

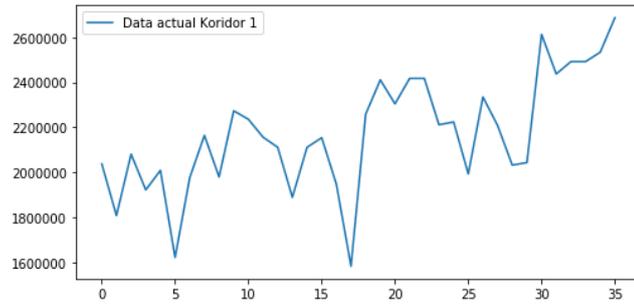
BAB III

METODOLOGI PENELITIAN

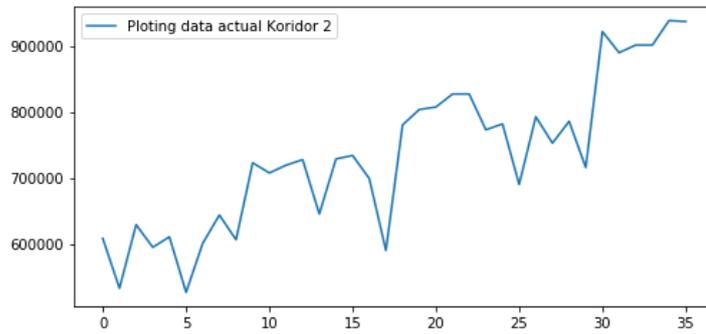
3.1 Analisis Permasalahan

Berdasarkan permasalahan yang telah dijabarkan pada subab latar belakang, jumlah penumpang bus Transjakarta setiap tahunnya mengalami peningkatan yang signifikan. Kemudian, intensitas jumlah penumpang yang tinggi diwaktu jam kerja menyebabkan penumpang harus rela berdesak-desakan di halte dan didalam bus tersebut. Penyebab utama terjadinya masalah tersebut yaitu jumlah armada bus dan kapasitas angkut yang tidak sesuai dengan jumlah penumpangnya sehingga terjadi penumpukan penumpang [2]. Sesuai dengan permasalahan tersebut, maka dibutuhkanlah prediksi jumlah penumpang bus Transjakarta diwaktu yang akan datang untuk memberikan solusi atas permasalahan tersebut.

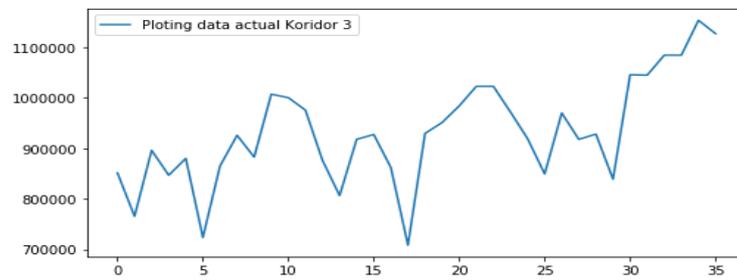
Metode yang digunakan dalam melakukan prediksi jumlah penumpang bus Transjakarta ini yaitu menggunakan metode *triple exponential smoothing*. Digunakannya metode *triple exponential smoothing* ini dikarenakan pola data *time series* data jumlah penumpang bus Transjakarta yaitu bersifat musiman. Menurut Makridakis [17], metode yang cocok untuk melakukan prediksi atau peramalan dengan pola *time series* data musiman adalah menggunakan metode *triple exponential smoothing*. Pola data musiman merupakan pola data yang mengalami fluktuatif yang sangat tajam perbedaannya dalam periode tertentu [19]. Berikut ini merupakan *ploting* data jumlah penumpang bus Transjakarta untuk Koridor 1, Koridor 2, Koridor 3, Koridor 4, Koridor 5, Koridor 7, Koridor 8, Koridor 9, Koridor 10, Koridor 11 dan Koridor 12 periode rentang waktu bulan Januari 2017 sampai bulan Desember 2019.



