 3,17 dengan *estimate value* = 1,10 dan *rounded value* = 1,00. Hal ini dapat diartikan bahwa data yang didapat dari BMKG stasiun Juanda Surabaya tersebut telah stasioner terhadap *varians*, dikarenakan nilai *rounded value* = 1 sehingga untuk selanjutnya juga akan diuji stasioneritas dala *mean*. Setelah didapat kestasioneran terhadap *varians* maka selanjutnya menguji kestasioneran terhadap *mean*. Dengan menggunakan *Time series* plot, maka kestasioneran data dalam rata-rata (*mean*) akan terlihat. Gambar yang muncul adalah plot series data hasil transformasi seperti berikut ini.

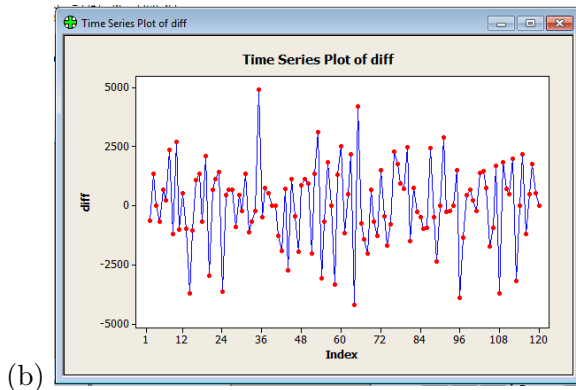


Gambar 4.3: *Plot Data Hasil Transformasi*

Setelah didapat kestasioneran terhadap *varians* maka juga akan diuji kestasioneran terhadap *mean*. Dengan menggunakan *Time series plot*, maka kestasioneran data dalam rata-rata (*mean*) akan terlihat. Gambar yang muncul adalah plot series yang sesuai dengan data hasil transformasi dimana dari tampilan plot tersebut dapat diamati kestasioneran data terhadap rata-rata atau *mean*. Pada Gambar 4.3, diperoleh plot data yang fluktuatif, karena tidak terdapat pola yang teratur maka dapat diartikan bahwa data hasil transformasi tersebut tidak stasioner dalam *mean*. Sehingga agar didapat data yang stasioner terhadap *mean* maka data harus dideferensial terlebih dahulu sampai plot data tidak berpola fluktuatif lagi. Cara mendifferensi (stat-time series-differences). Untuk mengetahui apakah data hasil diferensi tersebut stasioner terhadap *mean* maka kemudian diplot kembali data hasil *differences* tadi untuk melihat bentuk polanya dengan *time series plot*.

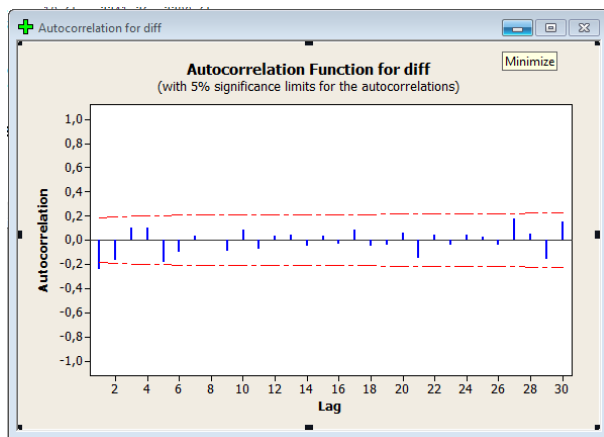
(a)

	C1	C2	C3	C4	C5	C6	C7	C8	C9
	suhu	trans	cek-trans	diff	for(1)			fix_forecast	fix_fix
1	27.2	20123.6	20123.6	*	672.531	672.531	*	*	-526.962
2	26.9	19465.1	19465.1	-658.54	350.122	350.122	*	*	-546.171
3	27.5	20796.9	20796.9	1331.77	217.181	217.181	672.531	672.531	-318.561
4	27.5	20796.9	20796.9	0.00	137.719	137.719	350.122	350.122	-182.511
5	27.2	20123.6	20123.6	-673.23	90.221		217.181	217.181	-101.190
6	27.5	20796.9	20796.9	673.23			137.719	137.719	-62.582
7	27.6	21024.6	21024.6	227.70			90.221	90.221	-23.527
8	28.6	23393.7	23393.7	2369.08			61.831	61.831	-6.160
9	28.1	22188.0	22188.0	1205.61			44.861	44.861	4.221



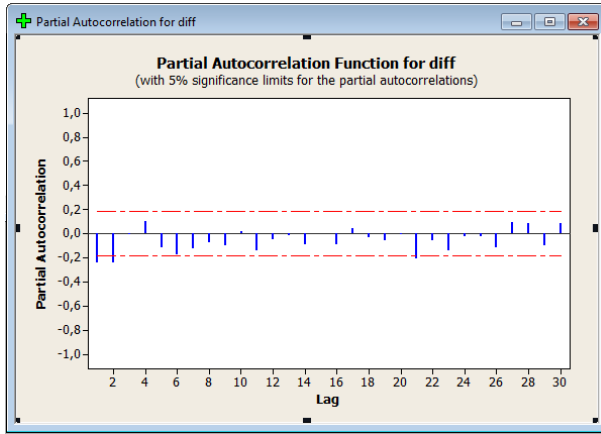
Gambar 4.4: (a) Hasil Differences dan (b) Plot Hasil Differences

Setelah dideferensiasi sekali, data menjadi stasioner dalam *mean* seperti yang terlihat pada Gambar plot 4.4 bagian (b) dimana pola data cenderung tidak fluktuatif lagi. Selain itu dapat kita lihat juga pada plot ACF dari data yang telah di *differencing* pada Gambar 4.5 tampak nilai-nilai autokorelasinya tidak turun secara lambat lagi menuju nol yang artinya bahwa data tersebut telah stasioner dalam *mean*. Karena data sudah stationer dalam *mean* dan *varians* maka disimpulkan bahwa uji stationeritas telah selesai. Langkah selanjutnya adalah menentukan model awal dari plot ACF dan PACF.



Gambar 4.5: Hasil Plot ACF (*Autocorrelation Function*)

Untuk mengetahui *orde* atau derajat tertinggi pada MA (*Moving Average*) maka dapat dilihat pada hasil plot ACF dari data yang telah stasioner terhadap rata-rata atau *mean* dan stasioner terhadap *varians*. Banyaknya *lag* yang keluar menggambarkan *orde* maksimum model. Dan pada Gambar 4.5 dapat diamati bahwa terdapat 1 *lag* yang melewati garis batas merah putus-putus yaitu pada *lag* 1 dengan nilai *Autocorrelation* = -0,246122. Dengan kata lain maka *orde* atau derajat maksimum dari model MA (*Moving Average*) adalah 1. Dari data yang sama juga akan dilakukan plot PACF (*Partial Autocorrelation Function*) untuk mengetahui derajat dari model AR (*Autoregressive*). Dan dapat diamati bahwa terdapat 3 *lag* yang melewati garis batas merah putus-putus yaitu pada *lag* 1 dengan nilai *Autocorrelation* = -0,246122, *lag* 2 dengan nilai *Autocorrelation* = -0,240513, dan *lag* 3 dengan nilai *Autocorrelation* = -0,210518 seperti yang ditunjukkan pada Gambar 4.6 berikut ini.



Gambar 4.6: Hasil Plot PACF (*Partial Autocorrelation Function*)

Ada berbagai kemungkinan model sementara yang terbentuk dengan melihat *lag* yang keluar dari batas garis merah putus-putus dimana banyaknya *lag* yang keluar menggambarkan derajat atau *orde* maksimum untuk AR dan MA. Pada plot ACF tampak terdapat 1 garis atau *lag* yang keluar atau orde tertinggi untuk MA yaitu 1, sedangkan pada plot PACF tampak terdapat 3 garis yang keluar atau orde tertinggi untuk AR yaitu 3. Sehingga dari kombinasi derajat AR dan derajat MA maka terdapat 7 model yang memungkinkan yaitu ARIMA (3,1,1), ARIMA (3,1,0), ARIMA (0,1,1), ARIMA (1,1,1), ARIMA (2,1,1), ARIMA (2,1,0), dan ARIMA (1,1,0) yang selanjutnya akan dilakukan uji *white noise* dan *significant* dari tiap-tiap model antara lain seperti pada tabel berikut ini:

Tabel 4.2: *Model Sementara Berdasarkan Plot ACF dan PACF*

model	white noise	significant	keterangan
3,1,1	x	x	-
3,1,0	x	x	-
0,1,1	x	v	-
1,1,1	v	v	tanpa konstanta
2,1,1	x	x	-
2,1,0	x	v	-
1,1,0	x	v	-

Kemudian juga akan dilakukan pemeriksaan diagnosa. Pemeriksaan diagnosa pada *residual* meliputi uji asumsi *white noise* (independen dan identik) dan berdistribusi normal. Pengujian dengan menggunakan uji *Ljung Box* dilakukan dengan hipotesa sebagai berikut:

Hipotesa

$$H_0 : p_1 = p_2 = \dots = p_k = 0$$

$H_1$  : minimal ada salah satu nilai  $p_k \neq 0, k = 1, 2, \dots, K$ .

Statistik uji

$$Q = n(n+2) = \sum_{k=0}^n (n-k)^{-1} \hat{p}_k^2 \text{ dengan}$$

$k$  : lag maksimum

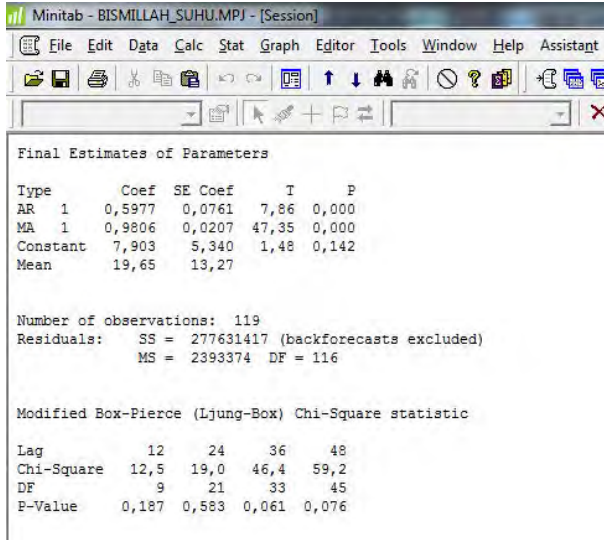
$n$  : banyak pengamatan

$\hat{p}_k^2$  : sampel ACF residual pada lag- $K$

Daerah kritis

$$\text{Jika } Q > X^2(1 - \alpha) ; \text{ df} = K - p - q$$

dimana  $p$  dan  $q$  order dari ARMA, atau *p-value* di Ljung-Box  $> \alpha(0,05)$ , maka  $H_0$  diterima artinya *residual white noise*. Dan dalam hal ini  $H_0$  diterima, artinya *residual white noise* karena *p-value* ARIMA (1, 1, 1)  $> 0,05$  yaitu 0,187 ; 0,583; 0,061 ; dan 0,076 seperti pada Gambar 4.7 berikut ini.



