



**TUGAS AKHIR - IS 184853**

**PERAMALAN JUMLAH KASUS DEMAM BERDARAH  
DENGUE DI KABUPATEN MALANG DENGAN METODE K-  
NEAREST NEIGHBOR REGRESSION**

***FORECASTING THE NUMBER OF DENGUE CASES IN  
MALANG REGENCY USING K-NEAREST NEIGHBOR  
REGRESSION***

**BERTA KARTIKA CHANDRA  
NRP 05211640000103**

**Dosen Pembimbing :  
Edwin Riksakomara, S.Kom, M.T  
Raras Tyasnurita, S.Kom, MBA**

**DEPARTEMEN SISTEM INFORMASI  
Fakultas Teknologi Elektro dan Informatika Cerdas  
Institut Teknologi Sepuluh Nopember  
Surabaya 2020**





**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

TUGAS AKHIR - IS 184853

# **PERAMALAN JUMLAH KASUS DEMAM BERDARAH DENGUE DI KABUPATEN MALANG DENGAN METODE K-NEAREST NEIGHBOR REGRESSION**

BERTA KARTIKA CHANDRA  
NRP 05211640000103

**Dosen Pembimbing :**  
Edwin Riksakomara, S.Kom, M.T  
Raras Tyasnurita, S.Kom, MBA

DEPARTEMEN SISTEM INFORMASI  
Fakultas Teknologi Elektro dan Informatika Cerdas  
Institut Teknologi Sepuluh Nopember  
Surabaya 2020



**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

FINAL PROJECT - IS 184853

# ***FORECASTING THE NUMBER OF DENGUE CASES IN MALANG REGENCY USING K-NEAREST NEIGHBOR REGRESSION***

BERTA KARTIKA CHANDRA  
NRP 05211640000103

## **SUPERVISOR :**

Edwin Riksakomara, S.Kom, M.T  
Raras Tyasnurita, S.Kom, MBA

DEPARTMENT OF INFORMATION SYSTEMS  
Faculty of Intelligent Electrical and Informatics Technology  
Institut Teknologi Sepuluh Nopember  
Surabaya 2020

**LEMBAR PENGESAHAN****PERAMALAN JUMLAH KASUS DEMAM BERDARAH  
DENGUE DI KABUPATEN MALANG DENGAN METODE  
K-NEAREST NEIGHBOR REGRESSION****TUGAS AKHIR**

Disusun Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer (S.Kom)  
pada

Departemen Sistem Informasi  
Fakultas Teknologi Elektro dan Informatika Cerdas (ELECTICS)  
Institut Teknologi Sepuluh Nopember

Oleh

**Berta Kartika Chandra**

**05211640000103**

Surabaya, 14 Agustus 2020

**Kepala Departemen Sistem Informasi**

**Dr. Mudjahidin, ST., MT.  
NIP. 197010102003121001**





**LEMBAR PERSETUJUAN**

**PERAMALAN JUMLAH KASUS DEMAM BERDARAH  
DENGUE DI KABUPATEN MALANG DENGAN METODE  
K-NEAREST NEIGHBOR REGRESSION**

**Tugas Akhir**

Disusun Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada

Departemen Sistem Informasi  
Fakultas Teknologi Elektro dan Informatika Cerdas  
Institut Teknologi Sepuluh Nopember

oleh:

**BERTA KARTIKA CHANDRA**

**NRP. 05211640000103**

Disetujui Tim Penguji : Tanggal Ujian : 2 Juli 2020  
Periode Wisuda : September 2020

**Edwin Riksakomara, S.Kom., M.T.**

  
(Pembimbing 1)

**Raras Tyasnurita, S.Kom., M.BA., Ph.D**

  
(Pembimbing 2)

**Ahmad Muklason, S.Kom., M.Sc., Ph.D.**

  
(Penguji 1)

**Faizal Mahananto S.Kom., M.Eng., Ph.D**

  
(Penguji 2)

# **PERAMALAN JUMLAH KASUS DEMAM BERDARAH DENGUE DI KABUPATEN MALANG DENGAN METODE K-NEAREST NEIGHBOR REGRESSION**

**Nama Mahasiswa : Berta Kartika Chandra**  
**NRP : 05211640000**  
**Jurusan : Sistem Informasi FTEIC-ITS**  
**Pembimbing 1 : Edwin Riksakomara, S.Kom, M.T**  
**Pembimbing 2 : Raras Tyasnurita, S.Kom, MBA**

## **ABSTRAK**

Kasus demam berdarah selalu muncul dan memakan korban jiwa setiap tahunnya. Menurut data Direktorat Jenderal Pencegahan dan Pengendalian Penyakit, Kementerian Kesehatan (Kemenkes) RI, terhitung sejak minggu pertama tahun 2018 hingga minggu pertama tahun 2019 telah tercatat 700 jumlah kasus demam berdarah di Jawa Timur. Angka itu berhasil membuat Jawa Timur menduduki peringkat satu kasus demam berdarah terbanyak di Indonesia. Tidak hanya itu, Jawa Timur juga menjadi wilayah dengan angka kematian akibat DBD terbanyak pada tahun 2018 yaitu sebanyak 47 korban. Wilayah dengan angka kasus demam berdarah yang tinggi di Jawa Timur pada tahun 2018 adalah Kabupaten Malang. Mewabahnya demam berdarah di suatu wilayah bisa dipengaruhi oleh kondisi geografis, lingkungan maupun kepadatan penduduk. Angka kasus demam berdarah ini bisa ditekan dengan upaya mitigasi yang tepat.

Informasi prediksi atau peramalan bisa digunakan untuk memberi bantuan dalam pengambilan keputusan untuk upaya mitigasi yang tepat di waktu yang tepat. Agar hasil peramalan bisa menjadi sebuah pendukung keputusan, maka hasil peramalan harus memiliki tingkat akurasi yang tinggi. Untuk itu, teknik peramalan menjadi sangat penting untuk mendapatkan tingkat akurasi yang

tinggi. Banyak teknik atau metode yang bisa digunakan, salah satunya adalah K-Nearest Neighbor Regression (KNNR) yang merupakan salah satu teknik peramalan dengan *machine learning*. Tugas akhir ini disusun untuk mengimplementasikan metode KNNR untuk menghasilkan model peramalan untuk kasus demam berdarah di Kabupaten Malang. Data yang digunakan adalah data kasus demam berdarah di Kabupaten Malang yang berasal dari Dinas Kesehatan Kabupaten Malang dan data cuaca dari dua stasiun BMKG di Kabupaten Malang. Model peramalan yang dibentuk melibatkan variabel suhu, curah hujan, kelembaban udara, kecepatan angin, angka bebas jentik dan jumlah penduduk. Model peramalan terbaik dipilih di setiap dataran rendah, dataran sedang dan dataran tinggi.

Model terbaik dengan metode KNNR memiliki nilai *Root Mean Square Error* 2.133 di dataran rendah, 2.205 di dataran sedang dan 1.543 di dataran tinggi. Model terbaik memiliki nilai *Symmetric Mean Absolute Error* sebesar 24.977% di dataran rendah, 15.944% di dataran sedang dan 23.136% di dataran tinggi.

***Kata kunci: demam berdarah, kabupaten malang, k-nearest neighbor, peramalan, Root Mean Square Error, Symmetric Mean Absolute Percentage***



**PERAMALAN JUMLAH KASUS DEMAM BERDARAH  
DENGUE DI KABUPATEN MALANG DENGAN METODE  
K-NEAREST NEIGHBOR REGRESSION**

**Nama Mahasiswa : Berta Kartika Chandra**  
**NRP : 0521164000103**  
**Jurusan : Sistem Informasi FTEIC-ITS**  
**Pembimbing 1 : Edwin Riksakomara, S.Kom, M.T**  
**Pembimbing 2 : Raras Tyasnurita, S.Kom, MBA**

**ABSTRACT**

Dengue fever cases always arise and take lives every year. According to data from the Directorate General of Disease Prevention and Control, the Indonesian Ministry of Health, starting from the first week of 2018 to the first week of 2019 there were 700 recorded cases of dengue fever in East Java. That number succeeded in making East Java ranked as the number one case of dengue in Indonesia. Not only that, East Java also became the region with the highest number of deaths due to DHF in 2018, totaling 47 victims. The region with a high number of dengue cases in East Java in 2018 is Malang Regency. The spread of dengue in an area can be influenced by geographical conditions, the environment and population density. The number of dengue cases can be reduced with appropriate mitigation efforts.

Prediction or forecasting can be used to provide assistance in making decisions for the right mitigation efforts at the right time. Forecasting results can be a decision support when have a high degree of accuracy. For this reason, forecasting techniques are very important to get a high level of accuracy. Many techniques or methods that can be used, one of which is K-Nearest Neighbor Regression (KNNR) which is one of the forecasting techniques with machine learning.

This final project is structured to implement the KNNR method to produce a forecast model for dengue cases in Malang Regency. The data used are data on dengue fever cases in Malang Regency from the Malang District Health Office and weather data from two BMKG stations in Malang Regency. Forecasting models that are formed involve variables of temperature, rainfall, humidity, wind velocity, larval free numbers and population. The best forecasting model is chosen in every lowland, medium and high altitude.

The best model with the KNNR method has a value of Root Mean Square Error 2,133 in the lowlands, 2,205 in the medium plains and 1,543 in the highlands. The best model has a Symmetric Mean Absolute Error value of 24,977% in the lowlands, 15,944% in the moderate plains and 23,136% in the highlands.

***Keywords: dengue fever, k-nearest neighbor regression, forecasting, Root Mean Square Error, Symmetric Mean Absolute Percentage***

## SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertandatangan di bawah ini:

Nama : Berta Kartika Chandra

NRP : 05211640000103

Tempat/Tanggal lahir : Kediri / 23 September 1997

Fakultas/Departemen : FTEIC / Sistem Informasi

Nomor Telp/HP/Email : 085649026764/chandraberta@gmail.com

Dengan ini menyatakan dengan sesungguhnya bahwa penelitian / makalah / tugas akhir saya yang berjudul

**PERAMALAN JUMLAH KASUS DEMAM BERDARAH DENGUE DI KABUPATEN MALANG DENGAN METODE K-NEAREST NEIGHBOR REGRESSION**

**Bebas Dari Plagiarisme Dan Bukan Hasil Karya Orang Lain.**

Apabila dikemudian hari ditemukan seluruh atau sebagian penelitian / makalah / tugas akhir tersebut terdapat indikasi plagiarisme, maka saya bersedia menerima sanksi sesuai peraturan dan ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sesungguhnya dan untuk dipergunakan sebagaimana mestinya

Kediri, 10 Agustus 2020



Berta Kartika Chandra  
NRP.05211640000103

## **KATA PENGANTAR**

Puji dan syukur penulis tuturkan ke hadirat Allah SWT, Tuhan Semesta Alam yang telah memberikan karunia dan hidayah-Nya kepada penulis sehingga penulis mendapatkan kelancaran dalam menyelesaikan tugas akhir dengan judul:

### **PERAMALAN JUMLAH KASUS DEMAM BERDARAH DENGUE DI KABUPATEN MALANG DENGAN METODE K-NEAREST NEIGHBOR REGRESSION**

Terima kasih penulis sampaikan kepada pihak-pihak yang telah mendukung, memberikan saran, motivasi, semangat, dan bantuan baik berupa material maupun moril demi tercapainya tujuan pembuatan tugas akhir ini. Tugas akhir ini tidak akan pernah terwujud tanpa bantuan dan dukungan dari berbagai pihak yang sudah meluangkan waktu, tenaga dan pikirannya. Secara khusus penulis akan menyampaikan ucapan terima kasih yang sebanyak-banyaknya kepada :

1. Bapak Mochammad Tauchid dan Ibu Tumiarsih selaku kedua orang tua serta Dewi Mita Rozali selaku saudara kandung dari penulis yang tiada henti memberikan dukungan dan semangat.
2. Bapak Edwin Riksakomara, S.Kom., MT. selaku dosen pembimbing 1 dan ibu Raras Tyasnurita, S.Kom, MBA selaku dosen pembimbing 2, serta ibu Wiwik Anggraeni, S.Si., M.Kom, yang senantiasa meluangkan waktu, memberikan ilmu dan petunjuk, serta memotivasi untuk kelancaran tugas akhir.
3. Ahmad Muklason, S.Kom, M.Sc, Ph.D dan Faisal Mahananto, S.Kom, M.Eng, Ph.D, selaku dosen penguji yang telah memberikan saran dan kritik yang sangat membangun untuk perbaikan tugas akhir.
4. Seluruh dosen Jurusan Sistem Informasi ITS yang telah memberikan ilmu yang bermanfaat kepada penulis.

5. Sahabat-sahabat yaitu Firin, Mira, Meli yang terus mendukung dan memberi inspirasi.
6. Geng avalon yaitu Detika, Syafira, Ifan, Abdil, Candra, Isa, Hilmy, Aulia yang terus menyemangati dan menjadi teman berdiskusi permasalahan.

Penyusunan laporan ini masih jauh dari kata sempurna sehingga penulis menerima adanya kritik maupun saran yang membangun untuk perbaikan di masa yang akan datang. Semoga buku tugas akhir ini dapat memberikan manfaat bagi pembaca.

## DAFTAR ISI

ABSTRAK .....	v
ABSTRACT .....	vii
KATA PENGANTAR.....	x
DAFTAR ISI .....	xii
DAFTAR GAMBAR.....	xvii
DAFTAR TABEL .....	xx
DAFTAR KODE .....	xxiv
BAB I PENDAHULUAN .....	25
1.1. Latar Belakang.....	25
1.2. Rumusan Permasalahan.....	26
1.3. Batasan Permasalahan .....	27
1.4. Tujuan.....	27
1.5. Manfaat.....	28
1.6. Relevansi .....	28
BAB II TINJAUAN PUSTAKA.....	30
2.1. Studi Sebelumnya.....	30
2.2. Dasar Teori .....	36
2.2.1. Demam Berdarah.....	36
2.2.2. Forecasting.....	38
2.2.3. Uji korelasi .....	39
2.2.4. Min-max normalization .....	39
2.2.5. Mengatasi <i>Missing Value</i> .....	40
2.2.6. Windowing .....	42
2.2.7. K-Nearest Neighbor Regression.....	44
2.2.8. Uji kelayakan.....	50
BAB III METODOLOGI PENELITIAN .....	52
3.1. Tahapan Pelaksanaan Tugas Akhir.....	52
3.2. Uraian Metodologi.....	53
3.2.1. Studi literatur .....	53
3.2.2. Pengumpulan data dan Informasi .....	53
3.2.3. Pemilihan variabel (feature selection) .....	54
3.2.4. Pra-proses data.....	54

3.2.5.	Pembangunan model peramalan dengan Metode <i>k-Nearest Neighbor Regression</i> .....	54
3.2.6.	Pengujian model .....	55
3.2.7.	Implementasi model peramalan.....	55
3.2.8.	Penyusunan Laporan Tugas Akhir .....	56
BAB IV PERANCANGAN .....		57
4.1.	Pengumpulan data .....	57
4.2.	Penentuan dataset kecamatan pembentuk model.....	57
4.3.	Praproses Data.....	58
4.3.1.	Menyelaraskan periode pada data.....	58
4.3.2.	Mengatasi nilai yang hilang ( <i>handling missing value</i> )...	59
4.3.3.	Normalisasi.....	60
4.3.4.	Windowing .....	61
4.4.	Penentuan Parameter KNNR, Independent Variable dan lag .....	61
4.4.1.	Menentukan parameter dan nilai parameter .....	62
4.4.2.	Uji korelasi setiap variabel observasi dengan <i>dependent variable</i> .....	63
4.4.3.	Menentukan <i>lag</i> .....	63
4.4.4.	Pembentukan skenario <i>tuning parameter</i> .....	64
4.5.	Pembentukan skenario <i>independent variable</i> .....	64
4.5.1.	Hanya melibatkan variabel <i>lag</i> .....	65
4.5.2.	Melibatkan satu variabel observasi dengan nilai korelasi tertinggi .....	65
4.5.3.	Melibatkan dua variabel observasi dengan nilai korelasi tertinggi .....	66
4.5.4.	Melibatkan tiga variabel observasi dengan nilai korelasi tertinggi .....	66
4.5.5.	Melibatkan semua variabel observasi.....	66
4.6.	Pembagian <i>training data</i> dan <i>testing data</i> .....	67
4.7.	Pengujian model KNNR.....	68
4.7.1.	Pengujian pada <i>Testing Data</i> .....	68
4.7.2.	Pengujian pada Proporsi Pembagian Data Lain .....	69
4.7.3.	Pengujian pada Dataset Kecamatan lain (kecamatan yang Bukan Sebagai Kecamatan Pembentuk Model) .....	69



4.7.4.	Pencarian Satu Model Terbaik untuk Seluruh Kecamatan	
	69	
4.7.5.	Pengujian dengan Metode lain.....	70
4.8.	Peramalan untuk 24 Periode .....	70
BAB V IMPLEMENTASI .....		72
5.1.	Persiapan implementasi .....	72
5.2.	Praproses Data .....	73
5.2.1.	Menyelaraskan Periode pada Data.....	73
5.2.2.	Pemilihan Kecamatan Pembentuk Model.....	74
5.2.3.	Uji Korelasi Variabel Observasi .....	75
5.2.4.	Mengatasi Nilai yang Hilang ( <i>Missing Value</i> ) .....	75
5.2.5.	Normalisasi .....	78
5.2.6.	<i>Windowing</i> .....	79
5.2.7.	Pembagian data.....	80
5.3.	Pembentukan Model .....	81
5.4.	Denormalisasi .....	82
5.5.	Uji Kelayakan .....	82
5.5.1.	RMSE82	
5.5.2.	SMAPE.....	83
5.6.	Peramalan 24 Periode .....	84
5.6.1.	Peramalan dengan <i>Double Exponential Smoothing</i> .....	85
5.6.2.	Peramalan dengan <i>Holt Winter</i> .....	86
5.6.3.	Peramalan dengan <i>Box Jenkins</i> atau ARIMA .....	87
BAB VI HASIL DAN PEMBAHASAN.....		91
6.1.	Hasil Praproses Data.....	91
6.1.1.	Penyelarasan Periode .....	91
6.1.2.	Mengatasi <i>Missing Value</i> .....	91
6.2.	Hasil Uji Korelasi untuk Memilih Kecamatan Pembentuk Model.....	98
6.2.1.	Kecamatan Pembentuk Model di Dataran Rendah .....	99
6.2.2.	Kecamatan Pembentuk Model di Dataran Sedang .....	99
6.2.3.	Kecamatan Pembentuk Model di Dataran Tinggi.....	99
6.3.	Hasil Uji Korelasi Variabel Observasi dengan Kasus Demam Berdarah.....	100

6.3.1.	Hasil Uji Korelasi Variabel Observasi di Dataran Rendah .....	100
6.3.2.	Hasil Uji Korelasi Variabel Observasi di Dataran Sedang .....	101
6.3.3.	Hasil Uji Korelasi Variabel Observasi di Dataran Tinggi ..	102
6.4.	Hasil Peramalan Variabel Observasi .....	103
6.4.1.	Hasil Peramalan Variabel Observasi dengan <i>Double Exponential Smoothing</i> .....	104
6.4.2.	Hasil Peramalan Variabel Observasi dengan <i>Holt's Winter Exponential Smoothing</i> .....	104
6.4.3.	Hasil Peramalan Variabel Observasi dengan <i>Box Jenkins</i> atau ARIMA .....	105
6.5.	Hasil Pembentukan Model di Dataran Rendah.....	106
6.5.1.	Perbandingan dengan Skenario Pembagian Data .....	108
6.5.2.	Uji Skenario Model pada Kecamatan Lain di Dataran Rendah.....	110
6.6.	Hasil Pembentukan Model di Dataran Sedang .....	113
6.6.1.	Perbandingan dengan Skenario Pembagian Data .....	115
6.6.2.	Uji Skenario Model pada Kecamatan Lain di Dataran Sedang .....	117
6.7.	Hasil Pembentukan Model di Dataran Tinggi .....	123
6.7.1.	Perbandingan dengan Skenario Pembagian Data .....	124
6.7.2.	Uji Skenario Model pada Kecamatan Lain di Dataran Tinggi	126
6.8.	Hasil Pencarian Satu Model Terbaik untuk Seluruh Kecamatan .....	128
6.9.	Perbandingan dengan Metode lain .....	129
6.10.	Perbandingan dengan Normalisasi di Setelah Proses Pembagian Data.....	130
6.11.	Kesimpulan Percobaan .....	130
6.12.	Analisis Hasil.....	132
6.12.1.	Analisis Hasil Peramalan.....	132
6.12.2.	Analisis Segi Manajerial.....	133
6.12.3.	Analisis Metode <i>K-Nearest Neighbor Regression</i> .....	135

BAB VII KESIMPULAN DAN SARAN .....	139
7.1. Kesimpulan .....	139
7.2. Saran.....	140
DAFTAR PUSTAKA.....	
BIODATA PENULIS.....	
LAMPIRAN A .....	
LAMPIRAN B.....	
LAMPIRAN C.....	
LAMPIRAN D .....	
LAMPIRAN E.....	
LAMPIRAN F.....	
LAMPIRAN G .....	

## DAFTAR GAMBAR

Gambar 1.1. Roadmap penelitian Laboratorium RDIB.....	29
Gambar 2.1. Ilustrasi permasalahan dengan interpolasi.....	41
Gambar 2.2. lagged input dan target [16].....	42
Gambar 2.3. Proses windowing [16].....	43
Gambar 2.4. ilustrasi k-nearest neighbor [13]. ....	44
Gambar 2.5. Ilustrasi Algoritma Brute Force dengan Nilai $k = 3$ [25].....	47
Gambar 2.6. Contoh Penggunaan <i>kd-tree</i> untuk KNN [24].....	48
Gambar 2.7. Contoh Penggunaan Algoritma <i>Ball-tree</i> : .....	48
Gambar 3.1. Tahapan pengerjaan Tugas Akhir.....	52
Gambar 4.1. Skema seluruh skenario pembentukan model.....	68
Gambar 6.1. Banyaknya Missing Value pada Data Angka Bebas Jentik .....	93
Gambar 6.2. Matrik Letak Missing Value pada Variabel ABJ ...	94
Gambar 6.3. Hasil Praproses Missing Value Variabel ABJ di Dataran Rendah .....	96
Gambar 6.4. Hasil Praproses Missing Value Variabel ABJ di Dataran Sedang .....	97
Gambar 6.5. Hasil Praproses Missing Value Variabel ABJ di Dataran Tinggi.....	97
Gambar 6.6. Hasil Praproses <i>Missing Value</i> variabel ABJ di Kecamatan Wajak.....	98
Gambar 6.7. Hasil Peramalan JP di Kecamatan Ngajum dengan <i>Double Exponential Smoothing</i> .....	104
Gambar 6.8. Hasil Peramalan SU Stasiun Karangates dengan <i>Holt's Winter Exponential Smoothing</i> .....	105
Gambar 6.9. Hasil Peramalan KU Stasiun Karangates dengan <i>Box Jenkins</i> .....	106
Gambar 6.10. Plot Prediksi dan <i>Testing Data</i> Dataran Rendah	107
Gambar 6.11. Plot Data Aktual dan Prediksi Dataran Rendah..	108
Gambar 6.12. Grafik Nilai RMSE Setiap Proporsi di Dataran Rendah.....	109

Gambar 6.13. Perbandingan Aktual dan Prediksi pada Kecamatan Gondanglegi .....	111
Gambar 6.14. Perbandingan Aktual dan Prediksi pada Kecamatan Donomulyo.....	111
Gambar 6.15. Perbandingan Aktual dan Prediksi pada Kecamatan Bululawang.....	112
Gambar 6.16. Perbandingan Aktual dan Prediksi pada Kecamatan Turen .....	113
Gambar 6.17. Plot Prediksi dan Data Testing Dataran Sedang ..	114
Gambar 6.18. Plot Data Aktual dan Prediksi Dataran Sedang ..	115
Gambar 6.19. Grafik Nilai RMSE Setiap Proporsi di Dataran Sedang .....	116
Gambar 6.20. Perbandingan Aktual dan Prediksi pada Kecamatan Tumpang.....	118
Gambar 6.21. Perbandingan Aktual dan Prediksi pada Kecamatan Lawang .....	119
Gambar 6.22. Perbandingan Aktual dan Prediksi pada Kecamatan Singosari.....	119
Gambar 6.23. Perbandingan Aktual dan Prediksi pada Kecamatan Wajak.....	120
Gambar 6.24. Perbandingan Aktual dan Prediksi pada Kecamatan Dampit .....	121
Gambar 6.25. Perbandingan Aktual dan Prediksi pada Kecamatan Sumbermanjing Wetan .....	122
Gambar 6.26. Perbandingan Aktual dan Prediksi pada Kecamatan Karangploso.....	122
Gambar 6.27. Plot Prediksi dan Data Testing Dataran Tinggi ..	123
Gambar 6.28. Plot Data Aktual dan Prediksi Dataran Tinggi ...	124
Gambar 6.29. Diagram Nilai RMSE Setiap Proporsi di Dataran Tinggi .....	125
Gambar 6.30. Perbandingan Aktual dan Prediksi pada Kecamatan Poncokusumo .....	127
Gambar 6.31. Perbandingan Aktual dan Prediksi pada Kecamatan Jabung.....	128

Gambar 6.32. Grafik Kasus Demam Berdarah di Kabupaten Malang.....	134
Gambar 6.33. Hasil Peramalan Jumlah Kasus Demam Berdarah di Kabupaten Malang tahun 2020.....	135
Gambar C.1. Skenario V0L01 di Dataran Rendah .....	
Gambar C.2. Skenario V1L09 di Dataran Rendah .....	
Gambar C.3. Skenario V2L02 di Dataran Rendah .....	
Gambar C.4. Skenario V3L06 di Dataran Rendah .....	
Gambar C.5. Skenario VAL11 di Dataran Rendah .....	
Gambar C.6. Skenario V0L01 di Dataran Sedang .....	
Gambar C.7. Skenario V1L03 di Dataran Sedang .....	
Gambar C.8. Skenario V2L00 di Dataran Sedang .....	
Gambar C.9. Skenario V3L02 di Dataran Sedang .....	
Gambar C.10. Skenario VAL001 di Dataran Sedang.....	
Gambar C.11. Skenario V0L09 di Dataran Tinggi.....	
Gambar C.12. Skenario V1L02 di Dataran Tinggi.....	
Gambar C.13. Skenario V2L02 di Dataran Tinggi.....	
Gambar C.14. Skenario V3L01 di Dataran Tinggi.....	
Gambar C.15. Skenario VAL01 di Dataran Tinggi.....	
Gambar E.1. Peramalan Jumlah Penduduk di Kecamatan Ngajum	
Gambar E.2. Peramalan Jumlah Penduduk di Kecamatan Jabung ..	
Gambar E.3. Peramalan Jumlah Penduduk di Kecamatan Poncokusumo .....	
Gambar E.4. Peramalan Curah Hujan di Kecamatan Ngajum .....	
Gambar E.5. Peramalan Curah Hujan di Kecamatan Jabung dan Poncokusumo .....	

## DAFTAR TABEL

Tabel 2.1. Penelitian sebelumnya.....	30
Tabel 4.1. Parameter dan nilai parameter pada KNNR .....	63
Tabel 4.2. Penamaan Skenario Independent Variable .....	65
Tabel 4.3 Skenario pembagian data.....	67
Tabel 4.4. Sampel Data untuk Uji Coba Metode Peramalan <i>Independent Variable</i> .....	71
Tabel 5.1. Spesifikasi Perangkat Keras dan Lunak .....	72
Tabel 5.2. Lingkungan Uji Coba .....	73
Tabel 5.3. Penjelasan Kode 5.1. ....	74
Tabel 5.4. Penjelasan kode 5.2. ....	75
Tabel 5.5. Penjelasan kode 5.3. ....	76
Tabel 5.6. Penjelasan kode 5.4. ....	77
Tabel 5.7. Penjelasan kode 5.5 .....	78
Tabel 5.8. Penjelasan kode 5.6. ....	79
Tabel 5.9. Penjelasan kode 5.8. ....	80
Tabel 5.10. Penjelasan kode 5.9. ....	81
Tabel 5.11. Penjelasan kode 5.10. ....	82
Tabel 5.12. Penjelasan kode 5.11. ....	83
Tabel 5.13. Penjelasan kode 5.12. ....	83
Tabel 5.14. Penjelasan kode 5.13. ....	84
Tabel 5.15. Penjelasan Kode 5.14 .....	86
Tabel 5.16. Penjelasan Kode 5.15. ....	87
Tabel 5.17. Penjelasan Kode 5.16. ....	88
Tabel 5.18. Penjelasan Kode 5.17. ....	89
Tabel 6.1. Sampel Dataset Cuaca Periode Bulan .....	91
Tabel 6.2. Banyaknya missing <i>value</i> pada Data Cuaca .....	92
Tabel 6.3. Sampel Data Cuaca Hasil Praproses <i>Missing Value</i> ...	95
Tabel 6.4. Nilai rata-rata Uji Korelasi tiap Kecamatan .....	98
Tabel 6.5. Uji Korelasi Variabel Observasi dan KDB di Dataran Rendah.....	100
Tabel 6.6. Uji Korelasi Variabel Observasi dan KDB di Dataran Sedang .....	101



Tabel 6.7. Uji Korelasi Variabel Observasi dan KDB di Dataran Tinggi .....	102
Tabel 6.8. Perbandingan Nilai RMSE pada Tiga Metode .....	103
Tabel 6.9. Model Peramalan pada training:test 80:20 di Dataran Rendah.....	106
Tabel 6.10. Perbandingan dengan Proporsi Pembagian Data Lain di Dataran Rendah.....	109
Tabel 6.11. Nilai RMSE di Setiap Kecamatan di Dataran Rendah .....	110
Tabel 6.12. Model Peramalan pada training:test 80:20 di Dataran Sedang .....	113
Tabel 6.13. Perbandingan dengan Proporsi Pembagian Data Lain di Dataran Sedang .....	115
Tabel 6.14. Nilai RMSE di Setiap Kecamatan di Dataran Sedang .....	117
Tabel 6.15. Model Peramalan pada training:test 80:20 di Dataran Tinggi .....	123
Tabel 6.16. Perbandingan dengan Proporsi Pembagian Data Lain di Dataran Tinggi .....	125
Tabel 6.17. Nilai RMSE di Setiap Kecamatan di Dataran Tinggi .....	126
Tabel 6.18. Perbandingan Median RMSE Hasil Pengujian di Seluruh Kecamatan.....	128
Tabel 6.19. Perbandingan Nilai RMSE dengan Metode lain ....	129
Tabel 6.20. Perbandingan RMSE dengan Normalisasi Setelah Pembagian Data.....	130
Tabel 6.21. Perbandingan RMSE Terbaik di Setiap Dataran ....	133
Tabel 6.22. Sensitifitas Parameter <i>n-neighbor</i> .....	136
Tabel 6.23. Sensitifitas Parameter <i>Metrics</i> .....	137
Tabel 6.24. Sensitifitas Parameter <i>Algorithm</i> .....	137
Tabel A.1. Hasil uji korelasi kasus demam berdarah di dataran rendah.....	146
Tabel A.2. Hasil Uji Korelasi di Dataran Sedang .....	146
Tabel A.3. Hasil Uji Korelasi di Dataran Tinggi.....	146

Tabel B.1. Hasil Peramalan Jumlah Penduduk di Kecamatan Ngajum .....	148
Tabel B.2. Hasil Peramalan Suhu (Stasiun BMKG Karangates) .....	148
Tabel B.3. Hasil Peramalan Kelembaban Udara (Stasiun BMKG Karangates).....	149
Tabel D.1. Sampel Dataset Cuaca BMKG .....	156
Tabel D.2. Data Cuaca Hasil Praproses.....	156
Tabel E.1. Hasil Peramalan Jumlah Penduduk di Kecamatan Ngajum .....	162
Tabel E.2. Hasil Peramalan Jumlah Penduduk di Kecamatan Jabung.....	163
Tabel E.3. Hasil Peramalan Jumlah Penduduk di Kecamatan Poncokusumo .....	163
Tabel E.4. Hasil Peramalan Curah Hujan di Kecamatan Ngajum .....	164
Tabel E.5. Hasil Peramalan Curah Hujan di Kecamatan Jabung dan Poncokusumo .....	164
Tabel F.1. Peramalan 24 Periode di Kecamatan Kepanjen .....	165
Tabel F.2. Peramalan 24 Periode di Kecamatan Gondanglegi ..	165
Tabel F.3. Peramalan 24 Periode di Kecamatan Donomulyo....	166
Tabel F.4. Peramalan 24 Periode di Kecamatan Bululawang...	166
Tabel F.5. Peramalan 24 Periode di Kecamatan Turen .....	166
Tabel F.6. Peramalan 24 Periode di Kecamatan Pakisaji .....	167
Tabel F.7. Peramalan 24 Periode di Kecamatan Tumpang .....	167
Tabel F.8. Peramalan 24 Periode di Kecamatan Lawang .....	168
Tabel F.9. Peramalan 24 Periode di Kecamatan Singosari.....	168
Tabel F.10. Peramalan 24 Periode di Kecamatan Wajak .....	169
Tabel F.11. Peramalan 24 Periode di Kecamatan Dampit.....	169
Tabel F.12. Peramalan 24 Periode di Kecamatan Sumbermanjing Wetan.....	170
Tabel F.13. Peramalan 24 Periode di Kecamatan Karangploso	170
Tabel F.14. Peramalan 24 Periode di Kecamatan Ngajum .....	171
Tabel F.15. Peramalan 24 Periode di Kecamatan Poncokusumo .....	171

Tabel F.16. Peramalan 24 Periode di Kecamatan Jabung .....	172
Tabel G.1. Pengujian Model Terbaik di Dataran Rendah .....	173
Tabel G.2. . Pengujian Model Terbaik di Dataran Sedang .....	173
Tabel G.3. . Pengujian Model Terbaik di Dataran Tinggi .....	174

## DAFTAR KODE

Kode 5.1. Uji korelasi kejadian demam berdarah antar kecamatan .....	74
Kode 5.2. Uji korelasi variabel observasi dengan variabel dependent.....	75
Kode 5.3. Mengisi missing value dengan nilai interpolasi .....	76
Kode 5.4. Mengisi missing value dengan regresi .....	77
Kode 5.5. Normalisasi min-max.....	78
Kode 5.6. Memanggil fungsi windowing .....	79
Kode 5.7. Fungsi <i>series_to_supervised</i> .....	80
Kode 5.8. Pembagian data training dan test data.....	80
Kode 5.9. Pembentukan model KNNR .....	81
Kode 5.10. Denormalisasi .....	82
Kode 5.11. Uji kelayakan dengan RMSE.....	83
Kode 5.12. Uji kelayakan model dengan SMAPE .....	83
Kode 5.13. Prediksi dengan model yang hanya melibatkan variabel lag .....	84
Kode 5.14. Peramalan Variabel Independen dengan <i>Double Exponential Smoothing</i> .....	85
Kode 5.15. Peramalan <i>Independent Variable</i> dengan Metode <i>Holt Winter</i> .....	86
Kode 5.16. Peramalan <i>Independent Variable</i> dengan Metode Box Jenkins .....	88
Kode 5.17. Peramalan dengan model yang melibatkan variabel observasi.....	89

## **BAB I**

### **PENDAHULUAN**

Pada bab ini, akan diuraikan proses identifikasi masalah penelitian yang meliputi latar belakang masalah, perumusan masalah, batasan masalah, tujuan penulisan beserta manfaatnya, serta relevansi pengerjaan tugas akhir.

#### **1.1. Latar Belakang**

Penyakit demam berdarah pertama kali dikenali di Filipina pada tahun 1953. Selang tiga dekade setelahnya, penyakit telah menyebar di Kamboja, Cina, Indonesia, Laos, Malaysia, Maldives, Myanmar, Singapura, Sri Lanka, Vietnam dan beberapa wilayah kepulauan pasifik. Di Indonesia, kasus DBD pertama kali ditemukan pada tahun 1970 [1]. Hingga saat ini selalu ada korban meninggal setiap tahunnya.

Menurut data Direktorat Jenderal Pencegahan dan Pengendalian Penyakit, Kementerian Kesehatan (Kemenkes) RI, terhitung sejak minggu pertama tahun 2018 hingga minggu pertama tahun 2019 telah tercatat 700 jumlah kasus demam berdarah di Jawa Timur dan menduduki peringkat satu kasus demam berdarah terbanyak di Indonesia [2].

Salah satu wilayah di Jawa Timur dengan angka kasus demam berdarah yang tinggi adalah Kabupaten Malang. Pada tahun 2018 terdapat 681 kasus demam berdarah dimana mengalami peningkatan dari tahun sebelumnya yang memiliki 451 kasus [3]. Angka kenaikan yang cukup besar ini menandakan perlu adanya perhatian lebih pada kasus ini.

