



**ENSEMBLE MODEL ARIMA DAN GSTAR PADA PERAMALAN
CURAH HUJAN MENGGUNAKAN *KALMAN FILTER***

TESIS

Oleh

**Unik Novita Wulandari
NIM: 171820101003**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2020**



**ENSEMBLE MODEL ARIMA DAN GSTAR PADA PERAMALAN
CURAH HUJAN MENGGUNAKAN *KALMAN FILTER***

TESIS

Diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
Untuk menyelesaikan Program Studi Magister Matematika (S2)
Dan Mencapai gelar Magister Sains

Oleh

**Unik Novita Wulandari
NIM: 171820101003**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2020**

PERSEMBAHAN

Dengan menyebut nama Allah Yang Maha Pengasih lagi Penyayang, serta sholawat atas nabi Muhammad SAW, tesis ini saya persembahkan untuk:

1. Ayahanda Sarjuni dan Ibunda Siti Indun yang selalu mendo'akan serta memberikan dukungan, semangat dan motivasi demi terselesaikannya tesis ini.
2. Almarhum eyang kakung Riono dan Eyang putri Tarbingatun yang selalu memberikan semangat disetiap kesulitan yang ada.
3. Suami tercinta Yogi Dian Praditya yang telah memberikan tenaga dan semangat dalam menyelesaikan tugas akhir ini.
4. Adik tersayang Ninis Febrian Yuliansya yang selalu memberikan semangat dan solusi pada setiap permasalahan.
5. Dr. Alfian Futuhul Hadi, S.Si., M.Si selaku Dosen Pembimbing Utama yang telah meluangkan waktu, pikiran dan perhatiannya dalam penulisan tugas akhir ini.
6. Prof. Drs. I Made Tirta, M.Sc., Ph.D dan Dr. Mohamad Fatekurohman, S.Si., M.Si selaku Dosen penguji yang telah memberi kritik dan saran demi kesempurnaan tugas akhir ini.
7. Seluruh dosen Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember yang membimbing penulis selama masa kuliah.
8. Almamater Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.
9. semua pihak yang tidak dapat disebutkan satu per satu.

MOTTO

..... Sesungguhnya Allah tidak mengubah keadaan suatu kaum sehingga mereka mengubah keadaan yang ada pada diri mereka sendiri

(terjemahan surat *Ar-Ra'd* ayat 11)*)



*) Departemen Agama Republik Indonesia. 2005. *Al Quran dan Terjemahannya*. Bandung: CV. Jumanatul Ali Art.

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Unik Novita Wulandari

NIM : 171820101003

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Ensemble Model ARIMA dan GSTAR pada Peramalan Curah Hujan Menggunakan *Kalman Filter*” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi manapun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak mana pun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, Mei 2020

Yang menyatakan,

Unik Novita Wulandari
NIM 1718201010032

TESIS

**ENSEMBLE MODEL ARIMA DAN GSTAR PADA PERAMALAN
CURAH HUJAN MENGGUNAKAN *KALMAN FILTER***

Oleh

Unik Novita Wulandari
171820101003

Pembimbing

Dosen Pembimbing Utama : Dr. Alfian Futuhul Hadi, S.Si., M.Si

PENGESAHAN

Skripsi yang berjudul “Ensemble Model ARIMA dan GSTAR Pada Peramalan Curah Hujan Menggunakan *Kalman Filter*” telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Jember

Tim Penguji:

Dosen Pembimbing Utama,

Dr. Alfian Futuhul Hadi, S.Si., M.Si
NIP. 197407192000121001

Dosen Penguji I,

Penguji II,

Prof. Drs. I Made Tirta, M.Sc., Ph.D
NIP. 195912201985031002

Dr. Mohamad Fatekurohman, S.Si., M.Si
NIP. 196906061998031001

Mengesahkan

Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

Universitas Jember

Drs. Sujito, Ph.D.
NIP. 196102041987111001

RINGKASAN

Ensemble Model ARIMA dan GSTAR pada Peramalan Curah Hujan Menggunakan Kalman Filter; Unik Novita Wulandari, 171820101003; 2020: 53 halaman; Program Studi Magister Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Jember.

Kabupaten Jember merupakan salah satu kabupaten yang memiliki potensi besar dalam bidang pertanian dimana curah hujan sangat berpengaruh terhadap hasil pertanian di daerah tersebut. Oleh karena itu, perlu dilakukan peramalan curah hujan sebagai informasi dasar dalam menentukan indeks kekeringan. Model ARIMA dan GSTAR adalah model yang digunakan dalam peramalan curah hujan. Pada model ARIMA dan GSTAR dilakukan pengklusteran pada data curah hujan untuk membuat model peramalan. Selain ARIMA dan GSTAR, metode yang sering digunakan adalah *Kalman Filter*. Metode *Kalman Filter* memiliki keunggulan yaitu peramalan dapat diperbaharui dengan data terbaru sehingga nilai prediksi selalu update. *Kalman Filter* biasanya digunakan untuk memperbaiki peramalan dengan menggunakan model ARIMA. Untuk mendapatkan hasil yang maksimal akan dilakukan penggabungan model ARIMA dan GSTAR dengan menggunakan metode Super-Ensemble *Kalman Filter* dalam meramalkan curah hujan di Kabupaten Jember. Data yang digunakan dalam penelitian ini adalah data curah hujan Kabupaten Jember periode Januari 2005 sampai Desember 2017. Selain itu, dalam penelitian ini yang digunakan sebagai *fitting* model dalam metode Super-Ensemble *Kalman Filter* adalah model ARIMA periode Januari 2005-Desember 2015 dan GSTAR periode Januari 2005-Desember 2015. Selanjutnya hasil peramalan antara metode Super-Ensemble *Kalman Filter* yang menggunakan *fitting* model ARIMA (A1) periode Januari 2005-Desember 2015 dan GSTAR periode Januari 2005-Desember 2015 akan dibandingkan dengan model ARIMA (A2) periode Januari 2005-Desember 2016.

Dalam penelitian ini data yang digunakan dibagi menjadi empat kelompok berdasarkan pengelompokan dari BPS dan pengelompokan dengan menggunakan algoritma *K-Means*. Berdasarkan pengelompokan ini, nantinya akan diperoleh model peramalan terbaik dari masing-masing kelompok. Model tersebut yang nantinya akan digunakan sebagai *fitting* model pada peramalan Super-Ensemble *Kalman Filter*. Pada model ARIMA dan GSTAR data yang digunakan dalam peramalan harus stasioner dalam mean dan varians, dimana pola data tidak mengalami perubahan yang signifikan.

Model ARIMA yang diperoleh dari masing-masing kluster akan dipilih model terbaik dan diperoleh Y_1 , Y_2 , Y_3 dan Y_4 yang menunjukkan model ARIMA (A1) terbaik pada kluster 1, kluster 2, kluster 3 dan kluster 4. Model terbaik ARIMA (A1) pada kluster pertama (4,0,4), kluster kedua (3,0,1), kluster ketiga (2,0,3) dan kluster keempat (2,0,5). Model ini yang nantinya akan digunakan sebagai *fitting* model dalam *Kalman Filter*. Sedangkan untuk Model ARIMA (A2) nantinya akan digunakan sebagai pembandingan hasil peramalan dengan Super-Ensemble *Kalman Filter*. Berdasarkan nilai AIC diperoleh model terbaik untuk ARIMA (A2) pada kluster 1 adalah ARIMA (4,0,4), Kluster 2 ARIMA (4,0,2), kluster 3 ARIMA (2,0,3) dan pada kluster 4 ARIMA (3,0,2). Sedangkan untuk model GSTAR pembagian kluster dilakukan dengan menggunakan algoritma *K-Means*. Dari penelitian sebetulnya diperoleh empat model di tiap klusternya.

Pada metode Super-Ensemble *Kalman Filter* ini akan menggabungkan model ARIMA dan GSTAR untuk melakukan peramalan curah hujan pada suatu periode waktu tertentu di Kabupaten Jember yang telah dibagi menjadi empat wilayah. Langkah awal pada metode Super-Ensemble *Kalman Filter* adalah dengan menentukan nilai awal atau dalam komputasi disebut dengan iterasi ke-0, dimana iterasi ke-0 adalah kondisi yang diperlukan untuk menentukan kondisi pada tahap berikutnya. Penentuan nilai awal pada metode Super-Ensemble *Kalman Filter* berupa asumsi suatu nilai yaitu nilai rata-rata maupun prediksi awal. Penentuan nilai awal dilakukan pada nilai dari bobot vektor, dimana nilai awal yang digunakan adalah nilai random.

Pada metode Super-Ensemble *Kalman Filter* ini, proses *Kalman Filter* hanya dilakukan pada bobot vektor waktu saja untuk memperoleh nilai bobot vektor waktu selama 12 kali data pengamatan. Selanjutnya dilakukan proses penyesuaian hasil estimasi dengan data pengamatan dimana nilai tersebut merupakan nilai sebenarnya sehingga akan dihasilkan perbaikan estimasi. Proses perbaikan estimasi ini dalam *Kalman Filter* biasa disebut dengan *Kalman Gain*. *Kalman Gain* berperan dalam meminimalisasi *norm kovarian error* pada proses estimasi. Pada bagian ini diperoleh nilai *norm kovarians error* semakin mengecil hingga iterasi terakhir. Selain itu juga dilakukan peramalan ARIMA (A2). Berdasarkan hasil peramalan dari keduanya, diperoleh kesimpulan bahwa peramalan menggunakan metode Kalman Filter menghasilkan nilai *Root Mean Square Error* (RMSE) yang lebih kecil dibandingkan dengan peramalan menggunakan model A2. Hal ini menunjukkan bahwa metode Super-Ensemble *Kalman Filter* lebih baik daripada model ARIMA (A2) untuk peramalan curah hujan di Kabupaten Jember. Diperoleh kesimpulan bahwa super-Ensemble *Kalman Filter* dapat digunakan sebagai metode peramalan curah hujan di Kabupaten Jember dengan menggunakan model ARIMA dan GSTAR sebagai *fitting* model.

KATA PENGANTAR

Puji syukur kehadiran Allah SWT karena atas segala rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan tesis yang berjudul “Ensemble Model ARIMA dan GSTAR pada Peramalan Curah Hujan Menggunakan *Kalman Filter*”. Tesis ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan strata dua (S2) pada Jurusan Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Penulis menyadari bahwa dalam penyusunan sampai terselesaikannya tesis ini tidak lepas dari bantuan berbagai pihak. Karenanya penulis ingin mengucapkan banyak terima kasih kepada:

1. Dr. Alfian Futuhul Hadi, S.Si., M.Si selaku Dosen Pembimbing Utama yang telah meluangkan waktu, pikiran dan perhatiannya dalam penulisan tugas akhir ini;
2. Prof. Drs. I Made Tirta, M.Sc., Ph.D dan Dr. Mohamad Fatekurohman, S.Si., M.Si selaku Dosen penguji yang telah memberi kritik dan saran demi kesempurnaan tugas akhir ini;
3. Seluruh staf pengajar Jurusan Matematika Fakultas MIPA Universitas Jember yang telah memberikan ilmu serta bimbingannya sehingga penulis dapat menyelesaikan tesis ini;
4. Keluarga Besar Mahasiswa Magister Matematika 2016 yang telah memberikan semangat dan bantuan dalam proses penulisan tesis ini;
5. semua pihak yang tidak dapat disebutkan satu per satu.

Karena tesis ini masih jauh dari sempurna, kritik dan saran yang sifatnya membangun sangat penulis harapkan. Akhirnya penulis berharap, semoga tesis ini dapat memberi manfaat dan sumbangan bagi ilmu pengetahuan.

