

**PREDIKSI URBAN HEAT ISLAND MENGGUNAKAN ALGORITMA  
*GRADIENT TREE BOOSTING* (STUDI KASUS: KOTA MALANG)**

**SKRIPSI**

Oleh :  
**MUHAMMAD DANIYAL**  
NIM. 200605110025



**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2024**

**PREDIKSI URBAN HEAT ISLAND MENGGUNAKAN ALGORITMA  
*GRADIENT TREE BOOSTING* (STUDI KASUS: KOTA MALANG)**

**SKRIPSI**

Diajukan kepada:  
Universitas Islam Negeri Maulana Malik Ibrahim Malang  
Untuk memenuhi Salah Satu Persyaratan dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :  
**MUHAMMAD DANIYAL**  
**NIM. 200605110025**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2024**

**HALAMAN PERSETUJUAN**

**PREDIKSI URBAN HEAT ISLAND MENGGUNAKAN ALGORITMA  
GRADIENT TREE BOOSTING (STUDI KASUS: KOTA MALANG)**

**SKRIPSI**

**Oleh :  
MUHAMMAD DANIYAL  
NIM. 200605110025**

Telah Diperiksa dan Disetujui untuk Diuji:  
Tanggal: 15 Mei 2024

Pembimbing I,



Hani Nurhayati, M.T  
NIP. 19780625 200801 2 006

Pembimbing II,



Dr. Irwan Budi Santoso, M.Kom  
NIP. 19770103 201101 1 004

Mengetahui,

Ketua Program Studi Teknik Informatika

Fakultas Sains dan Teknologi

Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Faonni Kurniawan, M.MT, IPM  
NIP. 19771020 200912 1 001

**HALAMAN PENGESAHAN**

**PREDIKSI URBAN HEAT ISLAND MENGGUNAKAN ALGORITMA  
GRADIENT TREE BOOSTING (STUDI KASUS: KOTA MALANG)**

**SKRIPSI**

**Oleh :  
MUHAMMAD DANIYAL  
NIM. 200605110025**

Telah Dipertahankan di Depan Dewan Penguji Skripsi  
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Sarjana Komputer ( S.Kom )  
Tanggal: 17 Mei 2024

**Susunan Dewan Penguji**

Ketua Penguji : Dr. Yunifa Miftachul Arif, M. T  
NIP. 19830616 201101 1 004

Anggota Penguji I : Dr. Fresy Nugroho, M.T  
NIP. 19710722 201101 1 001

Anggota Penguji II : Hani Nurhayati, M.T  
NIP. 19780625 200801 2 006

Anggota Penguji III : Dr. Irwan Budi Santoso, M.Kom  
NIP. 19770103 201101 1 004

(  
(  
(  
(

Mengetahui dan Mengesahkan,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Fachrul Kurniawan, M.MT, IPM  
NIP. 19771020 200912 1 001

## PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Muhammad Daniyal  
NIM : 200605110025  
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika  
Judul Skripsi : Prediksi *Urban Heat Island* Menggunakan  
Algoritma *Gradient Tree Boosting* (Studi Kasus:  
Kota Malang)

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 17 Mei 2024  
Yang membuat pernyataan,



Muhammad Daniyal  
NIM. 200605110025

## **MOTTO**

*“Jika kau tak tahan menahan lelahnya belajar, maka kau harus sanggup menahan pedihnya kebodohan.”*

## **HALAMAN PERSEMBAHAN**

Alhamdulillah Rabbil Aalamin, puji syukur kepada Allah SWT. Terimakasih atas karunia-Mu yang telah memberikan kemudahan dan kelancaran sehingga skripsi ini dapat terselesaikan dengan baik. Tak lupa juga sholawat selalu tercurahkan untuk baginda nabi besar Rasulullah SAW, sahabat-sahabatnya, keluarganya yang membawa manusia dari jahiliyah menuju peradaban yang lebih baik.

Penulis mempersembahkan karya ini kepada kedua orang tua, kerabat, teman, kakak tingkat, dan sahabat yang telah menemani perkuliahan daring maupun luring. Tanpa bantuan mereka penulis tidak akan bisa sampai di titik ini.

## KATA PENGANTAR

*Assalamualaikum Warahmatullahi Wabarakatuh*

Puji dan syukur penulis panjatkan atas ke hadirat Allah SWT, Tuhan Semesta Alam, yang telah melimpahkan Rahmat, hidayah, dan karunia-Nya yang telah memudahkan dalam penyelesaian skripsi yang berjudul **“Prediksi Urban Heat Island Menggunakan Algoritma Gradient Tree Boosting (Studi Kasus: Kota Malang)”** dengan baik. Shalawat dan salam senantiasa terlimpah kepada Nabi Muhammad Sallallahu ‘Alaihi wa Sallam. Dan semoga kita semua mendapat syafaatnya di hari kiamat nanti, Aamiin.

Penulis mengucapkan rasa syukur dan terima kasih kepada banyak pihak yang selalu memberikan semangat, bantuan dan motivasi kepada penulis. Untuk itu, penulis ingin menyampaikan rasa terima kasih kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Hj. Sri Hariani, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Fachrul Kurniawan, M.MT., IPM, selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Hani Nurhayati, M.T., selaku Dosen Pembimbing I yang telah memberikan banyak bimbingan dan bantuan selama penulisan skripsi ini.
5. Dr. Irwan Budi Santoso, M.Kom, selaku Dosen Pembimbing II yang telah memberikan bimbingan dalam penulisan skripsi ini.

6. Dr. Yunifa Miftachul Arif, M.T, selaku Ketua Penguji yang telah memberikan masukan dan dukungan yang sangat bermakna dalam menyelesaikan skripsi ini.
7. Dr. Fresy Nugroho, M.T., selaku dosen penguji I yang telah menguji serta memberikan masukan membangun dalam menyelesaikan skripsi ini.
8. Seluruh Dosen dan Jajaran Staf Program Studi Teknik Informatika yang telah memberikan banyak bantuan dalam skripsi ini.
9. Orang tua, serta keluarga yang telah memberikan dukungan, bantuan, serta doa yang tiada henti kepada penulis dalam menyelesaikan skripsi ini.
10. Warga kos 61A, Ingfo Life, dan teman-teman Angkatan 2020 Teknik Informatika yang telah memberikan banyak bantuan baik material maupun ilmu, semangat, serta motivasi dalam menyelesaikan penelitian skripsi ini.
11. Semua pihak yang tidak dapat disebutkan satu per satu, yang telah memberikan kontribusi, saran, dan dukungan dalam penyusunan penulisan skripsi ini.

Malang,

Muhammad Daniyal

## DAFTAR ISI

<b>HALAMAN PERSETUJUAN</b> .....	<b>iii</b>
<b>HALAMAN PENGESAHAN</b> .....	<b>iv</b>
<b>PERNYATAAN KEASLIAN TULISAN</b> .....	<b>v</b>
<b>MOTTO</b> .....	<b>vi</b>
<b>HALAMAN PERSEMBAHAN</b> .....	<b>vii</b>
<b>KATA PENGANTAR</b> .....	<b>viii</b>
<b>DAFTAR ISI</b> .....	<b>x</b>
<b>DAFTAR TABEL</b> .....	<b>xii</b>
<b>DAFTAR GAMBAR</b> .....	<b>xiii</b>
<b>ABSTRAK</b> .....	<b>xiv</b>
<b>ABSTRACT</b> .....	<b>xv</b>
<b>البحث مستخلص</b> .....	<b>xvi</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	5
1.3 Batasan Masalah .....	5
1.4 Tujuan Penelitian .....	6
1.5 Manfaat Penelitian .....	6
<b>BAB II STUDI PUSTAKA</b> .....	<b>7</b>
2.1. Penelitian Terdahulu .....	7
2.2. <i>Urban Heat Island</i> (UHI).....	9
2.2.1. Vegetasi.....	10
2.2.2. Permukaan Kedap Air .....	11
2.2.3. Badan Air .....	12
2.2.4. Panas .....	13
2.3. Pembelajaran Mesin ( <i>Machine Learning</i> ).....	13
2.4. Pembelajaran Terawasi ( <i>Supervised Learning</i> ) .....	14
2.5. <i>Gradient Tree Boosting</i> .....	14
2.5.1. <i>Loss Function</i> .....	17
2.5.2. <i>R-Squared</i> .....	21
2.5.3. <i>Mean Absolute Percentage Error</i> (MAPE).....	22
<b>BAB III DESAIN DAN IMPLEMENTASI</b> .....	<b>23</b>
3.1. Desain Penelitian.....	23
3.2. <i>Data Collecting</i> .....	23
3.2.1. Landsat-8.....	24
3.2.2. Sentinel-2 .....	25
3.2.3. <i>Shuttle Radar Topography Mission</i> (SRTM).....	26
3.3. <i>Data Preprocessing</i> .....	27
3.4. <i>Data Engineering</i> .....	28
3.4.1. <i>Feature Engineering</i> .....	28
3.4.2. <i>Feature Merging</i> .....	38
3.4.3. <i>Features Scaling</i> .....	38
3.5. <i>Model Building</i> .....	41

3.5.1. <i>Data Splitting</i> .....	41
3.5.2. <i>Model Training</i> .....	42
3.5.3. Prediksi.....	50
3.6. Analisis.....	51
3.6.1. Evaluasi Performa Model.....	51
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>52</b>
4.1. Uji Coba .....	52
4.2. Analisis Perbandingan Pemetaan LST Pada Data Aktual dan Prediksi.....	59
4.3. Integrasi Islam.....	65
4.3.1. Muamalah Mu'Allah.....	65
4.3.2. Muamalah Mu'annas.....	67
4.3.3. Muamalah Mu'alam.....	68
<b>BAB V KESIMPULAN DAN SARAN .....</b>	<b>70</b>
5.1. Kesimpulan .....	70
5.2. Saran.....	70
<b>DAFTAR PUSTAKA</b>	
<b>LAMPIRAN</b>	

## DAFTAR TABEL

Tabel 2.1 Penelitian Terkait .....	8
Tabel 2.2 Contoh perhitungan MSE menggunakan sampel data .....	21
Tabel 3.1 Detil Dataset.....	23
Tabel 3.2 Fitur-fitur untuk prediksi UHI.....	28
Tabel 3.3 Perhitungan Fitur.....	29
Tabel 3.4 Data setelah digabung .....	38
Tabel 3.5 Data sebelum dan sesudah normlisasi.....	41
Tabel 3.6 Detil skenario pengujian .....	49
Tabel 4.1 Hasil skenario pengujian .....	57
Tabel 4.2 Perbandingan LST aktual dan hasil rediksi model GTB .....	60

## DAFTAR GAMBAR

Gambar 2.1 Ilustrasi Fenomena UHI .....	9
Gambar 2.2. Vegetasi .....	11
Gambar 2.3 Permukaan kedap air .....	11
Gambar 2.4 Badan air .....	12
Gambar 2.5 Arsitektur <i>Gradient Tree Boosting</i> .....	15
Gambar 2.6 Representasi Visual Mean Squared Error (MSE) .....	20
Gambar 3.1 Alur Desain Penelitian .....	23
Gambar 3.2 Visualisasi NDVI .....	29
Gambar 3.3 Visualisasi NDBI.....	30
Gambar 3.4 Visualisasi NDWI .....	32
Gambar 3.5 Visualisasi NDMI.....	33
Gambar 3.6 Visualisasi Urban Density Index.....	35
Gambar 3.7 Pemetaan LST .....	37
Gambar 3.8 Rentang nilai tiap fitur sebelum dilakukan normalisasi .....	39
Gambar 3.9 Rentang nilai tiap fitur setelah dilakukan normalisasi .....	40
Gambar 3.10 Visualisasi <i>data splitting</i> .....	42
Gambar 3.11 Alur proses algoritma GTB .....	43
Gambar 3.12 Implementasi inisialisasi prediksi .....	44
Gambar 3.13 Implementasi perhitungan residual .....	44
Gambar 3.14 Implementasi pembuatan <i>tree</i> .....	45
Gambar 3.15 Visualisasi <i>Tree-1</i> .....	46
Gambar 3.16 Implementasi pembaharuan (penggabungan) prediksi.....	47
Gambar 3.17 Visualisasi kombinasi prediksi.....	48
Gambar 3.18 Implementasi algoritma <i>gradient tree boosting</i> .....	48

## ABSTRAK

Daniyal, Muhammad. 2024. **Prediksi *Urban Heat Island* Menggunakan Algoritma *Gradient Tree Boosting* (Studi Kasus: Kota Malang)**. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Hani Nurhayati, M.T (II) Dr. Irwan Budi Santoso, M.Kom.

**Kata kunci:** prediksi, *urban heat island*, *gradient tree boosting*.

Fenomena *Urban Heat Island* (UHI) merupakan masalah lingkungan yang signifikan di kota-kota besar, termasuk Kota Malang. Penelitian ini bertujuan untuk melatih model prediksi UHI di Kota Malang menggunakan *Algoritma Gradient Tree Boosting* (GTB). Metode penelitian melibatkan pemrosesan data citra satelit untuk memperoleh data suhu permukaan dan faktor penentu lainnya, yang kemudian digunakan untuk melatih model prediksi. Pelatihan model dilakukan dengan berbagai skenario kombinasi dari rasio data latih uji, *learning rate* dan *loss function* untuk mendapatkan model terbaik. Hasil penelitian menunjukkan bahwa model terbaik diperoleh pada skenario ketujuh dengan kombinasi rasio data latih 70%, *learning rate* 0,1, dan *loss function squared error*. Model ini menghasilkan nilai *Mean Squared Error* (MSE) yang rendah sebesar 1,76, *Mean Absolute Percentage Error* (MAPE) sebesar 2,79, dan *R-squared* ( $R^2$ ) sebesar 0,88. Nilai MSE dan MAPE yang rendah menunjukkan bahwa model memiliki performa prediksi yang baik dengan kesalahan yang minimal. Sementara itu, nilai  $R^2$  yang tinggi menunjukkan kemampuan model dalam memahami variasi data dan hubungan antar fitur dengan target secara baik. Analisis hasil pemetaan prediksi mengindikasikan peningkatan suhu hampir di seluruh wilayah Kota Malang, meliputi kecamatan Lowokwaru, Klojen, Blimbing, Kedungkandang, dan Sukun. Fenomena UHI diprediksi akan terjadi di beberapa bagian dari wilayah kecamatan tersebut.

## ABSTRACT

Daniyal, Muhammad. 2024. **Prediksi *Urban Heat Island* Menggunakan Algoritma *Gradient Tree Boosting* (Studi Kasus: Kota Malang)**. Thesis. Department of Informatics Engineering, Faculty of Science and Technology, Maulana Malik Ibrahim Islamic State University Malang. Promotor: (I) Hani Nurhayati, M.T (II) Dr. Irwan Budi Santoso, M.Kom.

**Keywords:** prediction, *urban heat island*, *gradient tree boosting*.

The *Urban Heat Island* (UHI) phenomenon is a significant environmental issue in large cities, including Malang City. This study aims to train a model to predict UHI in Malang City using the *Gradient Tree Boosting* (GTB) Algorithm. The research method involves processing satellite imagery data to obtain surface temperature and other determinant factors, which are then used to train the prediction model. *Model training* was conducted with various scenarios combining different ratios of training and testing data, *learning rates*, and *loss functions* to obtain the best model. The results show that the best model was obtained in the seventh scenario with a combination of a 70% training data ratio, a *learning rate* of 0.1, and a squared error *loss function*. This model produced a low Mean Squared Error (MSE) of 1.76, a *Mean Absolute Percentage Error* (MAPE) of 2.79, and an *R-squared* ( $R^2$ ) of 0.88. The low MSE and MAPE values indicate that the model has good predictive performance with minimal errors. Meanwhile, the high  $R^2$  value demonstrates the model's ability to understand data variations and the relationship between features and the target effectively. Analysis of the prediction mapping results indicates a temperature increase in almost all areas of Malang City, including the districts of Lowokwaru, Klojen, Blimbing, Kedungkandang, and Sukun. The UHI phenomenon is predicted to occur in some parts of these districts.

## البحث مستخلص

دانيال، محمد. 2024. تنبؤ جزيرة الحرارة الحضرية باستخدام خوارزمية تعزيز الشجرة المتدرجة (دراسة حالة: مدينة مالانغ). (الأطروحة). قسم هاني نور حياي (I): هندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانغ. المشرفون: د. إروان بودي سانتوسو، م. كوم (II) M.T.

الكلمات المفتاحية: التنبؤ، جزيرة الحرارة الحضرية، تعزيز الشجرة المتدرجة

قضية بيئية مهمة في المدن الكبيرة، بما في ذلك مدينة مالانغ. تهدف هذه الدراسة إلى تدريب (UHI) تعد ظاهرة جزيرة الحرارة الحضرية تتضمن طريقة البحث معالجة بيانات الصور. (GTB) في مدينة مالانغ باستخدام خوارزمية تعزيز الشجرة المتدرجة UHI نموذج للتنبؤ به الفضائية للحصول على درجة حرارة السطح والعوامل المحددة الأخرى، التي تُستخدم بعد ذلك لتدريب نموذج التنبؤ. تم إجراء تدريب النموذج باستخدام سيناريوهات مختلفة تجمع بين نسب مختلفة من بيانات التدريب والاختبار، ومعدلات التعلم، ووظائف فقدان للحصول على أفضل نموذج. أظهرت النتائج أن أفضل نموذج تم الحصول عليه في السيناريو السابع مع مزيج من نسبة بيانات التدريب 70٪، ومعدل تعلم 0.1 (MAPE) منخفض قدره 1.76، وخطأ نسبة مطلق متوسط (MSE) ووظيفة فقدان الخطأ التريبي. أنتج هذا النموذج خطأ تريبي متوسط إلى أن النموذج يتمتع بأداء تنبؤي جيد مع MAPE و MSE قدره 0.88. تشير القيم المنخفضة لـ  $R^2$  قدره 2.79، ومعامل تحديد تظهر قدرة النموذج على فهم تباينات البيانات والعلاقة بين الميزات والهدف بشكل فعال. تشير  $R^2$  أخطاء قليلة. في حين أن القيمة العالية لـ تحليلات نتائج خرائط التنبؤ إلى زيادة في درجة الحرارة في جميع مناطق مدينة مالانغ تقريباً، بما في ذلك مناطق لوكوارو، كلوين، بليمينغ. في بعض أجزاء هذه المناطق UHI كيدونغكاندانغ، وسوكين. من المتوقع حدوث ظاهرة

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Fenomena *Urban Heat Island* (UHI) merupakan kondisi di mana suhu udara permukaan di area perkotaan jauh lebih tinggi dibandingkan area sekitarnya. Perbedaan suhu ini dapat mencapai 1-3 °C bahkan lebih. Peningkatan suhu akibat UHI disebabkan oleh berbagai faktor seperti penggunaan material yang menyerap dan memancarkan panas dengan baik pada bangunan dan infrastruktur perkotaan, hilangnya area hijau yang berfungsi sebagai pendingin alami, serta panas buangan dari aktivitas penduduk perkotaan. Penelitian "*Three Decades of Urban Heat Islands and Mitigation Technologies Research*" yang dilakukan pada tahun 2016 menyebutkan bahwa peningkatan suhu ini berdampak buruk bagi lingkungan dan kehidupan. Peningkatan suhu ini berefek pada penggunaan energi, polusi udara, suhu luar ruangan, dan kesehatan masyarakat.

Salah satu perkotaan yang juga terdampak oleh fenomena ini ialah Kota Malang. Data dari Badan Pusat Statistik (BPS) Kota Malang menyebutkan terjadi kenaikan rata-rata suhu 0.6 °C dengan rentang waktu 1 tahun. Dengan dampak buruk yang akan terjadi akibat fenomena kenaikan suhu ini, permasalahan untuk memprediksi fenomena UHI menjadi penting untuk dapat menganalisis mitigasi untuk menanggulangnya.

Seiring perkembangan teknologi, pembelajaran mesin (*machine learning* / ML) menjadi salah satu solusi terbaik untuk berbagai permasalahan tak terkecuali prediksi UHI. Kelebihan dari penggunaan *machine learning* ini ialah waktu dan

ketepatannya. Salah satu algoritma machine learning yang populer untuk penyelesaian masalah prediksi ialah *Gradient Tree Boosting* (GTB).

Berdasarkan analisis penulis mengenai penelitian terkait, terdapat beberapa penelitian terdahulu yang menggunakan metode serupa namun pada permasalahan yang berbeda seperti yang dilakukan oleh Furuya dkk yang membandingkan berbagai macam metode *machine learning* sebagai pendekatan dalam pemetaan *Surface Urban Heat Island* (SUHI) di Brazil, dimana dari hasil penelitian mereka didapat algoritma *Decision Tree* (DT) memiliki performa yang terbaik dalam kasus ini (Furuya *et al.*, 2023). Dari situ penulis memiliki hipotesis bahwa metode *Gradient tree boosting* (GTB) yang merupakan algoritma lanjutan dari DT juga akan memiliki performa prediksi yang sama atau bahkan lebih baik dari DT. Hal ini mendorong penulis untuk mengeksplorasi dan menganalisis hasil prediksi dari algoritma ini terhadap permasalahan prediksi UHI sehingga diharapkan dapat memberikan kontribusi terhadap penelitian dimasa mendatang dan manfaat untuk lingkungan dan sesama manusia.