Dinas kesehatan telah menyiapkan upaya mitigasi pencegahan demam berdarah dengan menggalakkan gerakan memanen jentik dan *fogging* [4]. Sayangnya tindakan mitigasi yang dilakukan sedikit terlambat karena dilakukan setelah angka kasus demam berdarah mengalami kenaikan. Dalam kasus

seperti ini, adanya informasi berupa peramalan jumlah kasus demam berdarah dapat membantu Dinas Kesehatan untuk menentukan upaya mitigasi yang tepat pada waktu yang tepat.

Tugas akhir ini mengacu pada penelitian mengenai pemodelan vektor demam berdarah dengan *machine learning* [5]. Penelitian tersebut membandingkan empat metode *machine learning* yaitu *Support Vector Machine (SVM)*, *Artificial Neural Network (ANN)*, *K-nearest Neighbor Regression (KNNR)* dan *Decision Tree Regressor*. Hasil penelitian tersebut menunjukkan bahwa KNNR menunjukkan performa paling baik dengan nilai *Mean Square Error (MSE)* terkecil yaitu 0.494 [5]. Penelitian ini menjadi dasar penggunaan metode KNNR untuk melakukan peramalan kasus demam berdarah pada tugas akhir ini.

Algoritma *nearest neighbor* cukup populer dan sangat sering digunakan untuk klasifikasi dan juga bisa untuk melakukan regresi [6]. Algoritma ini disukai karena sederhana, intuitif dan juga sangat mudah diimplementasikan. Implementasi pada data *time series* telah sering dilakukan seperti untuk meramalkan kurs [7], meramalkan biaya listrik [8], dan dikombinasikan dengan dekomposisi untuk meramalkan harga saham [9].

## 1.2. Rumusan Permasalahan

Dari uraian latar belakang yang telah dijelaskan diatas, maka rumusan masalah pada tugas akhir ini adalah sebagai berikut.

1. Bagaimana membangun model *K-Nearest Neighbor Regression (KNNR)* untuk meramalkan jumlah kasus demam berdarah di Kabupaten Malang?
2. Bagaimana akurasi model peramalan dengan KNNR untuk kasus demam berdarah?

3. Apa saja informasi terkait hasil peramalan yang harus diketahui pemerintah atau Dinas Kesehatan Kabupaten Malang?

### **1.3. Batasan Permasalahan**

Pada pengerjaan tugas akhir ini berikut batasan permasalahan yang telah ditetapkan:

1. Data yang dipakai sebagai *dependent variable* adalah data jumlah kasus demam berdarah di kabupaten Malang periode bulanan mulai Januari 2010 hingga Desember 2018. Data didapatkan dari Dinas Kesehatan Kabupaten Malang.
2. Data yang dipakai sebagai *independent variable* adalah angka bebas jentik, kepadatan penduduk, kecepatan angin, curah hujan, kelembapan udara, dan suhu.
3. Data kecepatan angin, curah hujan, kelembapan udara, dan suhu didapatkan dari stasiun BMKG Karangploso dan stasiun BMKG Karangates.
4. Penelitian ini dilakukan untuk menghasilkan peramalan jumlah kasus demam berdarah di Kabupaten Malang pada tahun 2019-2020.
5. Bahasa pemrograman yang digunakan pada pengerjaan tugas akhir ini adalah python 3.0.

### **1.4. Tujuan**

Pengerjaan tugas akhir ini memiliki beberapa tujuan yaitu:

1. Menerapkan metode *K-Nearest Neighbor Regression* untuk membuat model peramalan jumlah kasus demam berdarah di Kabupaten Malang.
2. Mengetahui hasil dan akurasi model peramalan kasus demam berdarah dengan metode KNNR.



### 1.5. Manfaat

Tugas akhir ini diharapkan dapat memberikan manfaat antara lain:

#### 1. Bagi instansi

Membantu instansi dengan memberikan informasi berupa hasil peramalan atau prediksi. Hasil peramalan diharapkan bisa menjadi pertimbangan untuk mengambil tindakan yang tepat dalam upaya mitigasi wabah demam berdarah.

#### 2. Bagi peneliti

Mampu memahami dan menerapkan metode *K-Nearest Neighbor Regression* untuk meramalkan jumlah kasus demam berdarah di Kabupaten Malang.

### 1.6. Relevansi

Demam berdarah selalu memakan korban setiap tahunnya. Angka kasus demam berdarah di Indonesia juga selalu bertambah. Demam berdarah telah menyebar hampir di seluruh wilayah di Indonesia namun jumlah kasus tertinggi ada di pulau Jawa. Provinsi Jawa Timur menduduki peringkat satu dalam kasus demam berdarah terbanyak sepanjang tahun 2018. Salah satu wilayah dengan kasus demam berdarah yang cukup banyak adalah Kabupaten Malang. Banyaknya kasus demam berdarah ini tentu harus ditekan dengan melakukan mitigasi wabah penyakit. Salah satu upaya mitigasi adalah deteksi wabah demam berdarah sejak dini. Deteksi sejak dini tentu akan lebih mudah dilakukan bila telah ada prediksi akan terjadinya kasus demam berdarah. Prediksi bisa diperoleh dengan melakukan peramalan kasus demam berdarah dengan memanfaatkan data historis dan data-data pendukung lainnya sebagai *independent variable*. *Independent variable* tersebut merupakan data-data pendukung yang bisa menjadi faktor yang memengaruhi

terjadinya kasus demam berdarah seperti curah hujan, tingkat kelembapan, suhu dan lain-lain. Peramalan bisa dilakukan dengan banyak metode. Pada tugas akhir ini, metode yang digunakan adalah *K-Nearest Neighbor Regression*.

Tugas akhir ini relevan dengan salah satu laboratorium yaitu Rekayasa Data dan Intelegensi Bisnis (RDIB). Bidang keilmuan yang relevan dengan tugas akhir ini adalah *business analytic* yang meliputi optimasi, *data/text/web mining*, *web analytic* dan peramalan. *Roadmap* penelitian laboratorium RDIB dapat dilihat pada . Tugas akhir ini juga relevan dengan mata kuliah penggalian data, teknik peramalan, sistem pendukung keputusan dan statistika.



**Gambar 1.1. Roadmap penelitian Laboratorium RDIB**

## BAB II TINJAUAN PUSTAKA

Bab ini berisi tentang penelitian yang telah dilakukan sebelumnya dan teori yang mendukung terkait dengan tugas akhir ini.

### 2.1. Studi Sebelumnya

Terdapat beberapa penelitian tentang peramalan dengan metode *k-Nearest Neighbor Regression* baik untuk meramalkan demam berdarah maupun untuk peramalan lainnya. Penelitian yang menjadi rujukan pada tugas akhir ini dapat dilihat pada Tabel 2.1.

**Tabel 2.1. Penelitian sebelumnya**

Penelitian Pertama	
Judul Penelitian	Modeling Dengue vector population using remotely sensed data and machine learning [5].
Penulis; Tahun	J. Scavuzzo, F. Trucco, M. Espinosa, C. Tauro, M. Abril, C. M. Scavuzzo and A. C. Frery; 2018
Deskripsi Umum	Penelitian ini menggunakan empat metode <i>machine learning</i> untuk membuat model vector demam berdarah. Empat metode yang digunakan adalah <i>Support Vector Machine (SVM)</i> , <i>Artificial Neural Network (ANN)</i> , <i>K-Nearest Neighbor Regression (KNNR)</i> dan <i>Decision Tree Regressor</i> . Dari keempat metode yang digunakan, model terbaik yang dihasilkan adalah <i>KNNR</i> dengan nilai <i>Mean Square Error (MSE)</i> terkecil yaitu 0.494. Selain itu, pembentukan model dilakukan dengan

	melibatkan banyak variabel. Model yang dihasilkan dapat mendeskripsikan hubungan non-linear antar variabel.
Keterkaitan Penelitian	Penelitian ini menjadi dasar penggunaan metode untuk tugas akhir ini. Penelitian ini membuktikan bahwa metode <i>k-nearest neighbor regression</i> dapat menghasilkan model yang terbaik dengan <i>error</i> terkecil dibandingkan dengan tiga metode lainnya untuk kasus demam berdarah.
Kelebihan	Penelitian ini menyajikan perbandingan beberapa metode dengan visualisasi sehingga lebih jelas. <i>Source code</i> tersedia dan bisa dipraktikkan langsung. Pada penelitian ini menggunakan <i>tools</i> konfigurasi untuk mengurangi kompleksitas <i>parameter tuning</i> .
Kekurangan	Langkah-langkah setiap metode dan parameter yang digunakan tidak ada penjelasan.
<b>Penelitian kedua</b>	
Judul Penelitian	Dealing with seasonality by narrowing the training set in time series forecasting with k-NN [10].
Penulis; Tahun	F. Martinez, M. P. Frías, M. D. Pérez-Godoy dan A. J. Rivera; 2018
Deskripsi Umum	Penelitian ini menerapkan metode KNNR untuk data <i>time series</i> . Penelitian ini menitikberatkan pada perlakuan untuk data <i>time series</i> yang <i>seasonal</i> . Ada tiga model yang dibentuk untuk melakukan perbandingan. Model pertama adalah KNNR tanpa ada perlakuan khusus untuk <i>seasonality</i> . Model kedua adalah KNNR yang di kombinasikan dengan pendekatan dekomposisi. Dan model yang ketiga

	adalah model yang dibangun oleh penulis yaitu dengan membuat pembelajaran KNNR di setiap musimnya. Hasil penelitian ini menunjukkan bahwa model yang dibangun penulis efektif dalam meningkatkan akurasi pada algoritma KNNR.
Keterkaitan Penelitian	Pengerjaan tugas akhir memiliki metode yang sama dengan penelitian ini. Selain itu, memiliki tujuan yang serupa yaitu untuk membuat model peramalan dengan metode KNNR sehingga penelitian ini menjadi acuan langkah yang dilakukan untuk membangun model dengan KNNR.
Kelebihan	Penelitian ini menitikberatkan pada <i>timeseries forecasting</i> dan memaparkan teknik peramalan dengan jelas. Model yang dihasilkan pada penelitian ini sangat handal untuk data <i>seasonal</i> .
Kelemahan	Model yang berhasil dibangun masih belum bisa mengatasi data yang bersifat <i>trend</i> . Setiap model yang dibangun hanya digunakan untuk satu <i>season</i> saja. Bila <i>season</i> banyak, maka perlu membangun banyak model untuk setiap <i>season</i> .
<b>Penelitian ketiga</b>	
Judul Penelitian	Forecasting the Number of Dengue Fever Cases in Malang Regency Indonesia Using Fuzzy Inference System Models [11].
Penulis; Tahun	Wiwik Anggraeni, I Putu Agus Aditya Pramana, Febriliyan Samopa, Edwin Riksakomara, Radityo P.Wibowo, Lulus Condro T., Pujiadi;2017

Deskripsi Umum	<p>Penelitian ini bertujuan untuk meramalkan jumlah kasus demam berdarah dengue di Kabupaten Malang. Metode yang digunakan untuk peramalan adalah <i>fuzzy inference system model</i>. Model peramalan yang dibangun melibatkan kondisi geografis sebagai faktor yang memengaruhi peramalan. Penelitian ini juga melibatkan angka kepadatan penduduk sebagai <i>independent variable</i>. Pada penelitian ini setiap kecamatan di kelompokkan menjadi tiga kelompok besar. Kelompok besar diidentifikasi dari ketinggian wilayah. Ketiga kelompok besar adalah dataran rendah, dataran sedang dan dataran tinggi. Untuk setiap kelompok besar dicari model terbaiknya. Hasil peramalan model FIS pada penelitian ini cukup baik. Nilai MAPE dataran rendah 6%, dataran sedang 12% dan dataran tinggi 14%.</p>
Keterkaitan Penelitian	<p>Kasus yang dikerjakan dalam tugas akhir adalah kasus yang sama dengan penelitian ini namun berbeda dalam periode pembuatan model dan periode peramalan. Penelitian ini menjadi referensi dalam mengerjakan tugas akhir.</p>
Kelebihan	<p>Akurasi yang dihasilkan pada model peramalan cukup tinggi. Pemaparan sangat jelas dan lengkap.</p>
Kelemahan	<p>Terdapat beberapa gambar yang tidak terlihat jelas.</p>
<b>Penelitian keempat</b>	
Judul Penelitian	<p>A methodology for Applying k-Nearest Neighbor to Time Series Forecasting [12]</p>

Penulis; Tahun	F. Martínez, M. D. Pérez, M. P. Frías dan A. J. Rivera; 2017
Deskripsi Umum	Penelitian ini berfokus pada penggunaan metode <i>k-nearest neighbor regression</i> untuk melakukan peramalan <i>time series</i> . Penelitian ini mencoba menggunakan strategi yang berbeda-beda pada pra-proses dan pembangunan model. Penggunaan strategi yang berbeda-beda bertujuan untuk mengetahui bagaimana efek dari masing-masing strategi tersebut. Data yang digunakan dalam penelitian ini adalah data kompetisi NN3. Yang disimpulkan dalam penelitian ini, pembentukan model adalah tahap yang paling penting. Kunci untuk meningkatkan akurasi pada penelitian ini adalah pemilihan variabel <i>input</i> .
Keterkaitan Penelitian	Tugas akhir dan penelitian ini menerapkan metode yang sama pada <i>time series data</i> . Beberapa langkah yang dilakukan pada penelitian ini menjadi acuan untuk pengerjaan tugas akhir.
Kelebihan	Penelitian ini memaparkan secara jelas penerapan metode KNNR untuk <i>time series data</i> . Strategi yang digunakan untuk pra-proses dan pembentukan model bervariasi.
Kelemahan	Pada penelitian ini menyebutkan bahwa metode KNNR tidak bisa digunakan untuk data yang mengandung trend. Perlu pendekatan <i>detrending</i> untuk data dengan <i>trend</i> . Namun, pada penelitian ini dataset yang dipakai tidak bisa digunakan untuk pendekatan <i>detrending</i> . Strategi <i>detrending</i>

	tidak bisa dilakukan dan pernyataan bahwa KNNR tidak bisa digunakan untuk data dengan <i>trend</i> belum bisa dibuktikan pada penelitian ini.
<b>Penelitian kelima</b>	
Judul Penelitian	Relative Evaluation of Regression Tools for Urban Area Electrical Energy Demand Forecasting [13].
Penulis; Tahun	N. J. Johannesen, M. Kolhe dan M. Goodwin; 2019
Deskripsi Umum	Penelitian ini bertujuan untuk meramalkan data <i>time series</i> tentang kebutuhan listrik. Penelitian ini membandingkan tiga <i>tools</i> untuk regresi yaitu, <i>random forest regressor</i> , <i>k-nearest neighbor regressor</i> dan regresi linear. Data yang digunakan memiliki tujuh <i>independent variable</i> . Lalu menggunakan uji korelasi untuk memilih <i>independent variable</i> yang akan dijadikan input pada pemodelan. Pemodelan dibangun untuk setiap jangka waktu peramalan yang berbeda. Jangka waktu peramalan yang digunakan adalah jangka pendek (30 menit) dan jangka Panjang (24 jam). Setelah melakukan pembuatan model dan pengujian model, didapatkan bahwa model paling baik untuk jangka pendek adalah <i>random forest</i> dan model terbaik untuk jangka panjang adalah <i>k-nearest neighbor</i> .
Keterkaitan Penelitian	Tugas akhir dan penelitian ini sama-sama memiliki banyak <i>independent variable</i> . Strategi yang digunakan pada tugas akhir untuk memilih <i>independent variable</i> adalah uji korelasi. Langkah uji korelasi



	dalam penelitian ini digunakan sebagai acuan untuk mengerjakan tugas akhir.
Kelebihan	Penelitian ini memaparkan secara lengkap dan jelas pembangunan model peramalan dengan <i>k-nearest neighbor</i> .
Kelemahan	Penelitian ini memiliki tujuan untuk membandingkan ketiga metode namun penjelasan tidak berimbang. Tidak ada penjelasan secara lengkap penggunaan <i>random forest</i> dan regresi linier.

## 2.2. Dasar Teori

Sub-bab ini berisi teori yang mendukung pengerjaan tugas akhir. Teori didapatkan dari literatur yaitu jurnal, buku dan artikel ilmiah.

### 2.2.1. Demam Berdarah

Demam berdarah adalah penyakit yang disebabkan oleh infeksi virus. Masa inkubasi penyakit dimulai dari infeksi virus dengue menyerang selama 3-14 hari. Demam berdarah ditularkan melalui gigitan nyamuk *Aedes aegypti* dan *Aedes albopictus* yang telah terinfeksi virus DBD dari penderita lainnya. Kedua nyamuk ini sudah menyebar hampir di seluruh pelosok Indonesia namun tidak di daerah yang memiliki ketinggian lebih dari 1000 meter diatas permukaan laut [1]. Nyamuk *Aedes* berwarna hitam dengan bitnik putih di badan, tangan dan sayapnya. Nyamuk yang memakan darah manusia, hanya nyamuk betina sedangkan nyamuk jantan memakan cairan bunga. Nyamuk betina mencari mangsa pada siang (pukul 09.00-10.00) dan sore hari (pukul 16.00-17.00). Tempat yang disenangi nyamuk ini adalah tempat yang lembab dan gelap.

Nyamuk *Aedes* bertelur di air menggenang yang bersih dengan suhu yang cocok  $26^0$ - $28^0$ C dan kelembapan 26 – 28. Telur

menetas setelah berada di air selama dua hari dan akan menjadi jentik. Setelah 6-8 hari jentik akan berubah menjadi nyamuk. [14]

Gejala penyakit demam berdarah umumnya berawal dari demam yang muncul tiba-tiba, nyeri pada persendian dan muncul ruam di kulit. Warna ruam umumnya merah terang dan mulai muncul di badan bagian bawah dan menyebar keatas.

Pencegahan demam berdarah paling utama adalah pembasmian vektor demam berdarah atau nyamuk *Aedes*. Pembasmian nyamuk dapat dilakukan dengan beberapa metode, yaitu:

- Secara fisik atau lingkungan dengan Pemberantasan Sarang Nyamuk (PSM), pengelolaan sampah yang tepat, pengurusan bak air dan tempat penampungan air lainnya minimal sekali dalam satu minggu.
- Secara biologi yaitu dengan memanfaatkan predator nyamuk seperti ikan pemakan jentik.
- Secara kimiawi yaitu pembasmian dengan bahan-bahan kimia seperti *fogging* untuk membunuh nyamuk dan menaruh abate di tempat penampungan air.
- Secara terpadu yaitu dengan mengombinasikan ketiga metode.

Adapun faktor-faktor yang memicu terjadinya kasus DBD pada manusia adalah:

- a. Faktor lingkungan seperti perubahan suhu, kelembapan udara dan curah hujan. Faktor lingkungan tersebut mengakibatkan nyamuk semakin sering bertelur sehingga vektor penularan penyakit DBD semakin banyak menyebabkan wabah.
- b. Faktor perilaku. Kurangnya perhatian masyarakat menjaga kebersihan lingkungan membuat nyamuk berkembang biak dengan baik dan cepat. Urbanisasi meningkat, mobilitas antar-daerah yang semakin baik menyebabkan penyakit DBD semakin mudah meluas [14].

### 2.2.2. Forecasting

*Forecasting* atau peramalan adalah proses dalam mengorganisir informasi tentang suatu fenomena masa lampau untuk memprediksi masa yang akan datang [15]. Saat ini peramalan telah dilakukan di segala sektor seperti energi, finansial, kesehatan dan lain-lain. *Time series forecasting* adalah proses untuk memprediksi nilai di masa yang akan datang dari data *time series* dengan nilai hasil observasi dan input lainnya [16]. Secara umum, *time series forecasting* dibagi menjadi empat metode yaitu:

#### 1. Dekomposisi

Dilakukan dengan memecah data *time series* menjadi beberapa komponen yaitu *seasonality*, *trend*, dan *noise*. Prediksi dilakukan pada komponen *seasonality* dan *trend*.

#### 2. Smoothing

Peramalan *time series* dilakukan dengan data observasi yang dimuluskan. Pada metode berbasis *smoothing*, nilai *time series* yang akan datang adalah rata-rata tertimbang dari nilai masa lalu.

#### 3. Regresi

Teknik pada regresi mirip dengan model prediktif konvensional yang memiliki *dependent variable* dan variabel independen. Model paling sederhana yaitu linear regression dengan formula:

$$y_t = a \times t + b \quad (2-1)$$

$y_t$  = nilai target variabel pada waktu  $t$

$a, b$  = koefisien yang bisa diestimasi untuk memprediksi nilai  $y$

#### 4. Machine learning

Teknik peramalan yang berbasis *supervised machine learning*. Teknik ini digunakan untuk membangun model peramalan berdasarkan simpulan hubungan antara input yaitu data historis dan target atau data yang diramalkan [16].

### 2.2.3. Uji korelasi

Uji korelasi diperlukan untuk mengukur hubungan dua variabel. Hasil dari uji korelasi ini digunakan untuk pemilihan fitur. Semakin menjauhi nol maka semakin besar nilai korelasi. Bila nilai korelasi positif maka hubungan antara dua variabel berbanding lurus. Namun, bila nilai korelasi negatif maka hubungan antara dua variabel berbanding terbalik. Nilai korelasi dihitung dengan menggunakan *the pearson correlation coefficient* ( $r$ ) dengan formula sebagai berikut [13]:

$$r = r_{xy} = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n \sum x_i^2 - (\sum x_i)^2} \sqrt{n \sum y_i^2 - (\sum y_i)^2}} \quad (2 - 2)$$

$r$  = nilai korelasi dengan the pearson correlation coefficient  
 $n$  = jumlah record  
 $x_i, y_i$  = variabel x dan y ke- i

### 2.2.4. Min-max normalization

Normalisasi merupakan proses transformasi fitur data menjadi skala. Normalisasi dilakukan untuk menghindari adanya dominasi oleh beberapa fitur. Salah satu cara normalisasi yaitu dengan *min-max normalization* [17], nilai fitur diubah menjadi skala 0 sampai 1 dengan formula berikut:

$$v' = \frac{v - \min_A}{\max_A - \min_A} \quad (2-3)$$

Keterangan :

$\max_A$  = nilai maksimum variabel A

$\min_A$  = nilai minimum variabel A

$v'$  = nilai variabel setelah dinormalisasi

$v$  = nilai asli variabel

### 2.2.5. Mengatasi *Missing Value*

Mengatasi *missing value* merupakan salah satu langkah dalam praproses data. Praproses adalah fase atau langkah mengidentifikasi, memilih dan mengatasi masalah pada data. Masalah *missing value* sangat sering dijumpai pada data. Masalah *missing value* sering terjadi pada data observasi yang diambil dari sensor. Pada data observasi, *missing value* bisa muncul karena kesalahan dari alat sensor atau penerima sensor. Bila pada data survey, *missing value* bisa muncul bila responden tidak berkenan menjawab karena masalah privasi [18].

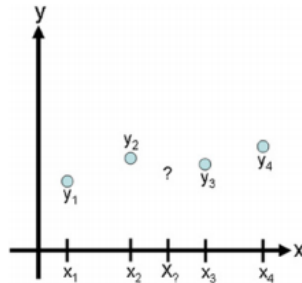
Ada banyak cara untuk mengatasi *missing value*. Metode paling dasar atau metode konvensional adalah dengan mengabaikan, menghapus, atau mengisi *missing value* tersebut. Ada beberapa metode untuk mengisi *missing value* yaitu dengan menganggapnya sebagai nol, atau mengestimasi nilai *missing value* dengan nilai rata-rata atau modus. Untuk mengetahui metode mana yang tepat digunakan, tentunya harus menyesuaikan dengan bentuk data. Metode konvensional hanya efektif bila persentase banyaknya *missing value* pada data kecil [18].

Langkah yang dipilih untuk mengatasi *missing value* akan berpengaruh pada akurasi analisis [19]. Selain metode konvensional yang telah dijelaskan, ada beberapa metode yang bisa dipakai seperti interpolasi linier dan regresi.

#### 2.2.5.1. Interpolasi Linier

Interpolasi adalah sebuah teknik untuk menghasilkan suatu fungsi estimasi yang melalui titik-titik yang dimiliki. Misalkan terdapat data  $x_i$  dan  $y_i$  atau *independent variable*  $x_i$  dan *dependent variable*  $y_i$ , tujuan kita adalah mencari sebuah fungsi  $f(x_i)=y_i$ . Fungsi  $f(x_i)$  inilah yang disebut dengan interpolasi [20]. Contoh

permasalahan yang diselesaikan dengan interpolasi dapat dilihat pada Gambar 2.1.



**Gambar 2.1. Ilustrasi permasalahan dengan interpolasi**

Umumnya, terdapat empat jenis fungsi interpolasi, yaitu:

- Interpolasi linear (*Linear interpolation*)
- Interpolasi kuadrat [21]
- Interpolasi *cubic-spline*
- Interpolasi *Lagrange-polynomial*

Yang akan dibahas kali ini adalah interpolasi linier. Interpolasi linier adalah salah satu jenis fungsi interpolasi yang paling sederhana. Dalam interpolasi linier, titik yang dicari atau nilai yang hilang diasumsikan terletak pada garis yang menghubungkan titik-titik terdekatnya. Misal, nilai  $x$  adalah nilai yang akan diestimasi. Titik  $x$  diasumsikan terletak diantara  $x_i$  dan  $x_{i+1}$ .

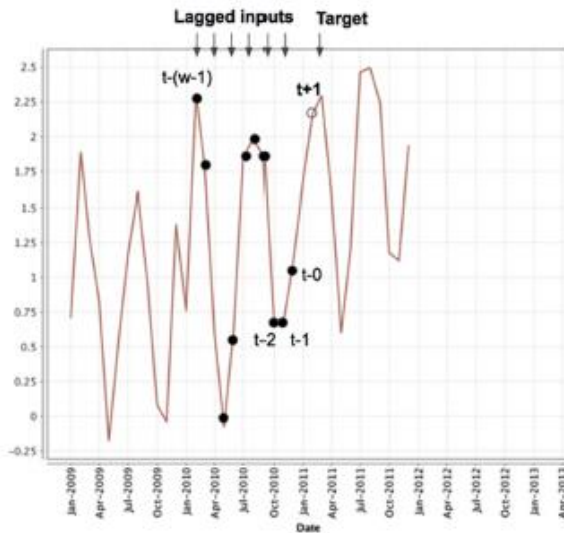
$$x_i < x_{i+1}, \quad x_i < x < x_{i+1}$$

Sehingga, fungsi interpolasi linier di  $x$  adalah:

$$f(x) = y_i + \frac{(y_{i+1} - y_i)(x - x_i)}{(x_{i+1} - x_i)}$$

### 2.2.6. Windowing

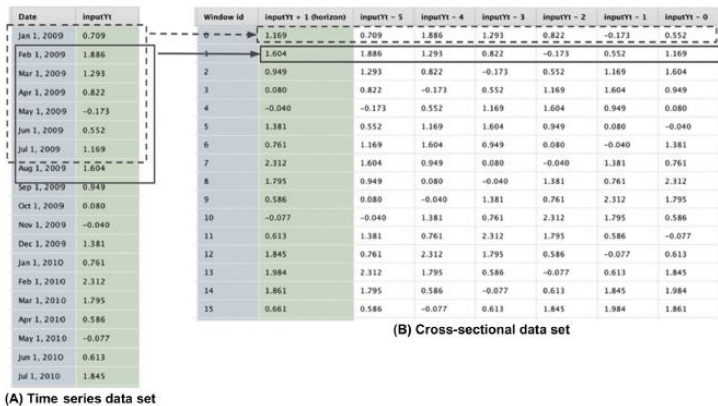
*Windowing* adalah Teknik untuk mengubah data *time series* menjadi *cross-sectional data* sebagai input *machine learning*. Himpunan data poin yang telah diketahui digunakan untuk memprediksi target. *Framework* umum dapat dilihat pada Gambar 2.1. dimana pendekatan ini serupa dengan teknik *supervised learning*.



Gambar 2.2. lagged input dan target [16]

Untuk melakukan *supervised learning* pada data *time series* maka perlu dilakukan *windowing*. Tahap ini bertujuan untuk mengubah data *time series* menjadi *window* atau *cross-sectional data*, dimana *record* terakhir menjadi target dan data poin lainnya atau *lagged data* menjadi variabel input untuk di bandingkan dengan target. Perubahan data *time series* menjadi data *cross-sectional* bisa dilihat pada Gambar 2.3. Setelah satu *window* konsekutif selesai didefinisikan, akan dilanjutkan

dengan *window* konsekutif selanjutnya sehingga antar *window* memiliki beberapa data *lagged* yang sama sebagai input variabel. Input variabel pada satu *window* bisa menjadi target variabel pada *window* lainnya. Model *supervised* dapat terbentuk dari proses pembelajaran hubungan antara variabel *lagged* input dan variabel target.



Gambar 2.3. Proses windowing [16]

Karakteristik *windowing* ditentukan oleh parameternya. Parameter yang bisa dispesifikasikan pada proses *windowing* adalah:

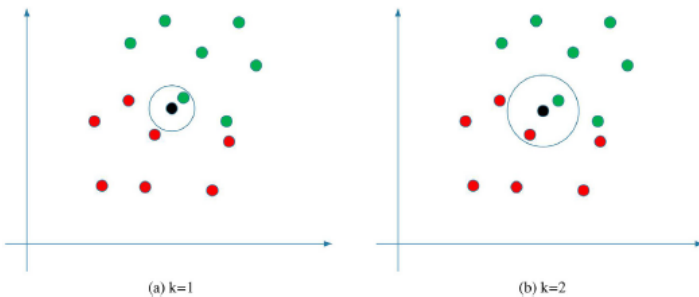
- *Window size*: banyaknya lag data poin yang digunakan sebagai input variable.
- *Step*: Jarak antara nilai pertama antara dua *window* konsekutif. Bila nilai *step* adalah 1 maka banyaknya *window* konsekutif yang terbentuk maksimal.
- *Horizon width*: Banyaknya *record* yang dijadikan sebagai variabel target.
- *Skip: offset* antara *window* dan horizon. Bila skip bernilai 0 maka data poin konsekutif dari *window* digunakan untuk horizon.



Pada gambar 2.2. parameter yang digunakan adalah *window size* = 6, *step* = 1, *horizon* = 1, dan *skip* = 0 [16].

### 2.2.7. K-Nearest Neighbor Regression

*K-nearest neighbor* adalah salah satu teknik *probabilistic classification* yang menggunakan pendekatan non-parametrik untuk mengestimasi kerapatan atau *density* [22]. KNN adalah algoritma klasik di *machine learning* yang digunakan untuk klasifikasi. Ilustrasi klasifikasi dengan *k-nearest neighbor* dapat dilihat pada Gambar 2.4. Bila diketahui data tanpa label dan data mayoritas dengan kelas  $k$  berdasarkan keserupaan, maka data mayoritas kelas  $k$  tersebut digunakan untuk memprediksi data tanpa label [10].



Gambar 2.4. ilustrasi k-nearest neighbor [13].

*K-nearest neighbor* dapat digunakan untuk regresi non-parametrik [6]. Pada KNN Regresi, rata-rata digunakan untuk menentukan variabel yang diregresi. Banyak pendekatan yang digunakan untuk menemukan KNN dari titik tertentu atau yang disebut *query point*. [23].

Untuk membuat model dengan KNN hal yang harus dilakukan terlebih dahulu adalah menentukan fungsi jarak dan nilai  $k$ .

### a. Menentukan metrik jarak

Untuk mengukur kedekatan antar titik data maka perlu untuk menghitung jarak antar titik. Beberapa metrik yang paling sering digunakan adalah [13]:

- *Manhattan/city block distance*

$$d(x, y) = \sum_{i=1}^k |x_i - y_i| \quad (2 - 4)$$

$x_i$  = nilai variabel  $x$  ke- $i$

$y_i$  = nilai variabel  $y$  ke- $i$

- *Euclidean distance*

$$d(x, y) = \sqrt{\sum_{i=1}^k (x_i - y_i)^2} \quad (2 - 5)$$

$x_i$  = nilai variabel  $x$  ke- $i$

$y_i$  = nilai variabel  $y$  ke- $i$

- *Minkowski distance*

$$d(x, y) = \left( \sum_{i=1}^k (|x_i - y_i|)^p \right)^{\frac{1}{p}} \quad (2 - 6)$$

$x_i$  = nilai variabel  $x$  ke- $i$

$y_i$  = nilai variabel  $y$  ke- $i$

*Minkowski* bisa disebut sebagai bentuk general dari *manhattan distance* dan *Euclidean distance*. Bila nilai  $p = 1$  maka sama dengan *manhattan distance*. Bila nilai  $p = 2$  maka sama dengan *Euclidean distance*.

- *Chebicev distance*

$$d(x, y) = \lim_{p \rightarrow \infty} \left( \sum_{i=1}^k (|x_i - y_i|)^p \right)^{\frac{1}{p}} \quad (2 - 7)$$

$x_i$  = nilai variabel  $x$  ke- $i$

$y_i$  = nilai variabel  $y$  ke- $i$

## b. Algoritma untuk Mencari Tetangga Terdekat

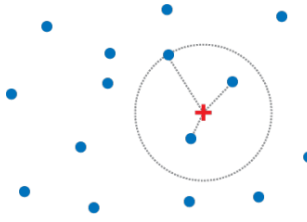
Ada beberapa pendekatan yang biasa digunakan untuk menemukan tetangga terdekat dari suatu titik yang ingin diprediksi (disebut *query point*). Algoritma yang biasa digunakan adalah *Brute Force*, *kd-tree*, dan *ball-tree*.

### 1) *Brute Force*

Kelebihan algoritma *brute force* adalah algoritma yang sangat sederhana untuk mencari titik terdekat dengan *query point* [24]. Titik  $q_i$  adalah *query point* dan titik  $r_i$  adalah titik referensi atau titik yang sudah diketahui nilai spesifiknya. Algoritma *brute-force* untuk mencari  $r_i$  terdekat dengan  $q_i$  adalah sebagai berikut [25]:

- a) Menghitung jarak antara  $q_i$  dengan semua  $r_i$
- b) Mengurutkan jarak dari yang terkecil.
- c) Memilih  $r_i$  sebanyak  $k$  berdasarkan  $k$  jarak terkecil.
- d) Mengulangi tahap a-c untuk setiap  $q_i$ .

Ilustrasi penggunaan *brute-force* dapat dilihat pada Gambar 2.5. Tanda (+) adalah *query point*. Garis melingkar menggambarkan jarak antara *query point* dengan  $r_i$  paling dekat ketiga.



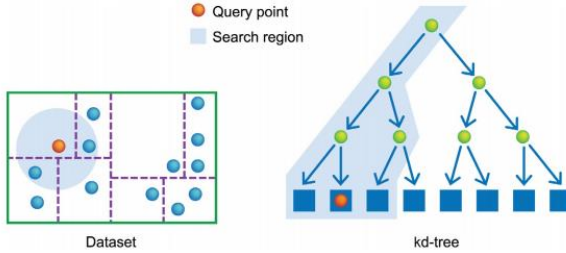
**Gambar 2.5. Ilustrasi Algoritma Brute Force dengan Nilai  $k = 3$  [25]**

Kelemahan algoritma ini adalah setiap proses *query point* membutuhkan memori dan waktu. Bila *query point* berjumlah besar, maka algoritma ini membutuhkan *runtime* cukup lama [24] sehingga algoritma ini biasa disebut “*exhaustive search*”.

## 2) *Kd-tree*

Algoritma *kd-tree* lebih efisien karena tidak perlu menghitung jarak antara *query points* dengan seluruh *reference point* untuk menemukan tetangga terdekat. Dengan algoritma *kd-tree*, banyak region yang telah dipangkas dan hanya beberapa titik referensi yang dihitung jaraknya. Konsep utama dalam algoritma *kd-tree* adalah *reference point* terdekat berada dalam satu partisi yang sama, sedangkan *reference point* yang tidak dalam partisi yang sama dapat diabaikan [24].

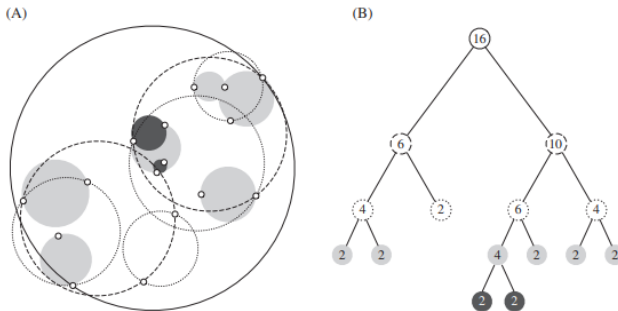
Contoh sederhana penggunaan *kd-tree* untuk *KNN* dapat dilihat pada Gambar 2.6. *kd-tree* adalah pohon biner yang membagi dataset dengan hyperplane dan kemudian membelah setiap partisi secara rekursif. Semua pembagian dibuat sejajar dengan salah satu sumbu baik secara vertikal maupun horizontal [26].