Jember, Mei 2020

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTTO	iii
HALAMAN PERNYATAAN.....	iv
HALAMAN PEMBIMBING	v
HALAMAN PENGESAHAN.....	vi
RINGKASAN	vii
KATA PENGANTAR.....	ix
DAFTAR ISI	x
DAFTAR GAMBAR.....	xii
DAFTAR TABEL	xiii
DAFTAR LAMPIRAN	xv
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan	3
1.4 Manfaat	4
BAB 2. TINJAUAN PUSTAKA	5
2.1 Analisis Time Series	5
2.2 Stasioneritas Data.....	6
2.2.1 Proses Differencing.....	7
2.2.2 Transformasi Box-Cox.....	8
2.3 Autocorelation Function (ACF) dan Partial Autocorelation Function (PACF)	9
2.4 Model Autoregressive Integrated Moving Average (ARIMA).....	11
2.5 Matrix Autocorrelation Function (MACF)	15

2.6 Matrix Partial Autocorrelation Function (MPACF)	16
2.7 Generalized Space-Time Autoregressive (GSTAR)	17
2.7.1 Pemilihan Bobot Lokasi Model GSTAR	18
2.8 Kalman Filter	18
2.9 Super – Ensemble Kalman Filter (SE)	21
2.10 Norm Kovarians Error.....	22
2.11 Root Mean Square Error (RMSE).....	23
BAB 3. METODE PENELITIAN.....	24
BAB 4. HASIL DAN PEMBAHASAN.....	29
4.1 Model ARIMA 2005-2015 (A1) pada Data Curah Hujan	29
4.1.1 Uji Kestasioneran Data	30
4.1.2 Model ARIMA Data Curah Hujan pada Kluster 1.....	31
4.1.3 Model ARIMA Data Curah Hujan pada Kluster 2.....	33
4.1.4 Model ARIMA Data Curah Hujan pada Kluster 3.....	34
4.1.5 Model ARIMA Data Curah Hujan pada Kluster 4.....	35
4.2 Model ARIMA 2005-2016 (A2) pada Data Curah Hujan	37
4.3 Model GSTAR pada Data Curah Hujan	37
4.4 Menentukan Model Super-Ensemble Kalman Filter	38
4.4.1 Model Super-Ensemble Kalman Filter pada kluster 1	38
4.4.2 Model Super-Ensemble Kalman Filter pada kluster 2	42
4.4.3 Model Super-Ensemble Kalman Filter pada kluster 3	45
4.4.4 Model Super-Ensemble Kalman Filter pada kluster 4.....	49
4.5 Perbandingan Kalman Filter pada Empat Kluster.....	52
BAB 5. PENUTUP.....	54
5.1 Kesimpulan	54
5.2 Saran.....	54
DAFTAR PUSTAKA	54

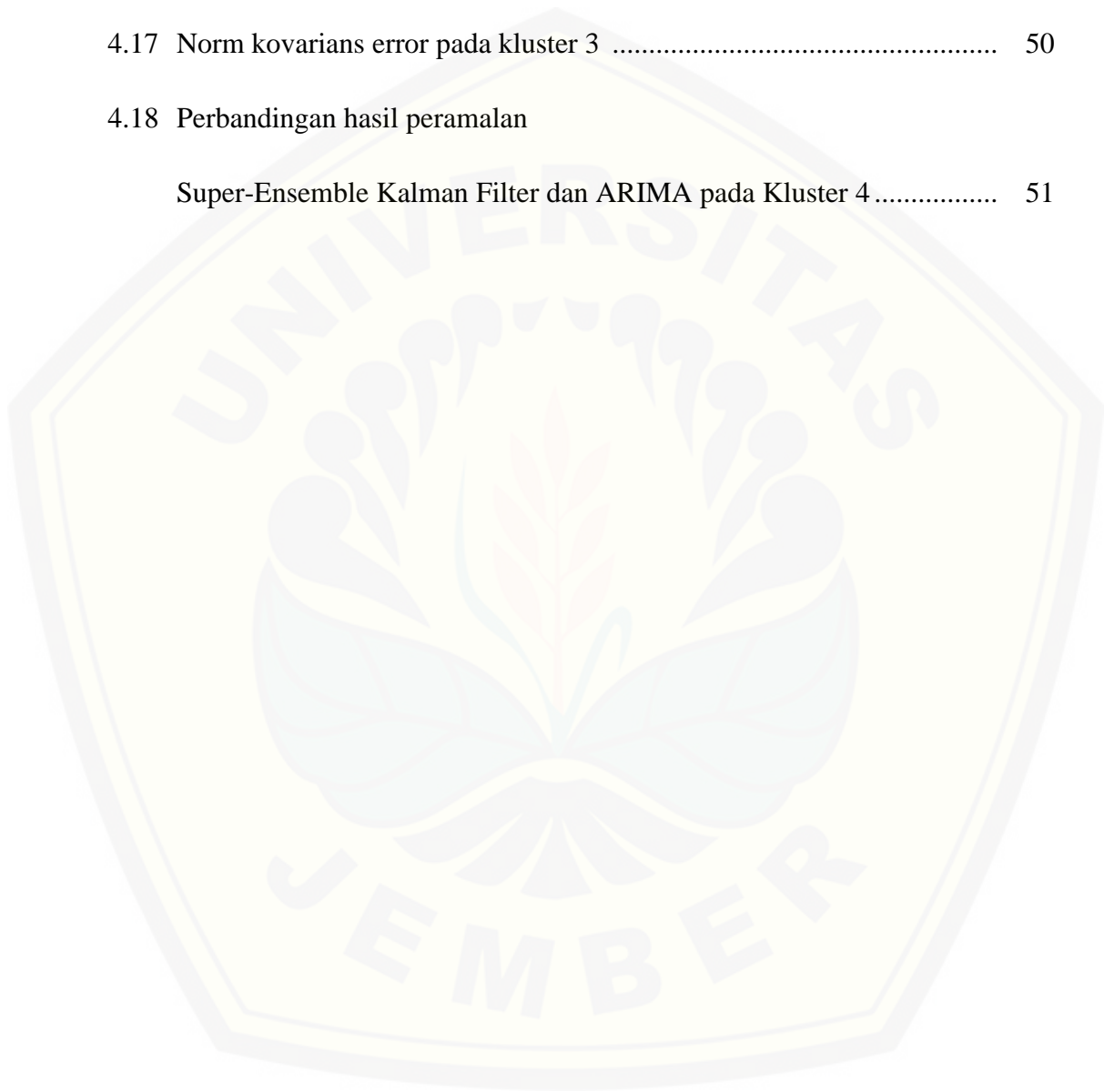
DAFTAR GAMBAR

	Halaman
4.1 Plot Data Curah Hujan di Empat Wilayah Kabupaten Jember Tahun 2005-2017	30
4.2 Grafik norm kovarians <i>error</i> pada tahap prediksi data curah hujan pada Kluster 1	40
4.3 Perbandingan metode Super-Ensemble <i>Kalman Filter</i> dan model ARIMA pada Kluster 1	42
4.4 Grafik norm kovarians <i>error</i> pada tahap prediksi data curah hujan pada Kluster 2	43
4.5 Perbandingan metode Super-Ensemble <i>Kalman Filter</i> dan model ARIMA pada Kluster 2	45
4.6 Grafik norm kovarians <i>error</i> pada tahap prediksi data curah hujan pada Kluster 3	46
4.7 Perbandingan metode Super-Ensemble <i>Kalman Filter</i> dan model ARIMA pada kluster 3	48
4.8 Grafik norm kovarians <i>error</i> pada tahap prediksi data curah hujan pada Kluster 4	50
4.9 Perbandingan metode Super-Ensemble <i>Kalman Filter</i> dan model ARIMA pada kluster 4	52

DAFTAR TABEL

	Halaman
2.1 Nilai λ dan transform	7
4.1 Hasil Uji ADF dan parameter λ	30
4.2 Estimasi Parameter Model Sementara pada Kluster 1	31
4.3 Nilai AIC Model pada Kluster 1	32
4.4 Estimasi Parameter Model Sementara pada Kluster 2	33
4.5 Nilai AIC Model pada Kluster 2	34
4.6 Estimasi Parameter Model Sementara pada Kluster 3	34
4.7 Nilai AIC Model pada Kluster 3	35
4.8 Estimasi Parameter Model Sementara pada Kluster 4	35
4.9 Nilai AIC Model pada Kluster 4	36
4.10 Nilai AIC ARIMA (A2) pada Tiap Kluster	37
4.11 Norm kovarians error pada kluster 1	40
4.12 Perbandingan hasil peramalan Super-Ensemble Kalman Filter dan ARIMA pada Kluster 1	41
4.13 Norm kovarians error pada kluster 2	43
4.14 Perbandingan hasil peramalan Super-Ensemble Kalman Filter dan ARIMA pada Kluster 2	44

4.15	Norm kovarians error pada kluster 3	47
4.16	Perbandingan hasil peramalan	
	Super-Ensemble Kalman Filter dan ARIMA pada Kluster 3	47
4.17	Norm kovarians error pada kluster 3	50
4.18	Perbandingan hasil peramalan	
	Super-Ensemble Kalman Filter dan ARIMA pada Kluster 4	51



DAFTAR LAMPIRAN

	Halaman
Lampiran A : Skrip Program R	57
A.1 Pemodelan ARIMA pada Kluster 1	57
A.2 Pemodelan ARIMA pada Kluster 2	57
A.3 Pemodelan ARIMA pada Kluster 3	58
A.1 Pemodelan ARIMA pada Kluster 4	58
Lampiran B : Nilai AIC ARIMA (A1) pada Keempat Kluster	59

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Bagi Indonesia yang penduduknya mayoritas berkerja dalam bidang agraris, faktor iklim benar-benar dijadikan salah satu pertimbangan dalam penentuan kecocokan jenis tanaman yang akan dibudidayakan dan dikembangkan pada suatu daerah.

Curah hujan merupakan salah satu faktor iklim yang penting dalam bidang pertanian, karena curah hujan menentukan ketersediaan air bagi tanaman pada suatu daerah. Selain curah hujan, masalah kekeringan juga berpengaruh dalam bidang pertanian. Kekeringan merupakan keadaan dimana suatu wilayah secara terus-menerus mengalami curah hujan di bawah rata-rata. Peramalan yang dilakukan selama ini dinilai masih kurang akurat karena masih terdapat peramalan yang meleset dari keadaan yang sebenarnya. Kondisi cuaca di Indonesia sulit diprediksi karena kondisi geografis yang bervariasi antara pegunungan, pantai dan daratan. Oleh karena itu, diperlukan metode peramalan yang sesuai dengan kondisi geografis disuatu daerah agar meminimalkan kerugian bagi para petani.

Peramalan (*forecasting*) adalah suatu proses perkiraan keadaan di masa yang akan datang dengan menggunakan data pada masa lampau. Peramalan adalah suatu kegiatan yang bertujuan untuk mengetahui nilai variabel dependen pada masa yang akan datang dengan mempelajari variabel independen pada masa lampau, yaitu dengan menganalisis pola data. Salah satu model yang sering digunakan dalam metode peramalan adalah model *time series*. Model *time series* merupakan model peramalan dengan memperhatikan data -data sebelumnya secara berurutan berdasarkan waktu dengan interval yang sama, bisa harian, mingguan, bulanan atau tahunan. Unsur ketergantungan waktu dan lokasi yang digabungkan pada suatu *time series* peubah ganda merupakan model ruang waktu (*space time*) dan merupakan salah satu model multivariat.

Kabupaten Jember merupakan salah satu kabupaten yang memiliki potensi besar dalam bidang pertanian dimana curah hujan sangat berpengaruh terhadap hasil pertanian di daerah tersebut. Data curah hujan sangat menarik untuk dikaji karena curah hujan merupakan salah satu faktor iklim yang penting dalam bidang pertanian, karena curah hujan dapat menentukan ketersediaan air bagi tanaman disuatu daerah. Oleh karena itu, perlu dilakukan peramalan curah hujan sebagai informasi dasar dalam menentukan indeks kekeringan.

Telah dilakukan beberapa penelitian mengenai peramalan curah hujan di Kabupaten Jember diantaranya Yudistira (2017) melakukan penelitian di Kabupaten Jember menggunakan model GSTAR dimana pada model ini dilakukan pengelompokkan dengan Algoritma *K-Means* sehingga menghasilkan pengelompokkan yang lebih baik dari pengelompokkan yang dilakukan oleh BPS. Pada tahun selanjutnya, Zulfi (2018) melakukan kembali penelitian mengenai peramalan curah hujan di Kabupaten Jember dengan menggunakan metode *Kalman Filter* yang menghasilkan bahwa metode *Kalman Filter* lebih baik daripada model ARIMA.