Seperti yang tertulis dalam Al-Qur'an sebagai pedoman hidup umat muslim, khususnya pada Al-A'raf ayat 56 dan Ali-Imran 190-191 yang berbunyi

وَلَا تُفْسِدُوا فِي الْأَرْضِ بَعْدَ إِصْلَاحِهَا وَادْعُوهُ حَوْفًا وَقَطْمَعًا إِنَّ رَحْمَتَ اللَّهِ قَرِيبٌ مِّنَ الْمُحْسِنِينَ ٥٦ ( الاعراف/7:  
(56)

“Janganlah kamu berbuat kerusakan di bumi setelah diatur dengan baik. Berdoalah kepada-Nya dengan rasa takut dan penuh harap. Sesungguhnya rahmat Allah sangat dekat dengan orang-orang yang berbuat baik.” (QS. Al-A'raf:56)

Dalam tafsir Al-Misbah, ayat ini ditafsirkan sebagai “Jangan kalian membuat kerusakan di muka bumi yang telah dibuat baik dengan menebar kemaksiatan,

kezaliman dan permusuhan. Berdoalah kepada-Nya dengan rasa takut akan siksa-Nya dan berharap pahala-Nya. Kasih sayang Allah sangat dekat kepada setiap orang yang berbuat baik, dan pasti terlaksana”.

Sedang menurut tafsir Ibnu Katsir, ayat ini ditafsirkan sebagai, berdoalah kepada Tuhan kalian dengan berendah diri dan suara yang lembut. Sesungguhnya Allah tidak menyukai orang-orang yang melampaui batas. Dan janganlah kalian membuat kerusakan di muka bumi, sesudah (Allah) memperbaikinya dan berdoalah kepada-Nya dengan rasa takut (tidak akan diterima) dan harapan (akan dikabulkan). Sesungguhnya rahmat Allah amat dekat kepada orang-orang yang berbuat baik. Allah subhanahu wa ta'ala memberikan petunjuk kepada hamba-hamba-Nya agar mereka berdoa memohon kepada-Nya untuk kebaikan urusan dunia dan akhirat mereka. Untuk itu Allah subhanahu wa ta'ala berfirman: Berdoalah kepada Tuhan kalian dengan berendah diri dan suara yang lembut. (Al-A'raf: 54) Menurut suatu pendapat, makna yang dimaksud ialah mengucapkan doa dengan perasaan yang rendah diri, penuh harap, dan dengan suara yang lemah lembut.

Dari kedua tafsiran di atas, dijelaskan bahwa menjaga dan merawat lingkungan adalah salah satu dari perintah Allah yang sangat penting. Allah telah menciptakan alam dengan segala kelengkapannya untuk kebutuhan manusia, dan manusia dilarang untuk merusaknya. Melalui kebaikan dan perbuatan yang baik, termasuk menjaga lingkungan, kita dapat mendekatkan diri kepada kasih sayang Allah dan memperoleh rahmat-Nya.

Dalam ayat lain Allah SWT juga berfirman dalam Surat Al-Imran ayat 190-191 yang berbunyi

إِنَّ فِي خَلْقِ السَّمَوَاتِ وَالْأَرْضِ وَاخْتِلَافِ اللَّيْلِ وَالنَّهَارِ لَآيَاتٍ لِأُولِي الْأَلْبَابِ ۚ ۱۹۰ الَّذِينَ يَذْكُرُونَ اللَّهَ قِيَامًا وَقُعُودًا  
وَعَلَى جُنُوبِهِمْ وَيَتَفَكَّرُونَ فِي خَلْقِ السَّمَوَاتِ وَالْأَرْضِ رَبَّنَا مَا خَلَقْتَ هَذَا بَاطِلًا سُبْحَانَكَ فَقِنَا عَذَابَ النَّارِ ۚ ۱۹۱ )  
ال عمران/3: 190-191)

*“Sesungguhnya dalam penciptaan langit dan bumi, dan silih bergantinya malam dan siang terdapat tanda-tanda (kebesaran Allah) bagi orang yang berakal. (yaitu) orang-orang yang mengingat Allah sambil berdiri, duduk, atau dalam keadaan berbaring, dan memikirkan tentang penciptaan langit dan bumi (seraya berkata), "Ya Tuhan kami, tidaklah Engkau menciptakan semua ini sia-sia. Maha Suci Engkau. Lindungilah kami dari azab neraka." (QS. Ali Imran: 190-191)*

Menurut Buya Hamka dalam Tafsir Al Azhar menjelaskan bahwa surat Ali Imran ayat 190-191 berisi tentang arahan Allah kepada hamba-Nya untuk merenungkan alam, langit, dan bumi. Allah SWT mengarahkan agar hamba-Nya menggunakan pikirannya dan memperhatikan pergantian antara siang dan malam. Semua itu penuh dengan tanda kebesaran Allah. Siapa yang mampu memahami bahwa penciptaan langit dan bumi serta pergantian siang dan malam merupakan tanda-tanda kekuasaan Allah, disebut ulul albab. Menurut Ibnu Katsir, ulul albab adalah mereka adalah orang yang memiliki akal sempurna dan memiliki kecerdasan. Sedangkan menurut Sayyid Qutb, mereka adalah orang-orang yang memiliki pemikiran dan pemahaman yang benar.

Disamping itu, merawat dan menjaga lingkungan juga ajarkan oleh Rasulullah SAW yang diriwayatkan oleh Imam Bukhari dalam kitabnya Al-Hibbah yang berbunyi

*Rasulullah S.a.w. bersabda: Barangsiapa ada memiliki tanah, maka hendaklah ia tanami atau serahkan kepada saudaranya (untuk dimanfaatkan), maka jika ia enggan, hendaklah ia memperhatikan sendiri memelihara tanah itu. “ (HR. Imam Bukhori).*

Hadist ini memberitahu umat muslim untuk memanfaatkan dan memelihara lingkungannya. Dalam konteks penelitian ini, UHI dinilai sebagai fenomena yang berdampak buruk bagi manusia dan lingkungan dan merupakan bentuk kerusakan di bumi. Sehingga berdasar pada analisis tafsir ayat dan hadist diatas, sebagai umat muslim kita di perintahkan untuk mengurangi dampak buruknya dengan melakukan analisis dan prediksi lebih lanjut pada fenomena ini sehingga dengan itu dapat melakukan tindakan mitigasi yang tepat supaya memberikan manfaat untuk mahluk hidup dan menjaga kelaestarian lingkungan.

Sebagai realisasi pemahaman mengenai tafsir-tafsir dan beberapa penjelasan mengenai UHI, penulis berinisiatif untuk melakukan penelitian yang mana menganalisis performa algoritma *Gradient Tree Boosting* (GTB) dalam memprediksi fenomena UHI. Pada prosesnya, penelitian ini juga menganalisis fenomena UHI dan memprediksinya untuk meminimalisir dampak negatif UHI terhadap alam dan lingkungan dengan tindakan mitigasi yang tepat.

## **1.2 Rumusan Masalah**

Dibutuhkan model untuk memprediksi UHI di Kota Malang dikarenakan terjadi fenomena UHI

## **1.3 Batasan Masalah**

Untuk menghindari pembahasan yang keluar dari konteks penelitian, maka dirumuskanlah batasan masalah yang ditentukan. Adapun batasan masalah sebagai berikut:

1. Penelitian ini terbatas hanya di wilayah-wilayah di Kota Malang

2. Analisis dalam penelitian ini hanya mencakup pengukuran akurasi prediksi dan evaluasi performa model prediksi menggunakan metrik *Mean Squared Error* (MSE) *Mean Absolute Percentage Error* (MAPE) dan *R-Squared* (R<sup>2</sup>).
3. Data yang digunakan dalam penelitian ini hanya menggunakan data Landsat-8 Sentinel-2 Dan SRTM pads rentang tahun 2019-2023.

#### **1.4 Tujuan Penelitian**

Tujuan penelitian dari rumusan masalah di atas adalah melatih model untuk memprediksi UHI di Kota Malang.

#### **1.5 Manfaat Penelitian**

Untuk akademisi, penelitian ini memberikan kontribusi positif dalam pengembangan ilmu pengetahuan terkait penerapan algoritma GTB untuk kasus prediksi UHI.

Untuk pemerintah Kota Malang, penelitian ini memberikan pemahaman yang lebih baik kepada pemerintah Kota Malang tentang bagaimana fenomena ini memengaruhi Kota Malang. Informasi ini dapat digunakan dalam perencanaan kota yang lebih berkelanjutan, termasuk dalam pembangunan infrastruktur, penggunaan lahan, dan penanaman vegeta

## BAB II

### STUDI PUSTAKA

#### 2.1. Penelitian Terdahulu

Furuya dkk melakukan penelitian pemetaan *Surface Urban Heat Island* (SUHI) di sebuah kota medium di Brasil dengan pendekatan machine learning. Ia memperoleh data *Land Surface Temperature* (LST) dari 15 citra Landsat 8 tahun 2019-2021. Variabel lingkungan diambil dari citra Sentinel-2 dan variabel sosial-ekonomi dari data sensus. Wilayah studi dibagi menjadi 309 sektor sensus. Enam algoritma ML diuji, yaitu *decision tree* (DT), kNN, *linear regression*, *multilayer perceptron*, *random forest* (RF), dan *support vector regression* (SVR). Hasilnya menunjukkan algoritma DT pada model 1 memiliki performa terbaik dengan nilai korelasi ( $r$ ) 0.96, *mean absolute error* (MAE) 1.49 dan *root mean square error* (RMSE) 1.88°C (Furuya *et al.*, 2023) .

Penelitian lain bertujuan untuk menganalisis fenomena *urban heat island* (UHI) di kota Jeddah, Arab Saudi, dari tahun 2000 hingga 2021. Peneliti menggunakan teknik analisis data penginderaan jauh untuk memetakan perubahan penggunaan lahan di Jeddah selama kurun waktu tersebut. Data *land surface temperature* (LST) juga diekstraksi dari citra satelit Landsat untuk kemudian dihitung nilai UHI berdasarkan nilai LST dan diklasifikasikan menjadi UHI rendah, sedang, dan tinggi. Hasil penelitian menunjukkan bahwa telah terjadi perubahan penggunaan lahan yang signifikan di Jeddah dari tahun 2000 hingga 2021. Misalnya, wilayah terbangun meningkat sekitar 80%, vegetasi meningkat 32%, lahan terbuka

berkurang 30%, dan badan air berkurang 12%. Selain itu, rata-rata LST Jeddah meningkat sebesar 8,32% dan temperatur maksimum meningkat 10,22% selama rentang waktu tersebut. Luas wilayah dengan nilai UHI tinggi juga terlihat meningkat dari 40,31% di tahun 2000 menjadi 45,27% di tahun 2021. Analisis statistik yang dilakukan menunjukkan terdapat korelasi kuat antara nilai UHI dengan LST dan *Normalized Difference Built-up Index* (NDBI). Model regresi linier juga mengindikasikan pengaruh yang signifikan dari variabel-variabel sosio-ekologi terhadap nilai UHI di Jeddah. Didapat sebagian besar wilayah Jeddah dicirikan oleh nilai UHI yang tinggi dan sangat tinggi (Addas, 2023).

Kaur *et al.* (2021) melakukan penelitian menarik tentang prediksi *Urban Heat Island* (UHI) menggunakan algoritma *artificial neural network* (ANN) dan analisis time series. Mereka mengumpulkan data cuaca yang komprehensif dari 6 stasiun di Rajasthan, India, dengan total 12.929 contoh data. Empat parameter input utama oleh peneliti, yaitu suhu maksimum, suhu minimum, kecepatan angin, dan kelembapan relatif, serta penggunaan radiasi matahari sebagai parameter output.

Tabel 2.1 Penelitian Terkait

Judul	Tahun	Persamaan	Perbedaan
<i>A machine learning approach for mapping surface urban heat island using environmental and socioeconomic variables: a case study in a medium-sized Brazilian city</i>	2020	Prediksi UHI menggunakan variabel NDVI, NDWI, NDMI, NDBI	Metode yang dipakai <i>Decision tree</i> , <i>kNN</i> , <i>linear regression</i> , <i>multilayer perceptron</i> , <i>random forest</i> , dan <i>support vector regression</i> dan lokasi penelitian di Brazil
<i>Machine learning Techniques to Map the Impact of Urban Heat Island: Investigating the City of Jeddah</i>	2021	Prediksi UHI Menggunakan variabel NDVI, NDWI, NDMI	Metode yang dipakai <i>Banging</i> , <i>Random sumspace</i> lokasi penelitian di Jeddah

Judul	Tahun	Persamaan	Perbedaan
<i>Urban Heat Island Prediction Using ANN</i>	2021	Prediksi UHI	Metode yang dipakai ialah algoritma ANN

## 2.2. *Urban Heat Island (UHI)*



Gambar 2.1 Ilustrasi Fenomena UHI

*Urban Heat Island (UHI)* mengacu pada fenomena di mana daerah perkotaan yang dipadati dengan kepadatan bangunan dan aktifitas urban mengalami suhu yang lebih tinggi dibandingkan dengan daerah sekitarnya yang masih minim bangunan dan terdapat banyak ruang hijau seperti yang ditunjukkan pada Gambar 2.1.

Terjadinya sebuah fenomena tentunya ada sebab yang melatarbelakanginya. Dikutip dari penelitian yang dilakukan Jabbar dkk di Kota Jeddah, fenomena UHI disebabkan oleh faktor-faktor seperti urbanisasi, industrialisasi, dan aktivitas manusia yang menghasilkan panas dan mengurangi tutupan hijau. Peningkatan suhu di ruang perkotaan dapat menyebabkan ketidaknyamanan dan berdampak negatif pada lingkungan dan kesehatan manusia. Memahami dan mengurangi UHI sangat penting untuk perencanaan dan desain perkotaan untuk menciptakan lingkungan perkotaan yang lebih berkelanjutan dan nyaman (Jabbar *et al.*, 2023).

Fenomena UHI dapat dianalisis dengan mengacu pada suhu permukaan bumi / *Land Surface Temperature* (LST). Dalam analisisnya, perhitungan nilai UHI didasarkan pada LST pada daerah manapun yang bisa di estimasi. Berdasarkan penelitian yang dilakukan Pratiwi A dkk, persamaan UHI dinyatakan sebagai berikut (Pratiwi & Jaelani, 2021)

$$UHI = LST - (\mu + 0,5 \sigma) \quad (2.1)$$

Dimana:

UHI = *Urban Heat Island*

LST = Suhu Permukaan (LST) (°C)

$\mu$  = Nilai rata-rata LST (°C)

$\sigma$  = Standar deviasi LST (°C)

Nilai LST yang tinggi menjadi indikator intensitas UHI yang tinggi maupun sebaliknya. Beberapa faktor yang sangat sensitif pada fenomena UHI ialah faktor *Social Ecology Variable* (SEV). Secara garis besar SEV dikategorikan menjadu 4 kelompok, yakni

1. Vegetasi
2. Permukaan kedap air
3. Badan Air
4. Panas

### **2.2.1. Vegetasi**

Vegetasi adalah tumbuhan yang menutupi permukaan bumi. Vegetasi terdiri dari berbagai jenis tumbuhan seperti pepohonan, semak, rumput, lumut, paku, dan tumbuhan berbunga. Vegetasi divisualisasikan sebagai berikut



Gambar 2.2. Vegetasi

Dalam Gambar 2.2, vegetasi digambarkan sebagai area yang tertutup pohon atau tanaman. Identifikasi dan pengukuran area ini dapat dilakukan menggunakan indeks *Normalized Difference Vegetation Index* (NDVI). Penelitian Grover dan Sigh yang menganalisis tentang keterkaitan antara NDVI terhadap UHI di Kota Mumbai dan Kota Delhi, menemukan bahwa NDVI yang tinggi berdampak pada turunnya suhu diantara kedua kota. Hasil ini menegaskan bahwa vegetasi memiliki pengaruh terhadap UHI. (Grover & Singh, 2015)

### 2.2.2. Permukaan Kedap Air



Gambar 2.3 Permukaan kedap air

. Permukaan Kedap Air (*Impervious Surface / IS*) mengarah pada permukaan lahan yang tidak dapat ditembus air seperti aspal, beton, logam, atau kaca seperti yang ditunjukkan oleh Gambar 2.3. Kaitan IS dengan fenomena UHI pernah diteliti di Kota Harbin, Dimana dalam penelitian *itu* menyebutkan peningkatan luas area IS menyebabkan peningkatan LST yang berdampak pada peningkatan intensitas UHI (Wu *et al.*, 2019). Oleh karena itu, peningkatan luas permukaan kedap air berkontribusi pada peningkatan intensitas UHI (Zha *et al.*, 2003a).

### 2.2.3. Badan Air



Gambar 2.4 Badan air

Area badan air ialah area yang digenangi oleh air seperti danau, laut sungai dan sejenisnya seperti yang ditunjukkan oleh Gambar 2.4. Penelitian yang dilakukan di Kota Beijing untuk menginvestigasi *Urban Cooling Island (UCI)* menemukan bahwa area badan air memiliki korelasi positif dengan UHI, dimana semakin luas area badan air, semakin rendah pula LST-nya (Sun & Chen, 2012). Pernyataan ini juga dipertegas oleh penelitian yang dilakukan di Ohio, yang menemukan bahwa badan air memiliki *pengaruh* yang sangat kuat terhadap LST yang merupakan faktor dominan terjadinya fenomena UHI (Peng *et al.*, 2020).

#### **2.2.4. Panas**

Panas mengacu pada temperatur suhu di permukaan. Pada konteks UHI, panas dinyatakan sebagai LST. Berdasar *pada* penelitian terdahulu, panas adalah faktor dominan yang mendasari fenomena UHI terjadi. Seperti yang telah disebutkan, peningkatan LST menggambarkan seberapa besar intensitas UHI. Peningkatan suhu yang disebabkan oleh beberapa faktor yang telah disebutkan sebelumnya menjadi tolak ukur seberapa tinggi intensitas UHI.

#### **2.3. Pembelajaran Mesin (*Machine Learning*)**

Pembelajaran Mesin (*Machine learning* / ML) ialah sub-bagian dari kecerdasan buatan (*Artificial Intelligence* / AI) dimana melibatkan algoritma dan pengembangan model yang *memungkinkan* sistem mempelajari pengetahuan dari pengalaman dan memahami pola dari data mentah sehingga dapat menentukan aturan mereka sendiri untuk mengoptimalkan kinerjanya tanpa perlu diprogram secara eksplisit (Hosny *et al.*, 2018) (Deekshith & Singh, 2020). Istilah “*machine learning*” dikemukakan pertama kali oleh Arthur Samuel pada tahun 1959 dalam artikel ilmiah berjudul "*Some Studies in Machine learning Using the Game of Checkers*" yang didalamnya membahas mengenai pengaplikasian *machine learning* pada *Game of checkers* (Samuel, 1959). Seiring berjalannya waktu, perkembangan algoritma *machine learning* dalam 100 tahun terakhir terpecah ke dalam beberapa bagian yakni:

1. *Supervised Learning*
2. *Unsupervised Learning*
3. *Reinforcement Learning*

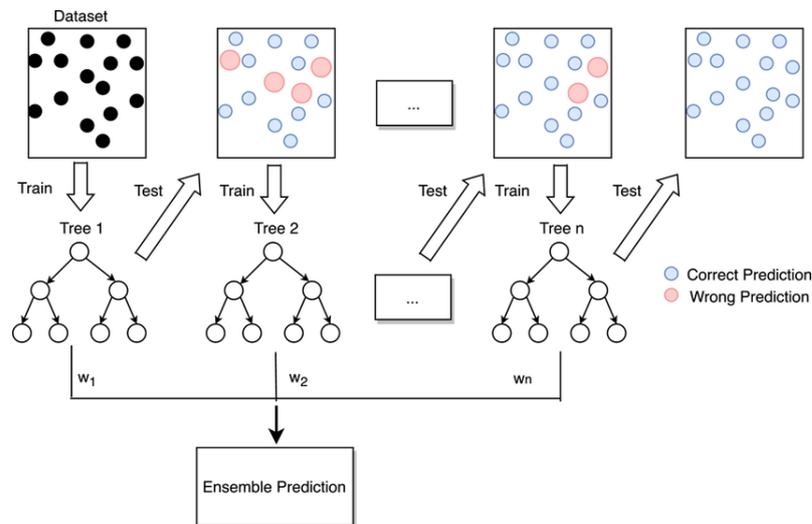
#### 4. *Deep Learning*

##### **2.4. Pembelajaran Terawasi (*Supervised Learning*)**

*Supervised Learning* ialah salah satu bagian dalam *machine learning* di mana model belajar dari data yang telah diberi label (Liu, 2011). Model ini memberikan contoh-contoh data yang sudah *diketahui* labelnya, dan tujuannya adalah untuk mempelajari tingkat atau hubungan antara fitur-fitur *input* data dengan label yang sesuai (Wang *et al.*, 2022). Masalah yang diselesaikan oleh metode pembelajaran ini terkait klasifikasi dan prediksi (Prof *et al.*, 2023) (Mumtaz *et al.*, 2023). Beberapa algoritma yang termasuk kedalam *supervised learning* diantaranya *Decision Tree* (DT), *Support Vector*, *Random Forest* (RF) hingga ***Gradient Tree Boosting*** (GTB). Beberapa algoritma diatas memiliki jenisnya untuk masalah klasifikasi maupun regresi.

