



**PENERAPAN METODE *EXTENDED KALMAN FILTER*
PADA KASUS PERTUMBUHAN PENDUDUK
KABUPATEN JEMBER**

SKRIPSI

Oleh

**Rory Ronella Agustin
NIM 1518101019**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2019**



**PENERAPAN METODE *EXTENDED KALMAN FILTER*
PADA KASUS PERTUMBUHAN PENDUDUK
KABUPATEN JEMBER**

SKRIPSI

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

Oleh

**Rory Ronella Agustin
NIM 1518101019**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2019**

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Ibunda Rohaniyah Kusniawati dan Ayahanda Mistori tercinta yang telah memberikan dukungan, semangat, kasih sayang, perhatian, dan pengorbanan yang begitu besar serta doa yang tak pernah putus untuk anaknya;
2. Guru-guru sejak dari sekolah dasar sampai dengan perguruan tinggi yang telah mendidik dan membimbing selama ini;
3. Almamater tercinta Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

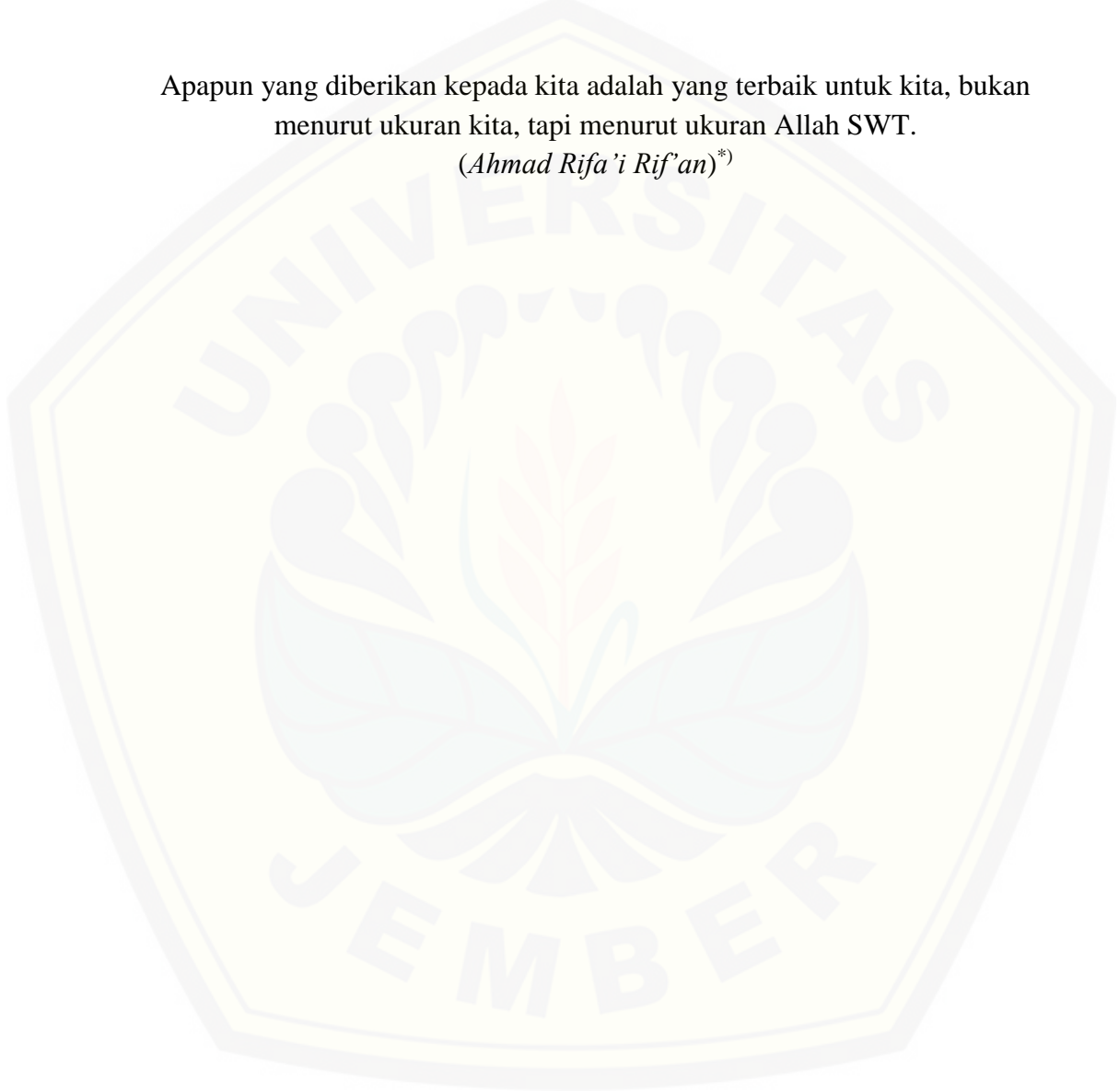
MOTTO

Kepuasan sebenarnya terletak dalam usaha yang kita lakukan
bukan dalam puncak pencapaiannya.

*(Mohandas Gandhi)**

Apapun yang diberikan kepada kita adalah yang terbaik untuk kita, bukan
menurut ukuran kita, tapi menurut ukuran Allah SWT.

*(Ahmad Rifa' i Rif'an)**



^{*)} Rif'an, A.R. 2011. *Man Shabara Zhafira*. Jakarta : PT Elex Media Komputindo.

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

nama : Rory Ronella Agustin

NIM : 151810101019

menyatakan dengan sesungguhnya bahwa skripsi yang berjudul "Penerapan Metode *Extended Kalman Filter* pada Kasus Pertumbuhan Penduduk Kabupaten Jember" adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi mana pun, 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, Juli 2019

Yang menyatakan,

Rory Ronella Agustin

NIM 151810101019

SKRIPSI

**PENERAPAN METODE *EXTENDED KALMAN FILTER*
PADA KASUS PERTUMBUHAN KABUPATEN JEMBER**

Oleh

Rory Ronella Agustin
NIM 1518101019

Pembimbing

Dosen Pembimbing Utama : Kosala Dwidja Purnomo, S.Si.,M.Si.

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

PENGESAHAN

Skripsi berjudul "Penerapan Metode *Extended Kalman Filter* pada Kasus Pertumbuhan Penduduk Kabupaten Jember" telah diuji dan disahkan pada:

hari, tanggal :

tempat : Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan
Alam Universitas Jember

Tim Penguji:

Ketua,

Anggota I,

Kosala Dwidja Purnomo, S.Si., M.Si
NIP. 196908281998021001

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

Anggota II,

Anggota III,

Dr. Firdaus Ubaidillah, S.Si.,M.Si
NIP.197006061998031003

Dian Anggraeni, S.Si.,M.Si
NIP.198202162006042002

Mengesahkan
Dekan,

Drs. Sujito, Ph.D.
NIP. 196102041987111001

RINGKASAN

Penerapan Metode *Extended Kalman Filter* pada Kasus Pertumbuhan Penduduk Kabupaten Jember; Rory Ronella Agustin; 151810101019; 2019; 33 halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Jember adalah sebuah wilayah kabupaten yang merupakan bagian dari wilayah Provinsi Jawa Timur. Kabupaten Jember merupakan Kabupaten ke-3 yang memiliki jumlah penduduk terbanyak di Jawa Timur. Jumlah penduduk yang besar merupakan suatu hal positif karena dapat dijadikan sebagai subjek pembangunan, dan perkembangan perekonomian. Namun di sisi lain, penduduk merupakan beban bagi pembangunan berkaitan dengan pemenuhan kebutuhan seiring dengan perkembangan jumlah penduduk tersebut. Nelson dan Leibstein menunjukkan bahwa pertumbuhan penduduk yang pesat di negara berkembang menyebabkan tingkat kesejahteraan masyarakat tidak mengalami perbaikan yang berarti dan dalam jangka panjang akan mengalami penurunan kesejahteraan serta meningkatkan jumlah penduduk miskin. Metode *Extended Kalman Filter* digunakan untuk mengestimasi hasil selanjutnya berdasarkan data-data yang telah ada sebelumnya. Pada penelitian ini metode *Extended Kalman Filter* diaplikasikan untuk mengestimasi pertumbuhan penduduk Kabupaten Jember dengan membandingkan dua model pertumbuhan logistik yaitu model pertumbuhan logistik dengan asumsi fungsi populasi linier dan model pertumbuhan logistik dengan asumsi fungsi populasi parabolik. Tujuan dari perbandingan asumsi dua model tersebut adalah untuk menentukan asumsi model yang sesuai dengan pertumbuhan penduduk Kabupaten Jember.

Hasil estimasi didapatkan dengan melakukan beberapa langkah, yaitu melakukan pengumpulan data jumlah penduduk Kabupaten Jember tahun 1990-2017, menentukan laju pertumbuhan penduduk dan ambang batas populasi Kabupaten Jember, melakukan diskritisasi pada model pertumbuhan logistik dengan asumsi fungsi populasi linier menggunakan metode beda hingga maju,

menghitung matriks Jacobi dan menambahkan *noise* pada model pertumbuhan logistik, mengimplementasikan algoritma EKF, serta menganalisis hasil simulasi penggunaan metode EKF.

Simulasi dilakukan sebanyak tiga kali menggunakan 6 data pengukuran, 14 data pengukuran dan 28 data pengukuran. Analisis yang dilakukan adalah membandingkan model pertumbuhan logistik yang digunakan. Hasil simulasi menunjukkan bahwa model pertumbuhan logistik dengan asumsi fungsi populasi parabolik lebih sesuai dengan pertumbuhan penduduk Kabupaten Jember pada keseluruhan simulasi yang dilakukan, baik menggunakan 6 data pengukuran, 14 data pengukuran, dan 28 data pengukuran. Hasil simulasi menunjukkan bahwa metode *Extended Kalman Filter* secara umum cukup baik digunakan untuk mengestimasi pertumbuhan penduduk Kabupaten Jember. Hal ini ditunjukkan dengan hasil norm kovariansi error yang memiliki rata-rata 0,022477 saat menggunakan 6 data pengukuran, 0,010042 saat menggunakan 14 data pengukuran, 0,0010406 saat menggunakan 28 data pengukuran. Semakin banyak data pengukuran yang digunakan, hasil estimasi yang dihasilkan semakin baik dan mendekati data sebenarnya.