Beberapa peramalan tentang curah hujan dengan berbagai model telah dilakukan dalam suatu wilayah yang sama. Hasil dalam setiap peramalan mungkin berbeda satu sama lain dan untuk memilih yang terbaik merupakan hal yang sulit. Model ARIMA dan GSTAR adalah salah satu model yang digunakan dalam peramalan curah hujan. Pada model ARIMA dan GSTAR dilakukan pengklusteran pada data curah hujan untuk membuat model peramalan. Selain ARIMA dan GSTAR, metode yang sering digunakan adalah *Kalman Filter*. Metode *Kalman Filter* memiliki keunggulan yaitu peramalan dapat diperbaharui dengan data terbaru sehingga nilai prediksi selalu update. *Kalman Filter* biasanya digunakan untuk memperbaiki peramalan dengan menggunakan model ARIMA.

Berdasarkan hal tersebut peneliti tertarik untuk menggunakan metode Super-Ensemble *Kalman Filter* yang menggabungkan dua atau lebih hasil peramalan menggunakan metode *Kalman Filter* dalam meramalkan curah hujan di Kabupaten Jember untuk mendapatkan hasil yang maksimal. Data yang digunakan dalam penelitian ini adalah data curah hujan Kabupaten Jember periode Januari

2005 sampai Desember 2017. Selain itu, dalam penelitian ini yang digunakan sebagai *fitting* model dalam metode Super-Ensemble *Kalman Filter* adalah model ARIMA periode Januari 2005-Desember 2015 dan GSTAR periode Januari 2005-Desember 2015.

1.2 Rumusan Masalah

Model *time series* seperti ARIMA dan GSTAR adalah model yang dalam peramalan curah hujan hanya menggunakan model dasar saja, sedangkan dalam teknik Super-Ensemble *Kalman Filter* selain menggunakan model dasar juga menggunakan model pengukuran. Dalam teknik Super-Ensemble *Kalman Filter* juga dilakukan pengabungan dua atau lebih hasil peramalan terbaik. Berdasarkan hal tersebut, diperoleh rumusan masalah sebagai berikut

- a. Bagaimana penerapan metode Super-Ensemble *Kalman Filter* pada model ARIMA dan GSTAR?
- b. Bagaimana perbandingan hasil peramalan antara metode Super-Ensemble *Kalman Filter* yang menggunakan *fitting* model ARIMA periode Januari 2005-Desember 2015 dan GSTAR periode Januari 2005-Desember 2015 dengan model ARIMA periode Januari 2005-Desember 2016?

1.3 Tujuan

Tujuan dari penelitian ini adalah:

- a. Memperoleh hasil penerapan *Kalman Filter* pada model ARIMA dan GSTAR yang ditinjau dari nilai *error*nya.
- b. Membandingkan hasil peramalan antara metode Super-Ensemble *Kalman Filter* yang menggunakan *fitting* model ARIMA periode Januari 2005-Desember 2015 dan GSTAR periode Januari 2005-Desember 2015 dengan model ARIMA periode Januari 2005-Desember 2016.

1.4 Manfaat

Manfaat yang diharapkan dari penelitian ini adalah dapat memberikan informasi tentang peramalan kekeringan dan dapat mendukung program pertanian di Kabupaten Jember. Selain itu, dengan adanya penelitian ini diharapkan dapat menambah wawasan tentang metode dalam peramalan curah hujan.



BAB 2. TINJAUAN PUSTAKA

2.1 Analisis Time Series

Data *time series* merupakan sekumpulan data pengamatan yang diperoleh dari perhitungan dari waktu ke waktu. Data yang digunakan adalah data yang telah dicatat dalam jangka waktu tertentu misalnya tiap bulan, tiap akhir tahun, sepuluh tahun dan sebagainya. Contoh data *time series* adalah pertumbuhan ekonomi suatu negara pertahun, jumlah produksi minyak perbulan, indeks harga saham perhari.

Hal yang perlu diperhatikan pada peramalan data *time series* adalah galat (*error*). Hasil dari prediksi biasanya berbeda dengan data sesungguhnya, maka hal yang bisa dilakukan adalah meminimalkan galatnya. Untuk meramalkan data *time series* dibutuhkan teknik peramalan yang baik. Teknik peramalan dapat bermacam-macam tergantung pada pola data yang ada Menurut Makridakis *et al* (1992) ada empat macam tipe pola data yaitu:

a. Pola Data Horizontal

Pola data horizontal terjadi saat data observasi berfluktuasi di sekitaran suatu nilai konstan atau *mean* yang membentuk garis horizontal. Data ini disebut juga dengan data stasioner. Contoh suatu produk penjualannya tidak meningkat atau menurun selama waktu tertentu.

b. Pola Data *Trend*

Pola data *trend* terjadi bilamana data pengamatan mengalami kenaikan atau penurunan selama periode jangka panjang. Suatu data pengamatan yang mempunyai *trend* disebut data nonstasioner.

c. Pola Data Musiman

Pola data musiman terjadi bilamana suatu data dipengaruhi oleh faktor musiman. Pola data musiman dapat mempunyai pola musim yang berulang dari periode ke periode berikutnya. Misalnya pola yang berulang setiap bulan tertentu, tahun tertentu atau pada minggu tertentu. Contoh suplai bahan makanan tiap bulan.

d. Pola Data Siklis

Pola data siklis terjadi bilamana data dipengaruhi oleh fluktuasi ekonomi jangka panjang seperti yang berhubungan dengan siklus bisnis.

2.2 Stasioneritas Data

Stasioneritas berarti kondisi tidak terdapat perubahan yang drastis pada data. Sekelompok data dinyatakan stasioner jika nilai rata-rata dan varians dari data *time series* tersebut tidak mengalami perubahan secara sistematis sepanjang waktu atau dengan kata lain rata-rata dan variansnya konstan. Stasionerisasi data merupakan hal yang diperlukan dalam analisis regresi deret waktu karena dapat memperkecil kesalahan model, sehingga jika data tidak stasioner, maka akan dilakukan transformasi stasioneritas melalui proses diferensi, jika *trend*-nya linier, sedangkan jika tidak linier, maka transformasi linieritas *trend* melalui proses logaritma natural jika *trend*-nya eksponensial, dan proses pembobotan (penghalusan eksponensial sederhana) jika bentuknya yang lain, yang selanjutnya dilakukan proses diferensi pada data hasil proses linieritas. Penyebab tidak stasionernya suatu data adalah adanya autokorelasi. Bila data telah stasioner maka autokorelasi akan hilang dengan sendirinya, karena itu transformasi data untuk membuat data yang tidak stasioner menjadi stasioner sama dengan transformasi data untuk menghilangkan autokorelasi. Untuk menguji apakah data bersifat stasioner atau tidak, umumnya digunakan uji akar unit. Terdapat banyak uji akar unit, tetapi yang paling umum dan banyak dipakai adalah *Augmented Dickey Fuller* (ADF).

Pengujian *Augmented Dickey Fuller* digunakan pada suatu data *time series* yang tidak stasioner pada orde nol, $I(0)$, maka stasioneritas data dapat dicari menggunakan order berikutnya sehingga didapatkan tingkat stasioneritas pada order ke- n (*first difference*) atau $I(1)$, atau *second difference* atau $I(2)$, dan seterusnya. Perlu diperhatikan plot data yang akan diuji sebelum melakukan uji ADF. Dimana jika data yang diuji mengandung unsur tren, maka dalam mengujinya digunakan uji ADF tipe tren, dan jika data tidak mengandung unsur tren, dapat dilihat apakah rata-ratanya sama dengan nol. Jika rata-rata data tidak

sama dengan nol, maka dapat diuji menggunakan uji ADF tipe konstanta, sedangkan jika rata-rata data sama dengan nol, pengujian dilakukan dengan uji ADF tipe *null*.

Uji ini memiliki persamaan :

$$\Delta Y_t = \beta_1 + \beta_2 t + \delta Y_{t-1} + \gamma_i \sum_{i=1}^m \Delta Y_{t-i} + \varepsilon_t \quad (2.1)$$

dengan:

ΔY_t = *first difference* dari Y

β_1 = nilai konstan atau *intercept*

β_2 = koefisien regresi untuk trend

δ = koefisien regresi untuk *lag Y*

γ = koefisien regresi untuk *difference lag Y*

ε = *error*

m = lag

t = waktu

dengan hipotesis :

H_0 : $\delta = 0$ (Terdapat akar unit, variable Y tidak stasioner)

H_1 : $\delta \neq 0$ Tidak terdapat akar unit, variable Y stasioner)

Statistik Uji:

$$t_\delta = \frac{\hat{\delta} - \delta_0}{se(\hat{\delta})} \quad (2.2)$$

Jika nilai t_δ lebih besar dari nilai kritis ADF maka gagal tolak hipotesis nol, yang berarti bahwa data tidak stasioner. Tetapi jika nilai t_δ lebih kecil dari nilai kritis ADF maka hipotesis nol diterima, yang berarti data stasioner (Makridakis *et al*, 1992).

2.2.1 Proses *Differencing*

Differencing adalah suatu cara yang biasa digunakan untuk menyelesaikan masalah data yang tidak stasioner. Proses *differencing* dilakukan

untuk beberapa periode sampai data stasioner, yaitu dengan cara mengurangi suatu data dengan data sebelumnya. Proses *differencing* dilakukan jika data tidak stasioner pada mean. Jika Y_t' adalah data yang telah di *differencing* maka proses *differencing* dirumuskan

$$Y'(t) = (1 - B)^d Y(t) \quad (2.3)$$

dengan B merupakan operator *backward shift*, yang mana operator *backward shift* adalah operator yang memperlihatkan perpindahan data suatu periode kebelakang. Sedangkan d adalah variabel yang merupakan orde dari *differencing*, yaitu banyaknya *differencing* yang akan dilakukan sampai data stasioner.

2.2.2 Transformasi *Box-Cox*

Asumsi yang biasa terkenal di bidang statistik yaitu kenormalan data, kehomogenan ragam dan linieritas tak dipenuhi dan sebagainya. Untuk mengatasi hal tersebut salah satu cara yang dapat dilakukan yaitu transformasi terhadap variabel respon dengan metode transformasi *Box-Cox*. Transformasi *Box-Cox* merupakan transformasi pangkat pada variabel tak bebas dimana variabel bebasnya bernilai positif. Box dan Cox mempertimbangkan kelas transformasi berparameter tunggal, yaitu λ yang dipangkatkan pada variabel tak bebas Y , sehingga transformasinya menjadi $Y^{-\lambda}$, dimana λ adalah parameter yang perlu diduga. Tabel 2.1 dibawah adalah beberapa nilai dengan transformasinya (Ispriyanti, 2004).

Tabel 2.1 Nilai λ dan transformasinya

λ	Transformasi
2	Y^2
0,5	\sqrt{Y}
0	$\log Y$ atau $\ln Y$
-0,5	$1/\sqrt{Y}$
-1	$1/Y$

2.3 Autocorrelation Function (ACF) dan Partial Autocorrelation Function (PACF)

Dalam metode *time series*, alat utama untuk mengidentifikasi model dari data yang akan diramalkan adalah dengan menggunakan fungsi Autokorelasi/*Autocorrelation Function (ACF)* dan fungsi Autokorelasi parsial/*Partial Autocorrelation Function (PACF)*. Menurut Wei (2006) proses stasioner suatu data *time series* (X_t) diperoleh $E(X_t) = \mu$ dan variansi $Var(X_t) = E((X_t - \mu)^2) = \sigma^2$, yang konstan dan kovariansi $Cov(X_t, X_{t+k})$, yang fungsinya hanya pada perbedaan waktu $|t - (t+k)|$. Maka, hasil tersebut dapat dituliskan sebagai kovariansi antara X_t dan X_{t+k} sebagai berikut:

$$\gamma_k = Cov(X_t, X_{t+k}) = E(X_t - \mu)(X_{t+k} - \mu) \quad (2.4)$$

dan korelasi antara X_t, X_{t+k} sebagai

$$\begin{aligned} \rho_k &= \frac{Cov(X_t, X_{t+k})}{\sqrt{Var(X_t)}\sqrt{Var(X_{t+k})}} \\ &= \frac{\gamma_k}{\gamma_0} \end{aligned} \quad (2.5)$$

Berdasarkan persamaan (2.4) dan karena adanya proses stasioner, maka variansi $Var(X_t) = Var(X_{t+k}) = \sigma^2 = \gamma_0$ pada waktu t adalah sama untuk waktu $t+k$.