Final Estimates of Parameters

Type	Coef	SE Coef	T	P
AR 1	0,5977	0,0761	7,86	0,000
MA 1	0,9806	0,0207	47,35	0,000
Constant	7,903	5,340	1,48	0,142
Mean	19,65	13,27		

Number of observations: 119  
Residuals: SS = 277631417 (backforecasts excluded)  
MS = 2393374 DF = 116

Modified Box-Pierce (Ljung-Box) Chi-Square statistic

Lag	12	24	36	48
Chi-Square	12,5	19,0	46,4	59,2
DF	9	21	33	45
P-Value	0,187	0,583	0,061	0,076

Gambar 4.7: Hasil Model ARIMA(1,1,1)

Pengujian parameter dilakukan dengan menggunakan statistik uji *t-student* dari model ARIMA (1,1,1)

Hipotesis :

$$H_0 = \text{Estimasi parameter } (\hat{\phi}_t = 0)$$

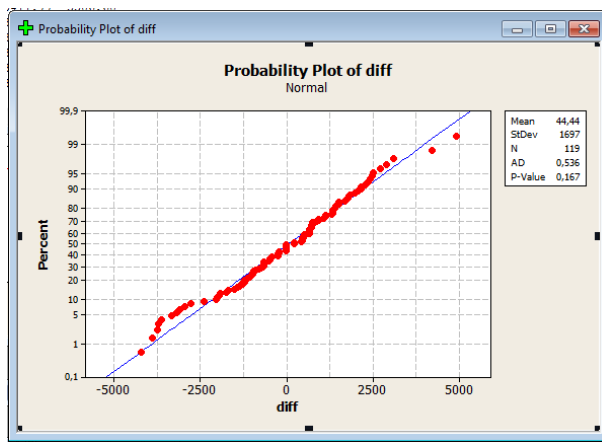
$$H_1 = \text{Estimasi parameter } (\hat{\phi}_t \neq 0)$$

$$\text{Statistik uji : } t = \frac{\hat{\theta}}{SE(\hat{\theta})}$$

Daerah Kritis :

Jika  $|t_{hitung}| > t_{\frac{\alpha}{2}, n-1}$  atau  $p\text{-value} < \alpha(0,05)$ , maka  $H_0$  ditolak (parameter significant).

Dan untuk model ARIMA(1,1,1) akan ditunjukkan *significant* sebagai berikut ini. Dan pada Gambar 4.7 tampak bahwa *p-value* ARIMA (1, 1, 1) < 0,05 yaitu bernilai 0,00 untuk AR(1) dan bernilai 0,00 untuk MA(1) maka  $H_0$  ditolak, artinya parameter dalam model telah *significant*.



Gambar 4.8: *Plot Kenormalan ARIMA(1,1,1)*

Pengujian asumsi distribusi normal dapat dilakukan dengan menggunakan uji *Kolmogorov-Smirnov* dengan  $\alpha = 0,05$ . Pengujian ini dilakukan dengan menggunakan *software* minitab16. Dengan menggunakan model ARMA(1,1,1) diperoleh uji normalitas seperti yang ditunjukkan pada Gambar 4.8.



Dengan bentuk umum ARIMA Box-Jenkins seperti Persamaan 2.8

$$\phi_p(B)(1-B)^D Y_t = \theta_0 + \theta_Q(B)a_t$$

sehingga untuk model ARIMA (1,1,1) didapat

$$\phi_1(B)(1-B)^1 Y_t = \theta_1(B)a_t \text{ atau}$$

$$(1 - \phi_1 B)(1 - B)Y_t = (1 - \theta_1 B)a_t \text{ atau}$$

$$(1 - \phi_1 B - B + \phi_1 B^2)Y_t = (1 - \theta_1 B)a_t \text{ atau}$$

$$Y_t - \phi_1 Y_{t-1} - Y_{t-1} + \phi_1 Y_{t-2} = a_t - \theta_1 a_{t-1} \text{ atau}$$

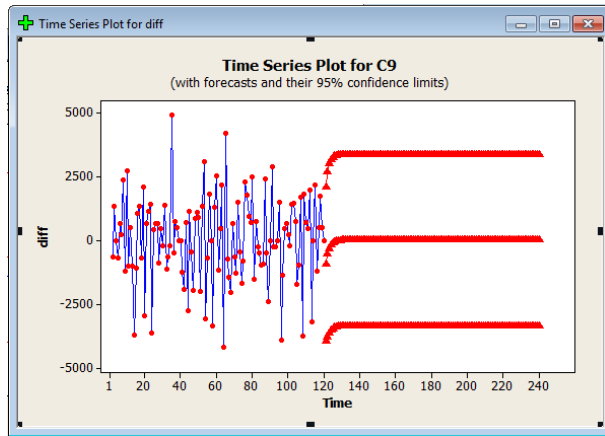
$$Y_t = \phi_1 Y_{t-1} + Y_{t-1} - \phi_1 Y_{t-2} + a_t - \theta_1 a_{t-1} \text{ atau}$$

dengan  $\phi_1 = 0,5977; \theta_1 = 0,9806$  (seperti yang ditunjukkan pada Gambar 4.7), sehingga

$$Y_t = 0,5977Y_{t-1} + Y_{t-1} - 0,5977Y_{t-2} + a_t - 0,9806a_{t-1} \text{ atau}$$

$$Y_t = 1,5977Y_{t-1} - 0,5977Y_{t-2} + a_t - 0,9806a_{t-1}$$

Dari model yang ada, akan dilakukan peramalan untuk 120 data suhu udara dengan menggunakan *software* minitab16 seperti pada Gambar 4.9.



Gambar 4.9: *Plot Peramalan ARIMA(1,1,1)*

Hasil peramalan ini nantinya akan diolah lebih lanjut menggunakan algoritma Filter Kalman yang dalam hal ini akan memperbaiki hasil estimasi dari peramalan dengan metode ARIMA (*Autoregressive Integrated Moving Average*) ini. Dimana nantinya akan dilakukan tiga perbandingan grafik antara data asli, hasil peramalan ARIMA, dan juga hasil estimasi dengan menggunakan algoritma Filter Kalman.

#### 4.1.2 Analisis Model Data dan Peramalan pada Data Kecepatan Angin

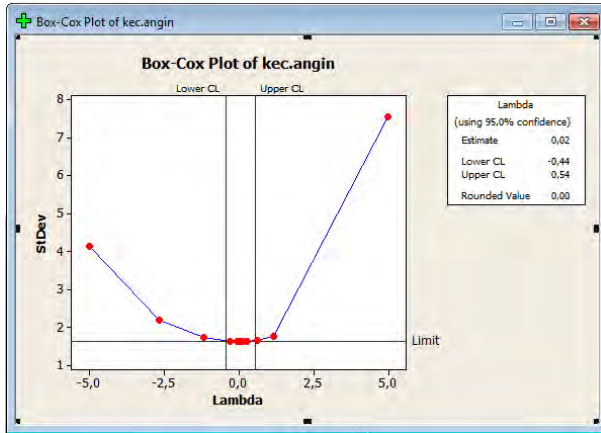
Pada tahapan ini akan dianalisis data yang didapat dari BMKG stasiun Juanda-Surabaya. Data yang didapat merupakan hasil pengamatan dari pihak terkait dengan menggunakan alat ukur kecepatan angin. Dan dalam hal ini data yang dianalisis adalah data kecepatan angin pada bulan Januari, Februari, Maret, dan April. Data yang ada akan diolah lebih lanjut dengan menggunakan *software* Minitab16. Dengan menggunakan metode analisis *time series* maka akan dicoba untuk mencari model data yang sesuai. Akan tetapi, kemungkinan model yang ada akan lebih dari satu tergantung dari hasil plot data yang ada. Berbagai kemungkinan model yang ada tersebut akan diuji kelayakannya sesuai dengan aturan-aturan yang ada pada metode analisis *time series* ini. Lalu akan diramalkan sebanyak 120 hari berikutnya.

Tabel 4.3: *Data Kecepatan Angin Bulan Januari-April 2013*

TANGGAL	JANUARI	FEBRUARI	MARET	APRIL
1	5	8	7,5	8
2	6,5	7	7	5
3	6	10	12	6
4	6,5	4	8,5	5
5	8	8	11,5	5,5
6	8	9	9,5	5
7	9,5	10	5,5	5,5
8	10,5	7	7,5	4,5
9	14,5	8	8,5	5
10	13	13	6	4,5

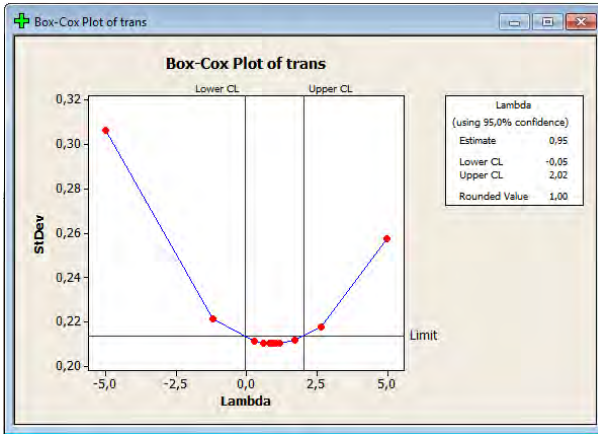
TANGGAL	JANUARI	FEBRUARI	MARET	APRIL
11	14,5	6	7,5	5
12	11	9	6	6,5
13	11	7	6,5	6
14	11	12	8	5
15	8	9	7,5	8
16	8	9	7	6,5
17	7,5	10	5,5	7
18	11,5	12	6	5,5
19	10	12	4,5	8
20	8	16	6,5	6,5
21	8,5	17	5	6
22	6,5	18	5,5	7
23	10	14	6,5	8
24	5	18	6	6
25	7,5	15	5,5	7
26	9	12	8	5,5
27	6	17	6	7
28	5,5	20	5,5	8,5
29	9		7	8
30	8,5		6	7
31	6		4	

Dari data di atas akan diuji stasioneritasnya dalam *varians* dan *mean*. Syarat terpenting yang harus dipenuhi agar data dapat diolah dengan metode *time series* adalah stasioner, baik dalam *varians* maupun *mean*[6]. Sebuah deret disebut stasioner jika sifat statistiknya bebas dari waktu periode selama pengamatan[7].



