PENGGUNAAN METODE *RESAMPLING BOOTSTRAP*
UNTUK DATA SIMULASI *TIME SERIES* MODEL *ARIMA*

SKRIPSI

Oleh:

Andika Monalisa
NIM 081810101016

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2016

PENGGUNAAN METODE *RESAMPLING BOOTSTRAP* UNTUK DATA SIMULASI *TIME SERIES* MODEL *ARIMA*

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan Program Studi Matematika (S1)
dan mencapai gelar Sarjana Sains

Oleh:

Andika Monalisa
NIM 081810101016

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2016

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Ibunda Sri Mulyawati, Ayahanda Rasidi, S.Pd., nenekku Supatmi, kakekku Sabi'in dan Arifin serta Adikku Ananda Akbar Rosyidi yang memberikan kasih sayang, doa dan restu dalam perjalanan hidupku;
2. Mas Sudarsono engkau adalah sumber semangatku yang selalu memberikan motivasi tanpa henti;
3. Adik-adikku Siti Mustaqimah, Siska Devianti, Riska Isal Noviati dan Bayu Prasetyo semoga kalian bisa lebih baik dari kakakmu ini;
4. Guru-guru sejak taman kanak-kanak sampai dengan perguruan tinggi yang telah memberikan ilmu dan membimbing dengan penuh kesabaran;
5. Almamater Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

MOTTO

Sesungguhnya sesudah kesulitan itu ada kemudahan. Maka apabila kamu telah selesai (dari sesuatu urusan), kerjakanlah dengan sungguh-sungguh (urusan yang lain) dan hanya kepada Tuhanmulah hendaknya kamu berharap.

(terjemahan Surat *Al-Insyirah* ayat 6-8)*)

---

* Departemen Agama Republik Indonesia. 1993. *Al Qur'an dan Terjemahannya.* Semarang. CV Alwaah.

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Andika Monalisa

NIM : 081810101016

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul "Penggunaan Metode *Resampling Bootstrap* Untuk Data Simulasi *Time Series* Model *ARIMA*" 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 manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, Februari 2016

Yang menyatakan,

Andika Monalisa

NIM. 081810101016

SKRIPSI

PENGGUNAAN METODE *RESAMPLING BOOTSTRAP* UNTUK DATA SIMULASI *TIME SERIES* MODEL *ARIMA*

Oleh

Andika Monalisa
NIM 081810101016

Pembimbing

Dosen Pembimbing Utama : Prof. Drs. I Made Tirta, M.Sc., Ph.D

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

PENGESAHAN

Skripsi berjudul "Penggunaan Metode *Resampling Bootstrap* Untuk Data Simulasi *Time Series* Model *ARIMA*" telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Tim Penguji

Ketua,	Sekretaris,
Prof. Drs. I Made Tirta, M.Sc., Ph.D NIP. 195912201985031002	Dr. Alfian Futuhul Hadi, S.Si., M.Si NIP. 197407192000121001
Anggota I,	Anggota II,
Dian Anggraeni, S.Si., M.Si NIP. 198202162006042002	M. Ziaul Arif, S.Si., M.Sc NIP. 198501112008121002

Mengesahkan
Dekan,

Drs. Sujito, Ph.D.
NIP. 196102041987711001

RINGKASAN

Penggunaan Metode *Resampling Bootstrap* Untuk Data Simulasi *Time Series* Model *ARIMA*; Andika Monalisa; 081810101016; 2016; 57 halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Time series adalah serangkaian nilai-nilai variabel yang disusun berdasarkan waktu. Analisis *time series* mempelajari pola gerakan nilai-nilai variabel pada suatu interval waktu (misalnya minggu, bulan, tahun) yang diatur. Dari analisis *times series* dapat diperoleh ukuran-ukuran yang dapat digunakan untuk membuat keputusan pada saat ini, untuk peramalan dan untuk merencanakan masa depan. Ada metode lain untuk meramalkan masa depan yang disebut model regresi. Keunggulannya adalah bahwa penyusunan model regresi didasarkan pada teori atau logika ekonomi, sementara model *time series* dapat dikatakan tanpa landasan teori, namun semua metode didasarkan pada asumsi bahwa pola lama akan terulang. Analisis *time series* yang dibicarakan di sini didasarkan pada model *time series* klasik (dekomposisi) (Akhyasrinuki, 2011).

Model *time series* mengasumsikan bahwa data masukan harus stasioner. Apabila data masukan tidak stasioner perlu dilakukan penyesuaian untuk menghasilkan data yang stasioner. Salah satu cara yang umum dipakai adalah metode pembedaan (*differencing*). Metode ini dilakukan dengan cara mengurangi nilai data pada suatu periode dengan nilai data periode sebelumnya.

Terdapat beberapa model *time series,* diantaranya adalah ARIMA. Metode *ARIMA* berbeda dari metode peramalan lain karena metode ini tidak mensyaratkan suatu pola data tertentu, sehingga model dapat dipakai untuk semua tipe pola data. Metode *ARIMA* akan bekerja baik jika data dalam *time series* yang digunakan bersifat dependen atau berhubungan satu sama lain secara statistik. Secara umum, model

ARIMA ditulis dengan *ARIMA* (*p, d, q*) yang artinya model *ARIMA* dengan derajat *AR* (*p*), derajat pembeda *d*, dan derajat *MA* (*q*).

Metode *bootstrap* adalah metode berbasis *resampling* data sampel dengan syarat pengembalian pada datanya dalam menyelesaikan statistik ukuran suatu sampel dengan harapan sampel tersebut mewakili data populai sebenarnya, biasanya ukuran resampling diambil secara ribuan kali agar dapat mewakili data populasinya. Metode ini bagus sekali untuk ukuran data sampel yang relatif kecil {dalam bukunya Walpole data kecil yaitu n<30}.

Penelitian dilakukan dalam beberapa tahap, tahap pertama melakukan *simulasi* data *time series*. Tahap berikutnya adalah mengecek bahwa data yang diambil benar-benar merupakan data *time series*. Berikutnya, dilakukan resampling bootstrap untuk memperluas ruang sample, dengan harapan bahwa akan menghasilkan data yang akurat.terakhir dilakukan pengecekan apakah data yang telah diresampling tetap pada model *ARIMA*.

Berdasarkan hasil analisis *time series* model *ARIMA* dengan paket *bootstrap tsboot* pada program *R* dan pembahasan dapat diambil kesimpulan bahwa dari tiga contoh simulasi time series model *ARIMA* dengan nilai *standar deviasi* yang berbeda yaitu *sqrt(0,1796), sqrt(3,1796)* dan *sqrt(0.0096)* didapatkan model terbaik pada simulasi dengan *sd=sqrt(0,0096)* yaitu *AIC*=-35,58 dan *log likelihood*=23,79 sebelum dilakukan *bootstrap* dan *AIC*=-1402,56 dan *log likelihood*=707,28 setelah dilakukan *bootstrap*.

PRAKATA

Alhamdulillah, segala puji hanya bagi Allah atas segala rahmat dan karunia-Nya sehingga dapat terselesaikannya skripsi dengan judul "Penggunaan Metode *Resampling Bootstrap* Untuk Data Simulasi *Time Series* Model *ARIMA*". Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan strata satu (S1) pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bentuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Prof. Drs. I Made Tirta, M.Sc., Ph.D selaku Dosen Pembimbing Utama, Dr. Alfian Futuhul Hadi, S.Si., M.Si selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi ini;
2. Dian Anggraeni, S.Si., M.Si selaku Dosen Penguji I dan M. Ziaul Arif, S.Si, M.Sc selaku Dosen Penguji II yang telah banyak memberikan kritik dan saran dalam penulisan skripsi ini;
3. Seluruh staf pengajar Jurusan Matematika Fakultas MIPA Universitas Jember yang telah memberikan ilmu serta bimbingannya sehingga penulis dapat menyelesaikan skripsi ini;
4. Ibunda Sri Mulyawati, Ayahanda Rasidi, S.Pd., nenekku Supatmi, kakekku Sabi'in dan Arifin serta Adikku Ananda Akbar Rosyidi yang memberikan kasih sayang, doa dan restu dalam perjalanan hidupku;
5. Mas Sudarsono engkau adalah sumber semangatku yang selalu memberikan motivasi tanpa hentinya memberikan semangat dan dukungannya;
6. Mikail Mintin dan Eko Sugiarto Terima kasih atas perhatian dan semangat dari kalian hingga skripsi ini selesai.

7. Teman-teman angkatan 2008 Jurusan Matematika yang tidak bisa disebutkan satu persatu terima kasih atas kebersamaan dan motivasinya;
8. Sahabatku Erna Yulianti, Eka Farista, Prasanti Mia dan Vianda Nuning Fitriani, terima kasih kalian telah memberikan semangat yang lebih kepada sahabatmu ini hingga skripsi ini selesai.
9. Temanku Novika Herawati, Rafiantika M.P, Dimas terimakasih atas bantuannya.
10. Teman-teman kosan Jawa VI No. 2A serta kosan CANTIQUE (Juwaria, Ike Diah, Siti Fatimah, dan savitri) yang memberikan dukungan dan motivasi;

Penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan skripsi ini. Akhirnya penulis berharap, semoga skripsi ini dapat bermanfaat.

Jember, Februari 2016 Penulis

DAFTAR ISI

Halaman

DAFTAR TABEL

DAFTAR GAMBAR

Halaman

DAFTAR LAMPIRAN

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Pengamatan adalah aktivitas yang dilakukan makhluk cerdas, terhadap suatu proses atau objek dengan maksud merasakan dan kemudian memahami pengetahuan dari sebuah *fenomena* berdasarkan *pengetahuan* dan *gagasan* yang sudah diketahui sebelumnya, untuk mendapatkan informasi-informasi yang berupa data pengamatan yang dibutuhkan untuk melanjutkan suatu penelitian, meningkatkan daya pikir, memperluas wawasan, dan melatih kepekaan. Di dalam penelitian, pengamatan dapat dilakukan dengan tes, kuesioner, rekaman gambar dan rekaman suara.

Data pengamatan dapat dikelompokkan menjadi data pengamatan yang dipengaruhi oleh waktu (*time series*) dan data pengamatan yang tidak dipengaruhi oleh waktu. Data *time series* dibangun secara berkala (berurutan dalam waktu). Data time series sendiri harus memenuhi asumsi stasioner yaitu memiliki nilai rata-rata dan varian yang *konstan* (tetap tidak berubah secara terus menerus) dalam arti data yang tidak mengalami kenaikan dan penurunan. Jika suatu data time series tidak memenuhi asumsi *stasioner* maka data tersebut harus di-*differencing* terlebih dahulu untuk mencapai stasioneritas yaitu dengan membangun unsur-unsur ensemble untuk mempertahankan bentuk dasar dan struktur ketergantungan waktu dari *Autocorrelation Function* (ACF) dan *Partial Autocorrelation Function* (PACF) dari serial aslinya.

Salah satu permasalahan penting dalam analisis deret waktu *(time series)* adalah peramalan nilai-nilai suatu variabel pada masa akan datang atau lebih dikenal dengan istilah *forecasting. Forecasting* merupakan hal yang mendasar dalam sebuah perencanaan dan pengambilan kebijakan dalam suatu instansi karena adanya ketidakpastian dari nilai-nilai suatu variabel pada masa yang akan datang. Metode yang paling sering digunakan dalam pemodelan deret waktu untuk peramalan adalah *Autoregressive Integrated Moving Average* (*ARlMA*) Box-Jenkins. Agar model *ARIMA* Box-Jenkins menghasilkan ramalan yang optimal, model tersebut harus memenuhi asumsi residual *white noise* dan berdistribusi normal. Yang menjadi masalah adalah kadang dijumpai adanya model *time series* dengan *residual* yang *white noise* tetapi tidak berdistribusi normal walaupun data telah ditransformasi dan dibedakan *(difference).* Sebagai akibat dari tidak normalnya residual maka model tidak dapat digunakan dalam inferensi. Salah satu cara estimasi parameter jika residual tidak memenuhi asumsi berdistribusi normal adalah dengan metode *bootstrap* yang pertama kali diperkenalkan oleh Efron pada tahun 1979. Pada metode *bootstrap time series* digunakan dua pendekatan yaitu *residual resampling* dan *moving blocks bootstrap* (Efron dan Tibshirani,1993). *Resampling* yang dilakukan bertujuan untuk memperkecil *standard error* parameter yang ditaksir. *Bootstrap* adalah metode resampling yang digunakan untuk membangun data baru dari data yang memiliki ukuran sample kecil.

Dalam jurnal yang di tulis oleh Siana Halim dan Herman Mallian(2006) yang menulis mengenai penggunaan *Bootstrap* untuk data *dependen* dan *independen.* Penulis tertarik untuk menulis skripsi yang berjudul “Penggunaan metode resampling *Bootstrap* untuk Data Simulasi *Time Series* Model *ARIMA*” agar dapat lebih memberikan informasi tentang penggunaan *Bootstrap*.

1.2 Perumusan Masalah

Berdasarkan latar belakang diatas, maka permasalahan yang akan diteliti dalam penelitian ini adalah:

a. Apa kelebihan penggunaan *bootsrap* untuk data *time series* model *ARIMA?*
b. Bagaimanakah implementasi *bootstrap* pada data simulasi *time series* model *ARIMA* pada progam *R*?

1.3 Tujuan

Berdasarkan rumusan masalah, tujuan yang ingin dicapai penulis yaitu menganalisis *time series* model *ARIMA* dengan *bootstrap* dengan menggunakan paket *tsboot* pada program *R*.

1.4 Batasan Masalah

Dalam skripsi ini akan diberikan batasan masalah yaitu data yang akan digunakan merupakan data simulasi *time series* model *ARIMA.* Tujuan dari asumsi ini yaitu untuk membatasi cakupan bahasan agar tidak terlalu meluas.

1.5 Manfaat

Adapun manfaat yang dapat diperoleh dari penulisan skripsi ini yaitu:

a. Dapat digunakan sebagai acuan bagi masalah-masalah yang berkaitan dengan penggunaan *bootstrap* untuk data *time series* model *ARIMA*;
b. Menambah pengetahuan tentang metode *bootstrap* pada program *R*;
c. Menambah wawasan keilmuan dalam analisis statistik untuk peramalan, khususnya menyelesaikan kasus pemodelan deret waktu yang residualnya tidak memenuhi asumsi berdistribusi normal.

BAB 2. TINJAUAN PUSTAKA

2.1 Deret Waktu (*Time Series*)

Deret waktu *(time series)* adalah suatu himpunan pengamatan yang dibangun secara berurutan dalam waktu. Waktu atau periode yang dibutuhkan untuk melakukan suatu peramalan itu biasanya disebut sebagai *lead time* yang bervariasi pada tiap persoalan. Berdasarkan himpunan pengamatan yang tersedia maka *time series* dikatakan kontinu jika himpunan pengamatan tersebut adalah kontinu dan dikatakan diskrit bila himpunan pengatamatan tersebut juga diskrit

Pembangunan data untuk time series diskrit dapat dilakukan dengan cara 2 macam, yaitu

1. Melalui sampling dari *time series* kontinu, artinya data yang kontinu diambil sampelnya dalam interval waktu yang sama.
2. Melalui akumulasi suatu peubah dalam suatu waktu tertentu. Misalnya curah hujan yang biasanya diakumulasikan melalui suatu periode waktu tertentu (hari, bulan,dst) (Halim, 2006).

Time series adalah serangkaian nilai-nilai variabel yang disusun berdasarkan waktu. Analisis *time series* mempelajari pola gerakan nilai-nilai variabel pada suatu interval waktu (misalnya minggu, bulan, tahun) yang diatur. Dari analisis *times series* dapat diperoleh ukuran-ukuran yang dapat digunakan untuk membuat keputusan pada saat ini, untuk peramalan dan untuk merencanakan masa depan. Ada metode lain untuk meramalkan masa depan yang disebut model regresi. Keunggulannya adalah bahwa penyusunan model regresi didasarkan pada teori atau logika ekonomi, sementara model *time series* dapat dikatakan tanpa landasan teori, namun semua metode didasarkan pada asumsi bahwa pola lama akan terulang. Analisis *time series* yang dibicarakan di sini didasarkan pada model *time series* klasik (dekomposisi) (Akhyasrinuki, 2011).

Time Series adalah set statistik, biasanya dikumpulkan secara berkala. Data *time series* yang terjadi secara alami di banyak area aplikasi.