Fungsi autokovariansi sampel dan fungsi autokorelasi sampel dapat ditulis sebagai berikut:

$$\gamma_k = \frac{1}{T} \sum_{t=1}^{T-k} (X_t - \bar{X})(X_{t+k} - \bar{X}) \quad (2.6)$$

dan

$$\rho_k = \frac{\gamma_k}{\gamma_0} = \frac{\sum_{t=1}^{T-k} (X_t - \bar{X})(X_{t+k} - \bar{X})}{\sum_{t=1}^T (X_t - \bar{X})^2}, k = 0, 1, 2, \dots \quad (2.7)$$

dengan

$$\bar{X} = \frac{1}{T} \sum_{t=1}^T X_t \quad (2.8)$$

Autocorrelation Function (ACF) adalah fungsi yang memperlihatkan besarnya korelasi antara pengamatan pada waktu ke- t dengan pengamatan pada

waktu sebelumnya $t-1$, ini juga dapat dipergunakan untuk dasar penentu kestasioneran. Deret waktu yang telah stasioner dapat terlihat dari pola fungsi korelasi diri yang menunjukkan penurunan nilai-nilai korelasi diri. Sedangkan deret waktu yang tidak stasioner dapat dilihat jika penurunan nilai-nilai korelasi diri lambat.

Autokorelasi parsial adalah ukuran lain pada analisa deret waktu. Autokorelasi parsial digunakan dalam mengukur tingkat korelasi antara X_t dan X_{t-k} , apabila pengaruh dari lag waktu $1, 2, \dots, k-1$ dianggap terpisah. Fungsi autokorelasi parsial merupakan fungsi yang menunjukkan besarnya korelasi antara pengamatan ke t yaitu X_t dengan pengamatan waktunya $X_{t-1}, X_{t-2}, \dots, X_{t-k}$. Menurut Wei (2006) fungsi autokorelasi parsial dinotasikan dengan:

$$\phi_{kk} = \text{corr}(X_t, X_{t+k}, X_{t+1}, X_{t+2}, \dots, X_{t+k-1}) \quad (2.9)$$

Nilai ϕ_{kk} dapat ditentukan melalui persamaan Yule-Walker

$$\rho_j = \phi_{k1}\rho_{j-1} + \phi_{k2}\rho_{j-2} + \dots + \phi_{kk}\rho_{j-k} \quad (2.10)$$

Dengan $j = 1, 2, \dots, k$ sehingga akan terbentuk sistem persamaan linier dengan k variabel. Sistem persamaan ini dapat terselesaikan menggunakan aturan *Cramer*. Dalam menyelesaikan persamaan Yule-Walker dapat menggunakan rumusan Durbin, dimana

$$\phi_{kk} = \frac{\rho_j - \sum_{j=1}^{k-1} \phi_{k-1,j} \rho_{k-j}}{1 - \sum_{j=1}^{k-1} \phi_{k-1,j} \rho_j} \quad (2.11)$$

Dengan $\phi_{kj} = \phi_{k-1,j} - \phi_{kk} \phi_{k-1,k-j}$; $j = 1, 2, \dots, k-1$. Estimator untuk ϕ_{kk} adalah $\hat{\phi}_{kk}$ yang diperoleh dengan mengganti ρ_k dalam persamaan Yule-Walker dengan penaksirnya yaitu $\hat{\rho}_k$, maka fungsi dari k , ϕ_{kk} disebut fungsi autokorelasi parsial (*Partial Autocorrelation function*) (Montgomery :2008).

Pada proses $\{a_t\}$ yang disebut *white noise* jika bentuk peubah acak yang berurutan tidak saling berkorelasi dan mengikuti distribusi tertentu yang hampir sama. Proses *white noise* memiliki fungsi mean konstan $E(a_t) = \mu_t$ untuk

$t = 1, 2, \dots$ dan $(a_t) = \sigma_a^2$, dan kovariansi untuk proses ini adalah $\gamma_k = \text{Cov}(a_t, a_{t+k}) = 0$ untuk $k \neq 0$.

2.4 Model Autoregressive Integrated Moving Average (ARIMA)

Model *Autoregressive Integrated Moving Average* (ARIMA) adalah campuran antara model *Autoregressive* (AR) berordo p dengan model *Moving Average* (MA) berordo q yang telah mengalami proses *differencing* sebanyak d kali dan model campuran antara model musiman *Autoregressive* (AR) berordo P dengan *Moving Average* (MA) berordo Q yang mengalami *seasonal difference* sebanyak D kali dengan periode musiman T . Secara umum model ARIMA (p, q, d) untuk data time series Y_t dinyatakan sebagai berikut:

$$\phi(B)(1-B)^d Y_t = \theta(B)a_t \quad (2.12)$$

dengan

Y_t : data observasi ke- t

B : operator *back shift*

$(1-B)^d$: *time series* yang stasioner pada perbedaan ke- d

a_t : nilai *error* pada waktu ke- t

p : order AR

d : order perbedaan

q : order M

Model ARIMA musiman adalah model ARIMA yang digunakan dalam menyelesaikan *time series* musiman yang terdiri atas dua bagian yaitu bagian non musiman dan bagian musiman. Berdasarkan Montgomery (2008) secara umum bentuk model ARIMA musiman atau ARIMA $(p, q, d) (P, Q, D)_S$ adalah sebagai berikut:

$$\phi_p(B^S)\phi_q(B)(1-B)^d(1-B^S)^D X_t = \theta_Q(B^S)\theta_q(B)a_t \quad (2.13)$$

dengan:

p, q, d : order AR, MA dan *differencing* non musiman

- P, Q, D : order AR, MA dan *differencing* musiman
 S : Jumlah periode setiap musim
 $(1 - B)^d$: *differencing* non musiman
 $(1 - B^S)^P$: *differencing* musiman

Metode Box-Jenkins dapat digunakan untuk memprediksi model ARIMA. Metode ini hanya dapat digunakan pada data deret waktu yang telah stasioner. Dalam metode ini terdapat tiga langkah yaitu identifikasi model, pendugaan parameter dan diagnostik model.

Identifikasi model adalah langkah pertama dalam menentukan model sementara pada deret waktu yang telah stasioner, yaitu dengan menentukan nilai p , q dan d pada ARIMA non musiman sedangkan nilai P , Q dan D pada ARIMA musiman. Penentuan nilai yang dibutuhkan dapat dilakukan dengan melihat grafik fungsi ACF (korelogram) dan PACF (korelogram parsial). Nilai pada grafik fungsi PACF dapat digunakan sebagai nilai p (ordo proses AR) dan nilai pada grafik fungsi ACF dapat digunakan sebagai nilai q (ordo proses MA). Pada dasarnya bentuk ACF dan PACF pada ARIMA non musiman hampir sama dengan model ARIMA musiman, hanya saja nilai yang keluar terlihat seperti berulang pada setiap periodenya.

Langkah kedua adalah pendugaan parameter. Pada langkah ini bertujuan untuk menentukan apakah parameter sudah layak untuk digunakan pada model. Dalam pendugaan parameter dapat menggunakan metode *Maximum Likelihood Estimation* (MLE). Menurut Box (1994) asumsi yang diperlukan pada metode ini adalah error a_t (nilai error waktu). Dengan begitu, fungsi kepadatan peluangnya adalah:

$$f(a_t | \sigma_a^2) = (2\pi\sigma_a^2)^{-\frac{1}{2}} \exp\left(-\frac{a_t^2}{2\pi\sigma_a^2}\right) \quad (2.14)$$

Perlu diingat bahwa *error* bersifat independen, maka distribusi bersama dari $a_1, a_2, a_3, \dots, a_n$ adalah

$$f(a_1, a_2, a_3, \dots, a_n | \sigma_a^2) = (2\pi\sigma_a^2)^{-\frac{n}{2}} \exp\left(-\frac{1}{2\sigma_a^2} \sum_i^n a_i^2\right) \quad (2.15)$$

error a_t dapat dinyatakan dalam fungsi Y_t dan parameter-parameter ϕ, θ, σ_a^2 serta *error* sebelumnya. Secara umum bentuk a_t adalah

$$a_t = Y_t - \phi_1 Y_{t-1} - \phi_p Y_{t-p} + \theta_1 a_{t-1} + \dots + \theta_q a_{t-q} \quad (2.16)$$

fungsi *likelihood* jika diketahui data pengamatan adalah

$$L(\phi, \theta, \sigma_a^2 | Y) = (2\pi\sigma_a^2)^{-\frac{n}{2}} \exp\left(-\frac{1}{2\sigma_a^2} S(\phi, \theta)\right) \quad (2.17)$$

dengan $S(\phi, \theta) = \sum_{t=1}^n -\left[(Y_t - \phi_1 Y_{t-1} - \phi_p Y_{t-p}) + (\theta_1 a_{t-1} + \dots + \theta_q a_{t-q}) \right]^2$ maka persamaan (2.17) diperoleh fungsi *log likelihood*nya adalah

$$l = \ln L(\phi, \theta, \sigma_a^2 | Y) = -\frac{n}{2} \log 2\pi - \frac{n}{2} \log \sigma_a^2 - \frac{1}{2\sigma_a^2} S(\phi, \theta) \quad (2.18)$$

Maksimum fungsi *log likelihood* dihitung dengan mencari turunan pertama persamaan (2.18) terhadap masing-masing parameter dan menetapkan turunan parsial yang dihasilkan sama dengan nol.

$$\frac{\partial(\phi, \theta, \sigma_a^2 | Y)}{\partial \phi} = 0; \frac{\partial(\phi, \theta, \sigma_a^2 | Y)}{\partial \theta} = 0; \frac{\partial(\phi, \theta, \sigma_a^2 | Y)}{\partial \sigma_a^2} = 0 \quad (2.19)$$

Untuk memperoleh standart baku, maka parameter diestimasi dengan menggunakan metode MLE. Dalam hal ini digunakan matriks informasi yang dinotasikan $I(\phi, \theta)$. Matrik ini didapatkan dengan menghitung turunan kedua pada setiap parameternya, yang dinotasikan dengan l_{ij} .

$$l_{ij} = \frac{\partial^2 l(\beta, \sigma_a^2 | Y)}{\partial \beta_i \partial \beta_j} \text{ dan } I(\beta) = -E(l_{ij}) \quad (2.20)$$

Untuk varians parameter dilambangkan dengan $V(\hat{\beta})$ dan standart baku parameternya adalah $SE(\hat{\beta})$

$$V(\hat{\beta}) = [I(\hat{\beta})]^{-1} \text{ dan } SE(\hat{\beta}) = [V(\hat{\beta})]^{1/2} \quad (2.21)$$

Parameter yang telah didapatkan selanjutnya diuji signifikansinya. Dengan perumusan hipotesis yang dilakukan adalah sebagai berikut:

$$H_0 : \hat{\beta} = 0 \text{ (parameter tidak signifikan terhadap model)}$$

$$H_0 : \hat{\beta} \neq 0 \text{ (parameter signifikan terhadap model)}$$

Statistik uji yang digunakan adalah

$$|t - \text{hitung}| = \frac{\hat{\beta}}{SE(\hat{\beta})} \quad (2.22)$$

Dimana $\hat{\beta}$ merupakan parameter dugaan dan $SE(\hat{\beta})$ adalah *standart error* pada setiap parameter dugaan.