Gambar 2.6. Contoh Penggunaan *kd-tree* untuk KNN [24]

### 3) *Ball-tree*

Struktur *kd-tree* adalah struktur data yang baik dalam menemukan titik terdekat dan cukup efisien namun masih belum sempurna. Struktur data mungkin saja memiliki kemiringan namun bentuk partisi dipaksakan lurus. Bentuk persegi atau segi panjang sebenarnya bukan bentuk yang paling yang bisa digunakan karena sudutnya. Solusi dari masalah ini adalah dengan bentuk *hyperspheres*. Bentuk *spheres* atau bola mungkin bertindihan, namun bentuk persegi berbatasan [26]. Contoh penggunaan algoritma *ball-tree* dapat dilihat pada Gambar 2.7.



Gambar 2.7. Contoh Penggunaan Algoritma *Ball-tree* :  
 (A). Reference Point dan Bola      (B) Trees

### c. Memilih $k$

Terdapat dua pendekatan untuk memilih banyaknya *neighbor*. Pendekatan pertama memilih  $k$  langsung menggunakan heuristik seperti menentukan nilai  $k$  dengan menghitung akar dari jumlah *training set*. Pendekatan kedua dengan menggunakan optimasi pada kasus ini *training data* dibagi lagi menjadi *training set* dan *validation set*. Nilai  $k$  dipilih bila berhasil meminimumkan *error* pada *validation set*. [12].

### d. Kombinasi target

Untuk menghasilkan peramalan, target harus di asosiasikan dengan tetangga terdekatnya. Pada tahap ini dilakukan beberapa eksperimen untuk mengkombinasikan target yaitu:

- Nilai rataan
- Nilai median
- Rataan berbobot dimana bobot target berhubungan dengan kedekatan tetangga yang terasosiasi dengan nilai yang diramalkan. Tetangga yang lebih dekat memiliki bobot yang lebih besar.
- Rataan yang dipangkas dimana nilai terendah dan tertinggi dihapus dari rata-rata. Jika  $k$  kurang dari atau sama dengan 2 maka rataan yang dipangkas dihitung sebagai rataan [12].

### e. Memilih variabel masukan

Ada beberapa pendekatan untuk memilih variabel masukan. Pendekatan pertama adalah dengan menentukan *lags* 1 ke  $n$ , dimana  $n$  adalah periode musim. Contoh, bila *lags* 1-4 artinya terdapat empat periode yang dijadikan variabel masukan.

Pendekatan kedua adalah dengan menggunakan *partial autocorrelation function* (PACF) untuk memilih variabel masukan dari *lags* dengan autokorelasi yang signifikan. Meskipun PACF hanya digunakan untuk relasi liner, namun terbukti bisa menjadi cara yang efektif untuk memilih variabel masukan [12].

### 2.2.8. Uji kelayakan

Bertujuan untuk menguji akurasi hasil peramalan yang dihasilkan oleh model. Ada beberapa teknik yang biasa dilakukan untuk uji kelayakan yaitu *Symmetric Mean Absolute Percentage Error (SMAPE)* dan *Root Mean Squared Error (RMSE)*.

#### 2.2.8.1. *Symmetric Mean Absolute Percentage Error (SMAPE)*

Salah satu teknik untuk uji kelayakan adalah menghitung *Symmetric Mean Absolute Percentage Error (SMAPE)*. Ukuran SMAPE sebenarnya adalah pengembangan dari MAPE. Ukuran MAPE tidak bisa digunakan untuk data yang memiliki nilai aktual nol karena angka MAPE akan menjadi sangat besar [27]. Pemakaian SMAPE dapat mengatasi masalah asimetri dan nilai ekstrim dari teknik MAPE. Persamaan untuk menghitung SMAPE dapat dilihat pada persamaan (2-8) formula sebagai berikut [28]:

$$SMAPE = \frac{1}{h} \sum_{t=1}^h \frac{2|y_t(i) - \widehat{F}_{jt}(y(i))|}{|y_t(i)| + |\widehat{F}_{jt}(y(i))|} \quad (2-8)$$

Dimana  $h$  adalah horizon peramalan,

$\widehat{F}_{jt}(y(i))$  = Fungsi peramalan untuk series  $i$  dan model  $j$   
 $y_t(i)$  = Nilai aktual

#### 2.2.8.2. *Root Mean Squared Error (RMSE)*

Teknik lainnya yang bisa digunakan untuk melakukan pengujian adalah *Root Mean Squared Error (RMSE)*. RMSE telah digunakan sebagai metrik untuk mengukur kinerja model dalam penelitian meteorologi, kualitas udara, dan iklim [29]. RMSE tidak memiliki makna yang ambigu dan RMSE lebih tepat untuk mewakili kinerja model daripada *Mean Absolute Error (MAE)* ketika distribusi error yang diharapkan menjadi gaussian. RMSE memenuhi persyaratan untuk metrik jarak [29].

Persamaan untuk menghitung RMSE dapat dilihat pada persamaan (2-9)[13].

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (y_t - y'_t)^2}{n}}$$

(2-9)

$y_t$  = Nilai aktual  
 $y'_t$  = Nilai hasil peramalan  
 $n$  = jumlah periode pada validasi

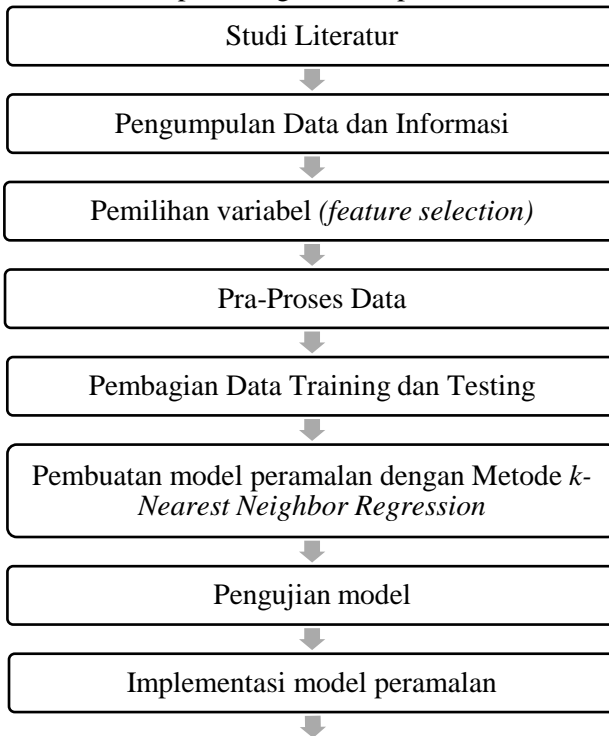


### BAB III METODOLOGI PENELITIAN

Pada bab ini berisi langkah-langkah penyusunan tugas akhir beserta penjelasannya.

#### 3.1. Tahapan Pelaksanaan Tugas Akhir

Langkah-langkah yang dilakukan dalam penyusunan tugas akhir ini bisa dilihat pada diagram alur pada Gambar 3.1.



Gambar 3.1. Tahapan pengerjaan Tugas Akhir

## **3.2. Uraian Metodologi**

Bagian ini berisi penjelasan secara rinci setiap tahapan pengerjaan tugas akhir yang sesuai dengan tahapan pada Gambar 3.1.

### **3.2.1. Studi literatur**

Tahap ini dilakukan dengan mencari literasi yaitu buku, jurnal dan artikel ilmiah lainnya dari berbagai sumber yang kredibel. Tujuan pada tahapan ini adalah untuk memahami konsep, teori dan mendapatkan masukan dari penelitian sebelumnya. Studi literatur yang dilakukan berdasarkan kasus yang serupa yaitu demam berdarah dan juga metode yang digunakan. Literatur yang digunakan adalah artikel ilmiah, jurnal dan buku.

Sumber referensi yang telah didapatkan dibaca, diringkas dan dicari keterkaitannya. Melalui proses ini, gagasan solusi permasalahan akan dihasilkan. Tahap selanjutnya adalah mempelajari *pra-processing* dan algoritma *K-Nearest Neighbor Regression*.

### **3.2.2. Pengumpulan data dan Informasi**

Tujuan dari tahapan ini adalah mendapatkan data dan informasi yang digunakan dalam penulisan tugas akhir ini. Data yang digunakan pada tugas akhir ini adalah data resmi dari Dinas Kesehatan Kabupaten Malang. Data yang didapatkan adalah angka kasus demam berdarah, kepadatan penduduk dan angka bebas jentik setiap bulan di setiap kecamatan. Data-data pendukung lain didapatkan dari Badan Meteorologi Klimatologi dan Geofisika (BMKG). Data yang didapatkan dari BMKG adalah kecepatan angin, curah hujan, kelembapan udara dan suhu.

Data setiap kecamatan akan dikelompokkan menjadi tiga kelompok berdasarkan kondisi geografisnya yaitu ketinggian wilayah. Kategori kelompok adalah dataran rendah (ketinggian

0 – 200 mdpl), dataran sedang (200 – 500mdpl) dan dataran tinggi (ketinggian 500 – 700 mdpl).

### 3.2.3. Pemilihan variabel (feature selection)

Data memiliki beberapa *independent variable*. Untuk melakukan peramalan yang melibatkan *independent variable* tidak harus semua *independent variable* dipakai. Salah satu cara adalah memilih satu variabel yang memiliki korelasi paling besar dengan *dependent variable* atau variabel yang ingin diramalkan. Uji korelasi dilakukan dengan persamaan (2).

### 3.2.4. Pra-proses data

Dilakukan pengecekan data terhadap *missing value*. Bila ditemui *missing value* maka harus dilakukan pra-pemrosesan untuk mengatasi *missing value* salah satunya dengan mengisinya dengan nilai *mean* atau modus dari variabel tersebut. Bila masalah *missing value* telah teratasi, selanjutnya melakukan pengecekan apakah antar variabel memiliki perbedaan rentang nilai yang tinggi. Bila nilai antar variabel memiliki perbedaan yang jauh maka perlu dilakukan normalisasi untuk menghindari dominasi variabel tertentu. Normalisasi dilakukan dengan persamaan (3).

Agar data *time series* bisa menjadi *input* metode KNNR, maka perlu dilakukan *windowing*. Hasil dari *windowing* adalah data *cross-sectional* yang akan menjadi *input* metode KNNR.

### 3.2.5. Pembangunan model peramalan dengan Metode *k-Nearest Neighbor Regression*

Tahap ini bertujuan untuk menghasilkan model yang digunakan untuk peramalan. Berikut adalah hal-hal penting untuk membangun model dengan KNNR.

### a. Memilih nilai $k$

Pendekatan yang dilakukan adalah membangun model dengan *training set* dengan variasi nilai  $k$ . Lalu memilih  $k$  dengan cara mencari nilai yang menghasilkan RMSE terkecil pada *test set*.  $K$  yang dapat menghasilkan *error* paling kecil dipilih untuk membangun melakukan peramalan.

### b. Menentukan metrik jarak

Pada tahap ini, dilakukan perhitungan jarak dengan metrik jarak yang telah ditentukan. Persamaan metrik jarak ada pada persamaan (2-4) hingga (2-7).

Pembangunan model peramalan menggunakan *training data* pada setiap skenario. Variasi model peramalan didapatkan dari *tuning parameter* sehingga banyak model yang akan dihasilkan.

### 3.2.6. Pengujian model

Pada tahap ini dilakukan pengujian model dengan menggunakan *testing data*. Performa didapatkan dari menghitung nilai *error* hasil peramalan model dan nilai aktual pada *testing data*. Pengukuran *error* dilakukan dengan SMAPE yang dihitung dengan persamaan (2-8) dan RMSE yang dihitung dengan persamaan (2-9). Hasil *error* inilah yang akan menjadi evaluasi model. Model dengan hasil akurasi tertinggi atau nilai *error* terkecil dipilih menjadi model terbaik yang akan digunakan untuk melakukan peramalan.

### 3.2.7. Implementasi model peramalan

Model dengan akurasi tertinggi atau *error* terkecil diimplementasikan untuk meramalkan *dependent variable*. *Horizon* atau periode yang akan diramalkan selama 24 periode atau dua tahun. Hasil peramalan tersebut dianalisis hingga menjadi informasi yang berguna untuk pihak Dinas Kesehatan Kabupaten Malang.

### **3.2.8. Penyusunan Laporan Tugas Akhir**

Pada penyusunan laporan tugas akhir ini, setiap langkah pengerjaan akan didokumentasikan. Hasil yang didapatkan dari setiap langkah akan disimpulkan. Penulisan laporan tugas akhir ini terdiri dari beberapa bagian sebagai berikut:

1. **BAB I PENDAHULUAN**  
Bab ini berisi penjelasan latar belakang penulisan, rumusan masalah, Batasan, tujuan dan manfaat penulisan, dan relevansi dengan penelitian laboratorium dan mata kuliah.
2. **BAB II TINJAUAN PUSTAKA**  
Bab ini berisi pembahasan penelitian yang telah dilakukan sebelumnya sebagai literatur dan landasan teori yang mendukung penulisan.
3. **BAB III METODOLOGI**  
Bab ini berisi langkah-langkah yang dilakukan untuk menyelesaikan penulisan. Langkah-langkah yang dilakukan pada bab ini merujuk kepada landasan teori pada bab sebelumnya.
4. **BAB IV PERANCANGAN**  
Bab ini berisi tentang bagaimana rancangan yang akan dilakukan untuk implementasi metode yang digunakan.
5. **BAB V IMPLEMENTASI**  
Bab ini membahas tentang setiap langkah yang dilakukan dalam implementasi metodologi yang digunakan pada tugas akhir.
6. **BAB VI HASIL DAN PEMBAHASAN**  
Bab ini berisi tentang hasil pembahasan dalam penyelesaian permasalahan yang diangkat dalam tugas akhir.
7. **BAB VIII KESIMPULAN DAN SARAN**  
Bab ini akan berisi kesimpulan dan saran sebagai pelengkap tugas akhir.

## **BAB IV PERANCANGAN**

Bab IV menjelaskan tentang perancangan untuk membangun model. Proses pembangunan model dilakukan sesuai dengan Langkah-langkah yang telah dijelaskan pada bab III metodologi. Langkah-langkah yang dilakukan dalam tahap perancangan meliputi pengumpulan data, praproses data, penentuan parameter, skenario dan *lag*.

### **4.1. Pengumpulan data**

Data yang digunakan dalam tugas akhir ini adalah data yang didapatkan dari Dinas Kesehatan Kabupaten Malang dan BMKG dari dua stasiun yang berbeda di Kabupaten Malang yaitu Stasiun BMKG Karangates dan stasiun BMKG Karangploso. Data dari Dinas Kesehatan Kabupaten Malang adalah angka kasus demam berdarah dan angka bebas jentik di Kabupaten Malang tahun 2010 – 2018 dalam periode bulan. Data yang didapatkan dari BMKG adalah suhu, kelembaban udara, curah hujan, dan kecepatan angin tahun 2010 – 2018 dengan periode harian.

Data kasus demam berdarah dan angka bebas jentik tersedia di 16 kecamatan di Kabupaten Malang. Enam belas kecamatan tersebut dikelompokkan menjadi tiga berdasarkan ketinggian wilayah. Kelompok tersebut adalah dataran rendah, dataran sedang dan dataran tinggi. Selanjutnya, data di setiap kecamatan akan dipasangkan dengan data BMKG dari stasiun yang terdekat dengan kecamatan tersebut.

### **4.2. Penentuan dataset kecamatan pembentuk model**

Kecamatan dikelompokkan menjadi 3 kelompok dataran berdasarkan letak geografis yaitu dataran rendah, dataran sedang dan dataran tinggi. Model akan dibangun untuk setiap kelompok dataran. Tujuan dari pengelompokan adalah untuk

melihat bagaimana pengaruh letak geografis terhadap kejadian demam berdarah [11]. Sebelum membangun model, langkah yang perlu dilakukan adalah memilih kecamatan yang paling mewakili datarannya. Untuk memilih satu kecamatan pembangun model maka dilakukan uji korelasi untuk antar kecamatan dalam dataran yang sama.

Langkah selanjutnya adalah menghitung rata-rata nilai korelasi di tiap kecamatan. Kecamatan dengan nilai rata-rata korelasi terbesar dianggap sebagai kecamatan yang paling mewakili semua kecamatan di datarannya. Kecamatan yang memiliki nilai rata-rata korelasi tertinggi akan menjadi kecamatan pembangun model.

### **4.3. Praproses Data**

Praproses data adalah tahapan yang perlu dilakukan sebelum membangun model. Tujuan dari praproses data adalah untuk mengurangi kompleksitas pada data mentah sehingga memudahkan proses selanjutnya. Ada beberapa hal yang dilakukan pada tahap praproses. Pertama, menyelaraskan periode pada data. Kedua, mengatasi nilai yang hilang atau *missing value* dengan cara yang disesuaikan bentuk masing-masing data. Ketiga, melakukan normalisasi atau penskalaan data sehingga tidak ada variabel yang mendominasi variabel lainnya.

#### **4.3.1. Menyelaraskan periode pada data**

Data yang akan digunakan memiliki periode bulanan dan harian. Sebelum masuk pada proses selanjutnya, periode pada data harus diselaraskan. Semua data akan disamakan periodenya menjadi bulanan. Untuk itu, yang perlu dilakukan adalah mengubah data dengan periode harian menjadi bulanan.

Data yang perlu diubah periodenya adalah data cuaca dari BMKG yang memiliki periode harian. Untuk mengubah periode

dari harian ke bulanan, cara yang dilakukan adalah menghitung rata-rata nilai setiap data dalam satu bulan. Hasil rata-rata dalam satu bulan tersebut menjadi nilai data cuaca pada bulan tersebut.

#### **4.3.2. Mengatasi nilai yang hilang (*handling missing value*)**

Pada tugas akhir ini, data yang digunakan memiliki beberapa nilai yang hilang atau *missing value*. Data-data yang hilang tersebut adalah data yang akan digunakan sebagai variabel terikat atau *dependent variable*. Data dengan *missing value* antara lain, data cuaca dari stasiun BMKG Karangates yaitu suhu, curah hujan, kelembaban udara dan kecepatan angin. Data dengan *missing value* juga ditemukan pada data dari Dinas Kesehatan Kabupaten Malang yaitu Angka Bebas Jentik (ABJ). Untuk mengatasi masalah *missing value* dilakukan beberapa Langkah yang berbeda karena disesuaikan dengan bentuk masing-masing data.

##### **a. Mengisi *missing value* dengan rata-rata**

Salah cara yang bisa dilakukan untuk mengisi *missing value* adalah dengan mencari nilai rata-rata dari *record* sebelumnya. *Mengisi missing value* dengan rata-rata sebaiknya dilakukan bila banyaknya *missing value* kurang dari 20%. Bila terlalu banyak *missing value*, mengisi dengan nilai rata-rata bukanlah cara yang tepat karena akan mengakibatkan banyak nilai yang sama [18].

Data suhu, curah hujan, kelembaban udara dan kecepatan angin dari BMKG memiliki sifat yang sama yaitu berbentuk musiman atau *seasonal* dalam 12 periode atau satu tahun. Cara yang dilakukan untuk mengatasi *missing value* pada data suhu, curah hujan, kelembaban udara dan kecepatan angin sama yaitu dengan mencari nilai rata-rata pada bulan yang sama pada periode sebelumnya.

##### **b. Mengisi *missing value* dengan interpolasi**



Data Angka Bebas Jentik (ABJ) dari Dinas Kesehatan memiliki *missing value* dengan jumlah yang cukup banyak yaitu diatas 20%. Tiap kecamatan memiliki angka dan letak *missing value* yang berbeda. Pada beberapa kecamatan, *missing value* terletak pada awal baris dan muncul berurutan. Data ABJ juga tidak bersifat musiman sehingga cara mengisi *missing value* dengan mencari rata-rata pada *season* yang sama tidak bisa dilakukan. Langkah mengisi *missing value* dengan interpolasi dilakukan pada data dengan jumlah *missing value* diatas 20% dan kemunculan *missing value* tidak berurutan.

### c. Mengisi *missing value* dengan regresi linear

*Missing value* diatas 20% dengan kemunculan yang berurutan tidak bisa diatasi dengan interpolasi karena akan mengakibatkan banyak nilai yang sama. Untuk mengisi *missing value* cara yang dilakukan adalah dengan regresi linier. Model regresi liner dibentuk dengan menggunakan data yang dimiliki.

Model regresi dibangun dengan menggunakan *record* yang utuh atau tanpa *missing value*. Model regresi tersebut digunakan untuk memprediksi nilai *missing value*.

#### 4.3.3. Normalisasi

Dataset yang digunakan memiliki rentang nilai yang jauh berbeda antara satu variabel dengan variabel lainnya. Perbedaan paling jauh pada variabel Kecepatan Angin (KA) dengan rentang nilai 0 – 2, sedangkan variabel Jumlah Penduduk (JP) memiliki nilai ratusan ribu. Perbedaan rentang nilai yang sangat besar menyebabkan dominasi variabel dengan rentang nilai yang besar terhadap variabel dengan rentang nilai yang kecil [17]. Untuk menghindari adanya dominasi variabel, data diubah bentuknya menjadi skala. Proses ini dinamakan normalisasi. Normalisasi dilakukan dengan *Min-Max* dengan formula (2-3).

Tahun	Periode	SU	KU	CH	KA	J
2010	Januari	26.02	84.74	10.52	0.84	1
2010	Februari	26.32	84.64	12.04	0.86	1
2010	Maret	26.39	85	8.65	0.55	1
2010	April	26.29	86.17	10.37	0.3	1
2010	Mei	26.57	85.61	7.35	0.58	1
2010	Juni	25.62	84.1	9.23	1.59	1
2010	Juli	25.35	83.52	2.16	1.77	1
2010	Agustus	25.35	81.71	3.94	1.68	1
2010	September	25.73	83.93	9.4	1.37	1
2010	Oktober	26.19	82.03	12.13	1.71	1
2010	November	26	84.53	17.53	1.23	1

#### 4.3.4. Windowing

Proses *windowing* adalah proses mengubah data *time series* menjadi *cross-sectional data* dengan cara menambah *window* baru atau menambah variabel *lag*. Seperti yang telah dijelaskan pada sub-bab 2.2.6, nilai *lag* yang dipilih akan mengurangi *record* sebanyak nilai *lag* tersebut. Untuk menghindari hilangnya banyak data, maka nilai *lag* dalam proses *windowing* ini berkisar antara 1 hingga 12.

#### 4.4. Penentuan Parameter KNNR, Independent Variable dan lag

Parameter KNNR yang digunakan akan mempengaruhi akurasi model. Penentuan parameter KNNR yang tepat akan menghasilkan model yang baik. Selain parameter KNNR,

*independent variable* juga dapat mempengaruhi akurasi model. *Independent variable* yang akan digunakan pada tugas akhir ini dikelompokkan menjadi dua yaitu *independent variable* observasi dan *independent variable lag*. *Independent variable* observasi terdiri dari data curah hujan, suhu, kelembaban udara, kecepatan angin, angka bebas jentik dan jumlah penduduk. Sedangkan, *independent variable lag* adalah *variable lag* yang dihasilkan dari proses *windowing*.

#### 4.4.1. Menentukan parameter dan nilai parameter

Untuk membuat model KNNR hal yang harus dilakukan adalah menentukan parameter beserta nilainya. Pada KNNR parameter yang mempengaruhi pembentukan model adalah *n\_neighbor*, *weights*, *algorithm*, *metric*. Nilai pada masing-masing parameter dapat dilihat pada Tabel 4.1.

##### a) *N\_neighbor*

*N\_neighbor* adalah banyaknya tetangga atau titik terdekat yang digunakan untuk mengevaluasi titik yang diestimasi atau yang akan diprediksi. Parameter *n\_neighbor* adalah parameter yang paling berpengaruh dalam pembentukan model *k-nearest neighbor regression*.

##### b) *Weights*

*Weights* adalah pembobotan pada titik yang paling dekat. Parameter *weights* memiliki dua nilai yaitu *distance* dan *uniform*. *Distance* bila dilakukan pembobotan berdasarkan jarak sedangkan *uniform* bila tidak dilakukan pembobotan pada suatu titik tertentu.

##### c) *Algorithm*

KNN dilakukan dengan mencari titik-titik terdekat untuk mengevaluasi nilai prediksi. Untuk mencari titik-titik terdekat,

dapat dilakukan dengan beberapa algoritma. Parameter *algorithm* digunakan untuk menentukan algoritma apa yang dipakai untuk mencari titik-titik terdekat.

#### d) *Metric*

Untuk mengetahui suatu titik apakah titik terdekat atau tidak maka dilihat dari jarak. Ada beberapa formula untuk menghitung jarak. Parameter *metric* digunakan untuk menentukan formula apa yang digunakan untuk menghitung jarak antara dua titik.

**Tabel 4.1. Parameter dan nilai parameter pada KNNR**

Parameter	Nilai
n_neighbor	2,3,4,5,6,7
Weights	Distance
Algorithm	Brute, kd_tree, ball_tree
Metric	Euclidean, manhattan, chebyshev

#### 4.4.2. Uji korelasi setiap variabel observasi dengan *dependent variable*

Nilai akurasi model juga dipengaruhi oleh *independent variable* yang terlibat baik dari variabel observasi maupun dari variabel *lag*. Variabel observasi terdiri dari data suhu, curah hujan, kelembaban udara, kecepatan angin, angka bebas jentik dan jumlah penduduk. Setiap variabel observasi akan dicari nilai korelasinya dengan *dependent variable*. *Dependent variable* adalah variabel yang dijadikan target prediksi. Variabel yang ingin diprediksi adalah variabel kejadian demam berdarah. Uji korelasi antara variabel observasi dan *dependent variable* dilakukan di setiap dataran.

#### 4.4.3. Menentukan *lag*

Data yang digunakan dalam pengerjaan tugas akhir adalah data *time series*. Data *time series* perlu diubah bentuknya sehingga

dapat menjadi *input* KNNR dengan cara *windowing* atau berbentuk *lag*. Jumlah *lag* yang digunakan juga akan memengaruhi akurasi model. Oleh karena itu, percobaan dilakukan dengan beberapa nilai *lag*. Namun, nilai *lag* yang besar dapat mengurangi *training data* sehingga model yang dibangun kurang baik. Untuk menghindari hilangnya *training data* terlalu banyak, maka pada tugas akhir ini dilakukan percobaan dengan *lag* 1 hingga *lag* 12. Variabel baru yang dihasilkan dari proses *windowing* disebut variabel *lag*.

#### **4.4.4. Pembentukan skenario *tuning parameter***

Skenario pembentukan model dibentuk berdasarkan *tuning parameter*, nilai *lag* dan *independent variable* yang terlibat. *Tuning parameter* yang dilakukan berdasarkan pada nilai parameter KNNR pada Tabel 4.1. Skenario *tuning parameter* berdasarkan Tabel 4.1 menghasilkan 54 kombinasi percobaan. Skenario *tuning parameter* akan digabungkan dengan skenario lain. Skema skenario pembentukan model dapat dilihat pada Gambar 4.1.

#### **4.5. Pembentukan skenario *independent variable***

*Independent variable* dapat berasal dari variabel observasi dan/atau variabel *lag*. Pada skenario *independent variable* terdapat lima kombinasi percobaan. Percobaan pertama, *independent variable* hanya berupa variabel *lag*. Percobaan kedua, *independent variable* berupa variabel observasi dengan nilai korelasi tertinggi. Percobaan ketiga, *independent variable* adalah dua variabel observasi dengan nilai korelasi tertinggi. Percobaan keempat, *independent variable* adalah tiga variabel observasi dengan nilai korelasi tertinggi. Dan percobaan kelima, *independent variable* adalah semua variabel observasi. Skenario *independent variable* akan dikombinasikan dengan skenario lainnya. Skema skenario pembentukan model dapat dilihat pada

Gambar 4.1. Penamaan skenario *independent variable* dapat dilihat pada Tabel 4.2.

**Tabel 4.2. Penamaan Skenario Independent Variable**

<b>Kode Skenario</b>	<b>Penjelasan</b>
V0LXX	Hanya melibatkan variabel <i>lag</i> . XX adalah nilai <i>lag</i> .
V1LXX	Melibatkan 1 variabel observasi dengan nilai korelasi tertinggi. XX adalah nilai <i>lag</i>
V2LXX	Melibatkan 2 variabel observasi dengan nilai korelasi tertinggi. XX adalah nilai <i>lag</i>
V3LXX	Melibatkan 3 variabel observasi dengan nilai korelasi tertinggi. XX adalah nilai <i>lag</i>
VALXX	Melibatkan semua variabel observasi. XX adalah nilai <i>lag</i>

#### **4.5.1. Hanya melibatkan variabel *lag***

Pembentukan model hanya melibatkan variabel *lag* sebagai *independent variable* untuk memprediksi variabel Kejadian Demam Berdarah (KDB). Percobaan dilakukan untuk nilai *lag* 1 hingga *lag* 12 dan dikombinasikan dengan skenario tuning parameter. Percobaan yang dilakukan untuk setiap skenario *independent variable* yang hanya melibatkan variabel *lag* adalah 648 percobaan dan akan dikombinasikan dengan skenario lainnya. Skema skenario pembentukan model dapat dilihat pada Gambar 4.1.

#### **4.5.2. Melibatkan satu variabel observasi dengan nilai korelasi tertinggi**

Skenario *independent variable* dengan melibatkan satu variabel observasi (Suhu/Curah hujan/Kelembaban Udara/Kecepatan Angin/Jumlah Penduduk/Angka bebas Jentik) dengan nilai korelasi tertinggi pada langkah 4.4.2. Variabel terpilih akan menjadi prediktor dan akan dikombinasikan dengan variabel *lag* 1 hingga *lag* 12. Total percobaan yang dilakukan adalah 702

percobaan (64 percobaan tanpa melibatkan *lag* dan 648 melibatkan *lag* 1 hingga *lag* 12) untuk setiap dataran.

#### **4.5.3. Melibatkan dua variabel observasi dengan nilai korelasi tertinggi**

Skenario *independent variable* dengan melibatkan dua variabel observasi (Suhu/Curah hujan/Kelembaban Udara/Kecepatan Angin/Jumlah Penduduk/Angka bebas Jentik) dengan nilai korelasi tertinggi pada langkah 4.4.2. Variabel terpilih akan menjadi prediktor dan akan dikombinasikan dengan variabel *lag* 1 hingga *lag* 12. Total percobaan yang dilakukan adalah 702 percobaan (64 percobaan tanpa melibatkan *lag* dan 648 melibatkan *lag* 1 hingga *lag* 12) untuk setiap dataran.

#### **4.5.4. Melibatkan tiga variabel observasi dengan nilai korelasi tertinggi**

Skenario *independent variable* dengan melibatkan tiga variabel observasi (Suhu/Curah hujan/Kelembaban Udara/Kecepatan Angin/Jumlah Penduduk/Angka bebas Jentik) dengan nilai korelasi tertinggi pada langkah 4.4.2. Variabel terpilih akan menjadi prediktor dan akan dikombinasikan dengan variabel *lag* 1 hingga *lag* 12. Total percobaan yang dilakukan adalah 702 percobaan (64 percobaan tanpa melibatkan *lag* dan 648 melibatkan *lag* 1 hingga *lag* 12) untuk setiap dataran.

#### **4.5.5. Melibatkan semua variabel observasi**

Skenario *independent variable* dengan melibatkan semua variabel observasi (Suhu/Curah hujan/Kelembaban Udara/Kecepatan Angin/Jumlah Penduduk/Angka bebas Jentik). Semua variabel terpilih akan menjadi prediktor dan akan dikombinasikan dengan variabel *lag* 1 hingga *lag* 12. Total percobaan yang dilakukan adalah 702 percobaan (64 percobaan tanpa melibatkan *lag* dan 648 melibatkan *lag* 1 hingga *lag* 12) untuk setiap dataran.

#### 4.6. Pembagian *training data* dan *testing data*

Pembagian data adalah satu tahap penting dalam proses peramalan. Proporsi pembagian dataset menjadi *training* dan *testing* data juga dapat memengaruhi akurasi model. Data *training* digunakan untuk membangun model. Sedangkan data *testing* digunakan untuk menguji model yang telah dibangun. Untuk mendapatkan model dengan nilai *error* terkecil maka dilakukan percobaan pada beberapa proporsi pembagian data. Proporsi *training* dan *testing data* bervariasi. Setiap variasi akan menjadi satu skenario pembangunan model. Skenario pembagian *training data* dan *testing data* dapat dilihat pada Tabel 4.3.

Perbedaan ukuran pembagian data *training-testing* dilakukan karena ukuran data *training* memiliki pengaruh pada performa model. Perbedaan ukuran data *training* juga digunakan untuk menguji model apakah performa model dengan skenario terbaik terdegradasi cukup banyak atau tidak [30]. Bila performa model dengan skenario terbaik terdegradasi cukup banyak saat data *training* dikurangi, maka model tersebut dikatakan tidak *robust* terhadap ukuran pembagian data.

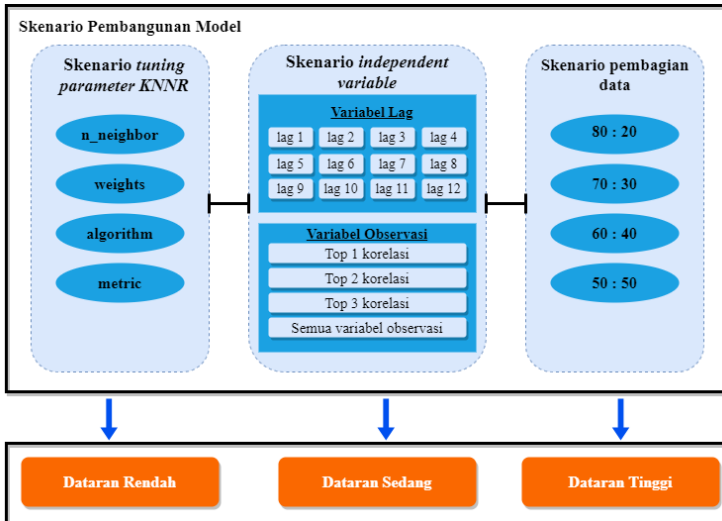
**Tabel 4.3 Skenario pembagian data**

Skenario	Pembagian data	
	<i>Training data</i>	<i>Testing data</i>
Skenario 1	80%	20%
Skenario 2	70 %	30 %
Skenario 3	60 %	40 %
Skenario 4	50 %	50 %

Percobaan dengan empat skenario pembagian data dikombinasikan dengan skenario *independent variable*. Total percobaan satu skenario pembagian data adalah 3456 percobaan (jumlah percobaan pada skenario *independent variable*). Percobaan dilakukan untuk setiap kelompok dataran (rendah,



sedang, tinggi). Skema skenario pembentukan model dapat dilihat pada Tabel 4.1.



**Gambar 4.1.** Skema seluruh skenario pembentukan model

## 4.7. Pengujian model KNNR

Model yang berhasil terbentuk akan diuji pada *testing data*. Lalu dipilih model terbaik di setiap skenario *independent variable*. Model dengan RMSE paling kecil akan dijadikan sebagai model terpilih untuk melakukan peramalan. Model terbaik setiap *independent variable* akan diuji menggunakan proporsi pembagian data lain dan data kecamatan lain.

### 4.7.1. Pengujian pada *Testing Data*

Tiap percobaan akan menghasilkan sebuah model. Model yang telah dibangun akan diuji dengan menggunakan *testing data*. Hasil prediksi pada *testing data* yang dihasilkan masih berbentuk data skala atau normal. Untuk itu, data harus diubah

dengan proses denormalisasi. Hasil prediksi yang telah didenormalisasi akan diuji dengan RMSE dan SMAPE pada formula (2-8) dan (2-9). Model terbaik yang dipilih didasarkan pada nilai RMSE dan SMAPE pengujian *testing data* yang paling kecil.

#### **4.7.2. Pengujian pada Proporsi Pembagian Data Lain**

Setiap skenario yang menghasilkan model terbaik pada setiap skenario *independent variable* akan diujikan pada proporsi pembagian data lain. Nilai RMSE di setiap proporsi pengujian akan dihitung dengan persamaan (2-8). Hasilnya, setiap model terbaik di skenario *independent variable* memiliki empat nilai RMSE dari pengujian empat proporsi pembagian data (80:20, 70:30, 60:40, 50:50). Dari keempat proporsi pembagian data tersebut, akan dihitung nilai rata-rata dan standar deviasi. Nilai rata-rata dan standar deviasi RMSE di setiap model terbaik akan digunakan untuk evaluasi *robustness* terhadap pembagian data yang berbeda dari setiap model terbaik.

#### **4.7.3. Pengujian pada Dataset Kecamatan lain (kecamatan yang Bukan Sebagai Kecamatan Pembentuk Model)**

Model terbaik di setiap *independent variable* juga akan diujikan ke kecamatan lain yang bukan sebagai pembentuk model. Nilai RMSE yang dihasilkan di setiap kecamatan akan dihitung nilai rata-ratanya. Nilai rata-rata RMSE uji kecamatan di setiap model terbaik akan dijadikan dasar untuk melakukan evaluasi *robustness* terhadap data di kecamatan lain.