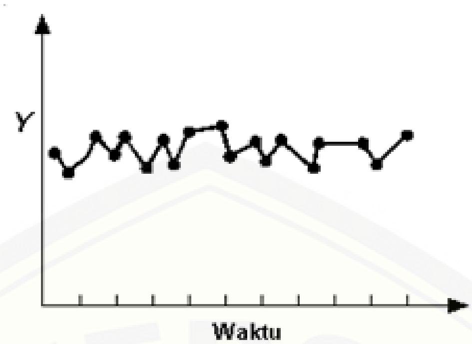
- ekonomi - misalnya, data bulanan untuk pengangguran, penerimaan rumah sakit, dll.
- keuangan - misalnya, kurs harian, harga saham, dll.
- lingkungan - misalnya, curah hujan harian, pembacaan kualitas udara.
- Obat - misalnya, aktivitas gelombang otak EKG setiap 2-8 detik.

Metode *time series* analisis pra-tanggal tersebut untuk proses stokastik umum dan Markov Chains. Tujuan dari analisis *time series* adalah untuk menggambarkan dan meringkas data *time series*, model-dimensi rendah fit, dan membuat perkiraan. Kami menulis seri bernilai real kami sebagai pengamatan. . . ,$X_{-2}, X_{-1}, X_0, X_1, X_2$ .., Sebuah ganda terbatas urutan acak variabel bernilai real diindeks oleh Z.

Analisis data *time series* mengidentifikasi pola historis (dengan menggunakan waktu sebagai rujukan), kemudian membuat prediksi dengan menggunakan ekstrapolasi berdasarkan waktu untuk pola-pola tersebut. Sebuah model *time series* mengasumsikan bahwa beberapa pola atau kombinasi pola akan berulang sepanjang waktu. Jadi, dengan mengidentifikasikan dan mengekstrapolasi itu, prediksi untuk periode-periode berikutnya dapat dikembangkan.

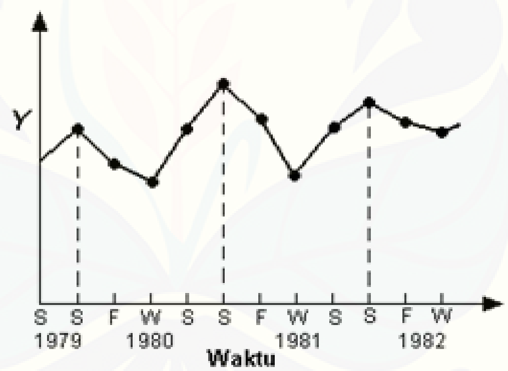
Pola data *time series* ada bermacam-macam. Pola data biasanya tidak ideal mempunyai garis yang halus, tetapi akan selalu mempunyai tingkat random disekitarnya. Kerandoman ini diakibatkan oleh fluktuasi data yang tidak bisa diprediksi.

1. Pola Data Konstan (horisontal), yaitu apabila data berfluktuasi di sekitar rata-rata secara stabil. Polanya berupa garis horizontal. Pola seperti ini terdapat dalam jangka pendek atau menengah, jarang sekali suatu variabel memiliki pola konstan dalam jangka panjang.



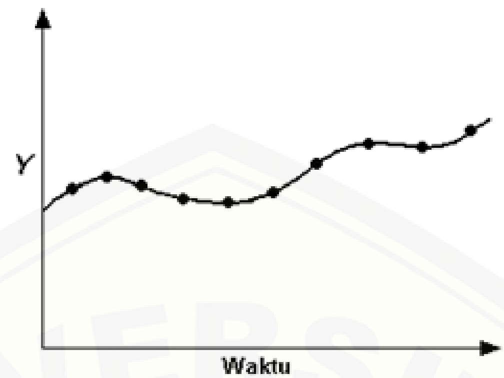
Gambar 2.1 Pola Data Horisontal

2. Pola Data Musiman, yaitu apabila polanya merupakan gerakan berulang-ulang secara teratur dalam setiap periode tentu, misalnya tahunan, semesteran, kuartalan, bulanan atau mingguan. Pola ini berhungan dengan faktor iklim / cuacu atau faktor yang dibuat manusia seperti liburan.



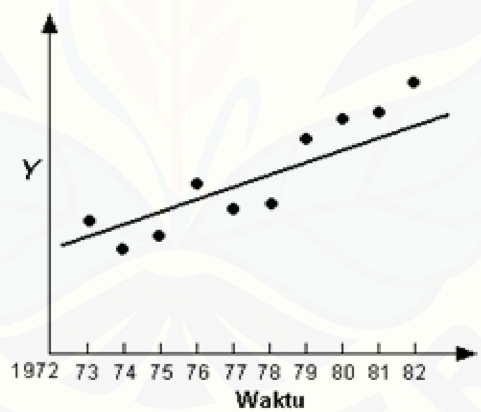
Gambar 2.2 Pola Data Musiman

3. Pola Data Siklus, yaitu apabila data dipengaruhi oleh fluktuasi ekonomi jangka panjang, seperti daur hidup bisnis. Perbedaan utama antara pola data musiman dan siklus adalah pola musiman mempunyai panjang gelombang yang tetap dan terjadi pada jarak waktu tetap, sedangkan pola siklus memiliki durasi yang lebih panjang dan bervariasi.



Gambar 2.3 Pola Data Siklus

4. Pola Data Trend, yaitu apabila data dalam jangka panjang mempunyai kecendrungan, baik yang arahnya meningkat dari waktu ke waktu maupun menurun. Pola ini disebabkan antara lain oleh bertambahnya populasi, perubahan pendapat, dan pengaruh budaya.



Gambar 2.4 Pola Data Trend

5. Pola Data Residu atau Variasi acak, yaitu apabila data tidak teratur sama sekali. Data yang bersifat residu tidak dapat digambarkan

Untuk melakukan prediksi data *time series* perlu diperhatikan tiga hal yaitu data historis, korelasi data dan kesalahan prediksi (*e*). Data historis digunakan sebagai dasar atau acuan dalam melakukan prediksi untuk N periode berikunya. Korelasi menggambarkan apa yang cenderung terjadi pada suatu nilai di suatu periode jika terjadi perubahan terhadap nilai pada periode lainnya. Sedangkan kesalahan prediksi digunakan untuk mengantasipasi fluktuatif pada data historis guna mendapatkan data prediksi yang mendekati data sebenarnya. Tingkat hubungan ini diukur dengan koefisien korelasi yang bervariasi antara +1 sampai -1. Nilai yang dekat dengan +1 menyiratkan hubungan yang kuat di antara dua nilai. Hal ini berarti ketika salah satu nilai meningkat, nilai lainnya cenderung meningkat pula. Dengan demikian, koefisien korelasi yang dekat dengan -1 menunjukan hal sebaliknya yaitu peningkatan salah satu nilai berasosiasi dengan penurunan pada nilai lainnya**(**Kahiruddin, 2012).

2.2 Stasioneritas Data

Data yang tidak stasioner memiliki rata-rata dan varian yang tidak konstan sepanjang waktu. Dengan kata lain, secara ekstrim data stasioner adalah data yang tidak mengalami kenaikan dan penurunan. Selanjutnya regresi yang menggunakan data yang tidak stasioner biasanya mengarah kepada regresi lancung. Permasalahan ini muncul diakibatkan oleh variabel (dependen dan independen) runtun waktu terdapat tren yang kuat (dengan pergerakan yang menurun maupun meningkat). Adanya tren akan menghasilkan nilai R^2 yang tinggi, tetapi keterkaitan antar variabel akan rendah (Firmansyah, 2000).

Model *time series* mengasumsikan bahwa data masukan harus stasioner. Apabila data masukan tidak stasioner perlu dilakukan penyesuaian untuk menghasilkan data yang stasioner. Salah satu cara yang umum dipakai adalah metode pembedaan (*differencing*). Metode ini dilakukan dengan cara mengurangi nilai data pada suatu periode dengan nilai data periode sebelumnya

Untuk keperluan pengujian stasioneritas, dapat dilakukan dengan beberapa metode seperti *autocorrelation function* (*correlogram*), uji akar-akar unit dan derajat integrasi.

a. Pengujian stasioneritas berdasarkan correlogram

Suatu pengujian sederhana terhadap stasioneritas data adalah dengan menggunakan fungsi koefisien autokorelasi (*autocorrelation function* / ACF). Koefisien ini menunjukkan keeratan hubungan antara nilai variabel yang sama tetapi pada waktu yang berbeda. *Correlogram* merupakan peta / grafik dari nilai ACF pada berbagai lag.

Secara matematis rumus koefisien autokorelasi adalah (Sugiharto dan Harijono, 2000:183) :

$$rk = \frac{\sum_{i=1}^{n-k}\left(Y_t - \bar{Y}\right)\left(Y_{t-k} - \bar{Y}\right)}{\sum_{i=1}^{n}\left(Y_t - \bar{Y}\right)^2} \tag{2.1}$$

Untuk menentukan apakah nilai koefisien autokorelasi berbeda secara statistik dari nol dilakukan sebuah pengujian. Suatu runtun waktu dikatakan stasioner atau menunjukkan kesalahan random adalah jika koefisien autokorelasi untuk semua lag secara statistik tidak berbeda signifikan dari nol atau berbeda dari nol hanya untuk berberapa lag didepan. Untuk itu perlu dihitung kesalahan standard dengan rumus :

$$se_{rk} = 1/\sqrt{n}$$

(2.2

Dimana n menunjukkan jumlah observasi. Dengan interval kepercayaan yang dipilih, misalnya 95 persen, maka batas signifikansi koefisien autokorelasi adalah :

$$-Z_{\alpha/2} x Se_{rk} s.d Z_{\alpha/2} x Se_{rk} \tag{2.3}$$

Suatu koefisien autokorelasi disimpulkan tidak berbeda secara signifikan dari nol apabila nilainya berada diantara rentang tersebut dan sebaliknya. Apabila koefisien autokorelasi berada diluar rentang, dapat disimpulkan koefisien tersebut signifikan, yang berarti ada hubungan signifikan antara nilai suatu variabel dengan nilai variabel itu sendiri dengan *time lag* 1 periode.

b. Uji akar-akar unit dan derajat integrasi

Sebuah tes stasioneritas (atau non-stasioneritas) yang menjadi sangat populer beberapa tahun belakangan adalah uji akar-akar unit (*unit root test*). Stasioneritas dapat diperiksa dengan mencari apakah data runtun waktu mengandung akar unit (*unit root*). Terdapat berbagai metode untuk melakukan uji akar unit diantarnya dickey-fuller, Augmented Dickey Fuller, Dickey-Fuller DLS (ERS), Philips-Perron, Kwiatkowski-Philips-Schmidt-Shin, Elliot-Rothenberg-Stock Point-Optimal, dan Ng-Perron. Dalam penelitian ini akan digunakan uji Augmented Dickey-Fuller untuk menentukan apakah suatu data runtun waktu mengandung akar unit atau bersifat non-stasioner.

Untuk memperoleh gambaran mengenai uji akar-akar ujit, ditaksir model autoregresif berikut ini dengan OLS (Insukrindo, 1994; Gujarati, 1995 dalam Firmansyah, 2000) :

$$DX_t = a_0 + a_1 BX_t + \sum_{i=1}^{k} b_i B^i DX_t \tag{2.4}$$

$$DX_t = a_0 + a_1 T + a_2 BX_t + \sum_{i=1}^{k} d^i B_i DX_i \tag{2.5}$$

Dimana, $DX_t = X_t - X_{t-t}, BX = X_{t-t}$, T = tren waktu, X_t = variabel yang diamati pada periode t. Selanjutnya dihitung statistik ADF. Nilai ADF digunakan untuk uji hipotesis bahwa $a1 = 0$ dan $c2 = 0$ ditunjukkan oleh nilai t statistik hitung pada koefisien BXt pada persamaan diatas. Jumlah kelambanan k ditentukan oleh $k = n^{1/5}$, dimana n = jumlah observasi. Nilai kritis (tabel) untuk kedua uji terkait dapat dilihat pada Fuller, 1976; Guilky dan Schmidt, 1989 (Insukrindo, 1994:130

dalam Firmansyah, 2000). Runtun waktu yang diamati stasioner jika memiliki nilai ADF lebih besar dari nilai kritis. Beberapa piranti lunak ekonometrika seperti EViews, SPlus, dan R menyediakan nilai kritis ini setiap kali kita melakukan running data.

Uji derajat integrasi adalah uji yang dilakukan untuk mengetahui pada derajat berapakah data yang diamati stasioner. Uji ini mirip atau merupakan perluasan uji akar-akar unit, dilakukan jika data yang diamati ternyata tidak stasioner sebagaimana direkomendasikan oleh uji akar-akar unit. Bentuk umum regresinya adalah :

$$D2X_t = e_0 + e_t BDX_t + \sum_{i=1}^{k} f_i B^i D2X_t \tag{2.6}$$

$$D2X_t = g_0 + g_1 T + g_2 BDX_t + \sum_{i=1}^{k} h_i B^i D2X_t \tag{2.7}$$

Dimana, $D2X_t = DX_t - DX_{t-1}$, $BDX_t = DX_{t-1}$, selanjutnya pengujiannya sama dengan uji akar-akar unit. Jika pada derajat pertama ini data masih belum stasioner, maka uji integrasi perlu dilanjutkan pada derajat berikutnya sampai memperoleh suatu kondisi stasioner.

2.3 Model *Autoregressive Integrated Moving Average* (ARIMA)

Beberapa model ARIMA yang dapat digunakan pada data *time series* yaitu:

1. **Model *Autoregressive* (AR)**

Autoregressive adalah suatu bentuk regresi tetapi bukan yang menghubungkan variabel tak bebas, melainkan menghubungkan nilai-nilai sebelumnya pada *time tag* (selang waktu) yang bermacam-macam. Jadi suatu model *Autoregressive* akan menyatakan suatu ramalan sebagai fungsi nilai-nilai sebelumnya dari *time series* tertentu (Makridakis, 1995:513).

Model *Autoregressive* (*AR*) dengan order *p* dinotasikan dengan *AR (p).* Bentuk umum model *AR (p)* adalah:

$$X_t = \emptyset_1 X_{t-1} + \ \dots + \emptyset_p X_{t-p} + \ \varepsilon_p \qquad (2.8)$$

dengan,

X_t : nilai variabel pada waktu ke-t

$X_{t-1}, X_{t-2}, \dots., X_{t-p}$: nilai masa lalu dari *time series* yang bersangkutan pada waktu t-1, t-2, ..., t-p

$\emptyset_1$: koefisien regresi, i: 1, 2, 3, ..., p.

ε_p : nilai error pada waktu ke-t

p : order *AR*

Persamaan (2.8) dapat ditulis menggunakan operator B (*backshift*):

$$X_t = \emptyset_1 B X_t + \emptyset_2 B^2 X_t + \ \dots + \emptyset_p B^p X_t + \ \varepsilon_t \qquad (2.9)$$

$\emptyset \ B \ X_t = \ \varepsilon_t$ dimana $\emptyset \ B = 1 - \ \emptyset_1 B - \emptyset_2 \ B^2 - \cdots -$ $\emptyset_p B^p. \ disebut \ operator \ AR$ (p).

Pada umumnya, order AR yang sering digunakan dalam analisis *time series* adalah p=1 atau p=2, yaitu model AR (1) dan AR (2).

Bentuk umum model Autoregressive order 1 atau AR (1), yaitu:

$$X_t = \emptyset_1 X_{t-1} + \varepsilon_t \qquad (2.10)$$

Persamaan (2.10) dapat ditulis dengan operator *backshift* (B), menjadi:

$$1 - \ \emptyset_1 B \ X_t = \ \varepsilon_t$$

Bentuk umum model *Autoregressive* order 1 atau AR (2), yaitu:

$$X_t = \emptyset_1 X_{t-1} + \emptyset_2 X_{t-2} + \varepsilon_t \qquad (2.11)$$

Persamaan (2.11) dapat ditulis dengan operator *backshift* (B), menjadi:

$$1 - \ \emptyset_1 B - \emptyset_2 \ B^2 X_t = \varepsilon_t$$

2. Model *Moving Average* (MA)

Menurut Wei (2006:47), model *Moving Average* order q dinotasikan MA (q) didefinisikan sebgai:

$$X_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}; \varepsilon_t \sim N(0, \sigma_t^2) \qquad (2.12)$$

dengan,

X_t : nilai variabel pada waktu ke-t

$\varepsilon_t, \varepsilon_{t-1}, \varepsilon_{t-2}, \dots, \varepsilon_{t-q}$: nilai-nilai *error* pada waktu *t*, *t-1*, *t-2*,... *t-q* dan ε_t diasumsikan *White noise* dan normal.

θ_i : koefisien regresi, i: $1, 2, 3, \dots, q$

ε_t : nilai *error* pada waktu ke-t

q : order *MA*

Persamaan di atas dapat ditulis menggunakan operator *backshift* (B), menjadi: $X_t = \theta(B)\, \varepsilon_t$ dengan $\theta\ B = 1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q$ merupakan operator *MA* (q). Secara umum, order *MA* yang sering digunakan dalam analisis *time series* adalah *q*=1 atau *q*=2, yaitu *MA* (1) dan *MA* (2).