Pendugaan parameter pada suatu model dikatakan berpengaruh signifikan atau

H_0 ditolak jika nilai $|t - \text{hitung}| > t - \text{tabel} \left(t_{\left(\frac{\alpha}{2}\right); df=n-1} \right)$ atau $p - \text{value} < \alpha$. Dimana

α merupakan taraf nyata (*level of significance*) yang pada hal ini bernilai 0.05 (5%). *Freedom of degree (df)* merupakan tingkat kepercayaan yang diperoleh dari operasi pengurangan antara jumlah data dengan jumlah perkiraan parameter, dan n jumlah pengamatan. Langkah terakhir yang dilakukan adalah mendiagnosa model untuk melihat model yang cocok dengan data. Dalam langkah ini model harus diuji kelayakan dengan melihat sifat sisaan dari sisi kebebasannya (*white noise*) dan kenormalan. Untuk melihat kebebasan sisaan model, dapat menggunakan uji Q, modifikasi Box-Pierce (Ljung-Box). Berikut persamaan uji Q.

$$Q = n(n+2) \left(\sum_{k=1}^k \frac{r_k^2}{n-k} \right) \quad (2.23)$$

dengan

r_k : nilai korelasi diri sisaan pada lag ke- k ,

n : banyaknya data yang akan diamati

k : lag maksimum

Statistik uji Q *Ljung-Box* menyebar mengikuti sebaran $\chi^2_{(k-p-q)}$ dengan p merupakan ordo AR dan q adalah ordo MA. Langkah pengujian *white noise*

dengan menggunakan uji *Ljung-Box* memiliki perumusan hipotesis sebagai berikut:

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

H_0 : minimal ada satu $\rho_i \neq 0$ untuk $i = 1, 2, \dots, k$ minimal ada satu (residual belum memenuhi syarat *white noise*)

Dimana daerah kritis adalah jika nilai Q lebih besar dari nilai $\chi^2_{(k-p-q)}$ pada tingkat kepercayaan tertentu ($df = k - p - q$) atau nilai peluang statistik Q *Ljung-Box* lebih kecil dari taraf nyata (α) atau $-value > \alpha$, sehingga dapat dikatakan sisaan tidak saling bebas (H_0 ditolak). Kenormalan pada sisaan dapat dilihat dari p -value hasil uji *shapiro-wilk normality*. Jika p -value yang dihasilkan lebih dari α , dapat disimpulkan jika sisaan telah memenuhi asumsi kenormalan sisaan, dengan $\alpha = 5\%$.

Jika semua proses dalam metode *Box-Jenkins* telah dilakukan, selanjutnya, akan dilakukan validasi model pada model sementara yang diperoleh dan menetapkan model terbaik yang akan digunakan untuk memprediksi beberapa waktu kedepan dengan membandingkan nilai *Akaike's Information Criterion* (AIC) dan *Root Mean Square Error* (RMSE). Dimana jika nilai AIC semakin kecil maka model yang diperoleh semakin baik, persamaan AIC adalah sebagai berikut:

$$AIC(M) = n \ln(\hat{\sigma}_a^2) + 2M \quad (2.24)$$

2.5 Matrix Autocorrelation Function (MACF)

Bila diberikan suatu vektor deret waktu (time series) sebanyak n pengamatan $Z_1, Z_2, Z_3, Z_4, \dots, Z_n$ maka matrik korelasi sampel dinyatakan :

$$\hat{\rho}(k) = [\hat{\rho}_{ij}(k)] \quad (2.25)$$

Dimana $\hat{\rho}_{ij}(k)$ merupakan korelasi silang sampel dari komponen deret ke- i dan ke- j yaitu

$$\hat{\rho}_{ij}(k) = \frac{\sum_{t=1}^{n-k} (z_{i,t} - \bar{z}_i)(z_{j,t+k} - \bar{z}_j)}{\left[\sum_{t=1}^n (z_{i,t} - \bar{z}_i)^2 \sum_{t=1}^n (z_{j,t+k} - \bar{z}_j)^2 \right]^{\frac{1}{2}}} \quad (2,26)$$

Dengan \bar{z}_i dan \bar{z}_j ialah rata - rata sampel dari komponen deret yang bersesuaian.

Dalam hal ini bentuk matriks dan grafik akan menjadi semakin kompleks jika dimensi dan vektor semakin besar, sehingga akan menyulitkan dalam pengidentifikasian. Untuk mempermudah dalam meringkas korelasi sampel digunakan metode sederhana yaitu dengan menggunakan simbol yang dinotasikan dengan (+), (-) dan (.) pada matriks korelasi sampel ke (i, j) . Arti dari simbol – simbol tersebut adalah sebagai berikut :

Simbol (+) menunjukkan nilai $\hat{\rho}_{ij}(k)$ lebih besar dari 2 kali standar error dari $\hat{\rho}_{ij}(k)$ dan menunjukkan hubungan korelasi positif antara komponen (i, j) .

- a) Simbol (-) menunjukkan nilai $\hat{\rho}_{ij}(k)$ lebih kecil dari -2 kali standar error dari $\hat{\rho}_{ij}(k)$ dan menunjukkan hubungan korelasi negatif antara komponen (i, j) .
- b) Simbol (.) menunjukkan nilai $\hat{\rho}_{ij}(k)$ lebih besar dari ± 2 kali standar error dari $\hat{\rho}_{ij}(k)$ dan menunjukkan tidak adanya hubungan korelasi antara komponen (i, j) (Wei, 2006).

Standart error dari $\hat{\rho}_{ij}(k)$ dapat dihitung dengan menggunakan persamaan:

$$S_{\hat{\rho}_{ij}(k)} = \sqrt{\frac{1}{T} (1 + 2\hat{\rho}_{i,j}^2(1) + 2\hat{\rho}_{i,j}^2(2) + \dots + 2\hat{\rho}_{i,j}^2(k-1))} \quad (2.27)$$

dengan T adalah banyaknya observasi.

2.6 Matrix Partial Autocorrelasi Function (MPACF)

Fungsi matriks autokorelasi parsial diperlukan dalam mengidentifikasi orde dari sebuah model AR. Korelasi antara Z_t dan Z_{t+k} dapat diketahui setelah ketergantungan liniernya pada $Z_{t+1}, Z_{t+2}, Z_{t+3}, \dots, Z_{t+k-1}$ dihilangkan. Persamaan *Matrix Partial Autocorrelasi Function* (MPACF) dirumuskan sebagai berikut :

$$\phi_{kk} = \frac{\text{cov}[(Z_t - \hat{Z}_t), (Z_{t+k} - \hat{Z}_{t+k})]}{\sqrt{\text{var}(Z_t - \hat{Z}_t)}\sqrt{\text{var}(Z_{t+k} - \hat{Z}_{t+k})}} \quad (2.28)$$

Matrix Partial Autocorrelation Function (MPACF) pada lag ke- k dinotasikan $P(k)$ merupakan koefisien matriks terakhir jika data diterapkan untuk suatu proses *vector autoregressive* pada orde ke- p . Seperti PACF untuk kasus *univariate time series*, MAPCF juga bersifat terputus setelah lag p pada model VAR (p) (Wei, 2006).

2.7 Generalized Space-Time Autoregressive (GSTAR)

Generalized Space-Time Autoregressive (GSTAR) pertama kali diperkenalkan oleh Borovkova, *et al* pada tahun 2002. Model GSTAR merupakan model pengembangan dari model *Space Time Autoregressive* (STAR) dimana GSTAR lebih fleksibel dibandingkan dengan model STAR. Menurut Suhartono dan Subanar (2006) secara matematis notasi dari model GSTAR ($1:p$) sama dengan model STAR ($1:p$). Pada model GSTAR nilai – nilai parameter lag spasial yang sama diperbolehkan berlainan, sedangkan pada model STAR pada parameter *autoregressive* diasumsikan sama pada semua lokasi.

Jika suatu deret $\{Z(t): t = 0, \pm 1, \pm 2, \dots, \pm n\}$ merupakan multivariat time series dari N lokasi, maka model GSTAR ($p; \lambda_1, \lambda_2, \dots, \lambda_p$):

$$Z(t) = \sum_{k=1}^p [\phi_{k0} + \phi_{k1}W]Z_{t-k} + e(t) \quad (2.29)$$

dengan :

ϕ_{k0} : matriks diagonal parameter space time lag spasial 0 dan parameter autoregressive lag waktu ke- k

ϕ_{k1} : matriks diagonal parameter space time lag spasial 1 dan parameter autoregressive lag waktu ke- k

W : matriks pembobot ($n \times n$)

$e(t)$: vektor noise ukuran ($n \times 1$)

λ_k : orde spasial dengan $k = 1, 2, \dots, p$

(Suhartono dan Subanar, 2006).

2.7.1 Bobot Lokasi Invers Jarak

Pada pembobotan ini, nilai diperoleh berdasarkan perhitungan jarak antar lokasi yang sebenarnya. Pembobotan ini mengacu pada jarak antar lokasi, misalkan jarak diantara 4 lokasi didefinisikan sebagai berikut :

r_1 = jarak antara lokasi 1 dengan lokasi 2

r_2 = jarak antara lokasi 1 dengan lokasi 3

r_3 = jarak antara lokasi 1 dengan lokasi 4

r_4 = jarak antara lokasi 2 dengan lokasi 3

r_5 = jarak antara lokasi 2 dengan lokasi 4

r_6 = jarak antara lokasi 3 dengan lokasi 4

Jika dituliskan dalam bentuk matriks :

$$\begin{bmatrix} 0 & \frac{r_2 + r_3}{r_1 + r_2 + r_3} & \frac{r_1 + r_3}{r_1 + r_2 + r_3} & \frac{r_1 + r_2}{r_1 + r_2 + r_3} \\ \frac{r_4 + r_5}{r_1 + r_4 + r_5} & 0 & \frac{r_1 + r_5}{r_1 + r_4 + r_5} & \frac{r_1 + r_4}{r_1 + r_4 + r_5} \\ \frac{r_4 + r_6}{r_2 + r_4 + r_6} & \frac{r_2 + r_6}{r_2 + r_4 + r_6} & 0 & \frac{r_2 + r_4}{r_2 + r_4 + r_6} \\ \frac{r_5 + r_6}{r_3 + r_5 + r_6} & \frac{r_3 + r_6}{r_3 + r_5 + r_6} & \frac{r_3 + r_5}{r_3 + r_5 + r_6} & 0 \end{bmatrix} = \begin{bmatrix} 0 & W_{12} & W_{13} & W_{14} \\ W_{21} & 0 & W_{23} & W_{24} \\ W_{31} & W_{32} & 0 & W_{34} \\ W_{41} & W_{42} & W_{43} & 0 \end{bmatrix}$$

Matriks tersebut distandarkan dalam bentuk W_{ij}^* untuk mendapatkan

$$\sum_{i \neq j} W_{ij}^{(i)} = 1 \text{ (Anggraeni et al, 2013).}$$

2.8 Kalman Filter

Metode Kalman Filter (KF) pertama kali diperkenalkan oleh Rudolph E. Kalman pada tahun 1960 melalui paper tentang suatu penyelesaian rekursif pada masalah filtering data diskrit yang linier (Welch & Bishop, 2006). KF merupakan metode estimasi variabel keadaan dari sistem dinamik stokastik linier diskrit yang meminimumkan kovariansi error estimasi. KF adalah suatu pendekatan teknis untuk menaksir fungsi parameter dalam peramalan *time series*. Metode KF mempunyai keunggulan yaitu kemampuannya dalam mengestimasi suatu keadaan

berdasarkan data yang minim. Data minim tersebut merupakan data pengukuran karena KF adalah metode yang menggabungkan model dan pengukuran. Data pengukuran baru menjadi bagian terpenting dalam algoritma KF karena data mutakhir berguna dalam mengoreksi hasil prediksi, sehingga hasil estimasi selalu mendekati keadaan sebenarnya (Masduqi, 2008).

Bentuk umum sistem dinamik stokastik linier diskrit adalah:

$$x_{k+1} = \mathbf{A}_k x_k + \mathbf{B}_k u_k + w_k$$

dengan pengukuran $z_k \in \mathfrak{R}^p$ yang memenuhi

$$z_k = \mathbf{H}_k x_k + v_k$$

$$x_0 \sim N(\bar{x}_0, P_{x_0}), w_k \sim N(0, Q_k), v_k \sim N(0, R_k)$$

dengan :

- x_0 : inisial dari sistem,
- x_{k+1} : variabel keadaan pada waktu $k+1$,
- x_k : variabel keadaan pada waktu k yang nilai estimasi awalnya \bar{x}_0 dan kovariansi awal P_{x_0} , $x_k \in \mathfrak{R}^n$,
- u_k : vektor masukan deterministik pada waktu k , $u_k \in \mathfrak{R}^m$
- w_k : *noise* pada sistem dengan mean $\bar{w}_k = 0$ dengan kovariansi Q_k
- z_k : variabel pengukur $z_k \in \mathfrak{R}^p$
- v_k : *noise* pengukuran dengan rata – rata $\bar{v}_k = 0$ dengan kovariansi R_k
- $\mathbf{A}_k, \mathbf{B}_k, \mathbf{H}_k$: matriks dengan nilai elemen – elemennya adalah koefisien variabel masing – masing

Variabel $w_k \sim N(0, Q_k)$ dan $v_k \sim N(0, R_k)$ diasumsikan *white*, tidak berkorelasi satu sama lain maupun dengan nilai estimasi awal \bar{x}_0 .