#### **4.7.4. Pencarian Satu Model Terbaik untuk Seluruh Kecamatan**

Model terbaik yang telah didapatkan dari setiap skenario di setiap dataran akan dibandingkan. Model terbaik di dataran rendah akan diujikan di kecamatan lain di dataran sedang dan tinggi. Model terbaik di dataran sedang akan diujikan di kecamatan lain di dataran rendah dan tinggi. Begitu pula, model

terbaik di dataran tinggi akan diujikan di kecamatan lain di dataran rendah dan sedang.

Pengujian model tersebut menghasilkan RMSE di setiap kecamatan yang dicoba. Nilai RMSE dari ke-enam belas kecamatan tersebut akan dicari nilai median. Model dengan nilai median RMSE paling kecil akan digunakan sebagai model tunggal yang bisa menghasilkan peramalan untuk semua kecamatan baik di dataran rendah, dataran sedang dan dataran tinggi. Nilai median digunakan karena tidak dipengaruhi oleh nilai ekstrim.

#### **4.7.5. Pengujian dengan Metode lain**

Data yang sama digunakan untuk membentuk model peramalan dengan metode lain. Tujuan tahapan ini adalah untuk membandingkan performa model yang dihasilkan *k-nearest neighbor* (KNN-R) dengan metode lain. Metode yang akan dibandingkan dengan KNN-R adalah metode RNN-LSTM, *Linear Regression*, XG-Boost, Random Forest, dan RBFNN.

#### **4.8. Peramalan untuk 24 Periode**

Model dengan *error* terkecil pada langkah 4.7. akan digunakan untuk memprediksi Kejadian Demam Berdarah (KDB) untuk 24 periode ke depan. Model terpilih di setiap dataran akan digunakan untuk menghasilkan prediksi di setiap kecamatan. Bila model terbaik adalah model dengan *multivariate* yang membutuhkan *independent variable*, maka *independent variable* tersebut akan diprediksi terlebih dahulu dengan menggunakan metode lain.

Metode yang dipilih untuk meramalkan *independent variable* adalah metode *Double Exponential Smoothing*, *Holt Winter* dan *Box Jenkins* atau yang biasa dikenal ARIMA. Ketiga metode tersebut akan diuji coba pada sampel data yang sama setiap variabel. Metode yang bisa menghasilkan RMSE paling kecil,

digunakan untuk meramalkan variabel tersebut selama 24 periode kedepan. Data sampel yang akan digunakan dapat dilihat pada

**Tabel 4.4. Sampel Data untuk Uji Coba Metode Peramalan *Independent Variable***

<b>Variabel</b>	<b>Data</b>
Suhu	Stasiun BMKG Karangates
Kelembaban Udara	Stasiun BMKG Karangates
Curah Hujan	Stasiun BMKG Karangates
Kecepatan Angin	Stasiun BMKG Karangates
Angka Bebas Jentik	Data Kecamatan Bululawang
Jumlah Penduduk	Data Kecamatan Ngajum

## BAB V IMPLEMENTASI

Bab ini menjelaskan tentang implementasi metode *k-nearest neighbor* kedalam bahasa python dengan menggunakan *tools jupyter notebook*. Implementasi dilakukan sesuai dengan urutan langkah yang telah dijelaskan pada Bab III Metodologi. Rancangan kode implementasi sesuai dengan perancangan yang telah dijelaskan pada BAB IV Perancangan.

### 5.1. Persiapan implementasi

Penelitian ini dilakukan dengan menggunakan perangkat keras dan perangkat lunak. Spesifikasi perangkat keras yang digunakan dalam penelitian ini dapat dilihat pada Tabel 5.1.

**Tabel 5.1. Spesifikasi Perangkat Keras dan Lunak**

Perangkat Keras dan Perangkat lunak	Spesifikasi
Laptop	Acer E5-553G-114Q
<i>Processor</i>	AMD Quad-core Processor A12-9700P
<i>Memory</i>	8 GB
Sistem Operasi	Windows 10 Pro 64-bit (10.0, Build 10240)

Percobaan dilakukan dengan memanfaatkan beberapa teknologi yaitu bahasa pemrograman *python*, *code editor* dan beberapa *library* yang telah tersedia. Rincian dari teknologi yang digunakan untuk melakukan percobaan dapat dilihat pada Tabel 5.2.

Tabel 5.2. Lingkungan Uji Coba

Teknologi	Spesifikasi
Bahasa Pemrograman	Python 3.0.
<i>Code editor</i>	Jupyter Notebook
<i>Library</i>	Numpy Pandas KNeighborsRegressor mean_squared_error MinMaxScaler train_test_split matplotlib

## 5.2. Praproses Data

Tahap praproses data adalah tahap untuk mempersiapkan data sebelum pembangunan model. Langkah yang dilakukan pada tahap ini adalah menelaraskan periode pada data, memilih kecamatan untuk membentuk model, melakukan uji korelasi variabel observasi, mengatasi masalah *missing value*, normalisasi, *windowing* dan pembagian data menjadi data *training* dan data *testing*.

### 5.2.1. Menyelaraskan Periode pada Data

Periode pada data suhu, curah hujan, kelembaban udara, kecepatan angin adalah harian. Sedangkan, periode pada data jumlah penduduk, angka bebas jentik dan kejadian demam berdarah adalah bulanan. Sebelum masuk ke tahap pemodelan, data harusnya selaras dan memiliki periode yang sama. Langkah yang dilakukan untuk menelaraskan periode adalah mengubah data harian suhu, curah hujan, kelembaban udara dan kecepatan angin menjadi periode bulanan seperti yang telah dijelaskan pada sub-bab 4.3.1. Proses mengubah data harian menjadi bulanan menggunakan persamaan 5.1.

$$\text{Rata - rata} = \frac{1}{n} \sum_{1}^n a \quad (5.1)$$

Dimana  $n$  adalah jumlah hari pada bulan  $x$  dan  $a$  adalah data suhu/curah hujan/kelembaban udara/kecepatan angin.

### 5.2.2. Pemilihan Kecamatan Pembentuk Model

Setiap dataran akan dipilih satu kecamatan pembentuk model. Kecamatan pembentuk model dipilih dari rata-rata nilai korelasi kejadian demam berdarah di kecamatan  $a$  dengan kecamatan lainnya. Uji korelasi dilakukan dengan *python* dengan memanfaatkan *library* *pandas*. Kode untuk melakukan uji korelasi kejadian demam berdarah antar-kecamatan dapat dilihat pada Kode 5.1.

```
df_kdb.corr(method='pearson')
```

#### Kode 5.1. Uji korelasi kejadian demam berdarah antar kecamatan

Keterangan dari kode 5.1 terdapat pada Tabel 5.3.

Tabel 5.3. Penjelasan Kode 5.1.

Atribut	Penjelasan
df_kdb	Dataframe yang berisi data kejadian demam berdarah di setiap kecamatan pada satu dataran.
corr	Fungsi untuk uji korelasi
'pearson'	Metode uji korelasi

Rata-rata nilai korelasi dapat dilihat pada persamaan 5.2.

$$\bar{a} = \frac{1}{n} \sum_1^n ap + aq + \dots + n \quad (5.2)$$

Dimana  $\bar{a}$  adalah rata-rata nilai korelasi di kecamatan  $a$ ,  $ap/aq$  adalah nilai korelasi kecamatan  $a$  dan kecamatan  $p/q$  dan  $n$  adalah banyaknya kecamatan pada dataran tersebut.

### 5.2.3. Uji Korelasi Variabel Observasi

Uji korelasi variabel observasi dilakukan pada kecamatan terpilih atau kecamatan dengan nilai rata-rata korelasi tertinggi seperti yang telah dijelaskan pada sub-bab 4.4.2. Kode untuk melakukan uji korelasi dapat dilihat pada Kode 5.2.

```
dr.corr(method='pearson')
```

#### Kode 5.2. Uji korelasi variabel observasi dengan variabel dependent

Keterangan dari kode 5.2 terdapat pada Tabel 5.4.

Tabel 5.4. Penjelasan kode 5.2.

Atribut	Penjelasan
dr	Dataframe yang berisi data suhu, curah hujan, kelembaban udara, kecepatan angin, jumlah penduduk, angka bebas jentik, kejadian demam berdarah
corr	Fungsi untuk uji korelasi
'pearson'	Metode uji korelasi

### 5.2.4. Mengatasi Nilai yang Hilang (*Missing Value*)

Seperti yang telah dijelaskan pada sub-bab 4.3.2, langkah yang dilakukan untuk mengatasi masalah *missing value* adalah dengan mengisi *missing value* dengan nilai tertentu. Nilai untuk mengisi *missing value* diperoleh dari hasil rata-rata, hasil interpolasi dan hasil regresi.

#### 5.2.4.1. Mengisi *Missing Value* dengan *Mean*

Salah satu nilai untuk mengisi *missing value* adalah dari rata-rata dari *record* sebelumnya. Pada tugas akhir ini *missing value* pada data suhu, curah hujan, kelembaban udara dan kecepatan angin diisi dengan nilai rata-rata dari *record* sebelumnya pada data yang sama. Pengisian nilai *missing value* dilakukan di



*microsoft excel*. Persamaan yang digunakan untuk mencari nilai rata dapat dilihat pada persamaan 5.3.

$$x_n = \frac{1}{n-1} \sum_1^{n-1} x_n \quad (5.3)$$

Dimana  $x_n$  adalah nilai untuk mengisi *missing value* pada periode ke- $n$ . Nilai  $n$  adalah tahun pada bulan yang sama.

#### 5.2.4.2. Mengisi *Missing Value* dengan Interpolasi

Mengisi nilai *missing value* menggunakan interpolasi seperti yang telah dijelaskan pada sub-bab 4.3.2 poin b dilakukan dengan Kode 5.3. *Library* yang digunakan adalah *scipy*.

```
df_all['ABJ'].interpolate(method='linear',
limit_direction='both')
```

**Kode 5.3.** Mengisi *missing value* dengan nilai interpolasi

Keterangan kode 5.3. Dapat dilihat pada Tabel 5.5.

**Tabel 5.5.** Penjelasan kode 5.3.

Atribut	Penjelasan
df_all['ABJ']	Dataframe ABJ adalah data yang mengandung <i>missing value</i> dan akan dilakukan interpolasi.
.interpolate	Memanggil fungsi interpolasi
'linear'	Metode interpolasi yang akan dilakukan
Limit_direction='both'	Memungkinkan interpolasi data yang kosong pada awal <i>record</i> .

#### 5.2.4.3. Mengisi *Missing Value* dengan Regresi

Mengisi *missing value* dengan nilai regresi seperti yang dijelaskan pada sub-bab 4.3.2 poin c dapat dilihat pada kode .

```
#import library
```

```

from sklearn.linear_model import LinearRegression
linreg = LinearRegression()
data = wajak_reg_abj2

#Langkah-1:pembagian data
x_train =
data[data['ABJ'].notnull()].drop(columns=['ABJ'])
y_train = data[data['ABJ'].notnull()]['ABJ']
x_test =
data[data['ABJ'].isnull()].drop(columns=['ABJ'])
y_test = data[data['ABJ'].isnull()]['ABJ']

#Langkah-2: pembentukan model
linreg.fit(x_train, y_train)

#Langkah-3: prediksi nilai missing value
predicted = linreg.predict(x_test)

#Langkah-4: mengkombinasikan dataset dengan nilai
prediksi missing value
wajak_reg_abj2.ABJ[wajak_reg_abj2.ABJ.isnull()] =
predicted

```

#### Kode 5.4. Mengisi missing value dengan regresi

Keterangan kode 5.4. dapat dilihat pada Tabel 5.6.

**Tabel 5.6. Penjelasan kode 5.4.**

Atribut	Penjelasan
LinearRegression()	Untuk memanggil fungsi regresi linear
Wajak_reg_abj2	Dataset kecamatan wajak yang berisi kolom SU, CH, KU, KA, ABJ, JP, KDB
x_train	Dataset kecamatan wajak (kecuali kolom ABJ) yang tidak memiliki <i>missing value</i> pada nilai ABJ.
y_train	Dataset kolom ABJ yang tidak kosong di kecamatan Wajak
x_test	Dataset kecamatan wajak (kecuali kolom ABJ) yang memiliki <i>missing value</i> pada nilai ABJ.

y_test	Dataset kolom ABJ yang kosong atau yang ingin diprediksi nilainya di kecamatan Wajak
.drop	Untuk memanggil fungsi yang menghapus kolom
.isnull	Untuk memanggil fungsi yang mengecek apakah terdapat nilai null atau tidak. Bila terdapat null maka pengembalian nilai boolean:TRUE.
.fit	Fungsi untuk memasang model yang akan dibangun dengan data <i>training</i>
.predict	Fungsi untuk memprediksi nilai dari model yang bergasil dibangun.

### 5.2.5. Normalisasi

Normalisasi dilakukan dengan menggunakan *min-max* seperti yang telah dijelaskan pada sub-bab 4.3.3. Normalisasi dengan *min-max* memanfaatkan *library sklearn* dan dapat dilihat pada kode 5.5.

```
scaler = MinMaxScaler(feature_range=(0,1))
scaler.fit(df)
```

**Kode 5.5. Normalisasi min-max**

Keterangan pada kode 5.5. dapat dilihat pada Tabel 5.7. Penjelasan kode 5.5.

**Tabel 5.7. Penjelasan kode 5.5**

Atribut	Penjelasan
Scaler	Nama untuk model normalisasi/penskalaan
<i>MinMaxScaler</i>	Fungsi untuk membentuk model penskalaan
Feature_range=(0,1)	Rentang nilai keluaran memiliki skala 0 sampai 1

.fit	Fungsi untuk memasang model penskalaan dengan dataset yang akan dinormalisasi
Df	Nama dataset yang akan dinormalisasi

### 5.2.6. *Windowing*

Proses *windowing* seperti yang telah dijelaskan pada sub-bab 4.3.4 di implementasikan pada kode Kode 5.6.

```
series_to_supervised(df_scaled, 3, 1)
```

#### Kode 5.6. Memanggil fungsi *windowing*

Keterangan pada kode 5.6. Dapat dilihat pada Tabel 5.8.

Tabel 5.8. Penjelasan kode 5.6.

Atribut	Penjelasan
Series_to_supervised()	Untuk memanggil fungsi <i>windowing</i>
df_scaled	Dataframe berbentuk <i>time series</i> yang akan diubah bentuknya menjadi <i>cross-sectional</i> data
'3'	Banyak variabel <i>lag</i> yang ingin dibentuk.
'1'	<i>Lag</i> adalah nilai keterlambatan atau nilai pada periode sebelumnya. Angka '1' disini atau yang disebut <i>n_out</i> adalah kebalikan dari <i>lag</i> ata nilai periode sesudahnya.

Isi dari fungsi *series\_to\_supervised( )* dapat dilihat pada kode 5.7.

```
def series_to_supervised(data, n_in=1, n_out=1,
                          dropnan=True):
    n_vars = 1 if type(data) is list else
    data.shape[1]
    dff = pd.DataFrame(data)
    cols, names = list(), list()
```

```

# input sequence (t-n, ... t-1)
for i in range(n_in, 0, -1):
    cols.append(dff.shift(i))
    names += [('var%d(t-%d)' % (j+1, i))
for j in range(n_vars)]
# forecast sequence (t, t+1, ... t+n)
for i in range(0, n_out):
    cols.append(dff.shift(-i))
    if i == 0:
        names += [('var%d(t)' %
(j+1)) for j in range(n_vars)]
    else:
        names += [('var%d(t+%d)' %
(j+1, i)) for j in range(n_vars)]
# put it all together
agg = pd.concat(cols, axis=1)
agg.columns = names
# drop rows with NaN values
if dropana:
    agg.dropna(inplace=True)
return agg

```

**Kode 5.7. Fungsi *series\_to\_supervised***

### 5.2.7. Pembagian data

Data perlu dibagi menjadi *training data* dan *testing data* dengan variasi proporsi seperti yang telah dijelaskan pada sub-bab 4.6. Implementasi pembagian data dapat dilihat pada Kode 5.8.

```

train_x, test_x, train_y, test_y =
train_test_split(df_lag3.iloc[:, :4], df_lag3.KDB, t
est_size = 0.2, shuffle = False)

```

**Kode 5.8. Pembagian data training dan test data**

Keterangan Kode 5.8 dapat dilihat pada

**Tabel 5.9. Penjelasan kode 5.8.**

Atribut	Penjelasan
---------	------------

<code>train_test_split( )</code>	Fungsi untuk membagi data menjadi <i>train_x</i> , <i>test_x</i> , <i>train_y</i> , <i>test_y</i>
<code>Df_lag3</code>	Dataframe yang akan dibagi menjadi <i>training</i> dan <i>test</i>
<code>[:,:4]</code>	4 kolom pertama menjadi <i>x</i>
<code>KDB</code>	Kolom 'KDB' menjadi <i>y</i>
<code>Test_size = 0.2</code>	Banyaknya data yang menjadi <i>test data</i>
<code>Shuffle = False</code>	Pembagian data dilakukan secara berurutan atau tidak ada pengacakan.

### 5.3. Pembentukan Model

Pembentukan model dengan metode *k-nearest neighbor regression* pada *training data* dilakukan dengan kombinasi skenario seperti yang telah dijelaskan pada sub-bab 4.4. Implementasi memanfaatkan *library KneighborsRegressor* dari Sklearn dan dapat dilihat pada Kode 5.9.

```
#Pembentukan model
knn_regressor =
KNeighborsRegressor(n_neighbors=7,algorithm="brute",weights="distance", metric="manhattan")
knn_regressor.fit(train_x, train_y)

#Prediksi
train_pred = knn_regressor.predict(train_x)
test_pred = knn_regressor.predict(test_x)
```

**Kode 5.9. Pembentukan model KNNR**

Keterangan Kode 5.9. dapat dilihat pada Tabel 5.10.

**Tabel 5.10. Penjelasan kode 5.9.**

Atribut	Penjelasan
<code>knn_regressor</code>	Nama model <i>KNNR</i>
<code>KneighborsRegressor( )</code>	Fungsi untuk membangun model <i>KNNR</i>
<code>n_neighbors=7</code>	Parameter jumlah titik yang menjadi tetangga terdekat

algorithm="brute"	Parameter algoritma yang dipilih untuk mencari titik terdekat
weights="distance"	Parameter pembobotan yang dipilih untuk memprediksi nilai target
train_pred	Prediksi nilai di <i>training data</i>
test_pred	Prediksi nilai di <i>test data</i>

#### 5.4. Denormalisasi

Masukan untuk membentuk model berbentuk normal atau skala sehingga keluaran model juga merupakan angka normalisasi atau skala. Untuk melakukan uji kelayakan angka normalisasi atau skala tersebut harus didenormalisasi. Implementasi denormalisasi dapat dilihat pada Kode 5.10.

```
scaler_y.inverse_transform(test_pred)
```

**Kode 5.10. Denormalisasi**

Keterangan untuk Kode 5.10 dapat dilihat pada Tabel 5.11.

**Tabel 5.11. Penjelasan kode 5.10.**

Atribut	Penjelasan
scaler_y	Nama <i>scaler</i> untuk normalisasi di tahap sebelumnya
inverse_transform()	Fungsi untuk mengembalikan nilai asli atau nilai yang bukan skala
test_pred	Nama data yang akan didenormalisasi

#### 5.5. Uji Kelayakan

Uji kelayakan dilakukan dengan hasil denormalisasi prediksi pada *test data*. Terdapat dua persamaan yang digunakan untuk menguji kelayakan model, yaitu RMSE dan SMAPE.

##### 5.5.1. RMSE

Seperti yang telah dijelaskan pada sub-bab 4.7, salah satu persamaan yang digunakan untuk menguji model adalah RMSE.

Nilai *error* RMSE didapatkan dari akar kuadrat perbedaan nilai aktual dan prediksi pada *testing data*. Implementasi pengujian model dengan RMSE dapat dilihat pada Kode 5.11.

```
np.sqrt(mean_squared_error(real, test_pred_denorm))
```

**Kode 5.11. Uji kelayakan dengan RMSE**

Keterangan untuk Kode 5.11 Dapat dilihat pada Tabel 5.12.

**Tabel 5.12. Penjelasan kode 5.11.**

Atribut	Penjelasan
np.sqrt()	Memanggil fungsi akar kuadrat dari numpy
mean_squared_error() )	Fungsi untuk mencari nilai MSE ( <i>Mean Squared Error</i> )
Real	Nilai aktual
Test_pred_denorm	Nilai prediksi yang telah di denormalisasi

### 5.5.2. SMAPE

Seperti yang telah dijelaskan pada sub-bab 4.7, SMAPE juga merupakan salah satu persamaan untuk menguji model. Implementasi pengujian model dengan menggunakan SMAPE dapat dilihat pada Kode 5.12.

```
#Mendefinisikan fungsi SMAPE
def smape(A, F):
    return 100/len(A) * (np.sum(np.abs(F - A)
    / (np.abs(A) + np.abs(F))))

#memanggil fungsi SMAPE
smape(real, test_pred_denorm)
```

**Kode 5.12. Uji kelayakan model dengan SMAPE**

Keterangan untuk Kode 5.12 dapat dilihat pada Tabel 5.13.

**Tabel 5.13. Penjelasan kode 5.12.**



Atribut	Penjelasan
len(A)	Mengembalikan banyaknya data pada data A (Actual)
np.sum()	Fungsi penjumlahan dari np ( <i>library numpy</i> )
np.abs()	Fungsi untuk mengembalikan nilai absolut
smape()	Untuk memanggil fungsi smape
'A'	Actual atau nilai asli
'F'	Forecast atau hasil prediksi

## 5.6. Peramalan 24 Periode

Peramalan dilakukan seperti yang telah dijelaskan pada sub-bab 4.8. Model yang hanya melibatkan variabel lag dapat diramalkan secara langsung. Implementasi model yang hanya melibatkan variabel *lag* dapat dilihat pada Kode 5.13.

```
#inisiasi dataframe baru
dff = pd.DataFrame(np.nan, index=range(0,24),
columns=['lag1', 'fc'])
dff.lag1[0] = scaled_kdb.tail(1).KDB_norm
#Prediksi dengan model terpilih
for i in range(0,24):
    pred =
pd.DataFrame(knn_regressor.predict(dff.iloc[:(i+1)
],:1]),columns=['p'])
    dff.fc[i] = pred.p[i]
    dff.lag1[(i+1)] = pred.p[i]
#denormalisasi
forecast = pd.DataFrame(dff['fc'])
forecast_denorm =
scaler_y.inverse_transform(forecast).round()
```

**Kode 5.13. Prediksi dengan model yang hanya melibatkan variabel lag**

Keterangan untuk Kode 5.13 dapat dilihat pada Tabel 5.14.

**Tabel 5.14. Penjelasan kode 5.13.**

Atribut	Penjelasan
dff	Nama dataframe hasil forecast
scaled_kdb	Nama dataframe data aktual yang sudah di normalisasi
.tail()	Fungsi untuk memanggil baris terakhir pada dataframe
KDB_norm	Nama kolom pada dataframe aktual
For	Fungsi untuk melakukan pengulangan
Range(0,24)	Fungsi untuk mengembalikan rentang
Knn_regressor	Nama model terpilih untuk peramalan
.predict()	Fungsi untuk mengembalikan hasil prediksi
.iloc[ ]	Fungsi untuk memanggil nilai dari <i>cell</i> tertentu pada dataframe
Scaler_y	Scaler normalisasi
.inverse_transform()	Fungsi untuk mengembalikan nilai asli dari nilai normalisasi
.round()	Fungsi untuk membulatkan nilai

Model yang melibatkan variabel observasi sebagai *independent variable* memiliki syarat tambahan untuk melakukan peramalan. Variabel observasi yang akan dijadikan *independent variable* harus sudah diramalkan terlebih dahulu.

### 5.6.1. Peramalan dengan *Double Exponential Smoothing*

```
#import library
from statsmodels.tsa.api import
ExponentialSmoothing, Holt

#pembentukan model
fit1 =
Holt(np.asarray(train['JP'])).fit(smoothing_level
= 0.8,smoothing_slope = 0.8)

#prediksi
y_hat_avg['Holt_linear'] =
fit1.forecast(len(test))
```

**Kode 5.14.** Peramalan Variabel Independen dengan *Double Exponential Smoothing*

Keterangan pada Kode 5.14 dapat dilihat pada Tabel 5.15.

**Tabel 5.15. Penjelasan Kode 5.14**

<b>Atribut</b>	<b>Penjelasan</b>
Fit1	Model yang dibangun dengan metode <i>Double Exponential Smoothing</i>
Holt()	Fungsi untuk membentuk model
Np.asarray()	Fungsi untuk mengubah dataframe menjadi <i>array</i>
Train['JP']	Dataframe Train kolom 'JP' adalah data yang digunakan untuk <i>training</i> model
.fit	Fungsi untuk mengimplementasikan pembangunan model dengan parameter tertentu
Smoothing_level	Parameter untuk membentuk model <i>double exponential smoothing</i> yang berisi nilai $\alpha$
0.8	Nilai parameter <i>smoothing_level</i> atau nilai $\alpha$ dan $\beta$
Smoothing_slope	Parameter untuk membentuk model <i>double exponential smoothing</i> yang berisi nilai $\beta$
.forecast()	Fungsi untuk menghasilkan peramalan
Len()	Fungsi yang mengembalikan jumlah suatu data
Test	Dataframe yang berisi <i>testing data</i>

### 5.6.2. Peramalan dengan *Holt Winter*

```
#Pembentukan model
fit1 =
ExponentialSmoothing(np.asarray(train['SU'])
,seasonal_periods=12 ,trend='add',
seasonal='add',).fit()

#Menghasilkan prediksi
y_hat_avg['Holt_Winter'] =
fit1.forecast(len(test))
```

**Kode 5.15. Peramalan *Independent Variable* dengan Metode *Holt Winter***

Keterangan untuk Kode 5.15 dapat dilihat pada Tabel 5.16.

**Tabel 5.16. Penjelasan Kode 5.15.**

<b>Atribut</b>	<b>Penjelasan</b>
Fit1	Model yang dibangun dengan metode <i>Double Exponential Smoothing</i>
ExponentialsMOOTHING( )	Fungsi untuk membentuk model
Np.asarray( )	Fungsi untuk mengubah dataframe menjadi <i>array</i>
Train[‘SU’]	Dataframe Train kolom ‘SU’ adalah data yang digunakan untuk <i>training</i> model
.fit	Fungsi untuk mengimplementasikan pembangunan model dengan parameter tertentu
Seasonal_period	Parameter untuk membentuk model <i>Winter exponential smoothing</i> yang berisi nilai panjang 1 <i>season</i>
12	Nilai parameter <i>seasonal_period</i> (12 periode)
trend	Parameter untuk membentuk model <i>Winter exponential smoothing</i>
seasonal	Parameter untuk membentuk model <i>Winter exponential smoothing</i>
Add	Metode yang dipilih adalah <i>additive</i>
.forecast( )	Fungsi untuk menghasilkan peramalan
Len( )	Fungsi yang mengembalikan jumlah suatu data
Test	Dataframe yang berisi <i>testing data</i>

### **5.6.3. Peramalan dengan *Box Jenkins* atau ARIMA**

Peramalan variabel observasi selama 24 periode menggunakan metode ARIMA dilakukan dengan Rstudio. *Library* yang dimanfaatkan adalah *library forecast*. Implementasi peramalan untuk variabel observasi dengan Rstudio dapat dilihat pada Kode 5.16.

```
#memuat data variabel observasi
jp_ngajum <- read_excel("C:/Users/Berta/TUGAS
AKHIR/input/JP Ngajum.xlsx")
jp_ngajum.ts <- ts(jp_ngajum, start = c(2010,1),
frequency = 12)

#membuat model peramalan
model = auto.arima(jp_ngajum.ts, trace=TRUE)

#menghasilkan peramalan
peramalan = forecast(model,24)
```

**Kode 5.16. Peramalan *Independent Variable* dengan Metode Box Jenkins**

Keterangan Kode 5.16 dapat dilihat pada Tabel 5.17.

**Tabel 5.17. Penjelasan Kode 5.16.**

<b>Atribut</b>	<b>Penjelasan</b>
jp_ngajum	Dataframe yang berisi variabel observasi jumlah penduduk di kecamatan Ngajum
read_excel()	Fungsi untuk membaca file excel
.ts	Fungsi yang mengubah dataframe menjadi dataframe <i>timeseries</i>
model	Nama model yang dibangun dengan metode ARIMA
auto.arima	Fungsi yang secara otomatis mencari dan menghasilkan model terbaik
forecast()	Fungsi yang mengembalikan hasil peramalan

Hasil peramalan variabel observasi menjadi masukan untuk menghasilkan permalan. Hasil permalaman variabel observasi harus dinormalisasi untuk menjadi masukan model peramalan. Implementasi peramalan dengan model yang melibatkan variabel observasi sebagai *independent variable* dapat dilihat pada Kode 5.17.

```

#inisialisasi dataframe untuk peramalan
dff = pd.DataFrame(np.nan, index=range(0,24),
columns=['CH', 'JP', 'lag2', 'lag1', 'fc'])
dff.lag2[0] = df2.KDB[106]
dff.lag1[0] = df2.KDB[107]
dff.CH = pred_CH_ngajum
dff.JP = pred_JP_ngajum

#Peramalan dengan model terpilih
for i in range(0,24):
    pred =
pd.DataFrame(knn_regressor.predict(dff.iloc[:(i+1
), :4]), columns=['p'])
    dff.fc[i] = pred.p[i]
    dff.lag1[(i+1)] = pred.p[i]
    dff.lag2[(i+1)] = dff.lag1[i]

#denormalisasi
forecast = pd.DataFrame(dff['fc'])
forecast_denorm =
pd.DataFrame(scaler_y.inverse_transform(forecast)
.round(), columns=['forecast'])

```

**Kode 5.17. Peramalan dengan model yang melibatkan variabel observasi**

Keterangan pada Kode 5.17 dapat dilihat pada Tabel 5.18.

**Tabel 5.18. Penjelasan Kode 5.17.**

<b>Atribut</b>	<b>Penjelasan</b>
dff	Dataframe peramalan
df2	Dataframe aktual
pred_CH_ngajum	Hasil prediksi curah hujan di kecamatan Ngajum
pred_JP_ngajum	Hasil predisi jumlah penduduk di Kecamatan Ngajum
knn_regressor	Model terpilih untuk peramalan
.predict()	Fungsi untuk mengembalikan hasil prediksi
.iloc[ ]	Fungsi untuk memanggil <i>cell</i> tertentu pada dataframe

scaler_y	Scaler normalisasi
.inverse_transform( )	Fungsi untuk mengembalikan nilai asli dari nilai normalisasi
.round( )	Fungsi untuk membulatkan nilai

## BAB VI HASIL DAN PEMBAHASAN

Bab ini berisi hasil dari percobaan yang telah dilakukan menggunakan metode *k-nearest neighbor regression* seperti yang telah dijelaskan pada bab V implementasi. Hasil dari setiap percobaan akan dibahas secara rinci dan dianalisis.

### 6.1. Hasil Praproses Data

Hasil dari praproses data adalah dataset yang siap untuk pembentukan model. Berikut adalah hasil dari setiap langkah praproses yang telah dilakukan.

#### 6.1.1. Penyelarasan Periode

Data dari BMKG memiliki periode harian (dapat dilihat pada Lampiran B Tabel D.1) diubah menjadi periode bulanan dengan persamaan 5.1. Sampel data cuaca dengan periode bulan dapat dilihat pada Tabel 6.1.

**Tabel 6.1. Sampel Dataset Cuaca Periode Bulan**

<b>Periode</b>	<b>SU</b>	<b>KU</b>	<b>CH</b>	<b>KA</b>
Jan-10	26.02	84.74	10.52	0.84
Feb-10	26.32	84.64	12.04	0.86
Mar-10	26.39	85	8.65	0.55
Apr-10	26.29	86.17	10.37	0.3
May-10	26.57	85.61	7.35	0.58
Jun-10	25.62	84.1	9.23	1.59
Jul-10	25.35	83.52	2.16	1.77
Aug-10	25.35	81.71	3.94	1.68
Sep-10	25.73	83.93	9.4	1.37
Oct-10	26.19	82.03	12.13	1.71

#### 6.1.2. Mengatasi *Missing Value*

Mengatasi masalah *missing value* dibagi menjadi dua tahap. Tahap pertama adalah mengidentifikasi *missing value* untuk



menentukan teknik yang digunakan. Tahap kedua adalah penyelesaian masalah *missing value* dengan teknik yang telah ditentukan.

### 6.1.2.1. Identifikasi *Missing Value*

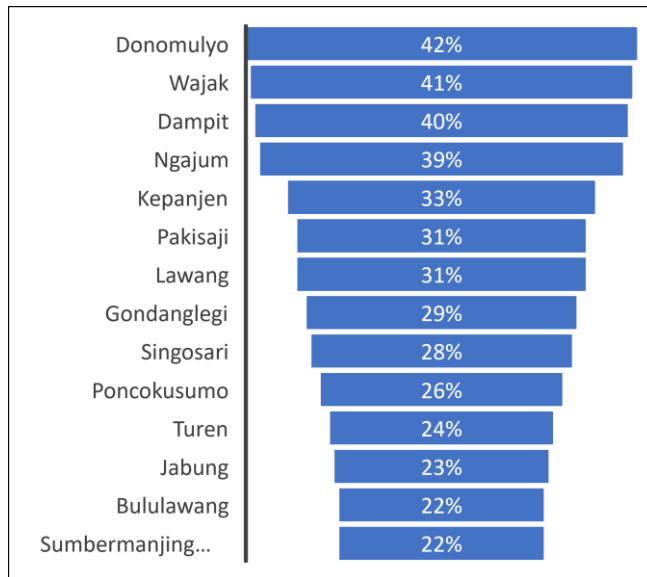
Langkah yang dilakukan untuk mengatasi masalah *missing value* disesuaikan dengan banyaknya *missing value* dan letak *missing value* tersebut. Banyaknya *missing value* pada variabel cuaca dapat dilihat pada Tabel 6.2 sedangkan pada variabel ABJ dapat dilihat pada Gambar 6.1.

**Tabel 6.2. Banyaknya *missing value* pada Data Cuaca**

Data	Banyaknya <i>missing value</i>	
	Stasiun BMKG Karangkates	Stasiun BMKG Karangploso
Suhu	8.3 %	0 %
Curah Hujan	8.3 %	0 %
Kelembaban Udara	17.59 %	0 %
Kecepatan Angin	4.6 %	0 %

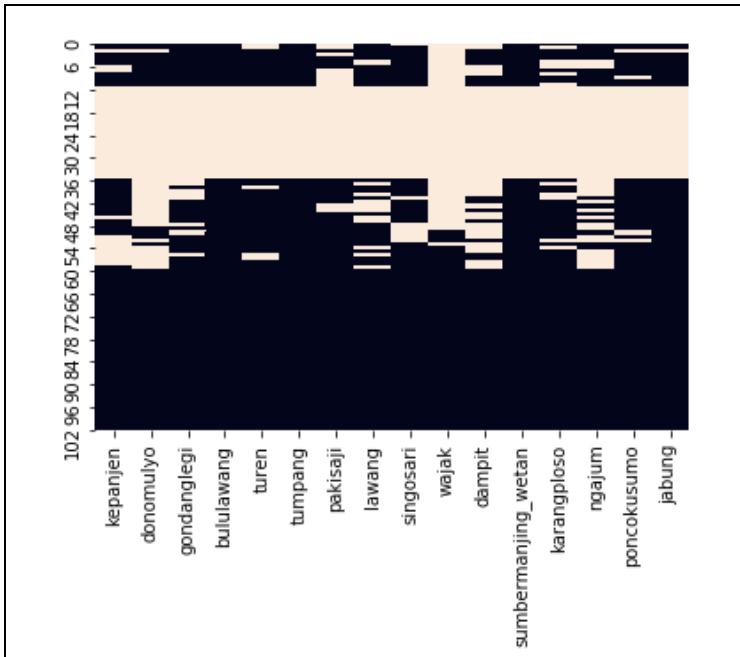
Dapat diketahui pada Tabel 6.2 bahwa *missing value* hanya terdapat pada data yang diambil dari stasiun BMKG Karangkates. Banyaknya *missing value* di semua data kurang dari 20% sehingga langkah yang digunakan untuk mengatasi *missing value* tersebut adalah menghitung nilai rata-rata dari data di periode sebelumnya. Data cuaca hasil praproses dapat dilihat pada Tabel D.2.

Banyaknya *missing value* pada data Angka Bebas Jentik (ABJ) tiap kecamatan dapat dilihat pada Gambar 6.1. Langkah yang dilakukan untuk mengatasi *missing value* dengan jumlah dibawah 40% adalah interpolasi.



**Gambar 6.1. Banyaknya Missing Value pada Data Angka Bebas Jentik**

Data ABJ di Kecamatan Donomulyo, Wajak dan Dampit memiliki *missing value* diatas 40% dari keseluruhan data. Pada Gambar 6.2) dapat dilihat bahwa letak *missing value* di Kecamatan Wajak muncul secara berurutan dan dimulai dari periode pertama. Sedangkan, *missing value* di Kecamatan Donomulyo dan Kecamatan Dampit tidak muncul berurutan. Melakukan interpolasi masih bisa dilakukan untuk mengatasi masalah *missing value* di Kecamatan Donomulyo dan Dampit. Untuk Kecamatan Wajak, langkah yang dilakukan untuk mengatasi *missing value* adalah dengan regresi linier.