Model *Moving Average* order 1 atau MA (1) secara matematis di definisikan menjadi:

$$X_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} \qquad (2.13)$$

Persamaan (2.13) dapat ditulis dengan operator B (*backshift*), menjadi:

$$X_t = (1 + \theta_1 B)\varepsilon_t$$

Sedangkan model *Moving Average* order 2 atau *MA* (2) secara matematis di definisikan

$$X_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} \qquad (2.14)$$

Persamaan (2.41) dapat ditulis dengan menggunakan operator B *(backshift)*, menjadi:

$$X_t = (1 + \theta_1 B + \theta_2 B^2)\varepsilon_t$$

3. Model *Autoregressive Moving Average* (ARMA)

Model *Auto Regressive Moving Average* (ARMA) merupakan suatu kombinasi dari model *AR* dan *MA* (Palit & Dobrivoje Popovic, 2005:28)

Bentuk umum model *ARM* (p,q), yaitu:

$X_t = \emptyset_1 X_{t-1} + \emptyset_2 X_{t-2} + \cdots + \emptyset_p X_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}$

Persamaan di atas menjadi

$$X_t = \emptyset_1 X_{t-1} + \emptyset_2 X_{t-2} + \cdots + \emptyset_p X_{t-p} = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} \quad (2.15)$$

Persamaan (2.15) dapat ditulis menggunakan operator b *(backshift)*, menjadi:

$\left(1 - \emptyset_1 B - \emptyset_2 B^2 - \cdots - \emptyset_p B^p\right) X_t = (1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q)\ \varepsilon_t$

Sehingga diperoleh

$\emptyset\ B\ X_t = \theta(B)\varepsilon_t$

dengan,

X_t : nilai variabel pada waktu ke-t

$\emptyset_i$: koefisien regresi ke-*i*, *i* = 1, 2, 3, ..., *p*

p : order *AR*

θ_i : parameter model *MA* ke-*i*, *i*=1, 2, 3, ..., *q*

ε_t : nilai *error* pada waktu ke-t

$\varepsilon_t, \varepsilon_{t-1}, \varepsilon_{t-2}, \dots, \varepsilon_{t-q}$: nilai-nilai *error* pada waktu *t, t-1, t-2,... t-q* dan ε_t diasumsikan *White noise* dan normal.

4. Model *Autoregressive Integrated Moving Average* (ARIMA)

Secara umum model *ARIMA (p, d, q)* untuk suatu data time series X_t adalah sebagai berikut(Pankratz, 1983: 99):

$$\emptyset\ B(1 - B)^d X_t = \ \theta(B)\varepsilon_t;\ \varepsilon_t \sim N(0, \sigma_t^2) \quad (2.16)$$

Persamaan (2.16) dapat ditulis menggunakan operator B *(backshift)*, menjadi:

$1 - B^d 1 - \emptyset_1 B - \emptyset_2 B^2 - \cdots - \emptyset_p B^p X_t = (1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q)\ \varepsilon_t$

Sehingga diperoleh

$1 - B^d X_t - \emptyset_1 X_{t-1} - \emptyset_2 X_{t-2} - \cdots - \emptyset_p X_{t-p} = +\theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}$

dengan,

X_t : nilai variabel pada waktu ke-t

B : operator *Backshift*

$1-B^dX_t$: *time series* yang stasioner pada pembedaan ke-d

ε_t : nilai *error* pada waktu ke-t

p : order *AR*

d : order pembedaan

q : order *MA*

Apabila pembedaan pertama dilakukan terhadap model agar menjadi stasioner maka model menjadi *ARIMA* (1, 1, 1) didefinisikan sebagai berikut:

$$1-B\ 1-\emptyset_1BX_t=(1+\theta_1B)\ \varepsilon_t \qquad (2.17)$$

5. Prosedur Pembentukan *ARIMA*

Metode *ARIMA* berbeda dari metode peramalan lain karena metode ini tidak mensyaratkan suatu pola data tertentu, sehingga model dapat dipakai untuk semua tipe pola data. Metode *ARIMA* akan bekerja baik jika data dalam *time series* yang digunakan bersifat dependen atau berhubungan satu sama lain secara statistik. Secara umum, model *ARIMA* ditulis dengan *ARIMA* (*p*, *d*, *q*) yang artinya model *ARIMA* dengan derajat *AR* (*p*), derajat pembeda *d*, dan derajat *MA* (*q*). Langkah-langkah pembentukan model secara iteratif adalah sebagai berikut:

a. Identifikasi Model

Hal pertama yang dilakukan pada tahap ini adalah apakah *time series* bersifat stasioner atau nonstasioner dan bahwa aspek-aspek *AR* dan *MA* dari model *ARIMA* hanya berkenaan dengan *time series* yang stasioner (Makridakis, 1995: 381). Kestasioneran suatu *time series* dapat dilihat dari plot *ACF* yaitu koefisien autokorelasinya menurun menuju nol dengan cepat, biasanya setelah *lag* ke-2 atau ke-3. Bila data tidak stasioner maka dapat dilakukan pembedaan atau *differencing*, orde pembedaan sampai deret menjadi stasioner dapat digunakan untuk menentukan nilai *d* pada *ARIMA(p,d,q)* .

Model *AR* dan *MA* dari suatu *time series* dapat dilakukan dengan melihat grafik *ACF* dan *PACF*.

1) Jika terdapat *lag autokorelasi* sebanyak *q* yang berbeda dari nol secara signifikan maka prosesnya adalah *MA(q)* .

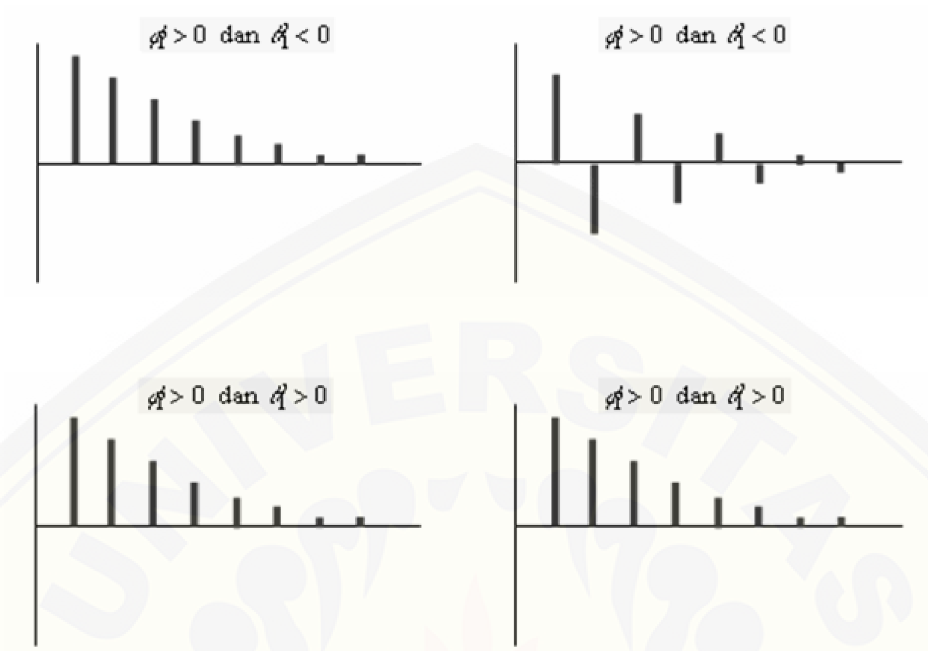
2) Jika terdapat *lag autokorelasi* parsial sebanyak *p* yang berbeda dari nol secara signifikan maka prosesnya adalah *AR(p)*. Secara umum jika terdapat *lag* autokorelasi parsial sebanyak yang berbeda dari nol secara signifikan, terdapat *lag* autokorelasi sebanyak *q* yang berbeda dari nol secara signifikan dan *d* pembedaan maka prosesnya adalah *ARIMA (p, d, q)*. Tabel 2.1 merupakan identifikasi order model *AR* dan *MA* dengan plot *ACF* dan *PACF*, yaitu:

Tabel 2.1 Identifikasi Order Model *ARIMA* dengan Pola Grafik *ACF* dan *PACF*

No	Model	ACF	PACF
1.	*AR (p)*	Menurun secara bertahap menuju ke-0	Menuju ke-0 setelah *lag* ke-*p*
2.	*MA (q)*	Menuju ke-0 setelah *lag* ke-*q*	Menurun secara bertahap menuju ke-0
3.	*ARMA (p,q)*	Menurun secara bertahap menuju ke-0	Menurun secara bertahap menuju ke-0

Dari tabel 2.1 Dapat dijelaskan sebagai berikut:

1. Jika plot *ACF* menurun secara bertahap menuju ke-0 dan plot *PACF* menuju ke-0 setelah *lag-p*, maka dugaan modelnya adalah *AR* (*p*).
2. Jika plot *ACF* menuju ke-0 setelah *lag-q* dan plot *PACF* menurun secara bertahap menuju ke-0, maka dugaan modelnya adalah *MA* (*q*)
3. Jika plot *ACF* dan plot *PACF* menurun secara bertahap menuju ke-0, maka dugaan modelnya adalah *ARMA* (*p*,*q*).



Gambar 2.5 Identifikasi Order Model *ARIMA* dengan Pola Grafik *ACF* dan *PACF*

b. Estimasi Parameter

Langkah berikutnya setelah menetapkan model sementara adalah estimasi parameter model. Salah satu metode yang digunakan yaitu *maximum likelihood*, untuk menduga parameter model *ARIMA* yaitu dan θ. Untuk fungsi *likelihood* nilai-nilai parameter yang memaksimalkan nilai fungsi *likelihood* disebut dugaan *maximum likelihood.* Penurunan fungsi *likelihood* pada suatu model *time series*, dapat digambarkan dengan mempertimbangkan model *ARMA* (Hamilton dan James, 1994: 143).

Diberikan *ARMA(p.q)* sebagai berikut:

$$X_t = c + \emptyset_1 X_{t-1} + \emptyset_2 X_{t-2} + \cdots + \emptyset_p X_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}$$

Dimana $\varepsilon_t \sim N\ 0, \sigma^2$ dan vektor populasi parameter yang akan diestimasi adalah

$\theta = \left(c, \emptyset_1, \emptyset_2, \dots, \emptyset_p, \theta_1, \theta_2, \dots, \theta_q, \sigma^2\right)$ misalkan $X_0 = (X_0, X_{-1}, \dots, X_{-p+1})$

Dan

$\varepsilon_0 = (\varepsilon_0, \varepsilon_{-1}, \dots, \varepsilon_{-q+1})$ adalah nilai awal yang digunakan untuk memperoleh estimator parameter *ARMA*.

$(\varepsilon_0, \varepsilon_{-1}, \dots, \varepsilon_{-q+1})$ dapat dihitung dari $X_0, X_1, \dots, X_T$ oleh iterasi pada

$$\varepsilon_t = X_t - c - \emptyset_1 X_{t-1} - \emptyset_2 X_{t-2} - \cdots - \emptyset_p X_{t-p} - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \cdots - \theta_q \varepsilon_{t-q} \quad (2.18)$$

Untuk t = 1, 2, ..., T

Estimator parameter maximum *likelihood* dapat diperoleh dengan memaksimalkan fungsi *likelihood* bersyaratnya dengan fungsi densitasnya.

$$f\ \varepsilon_t = \frac{1}{2\pi\sigma^2}\ exp^{\frac{\varepsilon_t^2}{2\sigma^2}}$$

Sehingga

$$L\theta = f_{X_t, X_{t-1}, \dots, X_1 | X_0 \varepsilon_0}\ X_T, X_{T-1}, \dots, X_1\ X_0, \varepsilon_0\ ; \vartheta = \prod_{t=1}^{T} \frac{1}{2\pi\sigma^2}^{1/2} exp - \frac{\varepsilon_t^2}{2\sigma^2} \quad (2.19)$$

Kemudian *log-likelihood* bersyarat dihitung sebagai berikut:

$$\begin{aligned} \log L\theta &= \log f_{X_t, X_{t-1}, \dots, X_1 |\ X_0, \varepsilon_0}(X_T, \dots, X_1 X_0, \varepsilon_0; \vartheta \\ &= \log \prod_{t=1}^{T} \frac{1}{2\pi\sigma^2}^{\frac{1}{2}} exp - \frac{\varepsilon_t^2}{2\sigma^2} \\ &= \log \prod_{t=1}^{T} (2\pi\sigma^2)^{-\frac{1}{2}} exp - \frac{\varepsilon_t^2}{2\sigma^2} \\ &= -\frac{T}{2}\log\ 2\pi - \frac{T}{2}\log\sigma^2 - \sum_{t=1}^{T} \frac{\varepsilon_t^2}{2\sigma^2} \end{aligned} \quad (2.20)$$

Selanjutnya ditentukan turunan dari $logL\ \vartheta$ terhadap ϑ menggunakan persamaan yaitu sebagai berikut:

$$\frac{\partial[\log L\ \vartheta]}{\partial\vartheta} = 0$$

$$\frac{\partial -\frac{T}{2}\log\ 2\pi - \frac{T}{2}\log\sigma^2 - \sum_{t=1}^{T} \frac{\varepsilon_t^2}{2\sigma^2}}{\partial\vartheta} = 0 \quad (2.21)$$

Kemudian ditentukan turunan kedua dari $logL\,\vartheta$ terhadap ϑ menggunakan persamaan (2.21), untuk membuktikan bahwa ϑ benar-benar memaksimumkan fungsi *likelihood* $L\,\vartheta$ yaitu sebagai berikut:

$$\frac{\partial[\log L\,\vartheta]}{\partial\vartheta^2} < 0$$

$$\frac{\partial^2 - \frac{T}{2}\log\ 2\pi - \frac{T}{2}\log\sigma^2 - {}_{t=1}^{T}\frac{\varepsilon_t^2}{2\sigma^2}}{\partial\vartheta^2} < 0$$

c. Uji Signifikansi Parameter

Dilakukan uji signifikansi parameter, setelah berhasil mengestimasi nilai-nilai parameter dari model *ARIMA* yang ditetapkan sementara untuk mengetahui apakah parameternya signifikan atau tidak.

Berikut merupakan uji signifikansi parameter model pada parameter *Autoregressive*, yaitu:

H_0 : $\phi = 0$ (parameter ϕ tidak signifikan dalam model)

H_1 : $\phi \neq 0$ (parameter ϕ signifikan dalam model)

Taraf signifikansi $\alpha = 0{,}05$

Statistik uji: uji t

$t_{hitung} = \frac{\emptyset}{SE(\emptyset)}$

Kriteria keputusan: tolak H_0 jika $\left|t_{hitung}\right| > t_{\frac{\alpha}{2}}$, dengan derajat bebas db=T-p, dengan T banyaknya data dan p adalah banyaknya parameter dalam model. Sedangkan pada parameter *Moving Average* digunakan hipotesis:

H_0 : $\theta = 0$ (parameter θ tidak signifikan dalam model)

H_1 : $\theta \neq 0$ (parameter θ signifikan dalam model)

Statistik uji yang digunakan adalah

$t_{hitung} = \frac{\theta}{SE(\theta)}$

Kriteria keputusan: H_0 jika $\left|t_{hitung}\right| > t_{\frac{\alpha}{2}}$, dengan derajat bebas db=T-p, dengan T banyaknya data dan p adalah banyaknya parameter dalam model.

d. Pemeriksaan Diagnostik

Setelah berhasil megestimasi nilai-nilai parameter dari model *ARIMA* yang ditetapkan sementara, selanjutnya perlu dilakukan pemeriksaan diagnostik untuk membuktikan bahwa model tersebut cukup memadai dan menentukan model mana yang terbaik digunakan untuk peramalan (Makridakis, 1999: 411). Pemeriksaan diagnostik ini dapat dilakukan dengan mengamati apakah residual dari model terestimasi merupakan proses *white noise* atau tidak (Nachrowi, 2006: 389). Model dikatakan memadai jika asumsi dari *error* (ε_t) memenuhi proses *white noise* dan berdistribusi normal. Apabila dijumpai penyimpangan yang cukup serius maka harus dirumuskan kembali model yang baru, selanjutnya diestimasi dan dilakukan pemeriksaan kembali. Satu cara pemeriksaan yang mudah adalah dengan menggunakan uji yang mampu menetapkan apakah sekumpulan *autokorelasi* secara keseluruhan menunjukkan berbeda dari nol yang disebut dengan uji Statistik *Ljung Box-Pierce.*

Uji kenormalan *error* digunakan untuk memeriksa apakah suatu proses *error* berdistribusi normal atau tidak. Uji kenormalan dapat dilakukan dengan uji Geary's a dengan hipotesis (Nachrowi, 2006):