Estimasi pada metode KF dilakukan dengan dua tahapan, yaitu dengan cara memprediksi variabel keadaan berdasarkan sistem dinamik yang disebut tahapan prediksi (*time update*) dan selanjutnya tahap koreksi (*measurement update*) terhadap data pengukuran untuk memperbaiki hasil estimasi.

Tahapan prediksi dipengaruhi oleh dinamika sistem dengan memprediksi variabel keadaan dengan menggunakan persamaan estimasi variabel keadaan sedangkan tingkat akurasi dihitung dengan persamaan kovariansi *error*.

Pada tahapan koreksi hasil estimasi variabel keadaan yang didapatkan pada tahapan prediksi dikoreksi dengan model pengukuran. Salah satu bagian dari tahapan ini adalah menentukan matriks Kalman Gain yang digunakan untuk meminimumkan kovariansi *error*. Tahap prediksi dan koreksi dilakukan secara rekursif dengan cara meminimumkan kovariansi *error*. Algoritma KF disajikan pada Tabel 2.1.

Tabel 2.1 Algoritma *Kalman Filter*

Model Sistem dan Model Pengukuran	
$x_{k+1} = \mathbf{A}_k x_k + \mathbf{B}_k u_k + w_k$ $z_k = \mathbf{H}_k x_k + v_k$ $x_0 \sim N(\bar{x}_0, P_{x_0}^-); w_k \sim N(0, Q_k); v_k \sim N(0, R_k)$	
Inisialisasi	
$\hat{x}_0 = \bar{x}_0, P_0 = P_{x_0}^-$	
Tahap Prediksi	
Estimasi	: $\hat{x}_{k+1}^- = \mathbf{A}_k \hat{x}_k + \mathbf{B}_k u_k$
Kovarian <i>error</i>	: $P_{k+1}^- = \mathbf{A}_k P_k \mathbf{A}_k^T + Q_k$
Tahap Koreksi	
Kalman Gain	: $K_{k+1} = P_{k+1}^- \mathbf{H}_{k+1}^T (\mathbf{H}_{k+1} P_{k+1}^- \mathbf{H}_{k+1}^T + R_{k+1})^{-1}$
Estimasi	: $\hat{x}_{k+1} = \hat{x}_{k+1}^- + K_{k+1} (z_{k+1} - \mathbf{H}_{k+1} \hat{x}_{k+1}^-)$
Kovariansi <i>error</i>	: $P_{k+1} = [I - K_{k+1} \mathbf{H}_{k+1}] P_{k+1}^-$

Pada Tabel 2.1 terlihat bahwa algoritma KF yang terdiri atas empat bagian yaitu bagian pertama mendefinisikan model sistem dan model pengukuran, bagian kedua adalah nilai awal (inisialisasi), sedangkan bagian ketiga dan keempat masing-masing adalah tahapan prediksi dan koreksi. Norm matriks kovarians *error* digunakan sebagai tolak ukur dari baik tidaknya metode Kalman Filter dalam mengestimasi suatu model. Jika nilai dari kovarians *error* makin kecil, maka hasil estimasi memiliki tingkat keakuratan yang semakin tinggi (Purnomo, 2008).

2.9 Super – Ensemble Kalman Filter

Metode kombinasi linier menggambarkan dengan sangat baik konsep dari super-ensemble (SE). Super-ensemble *Kalman Filter* menggunakan pendekatan yang sama seperti kombinasi linier, akan tetapi pada super-ensemble *Kalman Filter* ini, bobot dan matrik kovarians *error*nya bergerak secara dinamis sehingga memungkinkan hasil yang lebih baik dari pengamatan sebelumnya. Selama dalam periode *learning* formulasi untuk model *forecast* adalah sebagai berikut:

$$\mathbf{w}_j^f = \mathbf{I} \mathbf{w}_{j-1}^a, \quad j = 1, \dots, N_I$$

$$\mathbf{P}_j^f = \mathbf{I} \mathbf{P}_{j-1}^a \mathbf{I}^T + \mathbf{Q}_{j-1}, \quad j = 1, \dots, N_I$$

dimana:

f : *forecast* untuk pembobotan selama periode *learning*,

\mathbf{I} : matrik identitas ($\mathbf{M} \times \mathbf{M}$),

\mathbf{w}_j : bobot vektor pada waktu ke - j ,

\mathbf{P}_j : bobot matrik kovarian *error* ($\mathbf{M} \times \mathbf{M}$),

\mathbf{Q}_j : model matrik kovarians *error* ($\mathbf{M} \times \mathbf{M}$),

\mathbf{P} dan \mathbf{Q} : inisial diagonal

Beberapa elemen pada diagonal \mathbf{P} dapat berkembang sebagai filter pengamatan dalam mengasimilasi, sedangkan \mathbf{Q} diagonal tetap selama proses berlangsung. Pada langkah analisis, vektor keadaan dan matriks kovariansi diperbarui dengan

menambah sebuah komponen prediksi yang memperhitungkan model dan ketidakpastian dalam pengamatan, seperti yang ditunjukkan dalam persamaan

$$\mathbf{w}_j^a = \mathbf{w}_j^f + \mathbf{K}_j (y_j - \mathbf{x}_j \mathbf{w}_j^f) \quad , j = 1, \dots, N_I$$

$$\mathbf{P}_j^a = \mathbf{P}_j^f - \mathbf{K}_j \mathbf{x}_j \mathbf{P}_j^f \quad , j = 1, \dots, N_I$$

dengan:

a : analisis atau koreksi pada data baru,

\mathbf{x}_j : setara dengan operator observasi pada teori klasik KF dan dapat diartikan sebagai operator yang memetakan bobot ke ruang observasi,

\mathbf{R}_j : pengamatan matriks kovarian pada waktu ke - j

$\mathbf{K}_j = \mathbf{P}_j^f \mathbf{x}_j^T (\mathbf{x}_j \mathbf{P}_j^f \mathbf{x}_j^T + \mathbf{R}_j)^{-1}$ adalah matrik Kalman Gain pada waktu ke- j ($M \times 1$). Sehingga diperoleh persamaan *hindcast* dan *forecast* sebagai berikut:

$$h_j^{KF} = \sum_{i=1}^M x_{j,i} w_i \quad , j = 1, \dots, N_I$$

$$f_j^{KF} = \sum_{i=1}^M x_{j,i} w_i \quad , j = 1, \dots, N_I$$

(Lenartz *et al*, 2010).

2.10 Norm Kovarians Error

Dalam metode *Kalman Filter* norm kovarians *error* digunakan sebagai tolak ukur dalam mengestimasi suatu model pada metode KF. Semakin kecil nilai kovarians *error* maka semakin tinggi tingkat keakuratan estimasinya.

Norm matrik pada himpunan S yang memuat matriks berukuran $n \times n$, dituliskan menggunakan notasi $\|\cdot\|$ atau biasanya disebut panjang/ besar yang merupakan fungsi bernilai real dan positif yang terdefinisi pada S dan memenuhi (Purnomo 2008). Norm matriks didefinisikan

$$\|x\| = \sqrt{\sum_{i=1}^n \sum_{j=1}^n a_{ij}^2}$$

(2.31)

Kovarian dua variable acak x_1 dan x_2 didefinisikan

$$\text{cov}(x_1, x_2) = E[(x_1 - \hat{x}_1)(x_2 - \hat{x}_2)]$$

Dan kovarian vektor kolom $x = [x_1 \dots x_n]^T$ didefinisikan

$$\text{cov}(x) = E[(x - \hat{x})(x - \hat{x})^T]$$

Matrik kovarian *error* adalah hubungan antara *error* masing-masing variable pada proses estimasi. Semakin besar nilai norm kovarian *error* maka semakin besar pula korelasi antara *error* variabelnya maka *error* yang dihasilkan juga semakin besar sehingga menghasilkan estimasi yang semakin jauh dari kenyataannya, begitu juga sebaliknya.

2.11 Root Mean Square Error (RMSE)

Root Mean Square Error (RMSE) merupakan metode yang digunakan dalam mengevaluasi teknik peramalan untuk mengukur tingkat ketepatan hasil peramalan suatu model. RMSE adalah nilai rata-rata dari jumlah kuadrat kesalahan, dimana RMSE juga dapat digunakan untuk menyatakan ukuran besarnya suatu kesalahan yang dihasilkan suatu model peramalan. Nilai RMSE yang kecil menunjukkan bahwa variasi nilai yang dihasilkan oleh suatu model peramalan mendekati variasi nilai pengamatannya. Perhitungan RMSE dapat dilihat dalam persamaan berikut:

$$RMSE = \sqrt{\frac{1}{N} \sum_{t=1}^N (Y_t - \hat{Y}_t)^2} \quad (2.32)$$

dimana N adalah banyaknya data yang akan diramalkan (Wei, 2013).

BAB 3. METODOLOGI PENELITIAN

3.1 Data

Pada penelitian ini digunakan data curah hujan di kabupaten Jember dari bulan Januari 2005 sampai dengan bulan Desember 2017. Data tersebut merupakan data sekunder dari penelitian Retnaningrum (2015), penelitian Yudistira (2017) dan Dinas PU Bina Marga dan Sumber Daya Air. Data yang diperoleh dibagi menjadi dua yaitu data in-sampel dan data out-sampel. Data in-sampel adalah data yang akan digunakan untuk membentuk model peramalan, sedangkan data out sampel adalah data yang digunakan untuk memeriksa hasil peramalan dari model yang terbentuk dari data in-sample. Data bulan Januari 2005 sampai dengan data bulan Desember 2016 digunakan sebagai data in-sample, sedangkan data out-sample yang digunakan adalah data bulan Januari 2017 sampai bulan Desember 2017. Data akan dikelompokkan ke dalam empat kelompok yang masing-masing kelompok akan mewakili variabel.

3.2 Langkah – Langkah Penelitian

Langkah – langkah yang dilakukan dalam penelitian ini adalah sebagai berikut :

1. Identifikasi Data

Pada data insample perlu dilakukan plotting. Hal ini bertujuan untuk melihat stasioneritas dari data tersebut. Pengujian stasioneritas data secara matematis dapat dilakukan uji *Augmented Dickey-Fuller* (ADF). Jika pada langkah identifikasi data ini diperoleh data tidak stasioner dalam mean, maka perlu adanya proses *differencing* pada data. Tetapi jika data tidak stasioner dalam varians, maka langkah yang perlu dilakukan yaitu dengan cara mentransformasi data menggunakan transformasi Box-Cox. Selain itu, pada tahap ini juga dilakukan uji ACF dan PACF untuk

mengidentifikasi model sementara ARIMA sedangkan pada model GSTAR dilakukan dengan menentukan MACF dan MPACF. Pada setiap kluster akan disusun model ARIMA dan GSTAR dengan data observasi data curah hujan periode Januari 2005 sampai Desember 2015. Model ini yang nantinya akan digunakan untuk membangun *state space* pada metode Super-Ensemble *Kalman Filter*. Sedangkan model ARIMA yang kedua merupakan model pembandingan hasil peramalan dengan metode Super-Ensemble *Kalman Filter* untuk periode Januari 2016 sampai Desember 2016.

2. Estimasi Parameter

Pada tahapan ini digunakan untuk menentukan apakah parameter dari model sementara ARIMA dan GSTAR tersebut sudah layak untuk digunakan pada model selanjutnya. Pada model ARIMA dilakukan dengan melakukan uji signifikansi pada koefisien, jika koefisien pada model tidak signifikan maka model tersebut tidak layak digunakan. Sedangkan pada model GSTAR perlu adanya penerapan bobot lokasi, setelah penerapan bobot lokasi pada GSTAR maka akan dilakukan pendugaan parameter untuk semua bobot lokasi. Pendugaan dilakukan dengan metode kuadrat terkecil.

3. Uji Kelayakan Model

Proses ini digunakan untuk memeriksa kelayakan dari model yang didapat dan sebagai petunjuk dalam perbaikan model. Perbaikan model dilakukan dengan melakukan pemeriksaan terhadap sisaan (residual) yang meliputi asumsi *white noise* menggunakan Ljung-Box. Model yang telah memenuhi asumsi adalah model – model yang layak dan digunakan pada proses selanjutnya.