PRAKATA

Puji syukur ke hadirat Allah SWT, atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul "Penerapan Metode *Extended Kalman Filter* pada Kasus Pertumbuhan Penduduk Kabupaten Jember". 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 bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Bapak Kosala Dwidja Purnomo, S.Si., M.Si. selaku Dosen Pembimbing Utama dan Bapak Dr. Alfian Futuhul Hadi, S.Si.,M.Si selaku Dosen Pembimbing Anggota yang telah membimbing dan memberikan arahan kepada penulis;
2. Bapak Dr. Firdaus Ubaidillah, S,Si.,M.Si dan Ibu Dian Anggraeni, S.Si.,M.Si selaku Dosen Penguji yang telah memberikan masukan, saran dan kritik yang membangun dalam penyusunan skripsi ini;
3. Seluruh dosen dan karyawan Jurusan Matematika Fakultas MIPA Universitas Jember;
4. Seluruh Anggota UKM SPORA yang telah memberikan ilmu dan pengalaman selama berorganisasi, terutama Bidang Olahraga 2015/2016, Bidang PSM 2017, dan Inti BPH UKM SPORA 2018;
5. Dyan Evi Susanti, Ulil Abshari, Sutitah, Laylatul Febriana Nilasari, Wahdania Eka Putri, Miftachul Jannah, Yulianis Satur Rohmah, Melinda Sari Juanda, Heni Nursafitri atas keceriaan dan motivasinya;
6. Teman-teman seperjuangan Sigma Fifteen angkatan 2015 di Jurusan Matematika dan semua pihak yang tidak dapat disebutkan satu per satu.

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

Jember, Juli 2019

Penulis



DAFTAR ISI

	Halaman
HALAMAN JUDUL	ii
HALAMAN PERSEMBAHAN	iii
HALAMAN MOTTO.....	iv
HALAMAN PERNYATAAN.....	v
HALAMAN PEMBIMBINGAN.....	vi
HALAMAN PENGESAHAN.....	vii
RINGKASAN	viii
PRAKATA	x
DAFTAR ISI.....	xii
DAFTAR TABEL	xiv
DAFTAR GAMBAR.....	xv
DAFTAR LAMPIRAN	xvi
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 Pertumbuhan Penduduk.....	5
2.2 Model Pertumbuhan Logistik	6
2.3 Metode Beda Hingga	8
2.4 Metode Kalman Filter	9
2.5 Metode <i>Extended Kalman Filter</i>	11
2.6 Norm Kovariansi Error	14
BAB 3. METODE PENELITIAN	15
BAB 4. HASIL DAN PEMBAHASAN	19
4.1 Penentuan r dan K	19

4.2	Diskritisasi Model.....	21
4.3	Perhitungan Matriks Jacobi dan Penambahan <i>Noise</i>	22
4.4	Penerapan Algoritma <i>Extended Kalman Filter</i>	24
4.5	Analisis Hasil Estimasi	26
4.5.1	Hasil Estimasi dengan 6 data Pengukuran	28
4.5.2	Hasil Estimasi dengan 14 data Pengukuran	30
4.5.3	Hasil estimasi dengan 28 data pengukuran	31
BAB 5.	PENUTUP.....	34
5.1	Kesimpulan	34
5.2	Saran.....	34
DAFTAR PUSTAKA	35
LAMPIRAN	37

DAFTAR TABEL

	Halaman
Tabel 2.1 Persamaan – persamaan <i>Extended Kalman Filter</i>	13
Tabel 4.1 Jumlah Penduduk Kabupaten Jember tahun 1990 – 2017	19



DAFTAR GAMBAR

	Halaman
Gambar 2.1 Jumlah Penduduk Kabupaten Jember tahun 2016-2017	5
Gambar 2.2 Grafik Hubungan $f(N)$ dan N	6
Gambar 3.1 Skema Metode Penelitian.....	18
Gambar 4.1 Grafik Solusi Analitik untuk Model Linier.....	27
Gambar 4.2 Grafik Solusi Analitik untuk Model Parabolik	28
Gambar 4.3 Grafik Hasil Estimasi $N(t = 6)$	29
Gambar 4.4 Grafik Norm Kovariansi Error Metode EKF ($t = 6$).....	30
Gambar 4.5 Grafik Hasil Estimasi $N(t = 14)$	30
Gambar 4.6 Grafik Norm Kovariansi Error Metode EKF ($t = 14$).....	31
Gambar 4.7 Grafik Hasil Estimasi $N(t = 28)$	32
Gambar 4.8 Grafik Norm Kovariansi Error Metode EKF ($t = 28$).....	33

DAFTAR LAMPIRAN

	Halaman
A.1 Hasil Estimasi dengan 6 Data Pengukuran	37
A.2 Absolut Error Hasil Estimasi dengan 6 data pengukuran	38
A.3 Norm Kovariansi Error dengan 6 Data Pengukuran	39
A.4 Hasil Estimasi dengan 14 data pengukuran	40
A.5 Absolut Error Hasil Estimasi dengan 14 data pengukuran	41
A.6 Norm Kovariansi Error Hasil Estimasi dengan 14 data pengukuran	42
A.7 Hasil Estimasi dengan 28 data pengukuran	43
A.8 Absolut Error Hasil Estimasi dengan 28 data pengukuran	41
A.9 Norm Kovariansi Error Hasil Estimasi dengan 28 data pengukuran	42
B. Skrip Program Metode EKF	40

BAB 1. PENDAHULUAN

1.1 Latar belakang

Jember adalah sebuah wilayah kabupaten yang merupakan bagian dari wilayah Provinsi Jawa Timur. Dalam konteks regional, Kabupaten Jember mempunyai kedudukan dan peran strategis sebagai salah satu pusat kegiatan. Wilayah administrasi Kabupaten Jember terdiri dari 31 wilayah Kecamatan dan 248 desa atau kelurahan dengan luas daerah 3.293,34 km². Kabupaten Jember merupakan Kabupaten ke-3 yang memiliki jumlah penduduk terbanyak di Jawa Timur. Kepadatan Penduduk Kabupaten Jember tahun 2017 mencapai 786 jiwa/km². Kepadatan penduduk di 31 Kecamatan cukup beragam dengan tertinggi terletak di Kecamatan Kaliwates yang memiliki kepadatan penduduk sebesar 4.651 jiwa/km² dan kepadatan penduduk terendah di Kecamatan Tempurejo sebesar 140 jiwa/km² (BPS Jawa Timur, 2018).

Menurut Rochaida (2016) dalam penelitiannya, pertumbuhan penduduk sebenarnya merupakan keseimbangan dinamis antara dua kekuatan yang menambah atau mengurangi jumlah penduduk. Jumlah penduduk yang besar merupakan suatu hal positif karena dapat dijadikan sebagai subjek pembangunan, dan perkembangan perekonomian. Namun di sisi lain, penduduk merupakan beban bagi pembangunan berkaitan dengan pemenuhan kebutuhan seiring dengan perkembangan jumlah penduduk tersebut. Hal ini didukung oleh teori Malthus (dikutip dari Iswanto, 2012) yang menyatakan bahwa penduduk cenderung tumbuh secara deret ukur, sedangkan persediaan makanan bertumbuh secara deret hitung. Pertumbuhan penduduk sering dihubungkan dengan kesejahteraan masyarakat. Menurut Nelson dan Leibstein (dikutip dari Sukirno, 2000) terdapat pengaruh langsung antara penambahan penduduk terhadap tingkat kesejahteraan masyarakat. Nelson dan Leibstein menunjukkan bahwa pertumbuhan penduduk yang pesat di negara berkembang menyebabkan tingkat kesejahteraan masyarakat tidak mengalami perbaikan yang berarti dan dalam jangka panjang akan mengalami penurunan kesejahteraan serta meningkatkan jumlah penduduk miskin.

Pengaruh jumlah penduduk telah banyak diteliti diantaranya adalah Firdhania dan Muslihatinningsih (2017) menganalisis faktor yang mempengaruhi tingkat pengangguran dalam jurnalnya yang berjudul "Faktor-faktor yang mempengaruhi Tingkat Pengangguran di Kabupaten Jember". Penelitian tersebut menghasilkan kesimpulan jumlah penduduk, inflasi upah minimum, indeks pembangunan manusia berpengaruh signifikan terhadap tingkat pengangguran di Kabupaten Jember. Jumlah penduduk ternyata memiliki pengaruh penting diantaranya dalam hal tingkat pengangguran dan pertumbuhan ekonomi, sehingga perlu adanya estimasi jumlah penduduk sebagai salah satu cara sebagai perencanaan kesejahteraan masyarakat. Pertumbuhan penduduk dapat diestimasi menggunakan Analisis Kalman Filter.

Kalman Filter merupakan suatu metode yang digunakan untuk melakukan estimasi suatu nilai. *Extended Kalman Filter* merupakan salah satu modifikasi dari Algoritma Kalman Filter. *Extended Kalman Filter* digunakan pada model sistem dinamik non linier. *Extended Kalman Filter* digunakan untuk mengestimasi hasil selanjutnya berdasarkan data-data yang telah ada sebelumnya. Dalam *Extended Kalman Filter* sistem semacam ini perlu dilinierisasi (apabila sistem nonlinier), dilakukan pendiskritan sistem (apabila sistem kontinu), dan beberapa tahapan yang lain (Ichwan, 2010).

Extended Kalman Filter banyak diaplikasikan dalam berbagai permasalahan. Ichwan (2010) menerapkan metode *Extended Kalman Filter* untuk mengestimasi posisi dan kecepatan kapal selam pada model dinamika gerak kapal selam. Wicaksono (2010) mengaplikasikan algoritma *Extended Kalman Filter* dalam mendeteksi kerak yang terjadi akibat sistem perpindahan panas antara dua fluida pada alat penukar panas dengan menggunakan model yang merepresentasikan kondisi alat penukar panas. Prayudi (2012) mengaplikasikan metode *Extended Kalman Filter* pada model *Predator-Prey* dan menerapkan model tersebut pada suatu permasalahan di bidang pertanian atau perkebunan. Hasil penelitian sebelumnya, Purnomo (2008) tentang Kalman Filter mengestimasi pertumbuhan populasi plankton menyatakan penggunaan metode *Extended Kalman Filter* lebih cocok digunakan dalam pengestimasi daripada

metode *Ensemble Kalman Filter* pada sebagian variabel tertentu. Fitriani (2013) melakukan penelitian tentang tentang model pertumbuhan logistik namun belum diaplikasikan dalam permasalahan pertumbuhan populasi.

Pada penelitian ini metode *Extended Kalman Filter* diaplikasikan untuk mengestimasi pertumbuhan penduduk Kabupaten Jember. Estimasi ini dilakukan dengan membandingkan dua asumsi model antara lain model persamaan pertumbuhan logistik dengan asumsi parabolik terbuka ke bawah yang telah diteliti oleh Fitriani (2013) dan model persamaan linier. Tujuan dari perbandingan asumsi dua model tersebut adalah untuk menentukan asumsi model yang sesuai dengan pertumbuhan penduduk Kabupaten Jember.

1.2 Rumusan masalah

Rumusan masalah yang akan dibahas dalam penelitian ini adalah:

1. Bagaimana perbandingan estimasi antara model pertumbuhan logistik dengan asumsi fungsi populasi linier dan model pertumbuhan logistik asumsi fungsi populasi parabolik terbuka ke bawah?
2. Bagaimana hasil penerapan *Extended Kalman Filter* dalam estimasi pertumbuhan penduduk Kabupaten Jember?

1.3 Tujuan

Tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Mengetahui model pertumbuhan logistik yang sesuai dengan pertumbuhan penduduk Kabupaten Jember.
2. Mengetahui hasil penerapan *Extended Kalman Filter* dalam estimasi jumlah penduduk Kabupaten Jember.