H_0 : *error* berdistribusi normal

H_1 : *error* tidak berdistribusi normal

Taraf signifikansi atau α yang digunakan adalah 5 %

dengan statistik uji Geary's:

$$z = \frac{a - 0{,}7979}{0{,}2123/T}$$

nilai 0.7979 dan 0,2123 adalah konstanta untuk mencapai kenormalan

dengan $a = \frac{SAD}{T.SEE}$

$X = \frac{1}{T}\ {}^{T}_{t}X_t$

SAD (*Sum Absolute Deviation*)

SAD = ${}^{T}_{t=1}X_t - X$

SSE = (*Sum Square Error*)

SSE = ${}^{T}_{t=1}X_t^2 - TX^2 = {}^{T}_{t=1}X_t^2 - \frac{{}^{T}_{t=1}X_t^2}{T}$

$H_0\ ditolak\ jika\ nilai\ z > -z_{\frac{\alpha}{2}}$

e. Peramalan

Tujuan yang paling penting pada analisis *times series* adalah untuk meramalkan nilai masa depan (Wei, 2006: 88). Menurut Gujarati (2003), cara peramalan dengan menggunakan model *MA* dapat dijelaskan sebagai berikut:

Misalkam H_t merupakan himpunan *time series* yang lalu $(\Delta H_{t-1}, \Delta H_{t-2}, \Delta H_{t-3}, \ldots)$, maka

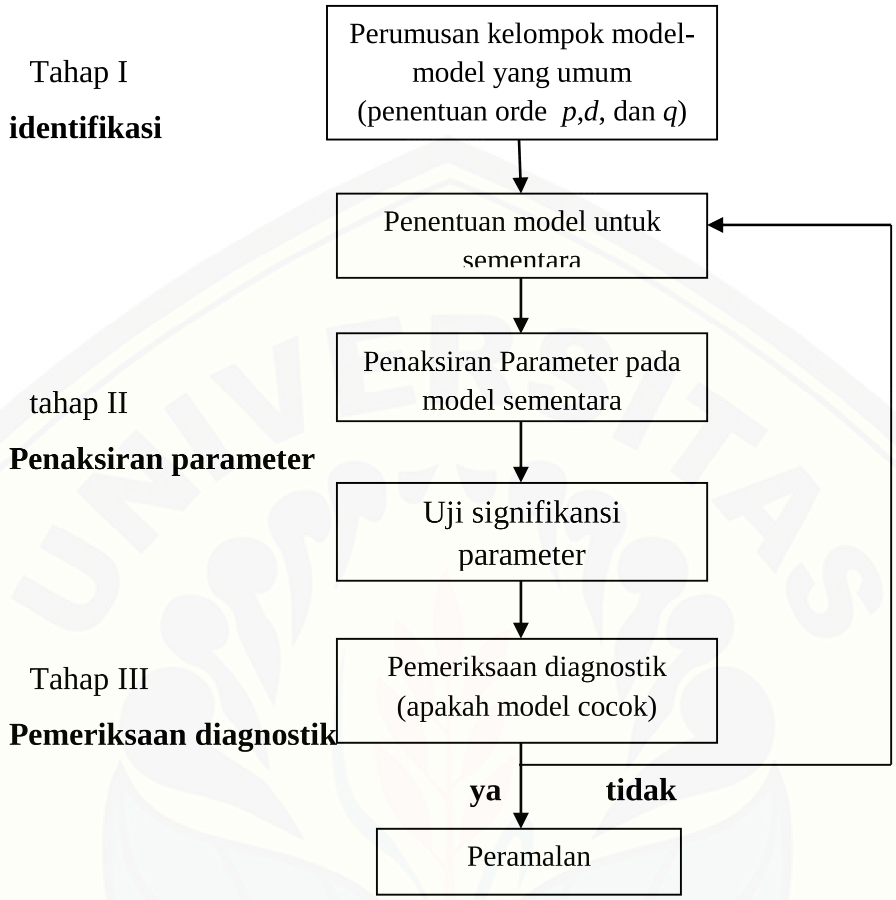
$$\begin{aligned} X''_t &= X'_t - X'_{t-1} \\ &= X_t - X_{t-1} - (X_{t-1} - X_{t-2}) \\ &= X_t - 2X_{t-1} + X_{t-2} \end{aligned}$$

$$\Delta X''_t = \theta_1 \varepsilon_{t-1}$$

kemudian X''_t dapat diperoleh dari $\Delta X''_t = X''_t - X''_{t-1}$

jika semua tahap telah dilakukan dan diperoleh model, maka model ini selanjutnya dapat digunakan untuk melakukan peramalan untuk data periode selanjutnya.

Berdasarkan langkah-langkah pemodelan *ARIMA*, Gambar 2.1 menunjukkan diagram alir langkah pemodelan *ARIMA* (Box, etc, 1994:17).



Gambar 2.6 Diagram Alir Pemodelan *ARIMA*

2.4 *Bootstrap*

Metode *bootstrap* adalah metode berbasis *resampling* data sampel dengan syarat pengembalian pada datanya dalam menyelesaikan statistik ukuran suatu sampel dengan harapan sampel tersebut mewakili data populai sebenarnya, biasanya ukuran resampling diambil secara ribuan kali agar dapat mewakili data populasinya. Metode ini bagus sekali untuk ukuran data sampel yang relatif kecil (dalam bukunya Walpole data kecil yaitu n<30)

Menurut Shao dan Tu (1995) serta Davison dan Hinkley (1997) dalam inferensi statistik parametrik klasik, distribusi sampling dianggap sebagai suatu model dengan sifat-sifat probabilitas yang diketahui, seperti asumsi distribusi yang memerlukan formula analitis berdasarkan pada model untuk mengestimasi secara analitis parameter dalam distribusi samplingnya. Dalam prakteknya, distribusi sampling tidak selalu memenuhi distribusi normal dan kadang-kadang memerlukan penurunan formulasi analitis yang sulit dilakukan sehingga dimungkinkan akurasi estimatornya tidak valid.

Bootstrap memungkinkan seseorang untuk melakukan inferensi statistik tanpa membuat asumsi distribusi yang kuat dan tidak memerlukan formulasi analitis untuk distribusi sampling suatu estimator. Sebagai pengganti, *bootstrap* menggunakan distribusi empiris untuk mengestimasi distribusi sampling. Jadi jika penyelesaian analitik tidak mungkin dilakukan dimana anggapan (suatu distribusi, misalnya kenormalan data) tidak dipenuhi maka dengan menggunakan *bootstrap* masih dapat dilakukan suatu inferensi.

Dasar pendekatan *bootstrap* adalah dengan memperlakukan sampel sebagai populasi dan dengan menggunakan sampling *Monte Carlo* untuk membangkitkan dan mengkonstruksi estimator empiris dari distribusi sampling statistik. Distribusi sampling dapat dipandang sebagai harga-harga statistik yang dihitung dari sejumlah tak terhingga sampel random berukuran n dari suatu populasi yang diberikan. Sampling *Monte Carlo* mengambil konsep ini untuk membangun distribusi sampling suatu estimator dengan mengambil sejumlah besar sampel erukuran secara random dari populasi dan menghitung statistik tersebut dari harga-harga distribusi sampling tersebut. Estimasi *Monte Carlo* yang sebenarnya memerlukan pengetahuan tentang seluruh populasi yang tidak mungkin selalu tersedia dalam prakteknya karena yang dipunyai dari hasil riset praktek adalah sampel dari populasi oleh karena itu dilakukan inferensi untuk *Tetha* dari distribusi samplingnya.

Teknik *bootstrap* merupakan suatu metode resampling untuk memperkirakan distribusi probabilitas suatu statistik. *Bootstrap* diperkenalkan oleh Bradley Efron pada tahun 1979. Istilah *bootstrap* berasal dari *"pull oneself up by one's bootstrap"*, yang berarti berpijak diatas kaki sendiri, berusaha dengan sumber daya minimal. Dalam sudut pandang statistika, sumber daya minimal adalah data yang sedikit, data yang menyimpang dari asumsi tertentu, atau data yang tidak mempunyai asumsi apapun tentang distribusi populasinya. Teknik ini mampu menciptakan ukuran-ukuran dari ketakpastian dan bias, khususnya pada estimasi parameter dari variabel-variabel yang independen dan berdistribusi identik.

Dalam dunia Bootstrap, sebuah sampel diambil dari suatu populasi. Sampel ini dinamakan dengan Sampel Asli. Sampel Asli kemudian diperlukan sebagai populasi dan diaplikasikan prosedur *Monte Carlo* pada sampel tersebut. Hal ini dilakukan dengan mengambil sejumlah besar Sampel Ulang (*Resample*) berukuran n dari Sampel Asli secara random dengan pengembalian. Dengan cara Resampling seperti ini akan diperoleh sampel yang berukuran sama dengan Sampel Asli dan dengan pengembalian dimungkinan diperoleh *Resample* yang sama lebih dari sekali dan mungkin saja berbeda dengan Sampel Asli.

Metode *Bootstrap* tidak selalu memerlukan asumsi distribusi dan formulasi analitis yang rumit untuk mengestimasi parameter dari suatu populasi. Jika asumsi dari suatu distribusi tidak diketahui maka disebut kasus nonparametrik. Langkah-langkah algoritma *bootstap* adalah sebagai berikut:

1. Urutkan data awal dari terkecil ke besar dan labeli data dengan label $x(t)$ $(t=1,..,n)$
2. Hitung titik tengah/mean $z_t=(x(t)+x(t+1))/2$ untuk $t=1,\ldots,t\text{-}1$ dari data yang telah diurutkan

3. Hitung mean sepotong *(ms)* dari *x(t)-x(t-1)* untuk semua data yang telah diurutkan. Hitung batas bawah Z_0 dan batas atas zt dengan $Z_0=x1-ms$ dan $zt=xt+ms$. Batas ini menjadi batas titik tengah.
4. Hitung mean dari kepadatan ME untuk setiap interval yang ada sehingga mean kendala terpenuhi (dinotasikan dengan mt). Mean awal dan mean akhir memiliki formula yang sederhana.
5. Bangkitkan nilai acak dari [0,1] interval uniform, hitung kuantil sampel dari kepadatan ME pada tiap titik dan urutkan.
6. Labeli ulang kuantil sampel dengan menggunakan pelabelan seperti langkah pertama, untuk menjaga hubungan ketergantungan data awal yang diamati.
7. Ulangi langkah ke-2 sampai ke-6 beberapa kali (misalkan 999 kali)(Vinod, 2004, 2006).

2.5 *Bootstrap* untuk data *time series*

Bootstrap untuk data *time series* (dependen) merupakan area riset yang sangat berkembang. Ada banyak ide dan proposal dalam membangun *Bootstrap* untuk data dependen. Hal ini karena r*esampling* pada data *dependen* harus dibangun sedemikian rupa sehingga struktur ketergantungan antara data tidak hilang. Beberapa proposal tentang *boostrap* untuk data dependen antara lain, *block bootstrap* (Kuensch, 1989), *bootstrap* untuk ARMA model, (Franke, 1992), *bootstrap* untuk model-model *nonparametric smoothing* (Franke, 2002a and 2002b).

Dari beberapa kemungkinan di atas, paling mudah dilakukan pada kasus model-model klasik ARMA (*Auto Regressive Moving Average*) yang berdimensi hingga dengan residual *i.i.d.* (Franke, 1992). Sebuah contoh untuk model linear *autoregressive*

$$X_t - \mu_X = \sum_{j=1}^{p} \rho_j \left(X_{t-j} - \mu_x\right) + \varepsilon_t, t \in Z$$

Dimana $\mu_x = E(X_t)$ adalah mean pengamatan dan $\{\varepsilon_t\}$ adalah deret inovasi yang bersifat *i.i.d* dengan sifat $E(\varepsilon_t) = 0$ dan ε_t indipenden terhadap $\{X_s,\ s < t\}$. *Parameter-parameter* $\rho_1, \dots, \rho_p$ *dapat di estimasi dengan menggunakan least square* ataupun dengan menggunakan persamaan-persamaan Yule Walker. Nilai residual dapat dicari melalui persamaan berikut

$$\tilde{\varepsilon}_t - X_t = \hat{\mu}_x - \sum_{j=1}^{p} \hat{\rho}_j \, (X_{t-j} - \hat{\mu}_x)$$

dimana $\hat{\mu}_x = n^{-1} \sum_{t=1}^{n} X_t$ *dan* $\hat{\rho}_1, \dots, \hat{\rho}_p$ adalah nilai estimasi dari parameter-parameter tersebut. *Bootstrap resample* dilakukan dengan membangkitkan

$$X_t^* - \hat{\mu}_x = \sum_{j=1}^{p} \hat{\rho}_j \left(X_{t-j} - \hat{\mu}_x\right) + \varepsilon_t^*$$

dimana ε_t^* dibangkitkan dengan pengembalian *(replacement)* dan residual terpusat *(centered residuals)* $\hat{\varepsilon}_t = \tilde{\varepsilon}_t - n^{-1} \sum_{i=1}^{n} \tilde{\varepsilon}_i$.

Salah satu aplikasi dari *bootstrap* dari data dependen ini adalah untuk mencari selang kepercayaan (*confidence interval*) dari parameter-parameter model peramalan yang bersesuaian.

Adapun langkah-langkah yang dilakukan untuk melakukan metode *bootstrap* adalah sebagai berikut:

Algoritma

Langkah 1 : Memberikan nilai indeks 1 sampai n pada *error* hasil peramalan. Melakukan *resampling* dengan pengembalian pada index *error*. Kemudian index *error* diganti dengan nilai *error* yang sebenarnya.

Langkah 2 : Menggunakan hasil perhitungan *error* pada Langkah 1 untuk membangun sejumlah 1000 sampel *Bootstrap error*$\varepsilon^{*1}, \varepsilon^{*2}, \dots, \varepsilon^{*1000}$. masing-masin sampel berisi n buah *random sampling error*.

Langkah 3 : Membangun 1000 *time series* baru dengan formulasi

Untuk *AR* : $Y_t^{*B} = \rho_1 Y_{t-1} + \rho_2 Y_{t-2} + \cdots + \rho_P Y_{t-P} + \varepsilon_t^{*B}$, t=1,2,...,n.

Untuk *MA* : $Y_t^{*B} = \varepsilon_t^{*B} - \theta_1 \varepsilon_{t-1}^{*B} - \theta_2 \varepsilon_{t-2}^{*B} - \cdots - \theta_q \varepsilon_{t-q}^{*B}$, t=1,2,...,n.

Untuk *ARMA*: $Y_t^{*B} = \varepsilon_t^{*B} + \rho_1 Y_{t-1} - \theta_1 \varepsilon_{t-1}^{*B}$

Langkah 4 : parameter *time series* baru yang dibangun pada Langkah 3. Parameter yang dihasilkan adalah parameter yang baru dan berjumlah 1000 buah.

Langkah 5 : Melakukan pengurutan nilai-nilai parameter dari yang terkecil hingga yang terbesar.

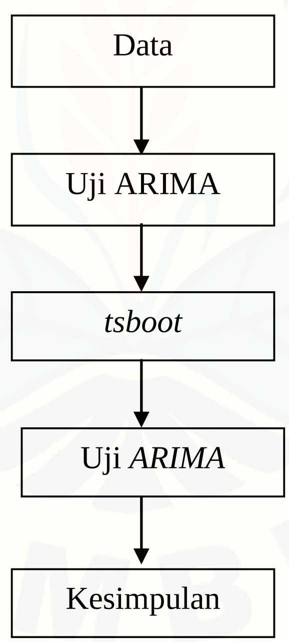
Langkah 6 : Memperoleh 95% *confidence interval* dengan cara membuang sejumlah 2,5% pada urutan parameter bagian atas dan sejumlah 2,5% pada urutan parameter bagian bawah. Parameter yang baru memiliki tingkat kepercayaan 95%.

BAB 3. METODE PENELITIAN

3.1 Data

Pada skripsi ini penulis menggunakan data simulasi *ARIMA* yang di bangkitkan dari program R sebanyak 20 data yang nantinya akan di proses menggunakan paket *tsboot* pada program R. Data ini dibangun dengan asumsi-asumsi yang diinginkan. Misalnya menyesuaikan derajat *AR*(*p*) dan *MA*(*q*) sesuai kebutuhan.