4. Pemilihan Model Terbaik

Pada tahapan ini digunakan untuk menetapkan model ARIMA dan GSTAR terbaik yang akan digunakan untuk prediksi masa yang akan datang, dapat dilakukan dengan melihat perbandingan nilai AIC dari

masing - masing model yang didapatkan. Model peramalan yang baik adalah model dengan nilai AIC kecil.

5. Identifikasi Model *State Space*

Setelah mendapatkan model terbaik dari masing – masing kluster, maka akan dibangun model *state space* pada *Kalman Filter* (KF) dengan mengensemble model ARIMA dan GSTAR yang didapatkan. Model *state space* merupakan suatu pendekatan yang digunakan untuk memodelkan dan memprediksi beberapa data deret waktu yang saling berhubungan secara bersamaan, dimana peubah yang ada didalamnya mempunyai interaksi yang dinamis dan untuk melihat peubah dalam deret waktu yang tidak terobservasi (*state vector*).

6. Penerapan *Kalman Filter* (KF)

Setelah mendapatkan model *state space* yang diinginkan, maka akan dilakukan peramalan dengan menggunakan metode *Kalman Filter* (KF). Pada tahapan ini model terbaik dari ARIMA dan GSTAR akan diensemble menggunakan metode *Kalman Filter* yaitu dengan menentukan nilai prediksi pada bobot vektor yang kemudian digunakan untuk menentukan bobot vektor pada iterasi selanjutnya, hal ini dilakukan pada tiap kluster. Pada metode *Kalman Filter* (KF) terdapat dua tahapan yang harus dilakukan dalam setiap iterasinya yaitu tahap prediksi dan koreksi. Data curah hujan pada tahun 2017 digunakan sebagai data pengamatan untuk model pada *Kalman Filter*. Tahap prediksi pada *Kalman Filter* menggunakan data curah hujan tahun 2005-2016 sedangkan untuk tahap koreksi menggunakan data curah hujan tahun 2017.

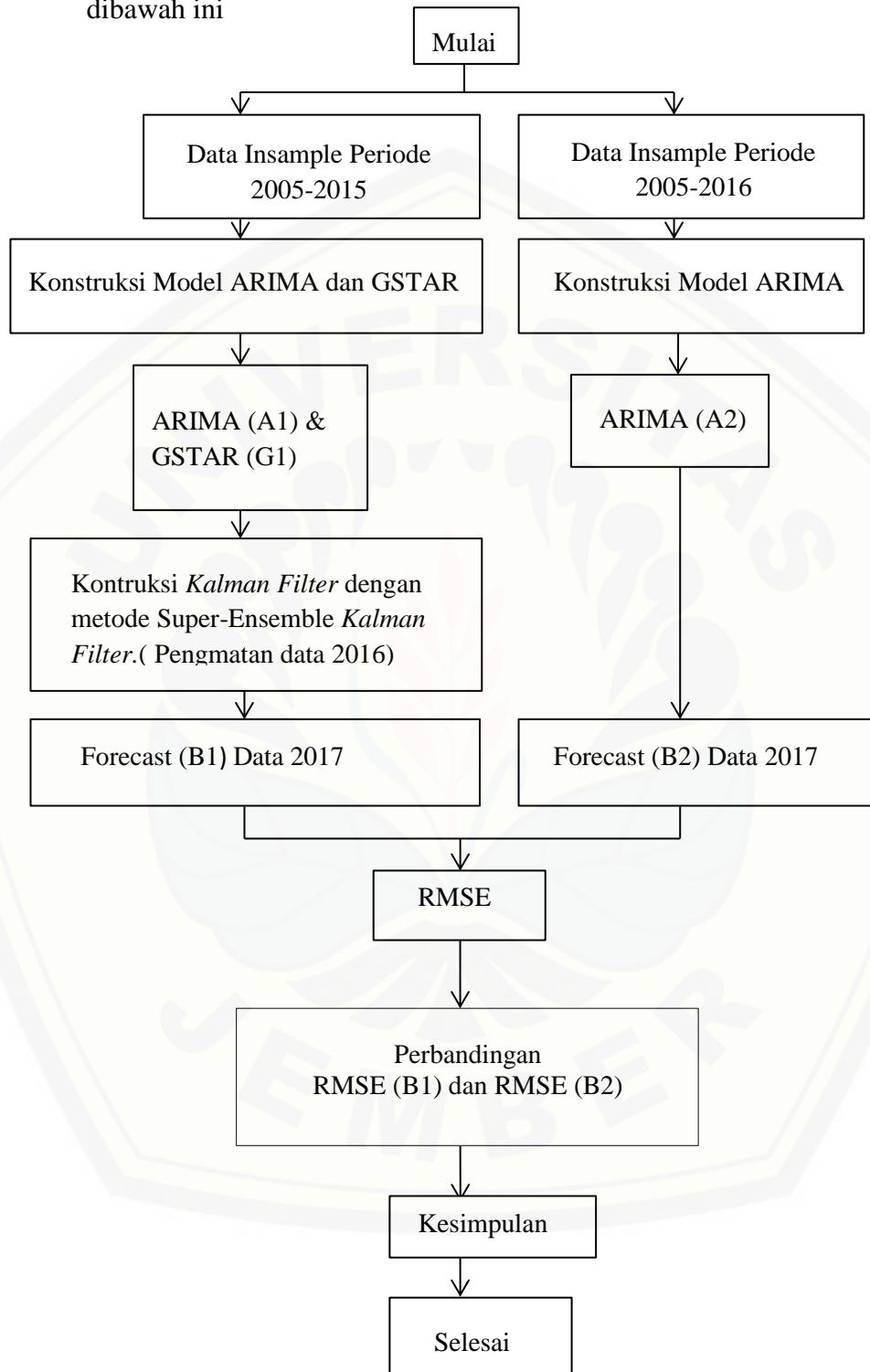
7. Kesimpulan

Pada tahapan ini akan disimpulkan perolehan hasil dari tahapan yang telah dilalui sebelumnya. Akan dilihat hasil perbandingan hasil peramalan yang diperoleh antara metode Super –Ensemble *Kalman Filter* yang menggunakan model ARIMA (A1) dan GSTAR (G1) dengan menggunakan model ARIMA (A2), apabila Super-Ensemble *Kalman Filter* lebih baik dari model ARIMA (A2) maka dapat disimpulkan bahwa

penggabungan dua model peramalan dengan menggunakan Super-Ensemble *Kalman Filter* menghasilkan peramalan yang lebih baik jika dibandingkan peramalan dengan menggunakan model ARIMA saja. Tetapi apabila Arima (A2) lebih baik dari pada Super-Ensemble *Kalman Filter*, maka dapat diambil kesimpulan bahwa metode Super-Ensemble *Kalman Filter* tidak dapat digunakan dalam peramalan curah hujan di Kabupaten Jember. Adapun cara yang digunakan untuk mengetahui hasil perbandingan Super-Ensemble *Kalman Filter* dengan ARIMA (A2) akan dilihat dari nilai *Root Mean Square Error* (RMSE) dari masing – masing peramalan.



Langkah-langkah pada penelitian ini dapat dilihat pada gambar 3.1 dibawah ini



Gambar 3.1 Langkah – langkah Penelitian

BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan hasil analisis data dan pembahasan maka diperoleh kesimpulan sebagai berikut:

1. Penggunaan metode Super Ensemble *Kalman Filter* pada model ARIMA dan GSTAR memberikan nilai norm kovarian *error* yang relatif stabil, dimana dalam metode *Kalman Filter* nilai norm kovarians error yang stabil memberikan hasil ramal yang baik.
2. Berdasarkan nilai RMSE yang dibandingkan pada keempat wilayah, metode Super-Ensemble *Kalman Filter* memberikan hasil yang lebih baik dibandingkan dengan model ARIMA (A2). Hal ini menunjukkan bahwa metode Super-Ensemble *Kalman Filter* dapat digunakan sebagai metode peramalan curah hujan di wilayah Kabupaten Jember.

5.2 Saran

Penelitian ini hanya menggunakan model ARIMA terbaru sebagai pembanding hasil peramalan menggunakan Super-Ensemble Kalman Filter. Dimana model ARIMA tidak peka dengan keheterogenan lokasi. Untuk mendapatkan hasil peramalan yang lengkap dapat dilakukan pembandingan antara metode Super-Ensemble *Kalman Filter*, ARIMA dan GSTAR dengan data terbaru.

DAFTAR PUSTAKA

- Anggraeni, D., Prahutama, A., dan Andari, S. 2013. Aplikasi Generalized Space Time Autoregressive (GSTAR) pada Pemodelan Volume Kendaraan Masuk Tol Semarang. *Media Statistika*. Vol. 6(2): 71-80.
- Borovkova, S., Lopuhaa, H.P., dan Ruchjana, B.N. 2008. Consistency and Asymptotic Normality of Least Squares Estimators in Generalized STAR Models. *Journal compilation Statistica Nederlandica*. 487-489.
- Box, G.E.P., Jenkins, G.M., dan Reinsel, G.C. 2008. *Time Series Analysis Forecasting and Control 4th ed*. New Jersey: John Wiley & Sons.
- Faragher, R. 2008. *The Basic of The Kalman Filter via a Simple and Intuitive Derivation*. IEEE Signal Processing Magazine
- Ichwan, A. 2010. *Estimasi Posisi Kapal Selam Menggunakan Metode Extended Kalman Filter*. [on line]. <http://digilib.its.ac.id/public/ITS-Undergraduate-9296-Paper.pdf>. [17 November 2017].
- Ispriyanti, D. 2004. Pemodelan Statistika dengan Transformasi Box Cox. *Jurnal Matematika dan Komputer*. Vol.7(3): 8-17.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. 2013. *An Introduction to Statistical Learning (p. 68)*. New York: Springer.
- Kleeman, L. Tanpa Tahun. *Understanding and Applying Kalman Filtering*. Clayton: Departement of Electicial and Computer System Engineering MonashUniversity.http://cs.cmu.edu/~motionplanning/paper_kleeman_kalman_basics.pdf. [15 November 2011].
- Lenartz, F., Beckers, J.M., Chiggiato, J., Mourre, B., Troupin, C., Vandenbulcke, L., Rixen, M. 2010. *Super-ensemble Techniques Applied to Wave Forecast: Performance and Limitations*. Ocean Sci. Vol 6 : 595-604.
- Makridakis, S., Wheelwriht, S.C., dan McGee, V.E. 1992. *Metode dan Aplikasi Peramalan*. Jakarta: Erlangga.

- Mckee, T.B., Doesken, N.J. & Kleist, J. 1993. *The Relationship of Drought Frequency And Duration To Time Scales*. Colorado: Department of Atmospheric Science.
- Montgomery, D. C., Jennings, C. L., Kulahci, M. 2008. *Introduction to Time Series Analysis and Forecasting*. John Wiley & Sons: Hoboken, New Jersey.
- Prillantika, J. R. 2017. *Perbandingan Model GSTAR dan GSTAR-Kalman Filter pada Peramalan Tingkat Inflasi di Tiga Kota di Jawa Timur*. Tidak Diterbitkan. Skripsi. Surabaya: Institut Teknologi Sepuluh November.
- Purnomo, K. D. 2008. *Aplikasi Metode Ensemble Kalman Filter pada Model Populasi Plankton*. Tidak Diterbitkan. Tesis. Surabaya: Program Pascasarjana Institut Teknologi Sepuluh November.
- Rothenberg, T. 2007. *State Space Models and the Kalman Filter*. Berkeley Education.
- Suhartono. 2005. Perbandingan antara Model GSTAR dan VARIMA untuk Peramalan Data Deret Waktu dan Lokasi. *Prosiding Seminar Nasional Statistika*, ITS, Surabaya.
- Suhartono & Subanar. 2006. The Optimal Determination of Space Weight in GSTAR Model by using Cross-correlation Inference. *Journal of Quantitative Methods: Journal Devoted to the Mathematical and Statistical Application in Various Fields*. Vol.2(2). pp. 45-53.
- Suhartono & Subanar. 2007. Some Comments on the Theorem Providing Stasionerity Condition for GSTAR Models in the Paper by Borovkova et al. *Journal of the Indonesian Mathematical Society (MIHMI)*. Vol.1(13).
- Welch, G., & Bishop, G. 2006. *An Introduction to the Filter Kalman*. Chapel Hill: Department of Computer Science, University of North Carolina.
- W.S. Wei, William. 2006. *Time Series Analysis*. Departemen of Statistics The Fox School of Business and Management Temple University, pp 68 – 86.