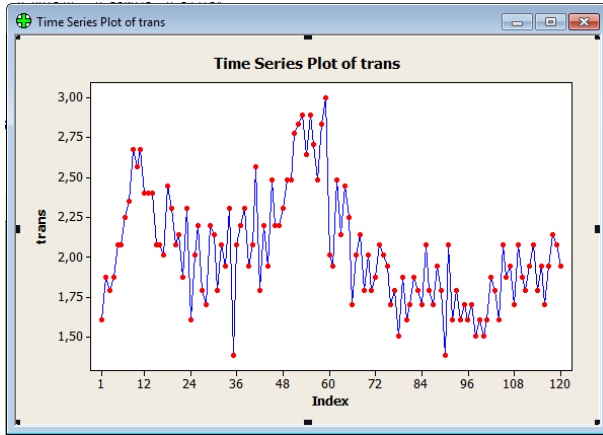
Gambar 4.10: Hasil Plot Box-Cox Data Asli

Pada tahapan ini akan dilakukan uji stasioneritas dalam *varians*. Suatu data dapat dikatakan stasioner dalam *varians* jika *varians* dari data tidak dipengaruhi oleh waktu. Kestasioneran data secara *varians* dapat dilihat dari hasil transformasi Box-Cox di mana dikatakan stasioner jika nilai rounded value-nya bernilai 1[6]. Dari data pada Tabel 4.3 akan dilakukan plot Box-Cox untuk mengetahui nilai *varians*-nya. Setelah dilakukan plot Box-Cox seperti pada Gambar 4.10 dapat diamati dalam kotak dialog bahwa nilai *lambda* berada diantara -0,44 dan 0,54 dengan *estimate value* = 0,02 dan *rounded value* = 0,00. Hal ini dapat diartikan bahwa data yang didapat dari BMKG stasiun Juanda Surabaya tersebut belum stasioner terhadap *varians*, dikarekan nilai *rounded value*  $\neq 1$  sehingga untuk menstasionerkan suatu data yang tidak stasioner terhadap *varians* dapat dilakukan transformasi pada data yang tidak stasioner tersebut.



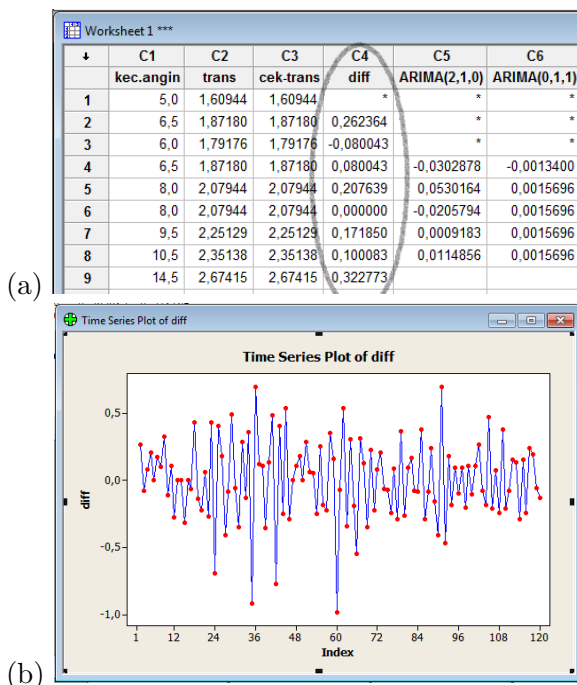
Gambar 4.11: Hasil Plot Box-Cox Data Transformasi

Setelah dilakukan transformasi, data hasil transformasi tersebut juga akan dilakukan plot Box-Cox untuk mengetahui kestasioneran dari data hasil transformasi tersebut. Karena pada plot Box-Cox awal didapat nilai *rounded value* = 0,00 maka dengan mengacu pada Tabel 2.1 transformasi yang dilakukan  $\ln$  dari data awal atau  $\ln(Z_t)$ . Dan seperti pada Gambar 4.11 dapat diamati dalam kotak dialog bahwa nilai *lambda* berada diantara -0,05 dan 2,02 dengan *estimate value* = 0,95 dan *rounded value* = 1,00. Hal ini dapat diartikan bahwa data yang didapat dari BMKG stasiun Juanda Surabaya tersebut telah stasioner terhadap *varians*, dikarekan nilai *rounded value*= 1 sehingga untuk selanjutnya juga akan diuji stasioneritas dala *mean*. Setelah didapat kestasioneran terhadap *varians* maka selanjutnya menguji kestasioneran terhadap *mean*. Dengan menggunakan *Time series* plot, maka kestasioneran data dalam rata-rata (*mean*) akan terlihat. Gambar yang muncul adalah plot *series* data hasil transformasi seperti berikut ini.



Gambar 4.12: *Plot Data Hasil Transformasi*

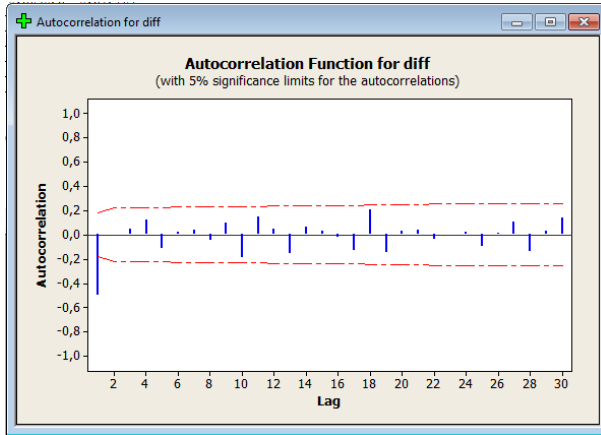
Setelah didapat kestasioneran terhadap *varians* maka juga akan diuji kestasioneran terhadap *mean*. Dengan menggunakan *Time series plot*, maka kestasioneran data dalam rata-rata (*mean*) akan terlihat. Gambar yang muncul adalah plot series yang sesuai dengan data hasil transformasi dimana dari tampilan plot tersebut dapat diamati kestasioneran data terhadap rata-rata atau *mean*. Pada Gambar 4.12, diperoleh plot data yang fluktuatif, karena tidak terdapat pola yang teratur maka dapat diartikan bahwa data hasil transformasi tersebut tidak stasioner dalam *mean*. Sehingga agar didapat data yang stasioner terhadap *mean* maka data harus dideferensial terlebih dahulu sampai plot data tidak berpola fluktuatif lagi. Cara mendifferensi (*stat-time series-differences*). Untuk mengetahui apakah data hasil diferensi tersebut stasioner terhadap *mean* maka kemudian diplot kembali data hasil *differences* tadi untuk melihat bentuk polanya dengan *time series plot*.



Gambar 4.13: (a) Hasil Differences dan (b) Plot Hasil Differences

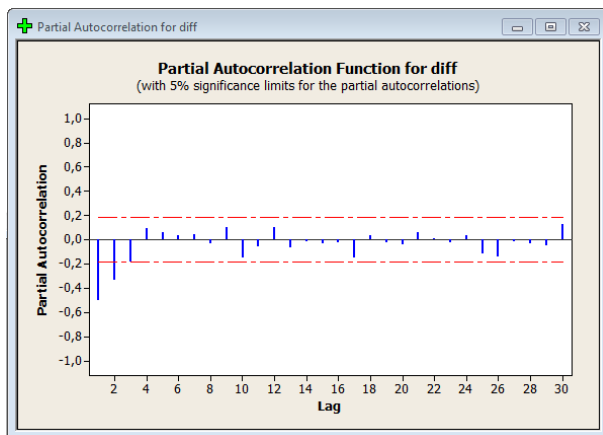
Setelah dideferensi sekali, data menjadi stasioner dalam *mean* seperti yang terlihat pada Gambar plot 4.13 bagian (b) dimana pola data cenderung tidak fluktuatif lagi. Selain itu dapat kita lihat juga pada plot ACF dari data yang telah di *differencing* pada Gambar 4.14 tampak nilai-nilai autokorelasinya tidak turun secara lambat lagi menuju nol yang artinya bahwa data tersebut telah stasioner dalam *mean*. Karena data sudah stationer dalam *mean* dan *varians* maka disimpulkan bahwa uji stationeritas telah selesai. Langkah selanjutnya adalah menentukan model awal dari plot ACF dan PACF.





Gambar 4.14: Hasil Plot ACF (Autocorrelation Function)

Untuk mengetahui *orde* atau derajat tertinggi pada MA (*Moving Average*) maka dapat dilihat pada hasil plot ACF dari data yang telah stasioner terhadap rata-rata atau *mean* dan stasioner terhadap *varians*. Banyaknya *lag* yang keluar menggambarkan *orde* maksimum model. Dan pada Gambar 4.14 dapat diamati bahwa terdapat 1 *lag* yang melewati garis batas merah putus-putus yaitu pada *lag* 1 dengan nilai *Autocorrelation* = -0,502370. Dengan kata lain maka *orde* atau derajat maksimum dari model MA (*Moving Average*) adalah 1. Dari data yang sama juga akan dilakukan plot PACF (*Partial Autocorrelation Function*) untuk mengetahui derajat dari model AR (*Autoregressive*). Dan dapat diamati bahwa terdapat 2 *lag* yang melewati garis batas merah putus-putus yaitu pada *lag* 1 dengan nilai *Autocorrelation* = -0,502370 dan *lag* 2 dengan nilai *Autocorrelation* = -0,335573 seperti yang ditunjukkan pada Gambar 4.15 berikut ini.