3.2 Langkah – langkah Penelitian



Gambar 3.1 Skema Penelitian

Dari gambar 3.1 dapat dijelaskan tentang langkah – langkah penelitian sebagai berikut :

a. Data

   Data yang digunakan dalam penelitian ini seperti yang dijelaskan pada bagian 3.1, data tersebut merupakan data simulasi *ARIMA*

b. Uji *ARIMA*

   Dalam tahap ini Uji *ARIMA* dilakukan untuk memastikan bahwa data simulasi yang diambil benar-benar merupakan data *ARIMA*. Dalam hal ini uji yang digunakan adalah uji stasioneritas berdasarkan correlogram yaitu dengan memperhatikan ACF dan PACF. Dan dilihat pula nilai AIC dan nilai log like *log likelihood* sebelum dilakukan resampling *bootstrap*.

c. *Resampling Bootstrap*

   Data simulasi dilakukan resampling dengan paket *tsboot* dengan pengembalian untuk memperbesar ukuran sampel. Jumlah ulangan pada *resampling bootstrap* berkisar dari 25-200 (Efron dalam Wahyuningsih 2012). Sehingga peneliti memutuskan untuk melakukan *resampling* sebanyak 200.

d. Uji *ARIMA*

   Data *ARIMA* yang sudah di resampling akan diuji kembali dilakukan untuk memastikan bahwa data yang telah diresampling masih tetap merupakan data *ARIMA*. Dalam hal ini uji yang digunakan adalah uji stasioneritas berdasarkan correlogram yaitu dengan memperhatikan ACF dan PACF. Dan dilihat pula nilai AIC dan nilai log like *log likelihood* setelah dilakukan resampling *bootstrap*.

e. Kesimpulan

   Kesimpulam didapat dari hasil uji *ARIMA* dan akan diketahui model yang terbaik setelah dilakukan resampling *bootstrap* dengan nilai *sd(standart deviasi)* yang berbeda.

3.3 Struktur Fungsi pada Program R

Dalam penelitian ini yang digunakan untuk menganalisis data penelitian menggunakan program R 3.0.3 dan fungsi yang digunakan adalah:

```
tsboot(tseries, statistic, R, l = NULL, sim = "model",
       endcorr = TRUE, n.sim = NROW(tseries), orig.t =
TRUE,
       ran.gen, ran.args = NULL, norm = TRUE, ...,
       parallel = c("no", "multicore", "snow"),
       ncpus = getOption("boot.ncpus", 1L), cl = NULL)
```

`tseries` : Sebuah time series univariat atau multivariat.

`Statistic` : Sebuah fungsi yang bila diterapkan `tseries` mengembalikan vektor yang berisi statistik (s) yang menarik. Setiap kali `statistic` melewati serangkaian waktu panjang `n.sim` yang dari kelas yang sama dengan `tseries` aslinya. Argumen lain yang membutuhkan waktu `statistik` harus tetap konstan untuk setiap bootstrap meniru dan harus dipasok melalui ... argumen untuk `tsboot`.

`R` : Sebuah bilangan bulat positif memberikan jumlah Bootstrap diperlukan.

`Sim` : Jenis simulasi yang diperlukan untuk menghasilkan time series replikasi. Nilai masukan yang mungkin adalah "model" (model berbasis resampling), "tetap" (block resampling dengan panjang blok tetap l), "GEOM" (block resampling dengan blok panjang memiliki distribusi geometris dengan mean l) atau "berebut" (fase berebut).

`l` : Jika sim "tetap" maka l adalah panjang blok tetap digunakan dalam menghasilkan time series replikasi. Jika sim "GEOM" maka l adalah mean dari distribusi geometrik digunakan untuk menghasilkan panjang blok. l harus bilangan bulat positif kurang dari panjang `tseries`. Argumen ini tidak diperlukan bila sim "model" tetapi diperlukan untuk semua jenis simulasi lainnya.

`Endcorr` : Variabel logis menunjukkan apakah koreksi akhir harus diterapkan ketika sim "tetap". Ketika sim "GEOM", endcorr secara otomatis diatur ke TRUE; endcorr tidak digunakan ketika sim "model" atau "berebut".

`n.sim` : Panjang time series simulasi. Biasanya ini akan menjadi sama dengan panjang dari time series asli tetapi ada situasi ketika akan lebih besar. Salah satu situasi yang jelas adalah jika prediksi diperlukan. Situasi lain di mana n.sim lebih besar dari panjang aslinya adalah jika tseries adalah time series sisa dari pas beberapa model untuk time series asli. Dalam hal ini, n.sim biasanya akan menjadi panjang waktu serial aslinya.

`orig.t` : Variabel logis yang menunjukkan apakah statistik harus diterapkan untuk tseries sendiri serta seri bootstrap replikasi. Jika statistik mengharapkan seri waktu lebih lama dari tseries atau jika menerapkan statistik untuk tseries tidak akan menghasilkan informasi yang berguna maka orig.t harus ditetapkan ke palsu.

`ran.gen` : Ini adalah fungsi dari tiga argumen. Argumen pertama adalah time series. Jika sim "model" maka akan selalu tseries yang dilewatkan. Untuk jenis simulasi lain itu adalah hasil dari pengamatan memilih n.sim dari `tseries` oleh beberapa skema dan mengkonversi hasilnya kembali menjadi serangkaian waktu dari bentuk yang sama seperti `tseries` (meskipun panjang `n.sim`). Argumen kedua ran.gen selalu nilai n.sim, dan argumen ketiga adalah ran.args, yang digunakan untuk memasok setiap benda lain yang dibutuhkan oleh ran.gen. Jika sim "model" maka generasi time series replikasi akan dilakukan dalam ran.gen (misalnya melalui penggunaan `arima.sim`). Untuk jenis simulasi lain ran.gen digunakan untuk 'post-menghitam'. Standarnya adalah bahwa fungsi hanya mengembalikan seri waktu berlalu untuk itu.

`ran.args` : Ini akan dipasok untuk ran.gen setiap kali disebut. Jika ran.gen membutuhkan argumen tambahan maka mereka harus disediakan sebagai komponen `ran.args`. Beberapa argumen dapat disahkan dengan membuat ran.args daftar. Jika `ran.args` adalah `NULL` maka tidak boleh digunakan dalam ran.gen tetapi perhatikan ran.gen yang masih harus memiliki argumen ketiga.

`Norma` : Argumen logis menunjukkan apakah margin normal harus digunakan untuk fase berebut. Jika norma yang SALAH maka margin sesuai dengan margin empiris yang tepat digunakan.

... : berisi perintah-perintah lain.

BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan hasil analisis *time series* model *ARIMA* dengan paket *bootstrap tsboot* pada program *R* dan pembahasan dapat diambil kesimpulan bahwa dari tiga contoh simulasi *time series* model *ARIMA* dengan nilai *standar deviasi* yang berbeda yaitu *sqrt(0,1796), sqrt(3,1796)* dan *sqrt(0,0096)* didapatkan model terbaik pada simulasi dengan *sd=sqrt(0,0096)* yaitu *AIC*=-35,58 dan *log likelihood*=23,79 sebelum dilakukan *bootstrap* dan *AIC*=-1402,56 dan *log likelihood*=707,28 setelah dilakukan *bootstrap*.

5.2 Saran

Pada penelitian ini, hanya menggunakan data simulasi *time series* model ARIMA. Untuk penelitian selanjutnya analisis *time series* dapat dilakukan untuk memprediksi data sebenarnya dalam kehidupan sehari-hari misalnya: hasil penjualan, waktu turun hujan, dll dengan *resampling bootstrap* apabila data yang dimiliki kurang dari 30 *series.*

DAFTAR PUSTAKA

Akhyasrinuki. 2011. *Pengertian Time Series*. [serial on line]. http://id.shvoong.com/writing-and-speaking/2122738-pengertian-time-series/#ixzz2WcNALjH3. [19 Juni 2013].

Box, G.E.P. Jenkins, G.M., Reinsel, G., C. 1994. *"Time Series Analysis"*. New Jersey: Prentice Hall.

Box, G.E.P. Jenkins, G.M., Reinsel, G., C. 1976. *"Time Series Analysis, Forecasting and Control "* Revised ed. Holden Day Inc California.

Davison AC, Hinkley DV. 1997. *Bootstrap Methods and Their Applications*. Cambridge: Cambridge University Press.

Efron, Bradley. 1994. *The Jackknife, The Bootstrap and The Other Resampling Plans.* Department of Statistics Stanford University.

Enders, W. 1985. *Applied Econometrics Time Series*. New York: John Willey & Sons, Inc.

Firmansyah .2000. *Peramalan Inflasi Dengan Metode Box-Jenkins(ARIMA) : Studi Kasus Tingkat Inflasi Kota Semarang dan Yogyakarta 1994-2000*. Media Ekonomi & Bisnis Vol. XII No.2 Desember 2000.

Franke, J. and Kreiss, J.P. 1992. *Bootstraping ARMA models, Journal of Time Series Analysis*,13:297-317

Franke, J,. Kreiss, J.P. and Mammen, E. 2002a. *Bootstrap of kernel smoothing in nonlinear time series*. *Bernoulli*, 8:1-37.

Franke,J,. Kreiss, J.P., Mammen, E., and Neumann, M.H. 2002b. *Properties of the Nonparametric Autoregressive Bootstrap, Journal of Time Series Analysis,* 23:555-585.

Kuensch, H.R., 1989. *The Jackknife and the bootstrap for general stationary observations, Annals of Statistics*, 17:1217-1241.

Gujarati, D.N. 2003. *Basic Econometric*. 4^{Th} Edition; McGraw Hill, Inc.

Halim, Siana. 2006. *Diktat – Time Series Analysis.*

Hamilton dan James, D. 1994. *Time Series Analysis*. Princeton : Princeton University Press.

Kahiruddin, Hanif. 2012. *Analisis Data Time Series*. [serial on line]. http://digilib.ittelkom.ac.id/index.php?option=com_content&view=article&id=978:analisis-data-time-series&catid=13:rpl&Itemid=14. [19 Juni 2013].

Makridakis, S., Wheelwright, S.C., dan McGee, V.E. 1995. *"Metode dan Aplikasi Peramalan" jilid I.* Jakarta: Bina Rupa Aksara.

Makridakis, W. M. G. 1999. *"Metode dan Aplikasi Peramalan"* Edisi kedua. Jakarta: Bina Rupa Aksara.

Nachrowi, D, dan Usman, Hardius. 2006. *"Pendekatan Praktis dan Populer: Ekonometrika Untuk Analisis dan Ekonomi",* Lembaga Penerbit FEUI.

Palit, A.K. & Popovic, D.2005. *Computational Intelegent in Time Series Forecasting: Theory and Engineering Application*. London: Springer.

Pankratz, Alan. 1983. *Forecasting with Univariate Box-Jenkins Models*. New York: John Wiley & Sons, Inc.

Pankratz, Alan. 1991. *Forecasting With Dynamic Regression Models*. New York: John Wiley & Sons, Inc.

Salamah, M., Suhartono., dan Wulandari S. 2003. *Analisis Time Series*. Surabaya: Jurusan Statistik ITS.

Shao, Jun and Tu, Dongsheng. 1995. *The Jackknife and Bootstrap*. NewYork: Springer-Verlag.

Sugiarto dan Harijono. 2000. "*Peramalan Bisnis"*. Jakarta: Gramedia Pustaka Utama.

Vinod, H.D. 2004. *Ranking mutual funds using unconventional utility theory and stochastic dominance,Journal of Empirical Finance, 11(3), pp. 353-377.*

Vinod, H.D. 2006. *Maximum Entropy Ensembles for Time Series Inference in Economics, Journal of Asian Economics, 17(6), pp. 955-978.*

Wei, W.W.S. 1990. *"Time Series Analysis : Univariate and Multivariate Methods"*. United State of America : Addison-Wesley Publishing Company. 52.

LAMPIRAN A. Skrip Program R untuk keseluruhan program

```
library(boot)
library(fpp)
library(forecast)

a.1=arima.sim(n=20, list(order = c(2,0,2),
               ar = c(0.8897, -0.4858),
               ma = c(-0.2279, 0.2488)), sd=sqrt(0.1796))
tsdisplay(a.1)

a.2=tsboot(a.1,mean,R=50,l=19,sim="fixed")
tsboot.a.1=boot.array(a.2)
tsboot.a.1=a.1[tsboot.a.1]
tsdisplay(tsboot.a.1)

a.3=arima.sim(n=20, list(order = c(2,0,2),
               ar = c(0.8897, -0.4858),
               ma = c(-0.2279, 0.2488)), sd=sqrt(3.1796))
tsdisplay(a.3)

a.4=tsboot(a.3,mean,R=50,l=19,sim="fixed")
tsboot.a.3=boot.array(a.4)
tsboot.a.3=a.3[tsboot.a.3]
tsdisplay(tsboot.a.3)

a.5=arima.sim(n=20, list(order = c(2,0,2),
               ar = c(0.8897, -0.4858),
               ma = c(-0.2279, 0.2488)), sd=sqrt(0.0096))
tsdisplay(a.5)
```

```
a.6=tsboot(a.5,mean,R=50,l=19,sim="fixed")
tsboot.a.5=boot.array(a.6)
tsboot.a.5=a.5[tsboot.a.5]
tsdisplay(tsboot.a.5)

fitsim1=arima(a.1,order=c(2,0,2))
tsdiag(fitsim1)

fitsim11=arima(a.1,order=c(2,0,2))
summary(fitsim11)


fitboot1=arima(tsboot.a.1,order=c(2,0,2))
tsdiag(fitboot1)


fitboot11=Arima(tsboot.a.1,order=c(2,0,2))
summary(fitboot11)

fitsim2=arima(a.3,order=c(2,0,2))
tsdiag(fitsim2)

fitsim22=arima(a.3,order=c(2,0,2))
summary(fitsim22)

fitboot2=arima(tsboot.a.3,order=c(2,0,2))
tsdiag(fitboot2)
```

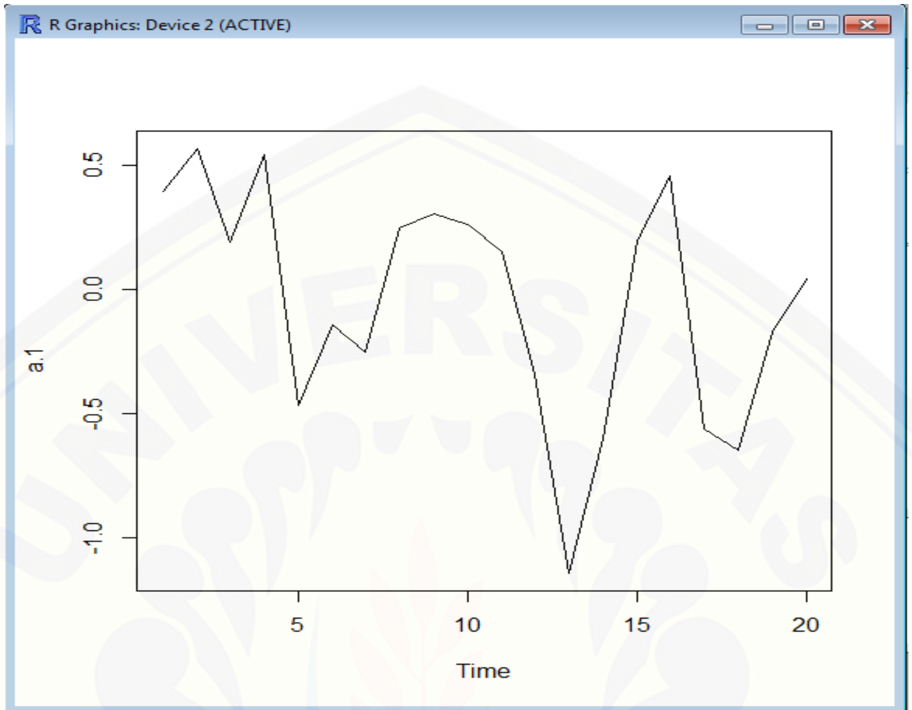
```
fitboot22=Arima(tsboot.a.3,order=c(2,0,2))
summary(fitboot22)

fitsim3=arima(a.5,order=c(2,0,2))
tsdiag(fitsim3)

fitsim33=arima(a.5,order=c(2,0,2))
summary(fitsim33)

fitboot3=arima(tsboot.a.5,order=c(2,0,2))
tsdiag(fitboot3)

fitboot33=Arima(tsboot.a.5,order=c(2,0,2))
summary(fitboot33)
```

LAMPIRAN B. Skrip Program R untuk menampilkan data simulasi *time series* model ARIMA

- **a.1=arima.sim(n=20, list(order = c(2,0,2),**
  ar = c(0.8897, -0.4858),
  ma = c(-0.2279, 0.2488)), sd=sqrt(0.1796))

```
Time Series:
Start = 1
End = 20
Frequency = 1
 [1]  0.39598210  0.56786140  0.18808129  0.54215746
-0.46508280 -0.14329381
 [7] -0.25265059  0.24836937  0.30292019  0.26046859
0.15326529 -0.34379302
[13] -1.14246417 -0.59931421  0.19536218  0.45561999
-0.56223180 -0.64462599
[19] -0.16990373  0.04209457
```

➢ **ts.plot(a.1)**



LAMPIRAN C. Skrip Program R untuk *resampling bootstrap* pada data simulasi *time series* yang telah didapatkan

- **a.2=tsboot(a.1,mean,R=50,l=19,sim="fixed")**

```
BLOCK BOOTSTRAP FOR TIME SERIES

Fixed Block Length of 19

Call:
tsboot(tseries = a.1, statistic = mean, R = 50, l =
19, sim = "fixed")


Bootstrap Statistics :
       original       bias    std. error
t1* -0.04855888 0.002488944  0.03250323
```