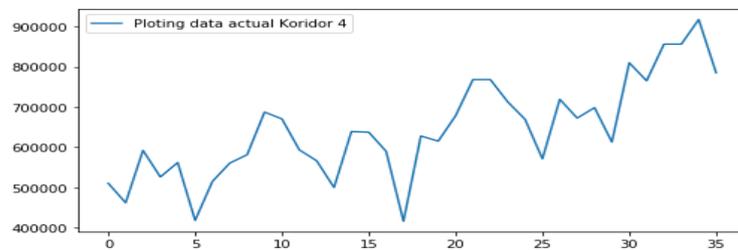
Gambar 3.1 Ploting dataset pada koridor 1



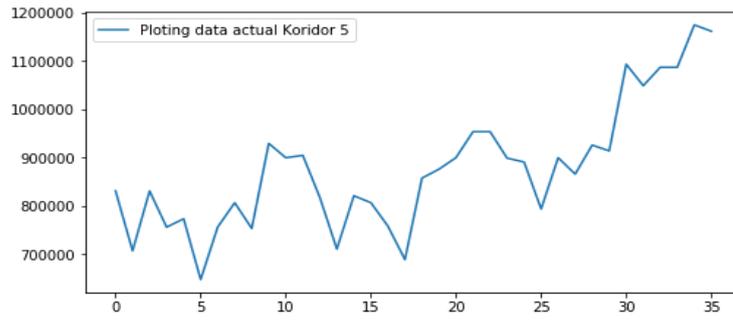
Gambar 3.2 Ploting dataset pada koridor 2



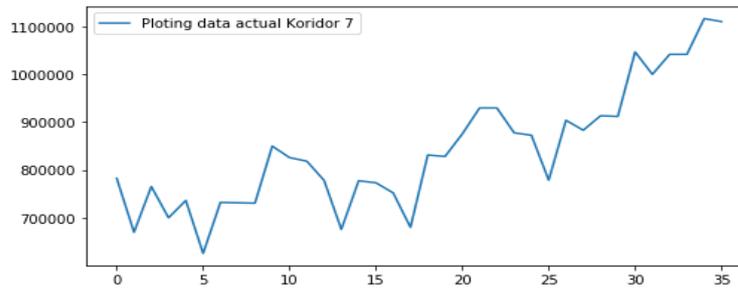
Gambar 3.3 Ploting dataset pada koridor 3



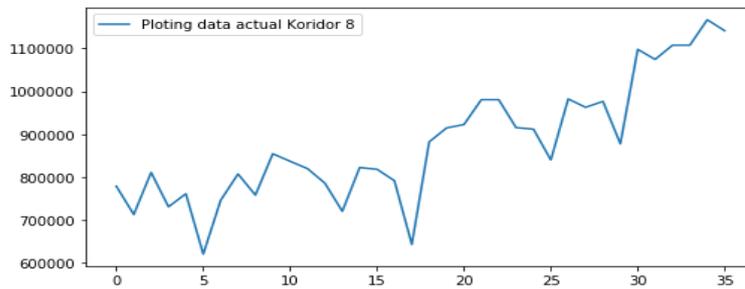
Gambar 3.4 Ploting dataset pada koridor 4



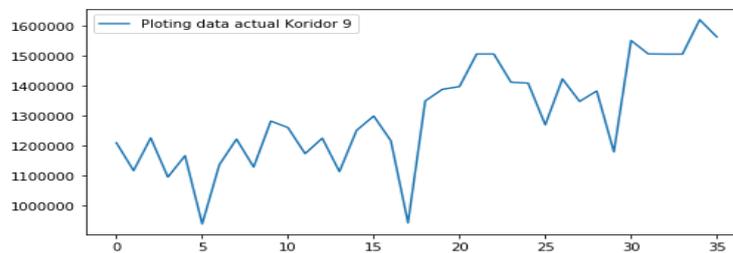
Gambar 3.5 *Ploting* dataset pada koridor 5



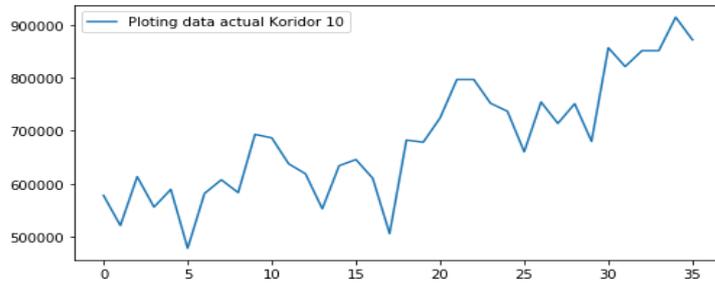
Gambar 3.6 *Ploting* dataset pada koridor 7