**Gambar 6.2.** Matrik Letak Missing Value pada Variabel ABJ

Gambar 6.2 menampilkan sebaran *missing value* pada variabel ABJ di setiap kecamatan. Warna cerah pada gambar adalah *missing value*. Sumbu X adalah periode sedangkan sumbu Y adalah kecamatan.

### 6.1.2.2. Hasil Penyelesaian Masalah *Missing Value*

Permasalahan *missing value* diselesaikan dengan cara mengisi *missing value* dengan nilai tertentu. Nilai untuk mengisi *missing value* tersebut didapatkan dari tiga teknik berbeda yaitu dengan nilai rata-rata dari periode sebelumnya, dengan nilai interpolasi dan dengan nilai regresi linier.

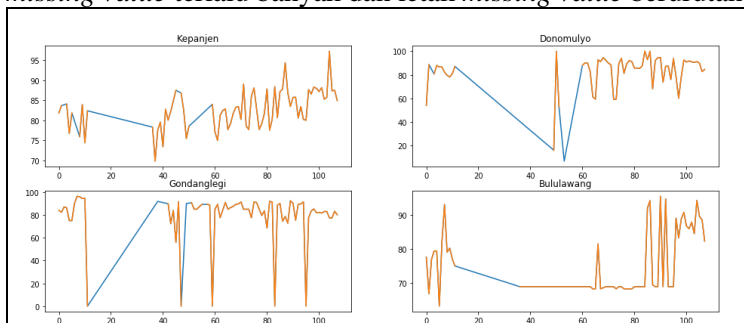
Teknik mengisi *missing value* dengan rata-rata di periode sebelumnya hanya dilakukan pada data cuaca. Sampel data cuaca hasil praproses *missing value* dapat dilihat pada Tabel 6.3.

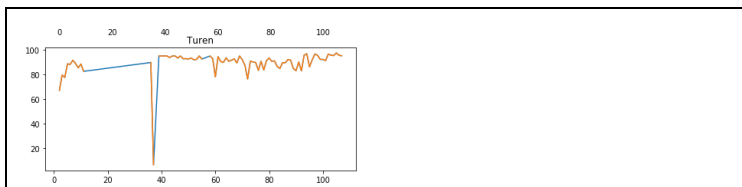
**Tabel 6.3. Sampel Data Cuaca Hasil Praproses *Missing Value***

Periode	SU	KU	CH	KA
Jan-10	26.02	84.74	10.52	0.84
Feb-10	26.32	84.64	12.04	0.86
Mar-10	26.39	85	8.65	0.55
Apr-10	26.29	86.17	10.37	0.3
May-10	26.57	85.61	7.35	0.58
Jun-10	25.62	84.1	9.23	1.59
Jul-10	25.35	83.52	2.16	1.77
Aug-10	25.35	81.71	3.94	1.68
Sep-10	25.73	83.93	9.40	1.37
Oct-10	26.19	82.03	12.13	1.71
Nov-10	26	84.53	17.53	1.23

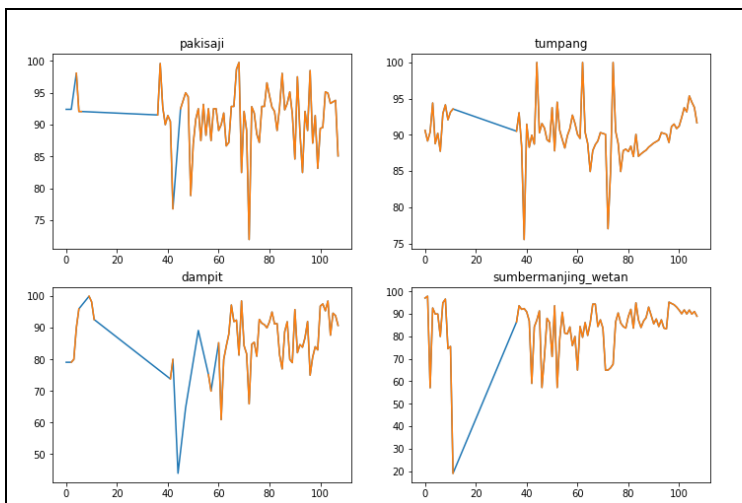
Hasil praproses *missing value* dengan teknik interpolasi dapat dilihat pada Gambar 6.3, Gambar 6.4, dan Gambar 6.5. Garis berwarna biru menunjukkan hasil dari pengisian *missing value* dengan nilai interpolasi.

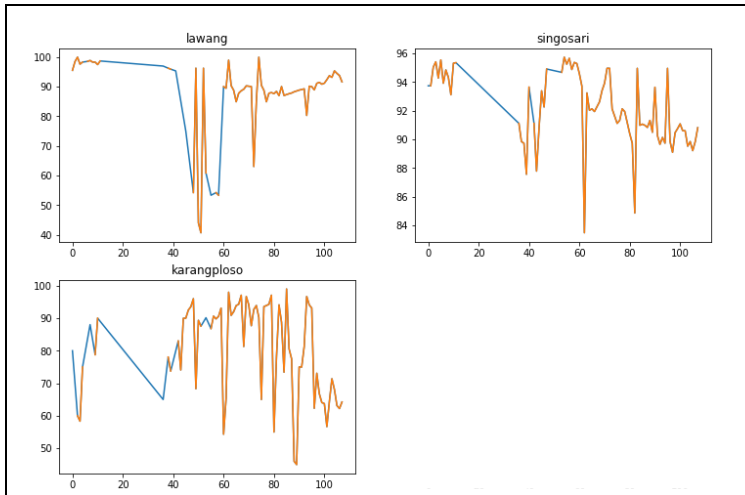
Teknik interpolasi menghasilkan garis lurus yang diambil dari titik sebelum muncul *missing value* dan titik setelah *missing value*. Teknik interpolasi sebaiknya tidak digunakan apabila *missing value* terlalu banyak dan letak *missing value* berurutan.



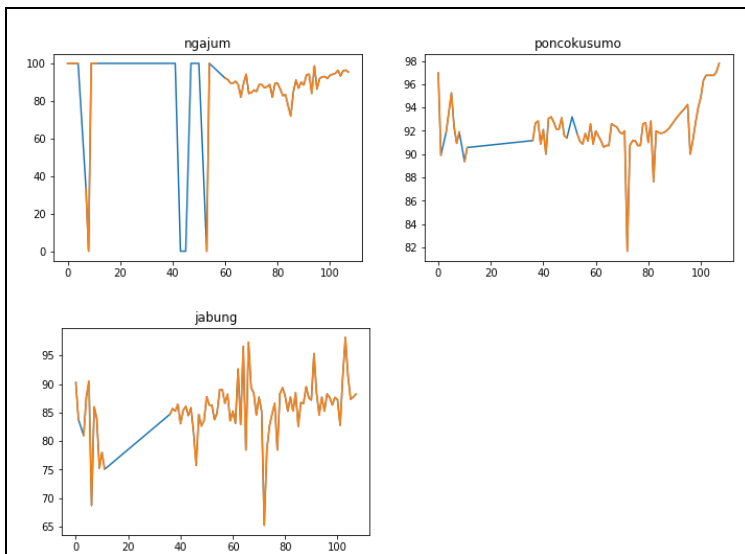


**Gambar 6.3. Hasil Praproses Missing Value Variabel ABJ di Dataran Rendah**



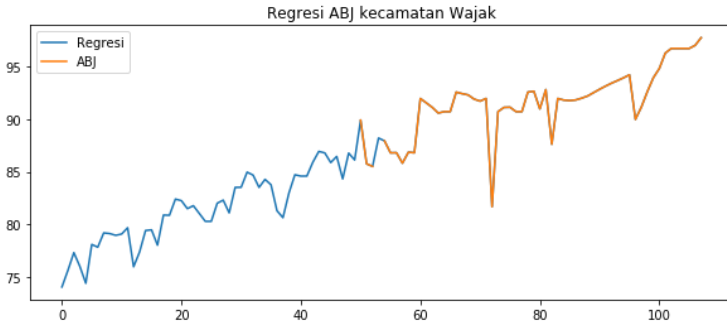


**Gambar 6.4. Hasil Praproses Missing Value Variabel ABJ di Dataran Sedang**



**Gambar 6.5. Hasil Praproses Missing Value Variabel ABJ di Dataran Tinggi**

Praproses *missing value* dengan teknik regresi linier dilakukan untuk variabel ABJ kecamatan Wajak. Hasil praproses *missing value* dengan teknik regresi linier dapat dilihat pada Gambar 6.6. Garis biru merupakan hasil praproses *missing value* dengan regresi linier.



**Gambar 6.6. Hasil Praproses *Missing Value* variabel ABJ di Kecamatan Wajak**

**6.2. Hasil Uji Korelasi untuk Memilih Kecamatan Pembentuk Model**

Kecamatan pembentuk model adalah kecamatan yang memiliki rata-rata nilai korelasi dengan kecamatan lain yang paling tinggi. Hasil dari uji korelasi dapat dilihat pada Tabel 6.4.

**Tabel 6.4. Nilai rata-rata Uji Korelasi tiap Kecamatan**

<b>Kecamatan</b>	<b>Rata-rata Nilai Korelasi</b>
<b>Dataran Rendah</b>	
Kepanjen	0.71
Donomulyo	0.57
Gondanglegi	0.60
Bululawang	0.54
Turen	0.64
<b>Dataran Sedang</b>	

Tumpang	0.38
Pakisaji	0.48
Lawang	0.40
Singosari	0.35
Wajak	0.45
Dampit	0.46
Sumbermanjing Wetan	0.45
Karangploso	0.34
<b>Dataran Tinggi</b>	
Ngajum	0.72
Poncokusumo	0.66
Jabung	0.59

### 6.2.1. Kecamatan Pembentuk Model di Dataran Rendah

Pada Tabel 6.4 dapat dilihat bahwa nilai rata-rata korelasi data KDB antar kecamatan di dataran rendah yang paling tinggi adalah Kecamatan Kepanjen dengan nilai 0.71. Kecamatan Kepanjen dipilih menjadi kecamatan pembentuk model karena dianggap yang paling berkorelasi dengan kecamatan lainnya di dataran rendah. Hasil uji korelasi data KDB antar kecamatan di dataran rendah dapat dilihat pada Tabel A.1

### 6.2.2. Kecamatan Pembentuk Model di Dataran Sedang

Pada Tabel 6.4 dapat diketahui bahwa nilai rata-rata korelasi KDB antar-kecamatan yang paling tinggi di dataran sedang adalah Kecamatan Pakisaji dengan nilai 0.48. Data di Kecamatan Pakisaji akan digunakan untuk membentuk model peramalan di dataran sedang karena dianggap sebagai kecamatan yang paling berkorelasi dengan kecamatan lainnya di dataran sedang. Hasil uji korelasi data KDB antar kecamatan di dataran sedang dapat dilihat pada Tabel A.2.

### 6.2.3. Kecamatan Pembentuk Model di Dataran Tinggi

Pada Tabel 6.4 dapat diketahui bahwa nilai rata-rata korelasi KDB antar-kecamatan yang paling tinggi di dataran tinggi adalah Kecamatan Ngajum dengan nilai 0.72. Data di



Kecamatan Ngajum akan digunakan untuk membentuk model peramalan di dataran tinggi karena dianggap sebagai kecamatan yang paling berkorelasi dengan kecamatan lainnya di dataran tinggi. Hasil uji korelasi data KDB antar kecamatan di dataran sedang dapat dilihat pada Tabel A.3.

### **6.3. Hasil Uji Korelasi Variabel Observasi dengan Kasus Demam Berdarah**

Variabel observasi akan dilibatkan dalam skenario *independent variable* pada proses pembentukan model. Hasil uji korelasi setiap variabel observasi dengan KDB dapat dilihat pada Tabel 6.5, Tabel 6.6, Tabel 6.7. Peringkat hasil uji korelasi berdasarkan dari nilai absolut. Tanda positif (+) mengindikasikan bahwa korelasi memiliki arah yang linier. Sedangkan, tanda negatif (-) pada hasil korelasi mengindikasikan bahwa arah korelasi tersebut berlawanan atau berbanding terbalik.

#### **6.3.1. Hasil Uji Korelasi Variabel Observasi di Dataran Rendah**

Hasil uji korelasi variabel observasi dan variabel KDB di dataran rendah dapat dilihat pada Tabel 6.5. Variabel jumlah penduduk (JP) memiliki korelasi tertinggi dengan variabel KDB dengan nilai 0.276 dan arah korelasi tersebut berlawanan. Variabel jumlah penduduk akan dilibatkan menjadi *independent variable* pada skenario V1. Peringkat kedua uji korelasi di dataran rendah adalah variabel Kelembaban Udara (KU) dengan nilai korelasi 0.235 dan arah korelasi linier. Variabel JP dan KU akan dilibatkan menjadi *independent variable* pada skenario V2. Peringkat ketiga nilai korelasi di dataran rendah adalah Curah Hujan (CH) dengan nilai 0.205 dan arah korelasi linier. Variabel JP, KU, dan CH akan dilibatkan menjadi *independent variable* pada skenario V3.

**Tabel 6.5. Uji Korelasi Variabel Observasi dan KDB di Dataran Rendah**

Variabel Observasi	Korelasi dengan KDB
Jumlah Penduduk	-0.276
Kelembaban Udara	0.235
Curah Hujan	0.205
Kecepatan Angin	0.125
Angka Bebas Jentik	-0.108
Suhu	0.051

Keenam variabel observasi tidak memiliki nilai yang cukup tinggi untuk dapat dikategorikan berkorelasi kuat. Keenam variabel observasi akan dilibatkan menjadi *variable independent* pada skenario VA.

### 6.3.2. Hasil Uji Korelasi Variabel Observasi di Dataran Sedang

Hasil uji korelasi variabel observasi dengan variabel KDB dapat dilihat pada Tabel 6.6. Peringkat satu nilai korelasi pada dataran sedang adalah variabel Angka Bebas Jentik (ABJ) dengan nilai korelasi 0.285 dan arah korelasi berlawanan. Variabel ABJ akan dilibatkan sebagai *independent variable* pada skenario V1. Peringkat kedua nilai korelasi di dataran sedang adalah variabel Curah Hujan (CH) dengan nilai 0.204 dan arah korelasi linier. Variabel ABJ dan CH akan dilibatkan sebagai *independent variable* pada skenario V2. Peringkat ketiga nilai korelasi adalah variabel Kelembaban Udara (KU) dengan nilai 0.191 dan arah korelasi linier. Variabel ABJ, CH dan KU akan dilibatkan sebagai *independent variable* pada skenario V3.

Tabel 6.6. Uji Korelasi Variabel Observasi dan KDB di Dataran Sedang

Variabel Observasi	Korelasi dengan KDB
Angka Bebas Jentik	-0.285
Curah Hujan	0.204
Kelembaban Udara	0.191
Jumlah Penduduk	-0.133
Kecepatan Angin	-0.045

Suhu	0.038
------	-------

Keenam variabel observasi di dataran sedang memiliki nilai yang tidak cukup tinggi untuk dikategorikan memiliki korelasi kuat. Keenam variabel observasi akan dilibatkan sebagai *independent variable* pada skenario VA.

### 6.3.3. Hasil Uji Korelasi Variabel Observasi di Dataran Tinggi

Hasil uji korelasi variabel observasi dan variabel KDB pada dataran tinggi dapat dilihat pada Tabel 6.7. Peringkat pertama nilai korelasi di dataran tinggi adalah variabel Curah Hujan (CH) dengan nilai korelasi 0.297 dan arah korelasi linier. Variabel CH akan dilibatkan sebagai *independent variable* pada skenario V1. Peringkat kedua nilai korelasi di dataran tinggi adalah variabel Jumlah Penduduk (JP) dengan nilai 0.279 dan arah korelasi linier. Variabel JP akan dilibatkan menjadi *independent variable* pada skenario V2. Peringkat ketiga nilai korelasi di dataran tinggi adalah variabel Suhu (SH) dengan nilai korelasi 0.106 dan arah korelasi linier. Variabel CH, JP dan SH akan dilibatkan sebagai *independent variable* pada skenario V3.

**Tabel 6.7. Uji Korelasi Variabel Observasi dan KDB di Dataran Tinggi**

Variabel Observasi	Korelasi dengan KDB
Curah Hujan	0.297
Jumlah Penduduk	0.279
Suhu	0.106
Kecepatan Angin	-0.091
Kelembaban Udara	0.087
Angka Bebas Jentik	0.076

Keenam variabel observasi memiliki nilai korelasi dengan KDB yang tidak cukup tinggi untuk dikategorikan berkorelasi kuat.

Keenam variabel observasi akan dilibatkan sebagai *independent variable* pada skenario VA.

#### 6.4. Hasil Peramalan Variabel Observasi

Metode yang dipilih untuk meramalkan variabel observasi yang menjadi *independent variable* adalah metode yang memiliki nilai RMSE paling kecil pada pembentukan model di data sampel. Hasil perbandingan nilai RMSE pada tiga jenis metode dapat dilihat pada Tabel 6.8.

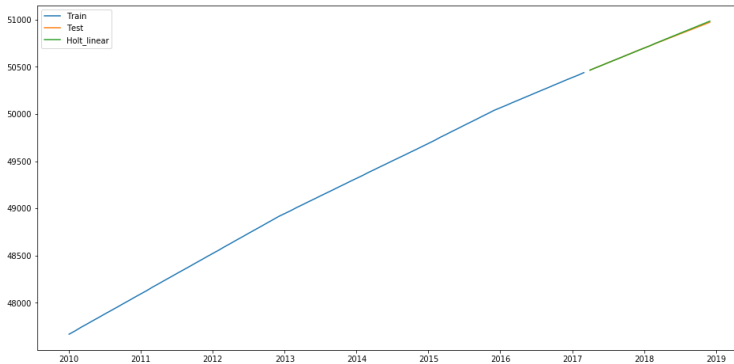
**Tabel 6.8. Perbandingan Nilai RMSE pada Tiga Metode**

Variabel	RMSE		
	Double exponential smoothing	Holt's Winter exponential smoothing	Box Jenkins (ARIMA)
SU	0.785	0.340	0.423
KU	3.370	2.089	1.981
CH	7.185	5.267	4.298
KA	0.330	0.208	0.288
JP	5.269	11.967	6.518
ABJ	8.249	10.536	5.981

Keenam variabel diramalkan dengan metode yang berbeda-beda. Metode yang dipilih adalah metode yang memiliki RMSE paling kecil yang diujikan di data sampel. Pada Tabel 6.8 dapat dilihat perbandingan RMSE ketiga metode saat diujikan di 6 variabel. Variabel Suhu (SU) dan Kecepatan Angin (KA) diramalkan dengan metode *Holt Winter Exponential Smoothing*. Variabel Kelembaban udara (KU), Curah Hujan (CH) dan Angka bebas Jentik (ABJ) diramalkan dengan metode Box Jenkins atau ARIMA. Sedangkan, variabel Jumlah Penduduk (JP) diramalkan dengan *Double Exponential Smoothing* atau yang biasa disebut *Holt's Linear*.

#### 6.4.1. Hasil Peramalan Variabel Observasi dengan *Double Exponential Smoothing*

Metode *Double Exponential Smoothing* dilakukan untuk meramalkan variabel Jumlah Penduduk (JP) karena memiliki RMSE paling kecil yaitu 5.269 (dapat dilihat pada Tabel 6.8). Peramalan jumlah penduduk yang telah dilakukan pada kecamatan Ngajum dapat dilihat pada Gambar 6.7. Hasil peramalan jumlah penduduk selama 24 periode di Kecamatan Ngajum dapat dilihat pada Lampiran B Tabel B.1.

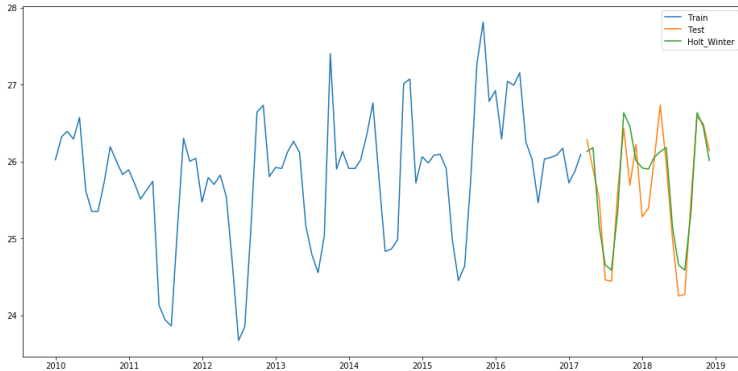


Gambar 6.7. Hasil Peramalan JP di Kecamatan Ngajum dengan *Double Exponential Smoothing*

#### 6.4.2. Hasil Peramalan Variabel Observasi dengan *Holt's Winter Exponential Smoothing*

Variabel Suhu (SU) dan Kecepatan Angin (KA) diramalkan dengan Metode *Holt's Winter exponential smoothing* karena memiliki RMSE yang paling kecil (dapat dilihat pada Tabel 6.8). Salah satu hasil peramalan dengan *Holt's Winter Exponential Smoothing* yaitu pada variabel suhu dapat dilihat pada Gambar 6.8. Hasil peramalan suhu (stasiun karangkates)

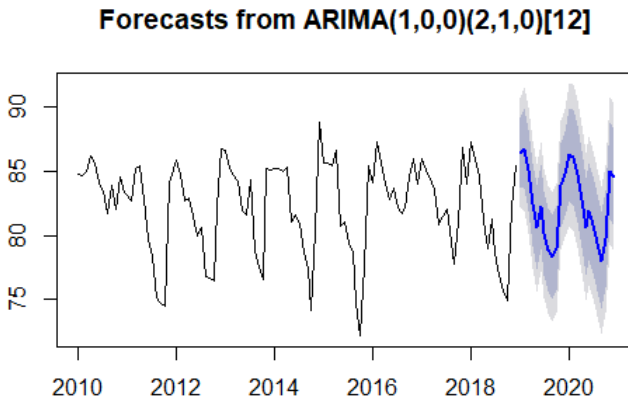
selama 24 periode kedepan dapat dilihat pada Lampiran B Tabel B.2.



**Gambar 6.8.** Hasil Peramalan SU Stasiun Karangates dengan *Holt's Winter Exponential Smoothing*

### **6.4.3. Hasil Peramalan Variabel Observasi dengan *Box Jenkins* atau ARIMA**

Variabel Kelembaban Udara (KU), Curah Hujan (CH) dan Angka Bebas Jentik (ABJ) diramalkan dengan metode *Box Jenkins* atau ARIMA karena memiliki RMSE paling kecil (dapat dilihat pada Tabel 6.8). Salah satu grafik hasil peramalan dengan menggunakan metode *Box Jenkins* yaitu pada variabel KU-Stasiun Karangates dapat dilihat pada Gambar 6.9. Hasil peramalan variabel KU-Stasiun BMKG Karangates selama 24 periode dapat dilihat pada Gambar 6.9. Hasil peramalan variabel KU-Stasiun BMKG Karangates selama 24 periode dapat dilihat pada Lampiran B Tabel B.3.



**Gambar 6.9.** Hasil Peramalan KU Stasiun Karangates dengan *Box Jenkins*

### 6.5. Hasil Pembentukan Model di Dataran Rendah

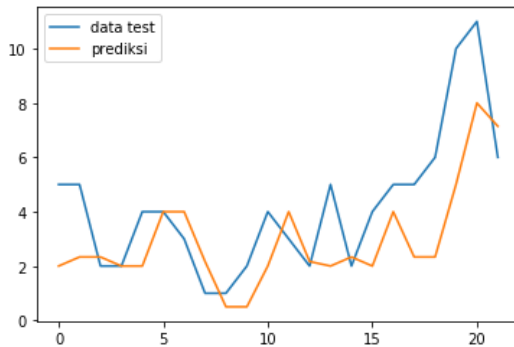
Model dengan RMSE paling kecil di dataran rendah terdapat di skenario pembagian data 80:20. Hasil RMSE paling kecil di setiap skenario *independent variable* (Skenario V0-V1-V2-V3-VA) dapat dilihat pada Tabel 6.9. Skenario yang menghasilkan model dengan RMSE paling kecil adalah skenario V0L01 yaitu skenario yang hanya melibatkan variabel *lag* dengan nilai *lag* 1. Nilai RMSE pada skenario V0L01 adalah 2.133. Parameter KNNR dengan model terbaik yaitu nilai  $n\_neighbor = 6$ ,  $algorithm = brute$ ,  $metrics = euclidean$ . Skenario V0L01 dipilih menjadi skenario untuk membentuk model peramalan.

**Tabel 6.9.** Model Peramalan pada training:test 80:20 di Dataran Rendah

Skenario	Parameter {n_neighbor, algorithm, metrics}	RMSE	SMAPE
V0L01	{6, brute, euclidean}	2.133	24.977

V1L09	{2, kd_tree, euclidean}	2.560	27.102
V2L03	{4, kd_tree, euclidean}	2.924	27.645
V3L06	{3, brute, manhattan}	3.127	28.503
VAL11	{3, brute, manhattan}	2.865	30.036

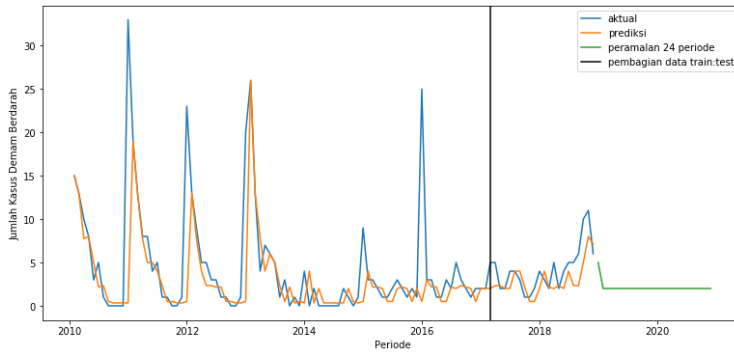
Grafik *testing data* dan prediksi dari skenario V0L01 dapat dilihat pada Gambar 6.10.



**Gambar 6.10. Plot Prediksi dan *Testing Data* Dataran Rendah**

Dapat dilihat pada Gambar 6.10, model prediksi dapat mengikuti data aktual namun terjadi keterlambatan. Pada akhir periode, model prediksi tidak bisa mengikuti data aktual yang mana terjadi puncak kasus. Grafik data aktual, prediksi beserta peramalan selama 24 periode dapat dilihat pada Gambar 6.6.





**Gambar 6.11. Plot Data Aktual dan Prediksi Dataran Rendah**

Hasil dari peramalan 24 periode dapat dilihat pada Lampiran F di Tabel F.1. Gambar 6.11 merupakan grafik *training data*, *testing data* dan peramalan 24 periode di kecamatan pembentuk model yaitu Kecamatan Kepanjen. Model terpilih dari skenario VOL01 memiliki nilai RMSE (*Root Mean Square Error*) 2.133 dan SMAPE (*Symmetric Mean Percentage Absolute Error*) sebesar 24.977%.

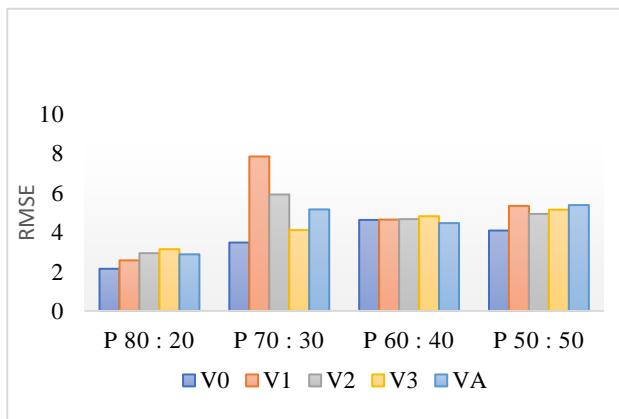
### 6.5.1. Perbandingan dengan Skenario Pembagian Data

Skenario yang menghasilkan model terbaik dengan RMSE paling kecil terdapat pada proporsi pembagian data 80:20. Model terbaik setiap skenario pada Tabel 6.11 akan dibandingkan dengan proporsi pembagian data lain seperti yang telah dijelaskan pada sub-bab 4.6.

Gambar 6.12 merupakan grafik perbandingan nilai RMSE skenario penghasil model terbaik pada proporsi pembagian data 80:20, 70:30, 60:40 dan 50:50. Nilai RMSE keempat skenario pembagian data tersebut dihitung rata-rata dan standar deviasinya untuk melihat *robustness* dari skenario penghasil model terbaik.

**Tabel 6.10. Perbandingan dengan Proporsi Pembagian Data Lain di Dataran Rendah**

Skenario	RMSE				Rata-rata	SD
	80:20	70:30	60:40	50:50		
V0L01	2.133	3.462	4.607	4.071	<b>3.568</b>	1.065
V1L09	2.560	7.823	4.625	5.318	5.082	2.171
V2L03	2.924	5.897	4.649	4.913	4.596	1.237
V3L06	3.127	4.095	4.796	5.128	4.286	<b>0.885</b>
VAL11	2.865	5.141	4.448	5.362	4.454	1.129



**Gambar 6.12. Grafik Nilai RMSE Setiap Proporsi di Dataran Rendah**

Nilai rata-rata RMSE paling kecil dimiliki oleh skenario V0 (hanya melibatkan variabel *lag*) dengan nilai 3.568. Skenario V0 juga memiliki nilai standar deviasi dua terbawah artinya, nilai RMSE yang muncul di setiap pembagian model tidak memiliki rentang yang jauh.

Standar deviasi paling kecil dimiliki oleh skenario V3 (melibatkan tiga variabel observasi) dengan nilai 0.885. Dapat dilihat pada Gambar 6.12 bahwa perubahan nilai pada diagram V3 tidak signifikan. Nilai rata-rata RMSE skenario V3 adalah nilai RMSE kedua terendah. Dilihat dari nilai rata-rata dan standar deviasinya, skenario V3 cukup baik dalam

mempertahankan RMSE pada proporsi pembagian data yang berbeda.

### 6.5.2. Uji Skenario Model pada Kecamatan Lain di Dataran Rendah

Model terbaik setiap skenario *independent variable* akan digunakan untuk menguji kecamatan lain. RMSE hasil pengujian di kecamatan lain di dataran rendah dapat dilihat pada Tabel 6.11.

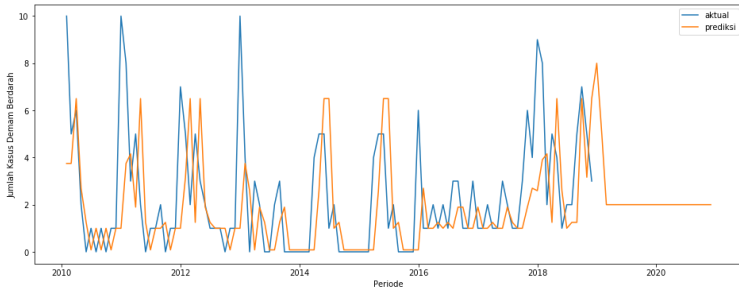
Pada Tabel 6.11, dapat dilihat bahwa rata-rata nilai RMSE kelima kecamatan di dataran rendah yang paling kecil adalah skenario VA (melibatkan semua variabel observasi). Dari nilai rata-rata tersebut, dapat disimpulkan bahwa model yang dihasilkan skenario VA cukup *robust* terhadap kecamatan lain.

**Tabel 6.11. Nilai RMSE di Setiap Kecamatan di Dataran Rendah**

	Skenario				
Kecamatan	V0L01	V1L09	V2L03	V3L06	VAL11
Kepanjen	2.133	2.560	2.924	3.127	2.865
Gondanglegi	2.535	2.676	2.411	2.372	2.459
Donomulyo	2.701	2.358	2.422	2.495	2.477
Bululawang	3.439	3.278	2.924	2.747	2.590
Turen	17.218	22.421	13.206	13.795	12.327
<b>Rata-rata</b>	<b>5.605</b>	<b>6.659</b>	<b>4.777</b>	<b>4.907</b>	<b>4.544</b>
<b>Median</b>	<b>2.701</b>	<b>2.676</b>	<b>2.924</b>	<b>2.747</b>	<b>2.59</b>

#### a) Kecamatan Gondanglegi

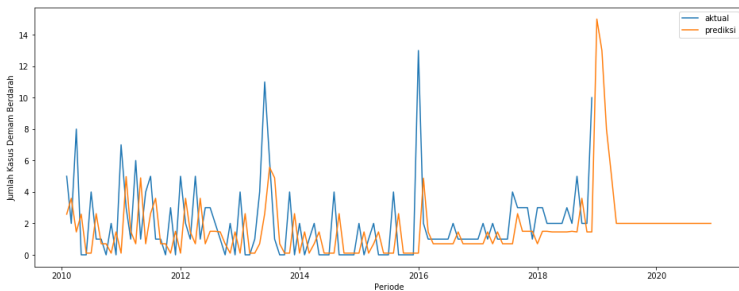
Dari Tabel 6.11 dapat diketahui bahwa nilai RMSE yang paling kecil di Kecamatan Gondanglegi adalah model dengan skenario V3 (melibatkan tiga variabel observasi). RMSE yang dihasilkan oleh model terpilih adalah 2.535. Grafik perbandingan data aktual dan prediksi pada Kecamatan Gondanglegi dapat dilihat pada Gambar 6.13. Hasil peramalan 24 periode di Kecamatan Gondanglegi dapat dilihat pada Lampiran D Tabel F.2.



**Gambar 6.13. Perbandingan Aktual dan Prediksi pada Kecamatan Gondanglegi**

b) Kecamatan Donomulyo

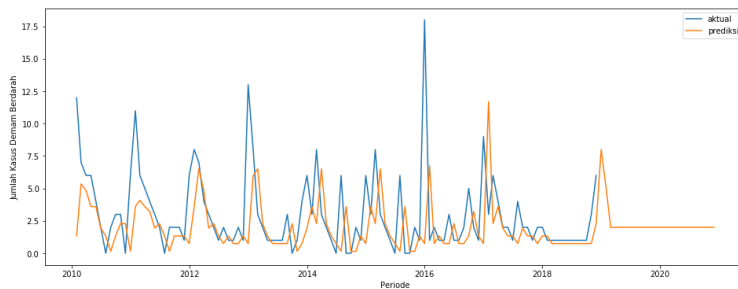
Dilihat dari Tabel 6.11, nilai RMSE paling kecil di Kecamatan Donomulyo ada pada model dengan skenario V1 (melibatkan satu variabel observasi). RMSE yang dihasilkan oleh model terpilih (model dengan skenario V0L01) adalah 2.701. Grafik perbandingan data aktual dan data prediksi di Kecamatan Donomulyo dengan model terpilih dapat dilihat pada Gambar 6.14. Hasil peramalan selama 24 periode di Kecamatan Donomulyo dapat dilihat pada Lampiran F Tabel F.3.



**Gambar 6.14. Perbandingan Aktual dan Prediksi pada Kecamatan Donomulyo**

c) Kecamatan Bululawang

Pada Tabel 6.11, dapat diketahui bahwa nilai RMSE yang paling kecil di kecamatan Bululawang ada pada model dengan skenario VA (melibatkan semua variabel observasi). RMSE pada model terpilih yaitu dari skenario V0 yang memiliki nilai 3.439. Grafik perbandingan data aktual dan hasil prediksi menggunakan model terpilih dapat dilihat pada Gambar 6.15. Hasil peramalan 24 periode di kecamatan Bululawang dapat dilihat pada lampiran D Tabel F.4.

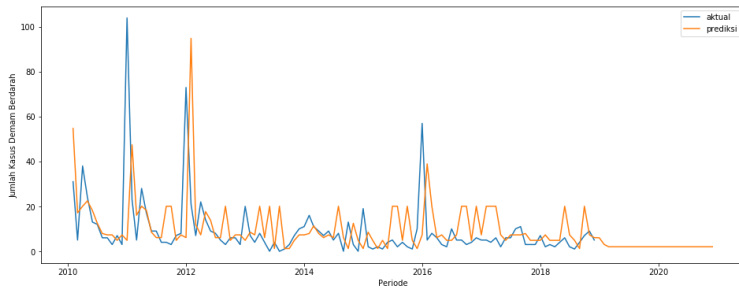


**Gambar 6.15. Perbandingan Aktual dan Prediksi pada Kecamatan Bululawang**

#### d) Kecamatan Turen

Kecamatan Turen merupakan Kecamatan yang memiliki nilai korelasi paling tinggi dengan kecamatan pembentuk model di dataran rendah yaitu Kecamatan Kepanjen. Nilai korelasi kasus demam berdarah antara Kecamatan Turen dan Kecamatan Kepanjen adalah 0.80 dengan arah korelasi linier. Angka korelasi tersebut tergolong tinggi dan dapat dikategorikan sebagai korelasi yang kuat. Meskipun demikian, nilai RMSE dengan model terpilih cukup tinggi yaitu sebesar 17.218. Pada Kecamatan Turen, nilai RMSE paling kecil ditemukan pada model dengan skenario VA (melibatkan semua variabel observasi). Grafik perbandingan data aktual dan hasil prediksi dengan model terpilih dapat dilihat pada Gambar 6.16. Hasil

peramalan selama 24 periode dapat dilihat pada lampiran D Tabel F.5.