- **boot.array(a.2)**(hasil resampling dalam bentuk kode nomor urut)

```
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
[,10] [,11] [,12]
 [1,]   10   11   12   13   14   15   16   17   18
19   20    1
 [2,]   13   14   15   16   17   18   19   20    1
2    3    4
 [3,]    3    4    5    6    7    8    9   10   11
12   13   14
 [4,]    4    5    6    7    8    9   10   11   12
13   14   15
```

```
 [5,]     5     6     7     8     9    10    11    12    13
14    15    16
 [6,]    20     1     2     3     4     5     6     7     8
9    10    11
 [7,]    17    18    19    20     1     2     3     4     5
6     7     8
 [8,]    15    16    17    18    19    20     1     2     3
4     5     6
 [9,]     9    10    11    12    13    14    15    16    17
18    19    20
[10,]    16    17    18    19    20     1     2     3     4
5     6     7
[11,]    17    18    19    20     1     2     3     4     5
6     7     8
[12,]    17    18    19    20     1     2     3     4     5
6     7     8
[13,]    15    16    17    18    19    20     1     2     3
4     5     6
[14,]     7     8     9    10    11    12    13    14    15
16    17    18
[15,]     2     3     4     5     6     7     8     9    10
11    12    13
[16,]     8     9    10    11    12    13    14    15    16
17    18    19
[17,]     6     7     8     9    10    11    12    13    14
15    16    17
[18,]    20     1     2     3     4     5     6     7     8
9    10    11
```

```
[19,]    7    8    9   10   11   12   13   14   15
16   17   18
[20,]   13   14   15   16   17   18   19   20    1
2    3    4
[21,]   12   13   14   15   16   17   18   19   20
1    2    3
[22,]   16   17   18   19   20    1    2    3    4
5    6    7
[23,]   20    1    2    3    4    5    6    7    8
9   10   11
[24,]   14   15   16   17   18   19   20    1    2
3    4    5
[25,]    5    6    7    8    9   10   11   12   13
14   15   16
[26,]    9   10   11   12   13   14   15   16   17
18   19   20
[27,]   17   18   19   20    1    2    3    4    5
6    7    8
[28,]   11   12   13   14   15   16   17   18   19
20    1    2
[29,]    3    4    5    6    7    8    9   10   11
12   13   14
[30,]   15   16   17   18   19   20    1    2    3
4    5    6
[31,]    4    5    6    7    8    9   10   11   12
13   14   15
[32,]   19   20    1    2    3    4    5    6    7
8    9   10
```

```
[33,]    18    19    20     1     2     3     4     5     6
7     8     9
[34,]    19    20     1     2     3     4     5     6     7
8     9    10
[35,]     6     7     8     9    10    11    12    13    14
15    16    17
[36,]    11    12    13    14    15    16    17    18    19
20     1     2
[37,]     1     2     3     4     5     6     7     8     9
10    11    12
[38,]     3     4     5     6     7     8     9    10    11
12    13    14
[39,]    14    15    16    17    18    19    20     1     2
3     4     5
[40,]    10    11    12    13    14    15    16    17    18
19    20     1
[41,]    18    19    20     1     2     3     4     5     6
7     8     9
[42,]     1     2     3     4     5     6     7     8     9
10    11    12
[43,]     7     8     9    10    11    12    13    14    15
16    17    18
[44,]     6     7     8     9    10    11    12    13    14
15    16    17
[45,]    17    18    19    20     1     2     3     4     5
6     7     8
[46,]    17    18    19    20     1     2     3     4     5
6     7     8
```

```
[47,]    12    13    14    15    16    17    18    19    20
1      2      3
[48,]     8     9    10    11    12    13    14    15    16
17     18     19
[49,]    18    19    20     1     2     3     4     5     6
7      8      9
[50,]     6     7     8     9    10    11    12    13    14
15     16     17
      [,13]  [,14]  [,15]  [,16]  [,17]  [,18]  [,19]
[,20]
 [1,]      2      3      4      5      6      7      8
1
 [2,]      5      6      7      8      9     10     11
12
 [3,]     15     16     17     18     19     20      1
5
 [4,]     16     17     18     19     20      1      2
4
 [5,]     17     18     19     20      1      2      3
5
 [6,]     12     13     14     15     16     17     18
7
 [7,]      9     10     11     12     13     14     15
12
 [8,]      7      8      9     10     11     12     13
15
 [9,]      1      2      3      4      5      6      7
15
```

```
[10,]     8     9    10    11    12    13    14
11
[11,]     9    10    11    12    13    14    15
12
[12,]     9    10    11    12    13    14    15
6
[13,]     7     8     9    10    11    12    13
17
[14,]    19    20     1     2     3     4     5
2
[15,]    14    15    16    17    18    19    20
12
[16,]    20     1     2     3     4     5     6
13
[17,]    18    19    20     1     2     3     4
8
[18,]    12    13    14    15    16    17    18
11
[19,]    19    20     1     2     3     4     5
19
[20,]     5     6     7     8     9    10    11
1
[21,]     4     5     6     7     8     9    10
8
[22,]     8     9    10    11    12    13    14
20
[23,]    12    13    14    15    16    17    18
10
```

```
[24,]     6     7     8     9    10    11    12
6
[25,]    17    18    19    20     1     2     3
4
[26,]     1     2     3     4     5     6     7
11
[27,]     9    10    11    12    13    14    15
19
[28,]     3     4     5     6     7     8     9
12
[29,]    15    16    17    18    19    20     1
19
[30,]     7     8     9    10    11    12    13
2
[31,]    16    17    18    19    20     1     2
13
[32,]    11    12    13    14    15    16    17
12
[33,]    10    11    12    13    14    15    16
20
[34,]    11    12    13    14    15    16    17
2
[35,]    18    19    20     1     2     3     4
7
[36,]     3     4     5     6     7     8     9
8
[37,]    13    14    15    16    17    18    19
1
```

```
[38,]    15    16    17    18    19    20     1
2
[39,]     6     7     8     9    10    11    12
5
[40,]     2     3     4     5     6     7     8
3
[41,]    10    11    12    13    14    15    16
5
[42,]    13    14    15    16    17    18    19
7
[43,]    19    20     1     2     3     4     5
16
[44,]    18    19    20     1     2     3     4
5
[45,]     9    10    11    12    13    14    15
2
[46,]     9    10    11    12    13    14    15
1
[47,]     4     5     6     7     8     9    10
13
[48,]    20     1     2     3     4     5     6
12
[49,]    10    11    12    13    14    15    16
15
[50,]    18    19    20     1     2     3     4
13
```

➢ **`tsboot.a.1`** (hasil resampling bootstrap dalam bentuk data sebenarnya)

```
 [1] -0.14329381  0.30292019  0.18808129  0.56786140 -
0.14329381 -0.16990373
 [7] -0.16990373 -0.16990373  0.30292019  0.04209457
0.30292019 -0.46508280
[13]  0.56786140  0.15326529 -0.14329381 -0.64462599
0.18808129  0.54215746
[19]  0.19536218 -0.14329381  0.15326529  0.56786140
0.24836937  0.19536218
[25]  0.56786140  0.39598210  0.15326529  0.26046859
0.30292019  0.54215746
[31] -0.59931421  0.45561999  0.56786140 -0.16990373
0.04209457  0.56786140
[37]  0.56786140 -0.16990373 -0.64462599 -0.64462599
0.30292019  0.39598210
[43] -0.59931421  0.45561999 -0.25265059 -0.64462599
0.54215746 -0.46508280
[49]  0.45561999  0.39598210 -0.25265059  0.26046859
0.54215746  0.18808129
[55] -0.25265059  0.04209457  0.04209457  0.04209457
0.26046859  0.39598210
[61]  0.26046859 -0.14329381  0.18808129 -0.34379302 -
0.25265059 -0.16990373
[67]  0.54215746 -0.46508280  0.45561999 -0.25265059 -
0.34379302  0.18808129
[73]  0.30292019  0.45561999  0.18808129  0.56786140 -
0.34379302  0.15326529
[79]  0.26046859 -0.46508280  0.19536218 -0.56223180
0.18808129  0.04209457
```

[85] 0.39598210 0.18808129 0.18808129 0.04209457 -0.16990373 -0.16990373
[91] 0.26046859 0.56786140 0.19536218 -0.56223180 0.24836937 -0.16990373
[97] -0.46508280 -0.14329381 -0.56223180 0.56786140 0.24836937 0.15326529
[103] -0.46508280 0.54215746 0.24836937 0.39598210 0.39598210 0.39598210
[109] 0.15326529 0.56786140 0.15326529 -0.25265059 0.54215746 -1.14246417
[115] 0.24836937 0.04209457 -0.46508280 -0.14329381 -0.56223180 0.24836937
[121] -1.14246417 0.54215746 0.26046859 -0.56223180 0.54215746 0.18808129
[127] -1.14246417 -0.34379302 0.15326529 -0.14329381 0.45561999 -0.64462599
[133] 0.54215746 0.39598210 0.56786140 0.54215746 0.54215746 0.39598210
[139] 0.04209457 0.04209457 0.15326529 0.18808129 0.45561999 -0.64462599
[145] 0.30292019 0.04209457 -0.14329381 -0.25265059 -0.64462599 0.18808129
[151] 0.30292019 -0.34379302 -0.14329381 -0.46508280 0.30292019 0.56786140
[157] 0.56786140 0.56786140 -0.34379302 0.18808129 -0.34379302 0.24836937
[163] -0.46508280 -0.59931421 0.30292019 0.39598210 -0.14329381 -0.25265059

```
[169] -0.64462599  0.30292019 -0.59931421 -0.46508280
 0.15326529 -0.64462599
[175] -0.46508280  0.54215746 -0.59931421 -1.14246417 -
 0.34379302 -0.25265059
[181] -0.56223180 -0.16990373 -0.46508280  0.56786140
 0.18808129 -0.46508280
[187] -0.46508280  0.56786140  0.39598210  0.39598210 -
 0.34379302  0.54215746
[193] -0.56223180 -0.16990373  0.26046859  0.39598210 -
 0.25265059  0.24836937
[199] -0.16990373  0.54215746  0.26046859 -1.14246417 -
 0.25265059 -0.14329381
[205]  0.26046859  0.18808129  0.18808129  0.18808129 -
 1.14246417  0.54215746
[211] -1.14246417  0.30292019 -0.14329381  0.19536218
 0.26046859  0.56786140
[217] -0.25265059  0.24836937 -0.16990373  0.26046859
 0.19536218 -0.14329381
[223] -0.34379302 -0.16990373 -0.14329381 -0.46508280
 0.19536218 -0.59931421
[229] -1.14246417  0.24836937 -0.64462599  0.04209457 -
 0.14329381  0.18808129
[235]  0.54215746 -0.14329381 -0.14329381  0.18808129
 0.56786140  0.56786140
[241] -1.14246417 -0.46508280 -0.64462599  0.04209457
 0.15326529  0.56786140
[247]  0.24836937  0.30292019  0.04209457 -0.46508280
 0.15326529 -0.59931421
```

```
[253]   0.24836937 -0.25265059   0.15326529   0.54215746
 0.54215746  0.54215746
[259] -0.59931421 -0.46508280 -0.59931421  0.26046859 -
 0.25265059  0.45561999
[265]   0.15326529   0.18808129   0.24836937   0.30292019
 0.04209457  0.15326529
[271]  0.45561999 -0.25265059 -1.14246417  0.04209457 -
 0.25265059 -0.14329381
[277]  0.45561999  0.19536218 -0.59931421  0.30292019 -
 0.16990373  0.39598210
[283] -0.25265059  0.54215746 -0.46508280 -0.25265059 -
 0.25265059  0.54215746
[289]  0.18808129  0.18808129 -0.59931421 -0.14329381 -
 0.16990373  0.39598210
[295] -0.34379302   0.18808129   0.30292019   0.26046859
 0.39598210 -0.14329381
[301] -0.34379302  0.19536218  0.30292019  0.24836937 -
 0.34379302 -0.46508280
[307]  -0.46508280  -0.46508280   0.19536218  -0.14329381
 0.19536218  0.15326529
[313]   0.24836937  -0.56223180  -0.34379302   0.54215746
 0.30292019  0.26046859
[319]  0.39598210 -0.34379302 -0.56223180  0.24836937 -
 0.59931421  0.39598210
[325]   0.24836937  -0.25265059  -0.56223180   0.45561999
 0.19536218  0.26046859
[331]  0.04209457  0.56786140  0.24836937 -0.46508280 -
 0.14329381  0.24836937
```

```
[337]   0.24836937  -0.46508280   0.54215746   0.54215746
 0.19536218 -0.25265059
[343]   0.04209457   0.56786140  -1.14246417   0.54215746
 0.26046859  0.15326529
[349]   0.56786140  -0.25265059  -1.14246417   0.45561999
 0.26046859  0.30292019
[355]  -1.14246417  -0.14329381  -0.14329381  -0.14329381
 0.45561999 -0.25265059
[361]   0.45561999 -0.34379302   0.30292019 -0.64462599 -
 1.14246417 -0.46508280
[367]   0.26046859   0.15326529   0.56786140 -1.14246417 -
 0.64462599  0.30292019
[373]   0.19536218   0.56786140   0.30292019   0.24836937 -
 0.64462599 -0.56223180
[379]   0.45561999   0.15326529   0.39598210   0.18808129
 0.30292019 -0.14329381
[385]  -0.25265059   0.30292019   0.30292019 -0.14329381 -
 0.46508280 -0.46508280
[391]   0.45561999   0.24836937   0.39598210   0.18808129 -
 0.59931421 -0.46508280
[397]   0.15326529 -0.34379302   0.18808129   0.24836937 -
 0.59931421 -0.56223180
[403]   0.15326529   0.26046859 -0.59931421 -0.25265059 -
 0.25265059 -0.25265059
[409]  -0.56223180   0.24836937  -0.56223180  -1.14246417
 0.26046859 -0.16990373
[415]  -0.59931421  -0.14329381   0.15326529  -0.34379302
 0.18808129 -0.59931421
```

```
[421] -0.16990373   0.26046859   0.45561999   0.18808129
 0.26046859  0.30292019
[427]  -0.16990373  -0.64462599  -0.56223180  -0.34379302
 0.56786140  0.54215746
[433]   0.26046859 -0.25265059   0.24836937   0.26046859
 0.26046859 -0.25265059
[439]  -0.14329381  -0.14329381  -0.56223180   0.30292019
 0.56786140  0.54215746
[445]   0.19536218  -0.14329381  -0.34379302  -1.14246417
 0.54215746  0.30292019
[451]   0.19536218 -0.64462599  -0.34379302   0.15326529
 0.19536218  0.24836937
[457]  0.24836937  0.24836937 -0.64462599  0.30292019 -
 0.64462599 -0.59931421
[463]  0.15326529  0.04209457  0.19536218 -0.25265059 -
 0.34379302 -1.14246417
[469]  0.54215746  0.19536218  0.04209457  0.15326529 -
 0.56223180  0.54215746
[475]  0.15326529  0.26046859  0.04209457 -0.16990373 -
 0.64462599 -1.14246417
[481]   0.18808129 -0.46508280   0.15326529   0.24836937
 0.30292019  0.15326529
[487]  0.15326529  0.24836937 -0.25265059 -0.25265059 -
 0.64462599  0.26046859
[493]  0.18808129 -0.46508280  0.45561999 -0.25265059 -
 1.14246417 -0.59931421
[499] -0.46508280  0.26046859  0.45561999 -0.16990373 -
 1.14246417 -0.34379302
```