##### **2.5. *Gradient Tree Boosting***

*Gradient Tree Boosting* (GTB) adalah salah satu model *ensemble*, yakni model yang mengandalkan kombinasi dari model lemah berbasis *decision tree* yang akan saling menguatkan sehingga *akan* menghasilkan model yang kuat. Model lemah ialah model prediktif sederhana yang relatif lemah yang secara individu mungkin tidak berkinerja baik. Model lemah dapat didefinisikan sebagai model yang sedikit lebih baik daripada menebak secara acak. Dalam konteks gradient boosting, pohon keputusan biasanya digunakan sebagai pembelajar lemah. Sebagai contoh adalah dahan keputusan atau pohon keputusan satu tingkat (CART yang kedalaman maksimumnya adalah 1).



Gambar 2.5 Arsitektur *Gradient Tree Boosting*

GTB bekerja secara iteratif dengan membentuk pohon secara berurutan sehingga model berikutnya dapat belajar dari kesalahan sebelumnya. Secara rinci, GTB dimulai dengan membuat pohon pertama menggunakan data latih untuk melakukan prediksi. Hasil prediksi awal ini kemudian dibandingkan dengan nilai target yang sebenarnya dari setiap sampel data. Perbedaan antara prediksi dan nilai target yang sebenarnya disebut sebagai residual, yang menunjukkan kesalahan prediksi (atau "error"). Setelah pohon pertama melakukan prediksi, *loss function* (misalnya, *squared error*, *absolute error*, *huber* dll) dihitung berdasarkan perbedaan antara prediksi model dan nilai target yang sebenarnya. *Loss function* memberikan gambaran seberapa besar kesalahan prediksi yang terjadi. Dalam konteks GTB, *loss function* yang paling umum digunakan ialah *squared error*. Selanjutnya, *gradien loss function* dihitung untuk menunjukkan arah dan seberapa besar perubahan yang perlu dilakukan pada prediksi model agar mengurangi kesalahan. Ini membimbing proses penyesuaian model pada iterasi berikutnya.

Proses ini diulang untuk setiap sampel data dalam dataset latih hingga semua residu diidentifikasi. Residu tersebut kemudian digunakan sebagai target baru untuk membuat pohon berikutnya, lalu dilanjutkan dengan menambahkan pohon-pohon tambahan untuk memperbaiki prediksi sebelumnya dengan menggunakan residu sebagai target baru pada setiap langkah. Setelah setiap pohon ditambahkan, *loss function* dihitung kembali untuk mengevaluasi peningkatan dalam prediksi dan menyesuaikan model berdasarkan *gradien loss function*. Ini berarti model berusaha untuk mengoptimalkan prediksi terhadap nilai target yang sebenarnya pada setiap iterasi.

Proses iteratif berlanjut sampai jumlah pohon yang diinginkan tercapai atau ketika peningkatan *prediksi* tidak signifikan lagi. Akhirnya, prediksi dari setiap pohon digabungkan untuk menghasilkan prediksi akhir yang lebih akurat dan kuat. Dengan iterasi ini, model secara bertahap belajar dari kesalahan prediksi sebelumnya dan meningkatkan kemampuannya untuk menghasilkan prediksi yang lebih akurat terhadap nilai aktual. Ini adalah hasil akhir dari akumulasi semua prediksi lemah dari setiap pohon seperti yang ditunjukkan oleh Gambar 2.5, (Asante *et al.*, 2021). Persamaan dari GTB dapat dituliskan sebagai berikut:

$$y_i = F_M(x_i) = \sum_m^M h_m(x_i) \quad (2.1)$$

Dimana,

$y_i$  : prediksi

$x_i$  : input

$h_m$  : prediksi / pembelajar lemah (weak learner).

$M$  : Konstanta jumlah pohon keputusan yang akan dibuat ( $n\_estimator$ ).

Berdasarkan beberapa literasi terdahulu dan penelitian mengenai prediksi covid-19 pada tahun 2021 yang menggunakan metode serupa, kasus regresi atau

prediksi umumnya dievaluasi menggunakan metrik *R-Squared* ( $R^2$ ) dan *Mean Squared Error* (MSE), namun pada penelitian kali ini penulis menambahkan *Mean Absolute Percentage Error* (MAPE) sebagai metrik evaluasi tambahan (Gumaei *et al.*, 2021).

Alasan penggunaan algoritma ini didasarkan pada penelitian terdahulu yang dilakukan oleh Furuya dkk yang mana algoritma *decision tree* memiliki hasil prediksi terbaik dengan  $R^2$  0.96 dan *Mean Absolute Error* (MAE) 1.49 pada kasus serupa yang diteliti di Brazil. Dari hasil penelitian itu, penulis memiliki hipotesa jika *Gradient tree boosting* yang merupakan metode lanjutan dan gabungan dari *decision tree* akan memiliki performa yang lebih baik dibandingkan *decision tree* pada kasus serupa.

### **2.5.1. Loss Function**

Dikutip dari penelitian Friedman pada tahun 2001 yang berjudul “*Greedy function approximation: A gradient boosting machine.*”, *Loss function* diartikan sebagai fungsi yang mengukur seberapa besar perbedaan antara nilai prediksi dan nilai aktual, dimana perbedaan itu mengarahkan model untuk mengecilkannya (Friedman, 2001).

Secara umum, *Loss function* adalah sebuah fungsi matematis yang digunakan untuk mengukur seberapa besar kesalahan prediksi model terhadap nilai target yang sebenarnya selama proses pelatihan model. Tujuan dari *loss function* ialah untuk memberikan indikasi seberapa baik model memprediksi nilai target yang sebenarnya, sehingga model dapat diperbaiki secara iteratif untuk mengurangi kesalahan prediksi tersebut.

*Loss function* memainkan peran penting dalam algoritma gradient boosting karena membimbing proses penambahan model (*weak learner*) tambahan. Setiap iterasi dalam algoritma GTB, model mencoba menambahkan model baru yang mengurangi kesalahan yang masih ada pada prediksi sebelumnya dengan mengoptimalkan *loss function* (meminimalisir nilainya).

Beberapa *loss function* yang umum digunakan dalam algoritma GTB adalah ***Squared error*** atau yang biasa disebut ***Squared Error, Absolute error*** dan ***Huber Loss***.

#### 2.5.1.1. Squared Error

*Squared error* / Kuadrat Kesalahan adalah nilai kuadrat dari selisih antara nilai prediksi dan aktual. Secara matematis, persamaan *squared error* dituliskan sebagai berikut

$$\text{Squared Error} = (y_i - \hat{y}_i)^2 \quad (2.2)$$

Dimana

$y_i$  = Nilai aktual sampel ke-i  
 $\hat{y}_i$  = Nilai prediksi untuk sampel ke-i

Dalam persamaan 2.2,  $y_i$  merujuk pada nilai aktual sampel ke-i dalam dalam *dataset*, sedangkan  $\hat{y}_i$  merujuk pada nilai prediksi untuk sampel ke-i. Hasil pengkuadratan nilai selisih dari keduanya menghasilkan *squared error*.

#### 2.5.1.2. Absolute Error

*Absolute error* / Kesalahan Absolut adalah nilai absolut dari selisih antara nilai prediksi dan actual. Secara matematis, persamaan *absolute error* dituliskan sebagai berikut

$$Absolute\ error = |y_i - \hat{y}_i|^2 \quad (2.3)$$

Dimana :

$y_i$  = Nilai aktual sampel ke-i  
 $\hat{y}_i$  = Nilai prediksi untuk sampel ke-i

Dalam persamaan 2.3,  $y_i$  merujuk pada nilai aktual sampel ke-i dalam dataset, sedangkan  $\hat{y}_i$  merujuk pada nilai prediksi untuk sampel ke-i. Nilai *absolute error* didapatkan dengan mengabsolutkan nilai selisih antara keduanya lalu dikuadratkan

### 2.5.1.3. Huber

*Huber error* adalah *loss function* yang menawarkan hasil terbaik dengan menyeimbangkan antara *squared error* dan *absolute error*. *Huber error* menghitung *squared error* ketika selisih antara prediksi dan nilai target masih kecil, dan menghitung *absolute error* ketika selisihnya besar (Balasundaram & Prasad, 2020). Persamaan huber error dituliskan sebagai berikut

$$Huber\ Loss = \begin{cases} \frac{1}{2}(y_1 - \hat{y}_i)^2 \\ \delta|y_1 - \hat{y}_i| - \frac{1}{2}\delta^2 \end{cases} \quad (2.4)$$

$IF |y_1 - \hat{y}_i| \leq \delta, otherwise$

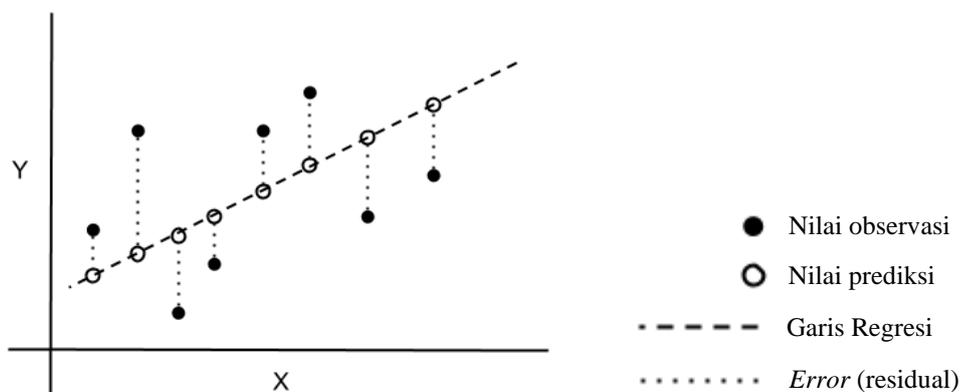
Dimana :

$\delta$  = Parameter penentu batas antara *squared error* dan *absolute error*  
 $n$  = Jumlah sampel data  
 $y_i$  = Nilai aktual untuk sampel ke -i  
 $\hat{y}_i$  = Nilai prediksi model untuk sampel ke-i

### 2.5.1.4. Mean Squared Error

*Mean Squared Error* (MSE) adalah salah satu metrik evaluasi yang umum digunakan dalam *machine learning* untuk mengukur seberapa dekat nilai prediksi model dengan nilai aktual dari data. Ini adalah nilai rata-rata dari *squared error* (Nuha, 2023).

*Squared Error* adalah perbedaan antara nilai prediksi yang dihasilkan oleh model dan nilai aktual dari data, yang kemudian dikuadratkan. Pengkuadratan ini bertujuan untuk menghindari adanya nilai positif dan negatif yang bisa saling meniadakan satu sama lain saat dijumlahkan. Dengan ini, memastikan bahwa setiap perbedaan yang signifikan tetap memiliki dampak pada keseluruhan evaluasi, tanpa dipengaruhi oleh tanda positif atau negatifnya. Representasi dari *squared error* ditampilkan sebagai berikut



Gambar 2.6 Representasi Visual Mean Squared Error (MSE)

Berdasarkan gambar 2.6, garis putus-putus berupa titik adalah *error*, MSE adalah rata-rata dari semua nilai kuadrat *error* yang ada. Ini memberikan gambaran tentang seberapa jauh rata-rata prediksi dari nilai aktual dari data. Semakin kecil nilai MSE, semakin baik performa model dalam memprediksi data. Persamaan MSE dituliskan sebagai berikut

$$MSE = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n} \quad (2.5)$$

Dimana:

- $n$  = jumlah sampel dari data
- $y_i$  = nilai aktual dari data ke-i.
- $\hat{y}_i$  = nilai prediksi dari model untuk data ke-i.

Seperti yang sudah di jalaskan *sebelumnya*, mse adalah nilai rata-rata dari *squared-error*, sedang *squared-error* adalah kuadrat dari *error*. Nilai *error* didapat dari selisih antara nilai prediksi model dan nilai actual seperti yang ditunjukkan oleh persamaan 2.5. Contoh perhitungan mse disajikan dalam tabel dibawah ini.

Tabel 2.2 Contoh perhitungan MSE menggunakan sampel data

index	aktual	prediksi	<i>Error</i> (aktual-prediksi)	<i>Squared-Error</i> ( <i>Error</i> <sup>2</sup> )
0	32.727	34.53104	-1.80404	3.254555
1	37.688	39.57304	-1.88504	3.553393
2	41.849	42.20257	-0.35357	0.125008
			MSE	2.310985

Hasil mse 2,31 dalam tabel 2.2 menunjukkan bahwa nilai MSE yang didapat untuk data diatas kecil mendekati 0, hal ini menunjukkan bahwa prediksi model sangat baik, karn rata-rata nilai prediksi model hamper sesuai dengan nilai actual.

### 2.5.2. *R-Squared*

*R-Squared* ( $R^2$ ) adalah salah *satu* metrik evaluasi yang umum digunakan untuk mengukur seberapa baik model melakukan prediksi dalam regresi linier. Nilai  $R^2$  menunjukkan proporsi variabilitas dalam variabel terikat yang dapat dijelaskan oleh variabel independen dalam model regresi. Dalam kata lain, metrik ini digunakan untuk mengetahui seberapa penting variabel independen dalam menjelaskan atau memprediksi variabel terikat dalam sebuah model regresi.

Persamaan  $R^2$  dituliskan sebagai berikut(Miles, t.t.).

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2} \quad (2.5)$$

Dimana:

$SS_{RES}$  = jumlah kuadrat residu (*residual sum of squares*), yaitu jumlah dari kuadrat selisih antara nilai-nilai prediksi dan nilai-nilai actual

$SS_{TOT}$  = adalah total jumlah kuadrat (*total sum of squares*), yaitu jumlah dari kuadrat selisih antara nilai-nilai actual dan nilai rata-rata dari variabel terikat.

$n$  = jumlah sampel data  
 $y_i$  = nilai aktual untuk sampel ke -i  
 $\hat{y}_i$  = nilai prediksi model untuk sampel ke-i  
 $\bar{y}_i$  = rata-rata dari nilai aktual

Nilai  $R^2$  berkisar antara 0 dan 1, di mana semakin dekat ke 1, semakin baik model tersebut cocok dengan data. Misalnya, jika  $R^2$  adalah 0, itu berarti model tidak menjelaskan *variasi* sama sekali, sedangkan jika  $R^2$  adalah 1, itu berarti model menjelaskan semua variasi dalam data.  $R^2$  sering digunakan sebagai ukuran penting dalam mengevaluasi kualitas model regresi.

### 2.5.3. Mean Absolute Percentage Error (MAPE)

MAPE adalah salah satu metrik evaluasi dalam *machine learning* yang umum digunakan sebagai pengukur keakuratan suatu algoritma prediksi. MAPE dinyatakan dalam persentase dan dihitung sebagai rata-rata dari selisih persentase absolut antara nilai aktual dan nilai prediksi. Persamaan dari MAPE dituliskan sebagai berikut dalam persamaan 2.5 :

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i} \quad (2.5)$$

Dimana:

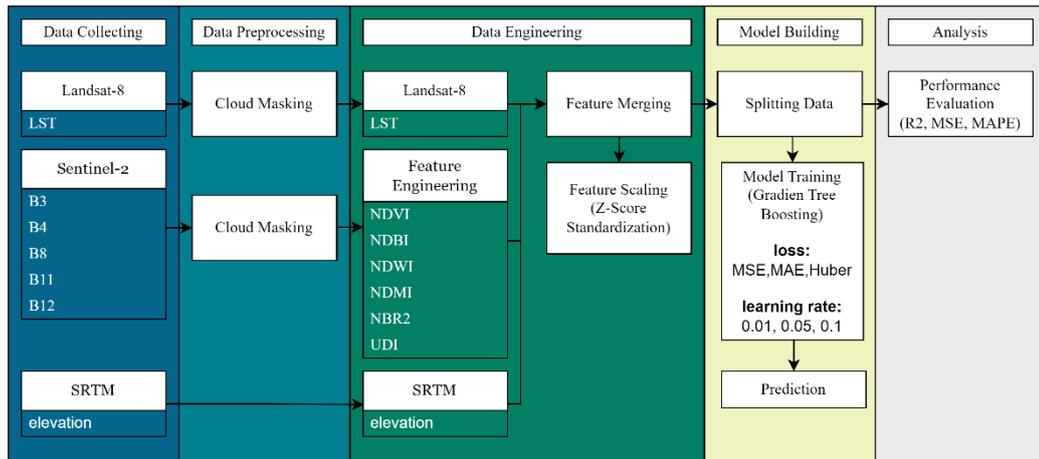
$n$  = jumlah sampel dari data  
 $y_i$  = nilai aktual dari data ke-i.  
 $\hat{y}_i$  = nilai prediksi dari model untuk data ke-i.

N merujuk pada jumlah total sampel data,  $y_i$  merujuk pada nilai aktual dan  $\hat{y}_i$  merujuk pada nilai prediksi.

## BAB III

### DESAIN DAN IMPLEMENTASI

#### 3.1. Desain Penelitian



Gambar 3.1 Alur Desain Penelitian

Dalam Gambar 3.1, Alur penelitian ini dibagi menjadi 5 bagian, yakni *data collecting*, *data preprocessing*, *data engineering*, *model building* dan *analysis* dimana tiap bagian ini memiliki sub-bagiannya masing-masing.

#### 3.2. Data Collecting

*Dataset* diperoleh dari situs penyedia data citra satelit yakni *Google Earth Engine*.

Yakni **Landsat-8** Sentinel-2, dan **SRTM** dalam rentang tahun 2015-2023 yang mana memiliki beberapa atribut sebagai berikut dituliskan dalam tabel 3.1

Tabel 3.1 Detil Dataset

No	Citra	Band	Deskripsi	Panjang Gelombang	Scale	Offset
1	Landsat-8	STB10	LST	10.60-11.19 $\mu\text{m}$	0.00341802	149

No	Citra	Band	Deskripsi	Panjang Gelombang	Scale	Offset
2	Sentinel-2	B3	GREEN	560nm (S2A) / 559nm (S2B)	0.0001	-
		B4	RED	664.5nm (S2A) / 665nm (S2B)	0.0001	-
		B8	Near Infrared (NIR)	835.1nm (S2A) / 833nm (S2B)	0.0001	-
		B11	Shortwave infrared 1 (SWIR 1)	1613.7nm (S2A) / 1610.4nm (S2B)	0.0001	-
		B12	Shortwave infrared 2 (SWIR 2)	2202.4nm (S2A) / 2185.7nm (S2B)	0.0001	-
3	SRTM	elevation	Elevation	-	-	-

Penggunaan *dataset* ini merujuk pada penelitian terdahulu yang mana dataset ini digunakan untuk menganalisis UHI (Furuya *et al.*, 2023). Data yang dipakai adalah data pada rentang tahun 2019-2023, hal ini dikarenakan data satelit sentinel-2 baru tersedia secara global pada desember 2018, karna itu penulis menggunakan data dimulai dari tahun 2019. Poin penting mengapa menggunakan *dataset* ini karna ketiga *data* ini memiliki unsur penting yang menjadi variabel dalam proses analisa UHI.

### 3.2.1. Landsat-8

Landsat-8 merupakan satelit penginderaan jauh yang dioperasikan oleh *National Aeronautics and Space Administration* (NASA) dan *United States Geological Survey* (USGS) Amerika Serikat.

Alasan utama penggunaan data satelit Landsat adalah ketersediaan sensor citra termal yang ditunjukkan oleh *band* STB10 yang mana menunjukkan suhu permukaan dengan kata lain adalah variabel LST yang merupakan faktor kunci dalam analisis UHI. Rentang waktu data satelit Landsat-8 mencakup periode tahun 2013 hingga sekarang. Satelit Landsat sebenarnya memiliki beberapa *band* yang diperlukan dalam penelitian ini namun data satelit ini memiliki resolusi yang lebih rendah jika dibandingkan dengan data yang diperoleh dari satelit Sentinel.

Dengan beberapa keunggulan diatas, peneliti mempertimbangkan untuk menggunakan data landsat-8 untuk penelitian ini.

### **3.2.2. Sentinel-2**

Sentinel-2 adalah salah satu instrumen yang terdapat pada satelit Sentinel. Instrumen ini mengumpulkan data citra dalam berbagai panjang gelombang, termasuk *red, green, blue, near-infrared, dan short-wave infrared*. Data yang terkumpul disimpan dalam berbagai kanal sesuai dengan panjang gelombang masing-masing, tiap kanal menunjukkan berbagai unsur yang diperlukan untuk analisis variabel yang diperlukan dalam menganalisis UHI. (Mushore *et al.*, 2022).

Data dari satelit ini dirancang untuk dimodifikasi dan disesuaikan dalam berbagai area tematik, seperti perencanaan tata ruang, pemantauan agro lingkungan, pemantauan air, pemantauan hutan dan vegetasi, pemantauan karbon tanah, sumber daya alam, serta pemantauan tanaman global. (Mushore *et al.*, 2022).

Keunggulan utama Data Sentinel-2 dibandingkan dengan Landsat-8 adalah resolusi citra yang lebih baik. Namun, satelit ini tidak mempunyai sensor termal

untuk menangkap suhu permukaan (LST) serta rentang waktu pengumpulan data yang terbatas mulai dari tahun 2017.

Luas cakupan dari satelit Sentinel-2 dapat terbilang universal, dimana juga mencakup area yang akan dianalisis dalam penelitian ini, yakni Kota Malang. Beberapa hal yang dijelaskan sebelumnya menjadi alasan mengapa peneliti memutuskan untuk menggunakan data Sentinel-

### **3.2.3. Shuttle Radar Topography Mission (SRTM)**

Data *Shuttle Radar Topography Mission* (SRTM) adalah sebuah dataset digital elevation model (DEM) global yang dikumpulkan oleh misi radar pesawat ulang-alik Space Shuttle Endeavour pada tahun 2000. Misi ini diluncurkan oleh NASA dan Badan Antariksa Jerman (DLR) dengan tujuan untuk menghasilkan peta topografi global dengan resolusi elevasi yang tinggi.