Gambar 4.15: Hasil Plot PACF (*Partial Autocorrelation Function*)

Ada berbagai kemungkinan model sementara yang terbentuk dengan melihat *lag* yang keluar dari batas garis merah putus-putus dimana banyaknya *lag* yang keluar menggambarkan derajat atau *orde* maksimum untuk AR dan MA. Pada plot ACF tampak terdapat 1 garis atau *lag* yang keluar atau orde tertinggi untuk MA yaitu 1, sedangkan pada plot PACF tampak terdapat 2 garis yang keluar atau orde tertinggi untuk AR yaitu 2. Sehingga dari kombinasi derajat AR dan derajat MA maka terdapat 4 model yang memungkinkan yaitu ARIMA (2,1,1), ARIMA (1,1,0), ARIMA (1,1,1), dan ARIMA (0,1,1) yang selanjutnya akan dilakukan uji *white noise* dan *significant* dari tiap-tiap model antara lain seperti pada tabel berikut ini:

Tabel 4.4: *Model Sementara Berdasarkan Plot ACF dan PACF*

model	white noise	significant	keterangan
2,1,1	v	x	-
1,1,0	x	v	-
1,1,1	v	x	-
0,1,1	v	v	tanpa konstanta

Kemudian juga akan dilakukan pemeriksaan diagnosa. Pemeriksaan diagnosa pada *residual* meliputi uji asumsi *white noise* (independen dan identik) dan berdistribusi normal. Pengujian dengan menggunakan uji *Ljung Box* dilakukan dengan hipotesa sebagai berikut:

Hipotesa

$$H_0 : p_1 = p_2 = \dots = p_k = 0$$

$H_1$  : minimal ada salah satu nilai  $p_k \neq 0, k = 1, 2, \dots, K$ .

Statistik uji

$$Q = n(n+2) = \sum_{k=0}^n (n-k)^{-1} \hat{p}_k^2 \text{ dengan}$$

$k$  : lag maksimum

$n$  : banyak pengamatan

$\hat{p}_k^2$  : sampel ACF residual pada lag- $K$

Daerah kritis

$$\text{Jika } Q > X^2(1 - \alpha) ; \text{ df} = K - p - q$$

dimana  $p$  dan  $q$  order dari ARMA, atau *p-value* di Ljung-Box  $> \alpha(0,05)$ , maka  $H_0$  diterima artinya *residual white noise*. Dan dalam hal ini  $H_0$  diterima, artinya *residual white noise* karena *p-value* ARIMA (1, 1, 1)  $> 0,05$  yaitu 0,313 ; 0,420; 0,248 ; dan 0,462 seperti pada Gambar 4.16 berikut ini.

Final Estimates of Parameters

Type	Coef	SE Coef	T	P
MA 1	0,6183	0,0724	8,54	0,000
Constant	0,001570	0,008705	0,18	0,857
Mean	0,001570	0,008705		

Number of observations: 119  
 Residuals: SS = 7,17065 (backforecasts excluded)  
 MS = 0,06129 DF = 117

Modified Box-Pierce (Ljung-Box) Chi-Square statistic

Lag	12	24	36	48
Chi-Square	11,6	22,7	39,2	46,3
DF	10	22	34	46
P-Value	0,313	0,420	0,248	0,462

Gambar 4.16: Hasil Model ARIMA(0,1,1)

Pengujian parameter dilakukan dengan menggunakan statistik uji *t-student* dari model ARIMA (0,1,1)

Hipotesis :

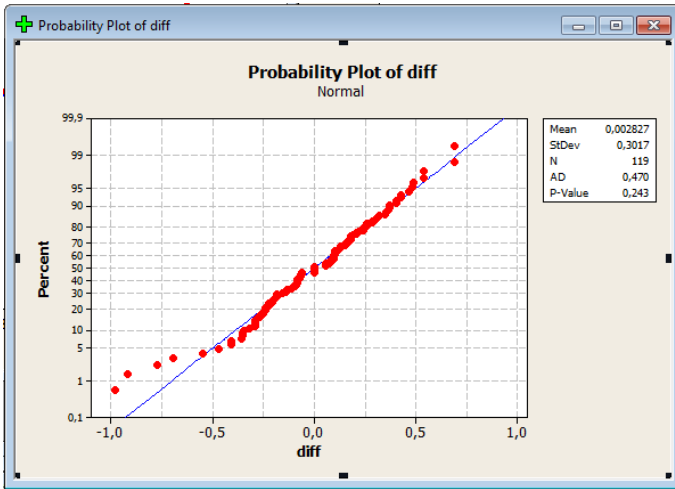
$$H_0 = \text{Estimasi parameter } (\hat{\phi}_t = 0)$$

$$H_1 = \text{Estimasi parameter } (\hat{\phi}_t \neq 0)$$

$$\text{Statistik uji : } t = \frac{\hat{\theta}}{SE(\hat{\theta})}$$

Daerah Kritis : Jika  $|t_{hitung}| > t_{\frac{\alpha}{2}, n-1}$  atau *p-value*  $< \alpha(0,05)$ , maka  $H_0$  ditolak (parameter significant).

Dan untuk model ARIMA(0,1,1) akan ditunjukkan *significant* sebagai berikut ini. Dan pada Gambar 4.16 tampak bahwa  $p\text{-value}$  ARIMA (0, 1, 1)  $< 0,05$  yaitu bernilai 0,00 untuk MA(1) maka  $H_0$  ditolak, artinya parameter dalam model telah *significant*.



Gambar 4.17: *Plot Kenormalan ARIMA(0,1,1)*

Pengujian asumsi distribusi normal dapat dilakukan dengan menggunakan uji *Kolmogorov-Smirnov* dengan  $\alpha = 0,05$ . Pengujian ini dilakukan dengan menggunakan *software* minitab16. Dengan menggunakan model ARMA(0,1,1) diperoleh uji normalitas seperti yang ditunjukkan pada Gambar 4.17.

Bentuk umum ARIMA Box-Jenkins seperti Persamaan 2.8

$$\phi_p(B)(1 - B)^D Y_t = \theta_0 + \theta_Q(B)a_t$$

sehingga untuk model ARIMA (0,1,1) didapat

$$\phi_0(B)(1 - B)^1 Y_t = \theta_1(B)a_t \text{ atau}$$

$$(1 - B)Y_t = (1 - \theta_1 B)a_t \text{ atau}$$

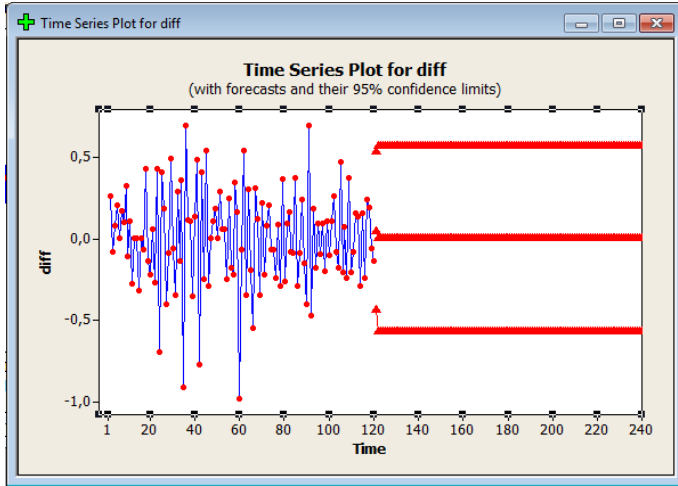
$$Y_t - Y_{t-1} = a_t - \theta_1 a_{t-1} \text{ atau}$$

$$Y_t = Y_{t-1} + a_t - \theta_1 a_{t-1}$$

dengan  $\theta_1 = 0,6183$  (seperti yang ditunjukkan pada Gambar 4.16)

$$Y_t = Y_{t-1} + a_t - 0,6183a_{t-1}$$

Dari model yang ada, akan dilakukan peramalan untuk 120 data yang akan datang menggunakan *software* minitab16 seperti pada Gambar 4.18.



Gambar 4.18: *Plot Peramalan ARIMA(0,1,1)*

Hasil peramalan ini nantinya akan diolah lebih lanjut menggunakan algoritma Filter Kalman yang dalam hal ini akan memperbaiki hasil estimasi dari peramalan dengan metode ARIMA (*Autoregressive Integrated Moving Average*) ini. Dimana nantinya akan dilakukan tiga perbandingan grafik antara data asli, hasil peramalan ARIMA, dan juga hasil estimasi dengan menggunakan algoritma Filter Kalman.

## 4.2 Penerapan Metode Filter Kalman dan Simulasi dengan Software Matlab

Pada tahapan ini akan dilakukan simulasi terhadap hasil perhitungan sebelumnya ke dalam algoritma Filter Kalman dengan menggunakan *software* Matlab R2010a. Pada kasus ini, akan diestimasi nilai koefisien polinomial  $a_0$  dan  $a_1$  yang ada pada Persamaan 2.18. Karena dalam estimasi ini akan mengambil polinomial dengan  $n = 2$  sehingga untuk  $y_i^0 = a_0 + a_1 m_i$  dengan  $x(t_i) = \begin{bmatrix} a_{0,i} \\ a_{1,i} \end{bmatrix}$  dan  $H_i = [ 1 \quad m_i ]$  dimana  $m_i$  merupakan data ke- $i$ . Selain itu juga akan ditentukan beberapa sebagai nilai awal yaitu

A sebagai matriks sistem.

Q sebagai matriks *kovarians*.

Rk = R sebagai matriks *kovarians*.

$a_{00}$  sebagai masukkan nilai awal  $a_{00}$ .

$a_{10}$  sebagai masukkan nilai awal  $a_{10}$ .

Setelah mempunyai nilai awal maka untuk selanjutnya akan mencari nilai dari *noise* dengan *random* yang berdistribusi normal.