[505] 0.45561999 0.30292019 0.30292019 0.30292019 -
0.16990373 0.26046859
[511] -0.16990373 0.19536218 -0.34379302 0.39598210
0.45561999 0.24836937
[517] -1.14246417 -0.59931421 -0.46508280 0.45561999
0.39598210 -0.34379302
[523] -0.64462599 -0.46508280 -0.34379302 0.15326529
0.39598210 0.04209457
[529] -0.16990373 -0.59931421 0.54215746 -0.14329381 -
0.34379302 0.30292019
[535] 0.26046859 -0.34379302 -0.34379302 0.30292019
0.24836937 0.24836937
[541] -0.16990373 0.15326529 0.54215746 -0.14329381 -
0.56223180 0.24836937
[547] -0.59931421 0.19536218 -0.14329381 0.15326529 -
0.56223180 0.04209457
[553] -0.59931421 -1.14246417 -0.56223180 0.26046859
0.26046859 0.26046859
[559] 0.04209457 0.15326529 0.04209457 0.45561999 -
1.14246417 0.56786140
[565] -0.56223180 0.30292019 -0.59931421 0.19536218 -
0.14329381 -0.56223180
[571] 0.56786140 -1.14246417 -0.16990373 -0.14329381 -
1.14246417 -0.34379302
[577] 0.56786140 0.39598210 0.04209457 0.19536218 -
0.46508280 -0.25265059
[583] -1.14246417 0.26046859 0.15326529 -1.14246417 -
1.14246417 0.26046859

```
[589]  0.30292019  0.30292019  0.04209457 -0.34379302 -
 0.46508280 -0.25265059
[595] -0.64462599  0.30292019  0.19536218  0.45561999 -
 0.25265059 -0.34379302
[601] -0.64462599  0.39598210  0.19536218 -0.59931421 -
 0.64462599  0.15326529
[607]   0.15326529   0.15326529   0.39598210  -0.34379302
 0.39598210 -0.56223180
[613]  -0.59931421   0.18808129  -0.64462599   0.26046859
 0.19536218  0.45561999
[619]  -0.25265059  -0.64462599   0.18808129  -0.59931421
 0.04209457 -0.25265059
[625]  -0.59931421  -1.14246417   0.18808129   0.56786140
 0.39598210  0.45561999
[631] -0.14329381  0.24836937 -0.59931421  0.15326529 -
 0.34379302 -0.59931421
[637]  -0.59931421   0.15326529   0.26046859   0.26046859
 0.39598210 -1.14246417
[643]  -0.14329381   0.24836937  -0.16990373   0.26046859
 0.45561999 -0.56223180
[649]   0.24836937  -1.14246417  -0.16990373   0.56786140
 0.45561999  0.19536218
[655]  -0.16990373  -0.34379302  -0.34379302  -0.34379302
 0.56786140 -1.14246417
[661]  0.56786140 -0.64462599  0.19536218  0.54215746 -
 0.16990373  0.15326529
[667]   0.45561999  -0.56223180   0.24836937  -0.16990373
 0.54215746  0.19536218
```

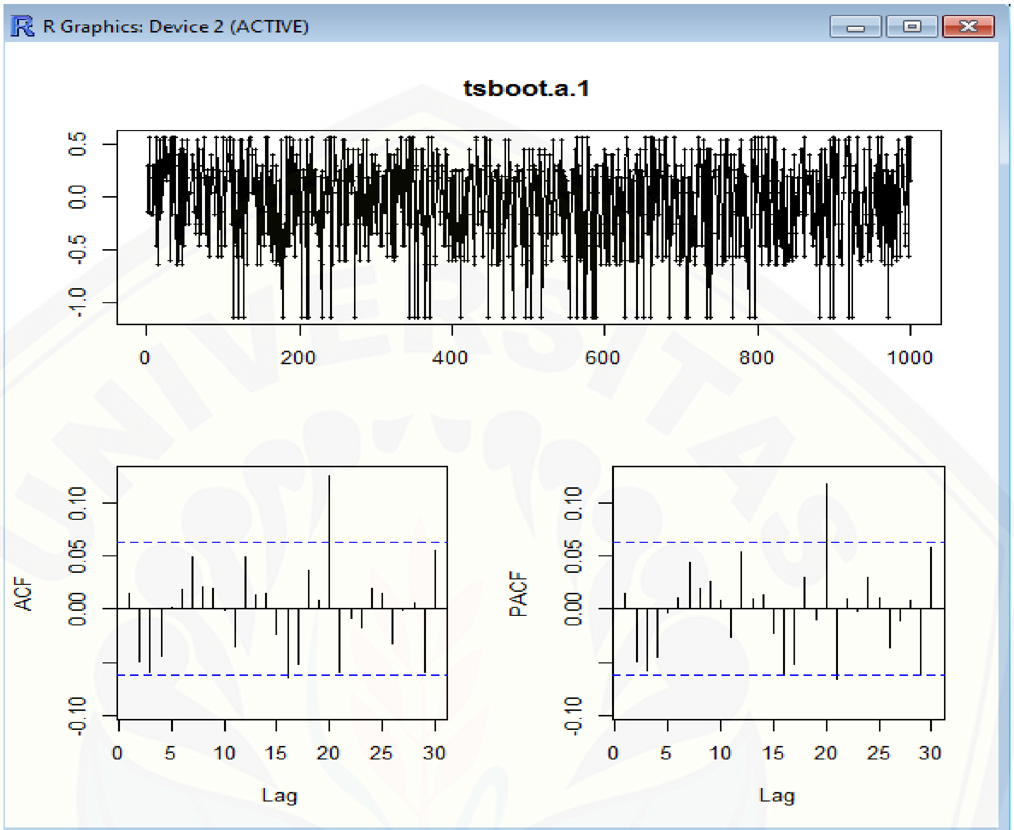
```
[673]   0.39598210   0.24836937   0.19536218  -0.59931421
 0.54215746  0.18808129
[679]   0.56786140  -0.56223180  -0.25265059   0.30292019
 0.19536218 -0.34379302
[685]  -1.14246417   0.19536218   0.19536218  -0.34379302
 0.15326529  0.15326529
[691]   0.56786140  -0.59931421  -0.25265059   0.30292019
 0.04209457  0.15326529
[697]  -0.56223180  -0.64462599   0.30292019  -0.59931421
 0.04209457  0.18808129
[703]  -0.56223180   0.45561999   0.04209457 -1.14246417 -
 1.14246417 -1.14246417
[709]   0.18808129  -0.59931421   0.18808129  -0.16990373
 0.45561999 -0.46508280
[715]   0.04209457  -0.34379302  -0.56223180  -0.64462599
 0.30292019  0.04209457
[721]  -0.46508280   0.45561999   0.56786140   0.30292019
 0.45561999  0.19536218
[727]  -0.46508280   0.54215746   0.18808129  -0.64462599
 0.24836937  0.26046859
[733]   0.45561999  -1.14246417  -0.59931421   0.45561999
 0.45561999 -1.14246417
[739]  -0.34379302  -0.34379302   0.18808129   0.19536218
 0.24836937  0.26046859
[745]   0.39598210  -0.34379302  -0.64462599  -0.16990373
 0.26046859  0.19536218
[751]   0.39598210   0.54215746  -0.64462599  -0.56223180
 0.39598210 -0.59931421
```

[757] -0.59931421 -0.59931421 0.54215746 0.19536218
0.54215746 0.04209457
[763] -0.56223180 -0.14329381 0.39598210 -1.14246417 -
0.64462599 -0.16990373
[769] 0.26046859 0.39598210 -0.14329381 -0.56223180
0.18808129 0.26046859
[775] -0.56223180 0.45561999 -0.14329381 -0.46508280
0.54215746 -0.16990373
[781] 0.30292019 0.15326529 -0.56223180 -0.59931421
0.19536218 -0.56223180
[787] -0.56223180 -0.59931421 -1.14246417 -1.14246417
0.54215746 0.45561999
[793] 0.30292019 0.15326529 0.56786140 -1.14246417 -
0.16990373 0.04209457
[799] 0.15326529 0.45561999 0.56786140 -0.46508280 -
0.16990373 -0.64462599
[805] 0.56786140 0.19536218 0.19536218 0.19536218 -
0.46508280 0.45561999
[811] -0.46508280 0.39598210 -0.64462599 -0.25265059
0.56786140 -0.59931421
[817] -0.16990373 0.04209457 0.15326529 0.56786140 -
0.25265059 -0.64462599
[823] 0.54215746 0.15326529 -0.64462599 -0.56223180 -
0.25265059 -0.14329381
[829] -0.46508280 0.04209457 0.26046859 -0.34379302 -
0.64462599 0.19536218
[835] 0.45561999 -0.64462599 -0.64462599 0.19536218 -
0.59931421 -0.59931421

```
[841] -0.46508280 -0.56223180  0.26046859 -0.34379302
 0.18808129 -0.59931421
[847]  0.04209457  0.39598210 -0.34379302 -0.56223180
 0.18808129 -0.14329381
[853]  0.04209457 -0.16990373  0.18808129  0.45561999
 0.45561999  0.45561999
[859] -0.14329381 -0.56223180 -0.14329381  0.56786140 -
 0.16990373  0.24836937
[865]  0.18808129  0.19536218  0.04209457  0.39598210 -
 0.34379302  0.18808129
[871]  0.24836937 -0.16990373 -0.46508280 -0.34379302 -
 0.16990373 -0.64462599
[877]  0.24836937 -0.25265059 -0.14329381  0.39598210
 0.15326529 -1.14246417
[883] -0.16990373  0.45561999 -0.56223180 -0.16990373 -
 0.16990373  0.45561999
[889]  0.19536218  0.19536218 -0.14329381 -0.64462599
 0.15326529 -1.14246417
[895]  0.54215746  0.19536218  0.39598210  0.56786140 -
 1.14246417 -0.64462599
[901]  0.54215746 -0.25265059  0.39598210  0.04209457
 0.54215746 -0.56223180
[907] -0.56223180 -0.56223180 -0.25265059 -0.64462599 -
 0.25265059  0.18808129
[913]  0.04209457  0.30292019  0.54215746  0.45561999
 0.39598210  0.56786140
[919] -1.14246417  0.54215746  0.30292019  0.04209457 -
 0.14329381 -1.14246417
```

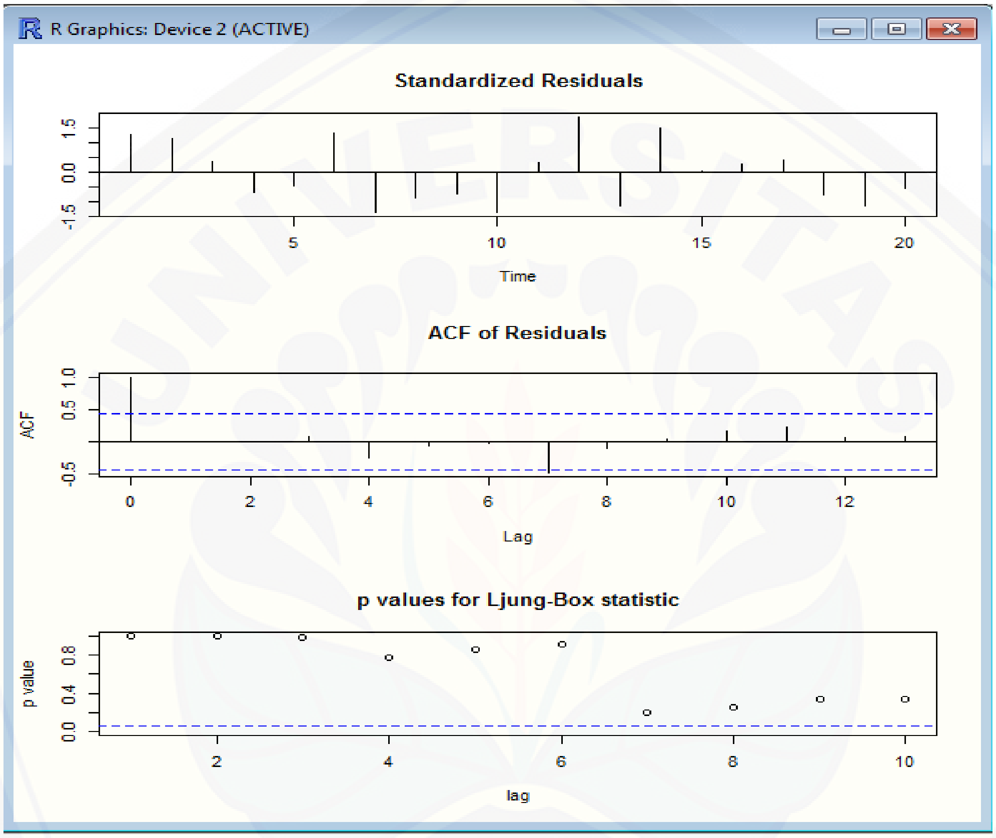
```
[925]  0.04209457 -0.16990373  0.30292019  0.24836937 -
 0.25265059  0.56786140
[931] -0.34379302 -0.59931421  0.04209457 -0.56223180 -
 0.64462599  0.04209457
[937]  0.04209457 -0.56223180  0.45561999  0.45561999 -
 0.25265059 -0.16990373
[943]  -0.34379302  -0.59931421  -0.46508280   0.45561999
 0.56786140  0.18808129
[949]  -0.59931421  -0.16990373  -0.25265059   0.04209457
 0.18808129  0.30292019
[955] -0.16990373  0.19536218  0.56786140 -0.34379302 -
 0.46508280  0.24836937
[961]  0.56786140 -0.59931421  0.04209457  0.24836937 -
 0.64462599 -0.64462599
[967]  -0.16990373   0.39598210   0.26046859  -1.14246417
 0.24836937  0.19536218
[973]  -0.46508280   0.24836937  -0.64462599   0.24836937
 0.24836937 -0.56223180
[979]  -0.25265059  -0.16990373   0.39598210  -0.59931421
 0.54215746 -0.14329381
[985]   0.54215746  -0.56223180  -0.14329381   0.30292019
 0.45561999  0.19536218
[991]   0.19536218  -0.34379302  -0.25265059  -0.46508280
 0.56786140  0.39598210
[997] -0.56223180  0.56786140  0.15326529  0.56786140
```

- **`tsdisplay(tsboot.a.1)`** (display data ACF dan PACF)



LAMPIRAN D. Skrip Program R untuk melihat model time series sebelum dan sesudah resampling

```
➤ fitsim1=arima(a.1,order=c(2,0,2))
tsdiag(fitsim1)
```

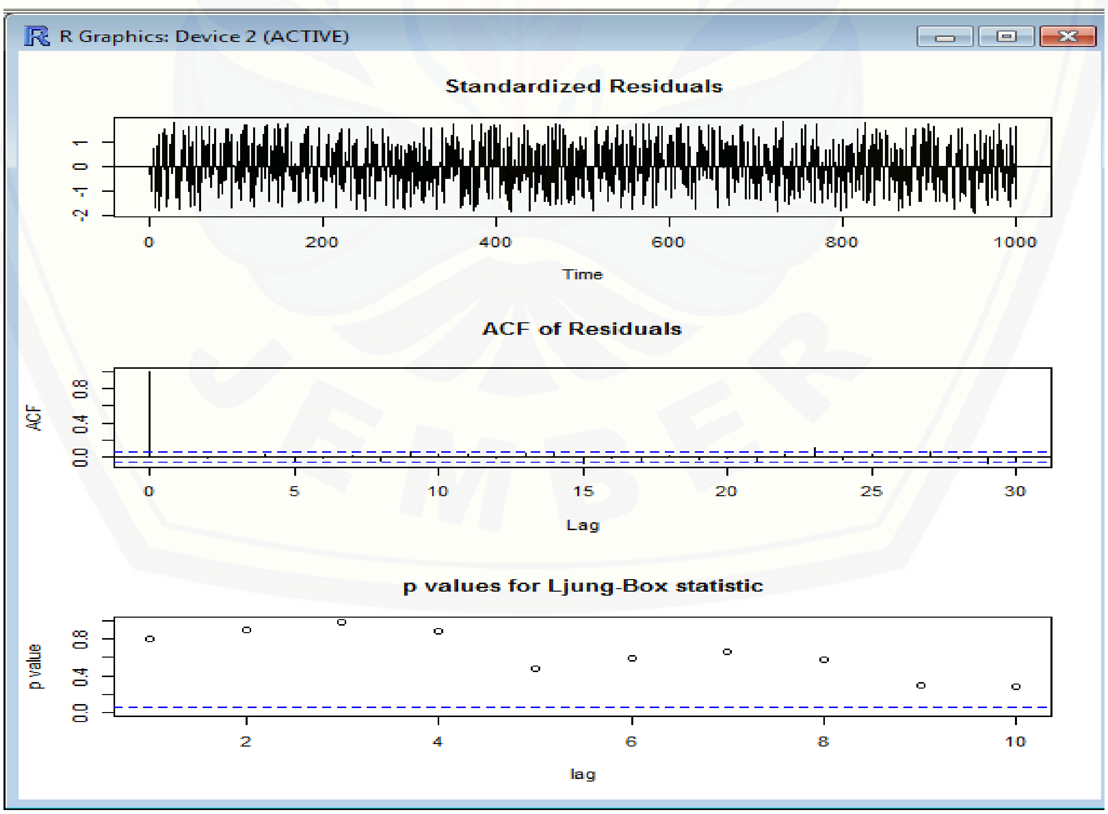



```
➤ fitsim11=arima(a.1,order=c(2,0,2))
> summary(fitsim11)

Call:
arima(x = a.1, order = c(2, 0, 2))

Coefficients:
          ar1     ar2     ma1      ma2  intercept
      -0.0292  0.1644  0.6107  -0.1379    -0.0146
```