1.4 Manfaat

Manfaat yang diperoleh dalam penelitian ini adalah:

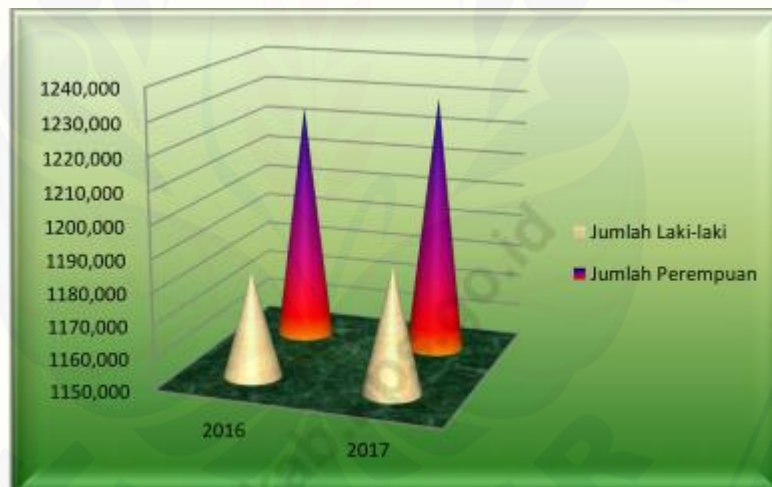
1. Mengetahui model yang sesuai dengan pertumbuhan penduduk Kabupaten Jember.
2. Memberikan alternatif metode penyelesaian dalam mengestimasi pertumbuhan penduduk Kabupaten Jember.



BAB 2. TINJAUAN PUSTAKA

2.1 Pertumbuhan Penduduk

Pertumbuhan penduduk adalah perubahan jumlah penduduk di suatu wilayah tertentu pada waktu tertentu dibandingkan waktu sebelumnya (Rochaida, 2016). Sumber utama data kependudukan adalah sensus penduduk yang dilaksanakan setiap sepuluh tahun sekali. Metode pengumpulan data dalam sensus dilakukan dengan wawancara antara petugas sensus dengan responden dan melalui *e-census*. Pada tahun yang tidak dilaksanakan sensus penduduk, data kependudukan diperoleh dari hasil proyeksi penduduk. Proyeksi penduduk merupakan suatu perhitungan ilmiah yang didasarkan pada asumsi dari komponen-komponen perubahan penduduk, yaitu kelahiran, kematian, dan migrasi (BPS Jawa Timur, 2018).



Gambar 2.1 Jumlah Penduduk Kabupaten Jember tahun 2016-2017

(Sumber: BPS Jember, 2017)

Berdasarkan Gambar 2.1, jumlah penduduk Kabupaten Jember berjumlah 2.430.185 jiwa yang terdiri atas 1.281.579 jiwa penduduk perempuan dan 1.194.496 jiwa penduduk laki-laki. Jumlah penduduk meningkat sebanyak 11.185 jiwa atau 0,46 persen dari tahun sebelumnya. Laju pertumbuhan penduduk adalah angka yang menunjukkan presentase penambahan penduduk dalam jangka waktu tertentu (r) (BPS Jember, 2017). Ambang batas populasi adalah nilai maksimum

pertumbuhan populasi (K). Laju pertumbuhan dan ambang batas populasi akan digunakan dalam estimasi pertumbuhan penduduk.

2.2 Model Pertumbuhan Logistik

Model Pertumbuhan logistik adalah model pertumbuhan populasi yang mempertimbangkan keterbatasan lingkungan, sehingga laju pertumbuhan populasi tergantung pada kerapatan populasinya. Model pertumbuhan logistik diperkenalkan oleh Verhulst pada tahun 1830 untuk menggambarkan pertumbuhan suatu populasi. Model pertumbuhan logistik ini menghasilkan solusi berbentuk fungsi yang monoton naik atau monoton turun (Rohaeni, 2017).

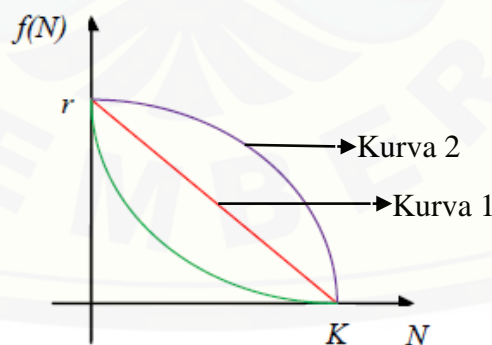
Persamaan pertumbuhan logistik dapat ditulis sebagai persamaan (2.1) berikut:

$$\frac{dN(t)/dt}{N(t)} = f(N(t)) \text{ atau } \frac{dN(t)}{dt} = f(N(t))N(t) \quad (2.1)$$

$N(t)$ adalah banyaknya populasi pada waktu t dan $\frac{dN(t)}{dt}$ adalah laju perubahan populasi terhadap waktu t . Persamaan (2.1) didapatkan jika laju reproduksi pertumbuhan tidak konstan, sehingga diwujudkan dalam sebuah fungsi $f(N)$. Jika laju perubahan populasi sebanding dengan banyaknya populasi yang ada dan memiliki laju reproduksi yang konstan, maka:

$$\frac{dN(t)}{dt} = kN(t) \text{ atau } \frac{dN(t)/dt}{N(t)} = k \quad (2.2)$$

dengan k merupakan konstanta dan disebut sebagai laju reproduksi.



Gambar 2.2 Grafik Hubungan $f(N)$ dan N

(Sumber: Fitriani, 2013)

Berdasarkan Gambar 2.2 diasumsikan $f(N(t))$ linier, yaitu $f(N(t)) = c_1N(t) + c_2$. Jika kita menggunakan kondisi $f(0) = r$ dimana r adalah laju pertumbuhan dan $f(K) = 0$ (K adalah *carrying capacity* atau ambang batas

populasi) didapatkan $c_2 = r$ dan $c_1 = -r/K$. Sehingga diperoleh bentuk $f(N(t)) = r - \frac{r}{K} N(t)$. Oleh karena itu, persamaan logistik dengan $f(N(t))$ linier adalah:

$$\frac{dN(t)}{dt} = \left(r - \frac{r}{K} N(t)\right)N(t) \quad (2.3)$$

(Zill, 2005)

Persamaan (2.3) mempunyai solusi analitik sebagai berikut:

$$N(t) = \frac{K}{1 + e^{-rt} c_K} \quad (2.4)$$

Solusi analitik merupakan solusi ideal dari persamaan yang akan dibandingkan dengan hasil estimasi menggunakan metode *Extended Kalman Filter*. Sebelum model pertumbuhan logistik diimplementasikan dalam metode *Extended Kalman Filter*, perlu dilakukan proses diskritisasi model (Fitriani, 2013). Diskritisasi model pertumbuhan logistik menggunakan metode beda hingga maju:

$$\frac{dN(t)}{dt} = \frac{N_{k+1} - N_k}{\Delta t} \quad (2.5)$$

Asumsi $f(N(t))$ linier (kurva 1) dikembangkan menjadi bentuk parabolik (kurva 2) seperti pada Gambar 2.2, dengan persamaan sebagai berikut:

$$f(N(t)) = a N^2(t) + b \quad (2.6)$$

dengan a dan b adalah konstanta.

Jika digunakan kondisi $f(0) = r$ dan $f(K) = 0$, maka didapatkan konstanta $a = -\frac{r}{K^2}$ dan $b = r$. Sehingga diperoleh bentuk $f(N(t)) = -\frac{r}{K^2} (N(t))^2 + r$. Oleh karena itu, persamaan logistik dengan $f(N(t))$ parabolik terbuka ke bawah adalah:

$$\frac{d(N(t))}{dt} = \left(-\frac{r}{K^2} (N(t))^2 + r\right) N(t)$$

atau

$$\frac{d(N(t))}{dt} = -\frac{r}{K^2} (N(t))^3 + r N(t) \quad (2.7)$$

Persamaan (2.7) mempunyai solusi analitik sebagai berikut:

$$N(t) = \sqrt{\frac{K^2}{1 + 2r e^{-2rt} c}} \quad (2.8)$$

Persamaan (2.9) merupakan hasil diskritisasi model pertumbuhan logistik dengan asumsi fungsi populasi parabolik terbuka ke bawah, didapatkan dengan mensubstitusikan persamaan (2.7) dalam persamaan (2.5).

$$\frac{N_{k+1}-N_k}{\Delta t} = -\frac{r}{K^2}N_k^3 + rN_k \quad \text{atau} \quad N_{k+1} = \left(-\frac{r}{K^2}N_k^3 + rN_k\right)\Delta t + N_k \quad (2.9)$$

2.3 Metode Beda Hingga

Deret Taylor dapat menyelesaikan persamaan differensial dalam permasalahan metode numerik. Differensial numerik digunakan untuk merubah bentuk persamaan differensial kontinu menjadi bentuk diskrit. Deret Taylor dituliskan pada persamaan (2.11)

$$f(x_{k+1}) = f(x_k) + \Delta x f'(x_k) + \frac{\Delta x^2}{2!} f''(x_k) + \dots \quad (2.10)$$

$$f(x_{k-1}) = f(x_k) - \Delta x f'(x_k) - \frac{\Delta x^2}{2!} f''(x_k) - \dots \quad (2.11)$$

Persamaan (2.10) dapat ditulis dalam bentuk:

$$f(x_{k+1}) = f(x_k) + \Delta x f'(x_k) + O(\Delta x)$$

atau

$$\begin{aligned} \frac{\partial f}{\partial x} &= f'(x_k) = \frac{f(x_{k+1})-f(x_k)}{\Delta x} + O(\Delta x) \\ \frac{\partial f}{\partial x} &\approx \frac{f(x_{k+1})-f(x_k)}{\Delta x} \end{aligned} \quad (2.12)$$

Persamaan (2.11) dapat ditulis dalam bentuk:

$$f(x_{k-1}) = f(x_k) - \Delta x f'(x_k) - O(\Delta x)$$

atau

$$\begin{aligned} \frac{\partial f}{\partial x} &= f'(x_k) = \frac{f(x_k)-f(x_{k-1})}{\Delta x} - O(\Delta x) \\ \frac{\partial f}{\partial x} &\approx \frac{f(x_k)-f(x_{k-1})}{\Delta x} \end{aligned} \quad (2.13)$$

Persamaan (2.12) disebut persamaan differensial beda hingga maju, karena menggunakan data pada titik x_k dan x_{k+1} . Persamaan (2.13) disebut persamaan differensial beda hingga mundur, karena menggunakan data x_k dan x_{k-1} . Sedangkan persamaan differensial beda hingga terpusat menggunakan data x_{k-1} dan x_{k+1} . Persamaan differensial beda hingga terpusat didapatkan dengan cara mengurangi persamaan (2.10) dengan (2.11), ditunjukkan dengan persamaan (2.14) berikut:

$$f(x_{k+1}) - f(x_{k-1}) = 2f'(x_k) - \Delta x + f'''(x_k) \frac{\Delta x^3}{3!} + \dots$$

atau

$$\begin{aligned} \frac{\partial f}{\partial x} &= f'(x_k) = \frac{f(x_{k+1}) - f(x_{k-1})}{2\Delta x} - O(\Delta x) \\ \frac{\partial f}{\partial x} &\approx \frac{f(x_{k+1}) - f(x_{k-1})}{2\Delta x} \end{aligned} \quad (2.14)$$

(Triatmodjo, 1992).