Model Sistem

$$BU : X_{k+1} = A_k X_k + B_k U_k + G_k w_k$$

$$\begin{bmatrix} a_{0,i} \\ a_{1,i} \end{bmatrix}_{k+1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a_{0,i} \\ a_{1,i} \end{bmatrix}_k + w_k$$

Model Pengukuran

$$BU : z_k = H_k X_k + v_k$$

$$y_k^0 = [ 1 \quad m_i ] \begin{bmatrix} a_0 \\ a_1 \end{bmatrix}_k + v_k$$

Pada Tahap Prediksi

$$\hat{X}_k^- = A \hat{X}_{k-1} + w_k$$



dengan  $\hat{X}_0 = \begin{bmatrix} -8 \\ 8 \end{bmatrix}$  ;  $A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$  dan  $w_k$  yang didapat dengan *random* berdistribusi normal.

$$P_k^- = AP_{k-1}A^T + Qk$$

$$\text{dengan } P_0 = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix} \text{ dan } Qk = \begin{bmatrix} 0,001 & 0 \\ 0 & 0,001 \end{bmatrix}.$$

Pada Tahap Koreksi

untuk mendapatkan nilai dari *Kalman Gain* akan menggunakan hasil perhitungan pada tahap prediksi.

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \text{ dengan } R = 1.$$

dan untuk mendapatkan nilai koreksi dari  $\hat{X}_k$  akan juga menggunakan nilai  $\hat{X}_k^-$  yang telah didapatkan pada tahap sebelumnya.

$$\hat{X}_k = \hat{X}_k^- + K_k(z_k - H\hat{X}_k^-)$$

dengan  $z_k$  merupakan identik dengan  $y_k^0$  yang didapatkan dari *bias* atau selisih antara data dengan hasil peramalan pada minitab pada waktu ke-k.

Lalu untuk menghitung nilai *kovarians error* pada tahap koreksi ini akan menggunakan nilai dari  $P_k^-$  yang didapatkan dari perhitungan di tahap prediksi.

$$P_k = (I - K_k H)P_k^- \text{ dengan } I \text{ merupakan matriks identitas.}$$

### 4.3 Simulasi dan Analisis Simulasi

Pada sub bab ini simulasi dilakukan dengan menerapkan algoritma Filter Kalman pada persamaan. Hasil simulasi akan dievaluasi dengan cara membandingkan keadaan *real* dengan hasil estimasi Filter Kalman dan ARIMA.

Dan pada tahapan ini, akan dilakukan 3 kali simulasi dengan banyak menjalankan perangkat lunak yang berbeda-beda yaitu 3, 5, dan 7 kali menjalankan perangkat lunak baik pada data suhu udara maupun kecepatan angin. Dan akan ditentukan beberapa nilai awalan untuk variabel yang ada yaitu diantaranya adalah

A sebagai matriks sistem adalah  $A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

Q sebagai matriks *kovarians* adalah 0.001 dengan  $Q_k = \begin{pmatrix} 0,001 & 0 \\ 0 & 0,0001 \end{pmatrix}$

Rk=R sebagai matriks *kovarians* adalah 1.

$a_{00}$  sebagai masukkan nilai awal  $a_{00}$  adalah -8.

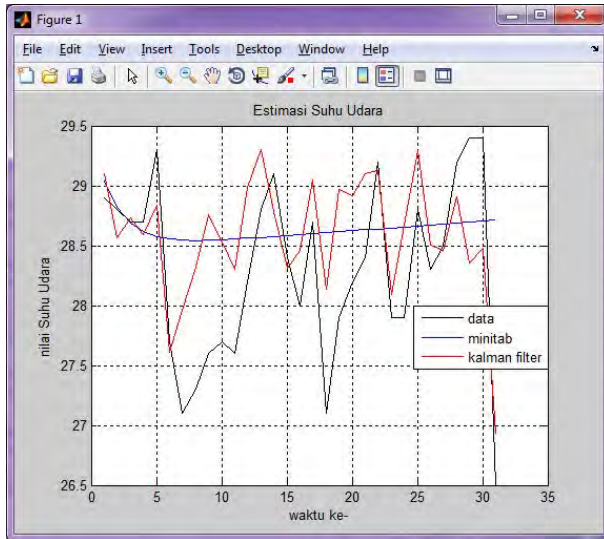
$a_{10}$  sebagai masukkan nilai awal  $a_{10}$  adalah 8.

nilai awal estimasi  $\hat{x}_0 = \begin{pmatrix} a_{00} \\ a_{10} \end{pmatrix}$  nilai awal kovarians error

$$P_0 = \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix}$$

#### 4.3.1 Simulasi I pada Data Suhu Udara dan Kecepatan Angin

Dengan nilai awalan yang telah ditentukan maka akan dilakukan perhitungan menggunakan algoritma Filter Kalman dengan menjalankan perangkat lunak sebanyak 3 kali.



Gambar 4.19: Hasil Simulasi I pada Data Suhu Udara

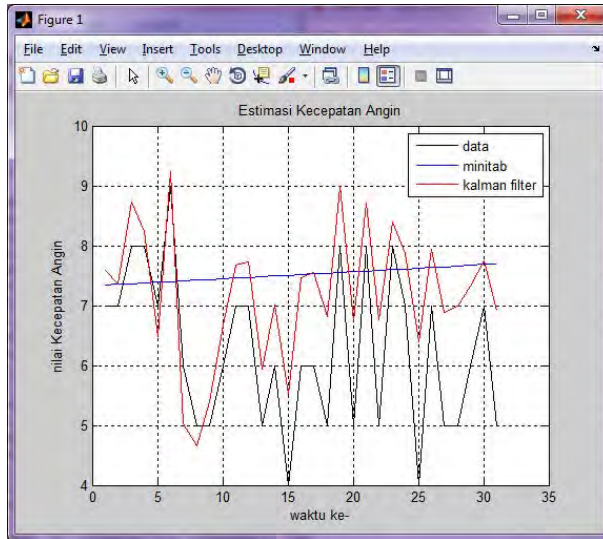
Dan pada Gambar 4.19 dapat diamati bahwa hasil estimasi suhu udara dengan menggunakan algoritma Filter Kalman memiliki akurasi ketepatan yang lebih tinggi terhadap data asli jika dibandingkan dengan hasil prediksi dengan menggunakan ARIMA. Selain itu, pada nilai mutlak kesalahan prediksi dari Filter Kalman tampak cenderung lebih kecil dari pada ARIMA, seperti ditunjukkan pada tabel berikut:

Tabel 4.5: Nilai Mutlak Kesalahan Prediksi Suhu Udara pada Simulasi I

Prediksi ke-	ARIMA	Filter Kalman
1	0,1381	0,2049
2	0,0206	0,235266667
3	0,0079	0,030866667

Prediksi ke-	ARIMA	Filter Kalman
4	0,0819	0,1086
5	0,7232	0,4601
6	0,8553	0,0913
7	1,4457	0,8639
8	1,2432	1,0204
9	0,9449	1,156166667
10	0,8492	0,8287
11	0,955	0,7069
12	0,3616	0,785133333
13	0,2311	0,499766667
14	0,5236	0,311433333
15	0,1841	0,0891
16	0,592	0,459266667
17	0,1001	0,3515
18	1,5078	1,0387
19	0,7158	1,0713
20	0,4238	0,712366667
21	0,2317	0,6981
22	0,5603	0,071533333
23	0,7477	0,194566667
24	0,7557	0,770366667
25	0,1364	0,492066667
26	0,3716	0,2075
27	0,1796	0,035233333
28	0,5125	0,288166667
29	0,7045	1,042766667
30	0,6966	0,9228
31	2,2114	0,4219

Sedangkan pada hasil menjalankan algoritma Filter Kalman dari simulasi I pada data kecepatan angin ditunjukkan seperti gambar berikut.



Gambar 4.20: Hasil Simulasi I pada Data Kecepatan Angin

Dan pada Gambar 4.20 dapat diamati bahwa hasil estimasi kecepatan angin dengan menggunakan algoritma Filter Kalman memiliki akurasi ketepatan yang lebih tinggi terhadap data asli jika dibandingkan dengan hasil prediksi dengan menggunakan ARIMA. Nilai mutlak kesalahan prediksi dari Filter Kalman tampak cenderung lebih kecil dari pada minitab, seperti ditunjukkan pada tabel berikut:

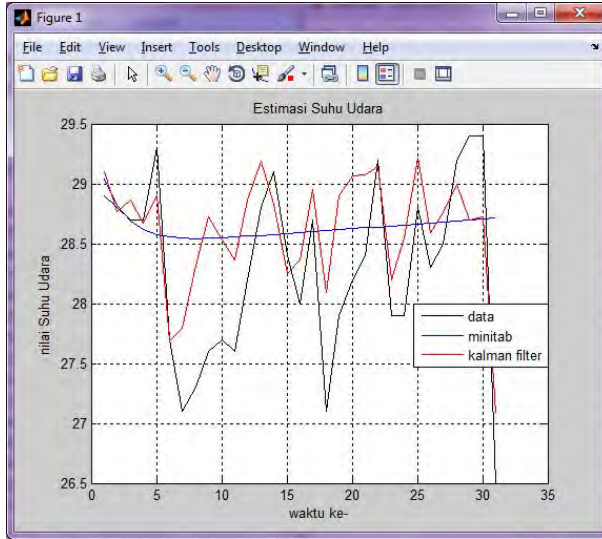
Tabel 4.6: Nilai Mutlak Kesalahan Prediksi Kecepatan Angin pada Simulasi I

Prediksi ke-	ARIMA	Filter Kalman
1	0,342915558	0,580566667
2	0,354450185	0,365633333
3	0,63399707	0,724
4	0,622426177	0,2372

Prediksi ke-	ARIMA	Filter Kalman
5	0,389162892	0,515033333
6	1,599229834	0,230033333
7	1,412395673	0,974766667
8	2,424039443	0,342
9	2,435701502	0,395066667
10	1,447381882	0,6672
11	0,459080609	0,6703
12	0,470797713	0,7203
13	2,482533223	0,945666667
14	1,494287168	1,0017
15	3,506059576	1,508866667
16	1,517850478	1,470966667
17	1,5296599	1,5524
18	2,541487874	1,8186
19	0,446665572	0,983566667
20	2,565199591	1,7503
21	0,422916608	0,712466667
22	2,588985861	1,754766667
23	0,399092973	0,390866667
24	0,61284692	0,8724
25	3,624805568	2,394133333
26	0,636783001	0,935066667
27	2,64877925	1,878
28	2,660794342	1,9903
29	1,672828309	1,328933333
30	0,684881179	0,753733333
31	2,696952982	1,924466667

#### 4.3.2 Simulasi II pada Data Suhu Udara dan Kecepatan Angin

Dengan nilai awalan yang telah ditentukan maka akan dilakukan perhitungan algoritma Filter Kalman.