**Gambar 6.16.** Perbandingan Aktual dan Prediksi pada Kecamatan Turen

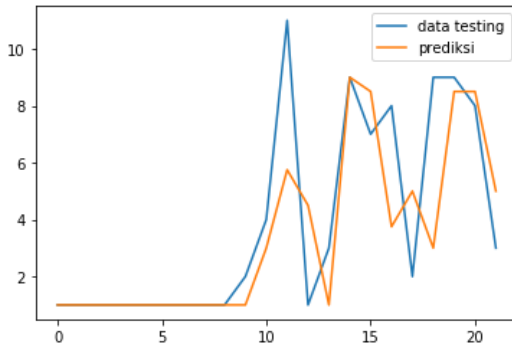
## 6.6. Hasil Pembentukan Model di Dataran Sedang

Model dengan nilai RMSE paling kecil di dataran sedang dihasilkan dari skenario pembagian data 80:20. Hasil RMSE paling kecil di setiap skenario *independent variable* (Skenario V0-V1-V2-V3-VA) dapat dilihat pada Tabel 6.12. Skenario yang menghasilkan model dengan RMSE paling kecil adalah skenario VOL01 yaitu skenario yang hanya melibatkan variabel *lag* dengan nilai *lag* 1. Nilai RMSE pada skenario VOL01 adalah 2.205. Parameter KNNR dengan model terbaik yaitu nilai  $n\_neighbor = 4$ ,  $algorithm = kd\_tree$ ,  $metrics = euclidean$ . Skenario VOL01 dipilih menjadi skenario untuk membentuk model peramalan.

**Tabel 6.12.** Model Peramalan pada training:test 80:20 di Dataran Sedang

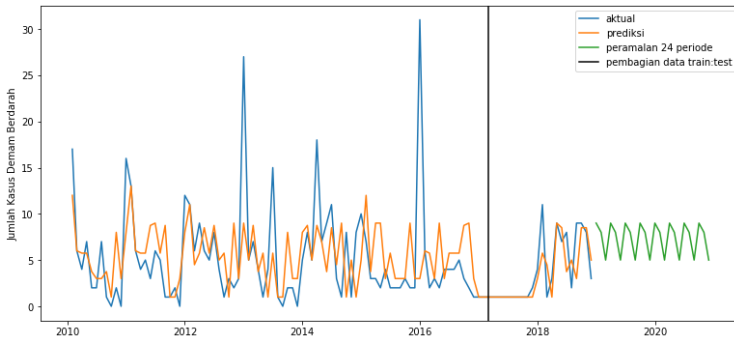
Skenario	Parameter {n_neighbor,algorithm, metrics}	RMSE	SMAPE
VOL01	{4, kd_tree, euclidean}	2.205	15.944
V1L03	{7, brute, manhattan}	2.915	36.353
V2L00	{2, brute, chebyshev}	3.631	36.380
V3L02	{4, brute, manhattan}	3.226	38.864
VAL01	{6, brute, euclidean}	3.416	39.404

Grafik *testing data* dan prediksi dari skenario VOL01 dapat dilihat pada Gambar 6.17.



**Gambar 6.17. Plot Prediksi dan Data Testing Dataran Sedang**

Model prediksi terpilih memiliki nilai RMSE dan SMAPE paling kecil diantara model terbaik di 5 skenario *independent variable*. Model terpilih yaitu model dari skenario VOL01 memiliki nilai RMSE sebesar 2.205 dan memiliki nilai SMAPE sebesar 15.944. Gambar 6.17 merupakan grafik perbandingan data *testing* dan prediksi yang dihasilkan untuk data *testing*. Model prediksi cukup bisa mengikuti nilai aktual di awal, namun saat nilai aktual mengalami kenaikan yang signifikan, model prediksi tidak bisa mengikuti. Perbandingan data aktual dan prediksi di seluruh periode dapat dilihat pada Gambar 6.18. Hasil prediksi selama 24 periode di kecamatan pembentuk model yaitu Kecamatan Pakisaji dapat dilihat pada Lampiran F Tabel F.6.



**Gambar 6.18. Plot Data Aktual dan Prediksi Dataran Sedang**

### 6.6.1. Perbandingan dengan Skenario Pembagian Data

Skenario yang menghasilkan model terbaik dengan RMSE paling kecil terdapat pada proporsi pembagian data 80:20. Model terbaik setiap skenario pada Tabel 6.12 akan dibandingkan dengan proporsi pembagian data lain seperti yang telah dijelaskan pada sub-bab 4.6. Hasil perbandingan nilai RMSE di setiap skenario pembagian data dapat dilihat pada Tabel 6.13.

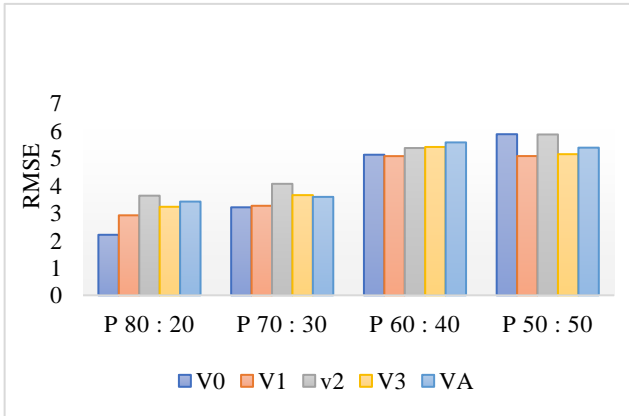
Gambar 6.19 merupakan grafik perbandingan nilai RMSE skenario penghasil model terbaik pada proporsi pembagian data 80:20, 70:30, 60:40 dan 50:50. Nilai RMSE keempat skenario pembagian data tersebut dihitung rata-rata dan standar deviasinya untuk melihat robustness dari skenario penghasil model terbaik.

**Tabel 6.13. Perbandingan dengan Proporsi Pembagian Data Lain di Dataran Sedang**

Skenario	RMSE				Rata-rata	SD
	80:20	70:30	60:40	50:50		
V0L01	2.205	3.207	5.124	5.872	4.102	1.691
V1L03	2.915	3.263	5.073	5.075	<b>4.081</b>	1.155



V2L00	3.631	4.063	5.368	5.862	4.731	<b>1.055</b>
V3L02	3.226	3.651	5.407	5.144	4.357	1.080
VAL01	3.416	3.587	5.573	5.381	4.489	1.146



**Gambar 6.19. Grafik Nilai RMSE Setiap Proporsi di Dataran Sedang**

Gambar 6.19 secara keseluruhan memperlihatkan bahwa semakin sedikit proporsi *training data* maka nilai RMSE semakin tinggi. *Robustness* model akan dilihat dari nilai rata-rata dan standar deviasi dari nilai RMSE di setiap proporsi pembagian data.

Nilai rata-rata RMSE paling kecil dimiliki oleh model dengan skenario V1L03 (melibatkan satu variabel observasi dan 3 variabel *lag*) dengan nilai 4.081. Nilai SMAPE pada model dengan skenario V1L03 juga tidak terlalu tinggi yaitu 1.055. Model yang dihasilkan skenario V1L03 cukup *robust*. Berbeda dengan skenario V2L00 (melibatkan 2 variabel observasi) yang memiliki nilai SMAPE paling kecil yaitu 1.055 namun memiliki nilai RMSE yang paling besar diantara kelima skenario sehingga tidak bisa dikatakan *robust*.

Model terpilih untuk melakukan peramalan yaitu VOL01 memiliki nilai RMSE yang cukup rendah diantara model dari skenario lain yaitu sebesar 4.102. Namun, model dari skenario VOL01 memiliki standar deviasi tertinggi diantara skenario lainnya yaitu sebesar 1.691. Model dari skenario VOL01 tidak cukup *robust*.

### 6.6.2. Uji Skenario Model pada Kecamatan Lain di Dataran Sedang

Model terbaik setiap skenario *independent variable* akan digunakan untuk menguji kecamatan lain. RMSE hasil pengujian di kecamatan lain di dataran rendah dapat dilihat pada Tabel 6.14.

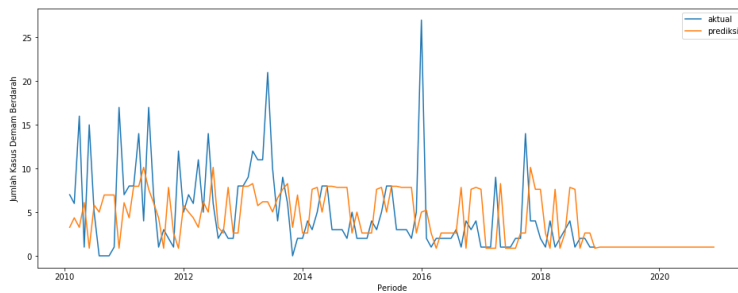
Pada Tabel 6.14, dapat dilihat bahwa rata-rata nilai RMSE kelima kecamatan di dataran rendah yang paling kecil adalah skenario VA (melibatkan semua variabel observasi). Dari nilai rata-rata tersebut, dapat disimpulkan bahwa model yang dihasilkan skenario VA cukup *robust* terhadap data kecamatan lain.

**Tabel 6.14. Nilai RMSE di Setiap Kecamatan di Dataran Sedang**

Kecamatan	Skenario				
	VOL01	V1L03	V2L00	V3L02	VAL01
Pakisaji	2.205	2.915	3.631	3.226	3.416
Tumpang	5.442	4.813	6.226	5.008	4.528
Lawang	7.442	3.524	6.728	7.730	6.777
Singosari	3.361	8.877	7.310	3.099	3.140
Wajak	3.570	9.555	7.071	2.977	3.042
Dampit	5.711	4.305	6.889	4.188	4.614
Sumbermanjing Wetan	10.744	2.735	7.637	8.820	9.024
Karangploso	3.389	8.051	6.828	3.290	3.020
<b>Rata-rata</b>	<b>5.233</b>	<b>5.597</b>	<b>6.540</b>	<b>4.792</b>	<b>4.695</b>
<b>Median</b>	<b>4.506</b>	<b>4.559</b>	<b>6.8585</b>	<b>3.739</b>	<b>3.972</b>

a) Kecamatan Tumpang

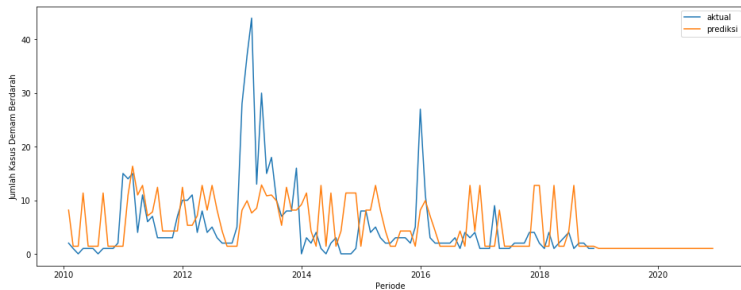
Model dengan RMSE paling kecil di Kecamatan Tumpang dihasilkan oleh skenario VAL01(melibatkan semua variabel observasi dan 1 variabel *lag*) yaitu sebesar 4.528. Sedangkan RMSE dari model terpilih menghasilkan RMSE terbesar kedua yaitu 5.442. Grafik perbandingan nilai aktual dan prediksi di Kecamatan Tumpang dapat dilihat pada Gambar 6.20. Hasil prediksi selama 24 periode di Kecamatan Tumpang dapat dilihat pada Lampiran F Tabel F.7.



**Gambar 6.20. Perbandingan Aktual dan Prediksi pada Kecamatan Tumpang**

b) Kecamatan Lawang

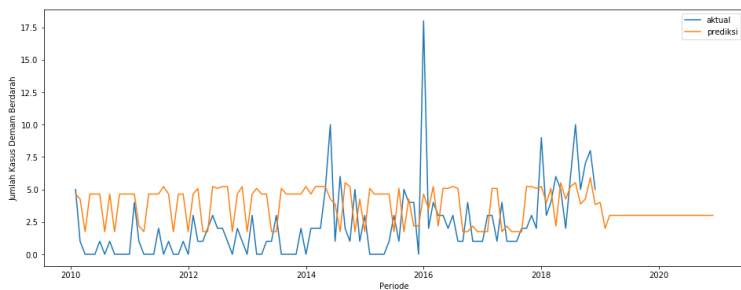
Nilai RMSE paling kecil di Kecamatan Lawang adalah 3.524 yang dihasilkan oleh model dengan skenario V1L03(melibatkan 1 variabel observasi dan 3 variabel *lag*). Sedangkan, RMSE model dari skenario terpilih yaitu V0L01 cukup berbeda jauh yaitu 7.442. Grafik perbandingan data aktual dan data prediksi dapat dilihat pada Gambar 6.21. Hasil prediksi selama 24 periode di Kecamatan Lawang dapat dilihat pada Lampiran F Tabel F.8.



**Gambar 6.21. Perbandingan Aktual dan Prediksi pada Kecamatan Lawang**

c) Kecamatan Singosari

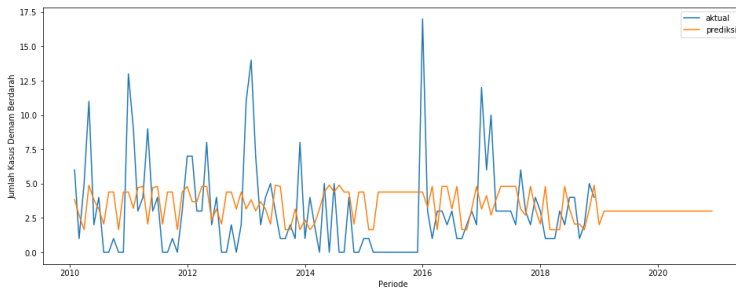
Nilai RMSE paling kecil di Kecamatan Singosari dimiliki oleh model dari skenario V3L02 (melibatkan 3 variabel observasi dan 2 variabel *lag*) yaitu sebesar 3.099. Nilai RMSE yang dihasilkan model dari skenario terpilih tidak terlalu berbeda jauh dengan nilai RMSE terkecil, yaitu 3.361. Grafik perbandingan nilai aktual dan hasil prediksi di Kecamatan Singosari dapat dilihat pada Gambar 6.22. Hasil peramalan 24 periode dengan model terpilih di Kecamatan Singosari dapat dilihat pada Lampiran F Tabel F.9.



**Gambar 6.22. Perbandingan Aktual dan Prediksi pada Kecamatan Singosari**

d) Kecamatan Wajak

Nilai RMSE paling kecil di Kecamatan Wajak dimiliki oleh model dengan skenario V3L02 (melibatkan 3 variabel observasi dan 2 variabel *lag*) yaitu 2.977. Nilai RMSE model dari skenario terpilih yaitu V0L01 sebesar 3.361. Grafik perbandingan nilai aktual dan prediksi di Kecamatan Wajak dapat dilihat pada Gambar 6.23. Hasil peramalan selama 24 periode di Kecamatan Wajak dapat dilihat pada Lampiran F Tabel F.10.

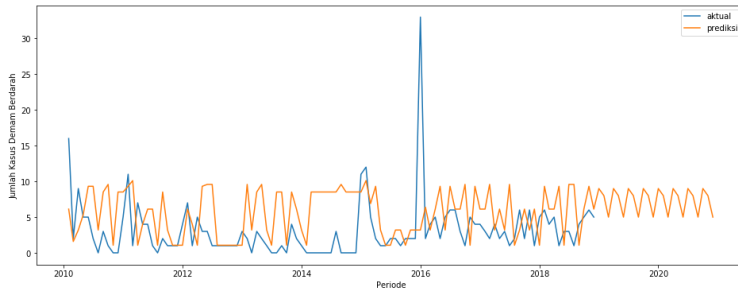


**Gambar 6.23. Perbandingan Aktual dan Prediksi pada Kecamatan Wajak**

#### e) Kecamatan Dampit

Kecamatan Dampit adalah kecamatan yang memiliki nilai korelasi paling tinggi dengan kecamatan pembentuk model yaitu Kecamatan Pakisaji. Nilai korelasi antara Kecamatan Dampit dan Kecamatan Pakisaji adalah 0.52 yang bisa dikategorikan berkorelasi cukup kuat. Meskipun demikian, model terbaik yang berhasil dibentuk di Kecamatan Pakisaji (model dengan skenario V0L01) tidak menjadi model terbaik di Kecamatan Dampit. Model terpilih tersebut saat diujika di Kecamatan Dampit memiliki RMSE 5.711. Sedangkan, nilai RMSE paling kecil di Kecamatan Dampit adalah model yang berasal dari skenario V3L02 (melibatkan 3 variabel observasi dan 2 variabel *lag*) yaitu sebesar 4.188. Grafik perbandingan nilai aktual dan hasil prediksi di Kecamatan Dampit dapat dilihat

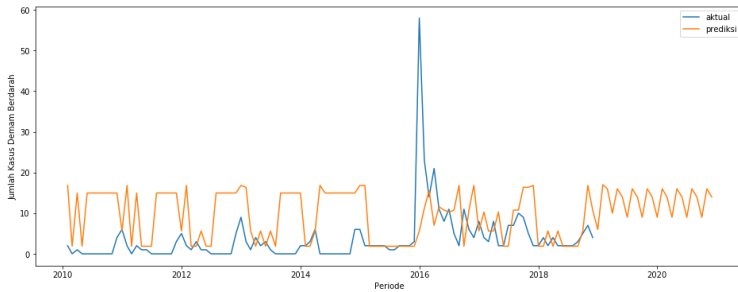
pada Gambar 6.24. Hasil peramalan selama 24 periode dengan model terpilih di Kecamatan Dampit dapat dilihat pada Lampiran F Tabel F.11.



**Gambar 6.24. Perbandingan Aktual dan Prediksi pada Kecamatan Dampit**

#### f) Kecamatan Sumbermanjing Wetan

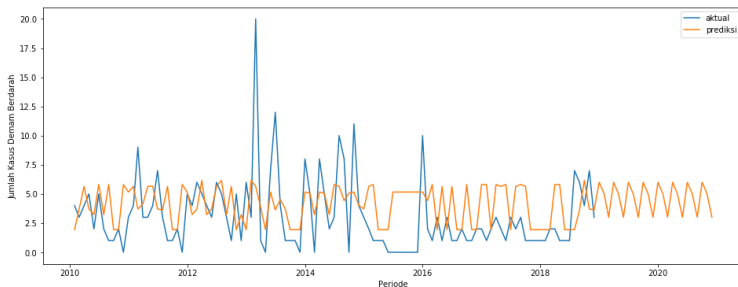
Model terbaik yang dihasilkan di kecamatan pembangun model berasal dari skenario V0L01 (hanya melibatkan 1 variabel *lag*) yang memiliki nilai RMSE 10.744 ketika diujikan di Kecamatan Sumbermanjing Wetan. Nilai RMSE tersebut menjadi nilai RMSE tertinggi bila dibandingkan dengan model yang dihasilkan skenario lain. Nilai RMSE paling kecil di Kecamatan Sumbermanjing Wetan adalah model dari skenario V1L03 (melibatkan 1 variabel observasi dan 3 variabel *lag*) yaitu sebesar 2.735. Grafik perbandingan nilai aktual dan hasil prediksi dengan model terpilih di Kecamatan Sumbermanjing Wetan dapat dilihat pada Gambar 6.25. Hasil peramalan selama 24 periode dengan model terpilih di Kecamatan Sumbermanjing Wetan dapat dilihat pada Lampiran F Tabel F.12.



**Gambar 6.25. Perbandingan Aktual dan Prediksi pada Kecamatan Sumbermanjing Wetan**

g) Kecamatan Karangploso

Nilai RMSE paling kecil di Kecamatan Karangploso yaitu 3.020 dihasilkan oleh model dengan skenario VAL01 (melibatkan semua variabel observasi dan 1 variabel *lag*). Model terpilih yaitu model dari skenario VOL01 menghasilkan RMSE 3.389. Grafik perbandingan nilai aktual dan hasil prediksi dapat dilihat pada Gambar 6.26. Hasil prediksi selama 24 periode dengan model terpilih di Kecamatan Karangploso dapat dilihat pada Lampiran F Tabel F.13.



**Gambar 6.26. Perbandingan Aktual dan Prediksi pada Kecamatan Karangploso**

## 6.7. Hasil Pembentukan Model di Dataran Tinggi

Model dengan nilai RMSE paling kecil di dataran sedang dihasilkan dari skenario pembagian data 80:20. Hasil RMSE paling kecil di setiap skenario *independent variable* (Skenario V0-V1-V2-V3-VA) dapat dilihat pada Tabel 6.15. Skenario yang menghasilkan model dengan RMSE paling kecil adalah skenario V2L02 yaitu skenario yang melibatkan variabel obserbasi dan variabel *lag* dengan nilai *lag* 2. Nilai RMSE pada skenario V2L02 adalah 1.543. Parameter KNNR dengan model terbaik yaitu nilai  $n\_neighbor = 2$ ,  $algorithm = brute$ ,  $metrics = chebyshev$ . Skenario V2L02 dipilih menjadi skenario untuk membentuk model peramalan.

**Tabel 6.15. Model Peramalan pada training:test 80:20 di Dataran Tinggi**

Skenario	Parameter {n_neighbor,algorithm, metrics}	RMSE	SMAPE
V0L09	{5, ball_tree_euclidean}	1.553	23.136
V1L02	{2, kd_tree, manhattan}	2.020	38.283
V2L02	{2, brute, chebyshev}	1.543	30.716
V3L01	{3, brute, chebysev}	1.615	28.998
VAL01	{5, brute, euclidean}	1.788	32.093

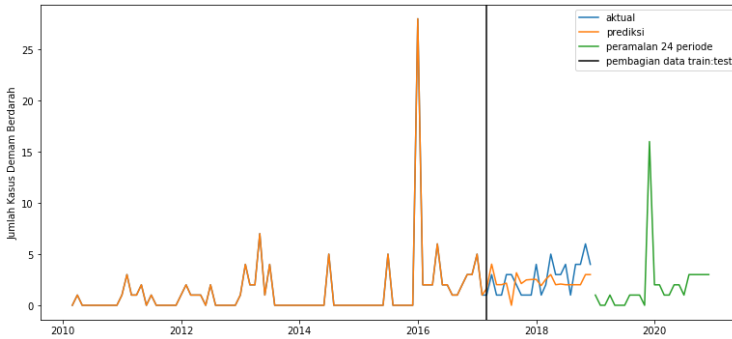
Grafik *testing data* dan prediksi dari skenario V2L02 dapat dilihat pada Gambar 6.27.



**Gambar 6.27. Plot Prediksi dan Data Testing Dataran Tinggi**



Model prediksi memiliki nilai RMSE paling kecil diantara model dari skenario lainnya yaitu 1.543. Model prediksi dapat mengikuti data aktual pada periode awal *testing*. Setelah periode kelima, model cenderung ada di pertengahan nilai data aktual sehingga nilai RMSE cenderung kecil. Grafik perbandingan data aktual dan hasil prediksi pada seluruh periode dapat dilihat pada Gambar 6.28. Hasil peramalan selama 24 periode di kecamatan pembentuk model dataran tinggi atau Kecamatan Ngajum dapat dilihat pada Lampiran D Tabel F.14.



**Gambar 6.28. Plot Data Aktual dan Prediksi Dataran Tinggi**

### 6.7.1. Perbandingan dengan Skenario Pembagian Data

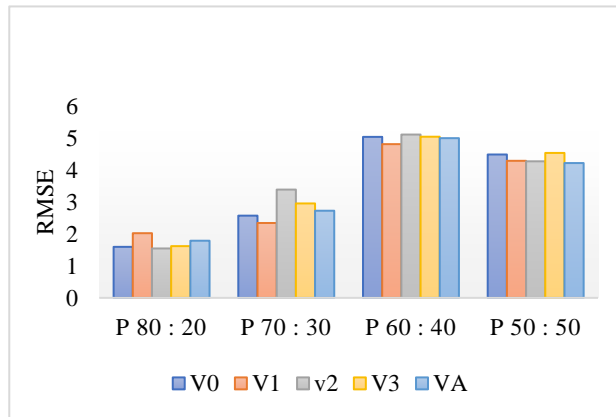
Skenario yang menghasilkan model terbaik dengan RMSE paling kecil terdapat pada proporsi pembagian data 80:20. Model terbaik setiap skenario pada Tabel 6.15 Tabel 6.12 akan dibandingkan dengan proporsi pembagian data lain seperti yang telah dijelaskan pada sub-bab 4.6. Hasil perbandingan nilai RMSE di setiap skenario pembagian data dapat dilihat pada Tabel 6.16.

Gambar 6.29 merupakan diagram perbandingan nilai RMSE skenario penghasil model terbaik pada proporsi pembagian data 80:20, 70:30, 60:40 dan 50:50. Nilai RMSE keempat

skenario pembagian data tersebut dihitung rata-rata dan standar deviasinya untuk melihat *robustness* dari skenario penghasil model terbaik.

**Tabel 6.16. Perbandingan dengan Proporsi Pembagian Data Lain di Dataran Tinggi**

Skenario	RMSE				Rata-rata	SD
	80:20	70:30	60:40	50:50		
V0L09	1.593	2.568	5.029	4.479	3.417	1.610
V1L02	2.020	2.339	4.802	4.282	<b>3.361</b>	<b>1.387</b>
V2L02	1.543	3.383	5.102	4.266	3.573	1.525
V3L01	1.615	2.949	5.034	4.528	3.532	1.556
VAL01	1.785	2.722	4.989	4.211	3.427	1.443



**Gambar 6.29. Diagram Nilai RMSE Setiap Proporsi di Dataran Tinggi**

Pada Gambar 6.29 dapat diketahui bahwa secara keseluruhan model yang dibangun dengan skenario proporsi pembagian data 60:40 memiliki RMSE paling tinggi. *Robustness* model akan dilihat dari nilai rata-rata dan standar deviasi dari nilai RMSE di setiap proporsi pembagian data.

Model dari skenario terpilih yaitu V2L02 (melibatkan 2 variabel observasi dan 2 variabel *lag*) memiliki nilai rata-rata 3.573 yang merupakan nilai rata-rata tertinggi. Sedangkan standar deviasi model V2L02 adalah 1.525, lebih kecil dari dua model lainnya. Dilihat dari nilai rata-rata dan standar deviasi model terpilih yaitu V0L02 dapat disimpulkan bahwa model V2L02 tidak cukup *robust*.

Nilai rata-rata RMSE dan standar deviasi yang paling kecil dimiliki oleh model dengan skenario V1L02 (melibatkan 1 variabel observasi dan 2 variabel *lag*).

### 6.7.2. Uji Skenario Model pada Kecamatan Lain di Dataran Tinggi

Model terbaik setiap skenario *independent variable* akan digunakan untuk menguji kecamatan lain. RMSE hasil pengujian di kecamatan lain di dataran rendah dapat dilihat pada Tabel 6.17.

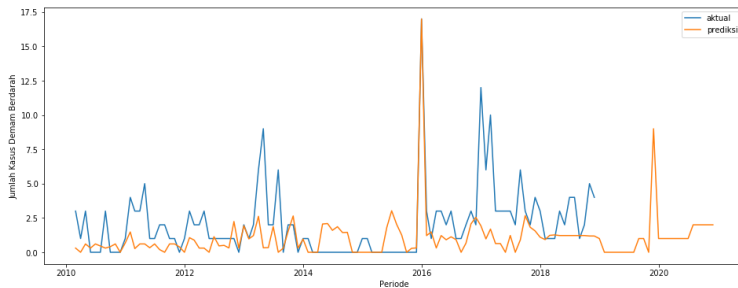
Pada Tabel 6.17, dapat dilihat bahwa rata-rata nilai RMSE ketiga kecamatan di dataran rendah yang paling kecil adalah skenario V2 (melibatkan 2 variabel observasi) yaitu 2.170. Dari nilai rata-rata tersebut, dapat disimpulkan bahwa model yang dihasilkan skenario V2 cukup *robust* terhadap data kecamatan lain.

**Tabel 6.17. Nilai RMSE di Setiap Kecamatan di Dataran Tinggi**

Kecamatan	Skenario				
	V0L09	V1L02	V2L02	V3L01	VAL01
Ngajum	1.553	2.020	1.543	1.615	1.788
Poncokusumo	2.818	2.827	2.340	2.423	2.620
Jabung	2.529	2.536	2.626	2.657	2.564
<b>Rata-rata</b>	<b>2.300</b>	<b>2.461</b>	<b>2.170</b>	<b>2.232</b>	<b>2.324</b>
<b>Median</b>	<b>2.529</b>	<b>2.536</b>	<b>2.34</b>	<b>2.423</b>	<b>2.564</b>

a) Kecamatan Poncokusumo

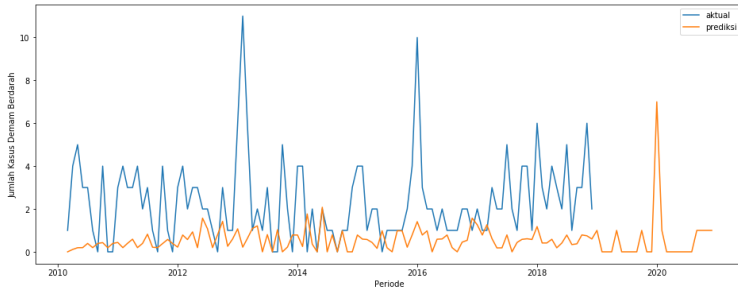
Model terpilih yaitu model dengan skenario V2L02 memiliki nilai RMSE paling kecil saat diujikan di Kecamatan Poncokusumo. Karena memiliki nilai RMSE yang paling kecil, bisa disimpulkan bahwa model terpilih cukup *robust* terhadap data di Kecamatan Poncokusumo. Grafik perbandingan data aktual dan hasil prediksi dengan model terpilih dapat dilihat pada Gambar 6.30. Hasil peramalan selama 24 periode dengan model terpilih di Kecamatan Poncokusumo dapat dilihat pada Lampiran F Tabel F.15.



**Gambar 6.30. Perbandingan Aktual dan Prediksi pada Kecamatan Poncokusumo**

b) Kecamatan Jabung

Model terpilih memiliki RMSE tertinggi dibanding dengan model lainnya yaitu sebesar 2.626 ketika diujikan di Kecamatan Jabung. Bila dibandingkan dengan model lainnya, model terpilih yaitu dengan skenario V2L02 tidak cukup *robust* terhadap data di Kecamatan Jabung. Grafik perbandingan nilai aktual dan hasil prediksi dengan model terpilih di Kecamatan Jabung dapat dilihat pada Gambar 6.31. Hasil peramalan selama 24 periode di Kecamatan Jabung dapat dilihat pada Lampiran D Tabel F.16.



**Gambar 6.31. Perbandingan Aktual dan Prediksi pada Kecamatan Jabung**

## 6.8. Hasil Pencarian Satu Model Terbaik untuk Seluruh Kecamatan

Model terbaik dari setiap skenario di setiap dataran diujikan di semua kecamatan di dataran lainnya. Nilai median RMSE hasil uji di seluruh kecamatan dapat dilihat pada Tabel 6.18. Hasil pengujian dapat dilihat lampiran G. Pada Tabel 6.18 dapat dilihat bahwa model dengan median RMSE paling kecil adalah model V3L02 Dataran Sedang yaitu dengan nilai RMSE 3.258. Model V2L02 Dataran Sedang dengan nilai parameter {4, brute, manhattan} dapat digunakan sebagai model tunggal yang bisa meramalkan semua kecamatan dengan baik.

**Tabel 6.18. Perbandingan Median RMSE Hasil Pengujian di Seluruh Kecamatan**

Dataran Pembangun Model	Skenario	Parameter {n_neighbor, algorithm, metrics}	Median RMSE
Dataran Rendah	V0L01	{6, brute, euclidean}	3.5274
	V1L09	{2, kd_tree, euclidean}	4.6438
	V2L03	{4, kd_tree, euclidean}	3.4918
	V3L06	{3, brute, manhattan}	3.7866
	VAL11	{3, brute, manhattan}	3.4434

Dataran Sedang	V0L01	{4, kd_tree, euclidean}	3.5481
	V1L03	{7, brute, manhattan}	5.6174
	V2L00	{2, brute, chebyshev}	3.757
	V3L02	{4, brute, manhattan}	<b>3.258</b>
	VAL01	{6, brute, euclidean}	3.278
Dataran Tinggi	V0L09	{5, ball_tree_euclidean}	3.8079
	V1L02	{2, kd_tree, manhattan}	3.963
	V2L02	{2, brute, chebyshev}	3.4555
	V3L01	{3, brute, chebyshev}	3.5626
	VAL01	{5, brute, euclidean}	3.848

### 6.9. Perbandingan dengan Metode lain

Dengan data yang sama, dilakukan percobaan untuk membuat model peramalan dengan metode lain. Metode yang digunakan adalah RNN-LSTM, *linear regression*, XG-Boost, Random Forest, dan RBFNN. Hasil perbandingan RMSE yang dihasilkan dari ke-enam metode dapat dilihat pada Tabel 6.19.

**Tabel 6.19. Perbandingan Nilai RMSE dengan Metode lain**

Metode	RMSE		
	DR	DS	DT
<i>K-nearest neighbor regression</i>	2.133	2.205	1.543
RNN-LSTM	2.158	3.473	1.632
<i>Linear regression</i>	2.008	3.508	1.888
XG-Boost	2.399	2.699	1.385
Random Forest	2.257	2.596	1.343
RBFNN	5.623	2.685	1.753

Peringkat RMSE yang dihasilkan dari setiap metode berbeda-beda di setiap datarannya. RMSE dengan nilai paling kecil di dataran rendah adalah model yang dihasilkan dengan metode *linear regression*. Model terbaik dengan RMSE paling kecil di dataran sedang adalah model yang dihasilkan dengan metode KNNR yaitu dengan RMSE sebesar 2.205. Model terbaik dengan RMSE paling kecil di dataran tinggi adalah model yang

dihasilkan dengan metode *random forest* yaitu dengan RMSE 1.343.

### 6.10. Perbandingan dengan Normalisasi di Setelah Proses Pembagian Data

Pada percobaan yang telah dilakukan untuk membentuk model, normalisasi dilakukan sebelum pembagian data sehingga data yang akan menjadi *testing data* sudah dinormalisasi dengan skala data keseluruhan (termasuk *training data*). Dilakukan percobaan lain sebagai pembandingan yaitu normalisasi setelah pembagian data dengan asumsi *testing data* adalah data yang dianggap belum diketahui nilainya diawal. Normalisasi *training data* dan *testing data* dilakukan terpisah. Perbandingan hasil dari normalisasi di tahap awal dan normalisasi setelah pembagian data dapat dilihat pada Tabel 6.20.

**Tabel 6.20. Perbandingan RMSE dengan Normalisasi Setelah Pembagian Data**

Dataran	RMSE	
	Normalisasi di awal	Normalisasi di akhir
Dataran Rendah	2.133	2.133
Dataran Sedang	2.205	2.292
Dataran Tinggi	1.543	3.960

Pada Tabel 6.20 didapatkan hasil bahwa melakukan normalisasi setelah pembagian data tidak berpengaruh besar terhadap nilai RMSE. Nilai RMSE normalisasi setelah pembagian data cenderung lebih besar dari normalisasi di tahap awal. Strategi untuk meminimalkan RMSE dengan melakukan normalisasi setelah pembagian data tidak perlu dilakukan.

### 6.11. Kesimpulan Percobaan

Tiga model peramalan terbaik yang dihasilkan dengan metode *k-nearest neighbor* di setiap dataran memiliki nilai *error* yang berbeda.

1. Di dataran rendah, kecamatan yang terpilih menjadi pembentuk model adalah Kecamatan Kepanjen. Model terbaik yang dihasilkan di Kecamatan Kepanjen memiliki nilai RMSE 2.133 dan memiliki nilai SMAPE 24.977%. Model terbaik di dataran rendah adalah model yang dibentuk dari skenario pembagian data 80:20, hanya melibatkan variabel *lag* 1 dan memiliki parameter  $n\_neighbor = 6$ ,  $algorithm = brute$ ,  $metrics = euclidean$ . Model dari skenario terpilih cukup *robust* saat diujikan pada proporsi pembagian data lain. Kecamatan Gondanglegi menjadi kecamatan lain yang memiliki RMSE paling rendah setelah diuji dengan model terpilih yang dibangun di Kecamatan Kepanjen. RMSE pada Kecamatan Gondanglegi sebesar 2.535.
2. Di dataran sedang, kecamatan yang terpilih menjadi pembentuk model adalah Kecamatan Pakisaji. Model terbaik yang dihasilkan di Kecamatan Pakisaji memiliki nilai RMSE 2.205 dan memiliki nilai SMAPE 15.944%. Model terbaik di dataran sedang adalah model yang dibentuk dari skenario pembagian data 80:20, hanya melibatkan variabel *lag* 1 dan memiliki parameter  $n\_neighbor = 4$ ,  $algorithm = kd\_tree$ ,  $metrics = euclidean$ . Model dari skenario terpilih tidak *robust* saat diujikan pada proporsi pembagian data lain. Kecamatan Singosari menjadi kecamatan lain yang memiliki RMSE paling rendah setelah diuji dengan model terpilih yang dibangun di Kecamatan Pakisaji. RMSE pada Kecamatan Singosari sebesar 3.361.
3. Di dataran tinggi, kecamatan yang terpilih menjadi pembentuk model adalah Kecamatan Ngajum. Model terbaik yang dihasilkan di Kecamatan Ngajum memiliki nilai RMSE 1.543 dan memiliki nilai SMAPE 30.716%. Model terbaik di dataran tinggi adalah model yang dibentuk dari skenario pembagian data 80:20, melibatkan 2 variabel observasi (Angka Bebas Jentik, Curah Hujan) dan variabel *lag* 2 dan memiliki parameter  $n\_neighbor = 2$ ,



*algorithm = brute, metrics = chebyshev*. Model dari skenario terpilih tidak cukup *robust* saat diujikan pada proporsi pembagian data lain. Kecamatan Jabung menjadi kecamatan lain yang memiliki RMSE paling rendah setelah diuji dengan model terpilih yang dibangun di Kecamatan Ngajum. RMSE pada Kecamatan Jabung sebesar 2.529.