Metode pendiskritan yang akan digunakan dalam model pertumbuhan logistik adalah metode beda hingga maju karena metode ini lebih sesuai dengan model pertumbuhan logistik, dengan menggunakan data pada saat k dan yang akan dicari adalah data saat $k + 1$.

2.4 Metode Kalman Filter

Kalman Filter merupakan salah satu metode estimasi suatu variabel keadaan dari sistem dinamik stokastik linear diskrit dengan meminimumkan *mean error*. Kalman Filter merupakan sekumpulan persamaan matematika yang diterapkan sebagai estimator yang meminimalkan error kovarian terestimasi sehingga diperoleh sistem yang optimal. Proses identifikasi sistem termasuk sistem dengan *noise* memanfaatkan kemampuan estimasi Kalman filter tersebut untuk memperoleh sistem yang sesungguhnya (Susanto, 2008).

Keunggulan metode Kalman Filter adalah kemampuannya dalam mengestimasi suatu keadaan berdasarkan data yang minim. Kalman Filter merupakan suatu metode yang menggabungkan model dan pengukuran. Data pengukuran terbaru digunakan sebagai pembandingan hasil prediksi, sehingga hasil estimasinya selalu mendekati kondisi yang sebenarnya (Masduqi dan Apriliani, 2008).

Berikut adalah persamaan dasar yang digunakan dalam estimasi:

$$x_{k+1} = A_x x_k + B_k u_k + w_k \quad (2.15)$$

dengan pengukuran $z_k \in R^p$ yang memenuhi $z_k = H_k x_k + v_k$

dengan kondisi awal $x_0 = \bar{x}_0, x_0 \sim N(\bar{x}_0, P_{x_0}); w_k \sim N(0, Q_k)$

keterangan:

x_0 = Inisial dari sistem,

- x_{k+1} = Variabel keadaan pada waktu $k + 1$ dan berdimensi $n \times 1$
 x_k = Variabel keadaan pada waktu k dan berdimensi $n \times 1$
 u_k = Vektor masukan deterministik pada waktu k dan berdimensi $m \times 1$
 w_k = *Noise* model sistem pada waktu k dan berdimensi $m \times 1$ dengan *mean* nol dan kovarian yang diketahui, yaitu Q atau ditulis $E[w_k] = 0$ dan $Cov\{w_k, w_k\} = E[w_k w_k^T] = Q$
 z_k = Vektor pengukuran pada waktu k dan berdimensi $l \times 1$
 v_k = *Noise* model pengukuran pada waktu k dan berdimensi $l \times 1$ dengan *mean* nol dan kovarian yang diketahui, yaitu R atau ditulis $E[v_k] = 0$ dan $Cov\{v_k, v_k\} = E[v_k v_k^T] = R$
 A = Matriks berdimensi $n \times n$
 B = Matriks berdimensi $n \times m$
 G = Matriks berdimensi $n \times m$
 H_k = Matriks berdimensi $p \times n$

w_k dan v_k diasumsikan tidak berkorelasi dan berdistribusi normal karena merupakan *white noise* yang ditulis secara umum sebagai berikut:

$$w_k \sim N(0, Q) \text{ dan } v_k \sim N(0, R)$$

(Prayudi, 2012).

Variabel $w_k \sim N(0, Q_k)$ dan $v_k \sim N(0, R_k)$ ini diasumsikan *white* (berdistribusi normal dengan *mean* 0), tidak berkorelasi satu sama lain begitu juga dengan nilai estimasi awal x_0 . Proses estimasi dalam Kalman Filter dilakukan dengan dua tahapan, yaitu tahap prediksi (*time update*) dan tahap koreksi (*measurement update*). Pada tahap prediksi, dilakukan proyeksi variabel keadaan dan kovariansi error estimasi pada waktu k untuk mendapatkan *a priori estimates* pada waktu $k + 1$. Pada tahap koreksi, pengukuran terbaru dimasukkan ke dalam *a priori estimates* untuk memperoleh *a posteriori estimates*. Tahap prediksi-koreksi ini dilakukan terus-menerus sampai pada k yang ditentukan (Prayudi, 2012).

Berikut ini adalah persamaan-persamaan Kalman Filter yang diperoleh dari proses estimasi:

1. Persamaan pada Tahap Prediksi

a. Estimasi

$$\hat{x}_{k+1}(-) = A\hat{x}_k + Bu_k \quad (2.16)$$

b. Kovariansi error

$$P_{x_{k+1}}(-) = AP_{x_k}A^T + GQG^T \quad (2.17)$$

2. Persamaan pada Tahap Koreksi

a. Kalman gain

$$K_{k+1} = P_{x_{k+1}}(-)H_{k+1}^T(R + H_{k+1}P_{x_{k+1}}(-)H_{k+1}^T)^{-1} \quad (2.18)$$

b. Kovariansi error

$$P_{x_{k+1}} = (I - K_{k+1}H_{k+1})P_{x_{k+1}}(-) \quad (2.19)$$

c. Estimasi

$$\hat{x}_{k+1} = \hat{x}_{k+1}(-) + K_{k+1}(z_{k+1} - H_{k+1}\hat{x}_{k+1}(-)) \quad (2.20)$$

Keterangan:

$\hat{x}_{k+1}(-)$: Estimasi sebelum pengukuran z_k dimasukkan pada waktu $k + 1$ atau *a priori estimates*.

$P_{x_{k+1}}(-)$: Kovariansi error estimasi sebelum pengukuran z_k dimasukkan pada waktu $k + 1$ atau *a priori estimates error covariance*.

\hat{x}_{k+1} : Estimasi setelah pengukuran z_k dimasukkan pada waktu $k + 1$ atau *a posteriori estimates*.

$P_{x_{k+1}}$: Kovariansi error estimasi setelah pengukuran z_k dimasukkan pada waktu $k + 1$ atau *a priori estimates error covariance*.

(Welch dan Bishop, 2006)

2.5 Metode *Extended Kalman Filter*

Extended Kalman Filter dapat digunakan untuk sistem yang tak linier dan juga kontinu. Sistem semacam ini perlu dilinierisasi (apabila sistem tidak linier), pendiskritan sistem (apabila sistem kontinu), dan beberapa tahapan lain. Algoritma *Kalman Filter* dikembangkan untuk nilai estimasi dalam bentuk rekursif dari model linier. Namun demikian, dalam kenyataannya banyak model

yang berbentuk non linear. Metode *Extended Kalman Filter* (EKF) adalah salah satu metode estimasi yang dikembangkan untuk menyelesaikan model non linier. Ketidaklinieran dari suatu sistem dinamika diperkirakan oleh sebuah versi linier dari model sistem non linier di sekitar taksiran keadaan terakhir. Linierisasi harus berupa sebuah taksiran yang utuh dari model non linier di semua domain tak tentu yang berhubungan dengan perhitungan keadaan agar taksirannya benar (Ribeiro, 2004).

Taksiran di sekitar estimasi dapat dilinierkan dengan menggunakan turunan-turunan parsial dari fungsi proses dan pengukuran dengan tujuan untuk menghitung estimasi. Persamaan (2.21) berikut adalah persamaan diferensial stokastik non linier yang mempunyai vektor keadaan $x \in R^m$.

$$x_k = f(x_{k-1}, u_k, w_{k-1}) \quad (2.21)$$

dengan sebuah pengukuran $z \in R^m$, yaitu

$$z_k = h(x_k, v_k) \quad (2.22)$$

Variabel-variabel acak w_k dan v_k menggambarkan *noise* sistem dan *noise* pengukuran. Fungsi non linier f pada persamaan (2.21) menghubungkan keadaan pada langkah sebelum $(k - 1)$ ke dalam keadaan sekarang k termasuk parameter-parameter vektor masukan deterministik u_{k-1} dan *noise* proses dengan *mean* nol w_k . Fungsi non linier h pada persamaan (2.22) menghubungkan keadaan x_k ke pengukuran z_k . Suatu keadaan dan vektor pengukuran saat diestimasi tanpa mensubstitusikan masing-masing nilai dari *noise* w_k dan v_k , seperti pada persamaan berikut:

$$\tilde{x}_k = f(\hat{x}_{k-1}, u_k, 0) \quad (2.23)$$

dan

$$\tilde{z}_k = h(\tilde{z}_k, 0) \quad (2.24)$$

\hat{x}_k adalah *a posteriori estimate* (dari langkah k sebelumnya).

$$x_k \approx \tilde{x}_k + A(x_{k-1} - \hat{x}_{k-1}) + W_{w_{k-1}} \quad (2.25)$$

$$z_k \approx \tilde{z}_k + H(x_k - \tilde{x}_k) + V_{v_k} \quad (2.26)$$

Persamaan (2.25) dan persamaan (2.26) adalah persamaan baru yang melinierkan persamaan (2.23) dan (2.24) untuk mengestimasi sebuah proses

dengan persamaan diferensial non linier dan hubungan–hubungan pengukuran dengan keterangan sebagai berikut:

x_k dan z_k = keadaan sebenarnya dan vektor pengukuran

\tilde{x}_k dan \tilde{z}_k = keadaan perkiraan dan vektor pengukuran dari persamaan (2.23) dan (2.24)

\hat{x}_k = suatu keadaan *a posteriori estimate* pada langkah k

w_k dan v_k = *noise* sistem dan *noise* pengukuran

A = matriks Jacobi dari turunan–turunan parsial f yang berhubungan dengan x , yaitu

$$A_{[i,j]} = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\hat{x}_{k-1}, u_k, 0),$$

W = matriks Jacobi dari turunan–turunan parsial h yang berhubungan dengan w , yaitu

$$W_{[i,j]} = \frac{\partial f_{[i]}}{\partial w_{[j]}}(\hat{x}_{k-1}, u_k, 0),$$

H = matriks Jacobi dari turunan–turunan parsial h yang berhubungan dengan x , yaitu

$$H_{[i,j]} = \frac{\partial h_{[i]}}{\partial x_{[j]}}(\tilde{x}_k, 0),$$

V = matriks Jacobi dari turunan–turunan parsial h yang berhubungan dengan v , yaitu

$$V_{[i,j]} = \frac{\partial h_{[i]}}{\partial v_{[j]}}(\tilde{x}_k, 0),$$

Persamaan-persamaan yang digunakan dalam metode *Extended Kalman Filter* adalah sebagai berikut :

Tabel 2.1 Persamaan – Persamaan *Extended Kalman Filter*

<p><u>Tahap Prediksi</u></p> $\hat{x}_k^- = f(\hat{x}_{k-1}, u_k, 0)$ $P_k^- = A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T$
<p><u>Tahap Koreksi</u></p> $K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1}$ $P_k = (I - K_k H_k) P_k^-$ $\hat{x}_k = \hat{x}_k^- + K_k (z_k - h(\hat{x}_k, 0))$

2.6 Norm Kovariansi Error

Norm kovariansi error merupakan tolak ukur dalam estimasi suatu model menggunakan Kalman filter untuk menyimpulkan baik tidaknya Kalman filter untuk estimasi sebuah model. Kovariansi error menyatakan kovarian perbedaan antara *true state vector* $x(t)$ dengan *estimated state vector* $\hat{x}(t)$ (Masduqi dan Apriliani, 2008).