Gambar 4.21: Hasil Simulasi II pada Data Suhu Udara

Dan dari Gambar 4.21 dapat diamati bahwa hasil estimasi suhu udara dengan menggunakan algoritma Filter Kalman memiliki akurasi ketepatan yang lebih tinggi terhadap data asli jika dibandingkan dengan hasil prediksi dengan menggunakan ARIMA. Nilai mutlak kesalahan prediksi dari Filter Kalman tampak cenderung lebih kecil dari pada ARIMA, seperti ditunjukkan pada tabel berikut:

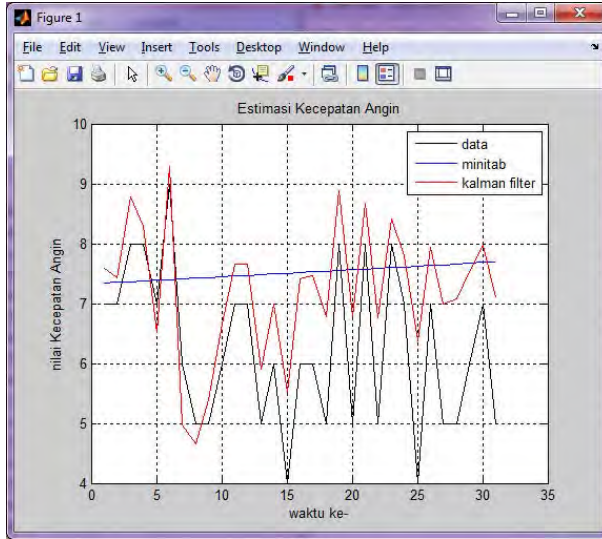
Tabel 4.7: Nilai Mutlak Kesalahan Prediksi Suhu Udara pada Simulasi II

Prediksi ke-	ARIMA	Filter Kalman
1	0,1381	0,20474
2	0,0206	0,03132
3	0,0079	0,16588

Prediksi ke-	ARIMA	Filter Kalman
4	0,0819	0,02748
5	0,7232	0,39668
6	0,8553	0,00994
7	1,4457	0,69416
8	1,2432	1,02192
9	0,9449	1,12532
10	0,8492	0,82768
11	0,955	0,76272
12	0,3616	0,66798
13	0,2311	0,39046
14	0,5236	0,27318
15	0,1841	0,14846
16	0,592	0,3624
17	0,1001	0,24862
18	1,5078	0,9906
19	0,7158	1,00148
20	0,4238	0,86512
21	0,2317	0,67836
22	0,5603	0,05856
23	0,7477	0,29372
24	0,7557	0,6489
25	0,1364	0,41764
26	0,3716	0,29226
27	0,1796	0,26182
28	0,5125	0,20368
29	0,7045	0,70364
30	0,6966	0,67608
31	2,2114	0,5851

Sedangkan pada hasil menjalankan algoritma Filter Kalman dari simulasi II pada data kecepatan angin ditunjukkan seperti gambar berikut.





Gambar 4.22: Hasil Simulasi II pada Data Kecepatan Angin

Dan pada Gambar 4.22 dapat diamati bahwa hasil estimasi kecepatan angin dengan menggunakan algoritma Filter Kalman memiliki akurasi ketepatan yang lebih tinggi terhadap data asli jika dibandingkan dengan hasil prediksi dengan menggunakan ARIMA. Nilai mutlak kesalahan prediksi dari Filter Kalman tampak cenderung lebih kecil dari pada ARIMA, seperti ditunjukkan pada tabel berikut:

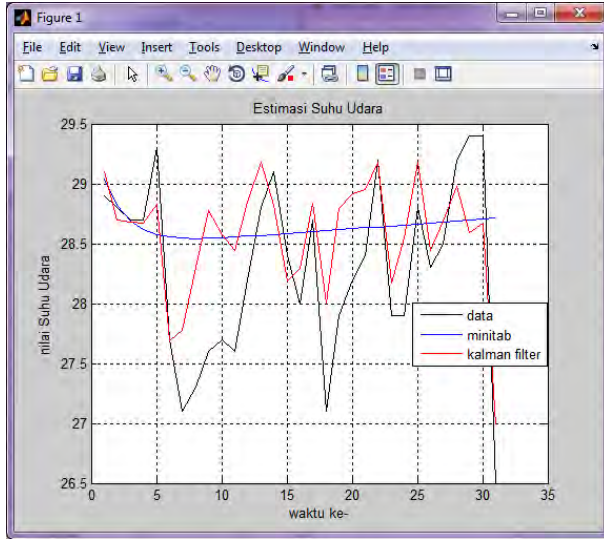
Tabel 4.8: Nilai Mutlak Kesalahan Prediksi Kecepatan Angin pada Simulasi II

Prediksi ke-	ARIMA	Filter Kalman
1	0,342915558	0,57986
2	0,354450185	0,43092
3	0,63399707	0,7862
4	0,622426177	0,2822

Prediksi ke-	ARIMA	Filter Kalman
5	0,389162892	0,47628
6	1,599229834	0,28768
7	1,412395673	1,0228
8	2,424039443	0,3373
9	2,435701502	0,38708
10	1,447381882	0,6544
11	0,459080609	0,66328
12	0,470797713	0,6513
13	2,482533223	0,90534
14	1,494287168	0,98574
15	3,506059576	1,51708
16	1,517850478	1,4174
17	1,5296599	1,46962
18	2,541487874	1,78108
19	0,446665572	0,88248
20	2,565199591	1,78432
21	0,422916608	0,67762
22	2,588985861	1,75644
23	0,399092973	0,41454
24	0,61284692	0,8067
25	3,624805568	2,37292
26	0,636783001	0,9419
27	2,64877925	1,985
28	2,660794342	2,07246
29	1,672828309	1,53516
30	0,684881179	0,97878
31	2,696952982	2,08948

#### 4.3.3 Simulasi III pada Data Suhu Udara dan Kecepatan Angin

Dengan nilai awalan yang telah ditentukan maka akan dilakukan perhitungan algoritma Filter Kalman.



Gambar 4.23: Hasil Simulasi III pada Data Suhu Udara

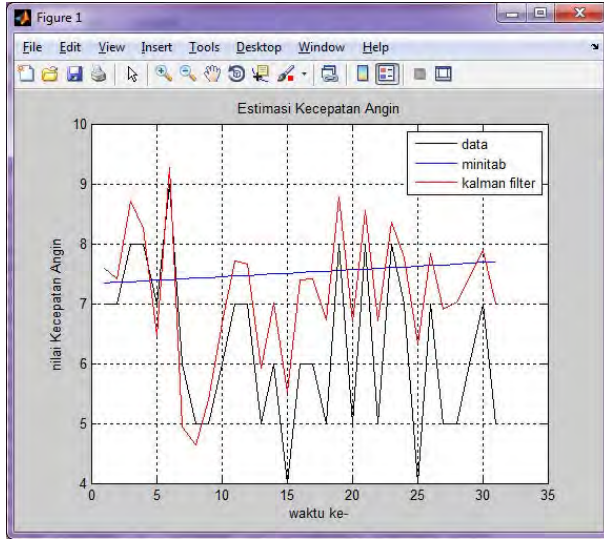
Dan pada Gambar 4.23 dapat diamati bahwa hasil estimasi suhu udara dengan menggunakan algoritma Filter Kalman memiliki akurasi ketepatan yang lebih tinggi terhadap data asli jika dibandingkan dengan hasil prediksi dengan menggunakan ARIMA. Nilai mutlak kesalahan prediksi dari Filter Kalman tampak cenderung lebih kecil dari pada ARIMA, seperti ditunjukkan pada tabel berikut:

Tabel 4.9: Nilai Mutlak Kesalahan Prediksi Suhu Udara pada Simulasi III

Prediksi ke-	ARIMA	Filter Kalman
1	0,1381	0,204742857
2	0,0206	0,098385714
3	0,0079	0,020571429

Prediksi ke-	ARIMA	Filter Kalman
4	0,0819	0,028128571
5	0,7232	0,4692
6	0,8553	0,0115
7	1,4457	0,680057143
8	1,2432	1,001028571
9	0,9449	1,1776
10	0,8492	0,876328571
11	0,955	0,8468
12	0,3616	0,643971429
13	0,2311	0,3794
14	0,5236	0,279814286
15	0,1841	0,210228571
16	0,592	0,290114286
17	0,1001	0,140457143
18	1,5078	0,908371429
19	0,7158	0,890142857
20	0,4238	0,7206
21	0,2317	0,552957143
22	0,5603	0,020214286
23	0,7477	0,272271429
24	0,7557	0,650714286
25	0,1364	0,377714286
26	0,3716	0,139628571
27	0,1796	0,188528571
28	0,5125	0,220228571
29	0,7045	0,810971429
30	0,6966	0,725285714
31	2,2114	0,490957143

Sedangkan pada hasil menjalankan algoritma Filter Kalman dari simulasi III pada data kecepatan angin ditunjukkan seperti gambar berikut.



Gambar 4.24: Hasil Simulasi III pada Data Kecepatan Angin

Dan pada Gambar 4.24 dapat diamati bahwa hasil estimasi kecepatan angin dengan menggunakan algoritma Filter Kalman memiliki akurasi ketepatan yang lebih tinggi terhadap data asli jika dibandingkan dengan hasil prediksi dengan menggunakan ARIMA. Nilai mutlak kesalahan prediksi dari KF tampak cenderung lebih kecil dari pada ARIMA, seperti ditunjukkan pada tabel berikut:

Tabel 4.10: Nilai Mutlak Kesalahan Prediksi Kecepatan Angin pada Simulasi III

Prediksi ke-	ARIMA	Filter Kalman
1	0,342915558	0,579928571
2	0,354450185	0,404485714
3	0,63399707	0,706271429

Prediksi ke-	ARIMA	Filter Kalman
4	0,622426177	0,261028571
5	0,389162892	0,527514286
6	1,599229834	0,270428571
7	1,412395673	1,052671429
8	2,424039443	0,3638
9	2,435701502	0,391857143
10	1,447381882	0,673614286
11	0,459080609	0,716842857
12	0,470797713	0,665285714
13	2,482533223	0,915928571
14	1,494287168	1,003114286
15	3,506059576	1,516585714
16	1,517850478	1,388328571
17	1,5296599	1,409142857
18	2,541487874	1,725642857
19	0,446665572	0,785342857
20	2,565199591	1,6906
21	0,422916608	0,554728571
22	2,588985861	1,7134
23	0,399092973	0,355071429
24	0,61284692	0,765428571
25	3,624805568	2,332457143
26	0,636783001	0,835257143
27	2,64877925	1,911328571
28	2,660794342	2,018685714
29	1,672828309	1,436314286
30	0,684881179	0,8833
31	2,696952982	1,988642857

Dan pada Gambar 4.19 - 4.24 dapat diamati bahwa kondisi yang paling sesuai untuk mengestimasi nilai polinomial yang nantinya berguna sebagai perbaikan prediksi dari ARIMA melalui minitab dengan Filter Kalman adalah disaat menggunakan kondisi pada simulasi III pada data suhu udara dan kondisi simulasi III pada data kecepatan angin. Dapat dibandingkan dari gambar simulasi I, simulasi II, dan simulasi III pada data suhu udara maka yang simulasi III menunjukkan hasil prediksi yang lebih akurat dan mendekati data aslinya seperti yang ditunjukkan pada Gambar 4.23.