4. Model terpilih dengan RMSE paling kecil saat diujikan dengan pembagian data lain dan dataset kecamatan lain tidak signifikan terhadap hasil diinginkan. Model yang memiliki RMSE paling kecil pada pembagian data tertentu belum tentu memiliki RMSE yang kecil pula bila jumlah *training data* dikurangi. Model yang memiliki RMSE paling kecil di kecamatan pembentuk model juga belum tentu memiliki RMSE yang kecil di kecamatan lainnya.
5. Mengapa metode terbaik untuk setiap dataran berbeda? Apakah karena karakteristik dataset di setiap data yang berbeda? Atau adakah faktor lain yang memengaruhi?

## 6.12. Analisis Hasil

Analisa hasil dibagi menjadi tiga bagian. Yang pertama adalah analisis hasil peramalan yang berisi analisa model terpilih. Kedua, analisa segi manajerial yang berisi *insight* dan saran-saran yang ditujukan untuk pemerintah terkait kasus demam berdarah di Kabupaten Malang. Ketiga, analisa metode *k-nearest neighbor regression* yang berisi *insight* dan saran untuk pengguna algoritma.

### 6.12.1. Analisis Hasil Peramalan

RMSE terbaik di setiap dataran memiliki nilai yang berbeda. Perbedaan nilai RMSE di dataran rendah dan di dataran sedang tidak terlalu jauh. Berbeda dengan RMSE terbaik pada dataran tinggi yang memiliki nilai yang cukup jauh berbeda dengan dua dataran lainnya. Perbedaan nilai RMSE pada tiap dataran dikarenakan setiap dataran memiliki sebaran data yang berbeda.

Sebaran data dapat dilihat dengan menghitung nilai standar deviasinya. Standar deviasi kasus demam berdarah di Dataran rendah di Kecamatan Kepanjen adalah 6.689, di dataran sedang yaitu Kecamatan Pakisaji sebesar 5.120, dan di dataran tinggi yaitu Kecamatan Ngajum sebesar 3.060.

**Tabel 6.21. Perbandingan RMSE Terbaik di Setiap Dataran**

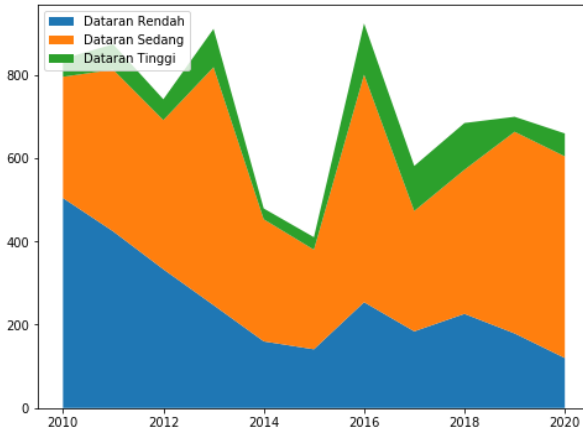
<b>Dataran</b>	<b>Skenario</b>	<b>RMSE Terbaik</b>
Rendah (Kecamatan Kepanjen)	V0L01	2.133
Sedang (Kecamatan Pakisaji)	V0L01	2.205
Tinggi (Kecamatan Ngajum)	V2L02	1.543

### **6.12.2. Analisis Segi Manajerial**

Proses-proses yang telah dilakukan pada tugas akhir ini juga menghasilkan informasi yang berguna untuk pemerintah guna mempersiapkan upaya mitigasi kasus demam berdarah. Grafik pada Gambar 6.32 memperlihatkan jumlah kasus demam berdarah di tiga dataran di Kabupaten Malang tahun 2010 – 2020. Data tahun 2010 hingga 2018 merupakan data aktual sedangkan, data tahun 2019 dan 2020 merupakan data hasil peramalan. Pada grafik terlihat bahwa warna orange mendominasi artinya, dataran sedang adalah dataran yang dominan atau memiliki kasus demam berdarah paling tinggi di Kabupaten Malang. Untuk itu, saran untuk pemerintah adalah meningkatkan upaya atau memaksimalkan program mitigasi demam berdarah di kawasan dataran sedang.

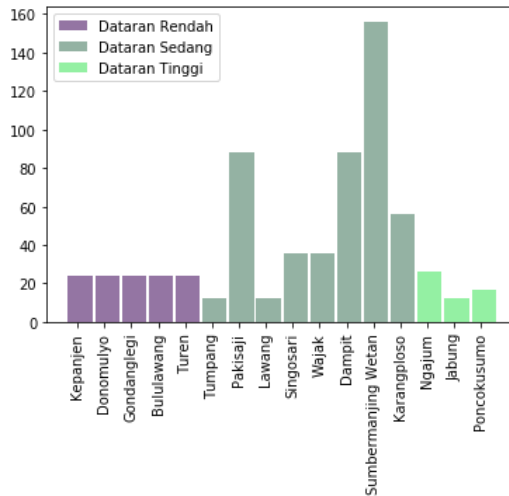
Pada Gambar 6.32 terlihat pada tahun 2019 dan 2020 terjadi tren penurunan baik di dataran rendah, sedang, ataupun tinggi. Hal ini bisa menjadi apresiasi untuk pemerintah Kabupaten Malang dan Dinas Kesehatan Kabupaten Malang. Penurunan jumlah kasus demam berdarah bisa mengindikasikan bahwa program mitigasi kasus demam berdarah yang telah dilakukan

pemerintah atau Dinas Kesehatan Kabupaten Malang berhasil dan efektif.



**Gambar 6.32. Grafik Kasus Demam Berdarah di Kabupaten Malang**

Hasil peramalan kasus demam berdarah Kabupaten Malang tahun 2020 di setiap kecamatan dapat dilihat pada Gambar 6.33. Kasus demam di Kabupaten Malang tahun 2020 diprediksi akan didominasi dari dataran sedang. Kecamatan di dataran sedang yang diprediksi terjadi kasus demam berdarah terbanyak adalah Kecamatan Sumbermanjing Wetan dengan kasus pertahun yang bisa mencapai 150 kasus. Diperlukan perhatian khusus untuk Kecamatan Sumbermanjing Wetan, Pakisaji dan Dampit.



**Gambar 6.33. Hasil Peramalan Jumlah Kasus Demam Berdarah di Kabupaten Malang tahun 2020**

Dataran rendah dan dataran tinggi memang diprediksi tidak banyak kasus yang terjadi di tahun 2020. Namun bukan berarti kecamatan di dataran rendah dan tinggi aman dari kasus demam berdarah karena diprediksi masih tetap terjadi namun di angka kurang dari 30 dalam satu tahun.

### 6.12.3. Analisis Metode *K-Nearest Neighbor Regression*

Analisis metode *k-nearest neighbor regression* dilakukan dengan menganalisa sensitifitas parameter. Ada tiga parameter yang digunakan untuk *tuning parameter* yaitu *n\_neighbor*, *metrics* dan *algorithm*. Sensitifitas dilihat dari nilai MSE karena MSE dapat melihat perbedaan nilai yang signifikan. Hasil sensitifitas setiap parameter adalah sebagai berikut:

#### 1) *N\_neighbor*

Sensitifitas parameter *n\_neighbor* dilihat dari perubahan nilai MSE saat nilai *n\_neighbor* diubah-ubah namun nilai parameter lainnya tetap. Sensitifitas *n\_neighbor* dilihat pada tiga model

terpilih yaitu model di dataran rendah, dataran sedang dan dataran tinggi. Nilai parameter tetap di dataran rendah adalah { *algorithm* : *brute*, *metrics* : *euclidean* }. Nilai parameter tetap di dataran sedang adalah { *algorithm* : *kd\_tree*, *metrics* : *euclidean* }. Nilai parameter tetap di dataran tinggi adalah { *algorithm* : *brute*, *metrics* : *chebyshev* }. Perbandingan MSE setiap nilai *n\_neighbor* dapat dilihat pada Tabel 6.22.

**Tabel 6.22. Sensitifitas Parameter *n\_neighbor***

N_neighbor	MSE		
	DR	DS	DT
2	9.914	8.364	2.381
3	6.91	6.015	2.901
4	6.632	5.256	4.33
5	5.355	6.182	4.24
6	4.549	6.605	4.19
7	4.984	6.968	3.954
<b>SD</b>	<b>1.959</b>	<b>1.055</b>	<b>0.82</b>

## 2) *Metrics*

Sensitifitas parameter *metrics* dilihat dari perubahan nilai MSE saat nilai *metrics* diubah-ubah namun nilai parameter lainnya tetap. Sensitifitas *metrics* dilihat pada tiga model terpilih yaitu model di dataran rendah, dataran sedang dan dataran tinggi. Nilai parameter tetap di dataran rendah adalah { *n\_neighbor* = 6, *algorithm* : *brute* }. Nilai parameter tetap di dataran sedang adalah { *n\_neighbor* = 4, *algorithm* : *kd\_tree* }. Nilai parameter tetap di dataran tinggi adalah { *n\_neighbor* : 2, *algorithm* : *brute* }. Perbandingan MSE setiap nilai *metrics* dapat dilihat pada Tabel 6.23.

Tabel 6.23. Sensitifitas Parameter *Metrics*

Metrics	MSE		
	DR	DS	DT
Euclidean	4.549	5.256	2.959
Manhattan	4.583	5.256	4.762
Chebshev	4.583	5.256	2.381
<b>SD</b>	<b>0.02</b>	<b>0</b>	<b>1.242</b>

### 3) *Algorithm*

Sensitifitas parameter *algorithm* dilihat dari perubahan nilai MSE saat nilai *algorithm* diubah-ubah namun nilai parameter lainnya tetap. Sensitifitas *metrics* dilihat pada tiga model terpilih yaitu model di dataran rendah, dataran sedang dan dataran tinggi. Nilai parameter tetap di dataran rendah adalah  $\{n\_neighbor : 6, metrics : euclidean\}$ . Nilai parameter tetap di dataran sedang adalah  $\{n\_neighbor : 4, metrics : euclidean\}$ . Nilai parameter tetap di dataran tinggi adalah  $\{n\_neighbor : 2, metrics : chebyshev\}$ . Perbandingan MSE setiap nilai *algorithm* dapat dilihat pada Tabel 6.24.

Tabel 6.24. Sensitifitas Parameter *Algorithm*

Algorithm	MSE		
	DR	DS	DT
Brute	4.549	14.276	2.381
Kd_tree	5.294	5.256	2.381
Ball_tree	5.294	5.256	2.381
<b>SD</b>	<b>0.43</b>	<b>5.208</b>	<b>0</b>

Hasil sensitifitas ketiga parameter pada Tabel 6.22, Tabel 6.23, dan Tabel 6.24 dapat diketahui bahwa parameter *n neighbor* menghasilkan nilai MSE yang lebih bervariasi dan memiliki perbedaan nilai yang cukup signifikan dibandingkan dengan dua parameter lainnya. Dapat dikatakan bahwa parameter *n neighbor* adalah parameter yang paling sensitif untuk algoritma *k-nearest neighbor regression*.

Meskipun parameter yang paling sensitif adalah parameter  $n$  *neighbor* namun, tidak menutup kemungkinan bahwa parameter lain juga bisa memiliki perbedaan yang signifikan. Pada Tabel 6.24 untuk model dataran sedang dapat dilihat bahwa MSE yang dihasilkan saat parameter *algorithm* bernilai *brute* memiliki perbedaan yang cukup jauh dengan kedua hasil lainnya.

## **BAB VII**

### **KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan dari seluruh tahapan pengerjaan tugas akhir yang telah dilakukan beserta saran untuk penelitian selanjutnya.

#### **7.1. Kesimpulan**

Kesimpulan yang didapatkan dalam tugas akhir ini antara lain:

1. Pembangunan model dengan metode KNNR di dataran rendah dilakukan di Kecamatan Kepanjen dengan proporsi pembagian data *train:test* sebesar 80:20 dan hanya melibatkan variabel *lag* 1. Model terbaik yang dihasilkan memiliki nilai RMSE 2.133 dan SMAPE sebesar 24.977%.
2. Pembangunan model dengan metode KNNR di dataran sedang dilakukan di Kecamatan Pakisaji dengan proporsi pembagian data *train:test* sebesar 80:20 dan hanya melibatkan variabel *lag* 1. Model terbaik yang dihasilkan memiliki nilai RMSE 2.205 dan SMAPE sebesar 15.944%.
3. Pembangunan model dengan metode KNNR di dataran tinggi dilakukan di Kecamatan Ngajum dengan proporsi pembagian data *train:test* sebesar 80:20 dan melibatkan variabel Angka Bebas Jentik, Curah Hujan dan variabel *lag* 1. Model terbaik yang dihasilkan memiliki nilai RMSE 1.543 dan SMAPE sebesar 30.716%.
4. Selama tahun 2010 hingga 2018, kasus demam berdarah didominasi oleh kecamatan yang berada di dataran sedang. Hasil peramalan menunjukkan adanya penurunan di tahun 2019 dan 2020. Meskipun terjadi penurunan, pemerintah dan Dinas Kesehatan Kabupaten Malang tetap harus waspada. Ada beberapa kecamatan yang harus menjadi perhatian khusus karena diprediksi akan terjadi



kasus demam berdarah dengan angka yang cukup tinggi. Kecamatan yang membutuhkan perhatian khusus adalah kecamatan yang terletak di dataran sedang. Tiga kecamatan yang diprediksi memiliki kasus demam berdarah yang tinggi di tahun 2020 adalah Kecamatan Sumbermanjing Wetan, Kecamatan Dampit dan Kecamatan Pakisaji.

## 7.2. Saran

Penelitian pada tugas akhir ini masih belum sempurna dan masih bisa dikembangkan. Beberapa saran untuk penelitian selanjutnya antara lain:

1. Pemilihan kecamatan pembentuk model tidak hanya berdasarkan uji korelasi saja. Namun bisa melibatkan standar deviasi atau uji pola data.
2. Memilih model terbaik sebaiknya tidak hanya dari RMSE terkecil pada kecamatan pembentuk model. Model terbaik pada kecamatan pembentuk model belum tentu menjadi model terbaik ketika diuji di kecamatan lain.
3. Pembentukan model dengan metode KNNR sebaiknya melibatkan parameter *leaf\_size* sebagai salah satu *tuning parameter*.

## DAFTAR PUSTAKA

- [1] G. Ginanjar, Demam Berdarah, PT. Mizan Publika.
- [2] Kementerian Kesehatan RI, “<https://www.kemkes.go.id>,” Kementerian Kesehatan Republik Indonesia, 17 Januari 2019. [Online]. Available: <https://www.kemkes.go.id/article/view/19011800001/kemenkes-imbau-seluruh-daerah-siaga-dbd.html>. [Diakses 31 Oktober 2019].
- [3] S. Hardiyanto dan D. A. A. Hapsari, “Januari, 72 Warga di Kabupaten Malang Terjangkit Demam Berdarah,” 31 Januari 2019. [Online]. Available: <https://radarmalang.id/januari-72-warga-di-kabupaten-malang-terjangkit-demam-berdarah/>. [Diakses 4 November 2019].
- [4] S. Desi, “Meningkat, Kabupaten Tertinggi Se-Jatim,” radar malang, 1 Februari 2019. [Online]. Available: <https://radarmalang.id/meningkat-kabupaten-tertinggi-se-jatim/>. [Diakses 4 November 2019].
- [5] J. Scavuzzo, F. Trucco, M. Espinosa, C. Tauro, M. Abril, C. M. Scavuzzo dan A. C. Frery, “Modeling Dengue vector population using remotely sensed data and,” *Acta Tropica*, vol. 185, pp. 167-175, 2018.
- [6] N. Altman, “An Introduction to Kernel and Nearest-Neighbor,” *The American Statistician*, vol. 46:3, pp. 175-185, 1992.
- [7] F. Rodríguez, S. S. Rivero dan J. A. Félix, “Exchange-rate forecasts with simultaneous nearest-neighbour methods: evidence from the EMS,” *International Journal of Forecasting*, vol. 15, no. 4, pp. 383-392, 1999.

- [8] A. T. Lora, J. M. R. Santos, A. G. Exposito, J. L. M. Ramos dan J. C. R. Santos, "Electricity Market Price Forecasting Based on Weighted Nearest Neighbors Techniques," *IEEE Transactions on Power Systems*, vol. 22, no. 3, pp. 1294 - 1301, 2007.
- [9] N. Zhang, A. Lin dan P. Shang, "Multidimensional -nearest neighbor model based on EEMD for financial time series forecasting," *Physica A: Statistical Mechanics and its Applications*, vol. 477, pp. 161-173, 2017.
- [10] F. Martinez, M. P. Frías, M. D. Pérez-Godoy dan A. J. Rivera, "Dealing with seasonality by narrowing the training set in time series forecasting with k NN," *Expert Systems With Applications*, vol. 103, pp. 38-48, 2018.
- [11] W. Anggraeni, I. P. A. A. Pramana, F. Samopa, E. Riksakomara, R. P. Wibowo, L. Condro T. dan Pudjiadi, "Forecasting The Number Of Dengue Fever Cases in Malang Regency Indonesia Using Fuzzy Inference System Models," *Journal of Theoretical and Applied Information Technology*, vol. 95, no. 1, 2017.
- [12] F. Martínez, M. D. Pérez, M. P. Frías dan A. J. Rivera, "A methodology for applying k-nearest neighbor to time," dalam *Artificial Intelligence Review*, 2017.
- [13] N. J. Johannesen, M. Kolhe dan M. Goodwin, "Relative evaluation of regression tools for urban area electrical energy," *Journal of Cleaner Production*, vol. 218, pp. 555-564, 2019.
- [14] A. Gama dan F. Betty, "Analisis Faktor Risiko Kejadian Demam Berdarah Dengue di Desa Mojosongo Kabupaten Boyolali," *Eksplanasi*, vol. 5, no. 2, 2010.
- [15] D. C. Frechtling, *Forecasting Tourism Demand.*, New York: Routledge, 2001.

- [16] V. Kotu dan B. Deshpande, *Data Science Concept and Practice*, Cambridge: Morgan Kaufmann, 2019.
- [17] S. Jain, S. Shukla dan R. Wadhvani, "Dynamic selection of normalization techniques using data complexity measures," *Expert System with Applications*, vol. 106, pp. 252-262, 2018.
- [18] I. Pratama, E. A. Permanasari, . I. Ardiyanto dan R. Indrayani, "A Review of Missing Values Handling Methods on Time-Series Data," dalam *2016 International Conference on Information Technology Systems and Innovation (ICITSI)*, Bandung – Bali, 2016.
- [19] A. dhevi, "Imputing Missing Values Using Inverse Distance Weighted Interpolation for Time Series Data," dalam *Sixth International Conference on Advanced Computing (ICoAC)*, 2014.
- [20] A. M. Bayen dan T. Siau, "Interpolation," dalam *An Introduction to MATLAB® Programming and Numerical Methods for Engineers*, 2015, p. 211–223.
- [21] S. P. & S. P. Venkateshan, "Interpolation," dalam *Computational Methods in Engineering*, 2014, p. 213–254.
- [22] M. J. Zaki dan W. Meira Jr., *Data Mining and Analysis Fundamental Concepts and Algorithm*, New York: Cambridge University Press, 2014.
- [23] J. Jeffers, J. Reinders dan A. Sondani, "Machine learning," dalam *Intel Xeon Phi Processor High Performance Programming*, Morgan Kaufmann, 2016, pp. 527-548.
- [24] J. jeffers, J. Reinders dan A. Sodani, "Machine Learning," dalam *Intel Xeon Phi Processor High Performance Programming (second edition)*, 2016, pp. 527-548.

- [25] G. V., D. E. dan B. M., "Fast k nearest neighbor search using GPU," dalam *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2008.
- [26] I. H. Witten, E. Frank, M. A. Hall dan C. J. Pal, "Algorithms: The Basic Method," dalam *Data Mining (Fourth Edition)*, Morgan Kaufmann, 2017, pp. 91-160.
- [27] M. Lawrence, M. O'Connor dan B. Edmundson, "A Field Study of Sales Forecasting Accuracy and Processes," *European Journal of Operational Research* , no. 122, pp. 151-60, 1988.
- [28] S. Jaganathan dan P. Prakash, "A combination-based forecasting method for the M4-competition," *International Journal of Forecasting*, 2019.
- [29] c. T. dan R. R. Draxler, "Root Mean Square Error (RMSE) or mean absolute error (MAE)?-Arguments against Avoiding RMSE in the Literature," *Geoscientific model development* 7.3., pp. 1247-1250, 2014.
- [30] R. Mehra dan shallu, "Breast Cancer Histology Images Classification Training from scratch or transfer Learning," *The Korean Institute of Communications and Information Sciences*, pp. 247-254, 2018.

## BIODATA PENULIS



Penulis bernama lengkap Berta Kartika Chandra dilahirkan di Kediri, 23 September 1997. Penulis telah menempuh pendidikan formal sejak tahun 2004 yaitu di TK Dharma Wanita, SD Negeri Mojojoto VI pada jenjang Sekolah Dasar, SMP Negeri 1 Kota Kediri pada jenjang sekolah menengah pertama dan dan SMA Negeri 2 Kota Kediri pada jenjang sekolah menengah atas.

Setelah lulus SMA pada tahun 2016, penulis melanjutkan pendidikan di Departemen Sistem Informasi Fakultas Teknologi Informasi dan Komunikasi – Institut Teknologi Sepuluh Nopember (ITS) Surabaya melalui jalur SBMPTN dan terdaftar sebagai mahasiswa dengan NRP 05211640000103. Selama menjadi mahasiswa penulis aktif di bidang akademik maupun non-akademik. Pada bidang akademik, penulis mengikuti beberapa perlombaan. Pencapaian yang pernah diraih penulis dalam bidang perlombaan akademik adalah juara II GEMASTIK 11 cabang lomba *data mining*. Penulis juga aktif di beberapa kegiatan non-akademik seperti kepanitiaan dan keterlibatan proyek. Kepanitiaan yang pernah diikuti penulis adalah ISE (*Information System Expo*) sebagai staff kompetisi. Penulis pernah terlibat dalam beberapa proyek salah satunya adalah proyek Sistem Pemerintahan Berbasis Elektronik atau SPBE di Kabupaten Kediri sebagai surveyor.

Penulis memiliki ketertarikan di bidang pengolahan data pada saat tahun kedua mahasiswa sehingga, penulis memutuskan untuk mengambil bidang minat Rekayasa Data dan Intelegensi Bisnis (RDIB) dengan topik peramalan. Jika ingin memberikan kritik dan saran terkait tugas akhir ini, dapat menghubungi penulis melalui email [chandraberta@gmail.com](mailto:chandraberta@gmail.com).

## LAMPIRAN A

**Tabel A.1. Hasil uji korelasi kasus demam berdarah di dataran rendah**

	<b>KJ</b>	<b>DM</b>	<b>GL</b>	<b>BL</b>	<b>TR</b>
<b>KJ</b>	1.00	0.52	0.64	0.56	0.80
<b>DM</b>	0.52	1.00	0.42	0.40	0.50
<b>GL</b>	0.64	0.42	1.00	0.39	0.54
<b>BL</b>	0.56	0.40	0.39	1.00	0.34
<b>TR</b>	0.80	0.50	0.54	0.34	1.00

**Tabel A.2. Hasil Uji Korelasi di Dataran Sedang**

	<b>TP</b>	<b>PS</b>	<b>LW</b>	<b>SS</b>	<b>WJ</b>	<b>DP</b>	<b>SW</b>	<b>KP</b>
<b>TP</b>	1.00	0.29	0.45	0.10	0.30	0.33	0.29	0.27
<b>PS</b>	0.29	1.00	0.38	0.35	0.49	0.52	0.43	0.40
<b>LW</b>	0.45	0.38	1.00	0.00	0.53	0.19	0.24	0.41
<b>SS</b>	0.10	0.35	0.00	1.00	0.20	0.48	0.50	0.20
<b>WJ</b>	0.30	0.49	0.53	0.20	1.00	0.45	0.39	0.23
<b>DP</b>	0.33	0.52	0.19	0.48	0.45	1.00	0.66	0.09
<b>SW</b>	0.29	0.43	0.24	0.50	0.39	0.66	1.00	0.10
<b>KP</b>	0.27	0.40	0.41	0.20	0.23	0.09	0.10	1.00

**Tabel A.3. Hasil Uji Korelasi di Dataran Tinggi**

	<b>NJ</b>	<b>PK</b>	<b>JB</b>
<b>NJ</b>	1.00	0.69	0.48
<b>PK</b>	0.69	1.00	0.28
<b>JB</b>	0.48	0.28	1.00

### **Keterangan Kode Kecamatan**

KJ	Kepanjen	SS	Singosari
DM	Donomulyo	WJ	Wajak
GL	Gondanglegi	DP	Dampit
BL	Bululawang	SW	Sumbermanjing Wetan
TR	Turen	KP	Karangploso
TP	Tumpang	NJ	Ngajum
PS	Pakisaji	PK	Poncokusumo
LW	Lawang	JB	Jabung



## LAMPIRAN B

**Tabel B.1. Hasil Peramalan Jumlah Penduduk di Kecamatan Ngajum**

<b>Periode</b>	<b>Peramalan</b>
Jan-2019	51009.66037
Feb-2019	51035.62647
Mar-2019	51061.59258
Apr-2019	51087.55868
Mei-2019	51113.52478
Jun-2019	51139.49089
Jul-2019	51165.45699
Agt-2019	51191.4231
Sep-2019	51217.3892
Okt-2019	51243.35531
Nov-2019	51269.32141
Des-2019	51295.28751

<b>Periode</b>	<b>Peramalan</b>
Jan-2020	51321.25362
Feb-2020	51347.21972
Mar-2020	51373.18583
Apr-2020	51399.15193
Mei-2020	51425.11804
Jun-2020	51451.08414
Jul-2020	51477.05024
Agt-2020	51503.01635
Sep-2020	51528.98245
Okt-2020	51554.94856
Nov-2020	51580.91466
Des-2020	51606.88077

**Tabel B.2. Hasil Peramalan Suhu (Stasiun BMKG Karangates)**

<b>Periode</b>	<b>Peramalan</b>
Jan-2019	51009.66037
Feb-2019	51035.62647
Mar-2019	51061.59258
Apr-2019	51087.55868
Mei-2019	51113.52478
Jun-2019	51139.49089
Jul-2019	51165.45699
Agt-2019	51191.4231
Sep-2019	51217.3892
Okt-2019	51243.35531
Nov-2019	51269.32141
Des-2019	51295.28751

<b>Periode</b>	<b>Peramalan</b>
Jan-2020	51321.25362
Feb-2020	51347.21972
Mar-2020	51373.18583
Apr-2020	51399.15193
Mei-2020	51425.11804
Jun-2020	51451.08414
Jul-2020	51477.05024
Agt-2020	51503.01635
Sep-2020	51528.98245
Okt-2020	51554.94856
Nov-2020	51580.91466
Des-2020	51606.88077

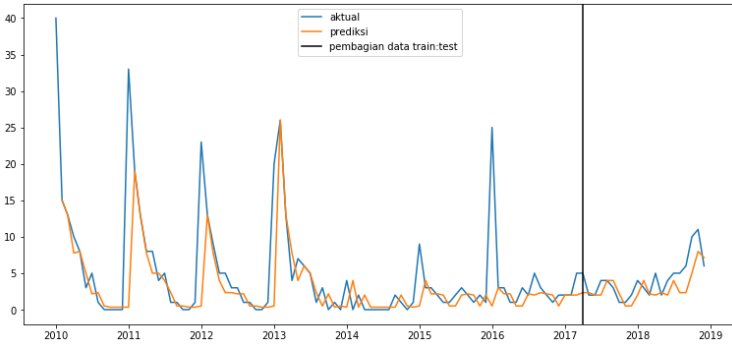
**Tabel B.3. Hasil Peramalan Kelembaban Udara (Stasiun BMKG Karangates)**

<b>Periode</b>	<b>Peramalan</b>
Jan-2019	86.47267146
Feb-2019	86.76349649
Mar-2019	85.1156201
Apr-2019	82.43619148
Mei-2019	80.6074905
Jun-2019	82.26771524
Jul-2019	80.17924652
Agt-2019	78.92634241
Sep-2019	78.34648483
Okt-2019	78.98521727
Nov-2019	83.89423043
Des-2019	84.82873214

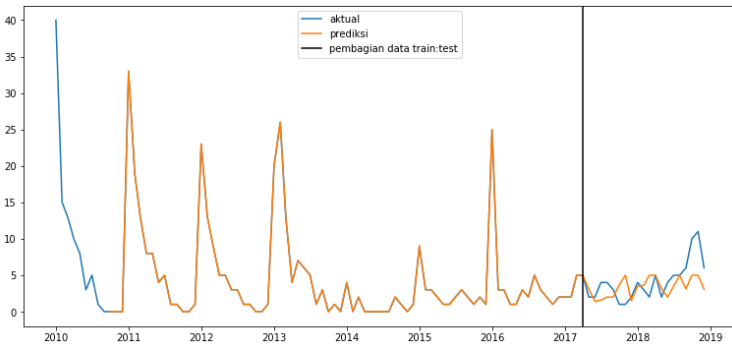
<b>Periode</b>	<b>Peramalan</b>
Jan-2020	86.31063477
Feb-2020	86.09341812
Mar-2020	84.82007327
Apr-2020	82.83172021
Mei-2020	80.6594614
Jun-2020	81.93071378
Jul-2020	80.8796692
Agt-2020	79.44413167
Sep-2020	78.04651352
Okt-2020	79.69949712
Nov-2020	85.02384038
Des-2020	84.53038781

## LAMPIRAN C

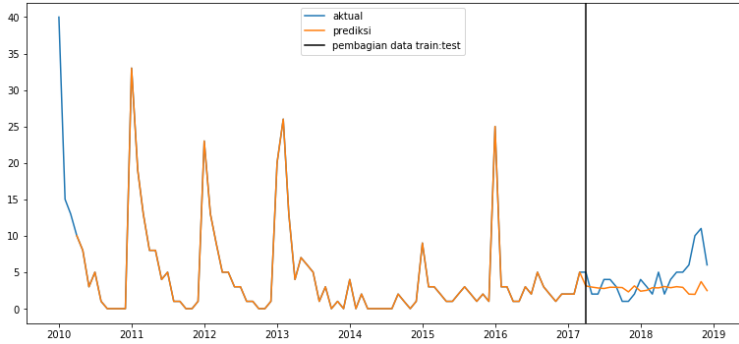
### C.1. Perbandingan Skenario Terbaik di Dataran Rendah



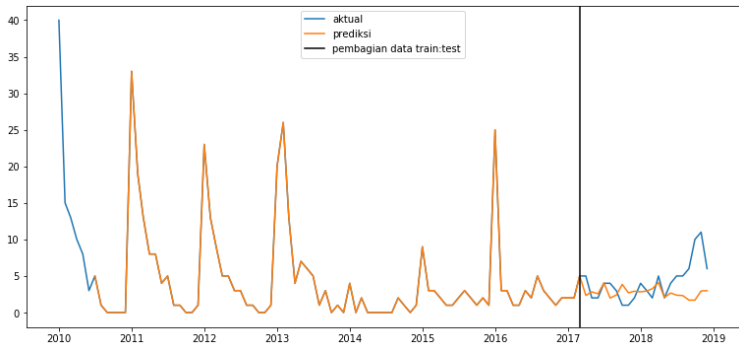
**Gambar C.1. Skenario V0L01 di Dataran Rendah**



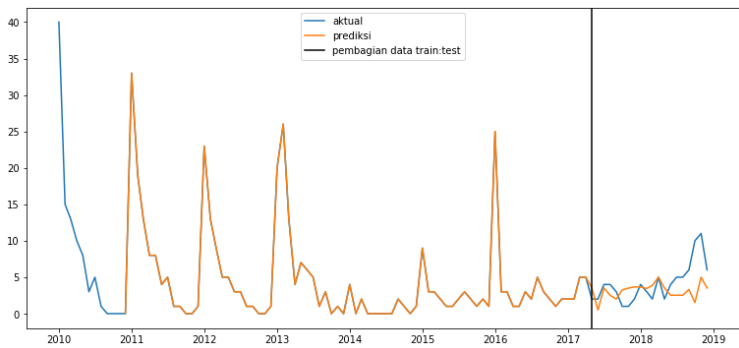
**Gambar C.2. Skenario V1L09 di Dataran Rendah**



**Gambar C.3. Skenario V2L02 di Dataran Rendah**

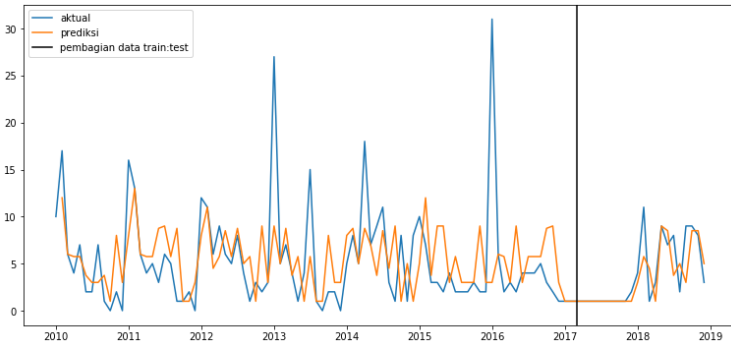


**Gambar C.4. Skenario V3L06 di Dataran Rendah**

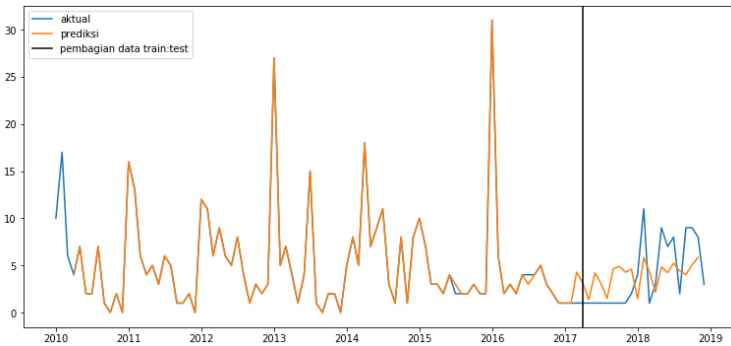


**Gambar C.5. Skenario VAL11 di Dataran Rendah**

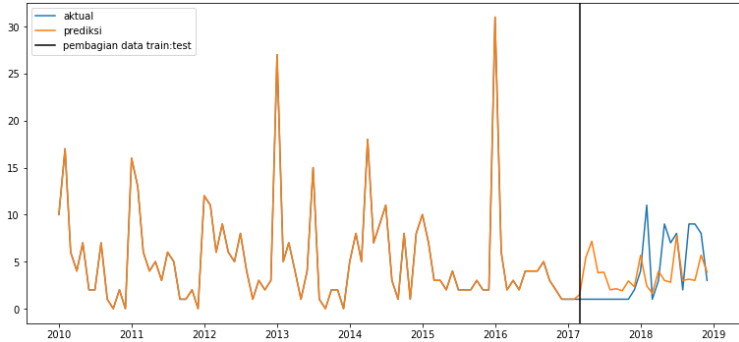
## C.2. Perbandingan Skenario Terbaik di Dataran Sedang



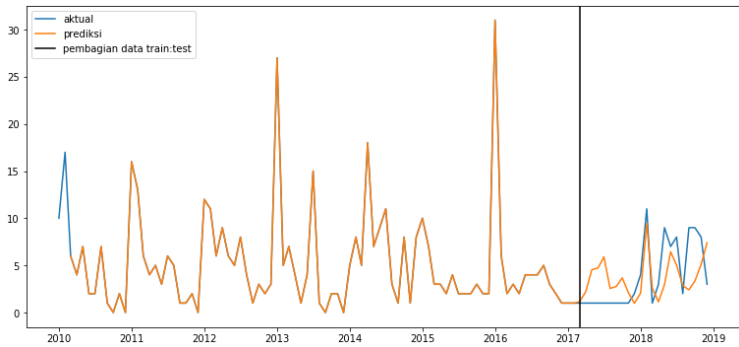
**Gambar C.6. Skenario V0L01 di Dataran Sedang**