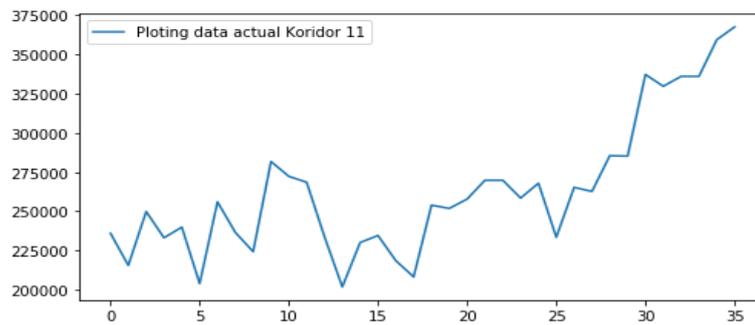
Gambar 3.7 *Ploting* dataset pada koridor 8



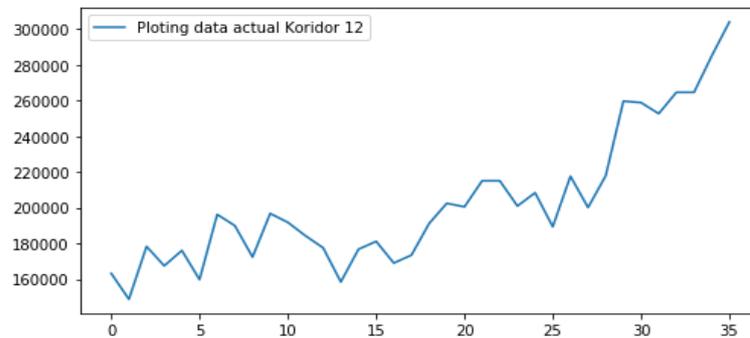
Gambar 3.8 *Ploting* dataset pada koridor 9



Gambar 3.9 *Ploting* dataset pada koridor 10



Gambar 3.10 *Ploting* dataset pada koridor 11



Gambar 3.11 *Ploting* dataset pada koridor 12

Dalam melakukan penelitian ini menggunakan data *time series* jumlah penumpang bus Transjakarta dari 11 koridor utama setiap bulannya dengan periode rentang waktu bulan Januari 2017 sampai bulan Desember 2019. Pada tabel 3.1 berikut ini merupakan contoh sampel dataset jumlah penumpang Transjakarta bulan Januari 2017.

Tabel 3.1 Sampel dataset penumpang bus Transjakarta
(Sumber : <https://data.jakarta.go.id>)

Jenis Layanan	Trayek	Jumlah Penumpang
Layanan Sistem BRT	KORIDOR 1	2036785
Layanan Sistem BRT	KORIDOR 2	607830
Layanan Sistem BRT	KORIDOR 3	851314
Layanan Sistem BRT	KORIDOR 4	510794
Layanan Sistem BRT	KORIDOR 5	831301
Layanan Sistem BRT	KORIDOR 7	782386
Layanan Sistem BRT	KORIDOR 8	778860
Layanan Sistem BRT	KORIDOR 9	1209820

Sesuai dengan tujuan penelitian ini yaitu menghasilkan model prediksi terbaik untuk memprediksi jumlah penumpang bus Transjakarta, maka dataset yang didapatkan tersebut akan dibagi menjadi 2 subset yaitu data *training* dan data *testing*. Proses *training* menggunakan metode *triple exponential smoothing* dengan data *training* untuk mendapatkan model terbaik. Selanjutnya setelah mendapatkan model terbaik dari proses *training*, yaitu dilakukan proses *testing* untuk memprediksi jumlah penumpang bus Transjakarta periode tahun 2019. Output dari penelitian ini yaitu berupa jumlah penumpang bus Transjakarta tiap trayek dengan periode rentang waktu perbulan untuk tahun 2019 dengan parameter evaluasi ketepatan prediksi menggunakan nilai MAPE terkecil. Tabel 3.2 berikut merupakan contoh output penelitian.