Norm matriks pada himpunan S yang memuat matriks-matriks berukuran $n \times n$, yaitu dituliskan dengan notasi $\|\cdot\|$ atau biasanya disebut panjang/besar yang merupakan fungsi bernilai real yang terdefinisi pada S dan memenuhi:

- i) $\|\cdot\| \geq 0$,
- ii) $\|A\| = 0$ jika dan hanya jika $A = 0$,
- iii) $\|\alpha A\| = |\alpha|\|A\|$, untuk semua bilangan real α ,
- iv) $\|A + B\| \leq \|A\| + \|B\|$,
- v) $\|AB\| \leq \|A\|\|B\|$,

untuk semua $A, B \in S$ (Purnomo, 2008).

Nilai $\|P\|$ sangat mempengaruhi hasil estimasi *Kalman Filter*. Hubungan/korelasi error variabel ditunjukkan oleh besar kecilnya nilai $\|P\|$. Semakin besar nilai $\|P\|$ maka error yang dihasilkan akan semakin besar dan mempengaruhi estimasi yang dilakukan.

BAB 3. METODE PENELITIAN

Langkah-langkah yang akan dilakukan dalam menyelesaikan penelitian ini adalah pengumpulan data jumlah penduduk Kabupaten Jember tahun 1990 sampai dengan 2017 dilanjutkan perhitungan laju pertumbuhan penduduk dan ambang batas populasi. Langkah selanjutnya adalah diskritisasi model pada model pertumbuhan logistik. Model yang telah didiskritisasi kemudian dilakukan penambahan *noise* yang selanjutnya model akan diterapkan dalam algoritma *Extended Kalman Filter*. Langkah-langkah tersebut diuraikan lebih rinci dalam subbab berikut:

a. Tahap Pengumpulan Data

Jenis data yang digunakan dalam penelitian ini adalah data sekunder. Data yang dibutuhkan dalam penelitian ini antara lain, data jumlah penduduk Kabupaten Jember yang berupa deret berkala (*time series*) selama periode tahun 1990 sampai dengan tahun 2017 dan data luas daerah Kabupaten Jember. Data tersebut berasal dari buku atau laporan yang diterbitkan oleh Badan Pusat Statistik Provinsi Jawa Timur dan Badan Pusat Statistik Kabupaten Jember. Data luas daerah Kabupaten Jember akan digunakan sebagai acuan menentukan nilai ambang batas populasi (K).

b. Penentuan r dan K

Analisis data dilakukan untuk menghitung laju pertumbuhan (r) selama periode tahun 1990 sampai dengan tahun 2017 dan ambang batas populasi K . Laju pertumbuhan tersebut merupakan nilai rata-rata dari laju pertumbuhan setiap tahun. Laju pertumbuhan setiap tahun dihitung dengan rumus sebagai berikut:

$$r = \frac{N_{k+1} - N_k}{N_k} \quad (3.1)$$

Keterangan:

r : laju pertumbuhan penduduk setiap tahun

N_{k+1} : jumlah penduduk saat tahun $k + 1$

N_k : jumlah penduduk saat tahun k

Ambang batas populasi didapatkan dengan menggunakan luas daerah Kabupaten Jember. Menentukan nilai ambang batas populasi menggunakan luas daerah, didapatkan dengan cara sebagai berikut:

$$K = \frac{\text{Luas daerah Kabupaten Jember}}{\text{Luas tempat tinggal minimum tiap individu}} \quad (3.2)$$

Luas tempat tinggal dan tempat tinggal minimum untuk tiap individu diasumsikan seluas 100 m² berdasarkan acuan SNI-03-1733-2004.

c. Diskritisasi Model

Model pertumbuhan logistik dengan asumsi linier merupakan persamaan sistem kontinu, sehingga perlu dilakukan proses diskritisasi model terlebih dahulu. Diskritisasi model pertumbuhan logistik (2.3) menggunakan metode beda hingga maju (2.12) dengan panjang grid waktu Δt . Proses diskritisasi akan mendapatkan sistem diskrit dari jumlah populasi (N) terhadap waktu yang selanjutnya akan diimplementasikan dalam metode *Extended Kalman Filter*.

d. Penambahan *noise*

Penambahan *noise* dilakukan dengan membangkitkan sejumlah bilangan *random* dari komputer dengan asumsi konstan. *Noise* diasumsikan berdistribusi normal dengan mean 0 karena merupakan *white noise*. Variansi *noise* sistem dinyatakan dengan Q_k dan variansi *noise* pengukuran dinyatakan dengan R_k . Q_k dan R_k pada umumnya bergantung pada waktu, namun dalam hal ini variansi diasumsikan konstan. Variabel acak dapat dibangkitkan dengan perintah `normrnd(0, sqrt(sigQ), 2, 1)` untuk matriks Q_k , sedangkan untuk matriks R_k dibangkitkan dengan perintah `normrnd(0, sqrt(sigR) 2, 1)`.

e. Penerapan Metode *Extended Kalman Filter*

Sistem stokastik yang diperoleh akan diimplementasikan pada metode *Extended Kalman Filter* dengan menerapkan langkah-langkah sesuai Tabel 2.1. Variabel *state* yang akan diteliti adalah jumlah populasi (N). Input nilai awal untuk variabel *state* yang akan diestimasi berdasarkan data dari Badan Pusat Statistik Kabupaten Jember. Hasil yang diperoleh dari penggunaan

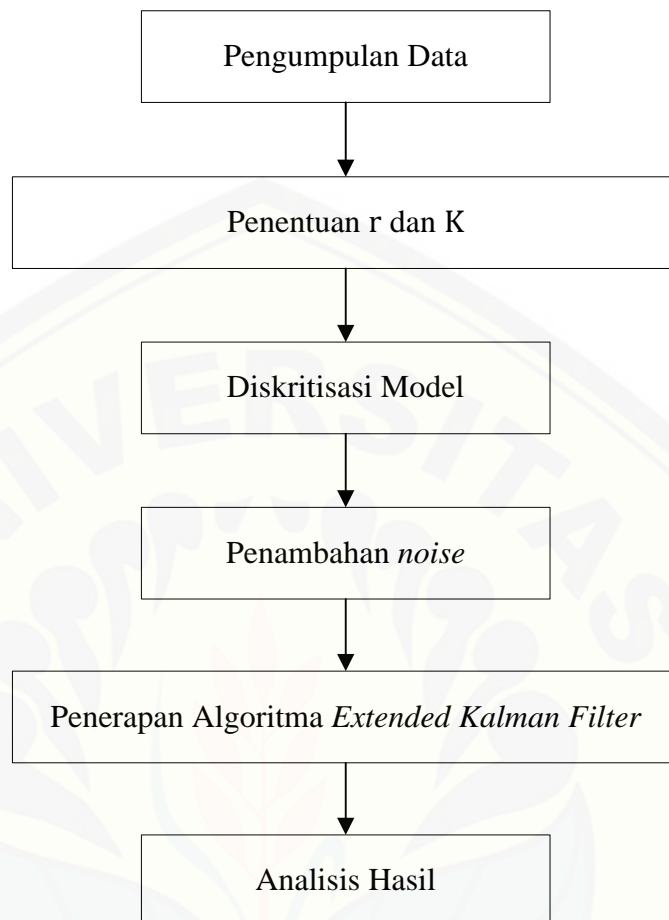
algoritma *Extended Kalman Filter* disimulasikan menggunakan program Matlab 2015.

f. Analisis Hasil

Hasil simulasi yang dianalisis adalah hasil estimasi jumlah penduduk Kabupaten Jember. Estimasi jumlah penduduk yang akan dibandingkan adalah:

1. Data asli Badan Pusat Statistik Kabupaten Jember.
2. Solusi analitik persamaan pertumbuhan logistik linier.
3. Solusi analitik persamaan pertumbuhan logistik parabolik terbuka ke bawah.
4. Hasil estimasi *Extended Kalman Filter* menggunakan persamaan pertumbuhan logistik linier dengan data pengukuran jumlah penduduk tahun 1990 sampai dengan tahun 2017.
5. Hasil estimasi *Extended Kalman Filter* menggunakan persamaan pertumbuhan logistik parabolik terbuka ke bawah dengan data pengukuran jumlah penduduk tahun 1990 sampai dengan tahun 2017.
6. Hasil estimasi *Extended Kalman Filter* menggunakan persamaan pertumbuhan logistik linier dengan data pengukuran jumlah penduduk tahun 1990 sampai dengan tahun 1995.
7. Hasil estimasi *Extended Kalman Filter* menggunakan persamaan pertumbuhan logistik parabolik terbuka ke bawah dengan data pengukuran jumlah penduduk tahun 1990 sampai dengan 1995.
8. Hasil estimasi *Extended Kalman Filter* menggunakan persamaan pertumbuhan logistik linier dengan data pengukuran jumlah penduduk tahun 1990 sampai dengan tahun 2003.
9. Hasil estimasi *Extended Kalman Filter* menggunakan persamaan pertumbuhan logistik parabolik terbuka ke bawah dengan data pengukuran jumlah penduduk tahun 1990 sampai dengan 2003.

Langkah-langkah tersebut secara skematik diilustrasikan pada Gambar 3.1



Gambar 3.1 Skema Metode Penelitian

BAB 5. PENUTUP

5.1 Kesimpulan

Kesimpulan yang diperoleh berdasarkan dari hasil analisis dan pembahasan adalah sebagai berikut:

1. Model pertumbuhan logistik dengan asumsi fungsi populasi parabolik lebih sesuai dengan pertumbuhan penduduk Kabupaten Jember selama periode tahun 1990-2017 karena absolut error model dengan asumsi fungsi populasi parabolik lebih kecil dibandingkan absolut error model dengan asumsi fungsi populasi linier dalam tiga hasil simulasi menggunakan 6 data pengukuran, 14 data pengukuran, dan 28 data pengukuran.
2. Hasil estimasi jumlah penduduk menggunakan metode *Extended Kalman Filter* memiliki tingkat kepercayaan yang cukup baik. Hal ini ditunjukkan dengan hasil norm kovariansi error yang memiliki rata-rata 0,022477 saat menggunakan 6 data pengukuran, 0,010042 saat menggunakan 14 data pengukuran, 0,0010406 saat menggunakan 28 data pengukuran. Semakin banyak data pengukuran yang digunakan, hasil estimasi yang dihasilkan semakin baik dan mendekati data sebenarnya.