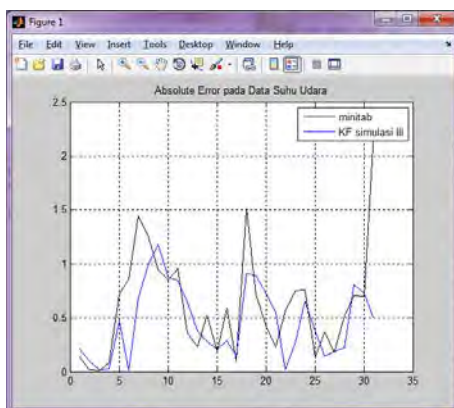
Dalam hal ini suatu peramalan tidak lepas dari suatu kesalahan perhitungan peramalan. Ada beberapa metode untuk mengevaluasi hasil peramalan yang nantinya dapat digunakan untuk mengetahui keakuratan hasil peramalan yang telah dilakukan terhadap data yang sebenarnya. Metode yang digunakan dalam penelitian ini yaitu *Mean Absolute Percentage Error* (MAPE) yang merupakan nilai tengah kesalahan persentase nilai mutlak dari suatu peramalan.

Didefinisikan[10] :  $MAPE = \frac{1}{N} \sum_{t=1}^N \left| \frac{X_t - F_t}{X_t} (100) \right|$  dengan  $X_t$  = nilai data ke-t,  $F_t$  = nilai peramalan ke-t, dan  $N$  = banyaknya data. Sehingga untuk nilai MAPE dari hasil peramalan yang telah dilakukan ditunjukkan oleh tabel berikut:

Tabel 4.11: *Nilai MAPE pada Hasil Peramalan Data suhu Udara*

No.	Model	Nilai MAPE
1	ARIMA	2,20288828
2	KF Simulasi I	2,100066667
3	KF Simulasi II	1,98968
4	KF Simulasi III	1,935357143

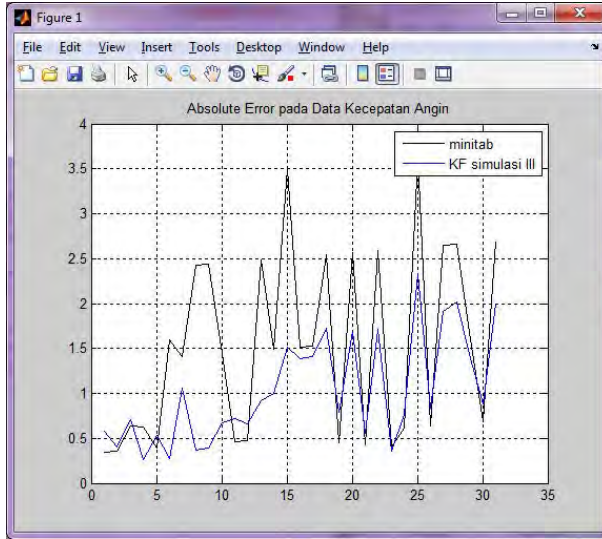
Sedangkan dapat juga dibandingkan dari gambar simulasi I, simulasi II, dan simulasi III pada data kecepatan angin maka yang simulasi III menunjukkan hasil prediksi yang lebih akurat dan mendekati data aslinya seperti yang ditunjukkan pada Gambar 4.24. Di samping itu juga dari gambar di bawah ini dapat dibandingkan nilai mutlak kesalahan peramalan baik untuk ARIMA dengan minitab dan Filter Kalman dengan hasil simulasi III pada data suhu udara menunjukkan bahwa nilai mutlak kesalahan peramalan untuk Filter Kalman simulasi III cenderung lebih kecil jika dibandingkan dengan yang lain.



Gambar 4.25: Nilai Mutlak Kesalahan Peramalan pada Data Suhu Udara

Dapat juga kita amati nilai tengah kesalahan persentase mutlak hasil peramalan pada data kecepatan angin bahwa pada Filter Kalman simulasi III memiliki nilai persentase yang lebih kecil jika dibandingkan dengan yang lain seperti pada Gambar 4.26 yang artinya dapat dikatakan hasil peramalan setelah diterapkan Filter Kalman dapat meminimumkan error yang didapat dari hasil peramalan ARIMA pada minitab.





Gambar 4.26: Nilai Mutlak Kesalahan Peramalan pada Data Kecepatan Angin

Selain itu, dapat diketahui juga diketahui akurasi peramalan yang ada dengan melihat nilai MAPE dari hasil peramalan yang telah dilakukan seperti yang ditunjukkan oleh tabel berikut:

Tabel 4.12: Nilai MAPE pada Hasil Peramalan Data Kecepatan Angin

No.	Model	Nilai MAPE
1	ARIMA	28,90889689
2	KF Simulasi I	19,0153
3	KF Simulasi II	19,30076
4	KF Simulasi III	18,73821429

## BAB V PENUTUP

Dari analisis dan pembahasan yang sudah dilakukan, dapat ditarik kesimpulan serta diberikan saran untuk pengembangan dan perbaikan penelitian ini.

### 5.1 Kesimpulan

Dari seluruh proses yang telah dibahas di atas dapat disimpulkan bahwa model data adalah ARIMA (1,1,1) untuk suhu udara dan ARIMA (0,1,1) untuk kecepatan angin. Selain itu, penggunaan algoritma Filter Kalman mempunyai pengaruh baik terhadap perbaikan hasil prediksi terbukti dari nilai MAPE pada data suhu udara dengan ARIMA adalah 2,20288828 sedangkan nilai MAPE dengan Filter Kalman mempunyai nilai yang lebih kecil yaitu 1,935357143. Hal yang sama juga berlaku pada data kecepatan angin dimana penggunaan algoritma Filter Kalman mempunyai pengaruh baik terhadap perbaikan hasil prediksi terbukti dari nilai MAPE dengan ARIMA adalah 28,90889689 sedangkan nilai MAPE dengan Filter Kalman mempunyai nilai yang lebih kecil yaitu 18,73821429.

### 5.2 Saran

Saran yang diberikan dari Tugas Akhir ini untuk penelitian selanjutnya adalah nantinya dapat digunakan metode Jaringan Saraf Tiruan dalam melakukan prediksi yang menggantikan penggunaan metode ARIMA.



*"Halaman ini sengaja dikosongkan"*

## DAFTAR PUSTAKA

- [1] Anonim. (2013). 10 Negara Dengan Jumlah Penduduk Terbanyak di Dunia. {<http://suhadahanum.blogspot.com/2013/09/10-negara-dengan-jumlah-penduduk.html>}. **diakses tanggal 27 Agustus 2013.**
- [2] Anonim. (2013). 10 Kota Terpadat di Indonesia. {<http://pengetahuanpintars.com/2013/11/10-kota-terpadat-di-indonesia.html>}. **diakses tanggal 29 Agustus 2013.**
- [3] Anonim. (2013). Penerbangan Juanda dihentikan akibat cuaca buruk. {<http://antaranews.com/2013/12/penerbangan-juanda-dihentikan-akibat-cuaca-buruk.html>}. **diakses tanggal 12 September 2013.**
- [4] Anonim. (2013). Cuaca. {<http://wikipedia.com/2013/09/cuaca.html>}. **diakses tanggal 29 Agustus 2013.**
- [5] Zakir, A. *Analisis dan Pengembangan Sistem Metode Prakiraan Cuaca di Bidang Informasi Meteorologi. Tugas Akhir, Universitas Gunadarma* : Jakarta.
- [6] Wei, W. S. (1990). *Time Analysis Univariate and Multivariate Methods*. Addison Wesley Publishing Company, Inc. : Amerika.

- [7] Makridakis, McGee, dan Wheelwright, W. (1999). *Metode dan Aplikasi Peramalan*. Edisi kedua. Bina Rupa Aksara : Jakarta.
- [8] Panchahayani, S. (2011). *Estimasi Lintasan Misil Dengan Metode Ensemble Kalman Filter*. **Tugas Akhir, Jurusan Matematika, ITS** : Surabaya.
- [9] Galanis, G., Louka P., Katsafados, P., Kallos, G., dan Phytharoulis, I. (2006). *Application of Kalman Filter Based On Non-Linear Function to Numerical Weather Prediction*. Copernicus GmbH : Yunani.
- [10] Hanke, J. E. dan Wichern, D.W. (2005). *Business Forecasting Eight Edition*. Prentice Hall : United States of America.
- [11] Fadlilah, A. (2012). *Perbandingan Metode Extended Kalman Filter (EKF) dan Ensemble Kalman Filter (EnKF) untuk Mendeteksi Waktu Terjadinya Kerak Pada Alat Penukar Panas*, **Tugas Akhir Jurusan Matematika, Institut Teknologi Sepuluh Nopember. Surabaya.**
- [12] Daniel, W. (1989). *Statistika Non Parametrik Terapan*. PT. Gramedia : Jakarta.
- [13] Welch, G. dan Bishop, G. (2001). *An Introduction to the Kalman Filter*. University of North Carolina : Chapel Hill : Amerika.