Tabel 3.2 Sampel output penelitian

Periode	Koridor 1
Jan-19	2036898
Feb-19	1890299
Mar-19	2070351
Apr-19	1721310

3.2 Kebutuhan Perangkat

3.2.1 Perangkat Keras

Tabel 3.3 merupakan spesifikasi perangkat keras yang digunakan pada penelitian ini.

Tabel 3.3 Spesifikasi perangkat keras

No	Perangkat Keras	Spesifikasi
1	Laptop	ASUS VivoBook Max X441B
2	<i>Processor</i>	AMD A6
3	RAM	4 GB
4	<i>Storage</i>	HDD 1 TB
5	<i>Grapics</i>	Radeon R5 Grapics

3.2.2 Perangkat Lunak

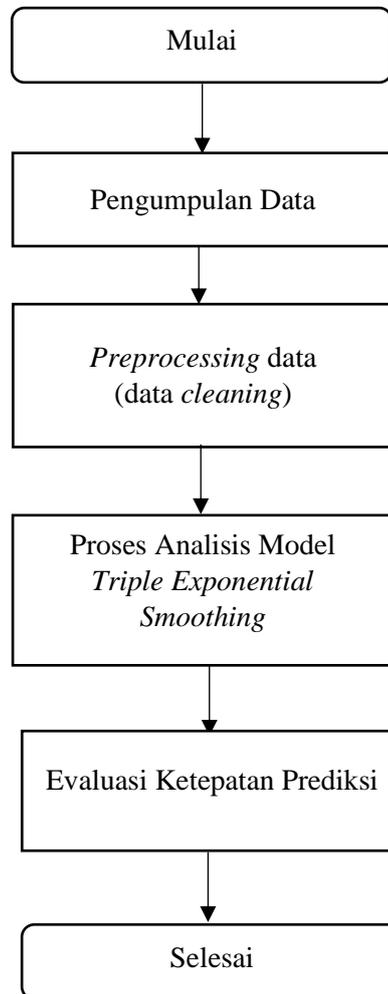
Tabel 3.4 merupakan spesifikasi perangkat lunak yang digunakan pada penelitian ini.

Tabel 3.4 Spesifikasi perangkat lunak

No	Perangkat Lunak	Spesifikasi
1	Sistem operasi	Windows 10
2	Bahasa pemrograman	Python 3.7
3	<i>Integrated Developent Environment (IDE)</i>	Jupyter notebook (Anaconda 3)
4	<i>Framework</i>	Numpy v1.17.3, Pandas v0.25.2, Matplotlib v3.1.1, Scipy

3.3 Tahapan Penelitian

Pada sub bab ini, akan membahas langkah-langkah tahap penelitian yang akan dilakukan berdasarkan tujuan pada penelitian. Gambar 3.12 merupakan diagram alir yang digunakan untuk mencari model *triple exponential smoothing* terbaik.



Gambar 3.12 Diagram alir penelitian

3.3.1 Pengumpulan Data

Penelitian ini membutuhkan data historis jumlah penumpang bus Transjakarta untuk dapat digunakan sebagai bahan dalam melaksanakan penelitian tugas akhir. Oleh sebab itu, dalam tahap ini dilakukan pencarian data yang akan diolah pada tugas akhir ini. Data yang digunakan untuk penelitian ini didapatkan dari website resmi Portal Data Terpadu Pemerintah Provinsi DKI Jakarta yaitu <https://data.jakarta.go.id> yang menyajikan data-data dari seluruh Satuan dan Unit Kerja di Pemerintah Provinsi DKI Jakarta. Data yang digunakan oleh penulis adalah

data jumlah penumpang bus Transjakarta dengan rentang waktu mulai dari bulan Januari 2017 hingga bulan Desember 2019. Data yang didapatkan dari website tersebut berupa data historis jumlah penumpang bus Transjakarta berdasarkan trayek dalam periode rentang waktu perbulan.