SRTM menghasilkan data DEM yang mencakup hampir seluruh permukaan bumi dengan resolusi spasial yang cukup tinggi. Data satelit ini menyediakan informasi tentang ketinggian permukaan bumi di berbagai lokasi, termasuk di wilayah perkotaan.

Kaitan data SRTM terutama band elevasi dengan UHI pernah diteliti pada tahun 2018 di Kanada yang menganalisis dan memodelkan UHI di 20 kota Kanada, penelitian ini menemukan bahwa korelasi elevasi terhadap perubahan UHI adalah negatif. Namun, korelasi negatif ini bukan berarti tidak ada sama sekali melainkan sangat kecil. Ini digambarkan dengan semakin tinggi elevasi suatu kota, semakin kecil pengaruhnya terhadap peningkatan UHI di masa depan (Gaur *et al.*, 2018).

Kota Malang yang merupakan wilayah yang diteliti dalam penelitian ini termasuk kedalam wilayah dataran tinggi yang secara geografis dikelilingi pegunungan. Hal yang menariknya adalah masih terjadi fenomena UHI, ini membuktikan penelitian sebelumnya bahwa elevasi tidak terlalu berpengaruh terhadap UHI. Namun, berdasar pada penjelasan sebelumnya, penulis memilih untuk tetap menggunakan variabel elevasi untuk dapat menganalisis perubahan UHI secara holistik.

### **3.3. Data Preprocessing**

Preproses Data (*Data Preprocessing*) yakni mempersiapkan data dengan cara menganalisis, membersihkan dan mengolah data supaya ketika diterapkan model algoritma akan menghasilkan performa yang baik.

Setelah dilakukan analisis citra, ternyata masih terdapat lapisan awan yang memberikan pengaruh negatif dalam pemrosesan data. Maka dari itu perlu dilakukan penghapusan awan (*cloud masking*).

Cloud masking atau penyaringan awan ialah salah satu teknik dalam pemrosesan citra jarak jauh untuk menghiangkan awan dari citra. Ini adalah langkah awal yang penting dalam mempersiapkan data sebelum diolah dalam penginderaan jauh, terutama untuk data dari satelit yang memiliki resolusi tinggi (Malladi *et al.*, 2019). Tujuannya untuk menghapus piksel-piksel yang terpengaruh oleh awan dari citra satelit.

### 3.4. Data Engineering

Terdapat beberapa tahapan dalam *data engineering*, yakni *feature engineering* dan *features merging*.

#### 3.4.1. Feature Engineering

Pada tahap ini, atribut-atribut data sebelumnya akan diolah sehingga didapat indeks pengukuran yang telah dijelaskan sebelumnya seperti NDVI, NDWI, NDBI, NDMI, NBR, UDI. Secara ringkas, beberapa fitur yang dihitung dapat disajikan sebagai berikut

Tabel 3.2 Fitur-fitur untuk prediksi UHI

No	Nama	Keterangan	Rumus
1	NDVI	Indeks untuk mengidentifikasi area hijau	$NDVI = \frac{NIR - RED}{NIR + RED} \quad (3.1)$
2	NDBI	Indeks untuk menganalisa kerapatan dan distribusi bangunan	$NDBI = \frac{SWIR2 - NIR}{SWIR2 + NIR} \quad (3.2)$
3	NDWI	Indeks untuk mengukur dan memetakan area berair	$NDWI = \frac{GREEN - NIR}{GREEN + NIR} \quad (3.3)$
4	NDMI	Indeks untuk mengukur kelembaban dan kelembaban tanah	$NDMI = \frac{NIR - SWIR1}{NIR + SWIR1} \quad (3.4)$
5	NBR2	Indeks untuk menganalisa sensitivitas vegetasi terhadap air	$NBR2 = \frac{SWIR1 - SWIR2}{SWIR1 + SWIR2} \quad (3.5)$
6	UDI	Indeks untuk menghitung seberapa padat sebuah area perkotaann	$\begin{aligned} Built &= NDBI \geq -0,1 \\ \cap NBR &\leq 0,2 \\ UDI &= Built_{mean} \end{aligned} \quad (3.6)$
7	elevation	Indeks untuk mengukur ketinggian dalam sebuah wilayah	-
8	LST	Suhu permukaan dalam data landsat 8	-

Tabel 3.2, memperlihatkan keterangan dan rumus perhitungan dari fitur yang akan ditambahkan. Secara matematis, perhitungan dari fitur-fitur diatas dituliskan dalam tabel 3.3 berikut

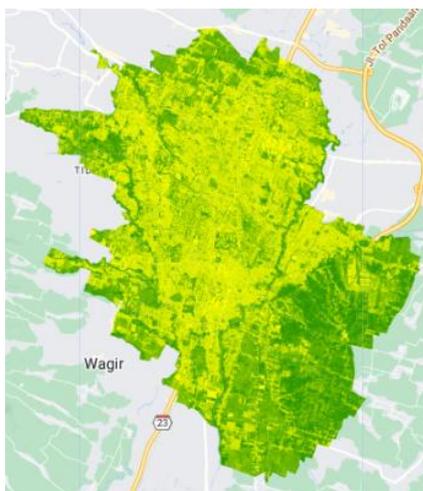
Tabel 3.3 Perhitungan Fitur

Var	Band 3 (G)	Band 4 (R)	Band 8 (N)	Band 11 (S1)	Band 12 (S2)	Perhitungan	Hasil
NDVI	0.0741	0.0668	0.23855	0.19025	0.1289	$NDVI = \frac{0.23855 - 0.0668}{0.23855 + 0.0668}$	0.144
NDBI						$NDBI = \frac{0.1289 - 0.23855}{0.1289 + 0.23855}$	0.149
NDWI						$NDWI = \frac{0.0741 - 0.23855}{0.0741 + 0.23855}$	-0.187
NDMI						$NDMI = \frac{0.23855 - 0.19025}{0.23855 + 0.19025}$	-0.174
NBR2						$NBR2 = \frac{0.19025 - 0.1289}{0.19025 + 0.1289}$	0.192

Tabel 3.3 menunjukkan hasil perhitungan untuk tiap fitur menggunakan sampel data.

#### 3.4.1.1. Normalized Difference Vegetation Index (NDVI)

*Normalized Difference Vegetation Index* (NDVI) adalah sebuah indeks yang umum digunakan dalam penginderaan jauh untuk mengukur jumlah dan kepadatan vegetasi di suatu daerah. Dalam peta, analisis vegetasi digambarkan sebagai berikut



Gambar 3.2 Visualisasi NDVI

Gambar 3.2 menampilkan visualisasi NDVI, semakin terang warnanya menunjukkan semakin sedikit vegetasi atau daerah hijau. Sebaliknya, semakin

gelap warnanya, semakin padat pula vegetasi area tersebut. Nilai NDVI di dapat dengan menggunakan *band Near Infrared (NIR)* dan *RED* dari *dataset Sentinel-2*.

Persamaan NDVI di nyatakan sebagai berikut (Wachid & Prananing Tyas, 2022)

$$NDVI = \frac{NIR - RED}{NIR + RED} \quad (3.1)$$

Dimana:

*NIR* = Band Near Infrared Sentinel-2

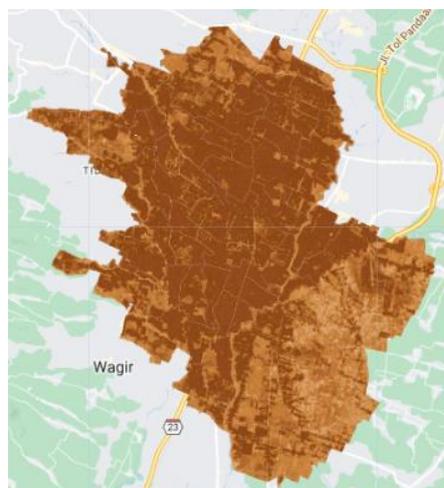
*RED* = Band Red Sentinel-2

Dalam persamaan 3.1 tersebut, *NIR* mengacu pada radiasi inframerah dekat yang berasal dari piksel, sedangkan *RED* merujuk pada radiasi cahaya merah dari piksel tersebut.

Rentang nilai NDVI berkisar dari -1 (yang umumnya menunjukkan area yang terdapat air) hingga +1 (menandakan keberadaan vegetasi yang sangat lebat).

#### 3.4.1.2. NDBI

*Difference Built-up Index (NDBI)* adalah sebuah indeks yang digunakan dalam penginderaan jauh untuk mengevaluasi distribusi dan densitas bangunan dalam sebuah daerah atau wilayah.



Gambar 3.3 Visualisasi NDBI

NDBI dapat membantu dalam identifikasi dan pemetaan perkembangan perkotaan, pemantauan pertumbuhan kota, serta analisis perubahan lahan. Nilai NDBI didapat dengan perhitungan menggunakan *band shortwave infrared* (SWIR) dan NIR dari *dataset* Sentinel-2. Persamaan NDBI dinyatakan sebagai berikut (Zha *et al.*, 2003b)

$$NDBI = \frac{SWIR2 - NIR}{SWIR2 + NIR} \quad (3.2)$$

Dimana:

*SWIR2* = Band Short Wave Infrared 2 Sentinel-2

*NIR* = Band Near Infrared Sentinel-2

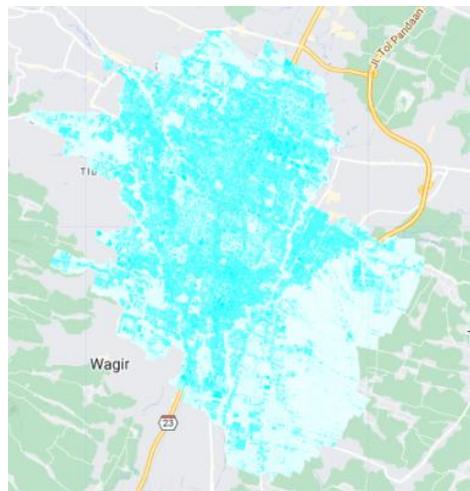
Rentang nilai NDBI adalah dari -1 hingga +1, sama seperti NDVI. Lebih jelasnya dapat dilihat pada Gambar 3.3 Dimana semakin terang warnnya, maka kepadatan bangunan di area tersebut rendah atau bahkan tidak ada, sedang semakin gelap warnanya menunjukkan kepadatan bangunan di area tersebut tinggi.

Dalam penelitian yang dilakukan tinufi dan Rahayu, mereka mengklasifikasikan nilai-nilai NDBI kedalam 3 bagian sebagai berikut (Trinufi & Rahayu, 2020).

1. Nilai negatif (kurang dari 0) menunjukkan daerah yang cenderung memiliki air atau vegetasi.
2. Nilai mendekati 0 menunjukkan daerah yang umumnya terdiri dari tanah kosong atau vegetasi yang jarang.
3. Nilai positif (lebih dari 0) menunjukkan daerah yang cenderung memiliki bangunan atau material padat.

### 3.4.1.3. NDWI

*Normalized Difference Water Index* (NDWI) merupakan metode untuk memetakan fitur air terbuka dalam citra digital yang diindera jauh, dengan menggunakan *band* NIR dan cahaya hijau (*GREEN*) dari dataset Sentinel-2. NDWI dapat berguna untuk pemetaan dan identifikasi area badan air seperti danau sungai dll.



Gambar 3.4 Visualisasi NDWI

Seperti yang dapat dilihat dalam Gambar 3.4 Semakin cerah warnanya, menunjukkan area yang minim atau bahkan tidak ada badan air, Sedangkan sebaliknya, semakin pekat warna menunjukkan daerah yang memiliki badan air.

Nilai NDWI didapat dengan perhitungan menggunakan *band GREEN* dan NIR. Persamaan NDWI dituliskan sebagai berikut (Octarina *et al.*, 2019).

$$NDWI = \frac{GREEN - NIR}{GREEN + NIR} \quad (3.3)$$

Dimana:

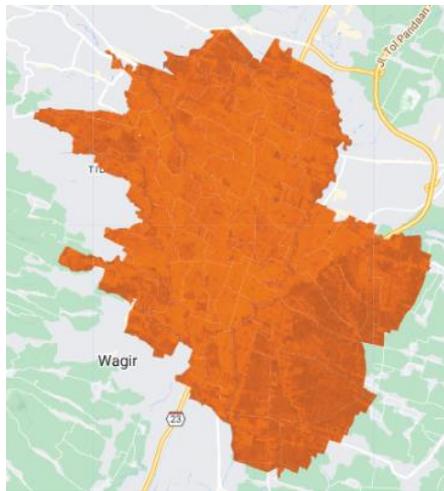
*GREEN* = Band Green Sentinel-2

*NIR* = Band Near Infrared Sentinel-2

Rentang nilai NDWI ada diantara -1 hingga 1. Nilai positif menggambarkan area yang menunjukkan adanya air. Sementara nilai negatif menggambarkan area yang tidak ada air.

#### 3.4.1.4. NDMI

*Normalized Difference Moisture Index* (NDMI) adalah sebuah metrik untuk mengukur kelembaban dan kelembaban tanah dalam citra penginderaan jauh.



Gambar 3.5 Visualisasi NDMI

Sama seperti sebelumnya, Gambar 3.5 Adalah visualisasi NDMI Dimana semakin cerah warnanya, maka semakin tidak lembab area tersebut. Sebaliknya, semakin gelap warnanya , maka semakin lembab pula area tersebut.

NDMI dapat dihitung menggunakan informasi dari *band* NIR dan SWIR 1 dari *dataset* Sentinel-2. Persamaan NDMI dinyatakan sebagai berikut (Muryati, 2019)

$$NDMI = \frac{NIR - SWIR1}{NIR + SWIR1} \quad (3.4)$$

Dimana:

*SWIR1* = Band Short Wave Infrared 1 Sentinel-2

*NIR* = Band Near Infrared Sentinel-2

Rentang nilai NDIM berkisar antara -1 hingga 1. Dimana, nilai positif menggambarkan kelembaban atau adanya air, sedangkan nilai negatif menggambarkan kekeringan atau tidak ada air.

#### 3.4.1.5. Normalize Burn Ratio (NBR2)

*Normalized Burn Ratio* (NBR) adalah salah satu indeks yang digunakan dalam penginderaan jauh untuk memantau dan menganalisis dampak kebakaran hutan atau lahan. Perhitungan indeks ini menggabungkan informasi dari dua panjang gelombang cahaya yang berbeda, yaitu inframerah dekat (NIR) dan inframerah jauh (SWIR) dari dataset Sentinel-2.

NBR2 merupakan adaptasi dari *Normalized Burn Ratio* (NBR), yang bertujuan untuk meningkatkan fokus pada respons vegetasi terhadap kelembaban dan dapat memberikan manfaat dalam analisis pemulihan setelah kebakaran. Perhitungannya melibatkan perbandingan nilai *band* NIR dengan *band* SWIR1 sebagai pengganti *band* NIR dalam perhitungan NBR asli (U.S. Geological Survey, 2022). Formula NBR2 dituliskan sebagai berikut

$$NBR = \frac{SWIR1 - SWIR2}{SWIR1 + SWIR2} \quad (3.5)$$

Dimana:

*SWIR2* = Band Short Wave Infrared 2 Sentinel-2

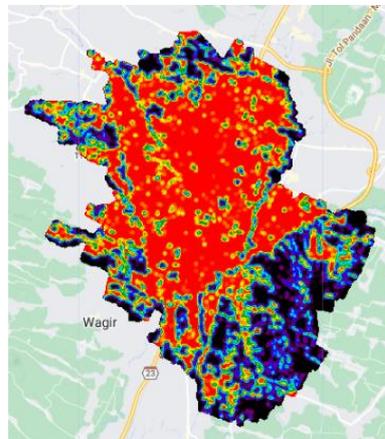
*NIR* = Band Near Infrared Sentinel-2

NBR2 memiliki nilai yang berkisar antara -1 hingga 1, dimana nilai -1 atau mendekatinya menunjukkan adanya kerusakan yang signifikan pada area tersebut,

sedangkan nilai 1 atau mendekatinya menunjukkan sedikit atau bahkan tidak ada vegetasi yang terpengaruh oleh kerusakan (U.S. Geological Survey, 2022).

#### 3.4.1.6. Urban Density Index

Indeks Kepadatan perkotaan (*Urban Density Index (UDI)*) digunakan untuk menghitung seberapa padat sebuah area perkotaan. Intensitas dari UDI dapat divisualisasikan seperti Gambar 3.6. yang disajikan dibawah ini.



Gambar 3.6 Visualisasi Urban Density Index

Penelitian Yunfei Li dkk yang menginvestigasi pengaruh kepadatan dan morfologi terhadap intensitas UHI, berkesimpulan bahwa kepadatan perkotaan berkaitan langsung dengan intensitas UHI.

Berdasarkan pencarian pada beberapa literasi, tidak ada yang menjelaskan persamaan UDI secara eksplisit. Namun, seorang ahli penginderaan jauh di salah satu kanal media *online youtube* menyebut bahwa UDI diperoleh dari nilai tengah *Built-up* (BU) Dimana persamaan BU indeks ialah sebagai berikut (Ramadhan, 2023):

$$BuildUp = (NDBI > -0,5) \cap (NBR \leq 0,2) \cap (NDWI \leq 0) \quad (3.6)$$

Dimana:

*NDBI* = Nilai index NDBI  
*NBR* = Nilai index NBR  
*NDWI* = Nilai index NDWI

Dari hasil perhitungan 3.6 diatas, nilai UDI dapat diperoleh dengan mencari nilai tengah dari *BuildUp*. Persamaannya dituliskan sebagai berikut

$$UDI = BuildUp_{mean} \quad (3.7)$$

Dimana:  
*BuildUp<sub>mean</sub>* = Nilai tengah indeks *BuildUp*

Dalam persamaan 3.7, UDI ialah nilai dari *Urban Density Indeks* (UDI), NDBI adalah nilai NDBI, NBR adalah nilai NBR dan NDWI adalah nilai NDWI. Biasanya UDI digunakan sebagai pendukung bagi indeks lainnya. Rentang dari indeks ini berkisar antara -1 hingga 1, dimana -1 (Ramadhan, 2023).

#### 3.4.1.7. Elevasi

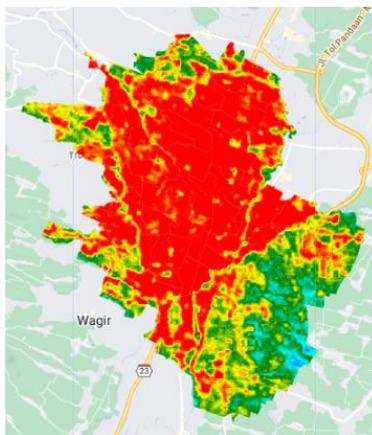
Elevasi (*elevation*) adalah ketinggian suatu lokasi di atas atau di bawah permukaan laut. Hal ini mengacu pada tinggi relatif suatu titik di permukaan bumi terhadap tingkat laut. Informasi tentang elevasi sangat penting dalam pemodelan dan pemahaman topografi suatu wilayah. Elevasi dapat mempengaruhi pola aliran udara dan distribusi suhu di dalam suatu kota atau daerah perkotaan yang mana ini berkaitan dengan UHI. Wilayah yang lebih tinggi memiliki kecenderungan untuk memiliki suhu yang lebih rendah karena udara cenderung dingin di tempat yang lebih tinggi. Sebaliknya, daerah yang lebih rendah atau cekungan (misalnya lembah) cenderung mengalami penumpukan udara dan panas, yang dapat menyebabkan suhu yang lebih tinggi.

Penelitian belakangan di Kota Fransisco yang bertujuan menganalisa parameter morfologi baru untuk prediksi UHI, menemukan bahwa elevasi memiliki dampak yang signifikan terhadap terjadinya fenomena UHI (Equere *et al.*, 2020).

Data elevasi dalam penelitian ini didapat dari citra NASA SRTM Digital Elevation 30m. Data ini memiliki band yang khusus elevasi dengan resolusi gambar yang bagus di 30m. Rentang nilai *elevation* berkisar antara -10 hingga 6500.

#### 3.4.1.8. *Land Surface Temperature (LST)*

*Land Surface Temperature (LST)* ialah suhu di permukaan bumi. Indikator LST berkaitan erat dengan UHI seperti yang telah dijelaskan sebelumnya dalam sub bab mengenai panas. Pada penelitian ini, Data LST didapat dari dataset Landsat-8 pada band STB10.



Gambar 3.7 Pemetaan LST

Pada Gambar 3.7, semakin merah warna area menunjukkan semakin tinggi pula intensitas UHI-nya, sebaliknya, jika semakin gelap warnanya (warna paling gelap adalah hitam), menunjukkan semakin rendah pula UHI-nya. Rentang LST sama dengan indeks indeks sebelumnya, LST memiliki rentang nilainya yang

digambarkan oleh warnanya. Pada kasus LST, warna gelap menunjukkan semakin rendahnya suhu dengan nilai -1, sedang warna yang semakin terang menuju merah menunjukkan suhu meingkat atau tinggi dengan nilai 1.