## BIODATA PENULIS



### **Tomy Kurniawan,**

kelahiran kota Surabaya, 21 Mei 1992, merupakan anak ketiga dari tiga bersaudara dari pasangan Lim Agustono dan Suyani. Menempuh pendidikan dimulai dari TK 0 besar Darmawanita Taman, SDN Geluran 2 Taman, SMPN 1 Taman kemudian SMAN 1 Taman. Dan melanjutkan studi ke Jurusan Matematika

ITS melalui jalur PMDK Reguler dan diterima sebagai mahasiswa angkatan 2010 dengan NRP 1210100030 serta mengambil rumpun permodelan matematika. Alamat email: [tomkur36@yahoo.com](mailto:tomkur36@yahoo.com).

# LAMPIRAN

*"Halaman ini sengaja dikosongkan"*



**LAMPIRAN I**  
**Listing Program KF pada Suhu Udara**

```
clc
clear all
disp(' ');
disp('||=====||');
disp('||-----||');
disp('||                               ||');
disp('||           SIMULASI PROGRAM           ||');
disp('||     PENERAPAN METODE KALMAN FILTER     ||');
disp('||           PADA MODEL TIME SERIES           ||');
disp('||                               ||');
disp('||           oleh : TOMY KURNIAWAN           ||');
disp('||           NRP  : 1210 100 030             ||');
disp('||                               ||');
disp('||     bersama : Drs. Lukman Hanafi, M.Sc    ||');
disp('||           Dr. Erna Apriliani, M.Si      ||');
disp('||                               ||');
disp('||-----||');
disp('||=====||');
disp(' ');
M = [1 0;0 1]; %M=A matriks sistem
N = input('Masukkan banyak data yang diinginkan (max 120
data) : ');
Q=0.001;
R=1;
Qk=[1 0;0 1]*Q;
Rk=R;
```

```

%inisialisasi variabel keadaan
a00 = input('Masukkan nilai awal a00 : ');
a10 = input('Masukkan nilai awal a10 : ');
xcapil(:,1)=[a00 a10];

% H=[1 0;0 1];
%data suhu dari bmkg
s=xlsread('data bmkg2.xlsx',4);

%data bias suhu = data_bmkg - for_minibab
t=xlsread('data bmkg2.xlsx',5);

%data for_suhu_minibab
m=xlsread('data bmkg2.xlsx',6);

H=[ones(length(s),1),s];

%prediksi & koreksi
for i=1:N
    wk(1,i) = normrnd(0,sqrt(Q));
    wk(2,i) = normrnd(0,sqrt(Q));

    vk(1,i) = normrnd(0,sqrt(R));
    vk(2,i) = normrnd(0,sqrt(R));

    xf(:,i)=M*xcapil(:,i)+wk(:,i);
    p(:,1)=[4 0];
    p(:,2)=[0 4];
    pcapil=[p(1,2*i-1) p(1,2*i);p(2,2*i-1) p(2,2*i)];
    pf=M*pcapil*M'+ Qk;
    pcapil=pf-(pf*H(i,:)'*inv((H(i,:)*pf*H(i,:)' +Rk))
    *H(i,:)*pf);
    p(:,2*i+1)=pcapil(:,1);

```

```

p(:,2*i+2)=pcapil(:,2);
xcapil(:,i+1)=xf(:,i)+pf*H(i,:)'*inv((H(i,:)*pf
*H(i,:)' + Rk))*(t(i,:)-(H(i,:)*xf(:,i)));
end

```

```

hasil = strcat('nilai a0 = ',num2str(xcapil(1,N)),',
dan a1 = ',num2str(xcapil(2,N)))

```

```

%plot nilai a0 dan a1
figure(1)
set(plot(xcapil(1,:)), 'color', 'black')
hold on
set(plot(xcapil(2,:)), 'color', 'red')
grid on
    title('Estimasi Koefisien polinomial');
    xlabel('waktu ke-');
    ylabel('nilai koefisien');
legend('a0', 'a1');

```

```

%=====

```

```

figure(2)
for k=1:N
bias(k) = xcapil(1,k+1) + xcapil(2,k+1)*s(k);
sf(k) = bias(k) + s(k);
ape(k) = (abs(s(k) - sf(k)) / s(k))*100; %absolute percentag
sape(1) = 0;
sape(k+1) = ape(k) + sape(k); %sigma dari ape
end
mape = sape(k+1)/N;
hasil2 = strcat('Nilai MAPE = ',num2str(mape))

```

```

%plot data, for_minitab, for_KF

```

```
set(plot(s),'color','black')
hold on
set(plot(m),'color','blue')
hold on
set(plot(sf),'color','red')
grid on
    title('Estimasi Suhu Udara');
    xlabel('waktu ke-');
    ylabel('nilai Suhu Udara');
legend('data','minitab','kalman filter');
```

## LAMPIRAN II

### Listing Program KF pada Kecepatan Angin

```

clc
clear all
disp(' ');
disp('||=====||');
disp('||-----||');
disp('||                               ||');
disp('||           SIMULASI PROGRAM           ||');
disp('||           PENERAPAN METODE KALMAN FILTER           ||');
disp('||           PADA MODEL TIME SERIES           ||');
disp('||                               ||');
disp('||           oleh : TOMY KURNIAWAN           ||');
disp('||           NRP  : 1210 100 030           ||');
disp('||                               ||');
disp('||           bersama : Drs. Lukman Hanafi, M.Sc ||');
disp('||           Dr. Erna Apriliani, M.Si ||');
disp('||                               ||');
disp('||-----||');
disp('||=====||');
disp(' ');
M = [1 0;0 1]; %M=A matriks sistem
N = input('Masukkan banyak data yang diinginkan (max 120
data) : ');
Q=0.001;
R=1;
Qk=[1 0;0 1]*Q;
Rk=R;

```

```

%inisialisasi variabel keadaan
a00 = input('Masukkan nilai awal a00 : ');
a10 = input('Masukkan nilai awal a10 : ');
xcapil(:,1)=[a00 a10];

% H=[1 0;0 1];
%data kec_angin dari bmkg
s=xlsread('data bmkg2.xlsx',7);

%data bias kec_angin = data_bmkg - for_minitab
t=xlsread('data bmkg2.xlsx',8);

%data for_kec_angin_minitab
m=xlsread('data bmkg2.xlsx',9);

H=[ones(length(s),1),s];

%prediksi & koreksi
for i=1:N
    wk(1,:) = normrnd(0,sqrt(Q),1,N);
    wk(2,:) = normrnd(0,sqrt(Q),1,N);

    vk(1,:) = normrnd(0,sqrt(R),1,N);
    vk(2,:) = normrnd(0,sqrt(R),1,N);

    xf(:,i)=M*xcapil(:,i)+wk(:,i);
    p(:,1)=[4 0];
    p(:,2)=[0 4];
    pcapil=[p(1,2*i-1) p(1,2*i);p(2,2*i-1) p(2,2*i)];
    pf=M*pcapil*M'+ Qk;
    pcapil=pf-(pf*H(i,:)'*inv((H(i,:)*pf*H(i,:)' +Rk))
        *H(i,:)*pf);
    p(:,2*i+1)=pcapil(:,1);

```

```

p(:,2*i+2)=pcapil(:,2);
xcapil(:,i+1)=xf(:,i)+pf*H(i,:)’*inv((H(i,:)’*pf
      *H(i,:)’+Rk))*(t(i,:)-(H(i,:)’*xf(:,i)));
end

```

```

hasil = strcat(’nilai a0 = ’,num2str(xcapil(1,N)),’
dan a1 = ’,num2str(xcapil(2,N)))

```

```

%plot nilai a0 dan a1
figure(1)
set(plot(xcapil(1,:)),’color’,’black’)
hold on
set(plot(xcapil(2,:)),’color’,’red’)
grid on
      title(’Estimasi Koefisien polinomial’);
      xlabel(’waktu ke-’);
      ylabel(’nilai koefisien’);
legend(’a0’,’a1’);

```

```

%=====

```

```

figure(2)
for k=1:N
bias(k) = xcapil(1,k+1) + xcapil(2,k+1)*s(k);
sf(k) = bias(k) + s(k);
ape(k) = (abs(s(k) - sf(k)) / s(k))*100; %absolute percentag
sape(1) = 0;
sape(k+1) = ape(k) + sape(k); %sigma dari ape
end
mape = sape(k+1)/N;
hasil2 = strcat(’Nilai MAPE = ’,num2str(mape))

```

```

%plot data, for_minitab, for_KF

```

```
set(plot(s),'color','black')
hold on
set(plot(m),'color','blue')
hold on
set(plot(sf),'color','red')
grid on
    title('Estimasi Kecepatan Angin');
    xlabel('waktu ke-');
    ylabel('nilai Kecepatan Angin');
legend('data','minitab','kalman filter');
```



### LAMPIRAN III

#### Listing Program Plot Absolute Error pada Data Suhu Udara

```
a=xlsread('error.xlsx',1,'D2:D121')
b=xlsread('error.xlsx',1,'E2:E121')
c=xlsread('error.xlsx',1,'G2:G121')
d=xlsread('error.xlsx',1,'I2:I121')
e=xlsread('error.xlsx',1,'K2:K121')
f=xlsread('error.xlsx',1,'M2:M121')
set(plot(a),'color','blue')
hold on
set(plot(b),'color','magenta')
set(plot(c),'color','green')
set(plot(d),'color','red')
set(plot(e),'color','yellow')
set(plot(f),'color','black')
legend('minitab','KF 60xiterasi','KF 75xiterasi','KF
90xiterasi','KF 105xiterasi','KF 120xiterasi')
grid on
title('absolute error pada data suhu udara')
```

*"Halaman ini sengaja dikosongkan"*

## LAMPIRAN IV

### Listing Program Plot Absolute Error pada Data Kecepatan Angin

```
a=xlsread('error.xlsx',2,'C2:C121')
b=xlsread('error.xlsx',2,'E2:E121')
c=xlsread('error.xlsx',2,'G2:G121')
d=xlsread('error.xlsx',2,'I2:I121')
e=xlsread('error.xlsx',2,'K2:K121')
f=xlsread('error.xlsx',2,'M2:M121')
set(plot(a),'color','blue')
hold on
set(plot(b),'color','magenta')
set(plot(c),'color','green')
set(plot(d),'color','red')
set(plot(e),'color','yellow')
set(plot(f),'color','black')
legend('minitab','KF 60xiterasi','KF 75xiterasi','KF
90xiterasi','KF 105xiterasi','KF 120xiterasi')
grid on
title('absolute error pada data kecepatan angin')
```