Tabel 3.5 Deskripsi dataset penumpang bus Transjakarta

Jumlah data	432
Rata-rata	867458.331
Nilai terendah	148744
Nilai tertinggi	2686458

Pada Tabel 3.5 terdapat jumlah data penumpang bus Transjakarta sebanyak 432, dengan nilai rata-rata jumlah penumpang 867458. Kemudian untuk jumlah penumpang dengan nilai tertinggi yaitu sebanyak 2.686.458 penumpang pada trayek Koridor 1 (Blok M – Kota) pada saat bulan Desember 2019. Sedangkan untuk jumlah penumpang dengan nilai terendah yaitu sebanyak 148744 penumpang.

3.3.2 Preprocessing Data

Setelah data historis jumlah penumpang bus Transjakarta didapatkan, langkah selanjutnya dalam melakukan penelitian ini adalah *preprocessing* data. Pada tahap *preprocessing* data ini digunakan untuk pengolahan data historis jumlah penumpang bus Transjakarta yang telah didapatkan pada tahap sebelumnya. Pada tahap *preprocessing* data dilakukan proses data *cleaning* untuk mendapatkan dataset yang siap digunakan untuk tahap proses analisis model *triple exponential smoothing*. Pada proses data *cleaning* tersebut dilakukan untuk membersihkan data historis jumlah penumpang bus Transjakarta dengan beberapa teknik seperti mengoreksi data yang tidak konsisten dan mengatasi *missing value*. Pada penelitian ini, data historis jumlah penumpang bus Transjakarta yang didapatkan tersebut masih memiliki nilai yang tidak konsisten. Kemudian data historis tersebut dilakukan proses pencarian kolom jumlah penumpang yang hilang. Selanjutnya setelah ditemukan kolom jumlah penumpang yang hilang tersebut dilakukan proses

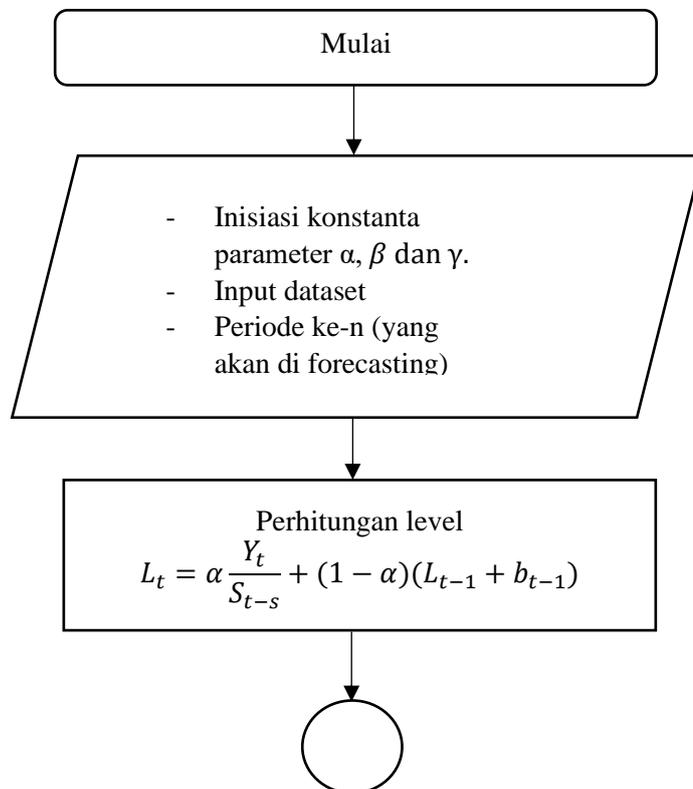
interpolasi dengan cara pengisian data jumlah penumpang dengan menggunakan persamaan (II.12).