### 3.4.2. Feature Merging

Ditahap rekayasa fitur sebelumnya, tiap fitur indeks tersimpan sebagai *dataset* yang berbeda, LST masih dalam dataset Landsat-8, NDVI, NDWI, NDMI, NDMI, NBR2 ditabel yang berbeda, serta data elevation juga ada dalam dataset yang berbeda. Karnanya perlu untuk menggabungkannya menjadi satu *dataset* utuh. Hal ini bertujuan untuk menyesuaikan format data dengan pengaplikasian algoritma GTB, semua fitur baru yang sudah dibuat dan terpecah belah menjadi beberapa dataset akan digabung menjadi satu *dataset* sehingga didapat dataset keseluruhan sebagai berikut

Tabel 3.4 Data setelah digabung

LST	NBR2	NDBI	NDMI	NDVI	NDWI	UDI	elevation
40.163	-0.042	0.199	-0.158	-0.033	0.009	0.931	491
39.746	-0.065	0.283	-0.222	-0.017	0.008	0.966	466
40.655	-0.018	0.173	-0.155	-0.001	0.001	0.931	424
42.532	0.028	-0.009	-0.019	-0.029	0.032	0.966	431
43.758	-0.018	0.237	-0.22	-0.02	0.012	0.966	463

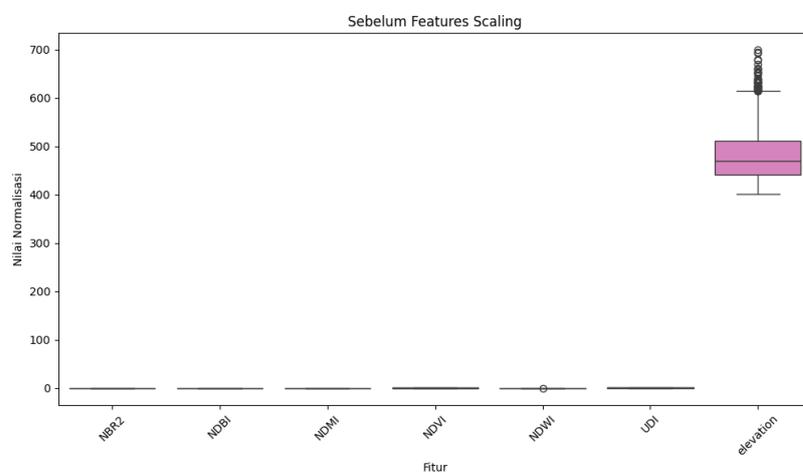
Tabel 3.4 menunjukkan hasil dari proses *feature merging* dimana *dataset* yang berisi fitur-fitur seperti LST, NDVI dan fitur lainnya disatukan.

### 3.4.3. Features Scaling

*Feature scaling* / penskalaan fitur adalah proses untuk mengubah skala atau rentang nilai dari berbagai fitur dalam dataset menjadi seragam atau relatif mendekati distribusi normal. Tujuannya untuk menghindari bias yang mungkin

timbul akibat perbedaan skala antar fitur, serta untuk meningkatkan kinerja algoritma pembelajaran mesin yang sensitif terhadap skala. *Feature scaling* sangat penting untuk algoritma yang menggunakan teknik penurunan gradien seperti GTB, karna proses optimasi dalam algoritma ini melibatkan perhitungan yang sangat sensitif terhadap perbedaan skala fitur, dimana jika ada fitur yang memiliki skala yang terlampau jauh dari fitur lainnya, maka fitur ini akan memiliki pengaruh besar dibanding fitur lainnya terhadap kesimpulan yang dibuat model sehingga berdampak pada hasil akhir prediksi. Dengan melakukan *feature scaling*, dapat memastikan bahwa setiap fitur memberikan kontribusi yang seimbang dalam membangun model machine learning.

Dalam konteks penelitian ini, *feature scaling* bertujuan untuk menyeragamkan perbedaan rentang nilai yang ada di fitur 'elevation' dengan fitur lainnya dan mengoptimalkan pelatihan model dengan algoritma GTB. Hal ini dikarenakan pada fitur tersebut memiliki rentang nilai yang jauh dibanding dengan fitur lainnya seperti yang ditunjukkan dalam grafik rentang nilai tiap fitur dalam gambar 3.8 dibawah ini.



Gambar 3.8 Rentang nilai tiap fitur sebelum dilakukan normalisasi

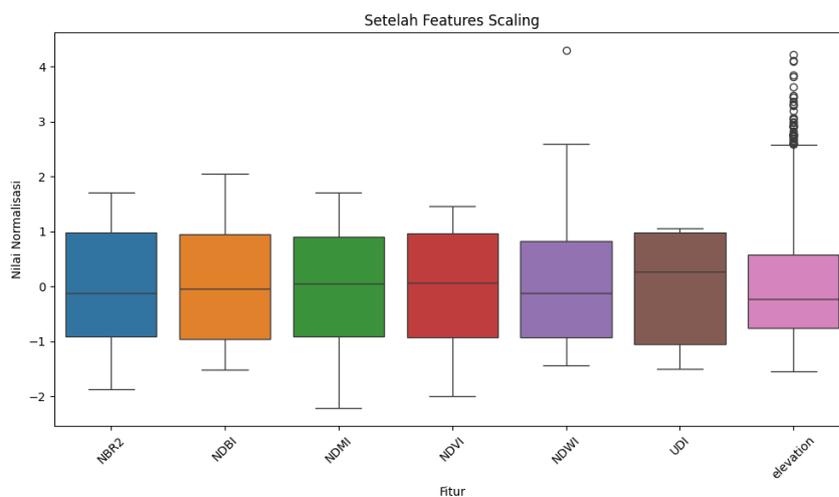
Dari gambar 3.8, dapat dilihat bahwa rentang nilai fitur *elevation* memiliki jauh diatas rata-rata rentang nilai fitur lainnya, dimana rentang nilai fitur-fitur selain *elevation* berkisar antara -1 hingga 1, sedang *elevation* memiliki rentang 400 hingga 6500. Untuk menangani masalah ini, salah satu teknik *feature scaling* yang umum digunakan dalam machine learning ialah *Z-Score Standardization* (Whendasmoro & Joseph, 2022). Persamaan *z-score standardization* sebagai berikut

$$Z = \frac{x - \mu}{\sigma} \quad (3.8)$$

Dimana:

$x$  = skor  
 $\mu$  = mean  
 $\sigma$  = standar deviasi

*Z-Score Standardization* adalah salah satu teknik normalisasi untuk mengubah setiap nilai fitur menjadi *z-score*. *Z-score* adalah jumlah standar deviasi dari nilai tersebut terhadap rata-rata fitur. Dengan kata lain, teknik ini mengubah distribusi nilai menjadi distribusi normal dengan rata-rata 0 dan standar deviasi 1. Hasil dari proses ini ialah sebagaimana yang disajikan dalam gambar 3.9 berikut ini



Gambar 3.9 Rentang nilai tiap fitur setelah dilakukan normalisasi

Gambar 3.9 menunjukkan perubahan rentang nilai yang awalnya jauh berbeda menjadi relatif seragam. Dengan seragamnya skala dari tiap fitur, dapat memastikan tidak ada fitur yang mendominasi dan menghindari adanya bias pada hasil prediksi model. Hasil proses *feature scaling* pada sampel data ialah sebagai berikut

Tabel 3.5 Data sebelum dan sesudah normalisasi

SEBELUM NORMALISASI							
No	NDVI	NDMI	NDBI	NDWI	UDI	Elevation	LST
1	0.144	-0.174	0.149	-0.187	1.000	444	41.1
2	0.076	-0.183	0.103	-0.100	1.000	438	40.6
3	0.182	-0.144	0.093	-0.238	1.000	507	40.1
SETELAH NORMALISASI							
No	NDVI	NDMI	NDBI	NDWI	UDI	Elevation	LST
1	0.145	0.114	0.867	0.800	1.000	0.146	0.784
2	0.062	0.237	0.821	0.920	1.000	0.126	0.755
3	0.192	0.172	0.811	0.729	1.000	0.361	0.722

Tabel 3.5 menunjukkan hasil normalisasi menggunakan *z-score* standardization pada 3 sampel *dataset*. Dapat dilihat terjadi perubahan nilai terutama pada kolom “*Elevation*” dan “*LST*” yang awalnya memiliki rentang 400-500 dan 40-41 menjadi 0,146 sampai 0,361.

### 3.5. Model Building

Tahap ini adalah tahap Pembangunan model ML. beberapa tahapan yang dilakukan pada tahap ini ialah *data splitting*, *model training* dan *prediction*.

#### 3.5.1. Data Splitting

*Data splitting* ialah proses membagi sampel data menjadi data latih (*train*) dan data uji (*test*). Secara visual, *data splitting* dapat digambarkan seperti berikut :

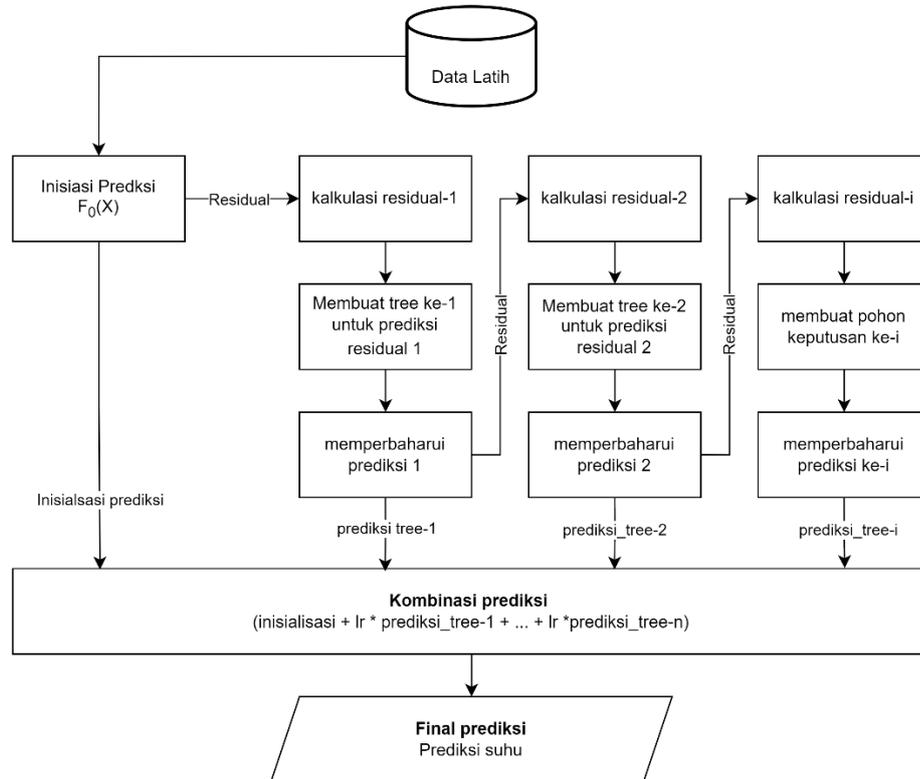


Gambar 3.10 Visualisasi *data splitting*

Dari Gambar 3.10, data latih adalah data yang digunakan untuk melatih model menggunakan algoritma *Gradient Tree Boosting* (GTB). Sedangkan data uji adalah data yang digunakan untuk menguji prediksi model. Penelitian ini menggunakan 3 rasio ukuran, yakni 70:30, 80:20 dan 90:10. Dimana 70%, 80% dan 90% untuk data latih dan sisanya 30%, 20% dan 10% untuk data uji.

### **3.5.2. Model Training**

Setelah data terbagi, selanjutnya ialah pembangunan model menggunakan algoritma *Gradient Tree Boosting* (GTB). Pada tahap ini, model akan dilatih menggunakan data latih sebelumnya untuk memahami pola dari data menggunakan algoritma *Gradient Tree Boosting* (GTB) yang telah dijabarkan dalam Bab II dengan diagram alir sebagai berikut.



Gambar 3.11 Alur proses algoritma GTB

Gambar 3.11 menunjukkan proses dalam algoritma *Gradient Tree Boosting* (GTB), dimana terdiri dari proses yang dilakukan secara iteratif meliputi inisiasi, kalkulasi *residual*, membuat pohon keputusan dll. Terkait contoh perhitungan manual untuk setiap proses penulis sertakan dalam bagian lampiran.

### 3.5.2.1. Inisialisasi prediksi

Inisialisasi dilakukan dengan menghitung nilai tengah dari kolom target. Implementasi dari pengambilan nilai tengah dilakukan dengan potongan kode berikut ini.

```

def train(self, X, y):
    self.y_mean = np.mean(y)
    pred_0 = np.array([np.mean(y)] * len(y))
    pred = pred_0.reshape(len(pred_0),1)

```

Gambar 3.12 Implementasi inisialisasi prediksi

Gamabr 3.12 menunjukkan potongan kode yang dijalankan ntuk melakukan inisialisas prediksi, Nilai *pred* dalam fungsi *train* adalah nilai inisialisasi prediksi yang merupakan nilai tengah dari target *y*.

### 3.5.2.2. Kalkulasi residual

Setelah menentukan inisialisasi prediksi, selanjutnya ialah kalkulasi dengan membandingkan nilai prediksi yang dalam hal ini adalah nilai inisialisasi prediksi dengan nilai aktual suhu permukaan (LST) dalam data sampel. Selisih antara nilai prediksi dan data aktual disebut sebagai *error* atau *residual*. Perhitungan *residual* ini dapat dituliskan sebagai berikut

$$r_{im} = - \left[ \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)} \quad \text{for } i = 1, \dots, n \quad (3.14)$$

Dimana:

$r_{i,m}$  = *Residual* untuk sampel ke-*i* pada pohon ke-*m*

$-\left[ \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)}$  = turunan dari *loss function* yang berhubungan dengan turunan dari inisialisasi prediksi  $\gamma$ .

Persamaan 3.14 adalah persamaan untuk menghitung *residual* untuk sampel ke-*I* pada pohon ke-*m*. Implementasi proses ini dalam *python* sebagai berikut

```

def __calculate_residual(self, y, y_pred):
    res = -(y-y_pred)
    return res

```

Gambar 3.13 Implementasi perhitungan residual

Dari Gambar 3.13, menunjukkan fungsi `__calculate_residual`, fungsi ini bertujuan untuk menghitung residual untuk tiap sampel dalam data latih. Nilai *residual* didapat dari selisih antara nilai target aktual  $y$  dikurangi nilai prediksi  $y_{pred}$ .

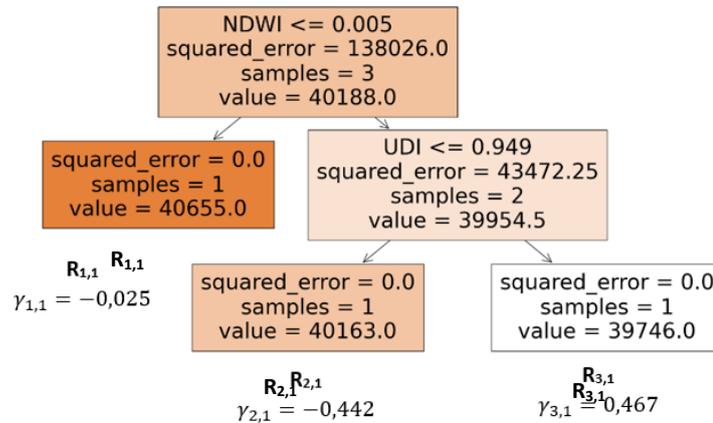
### 3.5.2.3. Membuat pohon keputusan

Hasil *residual* selanjutnya akan dijadikan target baru untuk pembuatan pohon (*tree*) berikutnya. Pembuatan *tree* ini dilakukan menggunakan *library* dari *sklearn* yakni `DecisionTreeRegressor`. Implementasi pembuatan *tree* dalam *python* disajikan sebagai berikut

```
def __create_tree(self, X, y):
    base = DecisionTreeRegressor(
        criterion='mse',max_depth=self.max_depth,
        min_samples_split=self.min_samples_split,
        min_samples_leaf=self.min_samples_leaf,
        max_features=self.max_features)
    base.fit(X,y)
    return base
```

Gambar 3.14 Implementasi pembuatan *tree*

Gambar 3.14 menunjukkan kode untuk pembuatan *tree* dengan data latih (*residual*) yang telah didapat sebelumnya. Hasil dari proses ini ialah sebuah nilai prediksi untuk target atau sering disebut nilai gamma. Fungsi ini selanjutnya akan dipanggil didalam fungsi *train* sebagai variabel *base* yang pada tahap selanjutnya akan di pakai dalam implementasi pembaharuan prediksi. Secara visual, *tree* yang dihasilkan oleh oleh kode diatasialah sebagai berikut

Gambar 3.15 Visualisasi *Tree-1*

Gambar 3.15 adalah visualisasi *tree* yang dihasilkan dari implementasi pembuatan *tree* pada Gambar 3.14.

#### 3.5.2.4. Memperbaharui Prediksi

Nilai *gamma* dari prediksi *tree* sebelumnya akan ditambahkan kedalam prediksi gabungan yang berisi prediksi sebelumnya (dalam konteks iterasi pertama, prediksi sebelumnya adalah inisialisasi prediksi ( $F_0(x)$ )). Hal ini dirumuskan sebagai berikut

$$F_m(x) = F_{m-1}(x) + V \sum_{j=1}^{J_m} \gamma_{j,m} I(x \in R_{j,m}) \quad (3.16)$$

Dimana

$F_{m-1}(x)$  = prediksi inisialisasi

$V$  = *learning rate*

$\sum_{j=1}^{J_m} \gamma_{j,m} I(x \in R_{j,m})$  = total nilai *gamma* terbaik dari seluruh prediksi *tree* yang dijumlahkan ( $y_{j,m}$ )

Pada persamaan 3.16, mengkombinasikan prediksi dengan menjumlahkan semua prediksi *tree* untuk menghasilkan prediksi baru yang lebih kuat, persamaan

ini akan diterapkan untuk tiap sampel. Implementasi dari proses ini disajikan sebagai berikut

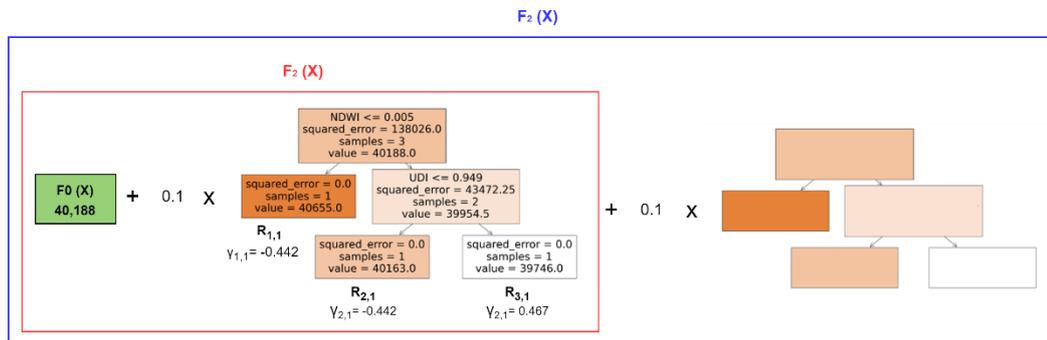
```
def predict(self,models,y,X):
    pred_0 = np.array([self.y_mean] * len(X))
    pred = pred_0.reshape(len(pred_0),1)
    for i in range(len(models)):
        temp = (models[i].predict(X)).reshape(len(X),1)
        pred -= self.lr * temp
    return pred

def train(self, X, y):
    # kode sebelumnya
    for epoch in range(self.num_iter):
        # kode sebelumnya
        r = (base.predict(X)).reshape(len(X),1)
        pred -= self.lr * r
        models.append(base)
    return models, losses, pred_0
```

Gambar 3.16 Implementasi pembaharuan (penggabungan) prediksi

Gambar 3.16 ialah kode implementasi untuk pembaharuan prediksi, fungsi *predict* bertujuan untuk melakukan prediksi untuk tiap iterasi, selanjutnya fungsi ini akan dipanggil pada fungsi *train* sebagai fungsi utama pelatihan model *Gradient Tree Boosting*. Hasil dari *predict* tiap iterasi akan dikali dengan *learning rate*. Perkalian dengan *learning rate* ini bertujuan untuk menentukan seberapa besar perubahan yang akan diterapkan pada hasil prediksi. Semakin besar *learning rate*, maka semakin besar pula perubahan yang terjadi tiap iterasinya. Pemilihan *learning rate* ini tidak ada yang secara eksplisit mewajibkan beraa nilainya, namun, dari beberapa literasi menggunakan nilai 0.1.

Hasil akhirnya ialah mengkombinasikan semua hasil prediksi yang telah dikali *learning rate* pada tiap iterasi sehingga akan medapat hasil prediksi yang kompleks untuk suhu permukaan (LST). Penggabungan prediksi pada Gambar 3.16 dapat divisualisasikan sebagai berikut



Gambar 3.17 Visualisasi kombinasi prediksi

Gambar 3.18 menunjukkan bagaimana model mengkombinasikan seluruh prediksi dari setiap iterasi sehingga menghasilkan prediksi yang kompleks. Proses ini akan berlanjut secara iteratif sampai jumlah *tree* tertentu atau perbaikan yang dilakukan sudah tidak signifikan lagi sehingga kombinasi ini akan menghasilkan prediksi yang kompleks dengan akurasi yang bagus.