5.2 Saran

Pada skripsi ini, estimasi jumlah penduduk Kabupaten Jember menggunakan metode *Extended Kalman Filter*, untuk penelitian selanjutnya estimasi dapat dilakukan menggunakan metode lain untuk mengetahui metode yang dapat memberikan hasil estimasi yang lebih baik.

DAFTAR PUSTAKA

- Badan Pusat Statistik. 2017. *Jember Dalam Angka*. Jember: BPS Kabupaten Jember.
- Badan Pusat Statistik. 2018. *Jawa Timur Dalam Angka*. Surabaya: BPS Provinsi Jawa Timur.
- Firdhania, R dan F. Muslihatinningsih. 2017. Faktor-faktor yang Mempengaruhi Tingkat Pengangguran di Kabupaten Jember. *e-Journal Ekonomi Bisnis dan Akuntansi*. 4(1):117-121.
- Fitriani, V.N dan K.D.Purnomo. 2013. Solution Estimation of Logistic Growth Model with Ensemble Kalman Filter Method. *Jurnal Ilmu Dasar*. 14(2):85-90.
- Ichwan, A. 2010. Estimasi Posisi dan Kecepatan Kapal Selam Menggunakan Metode Extended Kalman Filter. *Skripsi*. Surabaya : Fakultas Matematika dan Ilmu Pengetahuan Alam Institut Teknologi Sepuluh Nopember.
- Masduqi, A., dan Apriliani, E. 2008. Estimation of Surabaya River Water Quality Using Kalman Filter Algorithm. *The Journal for Technology and Science*. 19(3):87-91.
- Prayudi, V. 2012. Penerapan Extended Kalman Filter pada Model Predator-Prey. *Skripsi*. Jember : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.
- Purnomo, K.D. dan E. Apriliani. 2008. Estimasi Populasi Plankton dengan Ensemble Kalman Filter. *Jurnal Ilmu Dasar*. 9(1):38-44.
- Ribeiro, M.I. 2004. *Kalman and Extended Kalman Filters: Concept, Derivation, and Properties*. Institute for Systems and Robotics Instituto Superior Tecnico Lisboa. [Online] <http://users.isr.ist.utl.pt/~mir/pub/kalman.pdf> [26 Februari 2019].
- Iswanto, R.J . 2012. *Pemodelan Matematika, Aplikasi dan Terapannya*. Yogyakarta: Graha Ilmu.
- Rochaida, E. 2016. Dampak Pertumbuhan Penduduk Terhadap Pertumbuhan Ekonomi dan Keluarga Sejahtera di Provinsi Kalimantan Timur. *Forum Ekonomi*. 18(1).

- Rohaeni, O. 2017. Model Pertumbuhan Populasi Satu Spesies dengan Tundaan Waktu Diskrit. *Jurnal Teori dan Terapan Matematika* 16(1):17.
- Sukirno, S. 2000. *Pengantar Teori Mikroekonomi*. Jakarta : Raja Grafindo Persada.
- Susanto, E. 2008. *Minimum Varians untuk Sistem Multi Input Multi Output (Mimo)*. Bandung: Departemen Teknik Elektro Institut Teknologi Bandung.
- Triatmodjo, B. 1992. *Metode Numerik*. Yogyakarta: Jurusan Teknik Sipil Fakultas Teknik Universitas Gadjah Mada.
- Wicaksono, R. D. 2010. Penerapan Extended Kalman Filter untuk Mendeteksi Waktu Terjadinya Kerak pada Alat Penukar Panas. *Skripsi*. Surabaya : Institut Teknologi Sepuluh Nopember.
- Welch, G dan Bishop, G. 2006. *An Introduction to The Kalman Filter*, Chapel Hil: Department of Computer Science University of North Carolina.
- Zill, D. G. 2005. *A first Course in Differential Equations With Modelling Applications*. Eight Editions. Canada: Brooks/Cole, a division of Thomson Learning Inc.

LAMPIRAN

Lampiran A Hasil Estimasi Menggunakan Metode EKF

A.1 Hasil Estimasi dengan 6 Data Pengukuran

Tahun	EKF1	EKF2	Analitik1	Analitik2	BPS
1990	2.062.843	2.062.843	2.062.843	2.062.843	2.062.843
1991	2.074.246	2.074.960	2.074.275	2.074.995	2.036.792
1992	2.085.706	2.087.146	2.085.766	2.087.218	2.041.578
1993	2.097.228	2.099.406	2.097.317	2.099.513	2.043.255
1994	2.108.807	2.111.736	2.108.927	2.111.880	2.045.525
1995	2.120.447	2.124.137	2.120.597	2.124.318	2.049.422
1996	2.132.146	2.136.611	2.132.327	2.136.830	2.075.103
1997	2.143.905	2.149.158	2.144.117	2.149.414	2.078.701
1998	2.155.725	2.161.777	2.155.968	2.162.072	2.083.068
1999	2.167.605	2.174.470	2.167.880	2.174.804	2.106.632
2000	2.179.546	2.187.237	2.179.853	2.187.610	2.105.132
2001	2.191.548	2.200.079	2.191.888	2.200.491	2.120.074
2002	2.203.612	2.212.995	2.203.984	2.213.447	2.187.930
2003	2.215.737	2.225.986	2.216.142	2.226.478	2.231.793
2004	2.227.924	2.239.052	2.228.362	2.239.586	2.248.968
2005	2.240.173	2.252.195	2.240.645	2.252.770	2.263.794
2006	2.252.485	2.265.414	2.252.991	2.266.031	2.278.718
2007	2.264.859	2.278.710	2.265.399	2.279.369	2.293.740
2008	2.277.297	2.292.083	2.277.871	2.292.785	2.320.844
2009	2.289.797	2.305.534	2.290.406	2.306.280	2.327.857
2010	2.302.361	2.319.063	2.303.006	2.319.852	2.337.909
2011	2.314.989	2.332.671	2.315.669	2.333.504	2.353.025
2012	2.327.681	2.346.358	2.328.396	2.347.236	2.367.482
2013	2.340.437	2.360.124	2.341.188	2.361.047	2.381.400
2014	2.353.258	2.373.970	2.354.046	2.374.939	2.394.608
2015	2.366.144	2.387.897	2.366.968	2.388.912	2.407.115
2016	2.379.094	2.401.904	2.379.955	2.402.966	2.419.000
2017	2.392.110	2.415.993	2.393.009	2.417.101	2.430.185

A.2 Absolut Error Hasil Estimasi dengan 6 data pengukuran

Tahun	EKF1-BPS	EKF2-BPS	Analitik1-BPS	Analitik2-BPS
1990	0	0	0	0
1991	37.454	38.168	37.483	38.203
1992	44.128	45.568	44.188	45.640
1993	53.973	56.151	54.062	56.258
1994	63.282	66.211	63.402	66.355
1995	71.025	74.715	71.175	74.896
1996	57.043	61.508	57.224	61.727
1997	65.204	70.457	65.416	70.713
1998	72.657	78.709	72.900	79.004
1999	60.973	67.838	61.248	68.172
2000	74.414	82.105	74.721	82.478
2001	71.474	80.005	71.814	80.417
2002	15.682	25.065	16.054	25.517
2003	16.056	5.807	15.651	5.315
2004	21.044	9.916	20.606	9.382
2005	23.621	11.599	23.149	11.024
2006	26.233	13.304	25.727	12.687
2007	28.881	15.030	28.341	14.371
2008	43.547	28.761	42.973	28.059
2009	38.060	22.323	37.451	21.577
2010	35.548	18.846	34.903	18.057
2011	38.036	20.354	37.356	19.521
2012	39.801	21.124	39.086	20.246
2013	40.963	21.276	40.212	20.353
2014	41.350	20.638	40.562	19.669
2015	40.971	19.218	40.147	18.203
2016	39.906	17.096	39.045	16.034
2017	38.075	14.192	37.176	13.084
Rata-rata	42.836	35.928	42.574	35.606

A.3 Norm Kovariansi Error dengan 6 Data Pengukuran

Estimasi Absolut Error Norm Kovariansi Error

	Iterasi	Norm Kov Error
1	0	1.0000e-04
2	1	0.0010
3	2	1.0000e-03
4	3	0.0010
5	4	0.0010
6	5	0.0030
7	6	0.0050
8	7	0.0071
9	8	0.0092
10	9	0.0113
11	10	0.0134
12	11	0.0155
13	12	0.0177
14	13	0.0199
15	14	0.0221
16	15	0.0244
17	16	0.0266
18	17	0.0289
19	18	0.0312
20	19	0.0336
21	20	0.0360
22	21	0.0383
23	22	0.0408
24	23	0.0432
25	24	0.0457
26	25	0.0482
27	26	0.0507
28	27	0.0533

Rata-rata Norm Kov. Error = 0.022477

A.4 Hasil Estimasi dengan 14 data pengukuran

Tahun	EKF1	EKF2	Analitik1	Analitik2	BPS
1990	2.062.843	2.062.843	2.062.843	2.062.843	2.062.843
1991	2.074.246	2.074.960	2.074.275	2.074.995	2.036.792
1992	2.085.706	2.087.146	2.085.766	2.087.218	2.041.578
1993	2.097.228	2.099.406	2.097.317	2.099.513	2.043.255
1994	2.108.808	2.111.736	2.108.927	2.111.880	2.045.525
1995	2.120.447	2.124.138	2.120.597	2.124.318	2.049.422
1996	2.132.145	2.136.611	2.132.327	2.136.830	2.075.103
1997	2.143.905	2.149.157	2.144.117	2.149.414	2.078.701
1998	2.155.724	2.161.776	2.155.968	2.162.072	2.083.068
1999	2.167.603	2.174.469	2.167.880	2.174.804	2.106.632
2000	2.179.544	2.187.236	2.179.853	2.187.610	2.105.132
2001	2.191.547	2.200.077	2.191.888	2.200.491	2.120.074
2002	2.203.611	2.212.994	2.203.984	2.213.447	2.187.930
2003	2.215.736	2.225.985	2.216.142	2.226.478	2.231.793
2004	2.227.923	2.239.052	2.228.362	2.239.586	2.248.968
2005	2.240.173	2.252.195	2.240.645	2.252.770	2.263.794
2006	2.252.484	2.265.414	2.252.991	2.266.031	2.278.718
2007	2.264.859	2.278.710	2.265.399	2.279.369	2.293.740
2008	2.277.296	2.292.083	2.277.871	2.292.785	2.320.844
2009	2.289.797	2.305.534	2.290.406	2.306.280	2.327.857
2010	2.302.361	2.319.063	2.303.006	2.319.852	2.337.909
2011	2.314.989	2.332.671	2.315.669	2.333.504	2.353.025
2012	2.327.681	2.346.357	2.328.396	2.347.236	2.367.482
2013	2.340.437	2.360.124	2.341.188	2.361.047	2.381.400
2014	2.353.258	2.373.970	2.354.046	2.374.939	2.394.608
2015	2.366.143	2.387.896	2.366.968	2.388.912	2.407.115
2016	2.379.094	2.401.904	2.379.955	2.402.966	2.419.000
2017	2.392.110	2.415.992	2.393.009	2.417.101	2.430.185