Lampiran A : Skrip Program R

1. Pemodelan ARIMA pada Kluster 1

```
> fit3=arima(datamodel1,c(4,0,4))
> fit3
Call:
arima(x = datamodel1, order = c(4, 0, 4))

Coefficients:
ar1      ar2      ar3      ar4      ma1      ma2      ma3      ma4
intercept
0.1365  0.0085  0.8643  -0.3128  -0.1007  -0.0495  -0.9937  0.143
9 62.8953
0.7075  0.1282  0.0403  0.6333  0.7452  0.0602  0.0617  0.747
6 0.6877
sigma^2 estimated as 1605:  log likelihood = -677.98,  aic = 1375
.95
> Box.test(residuals(fit3))
> fit1<-arima(kluster1,order=c(4,0,0))
> coeftest(fit1)
```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(> z)
ar1	0.0819011	0.0782159	1.0471	0.295046
ar2	-0.0095747	0.0781271	-0.1226	0.902462
ar3	0.1044359	0.0781733	1.3360	0.181565
ar4	-0.2033147	0.0781632	-2.6012	0.009291 **
intercept	64.8717482	3.3028820	19.6410	< 2.2e-16 ***

 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

2. Pemodelan ARIMA pada Kluster 2

```
Call:
arima(x = datamodel2, order = c(3, 0, 2))
Coefficients:
ar1      ar2      ar3      ma1      ma2  intercept
1.1495  -0.1977  -0.1805  -1.1490  0.2704  70.0037
s.e.  0.3752  0.4201  0.1036  0.3766  0.3915  2.1319
sigma^2 estimated as 1937:  log likelihood = -687.02,  aic = 1388
.04
> Box.test(residuals(fit23))
Box-Pierce test
data:  residuals(fit23)
X-squared = 4.7564e-05, df = 1, p-value = 0.9945
> fit23<-arima(kluster2,order=c(3,0,2))
> coeftest(fit23)
```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(> z)
ar1	1.687679	0.087070	19.3829	< 2.2e-16 ***
ar2	-0.918942	0.147839	-6.2158	5.105e-10 ***
ar3	-0.044917	0.085509	-0.5253	0.5994
ma1	-1.732129	0.050335	-34.4118	< 2.2e-16 ***
ma2	0.981748	0.045697	21.4838	< 2.2e-16 ***
intercept	69.993984	3.016921	23.2005	< 2.2e-16 ***


```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3. Pemodelan ARIMA pada Kluster 3

```
> fit17=arima(datamodel3,c(2,0,3))
> fit17
Call:
arima(x = datamodel3, order = c(2, 0, 3))

Coefficients:
ar1      ar2      ma1      ma2      ma3  intercept
1.6338 -0.6645 -1.7197  0.5798  0.1828   69.3148
s.e.  0.1186   0.1159   0.1367   0.2312  0.1073    4.9232
sigma^2 estimated as 1832:  log likelihood = -684.03,  aic = 1382
.06
> Box.test(residuals(fit17))
Box-Pierce test
data:  residuals(fit17)
X-squared = 0.0076277, df = 1, p-value = 0.9304
> fit17<-arima(kluster3,order=c(2,0,3))
> coeftest(fit17)
```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(> z)
ar1	1.0008930	0.0088495	113.1010	<2e-16 ***
ar2	-0.9988185	0.0042520	-234.9081	<2e-16 ***
ma1	-1.0518161	0.0933477	-11.2677	<2e-16 ***
ma2	1.0522392	0.1009486	10.4235	<2e-16 ***
ma3	-0.0670047	0.0845146	-0.7928	0.4279
intercept	69.5830161	3.1049832	22.4101	<2e-16 ***

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4. Pemodelan ARIMA pada Kluster 4

```
> fit44=arima(datamodel4,c(6,0,2))
> fit44
Call:
arima(x = datamodel4, order = c(6, 0, 2))

Coefficients:
ar1      ar2      ar3      ar4      ar5      ar6      ma1      ma2
intercept
1.0737 -0.8688 -0.0880  0.0168  0.1867 -0.1601 -1.0623  1.000
0 63.2189
s.e.  0.0901   0.1267   0.1484   0.1481   0.1284   0.0916   0.0322
0.0347    3.8841
sigma^2 estimated as 1592:  log likelihood = -677.03,  aic = 1374
.05
> Box.test(residuals(fit44))
Box-Pierce test
data:  residuals(fit44)
X-squared = 0.0028312, df = 1, p-value = 0.9576
> fit44<-arima(kluster4,order=c(6,0,2))
> coeftest(fit44)
```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(> z)
ar1	-0.423433	0.387519	-1.0927	0.27454
ar2	-0.281658	0.366171	-0.7692	0.44178
ar3	-0.022414	0.093119	-0.2407	0.80979
ar4	-0.162045	0.091100	-1.7788	0.07528
ar5	0.040685	0.124032	0.3280	0.74290
ar6	0.125648	0.112566	1.1162	0.26433
ma1	0.515688	0.390174	1.3217	0.18627
ma2	0.378556	0.376711	1.0049	0.31495
intercept	65.240074	3.661933	17.8157	< 2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Lampiran B : Nilai AIC ARIMA (A1) pada Keempat Kluster

Kluster 1		Kluster 2		Kluster 3		Kluster 4	
Model	AIC	Model	AIC	Model	AIC	Model	AIC
ARIMA (4,0,0)	1381,34	ARIMA (0,0,1)	1388,19	ARIMA (0,0,1)	1385,53	ARIMA (0,0,1)	1380,05
ARIMA (0,0,4)	1379,78	ARIMA (0,0,2)	1338,37	ARIMA (0,0,2)	1386,02	ARIMA (0,0,2)	1381,44
ARIMA (4,0,4)	1375,95	ARIMA (0,0,3)	1390,32	ARIMA (0,0,3)	1387,52	ARIMA (0,0,3)	1382,59
		ARIMA (0,0,4)	1390,56	ARIMA (0,0,4)	1386,79	ARIMA (0,0,4)	1380,31
		ARIMA (0,0,5)	1389,42	ARIMA (0,0,5)	1388,79	ARIMA (0,0,5)	1382,24
		ARIMA (0,0,6)	1391,2	ARIMA (0,0,6)	1390,19	ARIMA (0,0,6)	1383,98
		ARIMA (1,0,0)	1388,15	ARIMA (1,0,0)	1385,54	ARIMA (1,0,0)	1379,96
		ARIMA (1,0,1)	1389,91	ARIMA (1,0,1)	1386,51	ARIMA (1,0,1)	1381,92
		ARIMA (1,0,2)	1390,35	ARIMA (1,0,2)	1386,79	ARIMA (1,0,2)	1383,26
		ARIMA (1,0,3)	1387,36	ARIMA (1,0,3)	1388,7	ARIMA (1,0,3)	1378,2
		ARIMA (1,0,4)	1388,29	ARIMA (1,0,4)	1388,79	ARIMA (1,0,4)	1382,27
		ARIMA (1,0,5)	1390,1	ARIMA (1,0,5)	1382,5	ARIMA (1,0,5)	1384,22
		ARIMA (1,0,6)	1392,05	ARIMA (1,0,6)	1383,06	ARIMA (1,0,6)	1385,85
		ARIMA (2,0,0)	1388,83	ARIMA (2,0,0)	1386,48	ARIMA (2,0,0)	1381,83
		ARIMA (2,0,1)	1390,57	ARIMA (2,0,1)	1386,55	ARIMA (2,0,1)	1383,63
		ARIMA	1388,01	ARIMA	1383,07	ARIMA	1380,91

		(2,0,2)		(2,0,2)		(2,0,2)	
		ARIMA (2,0,3)	1389,87	ARIMA (2,0,3)	1382,06	ARIMA (2,0,3)	1374,86
		ARIMA (2,0,4)	1390,85	ARIMA (2,0,4)	1382,21	ARIMA (2,0,4)	1383,88
		ARIMA (2,0,5)	1390,86	ARIMA (2,0,5)	1383,09	ARIMA (2,0,5)	1373,56
		ARIMA (2,0,6)	1392,89	ARIMA (2,0,6)	1384,97	ARIMA (2,0,6)	1383,73
		ARIMA (3,0,0)	1390,19	ARIMA (3,0,0)	1387,98	ARIMA (3,0,0)	1382,19
		ARIMA (3,0,1)	1386,45	ARIMA (3,0,1)	1388,2	ARIMA (3,0,1)	1383,6
		ARIMA (3,0,2)	1388,04	ARIMA (3,0,2)	1390,43	ARIMA (3,0,2)	1397,2
		ARIMA (3,0,3)	1390,14	ARIMA (3,0,3)	1382,53	ARIMA (3,0,3)	1375,42
		ARIMA (3,0,4)	1395,15	ARIMA (3,0,4)	1390,5	ARIMA (3,0,4)	1397,45
		ARIMA (3,0,5)	1390,32	ARIMA (3,0,5)	1385,88	ARIMA (3,0,5)	1375,05
		ARIMA (3,0,6)	1392,57	ARIMA (3,0,6)	1387,83	ARIMA (3,0,6)	1377,23
		ARIMA (4,0,0)	1390,56	ARIMA (4,0,0)	1385,91	ARIMA (4,0,0)	1382,33
		ARIMA (4,0,1)	1388,01	ARIMA (4,0,1)	1387,86	ARIMA (4,0,1)	1383,82
		ARIMA (4,0,2)	1390	ARIMA (4,0,2)	1382,33	ARIMA (4,0,2)	1383,55
		ARIMA (4,0,3)	1392,03	ARIMA (4,0,3)	1384,28	ARIMA (4,0,3)	1385,55
		ARIMA (4,0,4)	1393,42	ARIMA (4,0,4)	1392,56	ARIMA (4,0,4)	1374,91
		ARIMA (4,0,5)	1392,21	ARIMA (4,0,5)	1387,89	ARIMA (4,0,5)	1385,04
		ARIMA (4,0,6)	1394,4	ARIMA (4,0,6)	1392	ARIMA (4,0,6)	1374,64
		ARIMA (5,0,0)	1390,03	ARIMA (5,0,0)	1387,85	ARIMA (5,0,0)	1382,87
		ARIMA (5,0,1)	1389,97	ARIMA (5,0,1)	1387,6	ARIMA (5,0,1)	1384,58
		ARIMA (5,0,2)	1392,98	ARIMA (5,0,2)	1384,32	ARIMA (5,0,2)	1375,06
		ARIMA (5,0,3)	1390,72	ARIMA (5,0,3)	1386,09	ARIMA (5,0,3)	1376,41
		ARIMA	1392,09	ARIMA	1387,8	ARIMA	1384,96

		(5,0,4)		(5,0,4)		(5,0,4)	
		ARIMA (5,0,5)	1392,88	ARIMA (5,0,5)	1383,87	ARIMA (5,0,5)	1382,72
		ARIMA (5,0,6)	1394,64	ARIMA (5,0,6)	1385,59	ARIMA (5,0,6)	1378,61
		ARIMA (6,0,0)	1391,82	ARIMA (6,0,0)	1389,84	ARIMA (6,0,0)	1383,87
		ARIMA (6,0,1)	1391,68	ARIMA (6,0,1)	1388,8	ARIMA (6,0,1)	1385,44
		ARIMA (6,0,2)	1392,71	ARIMA (6,0,2)	1390,92	ARIMA (6,0,2)	1374,05
		ARIMA (6,0,3)	1392,63	ARIMA (6,0,3)	1387,87	ARIMA (6,0,3)	1370,74
		ARIMA (6,0,4)	1394,64	ARIMA (6,0,4)	1383,81	ARIMA (6,0,4)	1386,98
		ARIMA (6,0,5)	1396,64	ARIMA (6,0,5)	1385,59	ARIMA (6,0,5)	1384,89
		ARIMA (6,0,6)	1397,24	ARIMA (6,0,6)	1381,57	ARIMA (6,0,6)	1382,36