Secara keseluruhan, implementasi dari algoritma *gradient tree boosting* disajikan sebagai berikut

```
def train(self, X, y):
    models = []
    losses = []
    self.y_mean = np.mean(y)
    pred_0 = np.array([np.mean(y)] * len(y))
    pred = pred_0.reshape(len(pred_0),1)

    for epoch in range(self.num_iter):
        loss = self.__calculate_loss(y, pred)
        losses.append(loss)
        residuals = self.__calculate_residual(y, pred)
        base = self.__create_tree(X, residuals)
        r = (base.predict(X)).reshape(len(X),1)
        pred -= self.lr * r
        models.append(base)
    return models, losses, pred_0
```

Gambar 3.18 Implementasi algoritma *gradient tree boosting*

Gambar 3.17 ialah kode utama untuk mengimplementasikan algoritma gradient boosting pada data latih seperti yang telah ditampilkan pada gambar alur proses 3.11.

### 3.5.2.5. Skenario Pengujian

Terdapat beberapa skenario dalam pelatihan dan pengujian model ini, yang mana mencakup rasio data latih-uji dan *tuning* parameter (*learning rate* dan *loss function*) yang sebelumnya telah dibahas dalam Bab II. Rasio yang dipakai antara lain

- 70% (data latih) : 30% (data uji)
- 80% (data latih) : 20% (data uji)
- 90% (data latih) : 10% (data uji).

Pemilihan rasio ini didasarkan pada beberapa percobaan yang telah dilakukan. Dimana hasil yang didapat ialah akurasi yang dihasilkan pada percobaan dengan rasio diatas atau dibawah dari ketiga rasio ini lebih buruk dibandingkan ketiganya. *Learning rate* yang dipakai dalam skenario percobaan diantaranya 0.01, 0.05, dan 0.1. Sedang untuk parameter *loss function* yang digunakan ialah *squared error*, *absolute error* dan *huber*. Detil seluruh kombinasi skenario pengujian dituliskan sebagai berikut

Tabel 3.6 Detil skenario pengujian

Skenario	Rasio	Learning Rate	Loss Function
1	70	0,01	<i>squared error</i>
2		0,01	<i>absolute error</i>
3		0,01	<i>huber</i>
4		0,05	<i>squared error</i>
5		0,05	<i>absolute error</i>
6		0,05	<i>huber</i>

7		0,1	<i>squared error</i>
8		0,1	<i>absolute error</i>
9		0,1	<i>huber</i>
1	80	0,01	<i>squared error</i>
2		0,01	<i>absolute error</i>
3		0,01	<i>huber</i>
4		0,05	<i>squared error</i>
5		0,05	<i>absolute error</i>
6		0,05	<i>huber</i>
7		0,1	<i>squared error</i>
8		0,1	<i>absolute error</i>
9		0,1	<i>huber</i>
1	90	0,01	<i>squared error</i>
2		0,01	<i>absolute error</i>
3		0,01	<i>huber</i>
4		0,05	<i>squared error</i>
5		0,05	<i>absolute error</i>
6		0,05	<i>huber</i>
7		0,1	<i>squared error</i>
8		0,1	<i>absolute error</i>
9		0,1	<i>huber</i>

Tabel 3.10 menunjukkan detail skenario pengujian yang akan dilakukan sehingga mendapatkan hasil kombinasi mana yang terbaik berdasarkan metrik evaluasi.

Terkait penentuan nilai *learning rate*, tidak ada penelitian terkait yang membahas secara pasti mengenai nilai berapa saja yang bisa dipakai, sehingga penulis melakukan percobaan dengan beberapa *learning rate* diatas.

### 3.5.3. Prediksi

Model yang sudah dibangun menggunakan algoritma GTB sebelumnya akan diuji dengan data uji untuk melakukan prediksi. Hasil prediksi ini berupa regresi yang akan ditampilkan dalam bentuk diagram nilai metrik performa  $r^2$  dan MSE.

### **3.6. Analisis**

Analisis adalah tahap menganalisa performa dari model yang telah melakukan prediksi sebelumnya. Analisa model dilakukan dengan menguji bagaimana model bekerja dan seberapa baik model jika diukur dengan metrik evaluasi.

#### **3.6.1. Evaluasi Performa Model**

Seberapa baik prediksi dari model dapat diukur dengan metrik evaluasi. Seperti yang sudah dijelaskan sebelumnya, model prediksi dalam penelitian ini akan dievaluasi dengan metrik R<sup>2</sup> dan MSE.

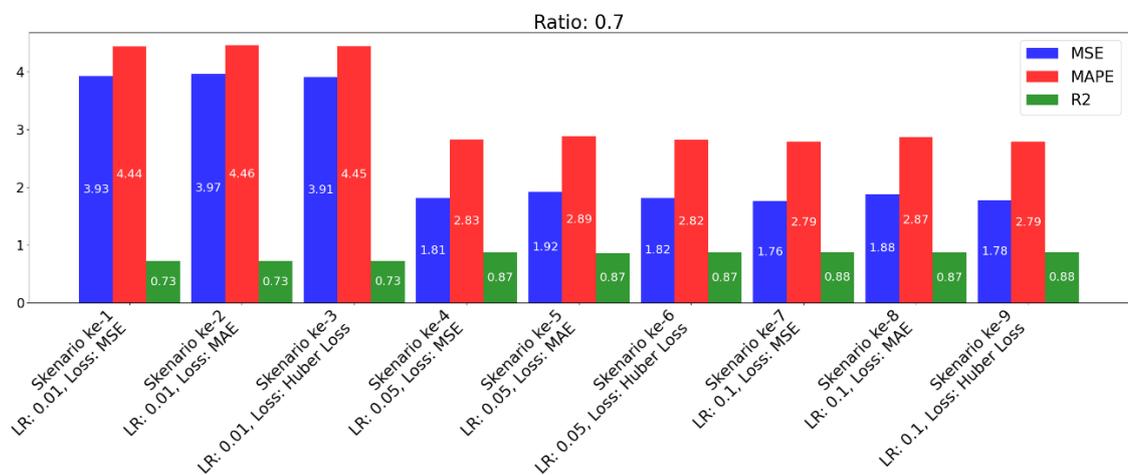
Metrik R<sup>2</sup> berfungsi untuk mengukur seberapa baik model memahami variabilitas dari data, sedang fungsi dari metrik MSE dan MAPE ialah untuk mengukur seberapa tepat model dalam melakukan prediksi dibandingkan data aktual

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1. Uji Coba

Berdasarkan penjelasan sebelumnya, pengujian dilakukan dengan beberapa skenario seperti yang sempat di singgung dalam Bab III pada tabel 3.11. Pada skenario ini model akan dilatih dengan berbagai macam kombinasi antara rasio data, *learning rate* dan *loss function*. Kombinasi ini nantinya akan menghasilkan nilai metrik performa yang bermacam-macam. Tujuannya untuk mengetahui kombinasi mana yang menghasilkan model dengan nilai metrik performa terbaik untuk memprediksi UHI. Secara keseluruhan, hasil dari ujicoba disajikan dalam grafik sebagai berikut



Gambar 4.1 Hasil pengujian pada rasio 70%

Dalam serangkaian pengujian dengan rasio 70, didapat hasil sebagai berikut. Skenario pertama dengan *learning rate* rendah (0.01) dan *loss function squared error* menghasilkan MSE sebesar 3.93 dan MAPE sebesar 4.44, dengan R2 sebesar 0.73. Hasil ini menunjukkan bahwa pada *learning rate* yang lebih lambat, model

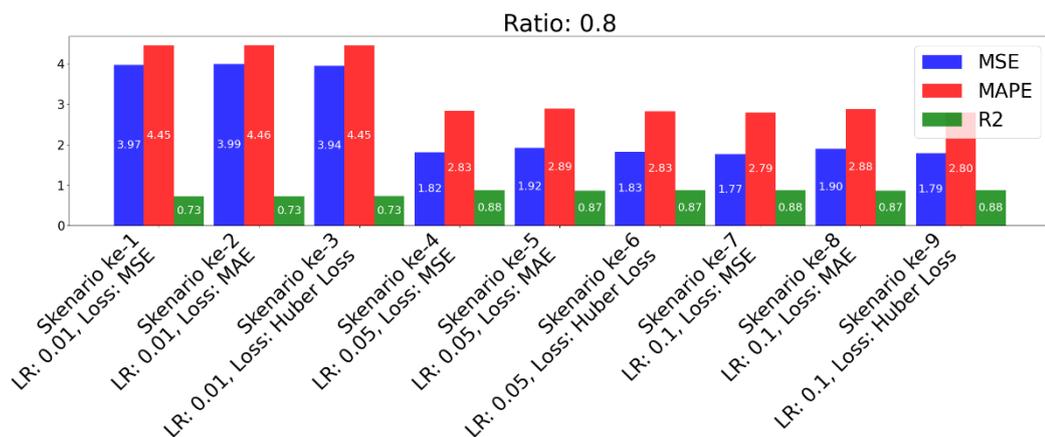
mengalami kesulitan dalam meminimalkan kesalahan prediksi, yang tercermin dari nilai MSE yang tinggi. Skenario kedua menggunakan *loss function absolute error* pada *learning rate* yang sama (0.01), dengan MSE sebesar 3.97, MAPE sebesar 4.46, dan R2 sebesar 0.73. Meskipun R2 tetap, kesalahan prediksi pada *loss function absolute error* cenderung lebih tinggi. Penggunaan *loss function Huber* pada skenario ketiga dengan *learning rate* 0.01 menghasilkan MSE sebesar 3.91, MAPE sebesar 4.45, dengan R2 yang tetap pada 0.73, menunjukkan efek serupa dengan *loss function squared error*.

Ketika *learning rate* ditingkatkan menjadi 0.05 dengan *loss function squared error* pada skenario keempat, terjadi penurunan signifikan dalam MSE menjadi 1.81, MAPE menjadi 2.83, dan peningkatan R2 menjadi 0.87. Hal ini mengindikasikan bahwa peningkatan *learning rate* membantu model dalam menyesuaikan diri dengan data latih dengan lebih baik. Skenario kelima dengan penggunaan *loss function absolute error* pada *learning rate* yang sama (0.05) menghasilkan MSE sebesar 1.92, MAPE sebesar 2.89, dengan R2 sebesar 0.87, menunjukkan performa yang sedikit lebih rendah dibandingkan dengan *loss function squared error*. Skenario keenam dengan *loss function Huber* pada *learning rate* 0.05 memberikan hasil yang mirip dengan *loss function squared error* (MSE = 1.82, MAPE = 2.82, R2 = 0.87).

Skenario terbaik ialah saat skenario ketujuh yang menggunakan *learning rate* 0.1 dengan *loss function squared error*, di mana model menghasilkan MSE sebesar 1.76, MAPE sebesar 2.79, dan R2 sebesar 0.88. Peningkatan *learning rate* pada tingkat yang lebih tinggi memberikan hasil yang optimal dalam mengurangi

kesalahan prediksi dan meningkatkan keakuratan model. Meskipun demikian, penggunaan *loss function absolute error* pada *learning rate* 0.1 dalam skenario kedelapan menghasilkan MSE sebesar 1.88, MAPE sebesar 2.87, dengan R2 sebesar 0.87, menunjukkan performa sedikit lebih rendah dibandingkan dengan *loss function squared error*. Penggunaan *loss function* Huber pada skenario kesembilan pada *learning rate* yang sama (0.1) memberikan hasil yang kompetitif dengan *loss function squared error* (MSE = 1.78, MAPE = 2.79, R2 = 0.88).

Hasil pengujian pada kelompok rasio data latih 70% menunjukkan bahwa *learning rate* 0.1 dengan *loss function squared error* memberikan performa terbaik dalam mengoptimalkan model GTB.



Gambar 4.2 Hasil skenario pada rasio 80%

Gambar 4.2 menampilkan hasil dari pengujian skenario dengan rasio 80%. Pada skenario pertama dengan *learning rate* rendah (0.01), hasil eksperimen menunjukkan bahwa kedua jenis *loss function*, baik *squared error* maupun *absolute error*, menghasilkan tingkat kesalahan prediksi yang tinggi, dengan nilai MSE sekitar 3.97-3.99 dan R2 sekitar 0.73. MAPE yang tercatat juga cukup tinggi, sekitar 4.45. Sedangkan untuk *loss function* Huber pada skenario ketiga, terdapat

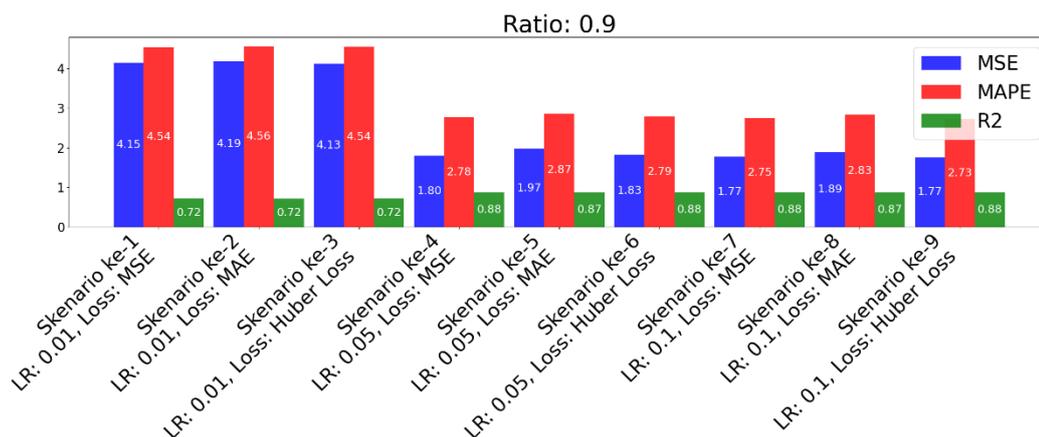
sedikit peningkatan dengan nilai MSE sekitar 3.94 dan MAPE sekitar 4.45, tetapi R2 tetap pada 0.73. Temuan ini mengindikasikan bahwa model mengalami kesulitan menyesuaikan diri dengan pola data yang kompleks pada *learning rate* yang rendah.

Ketika *learning rate* ditingkatkan menjadi 0.05 pada skenario berikutnya, terlihat peningkatan yang signifikan dalam kinerja model. Penggunaan *loss function* (skenario keempat) dan Huber (skenario keenam) pada *learning rate* ini menghasilkan penurunan MSE menjadi sekitar 1.82-1.83, dengan MAPE sekitar 2.83 dan R2 sekitar 0.87, menunjukkan adaptasi yang lebih baik terhadap data latih. Skenario kelima dengan *loss function absolute error* pada *learning rate* yang sama menghasilkan MSE sedikit lebih tinggi, yaitu sekitar 1.92, dengan MAPE sekitar 2.89, tetapi masih memberikan R2 yang kompetitif sekitar 0.87.

Skenario keenam menggunakan *learning rate* 0.05 dengan *loss function* Huber, menghasilkan MSE sekitar 1.83, MAPE sekitar 2.83, dan R2 sekitar 0.87. Meskipun demikian, performa ini sedikit lebih rendah daripada skenario keempat dengan *learning rate* 0,05 dan *loss function squared error* yang menghasilkan MSE 1,82 MAPE 2,83 R2 0,88.

Hasil terbaik diperoleh pada skenario ketujuh dengan *learning rate* 0.1 dan *loss function squared error*. Model pada skenario ini mampu menghasilkan MSE terendah sekitar 1.77, dengan MAPE terendah sekitar 2.79 dan R2 tertinggi sekitar 0.88. Skenario kedelapan (*absolute error*) dan kesembilan (Huber) dengan *learning rate* yang sama juga menunjukkan performa yang baik dengan MSE sekitar 1.90-1.79, dan MAPE sekitar 2.80-2.88, serta R2 sekitar 0.87-0.88. Peningkatan

signifikan pada *learning rate* membantu model dalam mengurangi kesalahan prediksi dan meningkatkan akurasi prediksi. Penyesuaian parameter seperti *learning rate* dan *loss function* sangat penting dalam memperbaiki performa prediksi model terhadap data latih.



Gambar 4.3 Hasil skenario pada rasio 90%

Gambar 4.3 menunjukkan hasil dari beberapa skenario pengujian yang menggunakan rasio 90%. Hasil analisis didapat, pada skenario dengan *learning rate* rendah (0.01), hasil eksperimen menunjukkan bahwa kedua jenis *loss function*, yaitu *squared error* (skenario pertama), *absolute error* (skenario kedua), dan huber (skenario ketiga), menghasilkan tingkat kesalahan prediksi yang tinggi dengan nilai MSE sekitar 4.13-4.19 dan R2 sekitar 0.72. MAPE untuk skenario 1 adalah sekitar 4.54%, skenario 2 sekitar 4.56%, dan skenario 3 sekitar 4.54%. Temuan ini menunjukkan bahwa model menghadapi kesulitan dalam menyesuaikan diri dengan pola data yang kompleks pada *learning rate* yang rendah.

Ketika *learning rate* ditingkatkan menjadi 0.05, terlihat peningkatan yang signifikan dalam kinerja model. Penggunaan *loss function squared error* (skenario keempat) dan huber (skenario keenam) pada *learning rate* ini menghasilkan

penurunan MSE menjadi sekitar 1.80-1.83 dengan R2 sekitar 0.88, menunjukkan adaptasi yang lebih baik terhadap data latih. Penggunaan *loss function absolute error* (skenario kelima) pada *learning rate* yang sama menghasilkan MSE sedikit lebih tinggi sekitar 1.97, tetapi masih memberikan R2 yang kompetitif sekitar 0.87. MAPE untuk skenario 4 adalah sekitar 2.78%, skenario 5 sekitar 2.87%, dan skenario 6 sekitar 2.79%.

Hasil terbaik diperoleh pada skenario ketujuh dengan *learning rate* 0.1 dan *loss function squared error*. Model pada skenario ini mampu menghasilkan MSE terendah sekitar 1.77 dengan R2 tertinggi sekitar 0.88. Peningkatan signifikan pada *learning rate* membantu model dalam mengurangi kesalahan prediksi dan meningkatkan akurasi prediksi. Nilai MSE 1.77 menunjukkan bahwa model dapat memprediksi target dengan baik dan akurat. MAPE untuk skenario 7 adalah sekitar 2.75%. Hasil ini lebih baik dibandingkan pada penggunaan *loss function absolute error* dan Hbuer pada skenario kedelapan dan kesembilan dengan MSE dan MAPE yang lebih tinggi dalam kisaran 2,80-2,88, namun, kedua *loss function* ini masih memiliki nilai R2 yang kurang lebih mirip.

Dari hasil pengujian pada rasio 90%, dapat disimpulkan bahwa penggunaan *learning rate* 0.1 dengan *loss function squared error* memberikan performa terbaik dalam memprediksi. Hal ini menunjukkan pentingnya *learning rate* yang optimal dalam menghasilkan prediksi yang lebih akurat dan konsisten. Secara keseluruhan, hasil pengujian pada setiap scenario disajikan sebagai berikut dalam bentuk table

Tabel 4.1 Hasil skenario pengujian

Skenario	Rasio	Learning Rate	Loss Function	MSE	MAPE	R <sup>2</sup>
1	70	0,01	<i>squared error</i>	3,93	4,44	0,73
2		0,01	<i>absolute error</i>	3,97	4,46	0,73

Skenario	Rasio	Learning Rate	Loss Function	MSE	MAPE	R <sup>2</sup>	
3		0,01	<i>huber</i>	3,91	4,45	0,73	
4		0,05	<i>squared error</i>	1,81	2,83	0,87	
5		0,05	<i>absolute error</i>	1,92	2,89	0,87	
6		0,05	<i>huber</i>	1,82	2,82	0,87	
7		0,1	<i>squared error</i>	1,76	2,79	0,88	
8		0,1	<i>absolute error</i>	1,88	2,87	0,87	
9		0,1	<i>huber</i>	1,78	2,79	0,88	
1		80	0,01	<i>squared error</i>	3,97	4,45	0,73
2			0,01	<i>absolute error</i>	3,99	4,46	0,73
3	0,01		<i>huber</i>	3,94	4,45	0,73	
4	0,05		<i>squared error</i>	1,82	2,83	0,88	
5	0,05		<i>absolute error</i>	1,92	2,89	0,87	
6	0,05		<i>huber</i>	1,83	2,83	0,87	
7	0,1		<i>squared error</i>	1,77	2,79	0,88	
8	0,1		<i>absolute error</i>	1,90	2,88	0,87	
9	0,1		<i>huber</i>	1,79	2,80	0,88	
1	90	0,01	<i>squared error</i>	4,15	4,54	0,72	
2		0,01	<i>absolute error</i>	4,19	4,56	0,72	
3		0,01	<i>huber</i>	4,13	4,54	0,72	
4		0,05	<i>squared error</i>	1,80	2,78	0,88	
5		0,05	<i>absolute error</i>	1,97	2,87	0,87	
6		0,05	<i>huber</i>	1,83	2,79	0,88	
7		0,1	<i>squared error</i>	1,77	2,75	0,88	
8		0,1	<i>absolute error</i>	1,89	2,83	0,87	
9		0,1	<i>huber</i>	1,77	2,73	0,88	

Berdasarkan keseluruhan hasil skenario pengujian yang tertulis dalam tabel 4.1, skenario terbaik ialah kombinasi rasio 70%, *learning rate* 0.1 dan *loss function squared error*, dimana skenario ini mendapatkan nilai MSE 1,76 MAPE 2,79 dan R<sup>2</sup> 0,88.