A.5 Absolut Error Hasil Estimasi dengan 14 data pengukuran

Tahun	EKF1-BPS	EKF2-BPS	Analitik1-BPS	Analitik2-BPS
1990	0	0	0	0
1991	37.454	38.168	37.483	38.203
1992	44.128	45.568	44.188	45.640
1993	53.973	56.151	54.062	56.258
1994	63.283	66.211	63.402	66.355
1995	71.025	74.716	71.175	74.896
1996	57.042	61.508	57.224	61.727
1997	65.204	70.456	65.416	70.713
1998	72.656	78.708	72.900	79.004
1999	60.971	67.837	61.248	68.172
2000	74.412	82.104	74.721	82.478
2001	71.473	80.003	71.814	80.417
2002	15.681	25.064	16.054	25.517
2003	16.057	5.808	15.651	5.315
2004	21.045	9.916	20.606	9.382
2005	23.621	11.599	23.149	11.024
2006	26.234	13.304	25.727	12.687
2007	28.881	15.030	28.341	14.371
2008	43.548	28.761	42.973	28.059
2009	38.060	22.323	37.451	21.577
2010	35.548	18.846	34.903	18.057
2011	38.036	20.354	37.356	19.521
2012	39.801	21.125	39.086	20.246
2013	40.963	21.276	40.212	20.353
2014	41.350	20.638	40.562	19.669
2015	40.972	19.219	40.147	18.203
2016	39.906	17.096	39.045	16.034
2017	38.075	14.193	37.176	13.084
Rata-rata	42.836	35.928	42.574	35.606

A.6 Norm Kovariansi Error Hasil Estimasi dengan 14 data pengukuran

Estimasi Absolut Error Norm Kovariansi Error

	Iterasi	Norm Kov Error
1	0	1.0000e-04
2	1	0.0010
3	2	1.0000e-03
4	3	1.0000e-03
5	4	1.0000e-03
6	5	0.0010
7	6	1.0000e-03
8	7	0.0010
9	8	0.0010
10	9	1.0000e-03
11	10	1.0000e-03
12	11	1.0000e-03
13	12	1.0000e-03
14	13	0.0030
15	14	0.0050
16	15	0.0071
17	16	0.0092
18	17	0.0113
19	18	0.0134
20	19	0.0155
21	20	0.0177
22	21	0.0199
23	22	0.0221
24	23	0.0244
25	24	0.0266
26	25	0.0289
27	26	0.0312
28	27	0.0336

Rata-rata Norm Kov. Error = 0.010042

A.7 Hasil Estimasi dengan 28 data pengukuran

Tahun	EKF1	EKF2	Analitik1	Analitik2	BPS
1990	2.062.843	2.062.843	2.062.843	2.062.843	2.062.843
1991	2.074.246	2.074.960	2.074.275	2.074.995	2.036.792
1992	2.085.705	2.087.145	2.085.766	2.087.218	2.041.578
1993	2.097.224	2.099.402	2.097.317	2.099.513	2.043.255
1994	2.108.804	2.111.733	2.108.927	2.111.880	2.045.525
1995	2.120.443	2.124.134	2.120.597	2.124.318	2.049.422
1996	2.132.140	2.136.605	2.132.327	2.136.830	2.075.103
1997	2.143.900	2.149.153	2.144.117	2.149.414	2.078.701
1998	2.155.720	2.161.772	2.155.968	2.162.072	2.083.068
1999	2.167.600	2.174.466	2.167.880	2.174.804	2.106.632
2000	2.179.544	2.187.235	2.179.853	2.187.610	2.105.132
2001	2.191.545	2.200.075	2.191.888	2.200.491	2.120.074
2002	2.203.609	2.212.992	2.203.984	2.213.447	2.187.930
2003	2.215.733	2.225.982	2.216.142	2.226.478	2.231.793
2004	2.227.921	2.239.049	2.228.362	2.239.586	2.248.968
2005	2.240.171	2.252.193	2.240.645	2.252.770	2.263.794
2006	2.252.483	2.265.412	2.252.991	2.266.031	2.278.718
2007	2.264.859	2.278.709	2.265.399	2.279.369	2.293.740
2008	2.277.297	2.292.084	2.277.871	2.292.785	2.320.844
2009	2.289.798	2.305.535	2.290.406	2.306.280	2.327.857
2010	2.302.361	2.319.063	2.303.006	2.319.852	2.337.909
2011	2.314.990	2.332.671	2.315.669	2.333.504	2.353.025
2012	2.327.682	2.346.359	2.328.396	2.347.236	2.367.482
2013	2.340.438	2.360.124	2.341.188	2.361.047	2.381.400
2014	2.353.259	2.373.971	2.354.046	2.374.939	2.394.608
2015	2.366.144	2.387.897	2.366.968	2.388.912	2.407.115
2016	2.379.095	2.401.904	2.379.955	2.402.966	2.419.000
2017	2.392.111	2.415.993	2.393.009	2.417.101	2.430.185

A.8 Absolut Error Hasil Estimasi dengan 28 data pengukuran

Tahun	EKF1-BPS	EKF2-BPS	Analitik1-BPS	Analitik2-BPS
1990	0	0	0	0
1991	37.454	38.168	37.483	38.203
1992	44.127	45.567	44.188	45.640
1993	53.969	56.147	54.062	56.258
1994	63.279	66.208	63.402	66.355
1995	71.021	74.712	71.175	74.896
1996	57.037	61.502	57.224	61.727
1997	65.199	70.452	65.416	70.713
1998	72.652	78.704	72.900	79.004
1999	60.968	67.834	61.248	68.172
2000	74.412	82.103	74.721	82.478
2001	71.471	80.001	71.814	80.417
2002	15.679	25.062	16.054	25.517
2003	16.060	5.811	15.651	5.315
2004	21.047	9.919	20.606	9.382
2005	23.623	11.601	23.149	11.024
2006	26.235	13.306	25.727	12.687
2007	28.881	15.031	28.341	14.371
2008	43.547	28.760	42.973	28.059
2009	38.059	22.322	37.451	21.577
2010	35.548	18.846	34.903	18.057
2011	38.035	20.354	37.356	19.521
2012	39.800	21.123	39.086	20.246
2013	40.962	21.276	40.212	20.353
2014	41.349	20.637	40.562	19.669
2015	40.971	19.218	40.147	18.203
2016	39.905	17.096	39.045	16.034
2017	38.074	14.192	37.176	13.084
Rata-rata	42.834	35.927	42.574	35.606

A.9 Norm Kovariansi Error Hasil Estimasi dengan 28 data pengukuran

Estimasi Absolut Error Norm Kovariansi Error

	Iterasi	Norm Kov Error
1	0	1.0000e-04
2	1	0.0010
3	2	0.0010
4	3	1.0000e-03
5	4	1.0000e-03
6	5	1.0000e-03
7	6	0.0010
8	7	1.0000e-03
9	8	0.0010
10	9	1.0000e-03
11	10	1.0000e-03
12	11	1.0000e-03
13	12	0.0010
14	13	0.0010
15	14	0.0010
16	15	0.0010
17	16	0.0010
18	17	0.0010
19	18	1.0000e-03
20	19	1.0000e-03
21	20	0.0010
22	21	1.0000e-03
23	22	1.0000e-03
24	23	1.0000e-03
25	24	0.0010
26	25	0.0010
27	26	1.0000e-03
28	27	0.0030

Rata-rata Norm Kov. Error = 0.0010406

Lampiran B Skrip Program Metode EKF

```

function varargout = ProgramBaru(varargin)
% PROGRAMBARU MATLAB code for ProgramBaru.fig
%   PROGRAMBARU, by itself, creates a new PROGRAMBARU or raises
the existing
%   singleton*.
%
%   H = PROGRAMBARU returns the handle to a new PROGRAMBARU or
the handle to
%   the existing singleton*.
%
%   PROGRAMBARU('CALLBACK',hObject,eventData,handles,...) calls
the local
%   function named CALLBACK in PROGRAMBARU.M with the given
input arguments.
%
%   PROGRAMBARU('Property','Value',...) creates a new
PROGRAMBARU or raises the
%   existing singleton*. Starting from the left, property
value pairs are
%   applied to the GUI before ProgramBaru_OpeningFcn gets
called. An
%   unrecognized property name or invalid value makes property
application
%   stop. All inputs are passed to ProgramBaru_OpeningFcn via
varargin.
%
%   *See GUI Options on GUIDE's Tools menu. Choose "GUI allows
only one
%   instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help ProgramBaru

% Last Modified by GUIDE v2.5 02-Jul-2019 07:53:02

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @ProgramBaru_OpeningFcn, ...
                  'gui_OutputFcn',  @ProgramBaru_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

```

```

% --- Executes just before ProgramBaru is made visible.
function ProgramBaru_OpeningFcn(hObject, eventdata, handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to ProgramBaru (see VARARGIN)

% Choose default command line output for ProgramBaru
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
clc;
set(gcf, 'Name', 'Model Pertumbuhan Penduduk')
movegui(gcf, 'center');
set(handles.uitable1, 'data', [], 'rowname', 1:10);
set(handles.edit1, 'string', '');
set(handles.edit2, 'string', '');
set(handles.edit3, 'string', '');
set(handles.edit4, 'string', '');
cla(handles.axes1, 'reset');
axes(handles.axes1);
xlabel('Tahun'); ylabel('Jumlah');
set(handles.axes1, 'FontSize', 8, 'Fontweight', 'bold');
cla(handles.axes2, 'reset');
axes(handles.axes2);
xlabel('Tahun'); ylabel('Absolut Error');
set(handles.axes2, 'FontSize', 8, 'Fontweight', 'bold');
cla(handles.axes3, 'reset');
axes(handles.axes3);
xlabel('Iterasi'); ylabel('Norm Kov. Error');
set(handles.axes3, 'FontSize', 8, 'Fontweight', 'bold');
set(handles.radiobutton1, 'value', 1);
set(handles.radiobutton2, 'value', 0);
set(handles.radiobutton3, 'value', 0);
set(handles.uitable2, 'data', [], 'rowname', 1:10, 'visible', 'on', 'colu
mnname', {'Tahun', 'EKF 1', 'EKF 2', 'Analitik 1', 'Analitik 2'});
set(handles.uitable3, 'data', [], 'rowname', 1:10, 'visible', 'off', 'col
umnname', {'Tahun', 'EKF - An 1', 'EKF - An 2'});
set(handles.uitable4, 'data', [], 'rowname', 1:10, 'visible', 'off', 'col
umnname', {'Iterasi', 'Norm Kov Error'});
set(handles.text8, 'string', 'Rata-rata Norm Kov. Error =');
% UIWAIT makes ProgramBaru wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = ProgramBaru_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);