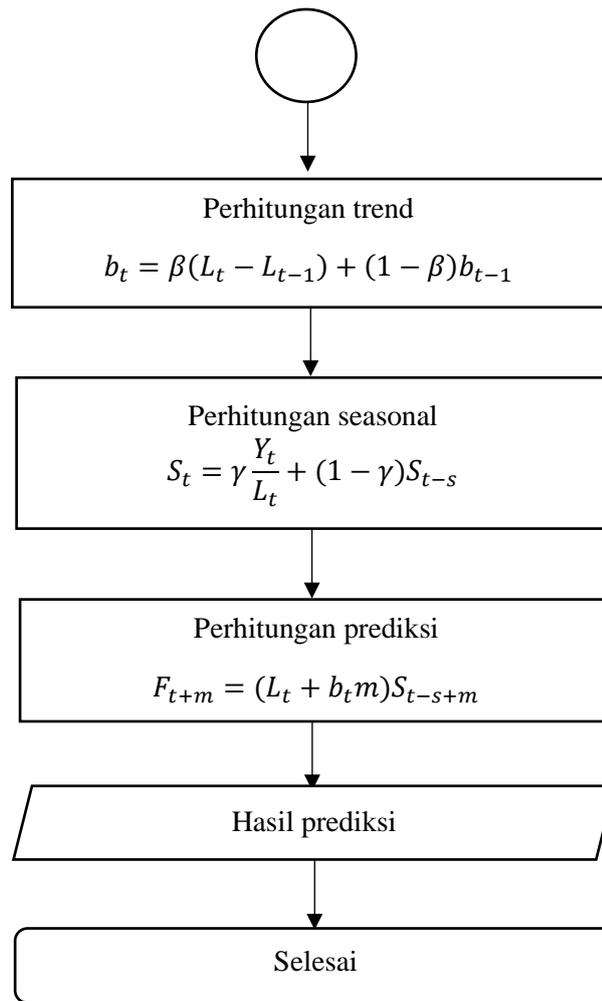
3.3.3 Proses Analisis Model Triple Exponential Smoothing

Dataset yang telah melewati tahap *preprocessing* data data selanjutnya yaitu dilakukan pembentukan model pada metode *triple exponential smoothing* untuk melakukan prediksi data jumlah penumpang bus Transjakarta yaitu dengan cara membagi dataset menjadi 2 subset yaitu data *training* dan data *testing*.

Untuk pembuatan model *triple exponential smoothing*, digunakan parameter konstanta pemulusan. Parameter konstanta tersebut yaitu terdiri dari parameter α untuk pemulusan keseluruhan/total, parameter β untuk pemulusan trend, dan parameter γ untuk pemulusan musiman [17]. Nilai dari setiap parameter tersebut yaitu antara 0 dan 1 [17]. Dalam mendapatkan model tersebut, dilakukan dengan cara mengkombinasikan ketiga parameter tersebut secara trial dan error (uji coba).

Pada gambar 3.13 berikut alur proses perhitungan pada metode *triple exponential smoothing* :





Gambar 3.13 Proses perhitungan metode *triple exponential smoothing*

Berikut ini merupakan langkah kerja dalam memprediksi menggunakan metode *triple exponential smoothing* :

1. Menentukan nilai konstanta α , β dan γ yang bernilai antara 0 dan 1.
2. Menghitung nilai awal pemulusan level dengan persamaan (II.9) menggunakan dataset satu periode (L) musiman lengkap.
3. Menghitung nilai awal pemulusan trend dengan persamaan (II.10) menggunakan dataset dua periode (2L) musiman.
4. Menghitung nilai awal pemulusan musiman dengan persamaan (II.11).
5. Menghitung nilai pemulusan level menggunakan persamaan (II.5).
6. Menghitung nilai pemulusan trend menggunakan persamaan (II.6).

7. Menghitung nilai pemulusan musiman dengan menggunakan persamaan (II.7).
8. Menghitung nilai prediksi untuk m periode dengan menggunakan persamaan (II.8).

3.3.4 Evaluasi Ketepatan Hasil Prediksi

Evaluasi ketetapan hasil prediksi dilakukan setelah mendapatkan model *exponential smoothing* terbaik pada saat tahap *training* data dan telah melakukan prediksi dengan menggunakan data historis jumlah penumpang bus Transjakarta pada tahap *testing* data. Hal tersebut perlu dilakukan untuk mengetahui tingkat akurasi model dari metode *triple exponential smoothing* yang telah didapatkan dalam melakukan prediksi tersebut. Untuk menghitung ketetapan model prediksi tersebut dapat dilakukan dengan mengukur nilai MAPE (*Mean Absolute Percentage Error*) berdasarkan persamaan (II.14) dengan cara menghitung perbandingan antara data aktual dan hasil prediksi. Digunakannya MAPE dalam mengukur akurasi ketepatan model karena dapat menyatakan persentase kesalahan hasil prediksi terhadap permintaan aktual selama periode tertentu dengan memberikan informasi persentase kesalahan prediksi.