Nilai MSE 0.88 menunjukkan bahwa model terbilang sangat baik dalam memprediksi target yang dibuktikan dengan kesalahan prediksi yang minim. Sama halnya dengan MSE. Nilai MAPE 2,79 menunjukkan bahwa kinerja model dalam prediksi ini tergolong sangat baik dengan persentase kesalahan hanya 2,79% . Nilai

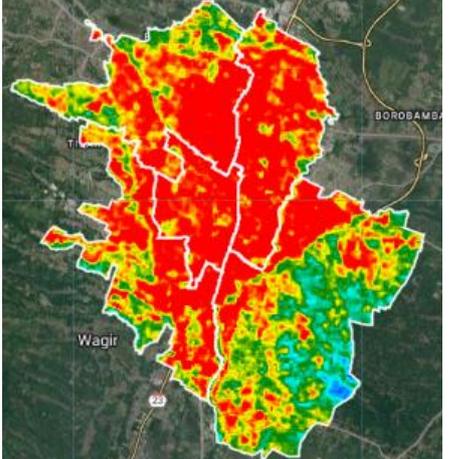
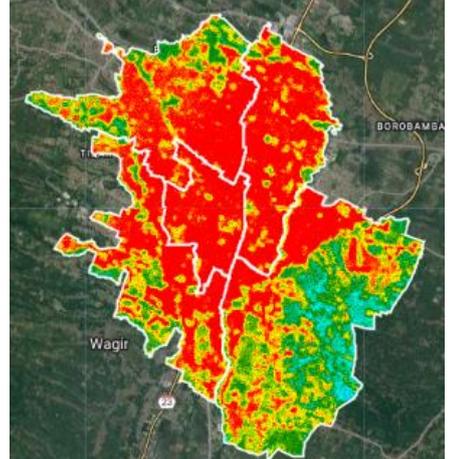
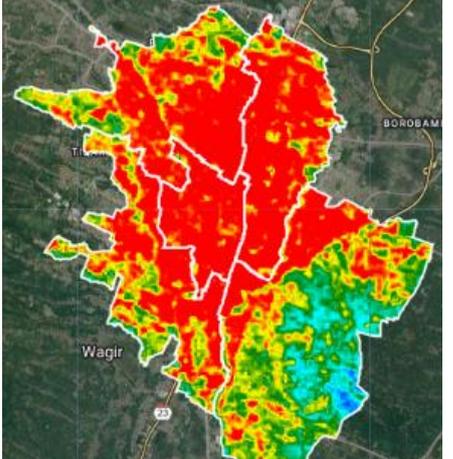
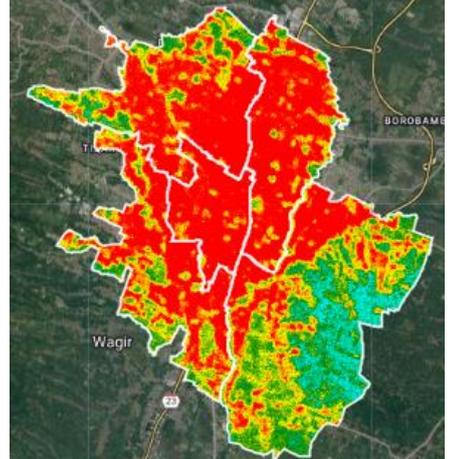
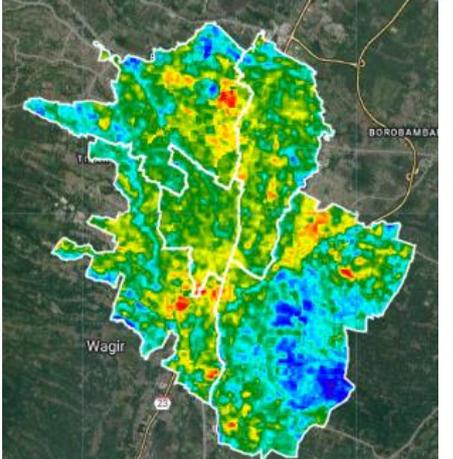
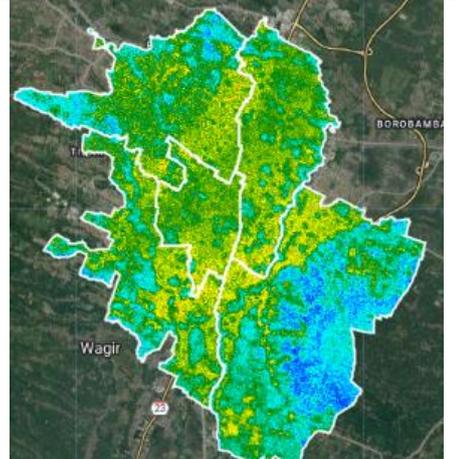
R2 0.88 menunjukkan bahwa model dapat memahami hubungan antar variabel dengan targetnya, dengan kata lain model cocok dengan data yang dipakai.

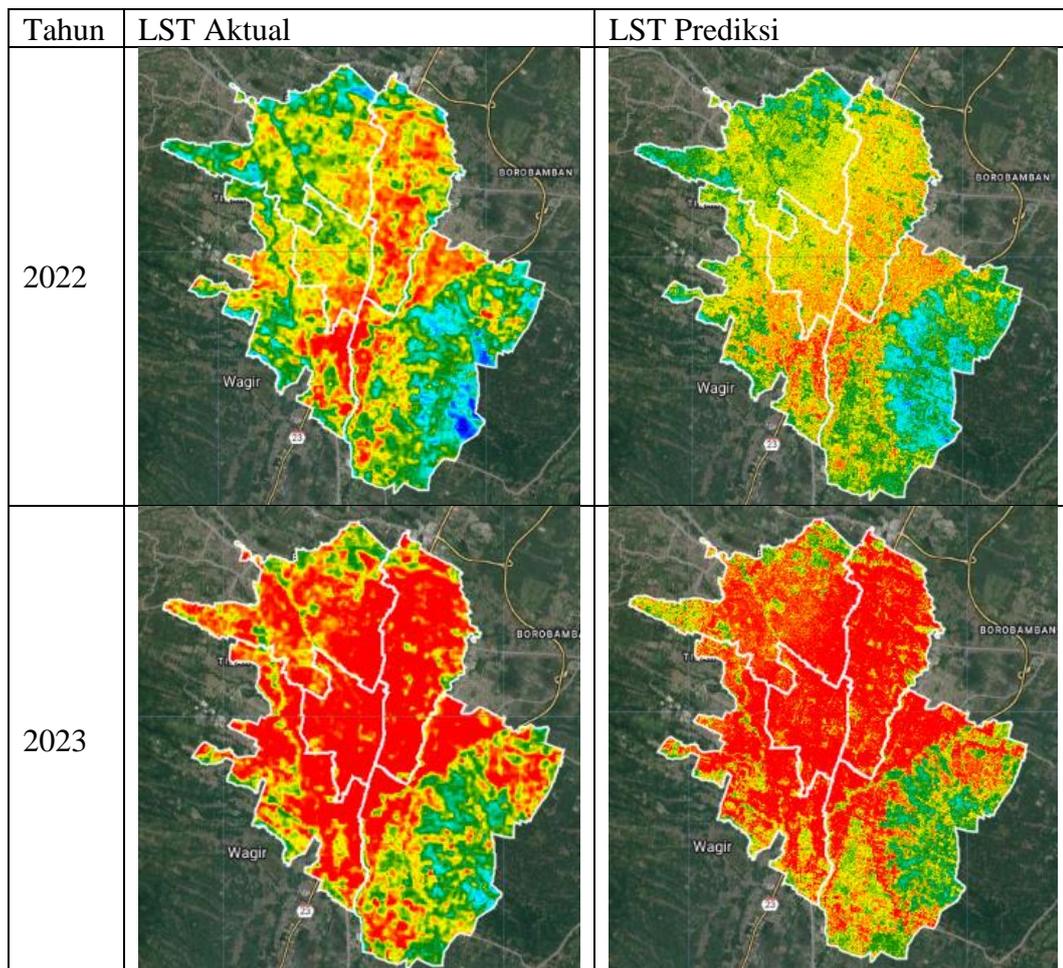
Hal yang menyebabkan mengapa skenario dengan kombinasi ini menjadi yang terbaik ialah rasio datanya, rasio data untuk skenario ini ialah 70%, dimana jumlah ini memberikan keseimbangan yang baik antara data yang digunakan untuk melatih model dan data yang digunakan untuk menguji model. Rasio ini cukup besar untuk melatih model dengan baik tanpa terlalu banyak mengorbankan data uji. *Learning rate* yang lebih tinggi membuat model lebih responsif terhadap perubahan dalam data training. Data mungkin memiliki variasi yang signifikan atau pola yang kompleks, sehingga model perlu dapat menyesuaikan diri dengan cepat untuk memahami pola-pola ini. *Learning rate* 0.1 memberikan fleksibilitas yang cukup untuk menangani dinamika ini.

#### **4.2. Analisis Perbandingan Pemetaan LST Pada Data Aktual dan Prediksi**

Hasil penelitian ini juga meliputi pemetaan prediksi GTB pada wilayah Kota Malang. Pemetaan ini dilakukan untuk mendapat informasi area mana saja yang diprediksi akan mengalami fenomena UHI. Prediksi dilakukan untuk tiap tahun dari 2019-2023. Secara keseluruhan, Hasil pemetaan LST aktual terhadap hasil prediksi model disajikan dalam tabel 4.2 Berikut ini

Tabel 4.2 Perbandingan LST aktual dan hasil rediksi model GTB

Tahun	LST Aktual	LST Prediksi
2019		
2020		
2021		



Dari hasil pemetaan untuk tiap tahun, didapatk analisis sebagai berikut: Pada tahun 2019, prediksi LST menunjukkan kenaikan suhu hampir di seluruh wilayah kecamatan di Kota Malang. Di Kecamatan Lowokwaru, diprediksi akan terjadi kenaikan suhu dan perluasan fenomena UHI di pesisir batas bagian atas, ditandai dengan perubahan warna dari kuning menjadi merah. Di Kecamatan Blimbing, fenomena UHI diprediksi akan meluas di bagian pusat, dengan perubahan suhu dari hijau menjadi kuning, bahkan di beberapa titik dari kuning menjadi hijau. Kecamatan Klojen diprediksi mengalami perluasan UHI di pusat dan sekitar batas wilayahnya. Kecamatan Sukun juga akan mengalami perluasan wilayah UHI,

meskipun lebih sedikit dibandingkan kecamatan lainnya, dengan kenaikan suhu dari hijau menjadi kuning terutama di bagian batas sebelah kanan. Kecamatan Kedungkandang diprediksi mengalami kenaikan suhu yang signifikan di bagian pusat hingga ke pojok kiri batas wilayah, dengan perubahan warna dari biru muda menjadi hijau dan kuning. Fenomena UHI di kecamatan ini juga diprediksi meluas, terutama di wilayah yang dekat dengan pusat Kota Malang dan batas bawah kecamatan.

Dibandingkan dengan LST aktual pada tahun 2020, hasil prediksi tahun 2019 menunjukkan kemiripan yang signifikan, terutama pada meluasnya fenomena UHI di lima kecamatan tersebut yang dimulai dari pusat kota dan kenaikan suhu di beberapa titik.

Pada tahun 2020, prediksi menunjukkan Kecamatan Lowokwaru akan mengalami kenaikan suhu hampir di seluruh wilayahnya, dengan fenomena UHI yang signifikan di bagian batas atas. Kecamatan Blimbing diprediksi mengalami kenaikan suhu di hampir seluruh wilayah, dengan kenaikan signifikan di batas atas sekitar pusat wilayah. Kecamatan Klojen sebagai pusat kota diprediksi mengalami perluasan UHI, meskipun beberapa titik akan mengalami penurunan suhu dari merah menjadi kuning. Kecamatan Sukun diprediksi mengalami perluasan UHI dengan kenaikan suhu yang signifikan di batas wilayah sebelah kanan, dari kuning menjadi merah. Kecamatan Kedungkandang juga diprediksi mengalami kenaikan suhu dan perluasan UHI hampir di seluruh wilayahnya, terutama di bagian tengah hingga batas kiri.

Perbandingan prediksi tahun 2020 dengan data aktual tahun 2021 menunjukkan hasil prediksi yang kurang akurat, dengan hampir seluruh wilayah mengalami penurunan suhu. Hal ini disebabkan oleh anomali yang terjadi akibat lockdown pada pandemi COVID-19 yang menyebabkan penurunan aktivitas urban dan berdampak pada penurunan suhu di Kota Malang. Perubahan data yang ekstrem ini menjadi alasan utama ketidakcocokan prediksi tahun 2020 dengan data aktual tahun 2021.

Pada tahun 2021, akibat dari pandemi COVID-19, terjadi penurunan suhu yang signifikan hampir di seluruh kecamatan di Kota Malang. Rata-rata penurunan ini terjadi dari merah ke kuning dan kuning ke hijau. Perubahan signifikan terlihat di Kecamatan Klojen, Lowokwaru, dan Sukun. Di Kecamatan Lowokwaru, penurunan suhu terutama terjadi di bagian tengah. Kecamatan Klojen mengalami penurunan suhu terutama di bagian atas batas wilayah. Di Kecamatan Sukun, penurunan suhu terlihat di wilayah sekitar batas kiri.

Perbandingan hasil prediksi tahun 2021 dengan data aktual tahun 2022 menunjukkan perbedaan, di mana prediksi menunjukkan penurunan suhu dan berkurangnya fenomena UHI, sementara data aktual menunjukkan kenaikan suhu dan kembalinya fenomena UHI. Perbedaan ini kemungkinan disebabkan oleh pelanggaran lockdown pada tahun 2022 yang meningkatkan aktivitas urban kembali.

Pada tahun 2022, data aktual menunjukkan kenaikan suhu kembali seperti pada tahun 2019, dengan beberapa wilayah pusat Kota Malang di Kecamatan Klojen, Sukun, Kedungkandang, dan Blimbing mengalami fenomena UHI.

Kenaikan ini kemungkinan disebabkan oleh penerapan new normal yang meningkatkan aktivitas urban di Kota Malang.

Prediksi menunjukkan beberapa kecamatan akan mengalami penurunan suhu yang tidak terlalu signifikan, terutama di wilayah Klojen dan Sukun bagian bawah, serta di batas wilayah Klojen dan Sukun. Namun, fenomena UHI tetap terjadi dengan intensitas yang lebih rendah. Kecamatan Lowokwaru diprediksi mengalami kenaikan suhu di batas atas dari biru muda ke hijau, serta penurunan dari merah ke kuning di batas wilayah dengan Blimbing. Kecamatan Blimbing diprediksi mengalami kenaikan suhu di hampir seluruh wilayahnya, dengan penurunan intensitas UHI dari merah menjadi oranye.

Perbandingan data aktual tahun 2023 dengan prediksi 2022 menunjukkan perbedaan yang cukup besar, di mana prediksi menunjukkan penurunan fenomena UHI sementara data aktual menunjukkan peningkatan UHI di hampir seluruh kecamatan. Perbedaan ini kemungkinan disebabkan oleh pembatasan sosial pada sebagian bulan tahun 2022 yang mempengaruhi model prediksi.

Pada tahun 2023, hasil pemetaan LST aktual menunjukkan peningkatan suhu yang signifikan di hampir seluruh kecamatan di Kota Malang dibandingkan tahun 2022. Kecamatan Klojen dan Blimbing mengalami fenomena UHI dengan intensitas dan luas area yang masif, diikuti oleh Sukun, Lowokwaru, dan Kedungkandang. Kecamatan Klojen sebagai pusat kota mengalami UHI di hampir seluruh wilayahnya dengan intensitas tinggi. Kecamatan Blimbing juga mengalami UHI dengan intensitas tinggi di hampir seluruh wilayahnya. Ketiga kecamatan

lainnya mengalami fenomena serupa dengan intensitas tinggi namun tidak semasih dua kecamatan tersebut.

Prediksi untuk tahun 2023 menunjukkan penurunan intensitas UHI di semua wilayah kecamatan di Kota Malang, namun dengan perluasan wilayah yang mengalami kenaikan suhu. Hal ini kemungkinan disebabkan oleh persebaran penduduk dan perluasan penggunaan lahan di sekitar batas wilayah terluar setiap kecamatan.

### 4.3. Integrasi Islam

#### 4.3.1. Muamalah Mu'Allah

Dalam Al-Qur'an, Allah SWT berfirman:

وَلَا تُفْسِدُوا فِي الْأَرْضِ بَعْدَ إِصْلَاحِهَا وَادْعُوهُ خَوْفًا وَطَمَعًا إِنَّ رَحْمَتَ اللَّهِ قَرِيبٌ مِّنَ الْمُحْسِنِينَ ٥٦ )  
 الاعراف/7: 56)

*“Janganlah kamu berbuat kerusakan di bumi setelah diatur dengan baik. Berdoalah kepada-Nya dengan rasa takut dan penuh harap. Sesungguhnya rahmat Allah sangat dekat dengan orang-orang yang berbuat baik.” (Al-A'raf/7:56)*

Dalam tafsir Al-Misbah, ayat ini ditafsirkan sebagai “Jangan kalian membuat kerusakan di muka bumi yang telah dibuat baik dengan menebar kemaksiatan, kezaliman dan permusuhan. Berdoalah kepada-Nya dengan rasa takut akan siksa-Nya dan berharap pahala-Nya. Kasih sayang Allah sangat dekat kepada setiap orang yang berbuat baik, dan pasti terlaksana”.

Sedang menurut tafsir Ibnu Katsir, ayat ini ditafsirkan sebagai, berdoalah kepada Tuhan kalian dengan berendah diri dan suara yang lembut. Sesungguhnya Allah tidakmenyukai orang-orang yang melampaui batas. Dan janganlah kalian

membuat kerusakan di muka bumi, sesudah (Allah) memperbaikinya dan berdoalah kepada-Nya dengan rasa takut (tidak akan diterima) dan harapan (akan dikabulkan). Sesungguhnya rahmat Allah amat dekat kepada orang-orang yang berbuat baik. Allah subhanahu wa ta'ala memberikan petunjuk kepada hamba-hamba-Nya agar mereka berdoa memohon kepada-Nya untuk kebaikan urusan dunia dan akhirat mereka. Untuk itu Allah subhanahu wa ta'ala berfirman: Berdoalah kepada Tuhan kalian dengan berendah diri dan suara yang lembut. (Al-A'raf: 54) Menurut suatu pendapat, makna yang dimaksud ialah mengucapkan doa dengan perasaan yang rendah diri, penuh harap, dan dengan suara yang lemah lembut.

Dari kedua tafsiran di atas, dijelaskan bahwa menjaga dan merawat lingkungan adalah salah satu dari perintah Allah yang sangat penting. Allah telah menciptakan alam dengan segala kelengkapannya untuk kebutuhan manusia, dan manusia dilarang untuk merusaknya. Melalui kebaikan dan perbuatan yang baik, termasuk menjaga lingkungan, kita dapat mendekatkan diri kepada kasih sayang Allah dan memperoleh rahmat-Nya. Fenomena UHI adalah fenomena yang memberikan dampak buruk kepada kehidupan manusia dan lingkungan, efeknya dapat mengganggu kesehatan dan menyebabkan penyakit dalam kasus ekstrim dapat menyebabkan kematian. Karnanya, dalam merealisasikan perintah Allah untuk memberikan manfaat kepada manusia, dan menjaga kelestarian lingkungan dan ekosistem, maka penelitian ini dilakukan untuk memprediksi fenomena ini supaya dapat menjadi informasi terkait kebijakan lanjutan untuk perencanaan pembangunan dan kawasan hijau supaya kehidupan manusia dan ekosistem di wilayah terdampak dapat tetap terjaga dan hidup dengan nyaman. Hal ini sejalan

dengan misi umat islam untuk menjadi manusia yang bermanfaat kepada sesamanya sebagaimana yang disebutkan dalam hadist Rasulullah SAW yang berbunyi

*Rasulullah saw bersabda : “Sebaik-baik manusia adalah yang paling bermanfaat bagi manusia” (HR. Ahmad, ath-Thabrani, ad-Daruqutni. Hadits ini dihasankan oleh al-Albani di dalam Shahihul Jami’ no: 3289).*

Hadist tersebut menjelaskan bahwa nabi secara eksplisit mengajarkan umat islam untuk menjadi orang yang bermanfaat bagi sesamanya. Disamping itu, inisiatif untuk berkontribusi dalam menangani hal ini juga didasari hadist lain yang diriwayatkan oleh Abu Sa’id Al-Khudri radhiyallahu ‘anhu,

Dari Abu Sa’id Al-Khudri radhiyallahu ‘anhu, ia berkata, “Aku mendengar Rasulullah shallallahu ‘alaihi wa sallam bersabda, ‘Barangsiapa dari kalian melihat kemungkaran, ubahlah dengan tangannya. Jika tidak bisa, ubahlah dengan lisannya. Jika tidak bisa, ingkarilah dengan hatinya, dan itu merupakan selemah-lemahnya iman.’” (HR. Muslim) [HR. Muslim, no. 49]

Dari penjelasan diatas, penulis berkesimpulan bahwa penelitian dihukumi sebagai sunnah, Karna penelitian ini berkontribusi pada bagian dalam menangani masalah fenomena UHI yang berarti bermanfaat dan berkemungkinan menyelamatkan nyawa dimasa yang akan datang.