```



```
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in radiobutton1.
function radiobutton1_Callback(hObject, eventdata, handles)
% hObject    handle to radiobutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
set(handles.radiobutton1, 'value', 1);
set(handles.radiobutton2, 'value', 0);
set(handles.radiobutton3, 'value', 0);
set(handles.uitable2, 'visible', 'on');
set(handles.uitable3, 'visible', 'off');
set(handles.uitable4, 'visible', 'off');
% Hint: get(hObject, 'Value') returns toggle state of radiobutton1

% --- Executes on button press in radiobutton2.
function radiobutton2_Callback(hObject, eventdata, handles)
% hObject    handle to radiobutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
set(handles.radiobutton1, 'value', 0);
set(handles.radiobutton2, 'value', 1);
set(handles.radiobutton3, 'value', 0);
set(handles.uitable2, 'visible', 'off');
set(handles.uitable3, 'visible', 'on');
set(handles.uitable4, 'visible', 'off');
% Hint: get(hObject, 'Value') returns toggle state of radiobutton2

% --- Executes on button press in radiobutton3.
function radiobutton3_Callback(hObject, eventdata, handles)
% hObject    handle to radiobutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
set(handles.radiobutton1, 'value', 0);
set(handles.radiobutton2, 'value', 0);
set(handles.radiobutton3, 'value', 1);
set(handles.uitable2, 'visible', 'off');
set(handles.uitable3, 'visible', 'off');
set(handles.uitable4, 'visible', 'on');
% Hint: get(hObject, 'Value') returns toggle state of radiobutton3

% --- Executes on button press in pushbutton1.
```

```

function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
[File, Path]=uigetfile({'*.xls;*.xlsx','Excel Files
(*.xls;*.xlsx)'},...
'Open File Data');
if File~=0
    set(handles.uitable1,'data',[],'Userdata',[],'rowname',1:30);
%reset tabel
    [data001,txt] = xlsread(fullfile(Path,File)); %baca data excel
    %menampilkan data

set(handles.uitable1,'data',data001,'Userdata',data001,'rowname','
numbered');
end

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
cla(handles.axes1,'reset');
axes(handles.axes1);
xlabel('Tahun'); ylabel('Jumlah');
set(handles.axes1,'FontSize',8,'Fontweight','bold');
cla(handles.axes2,'reset');
axes(handles.axes2);
xlabel('Tahun'); ylabel('Error');
set(handles.axes2,'FontSize',8,'Fontweight','bold');
cla(handles.axes3,'reset');
axes(handles.axes3);
xlabel('Iterasi'); ylabel('Norm Kov. Error');
set(handles.axes3,'FontSize',8,'Fontweight','bold');
set(handles.radiobutton1,'value',1);
set(handles.radiobutton2,'value',0);
set(handles.radiobutton3,'value',0);
set(handles.uitable2,'data',[],'rowname',1:10,'visible','on');
set(handles.uitable3,'data',[],'rowname',1:10,'visible','off');
set(handles.uitable4,'data',[],'rowname',1:10,'visible','off');
set(handles.text8,'string','Rata-rata Norm Kov. Error =');

%Data
Data=get(handles.uitable1,'UserData');
N0=Data(1,2);
TData=size(Data,1)-1;
dt=1;
Tmax=str2num(get(handles.edit5,'string'));
T=0:dt:Tmax;
Tahun=Data(1,1):1:Data(1,1)+Tmax;

%Parameter

```

```

r=str2num(get(handles.edit1,'string'));
K=str2num(get(handles.edit2,'string'));
sigQ=str2num(get(handles.edit3,'string'));
sigR=str2num(get(handles.edit4,'string'));

%Analitik
C1=(K/N0-1)/K;
Mana=K./(1+exp(-r*T)*C1*K);
C2=(K^2/N0^2-1)/(2*r);
Nana=sqrt(K^2./(1+2*r*exp(-2*r*T)*C2));

%Nilai awal EKF
XEKF=[N0;N0];
H2=eye(2);
Pk2=10^(-4)*eye(2);
Npk2(1)=norm(Pk2);

%Iterasi
for t=1:Tmax
    %Prediksi -----
    -----
    W2=normrnd(0,sigQ,2,1);
    M2=(r-r/K*XEKF(1,t))*XEKF(1,t)*dt+XEKF(1,t); %Model 1
    N2=(r-r/K^2*XEKF(2,t)^2)*XEKF(2,t)*dt+XEKF(2,t); %Model 2
    XEKF(:,t+1)=[M2;N2]+W2;
    MatA=[(r-2*r/K*XEKF(1,t))*dt+1 0;0 (r-
3*r/K^2*XEKF(2,t)^2)*dt+1];
    Pk2=MatA*Pk2*MatA'+sigQ;

    %Koreksi -----
    -----
    if t<TData
        V2=normrnd(0,sigR,2,1);
        Z2=H2*(repmat(Data(t+1,2),2,1))+V2;
        Kk2=Pk2*H2'*(H2*Pk2*H2'+sigR)^-1;
        XEKF(:,t+1)=XEKF(:,t+1)+Kk2*(Z2-
H2*(repmat(Data(t+1,2),2,1)));
        Pk2=(eye(2)-Kk2*H2)*Pk2;
    end

    %Norm Kovariansi Error -----
    -----
    Npk2(t+1)=norm(Pk2);

    %Error -----
    -----
    Error2a=abs(XEKF(1,:)-Mana(1:t+1));
    Error2b=abs(XEKF(2,:)-Nana(1:t+1));

    %Plot
    axes(handles.axes1);
    plot(Tahun(1:t+1),XEKF(1,:), 'g-', Tahun(1:t+1),XEKF(2,:), 'g-
.', Tahun(1:t+1),Mana(1:t+1), 'b-', ...
        Tahun(1:t+1),Nana(1:t+1), 'b-', Data(:,1), Data(:,2), 'k-');
    xlabel('Tahun'); ylabel('Jumlah');

```

```

    legend('EKF linier','EKF parabolik','Analitik
linier','Analitik parabolik','Data Real BPS');
    set(handles.axes1,'FontSize',8,'Fontweight','bold');
    axes(handles.axes2);
    plot(Tahun(1:t+1),Error2a,'r-',Tahun(1:t+1),Error2b,'b-.');
    xlabel('Tahun'); ylabel('Absolut Error');
    legend('EKF-analitik(linier)','EKF-analitik(parabolik)','EKF-
real(linier)','EKF-real(parabolik)','Analitik-
real(linier)','Analitik-real(parabolik)');
    set(handles.axes2,'FontSize',8,'Fontweight','bold');
    axes(handles.axes3);
    plot(0:t,Npk2,'b');
    xlabel('Iterasi'); ylabel('Norm Kov. Error');
    set(handles.axes3,'FontSize',8,'Fontweight','bold');

set(handles.uitable2,'data',round([Tahun(1:t+1)',XEKF(1,:)','XEKF(2
,:)','Mana(1:t+1)','Nana(1:t+1)']),'rowname','numbered',...
    'columnname',{'Tahun','EKF 1','EKF 2','Analitik
1','Analitik 2'});

set(handles.uitable3,'data',[Tahun(1:t+1)',Error2a',Error2b'],...
    'columnname',{'Tahun','EKF - An 1','EKF - An 2'},...
    'rowname','numbered');

set(handles.uitable4,'data',[(0:t)',Npk2'],'rowname','numbered','c
olumnname',{'Iterasi','Norm Kov Error'});
    pause(0.01);
end

set(handles.text8,'string',['Rata-rata Norm Kov. Error = '
num2str(mean(Npk2))]);

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
movegui(gcf,'center');
set(handles.uitable1,'data',[],'rowname',1:10);
set(handles.edit1,'string','');
set(handles.edit2,'string','');
set(handles.edit3,'string','');
set(handles.edit4,'string','');
set(handles.edit5,'string','');
cla(handles.axes1,'reset');
axes(handles.axes1);
xlabel('Tahun'); ylabel('Jumlah');
set(handles.axes1,'FontSize',8,'Fontweight','bold');
cla(handles.axes2,'reset');
axes(handles.axes2);
xlabel('Tahun'); ylabel('Error');
set(handles.axes2,'FontSize',8,'Fontweight','bold');
cla(handles.axes3,'reset');
axes(handles.axes3);

```

```
xlabel('Iterasi'); ylabel('Norm Kov. Error');
set(handles.axes3,'FontSize',8,'Fontweight','bold');
set(handles.radiobutton1,'value',1);
set(handles.radiobutton2,'value',0);
set(handles.radiobutton3,'value',0);
set(handles.uitable2,'data',[],'rowname',1:10,'visible','on');
set(handles.uitable3,'data',[],'rowname',1:10,'visible','off');
set(handles.uitable4,'data',[],'rowname',1:10,'visible','off');
set(handles.text8,'string','Rata-rata Norm Kov. Error');

% --- Executes on button press in checkbox1.
function checkbox1_Callback(hObject, eventdata, handles)
% hObject     handle to checkbox1 (see GCBO)
% eventdata   reserved - to be defined in a future version of
MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of checkbox1

% --- Executes on button press in checkbox2.
function checkbox2_Callback(hObject, eventdata, handles)
% hObject     handle to checkbox2 (see GCBO)
% eventdata   reserved - to be defined in a future version of
MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of checkbox2

% --- Executes on button press in checkbox3.
function checkbox3_Callback(hObject, eventdata, handles)
% hObject     handle to checkbox3 (see GCBO)
% eventdata   reserved - to be defined in a future version of
MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of checkbox3

% --- Executes on button press in checkbox4.
function checkbox4_Callback(hObject, eventdata, handles)
% hObject     handle to checkbox4 (see GCBO)
% eventdata   reserved - to be defined in a future version of
MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of checkbox4

function edit1_Callback(hObject, eventdata, handles)
```

```
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
%         str2double(get(hObject,'String')) returns contents of
edit1 as a double

% --- Executes during object creation, after setting all
properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit2_Callback(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit2 as text
%         str2double(get(hObject,'String')) returns contents of
edit2 as a double

% --- Executes during object creation, after setting all
properties.
function edit2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function edit3_Callback(hObject, eventdata, handles)
% hObject      handle to edit3 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit3 as text
%         str2double(get(hObject,'String')) returns contents of
edit3 as a double

% --- Executes during object creation, after setting all
properties.
function edit3_CreateFcn(hObject, eventdata, handles)
% hObject      handle to edit3 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit4_Callback(hObject, eventdata, handles)
% hObject      handle to edit4 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit4 as text
%         str2double(get(hObject,'String')) returns contents of
edit4 as a double

% --- Executes during object creation, after setting all
properties.
function edit4_CreateFcn(hObject, eventdata, handles)
% hObject      handle to edit4 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit5_Callback(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit5 as text
%        str2double(get(hObject,'String')) returns contents of
edit5 as a double

% --- Executes during object creation, after setting all
properties.
function edit5_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```