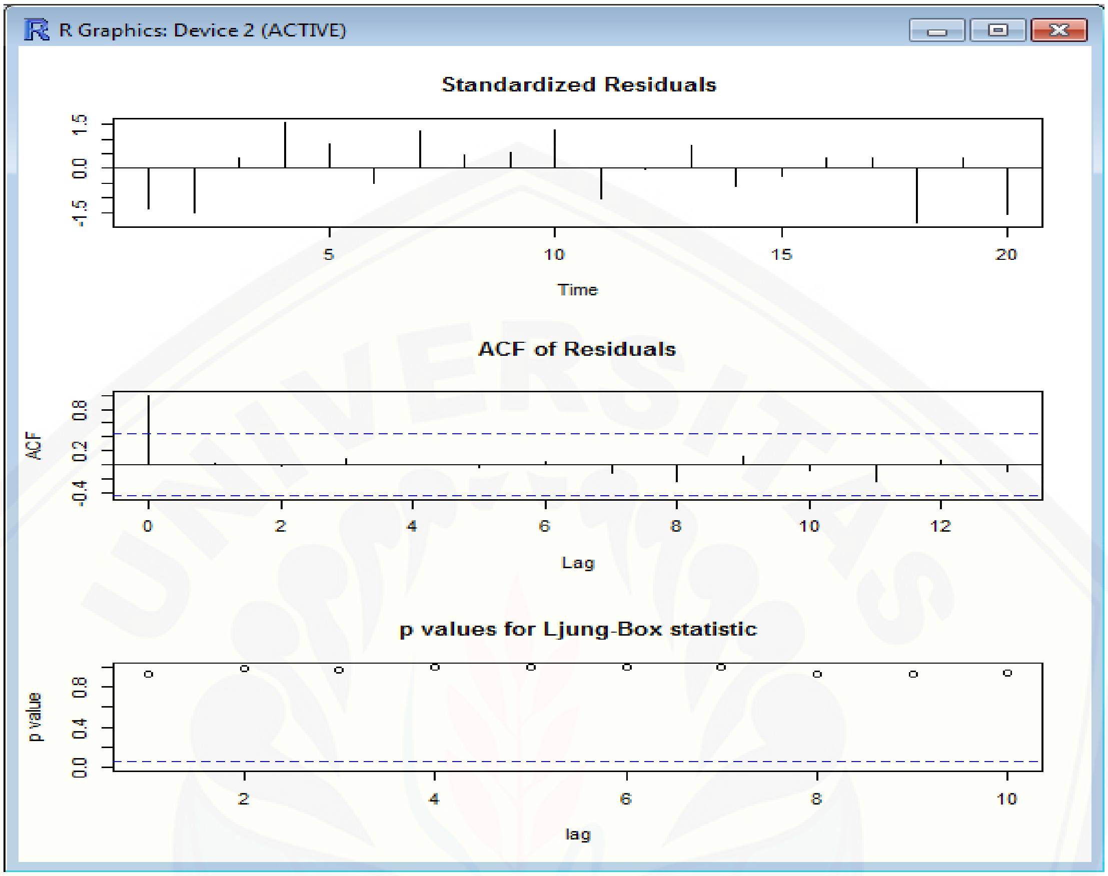
```
s.e.    0.8402   0.3257   0.8125    0.5782      0.1379

sigma^2 estimated as 0.1385:   log likelihood = -8.87,
aic = 29.74

Training set error measures:
                       ME         RMSE         MAE         MPE
MAPE       MASE
Training set -0.008463283 0.3721309 0.326398 106.2786
111.7791 0.8286921
                     ACF1
Training set 6.871139e-05
```

- fitboot1=arima(tsboot.a.1,order=c(2,0,2))

tsdiag(fitboot1)



```
➢ fitboot11=Arima(tsboot.a.1,order=c(2,0,2))
summary(fitboot11)
Series: tsboot.a.1
ARIMA(2,0,2) with non-zero mean

Coefficients:
          ar1     ar2     ma1      ma2  intercept
      -0.0970  0.5418  0.1202  -0.6006    -0.0104
s.e.   0.2744  0.2494  0.2627   0.2420     0.0129

sigma^2 estimated as 0.1894:  log likelihood=-587.01
AIC=1186.03   AICc=1186.11   BIC=1215.47

Training set error measures:
                        ME      RMSE       MAE      MPE
MAPE      MASE
Training set -7.347294e-05 0.4352053 0.3849773 98.68625
98.69397 0.7848566
                   ACF1
Training set 0.00762154

➢ fitsim2=arima(a.3,order=c(2,0,2))
tsdiag(fitsim2)
```

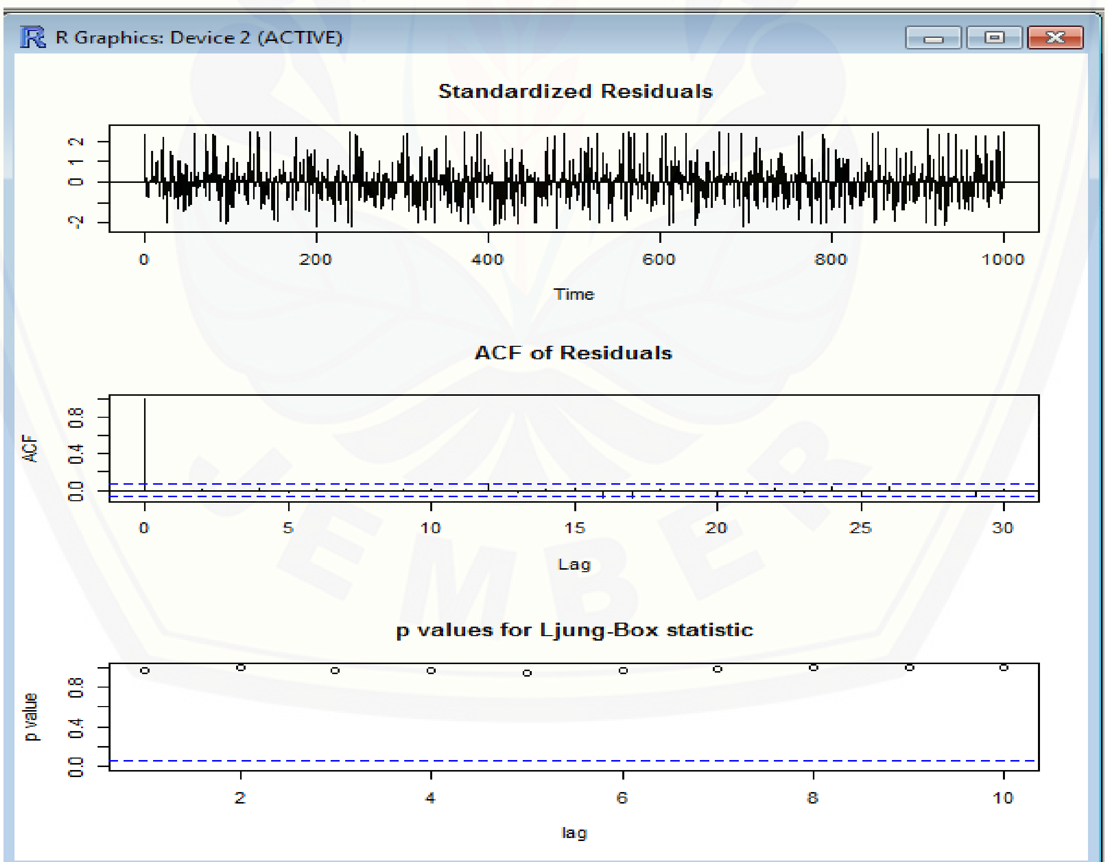



```
➤ fitsim22=arima(a.3,order=c(2,0,2))
summary(fitsim22)
Call:
arima(x = a.3, order = c(2, 0, 2))

Coefficients:
         ar1      ar2     ma1      ma2  intercept
      0.8602  -0.5156  0.0861  -0.0779     0.6800
s.e.  0.5097   0.3419  0.5506   0.5672     0.5443

sigma^2 estimated as 1.998:  log likelihood = -35.91,
aic = 83.82
```

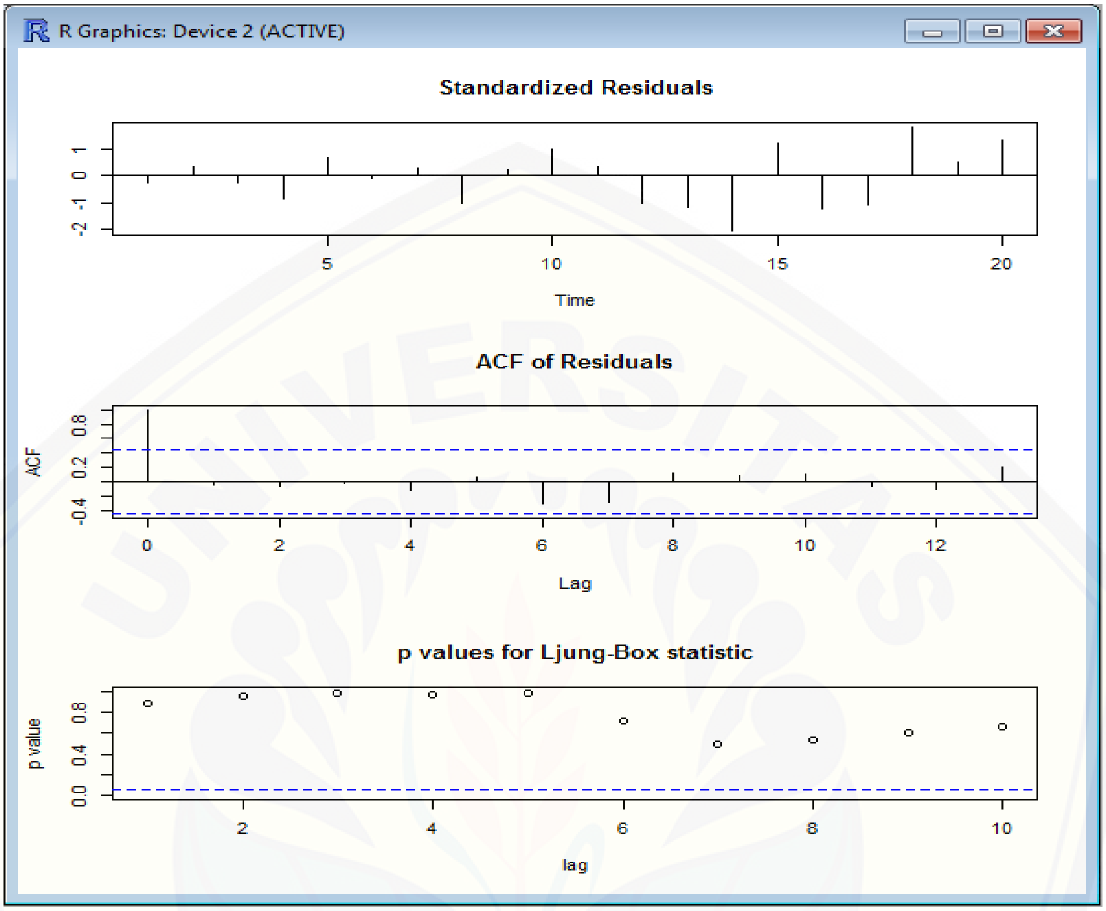
```
Training set error measures:
                     ME       RMSE        MAE        MPE
MAPE      MASE
Training  set  -0.04336214  1.413524  1.198427  2.786668
111.0589 0.8300051
                  ACF1
Training set 0.01799657
```

➤ `fitboot2=arima(tsboot.a.3,order=c(2,0,2))`

```
tsdiag(fitboot2)
```



```
➢ fitboot22=Arima(tsboot.a.3,order=c(2,0,2))
summary(fitboot22)
Series: tsboot.a.3
ARIMA(2,0,2) with non-zero mean

Coefficients:
         ar1      ar2      ma1     ma2  intercept
      0.2754  -0.2148  -0.3228  0.3223     0.6047
s.e.  0.2611   0.3247   0.2529  0.3151     0.0710

sigma^2 estimated as 4.453:  log likelihood=-2165.73
AIC=4343.46   AICc=4343.55   BIC=4372.91

Training set error measures:
                        ME       RMSE        MAE        MPE
MAPE      MASE
Training  set  -0.0001099277  2.110194  1.630352  34.50892
130.8991 0.6594536
                      ACF1
Training set -0.0009750868

➢ fitsim3=arima(a.5,order=c(2,0,2))
tsdiag(fitsim3)
```

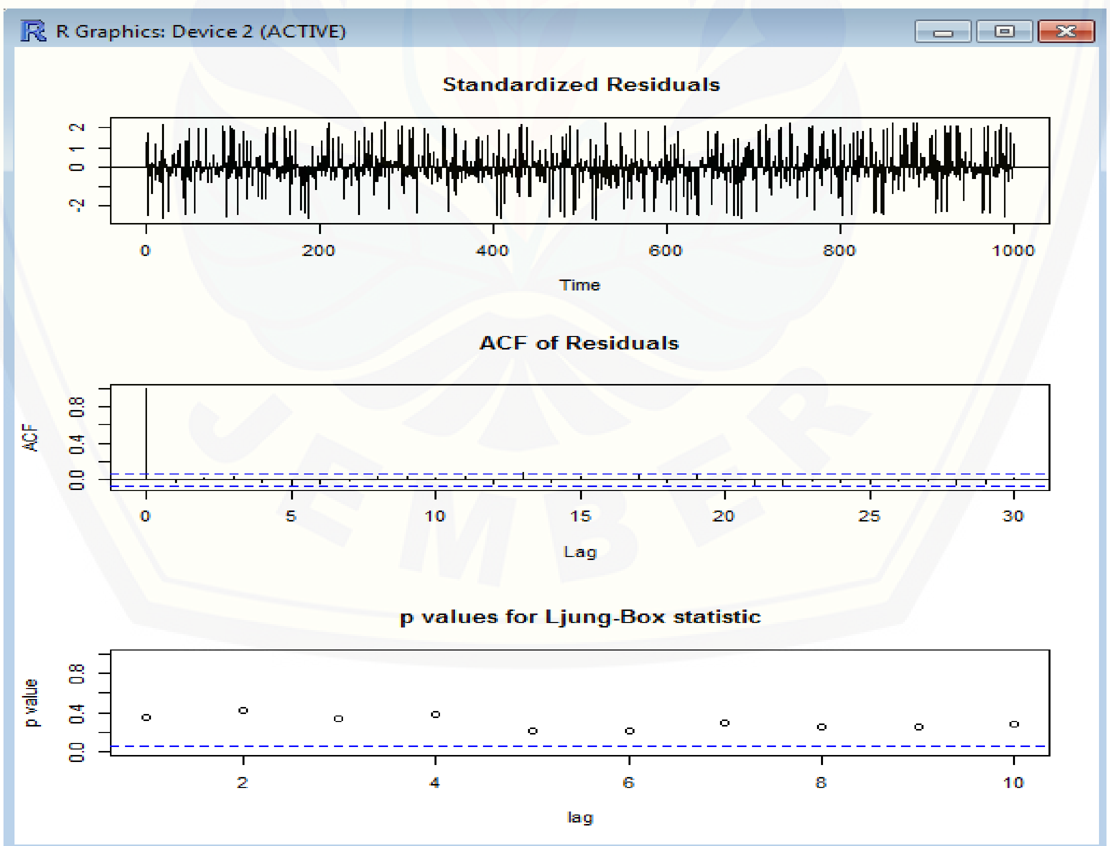



```
➢ fitsim33=arima(a.5,order=c(2,0,2))
summary(fitsim33)
Call:
arima(x = a.5, order = c(2, 0, 2))

Coefficients:
         ar1      ar2      ma1      ma2  intercept
      1.2600  -0.5828  -0.3437  -0.6563     0.0049
s.e.  0.2264   0.2711   0.3645   0.3473     0.0142
```

```
sigma^2 estimated as 0.004509:  log likelihood = 23.79,
aic = -35.58

Training set error measures:
                      ME       RMSE        MAE        MPE
MAPE     MASE
Training set -0.00398305 0.06715012 0.05653555 -140.0044
268.0055 0.7007077
                    ACF1
Training set -0.03063933
```

- fitboot3=arima(tsboot.a.5,order=c(2,0,2))

```
tsdiag(fitboot3)
```



```
➢ fitboot33=Arima(tsboot.a.5,order=c(2,0,2))
summary(fitboot33)
Series: tsboot.a.5
ARIMA(2,0,2) with non-zero mean

Coefficients:
          ar1      ar2     ma1     ma2  intercept
      -0.8752  -0.9350  0.9282  0.9874     0.0203
s.e.   0.0155   0.0167  0.0073  0.0087     0.0039

sigma^2 estimated as 0.01421:  log likelihood=707.28
AIC=-1402.56   AICc=-1402.48   BIC=-1373.12

Training set error measures:
                       ME      RMSE        MAE      MPE
MAPE      MASE
Training  set  2.320768e-06  0.1191939  0.08634195  55.46451
158.2836 0.6617642
                    ACF1
Training set -0.02962321
```