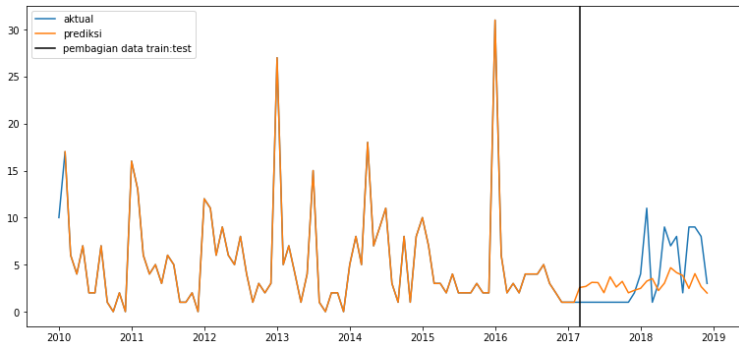
**Gambar C.7. Skenario V1L03 di Dataran Sedang**



**Gambar C.8. Skenario V2L00 di Dataran Sedang**

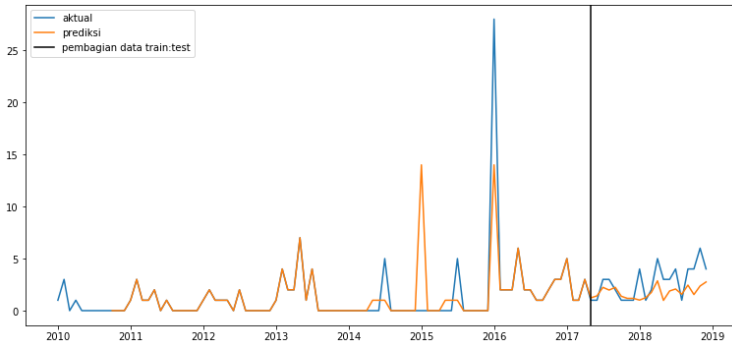


**Gambar C.9. Skenario V3L02 di Dataran Sedang**

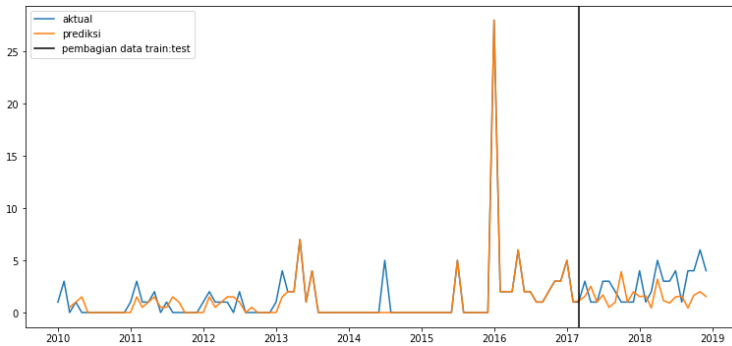


**Gambar C.10. Skenario VAL001 di Dataran Sedang**

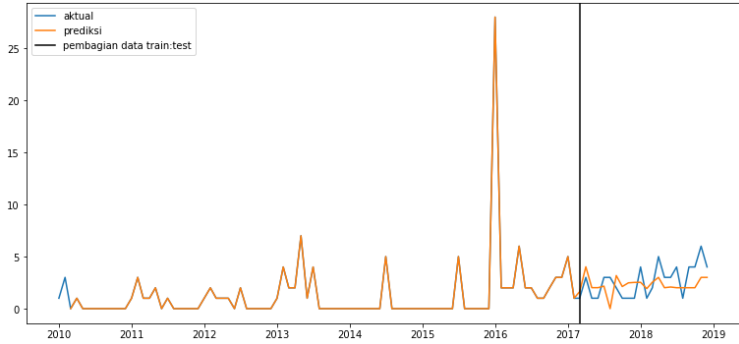
### C.3. Perbandingan Skenario Terbaik di Dataran Tinggi



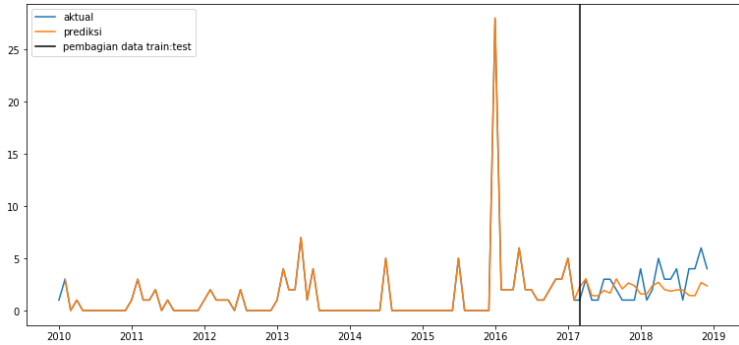
**Gambar C.11. Skenario V0L09 di Dataran Tinggi**



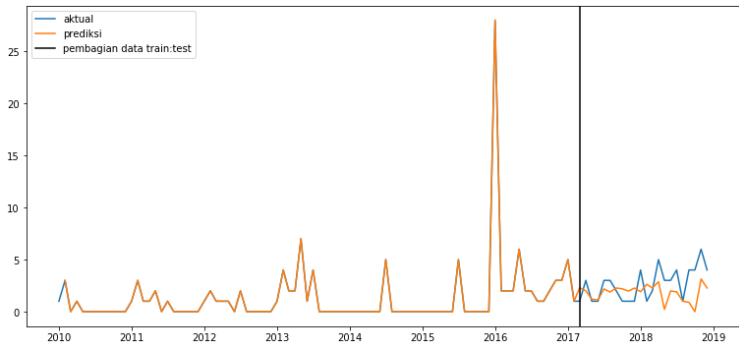
**Gambar C.12. Skenario V1L02 di Dataran Tinggi**



**Gambar C.13. Skenario V2L02 di Dataran Tinggi**



**Gambar C.14. Skenario V3L01 di Dataran Tinggi**



**Gambar C.15. Skenario VAL01 di Dataran Tinggi**



## LAMPIRAN D

**Tabel D.1. Sampel Dataset Cuaca BMKG**

<b>Tanggal</b>	<b>Tavg</b>	<b>RH_avg</b>	<b>RR</b>	<b>ff_avg</b>
01-01-2010	26.2	86	1	1
02-01-2010	26.9	85	10	0
03-01-2010	27.3	84	6	1
04-01-2010	27.2	84	10	1
05-01-2010	26.3	83	1	0
06-01-2010	26.8	83	0	0
07-01-2010	26.4	86	0	0
08-01-2010	25.7	88	15	0
09-01-2010	24.2	92	12	0
10-01-2010	25.6	88	7	0

**Keterangan:**

Tavg: Temperatur rata-rata (°C)

RH\_avg: Kelembapan rata-rata (%)

RR: Curah hujan (mm)

ff\_avg: Kecepatan angin rata-rata (m/s)

**Tabel D.2. Data Cuaca Hasil Praproses**

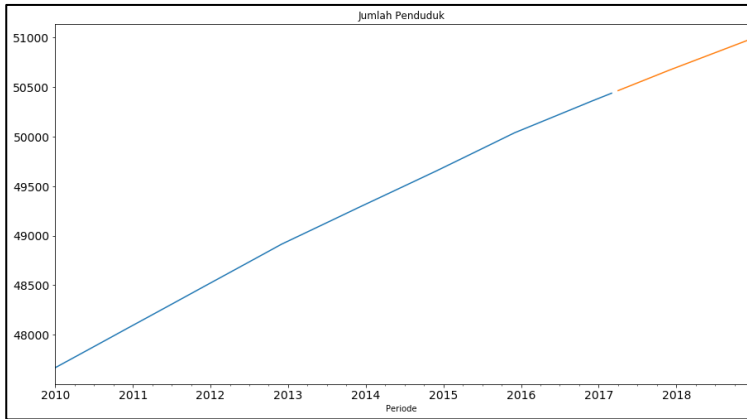
<b>Periode</b>	<b>SU</b>	<b>KU</b>	<b>CH</b>	<b>KA</b>
Jan-10	26.02	84.74	10.52	0.84
Feb-10	26.32	84.64	12.04	0.86
Mar-10	26.39	85	8.65	0.55
Apr-10	26.29	86.17	10.37	0.3
May-10	26.57	85.61	7.35	0.58
Jun-10	25.62	84.1	9.23	1.59
Jul-10	25.35	83.52	2.16	1.77
Aug-10	25.35	81.71	3.94	1.68
Sep-10	25.73	83.93	9.40	1.37
Oct-10	26.19	82.03	12.13	1.71
Nov-10	26	84.53	17.53	1.23

Dec-10	25.83	83.39	8.45	1.77
Jan-11	25.89	83.13	7.42	2.48
Feb-11	25.7	82.71	6.50	1.57
Mar-11	25.51	85.26	4.45	1.16
Apr-11	25.63	85.4	10.60	1.37
May-11	25.74	82.87	4.13	1.19
Jun-11	24.13	79.73	0.73	1.87
Jul-11	23.94	78.45	0.03	1.71
Aug-11	23.86	75.13	0.00	2.06
Sep-11	25.14	74.77	0.00	2.17
Oct-11	26.3	74.55	5.90	1.52
Nov-11	26	84.13	8.37	0.83
Dec-11	26.04	84.81	10.94	1.03
Jan-12	25.47	85.87	13.45	1.42
Feb-12	25.79	84.83	15.31	0.93
Mar-12	25.7	82.68	9.61	1.03
Apr-12	25.82	82.9	5.83	1.07
May-12	25.54	81.55	2.84	1.6
Jun-12	24.65	80	1.10	1.2
Jul-12	23.67	80.61	0.48	1.06
Aug-12	23.85	76.84	0.00	1.97
Sep-12	25.15	76.7	0.00	1.73
Oct-12	26.64	76.52	2.55	1.61
Nov-12	26.73	82.17	6.97	0.73
Dec-12	25.8	86.77	16.90	0.23
Jan-13	25.92	86.68	15.03	0.74
Feb-13	25.91	85.20	13.51	0.82
Mar-13	26.12	84.72	9.80	0.73
Apr-13	26.26	84.25	11.03	0.68
May-13	26.11	81.93	5.15	0.90
Jun-13	25.16	81.63	4.84	1.10
Jul-13	24.79	84.39	2.80	1.06
Aug-13	24.55	78.78	1.56	1.23
Sep-13	25.03	77.53	7.33	0.97
Oct-13	27.4	76.61	5.32	0.87
Nov-13	25.9	85.2	9.20	0.4
Dec-13	26.13	85.14	14.57	0.23
Jan-14	25.91	85.26	10.47	0.61
Feb-14	25.91	85.20	13.51	0.61
Mar-14	26.02	85.04	9.80	0.13

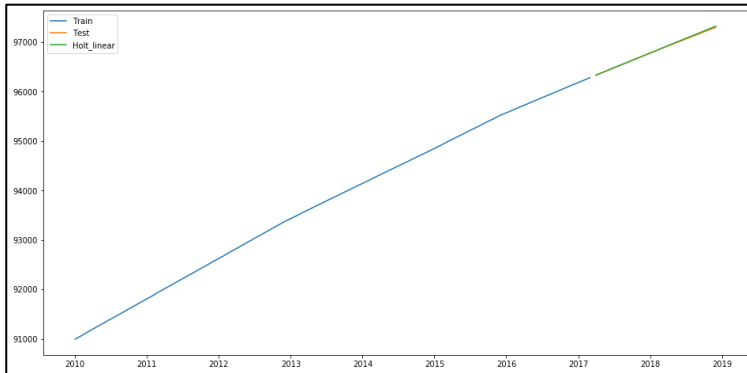
Apr-14	26.35	85.3	11.03	0.37
May-14	26.76	81.06	5.15	1.03
Jun-14	25.76	81.57	4.84	0.53
Jul-14	24.83	81	2.80	0.84
Aug-14	24.86	78.61	1.56	1.1
Sep-14	24.98	77.4	7.33	0.97
Oct-14	27.01	74.16	5.32	0.97
Nov-14	27.07	80.37	8.55	0.57
Dec-14	25.72	88.84	15.66	0.1
Jan-15	26.06	85.65	12.59	0.42
Feb-15	25.98	85.71	16.65	0.21
Mar-15	26.08	85.48	12.77	0.32
Apr-15	26.09	86.6	6.24	0.1
May-15	25.91	80.77	3.53	0.39
Jun-15	24.97	81.1	0.38	0.77
Jul-15	24.45	79.42	0.00	0.77
Aug-15	24.64	78.81	0.00	0.84
Sep-15	25.8	74.83	0.00	1.13
Oct-15	27.27	72.16	0.00	0.94
Nov-15	27.81	77.67	0.34	0.6
Dec-15	26.78	85.45	11.88	0.42
Jan-16	26.92	84.16	26.54	0.61
Feb-16	26.29	87.34	14.88	0.38
Mar-16	27.04	85.67	12.72	0.47
Apr-16	26.99	83.97	18.42	0.37
May-16	27.15	82.87	13.48	0.48
Jun-16	26.25	83.7	16.17	0.37
Jul-16	26.03	82.19	8.54	0.48
Aug-16	25.46	81.74	6.59	0.61
Sep-16	26.03	82.13	19.91	0.77
Oct-16	26.05	84.61	11.81	0.65
Nov-16	26.08	85.97	6.49	0.8
Dec-16	26.17	84	12.63	0.87
Jan-17	25.72	86	19.32	0.81
Feb-17	25.88	85.14	15.73	1.07
Mar-17	26.09	84.42	9.53	0.74
Apr-17	26.28	83.53	15.99	0.67
May-17	25.9	80.87	3.02	1.06

Jun-17	25.51	81.5	2.60	1
Jul-17	24.46	82.06	7.28	1.52
Aug-17	24.44	80.39	0.40	1.19
Sep-17	25.6	77.8	21.44	1.2
Oct-17	26.43	81.1	4.81	1.06
Nov-17	25.69	86.86	9.63	0.87
Dec-17	26.22	84.03	9.00	0.68
Jan-18	25.28	87.29	9.83	0.65
Feb-18	25.39	86	13.44	0.71
Mar-18	26	84.52	10.89	0.81
Apr-18	26.73	81.17	9.77	0.9
May-18	25.96	78.94	1.68	1.03
Jun-18	24.99	81.27	3.67	0.93
Jul-18	24.25	78.65	1.08	1.23
Aug-18	24.27	76.87	0.00	1.39
Sep-18	25.5	75.73	0.59	1.03
Oct-18	26.58	74.94	0.04	1.39
Nov-18	26.49	82.28	10.53	1
Dec-18	26.14	85.45	10.41	0.87

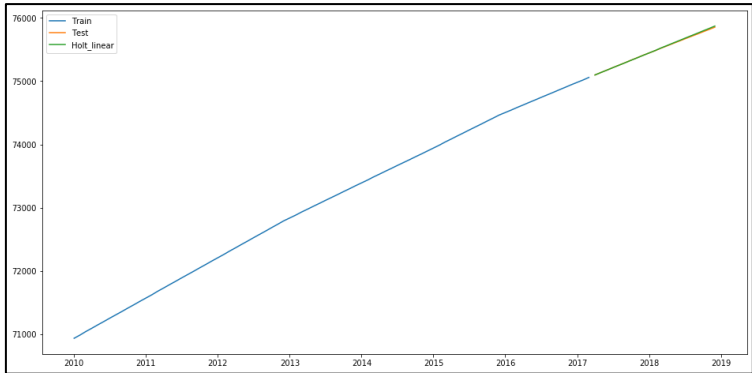
## LAMPIRAN E



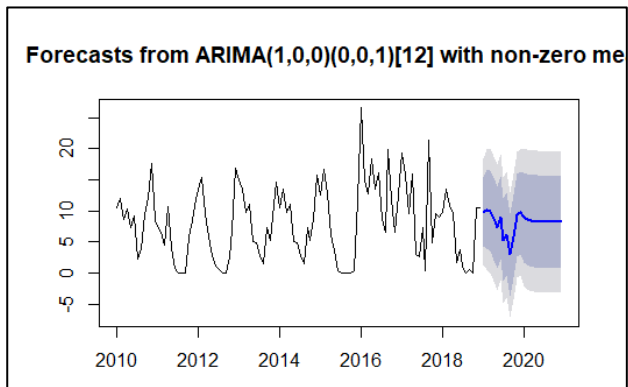
**Gambar E.1. Peramalan Jumlah Penduduk di Kecamatan Ngajum**



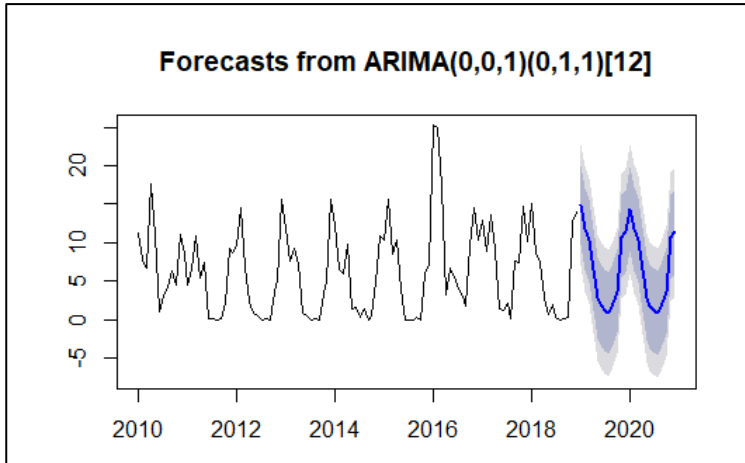
**Gambar E.2. Peramalan Jumlah Penduduk di Kecamatan Jabung**



**Gambar E.3. Peramalan Jumlah Penduduk di Kecamatan Poncokusumo**



**Gambar E.4. Peramalan Curah Hujan di Kecamatan Ngajum**



**Gambar E.5. Peramalan Curah Hujan di Kecamatan Jabung dan Poncokusumo**

**Tabel E.1. Hasil Peramalan Jumlah Penduduk di Kecamatan Ngajum**

Periode	Peramalan JP
Jan-2019	50997.03265
Feb-2019	51021.9981
Mar-2019	51046.96356
Apr-2019	51071.92902
Mei-2019	51096.89447
Jun-2019	51121.85993
Jul-2019	51146.82539
Agt-2019	51171.79085
Sep-2019	51196.7563
Okt-2019	51221.72176
Nov-2019	51246.68722
Des-2019	50997.03265

Periode	Peramalan JP
Jan-2020	51271.65267
Feb-2020	51296.61813
Mar-2020	51321.58359
Apr-2020	51346.54904
Mei-2020	51371.5145
Jun-2020	51396.47996
Jul-2020	51421.44541
Agt-2020	51446.41087
Sep-2020	51471.37633
Okt-2020	51496.34178
Nov-2020	51521.30724
Des-2020	51546.2727

**Tabel E.2. Hasil Peramalan Jumlah Penduduk di Kecamatan Jabung**

<b>Periode</b>	<b>Peramalan JP</b>
Jan-2019	75890.83541
Feb-2019	75928.002
Mar-2019	75965.16859
Apr-2019	76002.33517
Mei-2019	76039.50176
Jun-2019	76076.66835
Jul-2019	76113.83494
Agt-2019	76151.00152
Sep-2019	76188.16811
Okt-2019	76225.3347
Nov-2019	76262.50128
Des-2019	75890.83541

<b>Periode</b>	<b>Peramalan JP</b>
Jan-2020	76299.66787
Feb-2020	76336.83446
Mar-2020	76374.00104
Apr-2020	76411.16763
Mei-2020	76448.33422
Jun-2020	76485.5008
Jul-2020	76522.66739
Agt-2020	76559.83398
Sep-2020	76597.00057
Okt-2020	76634.16715
Nov-2020	76671.33374
Des-2020	76708.50033

**Tabel E.3. Hasil Peramalan Jumlah Penduduk di Kecamatan Puncokusumo**

<b>Periode</b>	<b>Peramalan JP</b>
Jan-2019	97350.09366
Feb-2019	97397.76443
Mar-2019	97445.43521
Apr-2019	97493.10599
Mei-2019	97540.77676
Jun-2019	97588.44754
Jul-2019	97636.11831
Agt-2019	97683.78909
Sep-2019	97731.45987
Okt-2019	97779.13064
Nov-2019	97826.80142
Des-2019	97350.09366

<b>Periode</b>	<b>Peramalan JP</b>
Jan-2020	97874.47219
Feb-2020	97922.14297
Mar-2020	97969.81374
Apr-2020	98017.48452
Mei-2020	98065.1553
Jun-2020	98112.82607
Jul-2020	98160.49685
Agt-2020	98208.16762
Sep-2020	98255.8384
Okt-2020	98303.50918
Nov-2020	98351.17995
Des-2020	98398.85073



**Tabel E.4. Hasil Peramalan Curah Hujan di Kecamatan Ngajum**

<b>Periode</b>	<b>Peramalan CH</b>
Jan-2019	9.831010618
Feb-2019	10.21350724
Mar-2019	10.01275208
Apr-2019	8.716275407
Mei-2019	7.282216126
Jun-2019	8.928112707
Jul-2019	5.292808039
Agt-2019	6.24327651
Sep-2019	2.91571253
Okt-2019	5.870607761
Nov-2019	9.320806598
Des-2019	9.831010618

<b>Periode</b>	<b>Peramalan CH</b>
Jan-2020	9.669815182
Feb-2020	8.9792586
Mar-2020	8.609926263
Apr-2020	8.412395199
Mei-2020	8.3067491
Jun-2020	8.250246096
Jul-2020	8.220026435
Agt-2020	8.203863967
Sep-2020	8.195219749
Okt-2020	8.190596537
Nov-2020	8.188123891
Des-2020	8.186801439

**Tabel E.5. Hasil Peramalan Curah Hujan di Kecamatan Jabung dan Poncokusumo**

<b>Periode</b>	<b>Peramalan CH</b>
Jan-2019	14.90863077
Feb-2019	11.75523156
Mar-2019	10.27353746
Apr-2019	6.526039022
Mei-2019	2.85410564
Jun-2019	1.761304617
Jul-2019	1.196757604
Agt-2019	0.785089831
Sep-2019	2.049115253
Okt-2019	3.630440647
Nov-2019	10.7446887
Des-2019	14.90863077

<b>Periode</b>	<b>Peramalan CH</b>
Jan-2020	11.4175033
Feb-2020	14.3789808
Mar-2020	11.75523156
Apr-2020	10.27353746
Mei-2020	6.526039022
Jun-2020	2.85410564
Jul-2020	1.761304617
Agt-2020	1.196757604
Sep-2020	0.785089831
Okt-2020	2.049115253
Nov-2020	3.630440647
Des-2020	10.7446887

## LAMPIRAN F

**Tabel F.1. Peramalan 24 Periode di Kecamatan Kepanjen**

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2019	5
Feb-2019	2
Mar-2019	2
Apr-2019	2
Mei-2019	2
Jun-2019	2
Jul-2019	2
Agt-2019	2
Sep-2019	2
Okt-2019	2
Nov-2019	2
Des-2019	2

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2020	2
Feb-2020	2
Mar-2020	2
Apr-2020	2
Mei-2020	2
Jun-2020	2
Jul-2020	2
Agt-2020	2
Sep-2020	2
Okt-2020	2
Nov-2020	2
Des-2020	2

**Tabel F.2. Peramalan 24 Periode di Kecamatan Gondanglegi**

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2019	8
Feb-2019	5
Mar-2019	2
Apr-2019	2
Mei-2019	2
Jun-2019	2
Jul-2019	2
Agt-2019	2
Sep-2019	2
Okt-2019	2
Nov-2019	2
Des-2019	2

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2020	2
Feb-2020	2
Mar-2020	2
Apr-2020	2
Mei-2020	2
Jun-2020	2
Jul-2020	2
Agt-2020	2
Sep-2020	2
Okt-2020	2
Nov-2020	2
Des-2020	2

**Tabel F.3. Peramalan 24 Periode di Kecamatan Donomulyo**

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2019	15
Feb-2019	13
Mar-2019	8
Apr-2019	5
Mei-2019	2
Jun-2019	2
Jul-2019	2
Agt-2019	2
Sep-2019	2
Okt-2019	2
Nov-2019	2
Des-2019	2

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2020	2
Feb-2020	2
Mar-2020	2
Apr-2020	2
Mei-2020	2
Jun-2020	2
Jul-2020	2
Agt-2020	2
Sep-2020	2
Okt-2020	2
Nov-2020	2
Des-2020	2

**Tabel F.4. Peramalan 24 Periode di Kecamatan Bululawang**

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2019	8
Feb-2019	5
Mar-2019	2
Apr-2019	2
Mei-2019	2
Jun-2019	2
Jul-2019	2
Agt-2019	2
Sep-2019	2
Okt-2019	2
Nov-2019	2
Des-2019	2

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2020	2
Feb-2020	2
Mar-2020	2
Apr-2020	2
Mei-2020	2
Jun-2020	2
Jul-2020	2
Agt-2020	2
Sep-2020	2
Okt-2020	2
Nov-2020	2
Des-2020	2

**Tabel F.5. Peramalan 24 Periode di Kecamatan Turen**

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2019	6

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2020	2

Feb-2019	3
Mar-2019	2
Apr-2019	2
Mei-2019	2
Jun-2019	2
Jul-2019	2
Agt-2019	2
Sep-2019	2
Okt-2019	2
Nov-2019	2
Des-2019	2

Feb-2020	2
Mar-2020	2
Apr-2020	2
Mei-2020	2
Jun-2020	2
Jul-2020	2
Agt-2020	2
Sep-2020	2
Okt-2020	2
Nov-2020	2
Des-2020	2

**Tabel F.6. Peramalan 24 Periode di Kecamatan Pakisaji**

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2019	9
Feb-2019	8
Mar-2019	5
Apr-2019	9
Mei-2019	8
Jun-2019	5
Jul-2019	9
Agt-2019	8
Sep-2019	5
Okt-2019	9
Nov-2019	8
Des-2019	5

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2020	9
Feb-2020	8
Mar-2020	5
Apr-2020	9
Mei-2020	8
Jun-2020	5
Jul-2020	9
Agt-2020	8
Sep-2020	5
Okt-2020	9
Nov-2020	8
Des-2020	5

**Tabel F.7. Peramalan 24 Periode di Kecamatan Tumpang**

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2019	1
Feb-2019	1
Mar-2019	1
Apr-2019	1

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2020	1
Feb-2020	1
Mar-2020	1
Apr-2020	1

Mei-2019	1
Jun-2019	1
Jul-2019	1
Agt-2019	1
Sep-2019	1
Okt-2019	1
Nov-2019	1
Des-2019	1

Mei-2020	1
Jun-2020	1
Jul-2020	1
Agt-2020	1
Sep-2020	1
Okt-2020	1
Nov-2020	1
Des-2020	1

**Tabel F.8. Peramalan 24 Periode di Kecamatan Lawang**

Periode	Peramalan KDB
Jan-2019	1
Feb-2019	1
Mar-2019	1
Apr-2019	1
Mei-2019	1
Jun-2019	1
Jul-2019	1
Agt-2019	1
Sep-2019	1
Okt-2019	1
Nov-2019	1
Des-2019	1

Periode	Peramalan KDB
Jan-2020	1
Feb-2020	1
Mar-2020	1
Apr-2020	1
Mei-2020	1
Jun-2020	1
Jul-2020	1
Agt-2020	1
Sep-2020	1
Okt-2020	1
Nov-2020	1
Des-2020	1

**Tabel F.9. Peramalan 24 Periode di Kecamatan Singosari**

Periode	Peramalan KDB
Jan-2019	4
Feb-2019	2
Mar-2019	3
Apr-2019	3

Periode	Peramalan KDB
Jan-2020	3
Feb-2020	3
Mar-2020	3
Apr-2020	3

Mei-2019	3
Jun-2019	3
Jul-2019	3
Agt-2019	3
Sep-2019	3
Okt-2019	3
Nov-2019	3
Des-2019	3

Mei-2020	3
Jun-2020	3
Jul-2020	3
Agt-2020	3
Sep-2020	3
Okt-2020	3
Nov-2020	3
Des-2020	3

**Tabel F.10. Peramalan 24 Periode di Kecamatan Wajak**

Periode	Peramalan KDB
Jan-2019	2
Feb-2019	3
Mar-2019	3
Apr-2019	3
Mei-2019	3
Jun-2019	3
Jul-2019	3
Agt-2019	3
Sep-2019	3
Okt-2019	3
Nov-2019	3
Des-2019	3

Periode	Peramalan KDB
Jan-2020	3
Feb-2020	3
Mar-2020	3
Apr-2020	3
Mei-2020	3
Jun-2020	3
Jul-2020	3
Agt-2020	3
Sep-2020	3
Okt-2020	3
Nov-2020	3
Des-2020	3

**Tabel F.11. Peramalan 24 Periode di Kecamatan Dampit**

Periode	Peramalan KDB
Jan-2019	9
Feb-2019	8
Mar-2019	5
Apr-2019	9
Mei-2019	8
Jun-2019	5
Jul-2019	9

Periode	Peramalan KDB
Jan-2020	9
Feb-2020	8
Mar-2020	5
Apr-2020	9
Mei-2020	8
Jun-2020	5
Jul-2020	9

Agt-2019	8
Sep-2019	5
Okt-2019	9
Nov-2019	8
Des-2019	5

Agt-2020	8
Sep-2020	5
Okt-2020	9
Nov-2020	8
Des-2020	5

**Tabel F.12. Peramalan 24 Periode di Kecamatan Sumbermanjing Wetan**

Periode	Peramalan KDB
Jan-2019	6
Feb-2019	17
Mar-2019	16
Apr-2019	10
Mei-2019	16
Jun-2019	14
Jul-2019	9
Agt-2019	16
Sep-2019	14
Okt-2019	9
Nov-2019	16
Des-2019	14

Periode	Peramalan KDB
Jan-2020	9
Feb-2020	16
Mar-2020	14
Apr-2020	9
Mei-2020	16
Jun-2020	14
Jul-2020	9
Agt-2020	16
Sep-2020	14
Okt-2020	9
Nov-2020	16
Des-2020	14

**Tabel F.13. Peramalan 24 Periode di Kecamatan Karangploso**

Periode	Peramalan KDB
Jan-2019	6
Feb-2019	5
Mar-2019	3
Apr-2019	6
Mei-2019	5
Jun-2019	3
Jul-2019	6
Agt-2019	5
Sep-2019	3

Periode	Peramalan KDB
Jan-2020	6
Feb-2020	5
Mar-2020	3
Apr-2020	6
Mei-2020	5
Jun-2020	3
Jul-2020	6
Agt-2020	5
Sep-2020	3

Okt-2019	6
Nov-2019	5
Des-2019	3

Okt-2020	6
Nov-2020	5
Des-2020	3

**Tabel F.14. Peramalan 24 Periode di Kecamatan Ngajum**

Periode	Peramalan KDB
Jan-2019	1
Feb-2019	0
Mar-2019	0
Apr-2019	1
Mei-2019	0
Jun-2019	0
Jul-2019	0
Agt-2019	1
Sep-2019	1
Okt-2019	1
Nov-2019	0
Des-2019	16

Periode	Peramalan KDB
Jan-2020	2
Feb-2020	2
Mar-2020	1
Apr-2020	1
Mei-2020	2
Jun-2020	2
Jul-2020	1
Agt-2020	3
Sep-2020	3
Okt-2020	3
Nov-2020	3
Des-2020	3

**Tabel F.15. Peramalan 24 Periode di Kecamatan Poncokusumo**

Periode	Peramalan KDB
Jan-2019	1
Feb-2019	0
Mar-2019	0
Apr-2019	0
Mei-2019	0
Jun-2019	0
Jul-2019	0
Agt-2019	0
Sep-2019	1
Okt-2019	1
Nov-2019	0
Des-2019	9

Periode	Peramalan KDB
Jan-2020	1
Feb-2020	1
Mar-2020	1
Apr-2020	1
Mei-2020	1
Jun-2020	1
Jul-2020	1
Agt-2020	2
Sep-2020	2
Okt-2020	2
Nov-2020	2
Des-2020	2



**Tabel F.16. Peramalan 24 Periode di Kecamatan Jabung**

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2019	1
Feb-2019	0
Mar-2019	0
Apr-2019	0
Mei-2019	1
Jun-2019	0
Jul-2019	0
Agt-2019	0
Sep-2019	0
Okt-2019	1
Nov-2019	0
Des-2019	0

<b>Periode</b>	<b>Peramalan KDB</b>
Jan-2020	7
Feb-2020	1
Mar-2020	0
Apr-2020	0
Mei-2020	0
Jun-2020	0
Jul-2020	0
Agt-2020	0
Sep-2020	1
Okt-2020	1
Nov-2020	1
Des-2020	1

## LAMPIRAN G

**Tabel G.1. Pengujian Model Terbaik di Dataran Rendah**

Kecamatan	Skenario				
	VOL01	V1L09	V2L03	V3L06	VAL11
Kepanjen	2.13	2.56	2.92	3.13	2.86
Gondanglegi	2.53	2.68	2.41	2.37	2.46
Donomulyo	2.7	2.36	2.42	2.49	2.48
Bululawang	3.44	3.28	2.92	2.75	2.59
Turen	17.22	22.42	13.21	13.79	12.33
Pakisaji	5.71	5.74	5.15	4.98	5.13
tumpang	5.12	5.68	5.35	4.96	4.92
lawang	6.27	8.39	6.31	6.61	6.1
singosari	2.95	5.04	3.63	6.61	3.35
wajak	3.65	3.33	3.24	2.94	2.91
dampit	4.55	6.74	3.9	3.89	4.61
sumbermanjing wetan	6.52	15.62	7.2	8.59	5.94
karangploso	3.62	4.02	3.35	3.68	3.84
Ngajum	3.36	7.24	4.28	4.59	3.54
Poncokusumo	2.7	4.24	2.74	2.76	3.1
Jabung	2.11	2.4	2.29	2.12	2.07
<b>Median</b>	<b>3.53</b>	<b>4.64</b>	<b>3.49</b>	<b>3.785</b>	<b>3.445</b>

**Tabel G.2. . Pengujian Model Terbaik di Dataran Sedang**

Kecamatan	Skenario				
	VOL01	V1L03	V2L00	V3L02	VAL01
Kepanjen	7.02	3.35	7.47	6	5.44
Gondanglegi	2.57	18.16	2.88	2.53	2.52
Donomulyo	2.67	13.45	2.64	2.36	2.4
Bululawang	3.53	8.88	3.88	2.83	2.67
Turen	21.8	2.01	21.54	16.32	16.41
Pakisaji	2.21	3.12	3.63	3.23	3.42
tumpang	5.44	4.85	5.34	5.01	4.53
lawang	7.44	3.45	8.33	7.73	6.78
singosari	3.36	8.87	3.31	3.1	3.14
wajak	3.57	5.99	3.24	2.98	3.04

dampit	5.71	4.32	5.28	4.19	4.61
sumbermanjing wetan	10.74	2.72	9.29	8.82	9.02
karangploso	3.39	8.05	3.61	3.29	3.02
Ngajum	5.56	5.25	4.69	4.28	4.05
Poncokusumo	3.24	9.56	2.51	2.46	2.68
Jabung	2.05	16	2.18	1.77	1.91
<b>Median</b>	<b>3.55</b>	<b>5.62</b>	<b>3.755</b>	<b>3.26</b>	<b>3.28</b>

**Tabel G.3. . Pengujian Model Terbaik di Dataran Tinggi**

<b>Kecamatan</b>	<b>Skenario</b>				
	<b>V0L09</b>	<b>V1L02</b>	<b>V2L02</b>	<b>V3L01</b>	<b>VAL01</b>
Kepanjen	6.66	6.71	6.05	6.3	6.61
Gondanglegi	2.97	2.96	2.93	3.07	3.09
Donomulyo	2.75	2.69	2.45	2.52	3.87
Bululawang	3.54	3.95	3.27	3.35	2.63
Turen	15.26	15.27	16.33	15.45	14.58
Pakisaji	5.94	6.08	5.18	5.56	6.13
tumpang	5.7	6.06	5.69	5.97	5.99
lawang	8.47	8.24	7.69	7.82	7.8
singosari	3.1	3.25	2.65	2.49	2.62
wajak	3.88	3.97	3.64	3.77	3.93
dampit	4.27	4.28	2.92	3.23	3.83
sumbermanjing wetan	6.93	7.85	4.45	4.75	5.08
karangploso	3.73	3.75	3.97	3.87	3.76
Ngajum	1.55	2.02	1.54	1.62	1.79
Poncokusumo	2.82	2.83	2.34	2.42	2.62
Jabung	2.53	2.54	2.63	2.66	2.56
<b>Median</b>	<b>3.805</b>	<b>3.96</b>	<b>3.455</b>	<b>3.56</b>	<b>3.85</b>