#### **4.3.2. Muamalah Mu’annas**

Seperti yang sudah dijelaskan dalam penjelasan terkait tafsir dari surat Al-A'raf ayat 56 sebelumnya, kesimpulan mengenai tafsir didapat Allah SWT memerintahkan untuk menjaga kelestarian lingkungan. Fenomena kenaikan suhu,

seperti yang terjadi dalam *Urban Heat Island* (UHI) di Kota Malang, memiliki dampak yang signifikan pada kesejahteraan manusia. Suhu yang berlebihan dapat menyebabkan berbagai masalah kesehatan, seperti dehidrasi, kelelahan panas, dan bahkan penyakit yang lebih serius seperti heatstroke. Selain itu, kelebihan panas juga dapat mengganggu aktivitas sehari-hari manusia, terutama bagi mereka yang bekerja di luar ruangan atau memiliki keterbatasan dalam mengakses perlindungan dari panas.

Untuk langkah kontribusi mencegah dampak dari fenomena ini, penelitian ini dilakukan untuk memprediksi fenomena UHI di Kota Malang supaya dapat menjadi informasi untuk pemerintah terkait kebijakan lanjutan yang lebih efektif untuk perencanaan pembangunan dan kawasan hijau, supaya dapat mencegah dampak buruk bagi kesehatan masyarakat di Kota Malang dan masyarakat dapat tetap melakukan kegiatan sehari-hari tanpa terganggu.

Berdasar pada penjelasan mengenai hadist dan penjelasan tafsir surat Al-A'raf ayat 56 yang dijelaskan dalam muamalah ma'Allah sebelumnya, penelitian ini dapat dihukumi sebagai sunnah untuk berkontribusi dalam upaya memberi manfaat kepada sesama manusia sejalan dengan hadist Nabi Muhammad SAW yang berbunyi,

*Rasulullah saw bersabda : “Sebaik-baik manusia adalah yang paling bermanfaat bagi manusia” (HR. Ahmad, ath-Thabrani, ad-Daruqutni. Hadits ini dihasankan oleh al-Albani di dalam Shahihul Jami’ no: 3289).*

#### **4.3.3. Muamalah Mu’alam**

Tidak hanya penting bagi kesejahteraan manusia, pemahaman yang lebih dalam tentang fenomena UHI juga untuk menjaga keseimbangan alam, ini sejalan

dengan penjelasan terkait tafsir dari surat Al-A'raf ayat 56 yang telah dijelaskan sebelumnya. Peningkatan suhu yang signifikan di daerah perkotaan tidak hanya memengaruhi manusia tetapi juga lingkungan sekitarnya. Suhu yang tinggi dapat memengaruhi ekosistem lokal dengan mempercepat penguapan air dari tanah dan vegetasi. Hal ini dapat menyebabkan penurunan kadar air tanah dan kekeringan yang berdampak pada pertanian, kehidupan hewan liar, dan keberlanjutan sumber daya alam.

Dengan demikian, penelitian ini dilakukan untuk memprediksi fenomena ini supaya dapat menjadi informasi lebih lanjut terkait kebijakan apa yang akan dilakukan untuk mengatasi masalah ini sehingga alam dapat tetap terjaga dan ekosistem didalamnya dapat tetap hidup dengan baik. Sama halnya dengan penjelasan dalam muamalah mu'annas sebelumnya, hukum dari penelitian ini terkait menjaga lingkungan dapat dikatakan sunnah, seperti halnya yang dijelaskan dalam hubungan dengan manusia karna penelitian ini berkontribusi dalam upaya pencegahan fenomena yang lebih buruk kedepannya.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1. Kesimpulan**

Pelatihan model dilakukan untuk menghasilkan model terbaik dalam memprediksi UHI di Kota Malang. Hasil pengujian seluruh skenario menunjukkan bahwa model terbaik didapat pada skenario ketujuh dengan kombinasi rasio 70%, *learning rate* 0,1 dan *loss function squared error* yang menghasilkan nilai MSE yang rendah hanya 1,76, MAPE yang juga rendah pada 2,79 dan R2 yang tinggi 0,88. Nilai MSE dan MAPE yang rendah menunjukkan bahwa model memiliki performa prediksi yang baik dengan ditandai kesalahan yang minim. Dan R2 yang tinggi menunjukkan bahwa model dapat memahami variasi dalam data dengan baik dan hubungan antar fitur dan target didalamnya.

Analisis hasil pemetaan prediksi menunjukkan akan terjadi peningkatan suhu hampir diseluruh wilayah Kota Malang yang mencakup wilayah lowokwaru, klojen, blimbing, kedungkandang dan sukun. Dimana fenomena UHI diprediksi akan terjadi di beberapa bagian dari wilayah kecamatan tersebut.

#### **5.2. Saran**

Berdasarkan hasil penelitian yang telah dilakukan mengenai prediksi *Urban Heat Island* (UHI) di Kota Malang, penulis memberikan beberapa saran yang diharapkan dapat bermanfaat bagi berbagai pihak, antara lain:

1. Penelitian selanjutnya dapat mempertimbangkan untuk menambah beberapa parameter dalam algoritma *Gradient Tree Boosting* guna memperoleh akurasi yang lebih tinggi.
2. Metode yang berbeda pada kasus serupa dapat dijadikan alternatif untuk melanjutkan eksplorasi dalam pembuatan model dengan akurasi yang lebih baik.
3. Disarankan untuk menggunakan data yang lebih beragam dan detail, seperti data satelit dengan resolusi tinggi, data iklim jangka panjang data penggunaan lahan yang lebih spesifik. Pendekatan ini diharapkan dapat meningkatkan akurasi model prediksi UHI.

## DAFTAR PUSTAKA

- Addas, A. (2023). Machine Learning Techniques to Map the Impact of *Urban Heat Island*: Investigating the City of Jeddah. *Land*, 12(6).  
<https://doi.org/10.3390/land12061159>
- Asante, D., Omar, T., Ganat, A., Oghenerurie, J., Tackie-otoo, B. N., & Yassir, M. (2021). Journal of Petroleum Science and Engineering Application of gradient boosting regression model for the evaluation of feature selection techniques in improving reservoir characterisation predictions. *Journal of Petroleum Science and Engineering*, July, 109244.  
<https://doi.org/10.1016/j.petrol.2021.109244>
- Balasundaram, S., & Prasad, S. C. (2020). Robust twin support vector regression based on Huber *loss function*. *Neural Computing and Applications*, 32(15), 11285–11309. <https://doi.org/10.1007/s00521-019-04625-8>
- Deekshith, P., & Singh, R. P. (2020). *Review on Advanced Machine Learning Model : Scikit-Learn Abstract : 3(4)*, 526–529.
- Equere, V., Mirzaei, P. A., & Riffat, S. (2020). Definition of a new morphological parameter to improve prediction of *urban heat island*. *Sustainable Cities and Society*, 56, 102021. <https://doi.org/10.1016/j.scs.2020.102021>
- Friedman, J. H. (2001). GREEDY FUNCTION APPROXIMATION: A GRADIENT BOOSTING MACHINE. In *The Annals of Statistics* (Vol. 29, Issue 5).
- Furuya, M. T. G., Furuya, D. E. G., de Oliveira, L. Y. D., da Silva, P. A., Cicerelli, R. E., Gonçalves, W. N., Junior, J. M., Osco, L. P., & Ramos, A. P. M. (2023). A machine learning approach for mapping surface *urban heat island* using environmental and socioeconomic variables: a case study in a medium-sized Brazilian city. *Environmental Earth Sciences*, 82(13).  
<https://doi.org/10.1007/s12665-023-11017-8>
- Gaur, A., Eichenbaum, M. K., & Simonovic, S. P. (2018). Analysis and modelling of surface *Urban Heat Island* in 20 Canadian cities under climate and land-cover change. *Journal of Environmental Management*, 206, 145–157.  
<https://doi.org/10.1016/j.jenvman.2017.10.002>
- Grover, A., & Singh, R. B. (2015). Analysis of *urban heat island* (Uhi) in relation to normalized difference vegetation index (ndvi): A comparative study of delhi and mumbai. *Environments - MDPI*, 2(2), 125–138.  
<https://doi.org/10.3390/environments2020125>

- Gumaei, A., Al-Rakhami, M., Al Rahhal, M. M., Albogamy, F. R. H., Al Maghayreh, E., & AlSalman, H. (2021). Prediction of COVID-19 confirmed cases using gradient boosting regression method. *Computers, Materials and Continua*, *66*(1), 315–329. <https://doi.org/10.32604/cmc.2020.012045>
- Hosny, A., Parmar, C., Quackenbush, J., Schwartz, L. H., & Aerts, H. J. W. L. (2018). Artificial intelligence in radiology. *Nature Reviews Cancer*, *18*(August). <https://doi.org/10.1038/s41568-018-0016-5>
- Jabbar, H. K., Hamoodi, M. N., & Al-Hameedawi, A. N. M. (2023). *Urban heat islands : a review of contributing factors , effects and data Urban heat islands : a review of contributing factors , effects and data*. *1129*(1), 012038–012038. <https://doi.org/10.1088/1755-1315/1129/1/012038>
- Liu, B. (2011). *Supervised Learning. In: Web Data Mining. Data-Centric Systems and Applications*. Springer. [https://doi.org/10.1007/978-3-642-19460-3\\_3](https://doi.org/10.1007/978-3-642-19460-3_3)
- Malladi, R. M. V., Nizami, A., Mahakali, M. S., & Krishna, B. G. (2019). Cloud Masking Technique for High-Resolution Satellite Data: An Artificial Neural Network Classifier Using Spectral & Textural Context. *Journal of the Indian Society of Remote Sensing*, *47*(4), 661–670. <https://doi.org/10.1007/s12524-018-0892-x>
- Miles, J. (2005). *R Squared, Adjusted R Squared*.
- Mumtaz, G., Akram, S., Iqbal, M. W., Ashraf, M. U., Almarhabi, K. A. L. I., Alghamdi, A. M., & Bahaddad, A. A. (2023). Classification and Prediction of Significant Cyber Incidents ( SCI ) Using Data Mining and Machine Learning ( DM-ML ). *IEEE Access*, *11*(February), 94486–94496. <https://doi.org/10.1109/ACCESS.2023.3249663>
- Muryati, N. (2019). *Analisis Tingkat Kekeringan Lahan Sawah Menggunakan Citra Landsat 8 dan Thermal (Studi Kasus: Kecamatan Gading Rejo, Kabupaten Pringsewu)*. 44–44.
- Mushore, T., Odindi, J., & Mutanga, O. (2022). “Cool” Roofs as a Heat-Mitigation Measure in *Urban Heat Islands: A Comparative Analysis Using Sentinel 2 and Landsat Data*. *Remote Sensing*, *14*(17). <https://doi.org/10.3390/rs14174247>
- Nuha, H. H. (2023). *Mean Squared Error (MSE) dan Penggunaannya*. <https://ssrn.com/abstract=4420880>
- Octarina, T. M., Nyoman, D., Putra, N., Kadek, N., & Wirdiani, A. (2019). *Penginderaan Jauh Pemrosesan Data Satelit Landsat 8 Untuk Deteksi Genangan*. *7*(1).

- Peng, J., Liu, Q., Xu, Z., Lyu, D., Du, Y., Qiao, R., & Wu, J. (2020). How to effectively mitigate *urban heat island* effect? A perspective of waterbody patch size threshold. *Landscape and Urban Planning*, 202. <https://doi.org/10.1016/j.landurbplan.2020.103873>
- Pratiwi, A. Y., & Jaelani, L. M. (2021). Analisis Perubahan Distribusi *Urban Heat Island* (UHI) di Kota Surabaya Menggunakan Citra Satelit Landsat Multitemporal. *Jurnal Teknik ITS*, 9(2). <https://doi.org/10.12962/j23373539.v9i2.53982>
- Prof, A., Ali, M., Vamshi, S., Shiva, S., & Prakash, S. B. (2023). *Virtual Assistant Using Supervised Learning*. 11(VI), 3239–3245. <https://doi.org/10.22214/ijraset.2023.54262>
- Ramadhan. (2023, November 5). *Create Urban Density Index using Spectral Indices and Convolution in Google Earth Engine - YouTube*. <https://www.youtube.com/watch?v=N5lb6rQkLDc>
- Samuel, A. L. (1959). *Some Studies in Machine Learning Using the Game of Checkers*.
- Sun, R., & Chen, L. (2012). How can urban water bodies be designed for climate adaptation? *Landscape and Urban Planning*, 105(1–2), 27–33. <https://doi.org/10.1016/j.landurbplan.2011.11.018>
- Trinufi, R. N., & Rahayu, S. (2020). *R u a n g Analysis of Vegetation and Building Density Changes in the Banda Aceh City Post-Tsunami Disaster*. 6(1), 28–37. <http://ejournal2.undip.ac.id/index.php/ruang/>
- U.S. Geological Survey. (2022). *Landsat Normalized Burn Ratio 2 | U.S. Geological Survey*. <https://www.usgs.gov/landsat-missions/landsat-normalized-burn-ratio-2>
- Wachid, N., & Prananing Tyas, W. (2022). *Analisis Transformasi NDVI dan kaitannya dengan LST Menggunakan Platform Berbasis Cloud: Google Earth Engine* (Vol. 19, Issue 1).
- Wang, Z., Jiang, J., & Long, G. (2022). Positive Unlabeled Learning by Semi-Supervised Learning. *2022 IEEE International Conference on Image Processing (ICIP)*, 2976–2980. <https://doi.org/10.1109/ICIP46576.2022.9897738>
- Whendasmoro, R. G., & Joseph, J. (2022). Analisis Penerapan Normalisasi Data Dengan Menggunakan Z-Score Pada Kinerja Algoritma K-NN. *JURIKOM (Jurnal Riset Komputer)*, 9(4), 872. <https://doi.org/10.30865/jurikom.v9i4.4526>

- Wu, X., Li, B., Li, M., Guo, M., Zang, S., & Zhang, S. (2019). Examining the Relationship Between Spatial Configurations of Urban Impervious Surfaces and Land Surface Temperature. *Chinese Geographical Science*, 29(4), 568–578. <https://doi.org/10.1007/s11769-019-1055-x>
- Zha, Y., Gao, J., & Ni, S. (2003a). Use of normalized difference built-up index in automatically mapping urban areas from TM imagery. *International Journal of Remote Sensing*, 24(3), 583–594. <https://doi.org/10.1080/01431160304987>
- Zha, Y., Gao, J., & Ni, S. (2003b). Use of normalized difference built-up index in automatically mapping urban areas from TM imagery. *International Journal of Remote Sensing*, 24(3), 583–594. <https://doi.org/10.1080/01431160304987>

## LAMPIRAN

### 1. Inisialisasi prediksi

#### a. Persamaan

$$F_0(x) = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^n L(y_i, \gamma) \quad (3.9)$$

Dimana:

$L(y_i, \gamma)$  = Loss function antara nilai prediksi ( $y_i$ ) dan nilai aktual pada sampel ke- $i$  ( $\gamma$ ).

#### b. Perhitungan manual

$$F_0(x) = -40,163 + \gamma - 39,746 + \gamma - 40,655 + \gamma$$

$$F_0(x) = -120,564 + 3\gamma$$

$$3\gamma = -120,564$$

$$\gamma = \frac{-120,564}{3} = 40,188$$

$$F_0(x) = 40,188$$

#### c. Hasil inisialisasi prediksi

Tabel 3.7 Nilai inisialisasi prediksi

NBR2	...	elevation	UDI	LST	Inisialisasi prediksi $F_0(x_i)$
-0.042	...	491.000	0.931	40.163	40.188
-0.065	...	466.000	0.966	39.746	40.188
-0.018	...	424.000	0.931	40.655	40.188

### 2. Kalkulasi residual

#### a. Persamaan

$$r_{im} = - \left[ \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)} \quad \text{for } i = 1, \dots, n \quad (3.14)$$

Dimana:

$r_{i,m}$  = Residual untuk sampel ke- $i$  pada pohon ke- $m$

$- \left[ \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)}$  = turunan dari loss function yang berhubungan dengan turunan dari inisialisasi prediksi  $\gamma$ .

b. Perhitungan manual

$$L(y_i, \gamma) = -y_i + \gamma$$

$$\frac{\partial L}{\partial \gamma} = -(y_i - \gamma)$$

$$-\frac{\partial L}{\partial \gamma} = (y_i - \gamma) \tag{3.15}$$

$$r_{1,1} = (40,163 - 40,188) = -0,025$$

$$r_{2,1} = (39,746 - 40,188) = -0,442$$

$$r_{3,1} = (40,655 - 40,188) = 0,467$$

c. Hasil perhitungan residual

Tabel 3.8 Hasil kalkulasi *residual*

NBR2	...	elevation	UDI	LST	Inisialisasi prediksi $F_0(x_i)$	$R_{i,m}$
-0.042	...	491.000	0.931	40.163	40.188	-0.025
-0.065	...	466.000	0.966	39.746	40.188	-0.442
-0.018	...	424.000	0.931	40.655	40.188	0.467

### 3. Membuat pohon keputusan

a. Persamaan

$$\gamma_{j,m} = \underset{x_i \in R_{ij}}{\operatorname{argmin}} \sum L(y_i, F_{m-1}(x_i) + \gamma) \tag{3.15}$$

Dimana:

$\gamma_{jm}$  = Nilai gamma untuk satu atau beberapa sampel pada *leaf* ke-j dalam *tree* ke-m

$y_i$  = Nilai aktual sampel ke-i

$F_{m-1}(x_i)$  = Prediksi sebelumnya

$\gamma$  = Nilai gamma yang dicari

b. Perhitungan manual

$$\gamma_{1,1} = \underset{x_i \in R_{2,1}}{\operatorname{argmin}} \sum L(y_i, F_{1-1}(x_i) + \gamma)$$

$$\gamma_{1,1} = \underset{\gamma}{\operatorname{argmin}} \frac{1}{2} (y_1 - F_0(x_1) + \gamma)^2$$

$$\gamma_{1,1} = \underset{\gamma}{\operatorname{argmin}} \frac{1}{2} (40,163 - (40,188) + \gamma)^2$$

$$\gamma_{1,1} = \frac{2}{2} (-0,025 + \gamma)(-1)$$

$$\gamma_{1,1} = -0,025 - \gamma$$

$$\gamma_{1,1} = -0,025$$

$$\gamma_{2,1} = \operatorname{argmin}_{x_i \in R_{2,1}} \sum L(y_i, F_{1-1}(x_i) + \gamma)$$

$$\gamma_{2,1} = \operatorname{argmin} \frac{1}{2} (y_2 - F_0(x_2) + \gamma)^2$$

$$\gamma_{2,1} = \operatorname{argmin} \frac{1}{2} (39,746 - (40,188) + \gamma)^2$$

$$\gamma_{2,1} = \frac{2}{2} (-0,442 + \gamma)(-1)$$

$$\gamma_{2,1} = -0,442 - \gamma$$

$$\gamma_{2,1} = -0,442$$

$$\gamma_{3,1} = \operatorname{argmin}_{x_i \in R_{2,1}} \sum L(y_i, F_{1-1}(x_i) + \gamma)$$

$$\gamma_{3,1} = \operatorname{argmin} \frac{1}{2} (y_3 - F_0(x_3) + \gamma)^2$$

$$\gamma_{3,1} = \operatorname{argmin} \frac{1}{2} (40.655 - (40,188) + \gamma)^2$$

$$\gamma_{3,1} = \frac{2}{2} (-0,467 + \gamma)(-1)$$

$$\gamma_{3,1} = 0.467 - \gamma$$

$$\gamma_{3,1} = 0.467$$

c. Hasil

Tabel 3.9 Hasil proses prediksi pada iterasi 1

UDI	LST	Inisialisasi prediksi $F_0(x_i)$	$R_{i,m}$	$Y_{j,m}$
0.931	40.163	40.188	-0.025	-0,025
0.966	39.746	40.188	-0.442	-0,442
0.931	40.655	40.188	0.467	0,467

#### 4. Pembaharuan prediksi

a. Persamaan

$$F_m(x) = F_{m-1}(x) + V \sum_{j=1}^{J_m} \gamma_{j,m} I(x \in R_{j,m}) \quad (3.16)$$

Dimana  
 $F_{m-1}(x)$  = prediksi inisialisasi  
 $V$  = learning rate

$\sum_{j=1}^{J_m} \gamma_{j,m} I(x \in R_{j,m}) = \text{total nilai gamma terbaik dari seluruh prediksi tree yang dijumlahkan}$   
 $(\gamma_{j,m})$

b. Perhitungan manual

$$F_m(x) = F_0(x) + V \times \gamma_{1,1}$$

$$F_1(x_1) = 40,188 + 0,1 \times -0,025$$

$$F_1(x_1) = 40,185$$

$$F_m(x) = F_0(x) + V \times \gamma_{2,1}$$

$$F_1(x_2) = 40,188 + 0,1 \times -0,442$$

$$F_1(x_2) = 40,143$$

$$F_m(x) = F_0(x) + V \times \gamma_{2,1}$$

$$F_1(x_3) = 40,188 + 0,1 \times 0,467$$

$$F_1(x_3) = 40,234$$

c. Hasil

Tabel 3.10 Hasil proses prediksi pada iterasi 1

UDI	LST	Inisialisasi prediksi $F_0(x_i)$	$R_{i,m}$	Prediksi 1 $F_1(x_n)$
0.931	40.163	40.188	-0.025	40.185
0.966	39.746	40.188	-0.442	40.143
0.931	40.655	40.188	0.467	40.234